New Energy Management Systems for Battery Electric Vehicles with Supercapacitor

by

Hoda Marefat

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Masters of Applied Science
in
Systems Design Engineering Department

Waterloo, Ontario, Canada, 2019

AUTHOR'S DECLARATION

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

I declare that all the models and controllers presented in this thesis were developed by me. This thesis contains some material that was taken from multi-author papers including:

- Hoda Marefat, Mehdi Jalamaab, and Nasser L. Azad,” Energy Energy Management of Battery Electric Vehicles Hybridized with Supercapacitor using Stochastic Dynamic Programming”, *SICE International Symposium on Control Systems, IEEE, March 9-11, 2018, Setagaya Campus, Tokyo City University, Tokyo, Japan*

Contribution of Mehdi Jalalmaab: *development of SDP solver*

- Hoda Marefat, Sadegh Tajeddin, and Nasser L. Azad, “A Newton/Generalized Minimal Residual Approach to Model Predictive Control of Battery Electric Vehicles Hybridized with a Supercapacitor” (Not submitted)

Contribution of Sadegh Tajeddin: adaptive strategy for control development

Abstract

Recently, the Battery Electric Vehicle (BEV) has been considered to be a proper candidate to terminate the problems associated with fuel-based vehicles. Therefore, the development and enhancement of the BEVs have lately formed an attractive field of study. One of the significant challenges to commercialize BEVs is to overcome the battery drawbacks that limit the BEV's performance.

One promising solution is to hybridize the BEV with a supercapacitor (SC) so that the battery is the primary source of energy meanwhile the SC handles sudden fluctuations in power demand. Obviously, to exploit the most benefits from this hybrid system, an intelligent Energy Management System (EMS) is required.

In this thesis, different EMSs are developed: first, the Nonlinear Model Predictive Controller (NMPC) based on Newton Generalized Minimum Residual (Newton/GMRES) method. The NMPC effectively optimizes the power distribution between the battery and supercapacitor as a result of NMPC ability to handle multi-input, multi-output problems and utilize past information to predict future power demand. However, real-time application of the NMPC is challenging due to its huge computational cost. Therefore, Newton/GMRES, which is a fast real-time optimizer, is implemented in the heart of the NMPC. Simulation results demonstrate that the Newton/GMRES NMPC successfully protects the battery during high power peaks and nadirs.

On the other hand, future power demand is inherently probabilistic. Consequently, Stochastic Dynamic Programming (SDP) is employed to maximize the battery lifespan while considering the uncertain nature of power demand. The next power demand is predicted by a Markov chain. The SDP approach determines the optimal policy using the policy iteration algorithm. Implementation of the SDP is quite free-to-launch since it does not require any additional equipment. Furthermore, the SDP is an offline approach, thus, computational cost is not an issue. Simulation results are considerable compared to those of other rival approaches.

Recent success stories of applying bio-inspired techniques such as Particle Swarm Optimization (PSO) to control area have motivated the author to investigate the potential of this algorithm to solve the problem at hand. The PSO is a population-based method that effectively seeks the best answer in the solution space with no need to solve complex equations. Simulation results indicate that PSO is successful in terms of optimality, but it shows some difficulties for real-time application.

Acknowledgements

I would like to express my sincere appreciation to Prof. Nasser L. Azad, the most supportive, erudite, and kind-hearted supervisor I have ever seen, whose guidance and constructive comments were invaluable to this investigation.

I would also like to thank my colleagues at the SHEV lab; it was fantastic to work with them in such a friendly and enthusiastic environment. Very special gratitude goes out to Parisa Golchoubian for her continuous support and assistance.

I have greatly appreciated the worthwhile comments and help of Mary McPherson from the UW Writing and Communication Centre.

Specifically, uniquely, and zealously, I thank my only son, Taha, who was born during this research. Although a baby, his adjustability, understanding, and tolerance have made this thesis possible. I am deeply indebted to him for his patience during this research.

I express my warm gratitude to my encouraging and wonderful parents for everything.

Lastly but most importantly, I truly admire my best friend and husband, Mahdi, for his unfailing support.

Dedication

To those who give the most meaning to my life: my lovely husband, Mahdi; my little one, Taha; and my wonderful parents.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

Governmental and state agencies aim to reduce vehicles fuel consumption due to its negative outcomes such as pollution and global warming [1]. Global interest in decreasing fuel consumption has led to emergence different kinds of Electric Vehicles (EVs). In one type, Battery Electric Vehicles (BEVs), the battery plays a major role in providing the power demanded by the driver.

Developing more environmentally friendly and fuel-efficient vehicles can be achieved by two different approaches: developing more-advanced automotive hardware, and better control policies [2]. Since the latter is much more cost-effective, more researchers focus on it [3]. Developing more-intelligent controllers that reduce the range anxiety promote EVs commercialization [4]. In particular, Plug-in Hybrid Electric Vehicles (PHEVs) and Hybrid Electric Vehicles (HEVs) in which there are more than one energy source, require an advanced, real-time implementable controller.

1.1 Motivations and Challenges

As mentioned, one of the highly promising solutions to reduce vehicles fuel consumption is Battery Electric Vehicles (BEVs). Unfortunately, there are some major practical problems for BEVs' widespread applications, such as limited battery life, high cost, and also a short

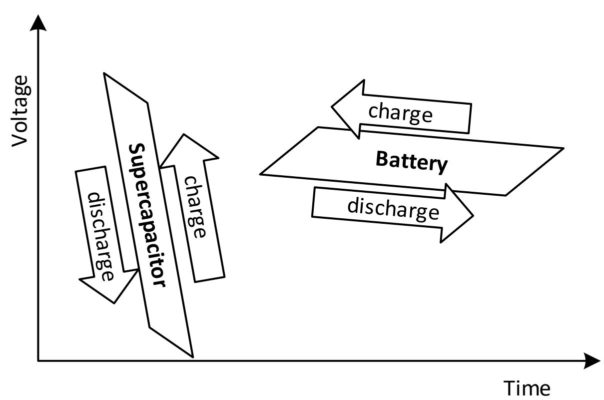

Fig. 1.1. Discharge and charge rate of the battery and supercapacitor [6]

Table 1-1 Comparing battery characteristics with SCs [5]

	Energy Density (kWh)	Power Density (kW)	Number of Cycles at 80% DOD
Li-Ion	50-80	1000-4000	3000
Supercapacitor	1-5	1000-30000	>1,000,000

driving range [5]. Automotive researchers have investigated many different strategies to extend the lifespan of the battery [5]. Some of them have suggested combining the battery with a Supercapacitor (SC) to compensate for BEVs drawbacks [7], [8].

With the current technology, there are various limitations in battery performance. First, the battery internal resistance converts a portion of energy to heat during charging or discharging cycles. Second, the battery capacity depends on the charge/discharge rate. Third, as shown in Fig. 1.1, the charge/discharge process of the battery is quite slow. Fourth, the battery cannot handle a large number of charging/discharging cycles.

In contrast, SCs do not have these deficiencies. Most importantly, they have much lower internal resistances, in other words, Peukert's Law does not affect SCs [5]. Furthermore, they can handle a great number of charging or discharging cycles efficiently. Also, as Fig. 1.1 shows, the supercapacitor can be charged/discharged much more quickly than a battery. Also, the implementation of a combined SC-battery system decreases the current required from the battery; therefore, the total heat loss will be reduced, which in turn increases the lifespan of the battery.

In current BEVs, the battery is the only source for providing the demanded power by the driver. BEVs require the high density levels of both energy and power for deceleration/accelerations to satisfy the expected operation rate, so, meeting these conditions by only using a battery leads to producing a costlier battery. Table 1 shows higher power density for SCs than for batteries. SCs have a longer life than batteries as well [9]. Also, SCs

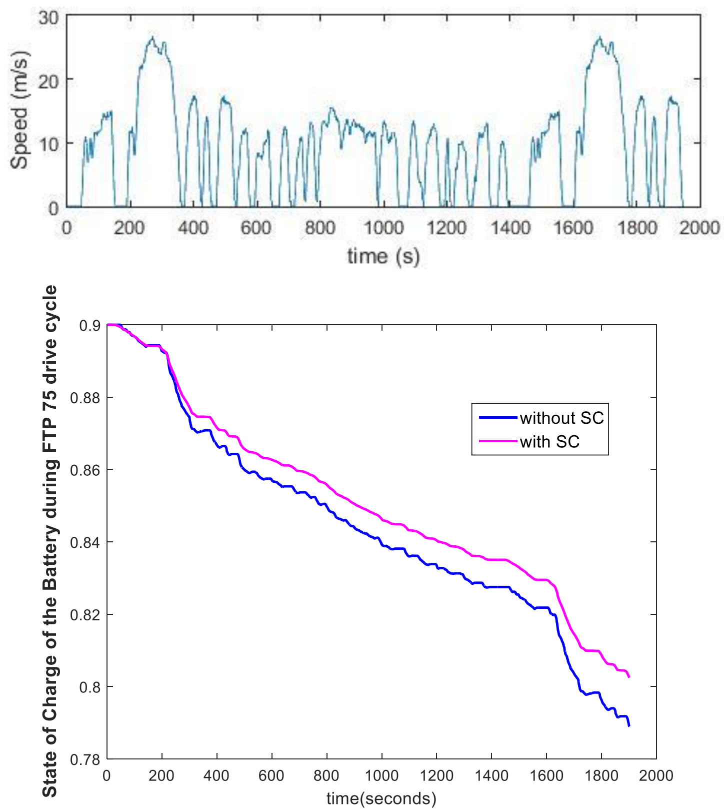

Fig. 1.2 Maximum potential benefits of a battery-SC hybrid storage system [5]

have a great energy density and can hold a charge for long periods [5]. Thus, in BEVs hybridized with SC, the required high power density and energy density can be provided by the SCs and batteries, respectively.

The potential advantage of using SCs in BEVs has been investigated previously. For example, as Fig. 1.2 shows, Golchoubian et al. [5] have addressed the maximum benefit of SC-battery hybrid storage system for a certain drive cycle. Styler et al. [7] have studied the impact of the SC-battery storage system on vehicle driving performance. Moreover, some researchers at Carnegie Mellon University have created the Charge-Car project [10], which intends to make BEVs less expensive by applying the SC-battery storage system as a practical solution. Carter et al. proposed a rule-based control (RBC) method [11], which is efficient in terms of computational time, but does not guarantee the optimality. Choi et al.

have suggested an optimization-based approach [12]. L. Cheng et al. proposed the hybrid battery/SC system for light rail vehicles, and solved the Energy Management

System (EMS) optimization problem by a simplified optimization method to reduce the computational cost [13].

These promising reports have motivated the author to exploit the potential benefits of combining SCs with batteries by designing an optimal EMS [5]. As shown in Fig. 1.3, the EMS determines the power distribution between the battery and the supercapacitor at each moment. However, in designing an efficient EMS, there are some big challenges: next power demand prediction, solution optimality, computational cost and so on.

In this thesis, four EMSs for the Toyota Rav4EV based on fundamentally different approaches are proposed: Model Predictive Control (MPC), Dynamic programming, Stochastic Dynamic Programming (SDP), and Particle Swarm Optimization (PSO).

1.1.1 Model Predictive Control (MPC)

Among various approaches proposed in the literature, the MPC algorithm seems to be a promising one for online optimization. It has demonstrated superior capability to deal with several inputs and outputs, satisfying constraints on states, updating the solution using current observations at each moment, and real-time optimization makes this controller a superior [14]. In the literature, numerous publications have shown the effectiveness of the MPC for different control applications, for instance, Ecological Adaptive Cruise Controller [15], traffic-information integration with the MPC [16], and improvements of fuel economy, safety, and comfort [17].

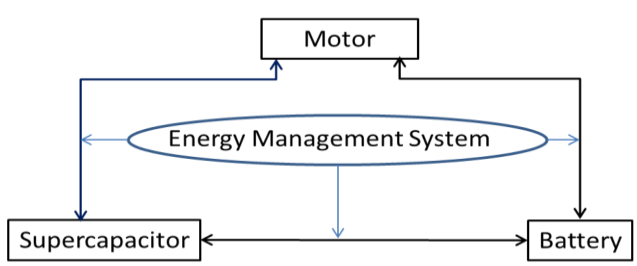

Fig. 1.3 Design of a combined SC-battery system [5]

The author has applied the Nonlinear Model Predictive Control (NMPC) to the problem in hand since the NMPC does not include the linearization errors of the Linear Model Predictive Control (LMPC) and yields a more accurate solution [18]. Although the NMPC imposes much more computational cost, recent advances in the computational hardware make it more implementable [14]. However, the NMPC remains a challenging method in terms of computational burden.

1.1.2 Dynamic Programming (DP)

In 1957, Bellman introduced the DP approach [19]. This method solves complex problems by breaking them down into smaller, simpler sub-problems and solves these sub-problems recursively. The DP methodology is widely used in different areas, from control problems to economics [5], [20]. The DP tries all the possible solutions and returns the best one as the final answer. So, if the discretization number is chosen properly, it returns the global optimum. DP is a very effective method for optimization. However, since it calculates the cost of all of the possible routes, it entails a huge computational expense. In fact, it is an offline method. In this thesis, the maximum potential of the proposed EMS is investigated by DP. The DP solution provides an effective basis for evaluating the performance of the other proposed methods.

1.1.3 Stochastic Dynamic Programming (SDP)

Obviously, real-world driving involves a lot of uncertainties such as behavior of nearby cars, slip ratio of streets, and so on. As a result, the power demanded by a driver is highly nondeterministic. Consequently, an EMS should be able to handle the probabilistic nature of the power demand. Many researchers have proposed stochastic strategies for BEVs.

So, the author has been motivated to solve this EMS design problem by SDP. In fact, the main advantage of the SDP method is that unlike many other techniques that consider the demanded power pre-defined values, the power demand is assumed to be unknown and stochastically changing [21]. In the SDP chapter, the control problem in hand is converted into SDP form, also, the power demand has been predicted based on a Markov chain assumption using some real drive cycles data points. The used drive cycles are categorized in two groups, which are training drive cycles and test ones. The Transition Probability Matrix (TPM) is built by the training cycles; meanwhile simulation results are based on the test drive cycles.

1.1.4 Particle Swarm Optimization (PSO)

In 1995, Dr. Eberhart and Dr. Kennedy introduced the PSO algorithm. This algorithm is inspired by the behavior of flock of birds or school of fish. This bio-inspired algorithm tries to optimize the objective function with no need to solve mathematical equations. It is initialed with a population of random particles. In the next iterations, each particle moves according to its velocity. The velocity of each particle is calculated based on both its own personal best position and the best value the population has obtained so far [22].

PSO does not have many complicated calculations, only a few simple updating formulas. As a result, it can search the solution area quite quickly. This characteristic makes the PSO a suitable candidate for online control optimization problems.

In this thesis, the PSO is applied for designing a novel EMS. PSO parameters are tuned accordingly to improve the accuracy and convergence speed. The performance of the PSO-based EMS is evaluated in the simulation section using a control-oriented model.

1.2 Problem Statement and Proposed Approach

The purpose of this study is to propose an efficient EMS for a BEV hybridized with SC, in order to maximize the battery lifespan. This controller should perform more efficiently than other EMSs in the literature. Designing an EMS includes the following steps:

Developing a control-oriented model, predicting the next power demand by the driver, designing a controller that maximizes the battery lifespan, implementing a nonlinear programming optimizer in the heart of the controller, and evaluating a proposed EMS.

1.3 Thesis Layout

The rest of the thesis is organized as follows. Chapter 2 reviews the relevant literature and introduces fundamental concepts applied through the thesis. In Chapter 3, a control-oriented model for the vehicle system is developed. Chapter 4 applies the Newton/GMRES-based NMPC to the problem, followed by Chapter 5 in which the problem is solved by the DP method. Chapter 6 depicts the problem at hand in the framework of SDP and compares the results to those for rival approaches. Chapter 7 presents the PSO algorithm, describes the control problem in the PSO form, and provides the simulation results. Finally, Chapter 8 concludes and suggests future work.

Chapter 2

Literature Review and Background

This chapter reviews the literature on various EMS control strategies, and also, briefly describes some basic concepts used in this investigation. Some commonly used architectures for a BEV hybridized with SC are addressed and the selected one is validated. Plus, the bases of the model predictive control and the stochastic dynamic programming are explained.

2.1 BEV Hybridized with Supercapacitor (BEV-HSC) Topology

In this publication, topology means a pattern by which the battery, supercapacitor, converter and other devices of the engine are connected. Many researchers have proposed topologies for a BEV Hybridized with Supercapacitor (BEV-HSC). From these available topologies, the author has chosen the one with sufficient degrees of freedom to allow various designs to be implemented. The selected topology also effectively balances circuit complications, accuracy and computational cost. A review of most popular architectures are described in the following [23].

Fig. 2.1 shows the most basic hybridizing architecture in which there is no converter or inverter to connect the battery and SC. In this architecture, the supercapacitor plays a low-pass filter (LPF) role. The DC, the battery, and the SC are parallel, and consequently, all have the same voltage. This strategy is quite cost-effective, but it is unable to effectively utilize the supercapacitor as shown in [23] .

Fig. 2.2 illustrates a Supercapacitor/Battery architecture, in which a bidirectional DC/DC converter is placed between the battery/DC bus and the SC, so, the SC's voltage can vary

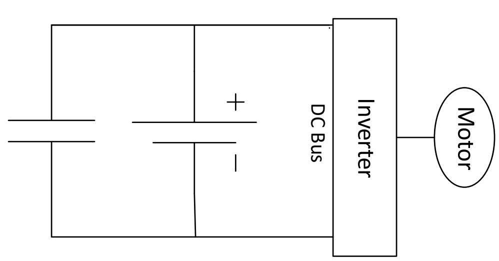

Fig. 2.1. Basic Supercapacitor/Battery architecture

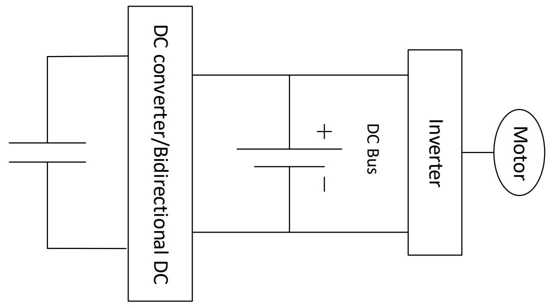

Fig. 2.2 Supercapcitor/Battery architecture

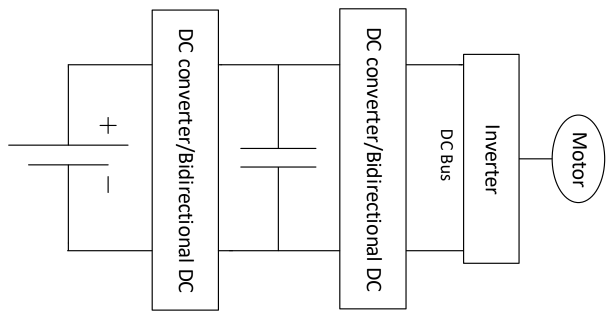

Fig. 2.3 Cascaded architecture

within a large range. In this topology, the converter should be bigger to handle a supercapacitor's voltage. Since the battery is directly linked to the DC/bus, the DC/bus voltage cannot change. This strategy is the one most applied in hybrid energy storage systems [23], [24]. In addition, it provides enough degree of freedom to implement different control policies. Moreover, since there is only a single converter, the circuit complexity and cost are quite low.

Placing the second bidirectional DC/DC converter between the supercapacitor and the DC bus results in a new architecture, shown in Fig. 2.3. This topology, the so-called cascaded configuration, extends the supercapacitor's working range, but implementing the second converter imposes additional costs.

In another design, known as the multiple converter configuration, one converter is paralleled with the battery, the other one with the SC, as in Fig. 2.4. Meanwhile both converters are paralleled with each other. This architecture allows the voltage of both the supercapacitor and battery to change more freely. In addition, the charge saved with the

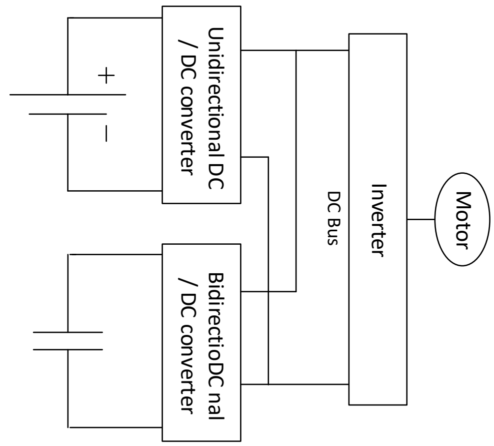

Fig. 2.4 Multiple converter architecture

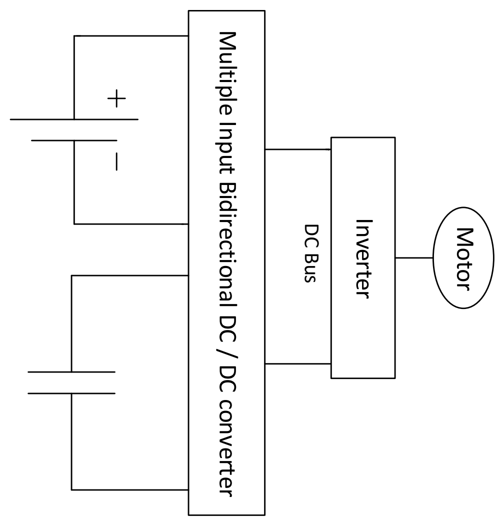

Fig. 2.5 Multiple input converter architecture

supercapacitor can be completely utilized. However, the implementation of this topology is quite costly since it requires two complete converters [23].

In order to decrease the expense of two full-sized converters in the multiple converter design, Napoli et al. suggested a multiple input converter configuration, as shown in Fig. 2.5 [25].

2.2 Energy Management System (EMS)

Although hybridizing BEVs with supercapcitors is considered to be an effective approach, exploiting the maximum potential of this hybrid system requires an efficient EMS. The EMS determines the optimal power distribution between the battery and the SC at each moment in order to promote battery health and lifespan.

Many researchers have proposed different EMSs for BEVs hybridized with SC, such as rule-based and optimization-based control. Several rule-based control (RBC) approaches have been addressed in[23],[11]. RBC strategies are quick enough for real-time applications, but they have shown some difficulties in finding the optimum input. Trovao et al. designed a rule-based controller in which the battery minimum and charging power are changeable [26]. In [27], offline control methods based on dynamic programming (DP) are presented. DP finds the global optimum, but since it assumes perfect knowledge of the future and requires huge computational time and memory, it is only an off-line optimization method and cannot be used for real-time applications. Song et al. have utilized a DP method to promote RBC optimality [28]. In [5], the maximum benefit of hybridizing BEVs with SC is investigated using DP for the Toyota Rav4EV.

In [29], a novel optimization-oriented approach is presented and proved to be quick enough for online applications. In dealing with unknown situations, this approach shows similar performance to that of RBC. Ortuzar et al. have designed an EMS by developing a neural network for a battery/SC hybrid system and trained the network using several databases [24].

Many researchers have applied the MPC method for energy management of the BEV-HSC. The MPC has efficiently solved different types of control problems [1], [30] . Hrezack et al. have developed a linear MPC (LMPC) approach and validated this approach experimentally [31]. The LMPC predicts the next demanded power using an estimator; and linearizes the control problem to reduce computational cost. Song et al. have compared two RBC methods, a fuzzy-based controller and the LMPC, assuming there is no information about the next demanded power [28]. Their simulation results show that the performance of the RBC is superior to that of the LMPC; the performances of the fuzzy-based controller and RBC do not noticeably differ.

Recently, stochastic controllers have been used to predict the next power demand [32]. In [33], the authors have converted the energy management problem to a stochastic planning

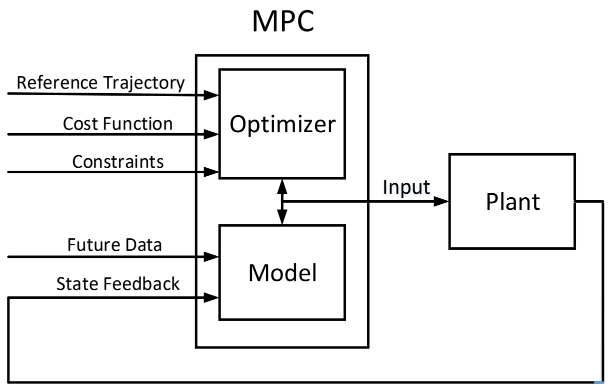

Fig. 2.6: MPC Block Diagram

one and solved it using reinforcement learning, computational sustainability, regression and so on. Laldin et al. have predicted future load demands using a Markov chain [34]. They have defined a limited number of states by defining ratio of velocity over acceleration based on a few drive cycles and applied Markov chain concept to find a solution.

Not long ago, bio-inspired methods were found that can efficiently solved MPC problems and satisfyingly meet the constraints of control problems [35]. Zou et al. have designed an MPC using Particle Swarm Optimization (PSO) for a greenhouse climate control problem [36]. In [35], Susuki et al. used PSO for automatic tuning of MPC parameters.

2.2.1 Model Predictive Control (MPC)

MPC is an efficient approach that exploits the potentials of cutting-edge optimization concepts and satisfies automotive necessities, due to its capability of solving optimal control problems with multiple inputs and outputs [38]. Many researchers have addressed application of this controller for EMS of HEVs. For instance, Wang has designed an online controller for several hybrid architectures based on the MPC [39]. In [40], Nonlinear MPC (NMPC) has been applied to optimizes fuel economy while considering constraints on battery, vehicle, and location. Other researchers have utilized an MPC to compute hybrid EV power-split ratio. They have applied NMPC, LMPC, and RBC methods using the PSAT software. Simulation results have shown superiority of the NMPC [41],[42].

MPC is a repetitive optimization process of a constrained control problem as it is shown in Fig. 2.6. Usually, an MPC-based controller is designed on a simplified yet accurate enough model of a plant, so-called control-oriented model. It utilizes information of the previous states and control actions to optimize a performance index of the limited prediction horizon [43].

The NMPC is a nonlinear version of the MPC, in which a nonlinear control-oriented model is utilized to predict future variables [44]. The NMPC technique is an effective approach, especially if the power demand is predicted accurately. However, this method has shown some difficulties for online applications due to its high computational cost. Recent developments in advanced computational equipment have encouraged many researchers to implement the NMPC method as the heart of computationally-efficient controllers [45],[46], [47]. Several fast nonlinear controllers have been developed, such as Generalized Minimum Residual (GMRES) [48], Shooting-based Newton [49], and so on. This thesis proposes the NMPC using a Newton/GMRES approach.

2.2.2 Dynamic Programming (DP)

DP is an effective optimization methodology that finds the global optimum solution. Since it comes at a huge computational cost, it is usually not practical for real-time applications. However, it is widely utilized for control problems by some researchers. In [50], [51] the authors improved the fuel efficiency by extracting near-optimal rules for the EMS of parallel HEVs using a DP global optimum. Based on this strategy, Lin et al. found an optimal power split between the engine and electric motor. In the other investigation, Gong et al. [52] enhanced fuel economy by applying DP to reinforce the charge-depletion policy to manage SOC drops in a PHEV using trip information. O'Keefe et al. [53] derived a near-optimal control strategy using DP and implemented the DP results as a basis of assessment for other approaches. They demonstrated that in optimum battery charge split policies, the state of charge reaches the lower limit at the last moment of the trip. In [54], a DP-oriented method is implemented to compute the optimum control in the framework of MPC while GPS and ITS provide incomplete information of the next power demands.

2.2.3 Stochastic Dynamic Programming (SDP)

In EMS control problems, predicting the next power demand is a challenge since uncertainties are inevitable parts of driving. Therefore, stochastic decision making schemes have successfully been used for solving automotive EMS problems where their effectiveness are shown for dealing with future power demand uncertainties [32]. For instance, Golchoubian et al. have designed a Stochastic Nonlinear Model Predictive Controller (S-NMPC) for SC-battery storage systems through applying a two-stage version of stochastic programming and demonstrated significant performance improvements [18]. Although this approach sounds promising due to its near-optimum solution, there are some difficulties in its real-time implementation because of high computational costs. Ermon et al. have formulated a hybrid capacitor/battery EMS logic as a stochastic problem that considers the probabilistic nature of power demand. Their newly proposed approach is based on a combination of optimization, data-mining, and machine learning which optimizes battery usage [33]. Zhang et al. have optimized an EMS logic for PHEVs, considering probabilistic drive-cycles, and proposed a stochastic drive cycle scheme by applying SDP to promote vehicle performance [55]. Opila et al. have investigated EMS control designs based on shortest path SDP (SP-SDP). The controllers have been tested on a Ford automobile using many real driving cycles. Simulation results have indicated that the SP-SDP controllers increase performance by 2-3% compared to the performance of a Ford controller for a prototype automobile [56].

Evaluating stochastic approaches requires generating probabilistic drive cycles. Schewarzer et al. have designed a method to create driving cycles from stochastic driving profiles. This methodology has been proposed to stochastically optimize EMS controllers in EVs [57].

In this thesis, the author has solved the problem in hand by the SDP approach and demonstrated successful simulation results. In this investigation, the power demand has been predicted based on a Markov chain assumption using some real drive-cycle data points. The used drive cycles are categorized in two groups, training drive cycles and test ones. The Transition Probability Matrix (TPM) is built by the training cycles; meanwhile simulation results are based on the test drive cycles. In comparison to the results of other methods, the

SDP results show more improvements. In addition, in terms of computational costs, it has a significant advantage over the rival approaches.

2.2.4 Particle Swarm Optimization

The other method applied in this thesis is the Particle Swarm Optimization (PSO) approach. PSO is a bio-inspired and population-based algorithm that seeks the best answer in the solution area. It begins with a number of particles that randomly are located in the solution area and tries to converge to an optimum point through a number of iterations. This approach is inspired by the way flocks of birds or schools of fish search for food. This method has a number of advantages as follows:

First, PSO can be used in either engineering applications or scientific investigations since it is based on learning (intelligence). Second, this algorithm has a simple structure. Unlike many other evolutionary computation algorithms (such as ant colony), there is no need to adjust many parameters at the beginning. In other words, this algorithm does not involve mutation or overlapping parameters. In the process of developing several generations, only the best particle spreads information to other particles. Third, it has no complicated calculations, but only a few simple updating formulas; as a result, it searches the solution area very quickly. Fourth, for a unimodal function, it converges quickly to the optimum point if the parameters are tuned correctly. Next, there is no need for the function to be differentiable; only fitness values are required. This characteristic makes the algorithm applicable to a large range of objective functions. Finally, PSO is less dependent on the initial points compared to other heuristic methods. Consequently, convergence of the method is quite robust [58], [59].

PSO has been successfully applied to many different control problems. For instance, Coelho et al. proposed a model-free learning adaptive control approach that applied PSO for optimization [60]. In [61], the authors have designed a new version of PSO that utilizes Gaussian and Cauchy distributions to generate random values. Applying Cauchy random numbers helps PSO to escape from local optimums, and applying Gaussian ones speeds up the convergence process. This modified PSO has satisfyingly optimized generalized predictive controllers parameters. In [62], PSO has been applied for optimization procedure of MPC for nonlinear processes. Simulation results have shown PSO-based MPC to be very robust. Han et al. implemented a feedforward neural network at the heart of the MPC and tuned the parameters of the neural network by using PSO [63].

In this investigation, PSO is implemented to optimize the power distribution between the battery and the supercapacitor at each moment.

Chapter 3
System Modeling

This chapter presents the control-oriented model of the Toyota Rav4EV. First, longitudinal dynamics of Rav4 EV is investigated, The next section depicts the problem in hand from the mathematical point of view. It presents and explains the objective function, constraints of the problem, and other related formulas. Moreover, it shows how the state of charge of the battery (SOC_{batt}) and the supercapacitor (SOC_{SC}) are updated at each moment; and how we can optimize this system by playing with our control input (r).

3.1 Longitudinal Dynamics of Toyota Rav4 EV

In order to evaluate different proposed EMSs in this thesis, they should be tested for standard drive cycles such as Federal Test Procedure (FTP75), Urban Dynamometer Driving Schedule (UDDS) and so on. These drive cycles show the velocity over time. Since the input of EMS is the power demand (P_{dem}), the velocity should be converted to P_{dem}. Converting the speed to the corresponding P_{dem} depends on the characteristics of each EV.

The parameters of this vehicle are presented in Table 3.1 Barta et al. [64] derived these parameters by vehicle road tests and created a front-drive chassis model that was used as the powertrain model in the Maplesim simulation environment. The motor torque and power were calculated based on the longitudinal dynamics of the Rav4EV. Then, the model was

Table 3-1 Longitudinal dynamics of the Rav4EV

Parameter	Value
Vehicle mas (M)	1970 kg
Frontal area (A)	2.464 m^2
Wheel radius (r)	0.355 m
Air density (ρ)	1.29 $\mathrm{kg/m}^3$
Drag coefficient (D_c)	0.3
Rolling resistance coefficient (R_c)	0.015
Gear ratio (G)	0.973

transferred to the Simulink environment. More information about this model can be found in [64]. According to Table 3.1, for a Rav4EV moving at speed $v(t)$ and slope $\alpha(t)$, the resistance force (F_R) is calculated as follows:

$$F_R(t) = \frac{1}{2}\rho D_c A v(t)^2 + R_c M g + M g sin\big(\alpha(t)\big), \tag{3.1}$$

The motor generates the force and transfers it to wheels through the gearbox. The demanded torque at each moment, $\tau(t)$, is calculated as follows:

$$\tau(t) = \big(F_R(t) + M v'(t)\big)\left(\frac{r}{G}\right), \tag{3.2}$$

All of the variables are defined in Table 3.1. Consequently, the demanded power is computed as:

$$\begin{gathered} If\ \tau(t) \geq 0, \qquad P_{dem}(t) = \frac{\tau(t)\omega(t)}{E_a}, \\ If\ \tau(t) < 0, \qquad P_{dem}(t) = \ \tau(t)\omega(t)E_d, \\ \omega(t) = \frac{v(t)G}{r}. \end{gathered} \tag{3.3}$$

$\omega(t), E_a, and\ E_d$ refer to rotational velocity, efficiency while acceleration, and efficiency while deceleration, respectively. According to [65], $E_a = 0.85$ and $E_d = 0.35$.

3.2 The Topology of BEV Hybridized with the SC

Fig. 3.1 depicts the used technology in this investigation. The supercapacitor is connected by a DC converter/bidirectional DC with the battery/DC bus, so, the supercapacitor voltage can change within a large range. The battery is directly linked to the DC bus, as a result, the DC bus voltage does not change noticeably. This topology provides large enough freedom degrees so that different EMSs can be applied. Since there is only one converter in the system, it fairly balances the circuit complexity, efficiency, and expense. This topology is widely used in BEVs hybridized with SC [65].

As mentioned before, the demanded power by the driver is provided by both the battery and the SC, i.e.

$$P_{dem} = P_{batt} + E_c P_{SC}. \tag{3.4}$$

$P_{dem}, P_{batt}, P_{SC}$, and E_c refer to Power demand, power of the battery, power of the supercapacitor and efficiency of the converte,r respectively.

Power loss of the DC bus is usually negligible, in other words, E_c is assumed to equal one in this model. Since the converter structure does not have a noticeable effect on the EMS [66], it is not addressed in detail in this study. Since the motor and its inverter effect on P_{dem} [65], do not influence the proposed control-based model, their structures are not described in detail in this study.

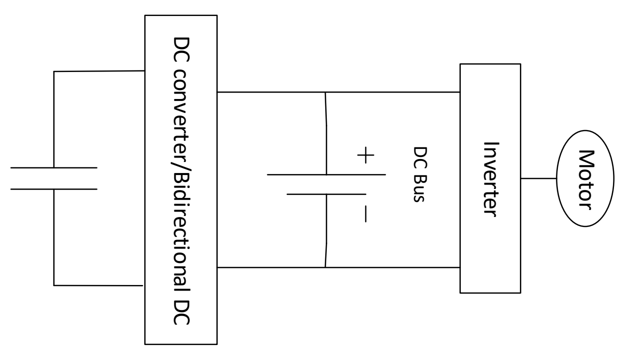

Fig 3.1. The EMS topology

3.3 Control-oriented model

The considered problem seeks an optimal energy management strategy for the combined supercapacitor-battery storage system of a given BEV. In this investigation, the vehicle's motor can receive energy from the both supercapacitor and battery. Because of the aforementioned technical reasons, the battery should be used less whenever possible. As a result, the objective function is defined as follows:

$$Z = \sum I_{batt}^2. \tag{3.5}$$

Where I_{batt} is the current from the battery.

There are some constraints, which should be taken into account.

When the supercapacitor's voltage is low, the SC should supply higher current to produce the same power. The higher the supplied current, the greater the conduction loss, necessitating a larger and more costly converter.

As a result, the state of charge of the supercapacitor (SOC_{SC}) is limited to greater than 0.5. This limitation leads to implementing a less expensive and smaller converter since it prevents high current and huge conduction loss. The state of charge of the battery (SOC_{batt}) should change between the lower band and the upper band to hinder battery over-charges when a fully charged battery is exposed to downhills. Also, the battery and the supercapacitor provide the demanded power (P_{dem}) for the motor. So, the constraints will be as follows:

$$\begin{gathered} SOC_{batt\_min} \leq SOC_{batt} \leq SOC_{batt\_max}, \\ SOC_{SC\_min} \leq SOC_{SC} \leq SOC_{SC\_max}, \\ P_{dem} = P_{batt} + P_{SC}. \end{gathered} \tag{3.6}$$

The following values for the upper and lower limits are considered [5]:

$$\begin{gathered} SOC_{batt\_min} = 0.2, \\ SOC_{batt\_max} = 0.9, \\ SOC_{SC\_min} = 0.6, \\ SOC_{SC\_max} = 0.8. \end{gathered} \tag{3.7}$$

Moreover, the optimization variable "r" is defined as given below:

$$P_{batt} = rP_{dem},$$
$$P_{SC} = (1 - r)P_{dem}, \tag{3.8}$$

While accelerating, the battery and the supercapacitor provide the demanded power by the motor. In this case, the power demand is positive. On the contrary, while braking, the battery and/or the supercapacitor receive electrical energy from the motor. Hence, the power demand is negative.

$$-1 \leq r \leq +1. \tag{3.9}$$

To sum up, the optimization problem is to minimize the squared current from the battery over the entire vehicle trip (where t_f is the length of the trip) considering the previously defined constraints.

$$\begin{gathered} Z = min \sum_{t=t_0}^{t_f} I_{batt}^2 (t). \\ SOC_{batt\_min} \leq SOC_{batt}(t) \leq SOC_{batt\_max}, \\ SOC_{SC\_min} \leq SOC_{SC}(t) \leq SOC_{SC\_max}, \\ -1 \leq r(t) \leq +1. \end{gathered} \tag{3.10}$$

In order to solve this decision making problem, the relationship between the state variables should be investigated. The electrical circuit models of the battery and the supercapacitor are shown in Fig. 3.1.

Using the considered model for the supercapacitor, the following equations are written [5]:

$$P_{SC} = I_{SC}V_{SC} - R_{SC}I_{SC}^2,$$
$$V' = -\frac{I_{SC}}{C},$$
$$SOC_{SC} = \frac{V_{SC}}{V_{SC\,max}}, \tag{3.11}$$
$$SOC'_{SC} = -\frac{\left[SOC_{SC}V_{SC\,max} - \sqrt{(SOC_{SC}V_{SC\,\mathrm{max}})^2 - 4P_{SC}R_{SC}}\right]}{2R_{SC}CV_{SC\,max}}.$$

I_{SC} and V_{SC} denote the current and voltage of the supercapacitor. Capacitance (C), internal resistance (R_{SC}) of the supercapacitor, and the maximum voltage $(V_{SC\,max})$ that the supercapacitor can hold are assumed to be constant.

Similarly, we can write the following equations for the battery:

$$P_{batt} = I_{batt}V_{batt} - R_{batt}I_{batt}^2,$$
$$I_{batt} = \frac{[V_{OC} - \sqrt{V_{OC}^2 - 4R_{batt}P_{batt}}]}{2R_{batt}},$$
$$SOC'_{batt} = -\frac{I_{batt}}{C_{batt}}, \tag{3.12}$$
$$SOC'_{batt} = -\frac{[V_{OC} - \sqrt{V_{OC}{}^2 - 4P_{batt}R_{batt}}\ ]}{2R_{batt}C_{batt}}.$$

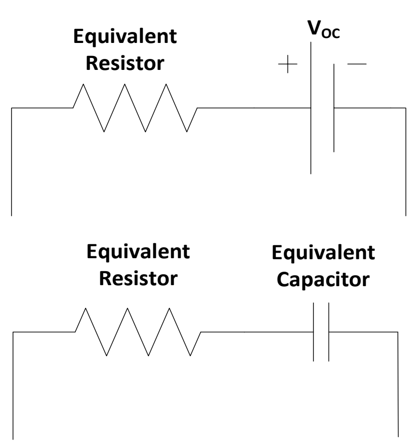

Fig. 3.1 Internal resistances of supercapacitor and battery [5]

V_{batt} is the voltage of the battery. Capacitance (C_{batt}), internal resistance (R_{batt}), and the open circuit voltage (V_{OC}) of the battery are assumed to be constant.

As a result, the next-step state formulations for the state of charge of the battery and the supercapacitor will be:

$$SOC_{batt}(t+1) = SOC_{batt}(t) - \frac{\left[V_{OC} - \sqrt{{V_{OC}}^2 - 4P_{batt}R_{batt}}\right]\Delta\mathrm{t}}{2R_{batt}C_{batt}},$$
$$\beta = \sqrt{(SOC_{SC}(t)V_{SC\,max})^2 - 4P_{dem}R_{SC}\big(1 - r(k)\big)}, \tag{3.14}$$
$$SOC_{SC}(t+1) = SOC_{SC}(t) + \frac{-[SOC_{SC}(t)V_{SC\,max} - \beta\,]\Delta\mathrm{t}}{2R_{SC}C_{SC}V_{C\,max}}.$$

Chapter 4

Newton/GMRES-based Nonlinear Model Predictive Control (NG-NMPC)

As mentioned, success stories of implementing MPC in control problems [45], [46], [47] have motivated the author to implement the nonlinear MPC to the problem at hand. In this chapter, the control problem is transformed to the framework of NMPC; then, Newton/GMRES approach is applied at the heart of the NMPC for optimization process. Simulation results show superiority of this method.

4.1 Structure of NMPC

NMPC is a nonlinear version of MPC, which is more accurate than linear MPC since it does not have linearization errors. A control-oriented model implemented at the NMPC utilizes future data and current states to predict next states. By estimating the states over the prediction interval with the length Np, NMPC optimizes an open-loop finite interval control problem at each moment. Doing so, NMPC determines the optimum control output at each time span of the control interval of size Nc. Only the first optimum control output is applied, the next ones are ignored. Applying the first optimal control action updates the initial condition of the next control problem. The process of applying the first control input and updating calculations is repeated until the last time step of the drive cycle.

Manifestly, the applied control-oriented model at the heart of an MPC-based controller deeply influences on its efficiency. The more detailed the control-oriented model depicts the controlled system, the more effectively the MPC predicts, thus, the better performance it yields.

Nonetheless, implementing accurate control-oriented models imposes high computational cost and slows the convergence process, which in turn hinders applicability of the MPC in practice [43]. Figure 4.1 depicts the structure of NMPC.

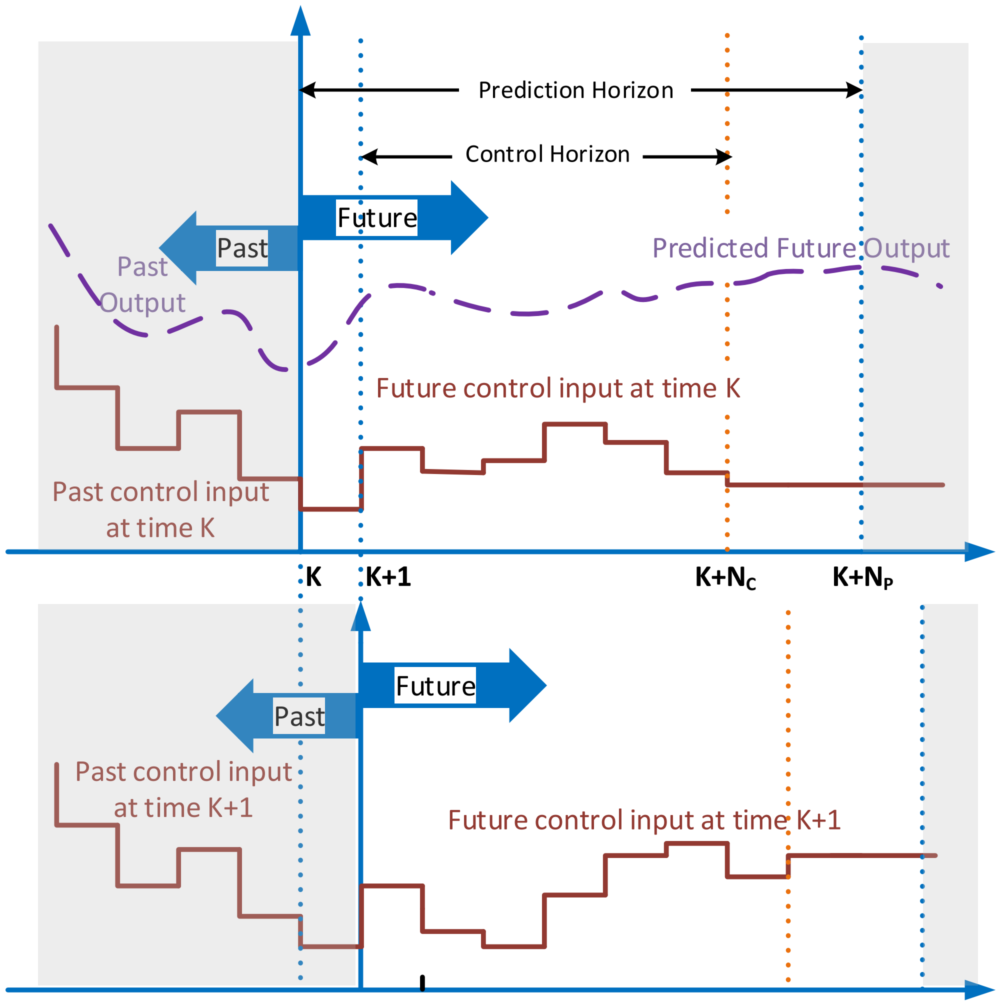

Fig. 4.1 Basis of the NMPC [67]

4.2 Newton/GMRES NMPC

Not long ago, the emergence of a new and fast optimization-based method, Generalized Minimum Residual (GMRES), for control applications inspired many researchers to develop GMRES-based optimal controllers [47]. Ohtsuka proposed combining the Continuation and GMRES methods; so-called C/GMRES method. C/GMRES is a swift numerical optimal controller for control input series. Since it finds an answer for the differential equation only once each step, it notably reduces computational cost. Therefore, a C/GMRES-based NMPC is a valid candidate for a real-time implementable EMS. A comprehensive explanation of this solver can be found in [48]. Furthermore, Kelly developed a Newton/GMRES method that utilizes a forward-difference GMRES algorithm that employs Newton steps to solve equations. It is proved that using finite difference does not weaken the Newton/GMRES performance, if the steps of the forward difference approximation of derivatives are small enough. The Newton/GMRES finds an acceptable answer using the Newton-iterative method through only a limited number of iterations [68].

The above-mentioned challenges have motivated the authors of this paper to design a novel EMS, the NG-NMPC, for the problem in hand, based on the longitudinal dynamic characteristics of Rav4EV. The main purpose of the proposed EMS is to maximize battery life.

4.3 Newton/GMRES-Based Nonlinear Model Predictive Controller Approach (NG-NMPC)

As mentioned before, the main goal is to extend battery longevity. The EMS distributes the demanded power between the battery and the supercapacitor in a way that protects the battery during sudden acceleration or braking. In other words, the EMS handles power fluctuations by the supercapacitor as much as possible; meanwhile the battery is the major power source. In this study, the authors propose a novel EMS using Nonlinear MPC (NMPC). MPC is proved to be an efficient controller for the BEV hybridized with the SC. NMPC is a kind of MPC for handling nonlinear systems. NMPC is preferable to MPC since it does not have linearization errors although it increases the computational cost [18].

A control-based model implemented at the heart of the NMPC utilizes future data and current states to predict next states. By estimating P_{dem} over the prediction interval with the length N_p, NMPC optimizes an open-loop finite interval control problem at each moment. As a result, NMPC determines the optimum power distribution between the battery and the supercapacitor at each time span of the control interval of size N_c. Only the first optimum power distribution is applied, the next ones are ignored. Applying the first optimal power split updates the initial condition of the next control problem. The process of applying the first control output and updating situations is repeated until the last time step of the drive cycle.

NG-NMPC is a novel version of NMPC that solves the problem using a Newton/GMRES-based approach. Briefly, this controller finds the root of a nonlinear problem in real-time. Using a single-shooting method is preferred rather than multiple-shooting one since the problem is not big and nonlinear enough to utilize a multiple-shooting technique.

One of the major difficulties of applying NMPC is satisfying inequality constraints. Many researchers have proposed different handling methods, but none of them sounds perfect. In fact, each of those methods has its own disadvantages. Huang et al. compared different constraint-handling strategies in [47].

4.4 Optimal Control Theory

In this section, the mathematics form of the NMPC is presented. The Optimal Control theory proves that any optimization problem in the form of equation (7) can be converted to a root finding problem of a set of nonlinear equations, if necessary optimality conditions are satisfied. Suppose the following performance index that is a function of a state x and an input u

$$J = \varphi\left(x\left(t, N_p\right)\right) + \int_0^{N_p} C\left(x(t,\omega), u(t,\omega)\right) d\omega. \tag{4.1}$$

Constraints:

$$\begin{gathered} \dot{x}(t) = f\left(x(t), u(t)\right), \\ C_{equal}\left(x(t), u(t)\right) = 0. \end{gathered} \tag{4.2}$$

$N_{pr}, t, \varphi, C, f(.)$, and C_{equal} denote the prediction horizon, time, the terminal cost at the end of the prediction horizon, the trajectory cost, system dynamics, equality constraints respectively. Suppose that prediction time, ω, consists of n timesteps with the length of $\Delta\omega$. So, the performance index, J, can be written as follows:

$$J = \varphi\big(x_n(t)\big) + \sum_{i=0}^{n-1} C\big(x_i(t), u_i(t)\big)\Delta\omega.$$
$$x_{i+1}(t) = x_i(t) + f\big(x_i(t), u_i(t)\big)\Delta\omega. \tag{4.3}$$
$$C_{equal}\big(x_i(t), u_i(t)\big) = 0.$$

Ohtsuka defined the $Hamiltonian, H,$ as follows [48]:

$$H(x, u, \varphi, v) = C(x, u) + \varphi^T f(x, u) + v^T C_{equal}(x, u). \tag{4.4}$$

φ and v refer to Costates and Lagrange multipliers. If the following conditions are satisfied, the above problem can be transformed to Two Point Boundary Value Problem.

$$\begin{gathered} x_{i+1} = x_i + f(x_i, u_i)\Delta\tau, \qquad (state\ eq.) \\ \varphi_i = \varphi_{i+1}(t) + H_x^T(x_i, u_i, \varphi_{i+1}, v_i)\Delta\tau, \qquad (costate\ eq.) \\ H_u^T(x_i, u_i, \varphi_{i+1}, v_i) = 0, \\ C_{equal}(x_i, u_i) = 0. \end{gathered} \tag{4.5}$$

Consequently, variable states within the prediction horizon can be computed recursively. State variables update by time-forward equations. The first updating state equation is $x_0(t) = x(t)$. In contrast, costate variables update by time-backward equations. The first updating costate equation is $\varphi_n(t) = \theta_x^T\big(x_n(t)\big)$. As a result, the $(x_i)_{i=0}^{n}$ and $(\lambda_i)_{i=0}^{n}$ series are composed in terms of the $(u_i)_{i=0}^{n-1}$ and $(v_i)_{i=0}^{n-1}$ series. Vector $U(t)$ can be defined as a combination of $(u_i)_{i=0}^{n-1}$ and $(v_i)_{i=0}^{n-1}$ series as below:

$$U(t) = [u_0^T(t), v_0^T(t), u_1^T(t), v_1^T(t), \dots , u_{n-1}^T(t), v_{n-1}^T(t)]^T. \tag{4.6}$$

$U(t)$ can be calculated by solving the equation (4.6), which describe the necessary constraints of optimality. Briefly, $U(t)$ is calculated as follows:

$$G\big(U(t), x(t)\big) = \begin{bmatrix} H_u^T(x_0, u_0, \varphi_1, v_0) \\ C_{equal}(x_0, u_0) \\ \vdots \\ H_u^T(x_i, u_i, \varphi_{i+1}, v_i) \\ C_{equal}(x_i, u_i) \\ \vdots \\ H_u^T(x_{n-1}, u_{n-1}, \varphi_n, v_{n-1}) \\ C_{equal}(x_{\mathrm{n}-1}, u_{n-1}) \end{bmatrix} = 0. \tag{4.7}$$

4.5 Newton/GMRES Method

In the previous section, it was shown that the control optimization problem leads to solving the equation (4.7). In the first step, Newton's approach can be applied to this problem.

$$\begin{gathered} G_U\left(U^i(t), x^i(t)\right)\Delta U(t) = -G\left(U^i(t), x^i(t)\right). \\ U^{i+1} = U^i(t) + \Delta U(t). \end{gathered} \tag{4.8}$$

Obviously, the larger the states number and prediction horizon, the more challenges to compute the Jacobian, $G_U(U, x)$, will be. Also, Jacobian estimation using approximation techniques imposes a high computational cost.

Some researchers have developed alternatives based on inner iterations of Newton's technique [68]. Theses methods, so-called Newton-iterative, find the solution of equation (4.8) through a few inner iterations on ΔU without calculating $G_U(U, x)$ accurately. A Newton-GMRES approach is one of the Newton-iterative methods that applies forward difference Generalized Minimal Residual (FDGMRES) technique to calculate Newton steps. This method approximates the solution of equation (4.8) as follows:

$$G_u(U, x)z \approx D_h G(U, x : z, 0) = \frac{G(U + hz, x) - G(U, x)}{h}. \tag{4.9}$$

In the above equation, h represents a small value that is greater than zero.

This method is promising since the linear equation converges after only a limited number of iterations. Reference [53] provides more information about FDGMRES method. [68]

Implementing FDGMRES method for finding roots in continuous time develops another method, the Continuation/GMRES (C/GMRES) approach [48]. This method solves $G(U,x)=0$ by stabilizing $G(U,x)$ to zero through the computation of the derivative of $U(t)$ with respect to time using the following equation:

$$G'\big(U(t),x(t)\big) = -E_p F\big(U(t),x(t)\big). \tag{4.10}$$

Where E_p refers to a matrix of positive eigenvalues. Differentiation of the above equation leads to the following [69]**Error! Reference source not found.**

$$F_U\big(U(t),x(t)\big)U' = -E_p F\big(U(t),x(t)\big) - G_x\big(U(t),x(t)\big)x'. \tag{4.11}$$

Equation (4.11) is a linear algebraic equation, which finds U' by FDGMRES method [48]. $U(t)$ is found by computing the integral of $U'(t)$ in real time.

4.6 Simulation Results

To investigate the effectiveness of the proposed NG-NMPC, it was implemented for one-and-half consecutive Urban Dynamometer Driving Schedule (1.5xUDDS) drive cycle. As discussed in details in Chapter 3, the performance index is to minimize the battery consumption at the end while satisfying the problem constraints. To extend the battery lifespan, an efficacious EMS should protect the battery against the sudden fluctuations of the power demand.

Fig. 4.2 shows the smooth reduction of SOC_{batt} in which the NG-NMPC effectively protects the battery against power-demand fluctuations. As mentioned, the power-surges into and out of the battery raise its temperature, which ultimately shorten its life and efficiency [65]. Consequently, the smooth trend of SOC_{batt} leads to extend the battery lifespan. In contrast, SOC_{SC} has a lot of sudden rises and drops; that is, the NG-NMPC makes the SC responsible for handling sudden acceleration/deceleration as much as possible.

Fig. 4.3 and 4.4 demonstrate the power distribution between the battery and the supercapacitor for the test drive cycle. As expected, the plot of P_{batt} is smoother than that of P_{SC}. Fig. 4.5 shows the result of implementing NG-NMPC on the cost function.

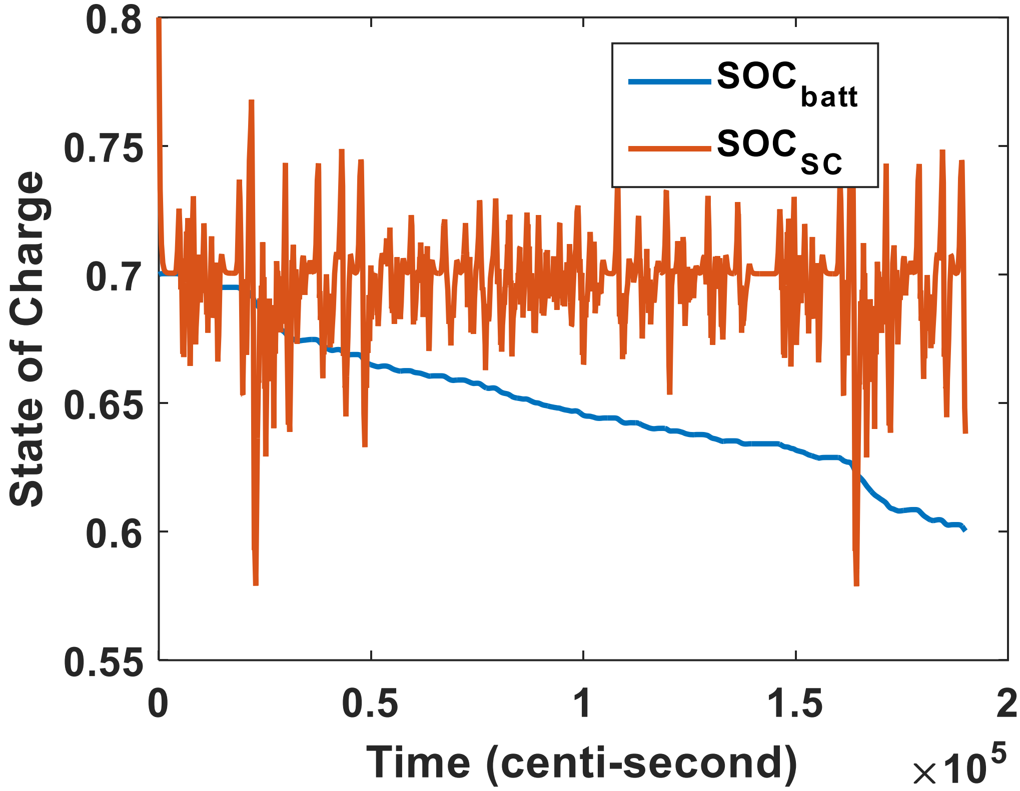

Fig. 4.2 SOCs vs. time for the UDDS drive cycle

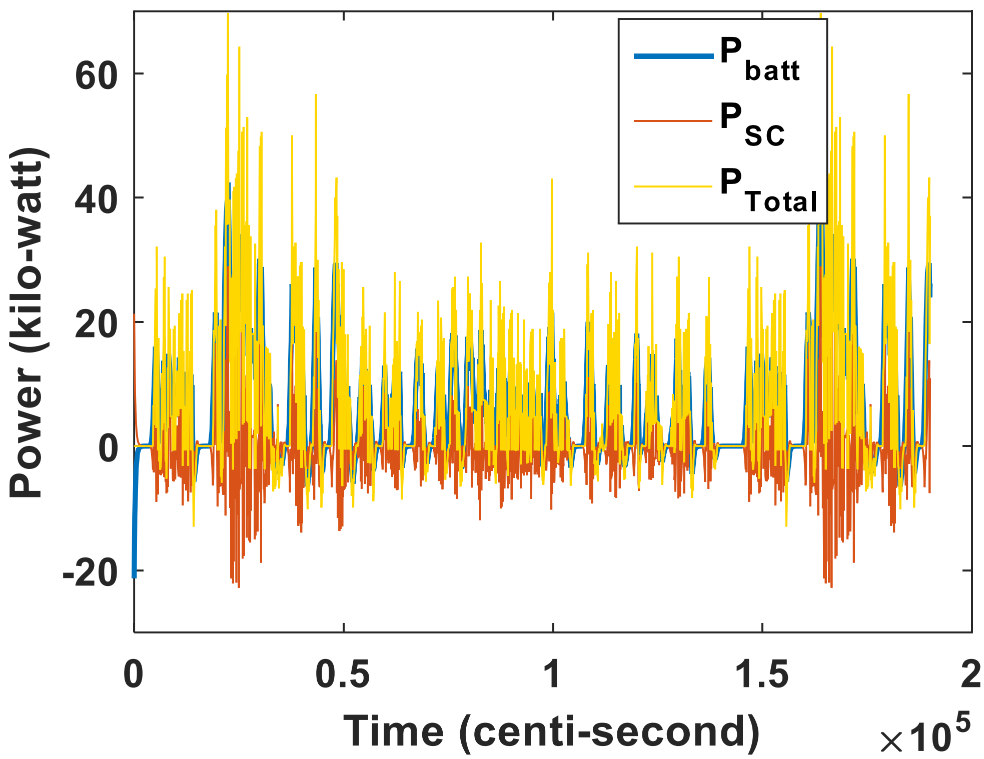

Fig. 4.3 Powers vs. time for the UDDS drive cycle

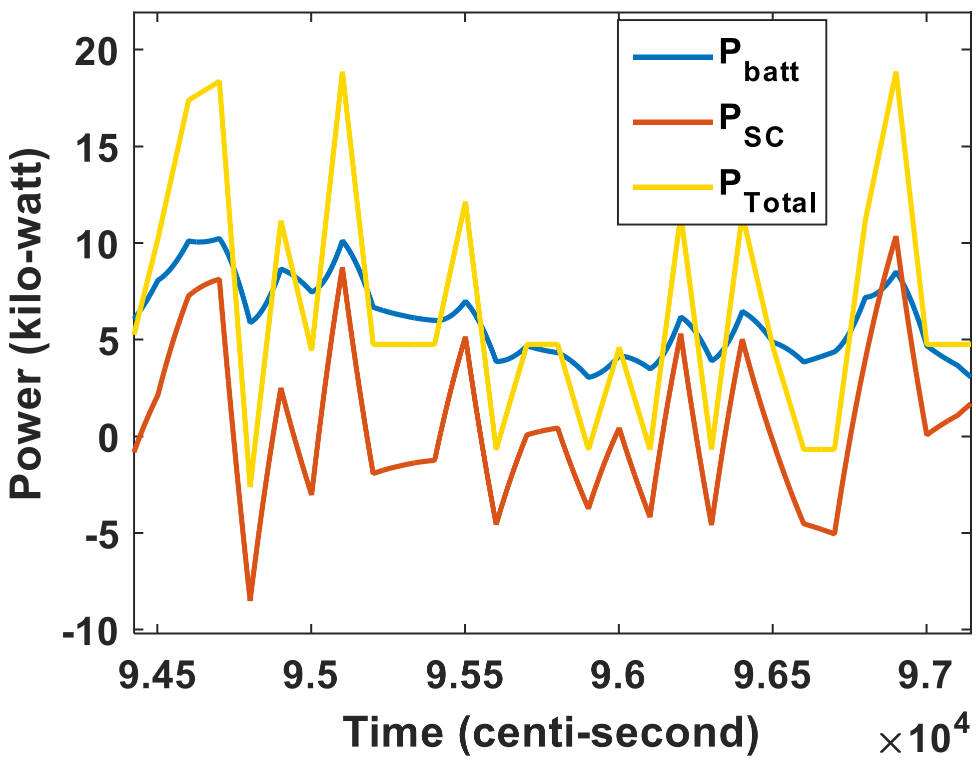

Fig. 4.4 Close-up view of Fig. 4.3

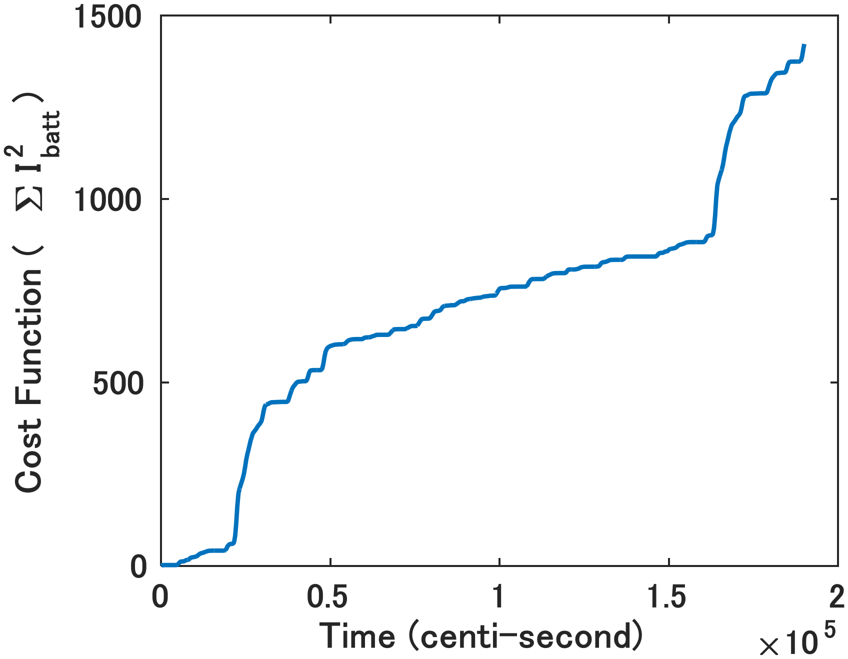

Fig. 4.5 Cost Function vs. time (0.01s) for the UDDS drive cycle

Chapter 5
Dynamic Programming

In this chapter, Dynamic Programming (DP) is applied to the problem at hand. Although because of its huge computational cost, it cannot be used for super-fast online applications, DP is worthwhile to implement since it yields the global optimum. The DP solution is an effective basis for evaluating the accuracy of other approaches' solutions.

In the following, the problems is converted to the DP form. Then, the sensitivity analysis of the DP's parameters is provided; the simulation results are presented. At the end, Table 5.1 provides the comparison between different cases.

5.1 Structure of DP

DP is an effective and deterministic optimization approach. It computes the cost of all the possible solutions and selects the best one as the final answer. In particular, for this control problem, it considers all the feasible values of $P_{SC}\ and\ P_{batt}$ for a given P_{dem} as a priori and calculates the objective function ($\sum\ I_{batt}^2$) for the prediction horizon for each possible power distribution. Obviously, DP chooses the minimum performance index ($Z^* = \sum\ I_{batt}^2$) as the optimum solution and returns the corresponding $P_{SC}\ and\ P_{batt}$ for the prediction horizon as the optimum power split between the battery and supercapacitor. Since DP can be applied only on discrete models, the control inputs (r, P_{batt}, P_{SC}) and control-oriented model variables (SOC_{batt}, SOC_{SC}) are discretized by $dt = 1s$. At each moment, the state of the system is described by $SOC_{batt}, and\ SOC_{SC}$. So, the state variable (x_t) can be defined as follows:

$$x_t = \begin{bmatrix} SOC_{batt,t} \\ SOC_{SC,t} \end{bmatrix}, \qquad t = 0, 1 \dots, T-1 \tag{5.1}$$

The control input (u_t) is defined as follows:

$$u_t = \begin{bmatrix} P_{batt,t} \\ P_{SC,t} \end{bmatrix}, \qquad t = 0, 1 \dots, T-1 \tag{5.2}$$

As mentioned, the DP solves the problem recursively, that is, it begins at the final point $(t = T)$ assuming Z^*_{TT} = 0 and moves backward to the first point. Within the framework of DP, the performance index is written as follows:

$$Z^*_{T-t,T}(x_{T-t}) = min\{f(\ x_{T-t},\ u_{T-t}) + Z^*_{T-(t-1),T}(g(x_{T-t}, x_{T-t})\},$$
$$x_{t+1} = g(x_t, u_t), \qquad t = 1, \dots, T \tag{5.3}$$

As shown, the minimum performance index of the t-stage policy of a T-stage process $(Z^*_{T-t,T}(x_{T-t}))$ is calculated based on the minimum performance index of the previous (t-1)-stage policy. This recursive minimization will be repeated until the whole prediction horizon is covered. More information is provided in [67], [70]. Fig. 5.1 shows the simulation results of applying DP in a framework of MPC to the control problem. DP-MPC approach is implemented via different prediction horizons and grid sizes to investigate the effect of each parameter.

5.2 Sensitivity Analysis of DP's Parameters

As expected, increasing the prediction horizon enhances the solution's accuracy. DP uses the perfect knowledge of the future as a priori; hence, increasing the time horizon provides more data about the next seconds and improves the optimality. In particular, for the problem at hand, the longer the prediction horizon is, the less battery will be used, and the greater the battery state of charge will be. Fig. 5.2, 5.3, and 5.4 illustrates the effect of the prediction horizon parameter on the MPC performance.

Another parameter that effects DP performance is the grid size. In fact, DP yields the global optimum if and only if the grid size is chosen properly. Obviously, a greater grid size leads to a more accurate and optimum final answer; however, increasing the grid number will exponentially increase the calculation cost.

Fig. 5.5 and 5.6 show the simulation results of implementing DP with different grid sizes.

Similarly, increasing the grid size will lower the battery usage, which in turn decreases the cost function and ultimately yields a greater state of charge of the battery.

DP uses the future information as a priori; also, as shown in Table 5.1, it has a huge computational time that exponentially increases by increasing the grid size. Thus, it is far from online application in practice. It only provides a good baseline for comparing the performance of the other implemented approaches.

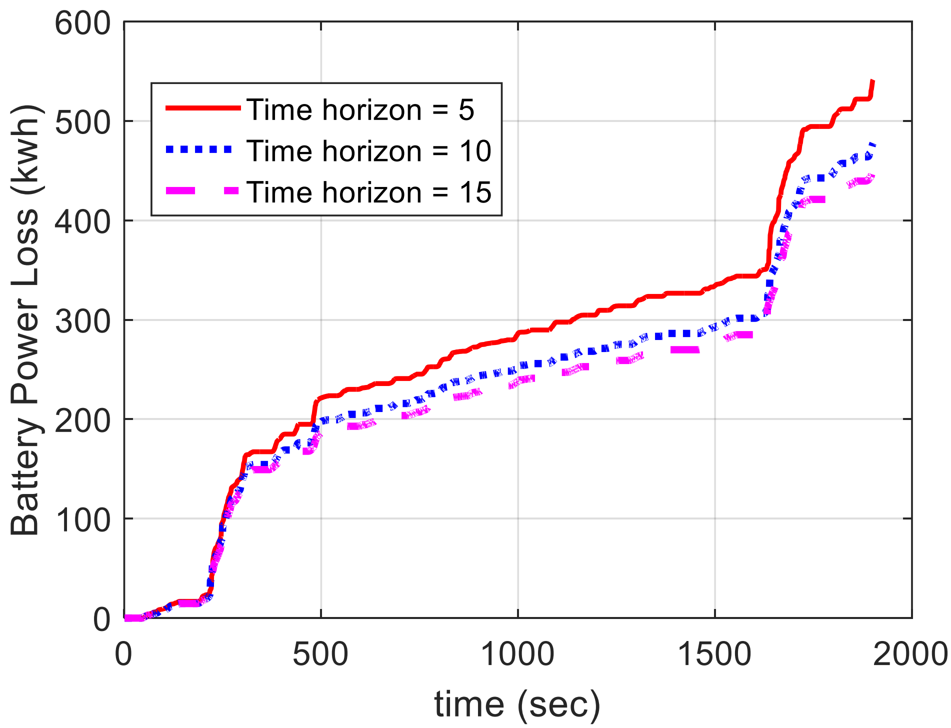

Fig. 5.1 Comparison of the battery power loss for different prediction horizons in DP-MPC

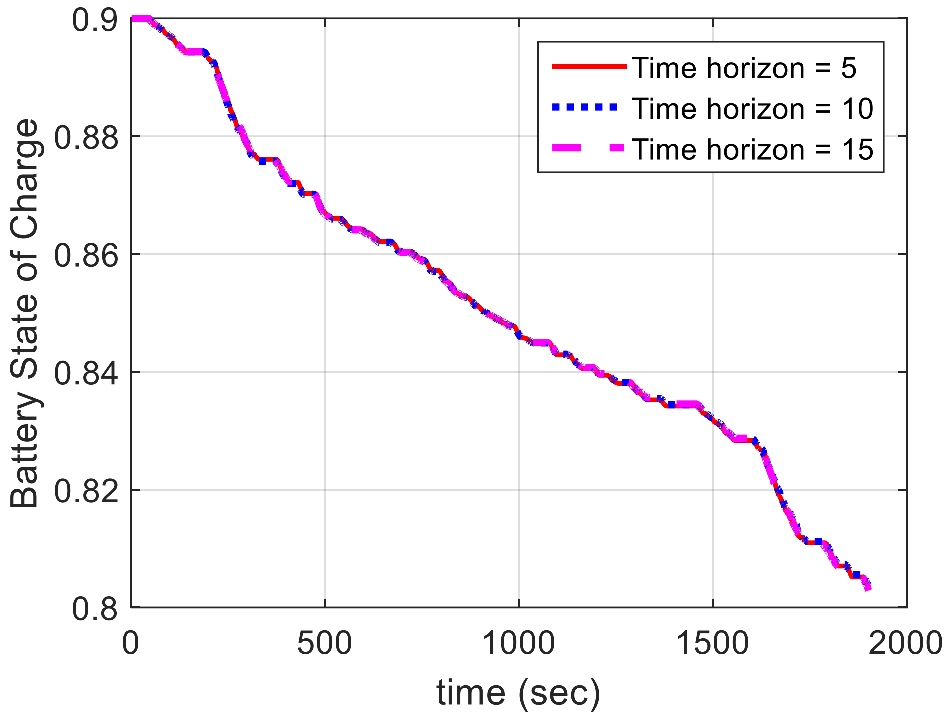

Fig. 5.2 Comparison of State of Charge of the Battery for different prediction horizons in DP-MPC

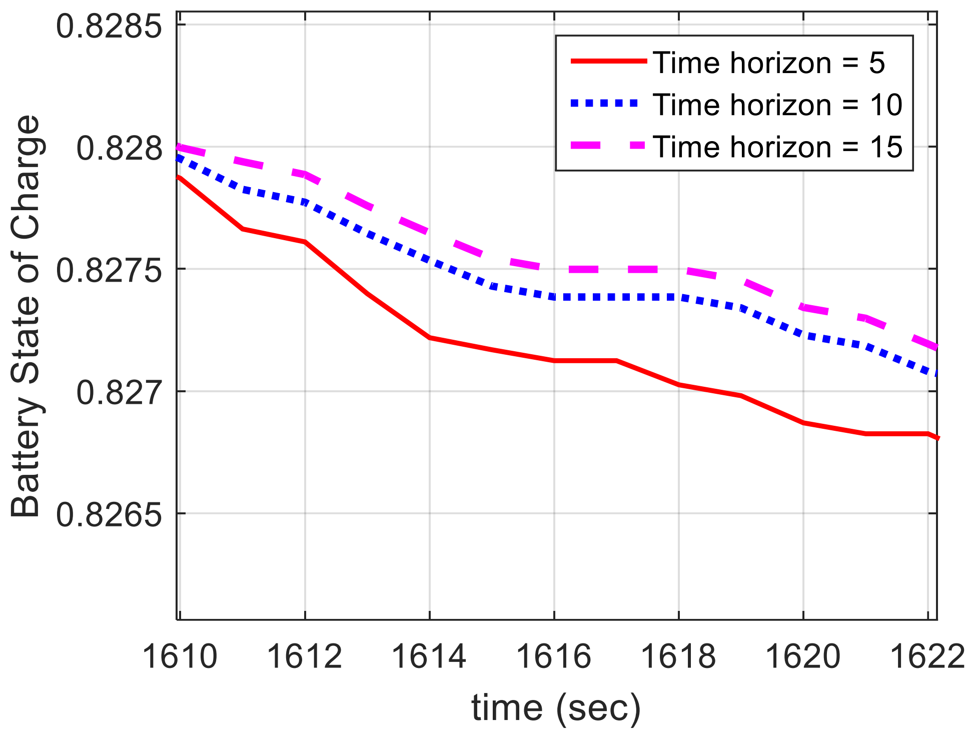

Fig. 5.3 Close-up view of Fig. 5.2

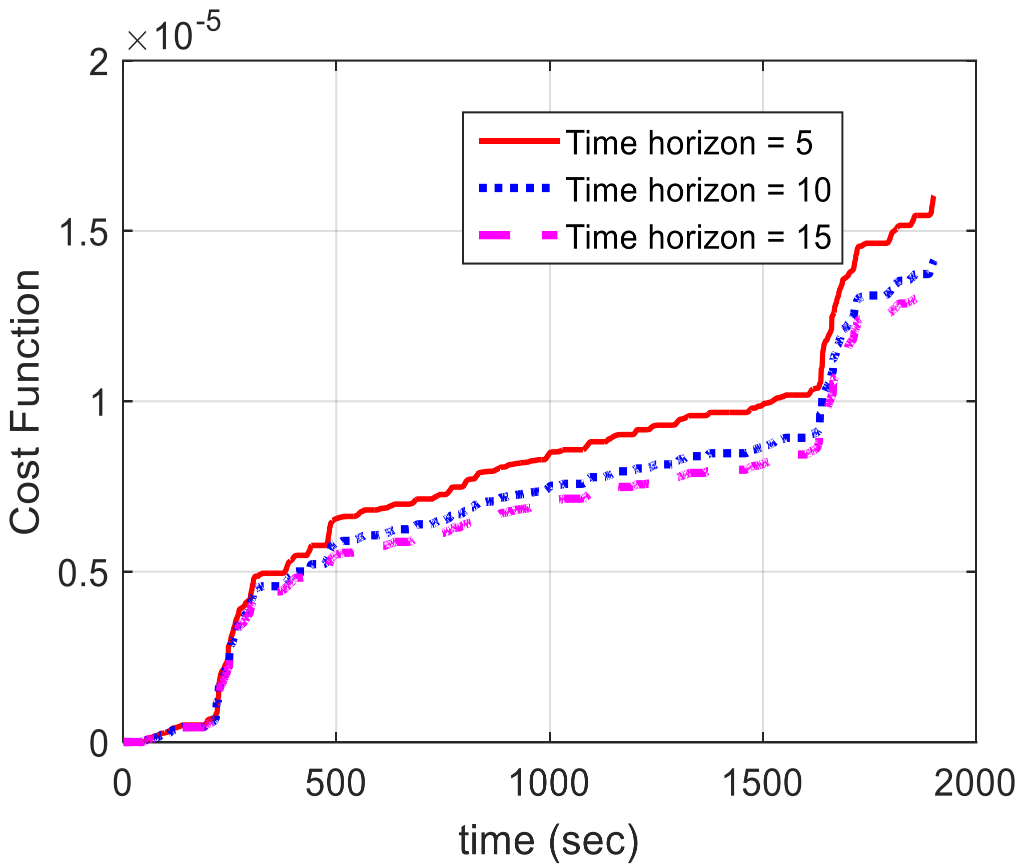

Fig. 5.4 Comparison of cost function for different prediction horizon in DP-MPC

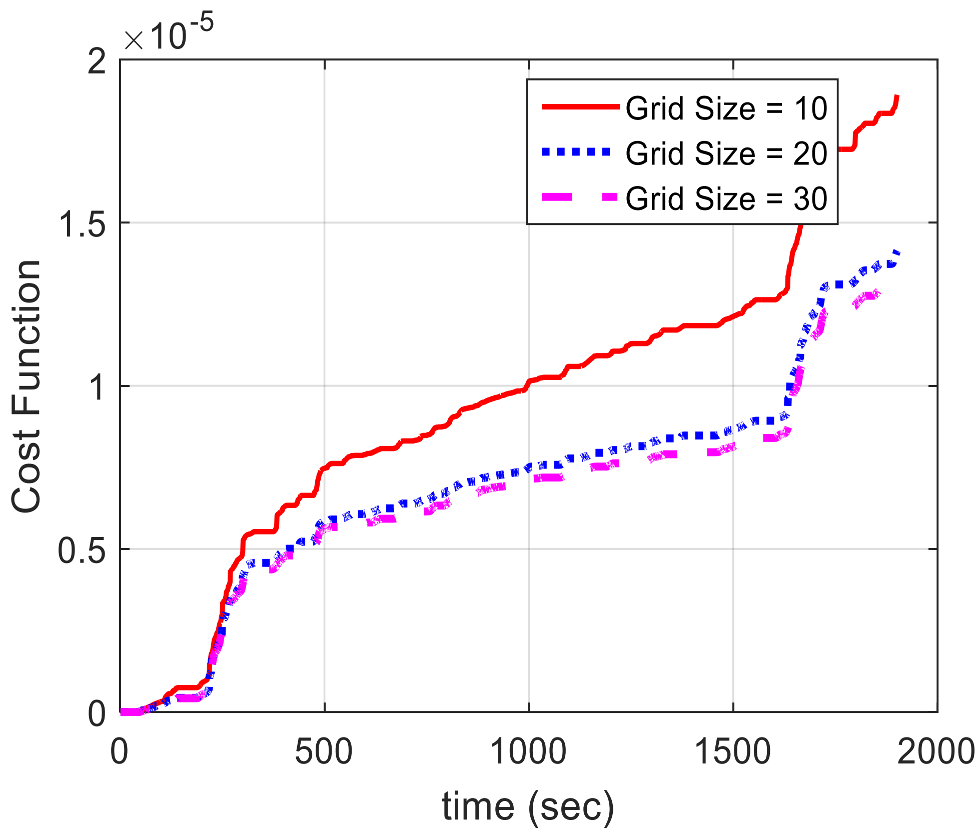

Fig. 5.5 Comparison of the cost function for different grid sizes in DP-MPC

Table 5-1 Comparing computational cost of different methods in DP-MPC

Method	Time (seconds)	Final Battery Power Loss (kWh)
DP-MPC Grid Size = 10	8	640
DP-MPC Grid Size = 20	52	477
DP-MPC Grid Size = 30	165	452

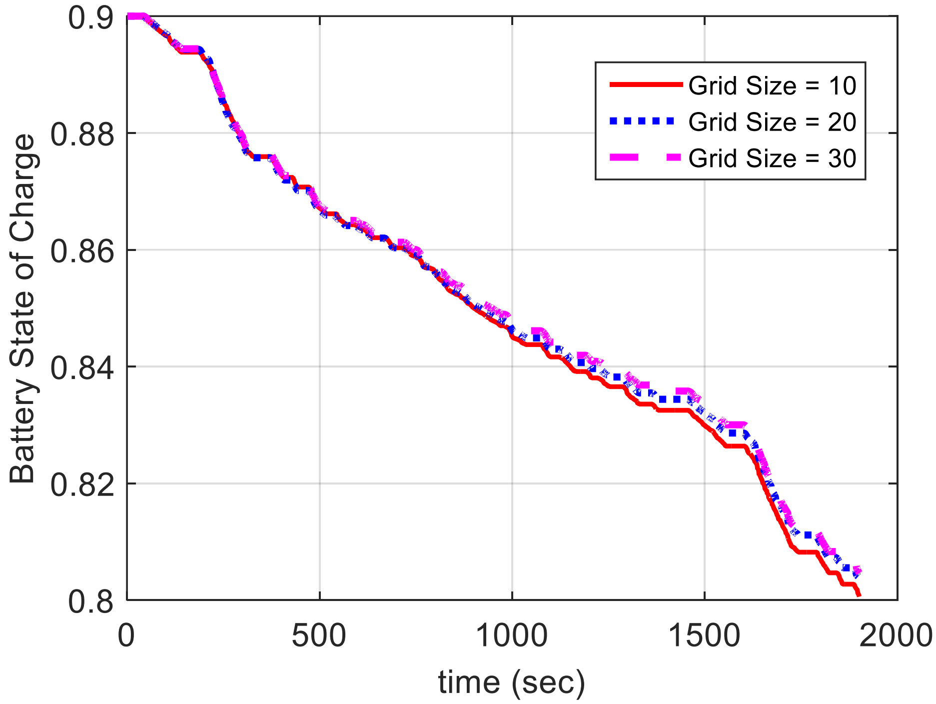

Fig. 5.6 Comparison of the battery state of charge for different grid sizes in DP-MPC

Chapter 6
Stochastic Dynamic Programming (SDP)

As mentioned previously, the uncertain nature of the demanded power has motivated the author to solve this EMS design problem by SDP. In fact, the main advantage of this work is that unlike many other studies that consider the demanded power to be pre-defined values, the power demand is assumed to be unknown beforehand and stochastically changing, which is generated using a Markov chain [21]. In other words, the Transition Probability Matrix (TPM) is created assuming the power demand as a Markovian state. This means that the next power demand (P_{dem}) is only dependent on the current power demand and not on the previous ones. Another assumption is that power demand changes within a finite range. The TPM determines the next possible power demands with their corresponding probabilities. In this study, the power demand has 20 discrete states. Obviously, the larger number of discretization, the more accurate the TPM will be. However, a large discretization number leads to high computational costs. By trial-and-errors, the number is set to be 20 [18]. This number turns out to be accurate enough and does not impose too much complexity in our model. As a result, the TPM is a 20 by 20 matrix that maps the current P_{dem} to the next P_{dem} [18]. Therefore, the Markov chain is formulated as follows:

$$p_{nm} = Pr\{P_{dem}(t+1) = p_n | P_{dem}(t) = p_m\}, \\ n, m \epsilon \{1,2, \dots ,20\}. \tag{6.1}$$

Obviously, the sum probability of all the possible next states equals to one:

$$\sum_{m} p_{nm} = 1. \tag{6.2}$$

$P_{dem}(t)$ is a Markovian state at time t and p_{nm} represents the probability of future power demands. In other words, it indicates the probability of transition from the current state ($P_{dem}(t)$) to the future state ($P_{dem}(t+1)$).

These probabilities are calculated using a number of driving cycles from the ChargeCar website [71]. The drive cycles represent vehicle speeds over time, but TPM is based on the power demands. Thus, the vehicle speeds are converted to the corresponding power demands according to the characteristics of Toyota Rav4EV as mentioned in Chapter 3.

6.1 Stochastic Energy Management System Design

In this section, a stochastic optimization formulation is developed. Since there is no final cost or final constraint in our model and the model equations are time-invariant, the problem is considered as an infinite horizon problem. The infinite horizon approach creates a collection of time-independent policies that can be applied for an online optimal power distribution efficiently. In general, for a Markovian SDP problem, the expected cost is defined as follows:

$$J_{\pi}(x_0) = \lim_{N\to\infty} E_w\left\{\sum_{k=0}^{N-1} \gamma^k \varphi\big(x_k, \pi(x_k)\big)\right\}. \tag{6.3}$$

$J_{\pi}(x_0)$ refers to the expected cost, $\pi(x_k)$ is the control policy, φ is the one-time step cost, and γ is the discount factor ranging between zero and one. In our problem, the objective function is sum of the squared I_{batt} and the penalty for the power demand prediction as follows:

$$\varphi = I_{batt}^2 + \alpha.M, \tag{6.4}$$

M is the squared value of difference between the predicted power demand and the real power demand. α is a weighting factor. The policy iteration algorithm has two stages: the policy assessment and revising the policy. In every iteration, first, $J_{\pi}(x_0)$ for the current policy is calculated as given below:

$$J_{\pi}^{s+1}(x^i) = \varphi\left(x^i, \pi(x^i)\right) + E_{P_{dem(i+1)}}\{\gamma J_{\pi}^{s}(x')\}, \tag{6.5}$$

x' shows the generated states at the end of a time step. Then, J_{π} is calculated to update the policy by minimizing the following equation:

$$\pi(x^i) = argmin[\varphi(x^i, u) + E_{P_{dem(i+1)}}\{\gamma J_{\pi}(x')\}]. \tag{6.6}$$

This optimum policy comes back to the first step to update the performance index. The updating process repeats until $\pi(x)$converges, that is, it does not improve noticeably anymore [32].

In order to apply the SDP method effectively, the first step is to find the proper state number. A large number of the states makes the problem too complex and computationally expensive which is also called “curse of dimensionality”. On the other hand, by choosing small number of states, the considered model will be inaccurate and miss a lot of details. Consequently, the minimum number of states should be selected to represent major characteristics of the system efficiently. In practice, a proper number of states that depict the system accurately enough is found by trial-and-errors. In this investigation, the authors consider three state variables: SOC_{batt}, SOC_{SC} and P_{dem}.

Moreover, every state variable should be discretized. Choosing the correct number of discretization is also critical since a large number of discretization makes SDP too slow [32]. On the contrary, a small value will yield invaluable results. Here, SOC_{batt}, SOC_{SC} and P_{dem} are discretized as follows:

$$\begin{gathered} P_{dem} \in \{P_{dem}^1, P_{dem}^2, \dots, P_{dem}^m\}, \\ SOC_{batt} \in \{SOC_{batt}^1, SOC_{batt}^2, \dots, SOC_{batt}^{N_b}\}, \\ SOC_{SC} \in \{SOC_{SC}^1, SOC_{SC}^2, \dots, SOC_{SC}^{N_c}\}. \end{gathered} \tag{6.7}$$

So, the total space indexing will be as given below:

$$\begin{gathered} \{x^i,\ i = 1,\ 2, \dots,\ mN_bN_c\}, \\ x^1 = \{P_{dem}^1, SOC_{batt}^1, SOC_{SC}^1\}, \\ SOC_{SC} \in \{SOC_{SC}^1, SOC_{SC}^2, \dots, SOC_{SC}^{N_c}\}. \end{gathered} \tag{6.8}$$

Solving this problem by SDP provides a look-up table that can be used online. In fact, the main advantage of applying SDP is that it does not compute the control variable online. At each moment, the EMS controller simply finds and applies the stored control index that

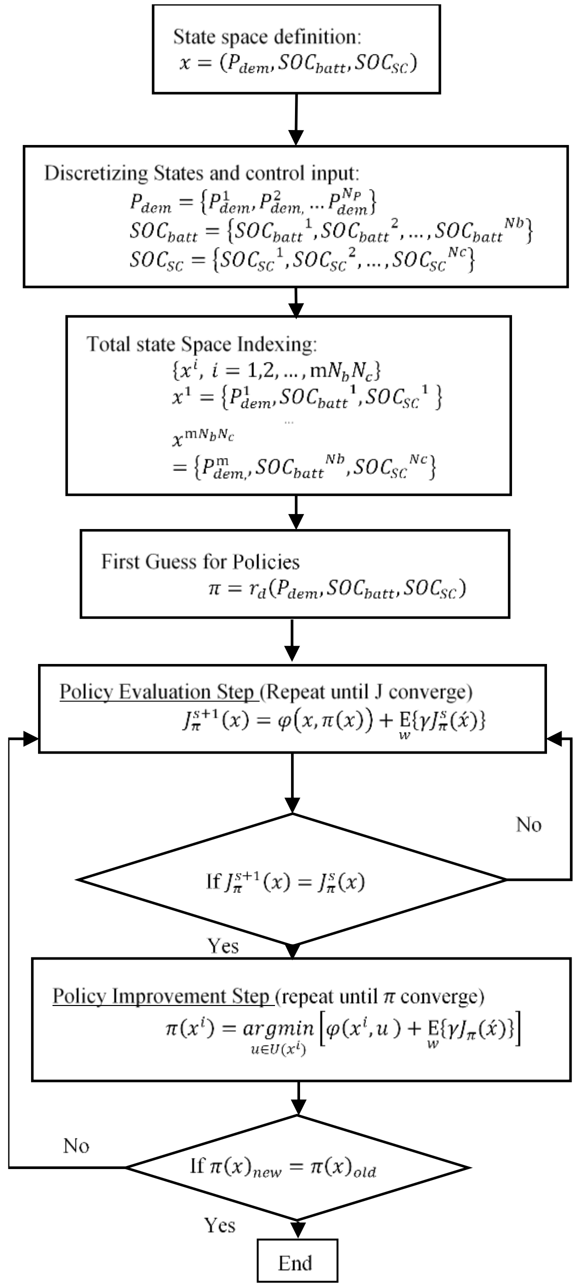

Fig. 6.1. Flowchart of a policy iteration algorithm [72]

corresponds to the current situation. A current situation is described by the state variables to the controller. So, in practice, applying SDP is quite computationally cost-effective. Furthermore, applying SDP does not require any additional equipment to predict the future power demands, for instance advanced ITS and GPS sensors and devices. Moreover, it is able to handle complicated nonlinear objective functions and constraints that cannot be easily solved by any other mathematical and numerical approaches [32].

6.2 SDP-based Simulation Results

The output of the SDP is an optimum policy consisting of several rules, which determine the power distribution between the battery and the supercapacitor every second. In order to evaluate the policy, it is compared to the no supercapacitor (no-SC), the buffer, the Generalized Rule-based Dynamic Programming (GRDP) [73] and Dynamic Programing (DP) methods. The buffer method is the most basic approach to supervise BEVs hybridized with SCs. In this approach, the SC provides the demanded power as much as possible. If the stored power in the SC is not enough, the battery helps the SC to handle P_{dem} [73].

In the GRDP method, the DP algorithm is implemented for a training drive cycle. Then, one simple linear rule is obtained to determine the optimum control variable, see Fig. 6.2.

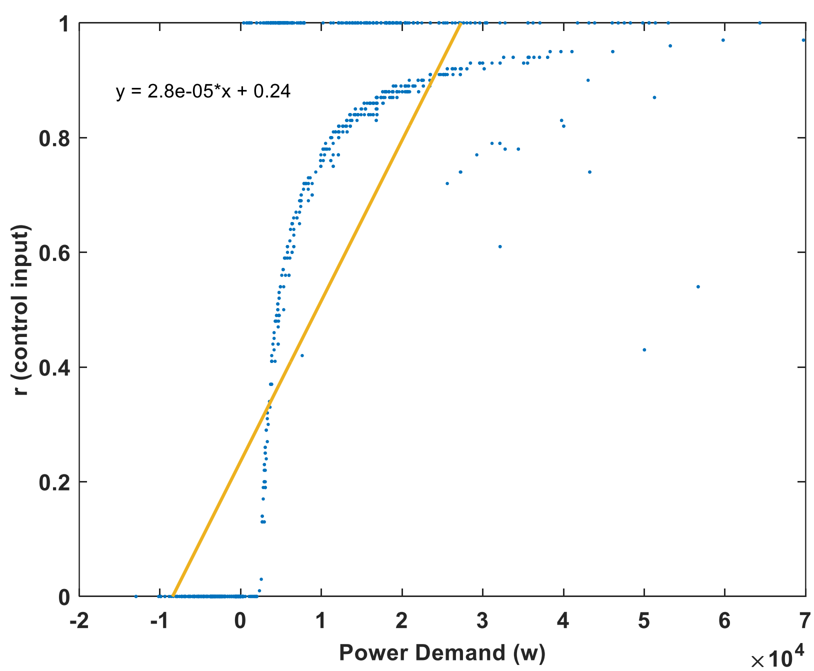

Fig. 6.2. Calculating GRDP from a training set

Fig. 6.3 demonstrates the performance of different applied methods to this problem. As expected, the DP has the best performance since it finds the global optimum by discretizing the state variables. Unlike many other mathematical approaches that can be stuck in local optimums, it selects the global optimum among all of the possible solutions. But the DP strategy assumes the perfect information of the future power demands that would never happen in the reality. Therefore, no method can outperform DP in terms of optimality. However, due to high computational cost and using future information as a priori, it is not a practical approach.

On the contrary, the no-SC method has the worst performance as expected since there is nothing to contribute the battery to provide the demanded power.

The GRDP only outperforms the no-SC method. Although the GRDP strategy is based on the DP that finds the best solution, it shows a weak performance since the GRDP is completely dependent on the drive cycles that are used as training data. Therefore, when the obtained simple linear rule is applied to the test data, it is not as effective as a buffer or an SDP algorithm. The buffer method outperforms the no-SC and GRDP since it always tries to handle the power demand by the SC as much as possible. This approach lowers the power demand load on the battery.

The SDP shows a satisfying performance which means that by applying this algorithm the battery current decreases notably. As a result, the lifespan of the battery will increase. Naturally, it cannot beat the DP since the DP assumes the full-knowledge of the future power demand as a priori. But, it outperforms other methods mainly because of the SDP ability to handle uncertainties. This result is noticeable, since applying SDP does not need implementing new sensors or equipment, so it is quite cost-effective. On the other hand, SDP does not require online calculations; it is an offline method and provides a look-up table that can be implemented quite fast. Comparing to the S-NMPC version of this problem which involves high online computational cost that makes real-time applications a challenge [5], this algorithm is much more practical.

Also, the small distance between SDP and DP performance (5%) demonstrates that SDP can find near-global optimum solutions.

Fig. 6.4 compares the performance of SDP with those of other approaches. Similar to Fig. 6.3, the SDP outperforms the no-SC, buffer, and GRDP.

As shown in Fig. 6.5, by applying the SDP algorithm the supercapacitor handles more fluctuations compared to other considered methods except the DP. This means that the SDP more effectively protects the battery during sudden accelerations and brakes. Since the battery can handle only a limited number of charge/discharge cycles, this policy contributes to extending the lifespan of the battery. Clearly, we cannot plot SOC_{SC} when there is no supercapacitor (no-SC).

Fig. 6.6 shows that by applying the SDP method, the battery consumes less electrical energy than with the buffer and no-SC approaches, thus the battery lifespan will increase. Also, Fig. 6.7 illustrates in more detail that the SDP performance is close to the DP performance, and superior to the buffer and no-SC performance.

Although the difference seems very small, it is promising since these results are based on only 2000 seconds. Consequently, long-term applications of the SDP technique can protect the battery and promote its health effectively.

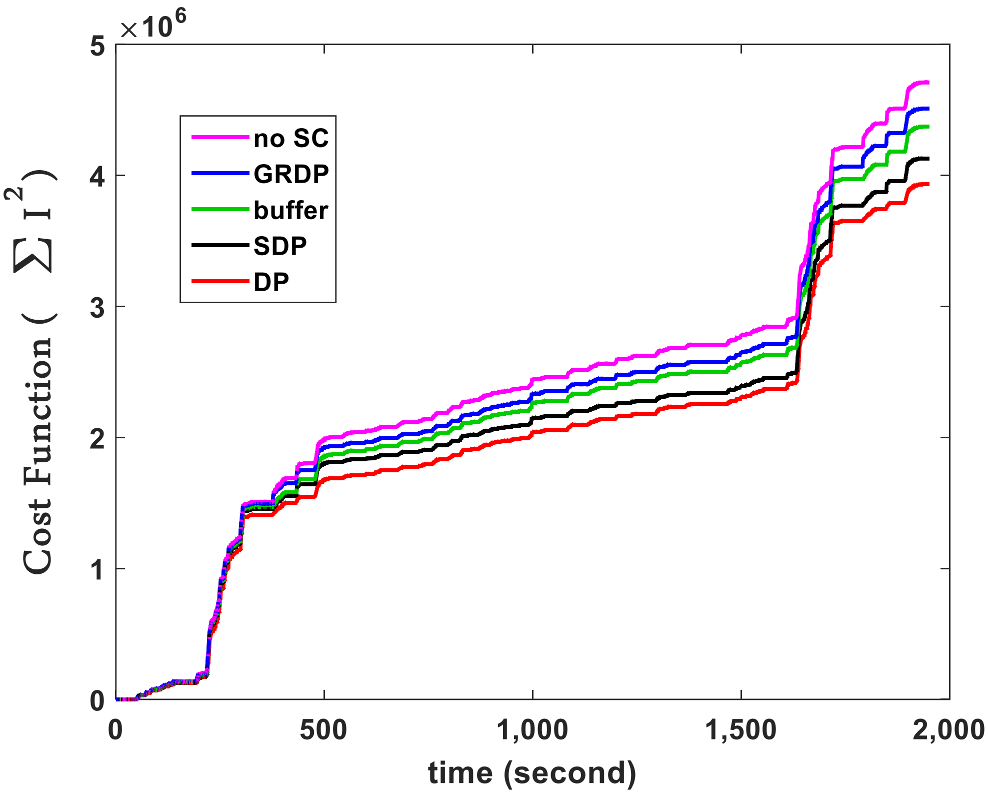

Fig. 6.3. The cost function vs. time (seconds) for the FTP 75 drive cycle

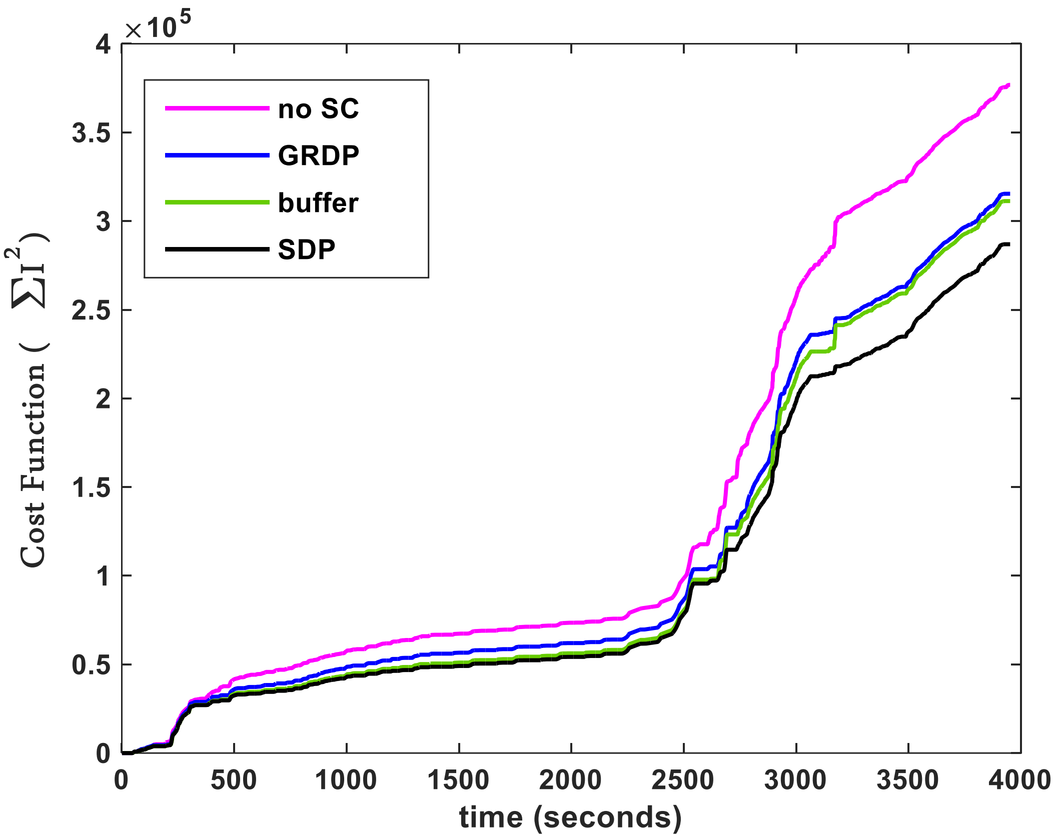

Fig. 6.4. Comparison of the battery current-squared sum vs time (seconds) for the UDDS drive cycle for different solvers

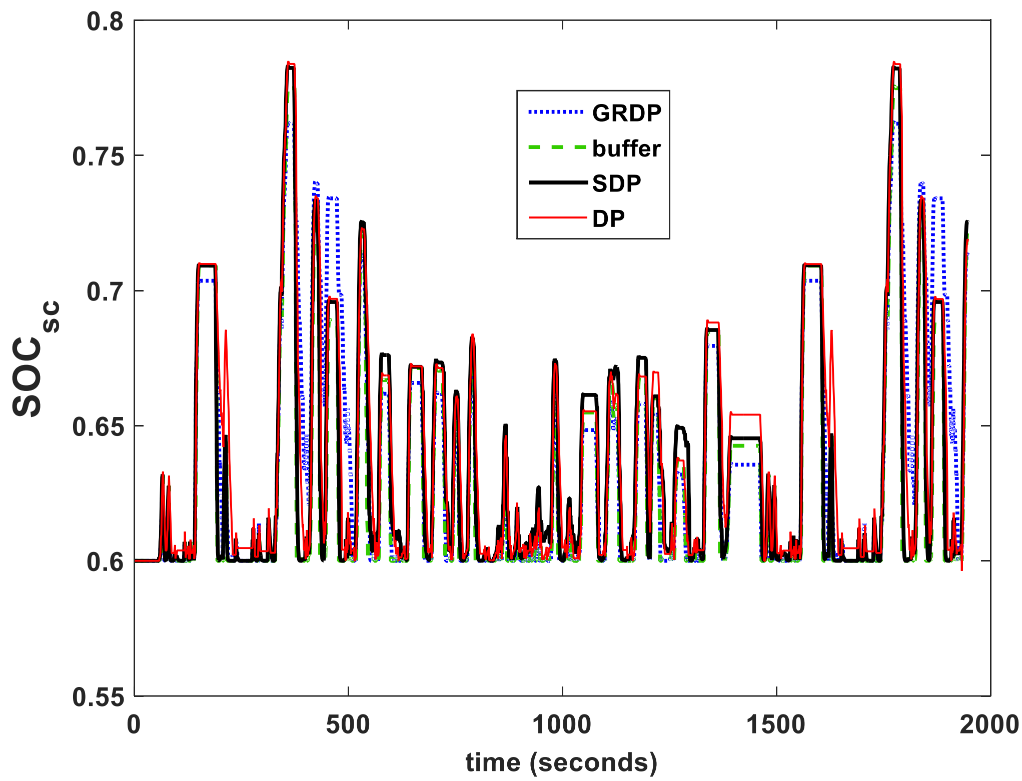

Fig. 6.5 State of charge of the supercapacitor vs time (seconds) for the FTP 75 drive cycle

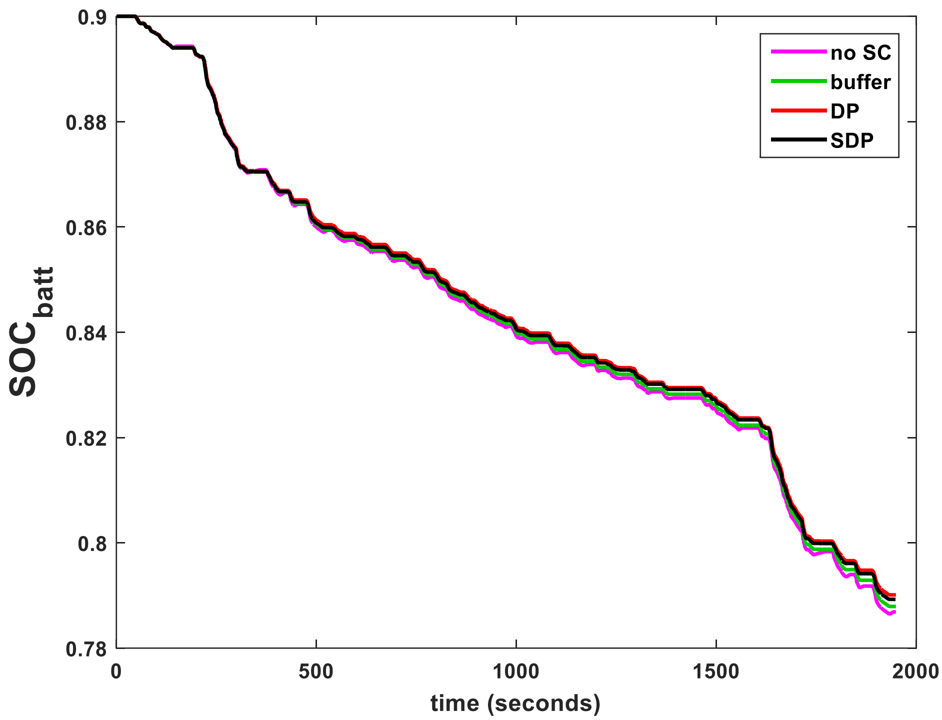

Fig. 6.6 State of charge of the battery vs. time for the FTP drive cycle

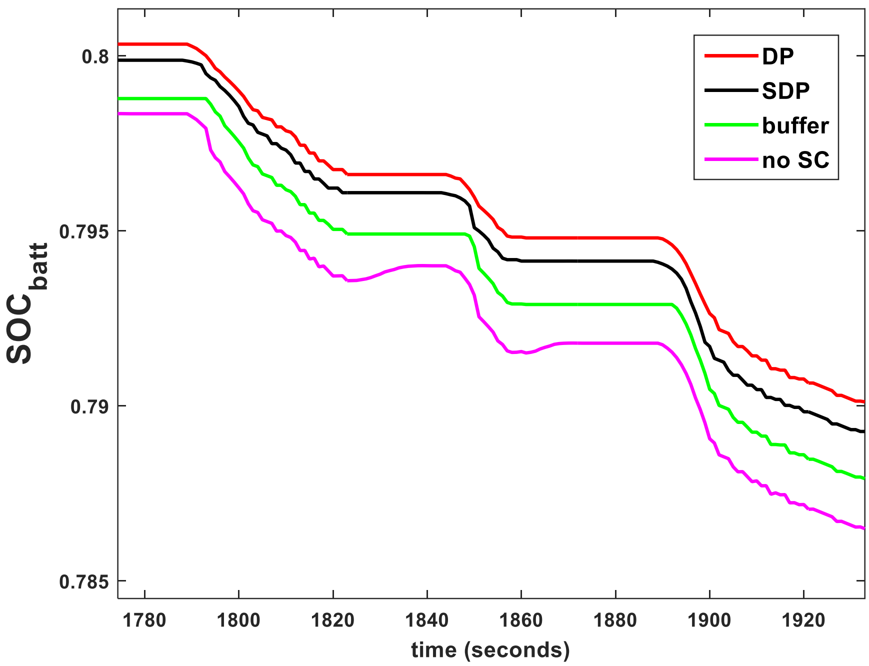

Fig. 6.7 Close-up view of Fig. 6.6

Chapter 7
Particle Swarm Optimization

As mentioned in the introduction, many researchers have applied Particle Swarm Optimization (PSO) to optimize MPC control commands [36], [37]. The success stories about implementing PSO in control problems have motivated the author to solve this problem with this algorithm.

In this chapter, PSO is applied at the heart of MPC to optimize the power distribution between the battery and supercapacitor for the control horizon. Only the calculated distribution (control command) of the first time step is applied; the calculated control commands of all other steps are ignored. Then, the control horizon moves forward for one time step; after that, the optimization process is repeated until the whole prediction horizon is covered.

This chapter is organized as follows: in section 7.1, PSO structure is described, followed by section 7.2 in which the effectiveness of this algorithm is investigated by optimizing some benchmark problems. In section 7.3, the control problem is transformed to a PSO structure. The final section shows the result of implementing PSO in the control problem.

7.1 Structure of PSO

PSO is a bio-inspired technique that seeks the optimum point of the objective function. It is initialized by a population of random particles. In the next iterations, each particle moves according to its velocity. The velocity of each particle is calculated based on both its own personal best position ($Pbest$), and the best position of the group obtained so far_global best ($Gbest$). The process of updating the position of each particle is repeated until the iteration number is reached, or the difference between the global best of the last two iterations is negligible. At the end, the algorithm returns the last global best as the final optimum.

In general, the formula for updating the position of PSO particles is as follows:

$$V_j^{t+1} = wV_j^t + c_1 r_1 \times \left(Pbest_j^t - P_j^t\right) c_2 r_2 \times \left(Gbest^t - P_j^t\right). \tag{7.1}$$

$$P_j^{t+1} = P_j^t + V_j^{t+1}.$$

Introduction of variables:

V_j^t	velocity of bird (particle) j at iteration t
w	inertia parameter
c_1, c_2	cognitive and social coefficient
r_1, r_2	random numbers from uniform distribution
P_j^t	position of bird (particle) j at t^{th} iteration
$Pbest_j^t$	best location of bird (particle) j in the previous iterations
$Gbest^t$	best position of the group obtained so far

c_1 and c_2 should be set up in advance; they are usually equal to two. w is an inertia parameter; mostly it is large at first iterations and gradually decreases, since, in many cases, the algorithm aims to search more freely at first iterations and converges at the end. w is updated according to equation 7.2.

$$w = w_{max} - \frac{w_{max} - w_{min}}{Iter_{max}} \times Iter. \quad (7.2)$$

In equation 7.2, variables are defined as below:

w_{max}	inertia weight at the beginning
w_{min}	inertia weight at the ending
$Iter_{max}$	Maximum number of iteration
$Iter$	Current number of iteration

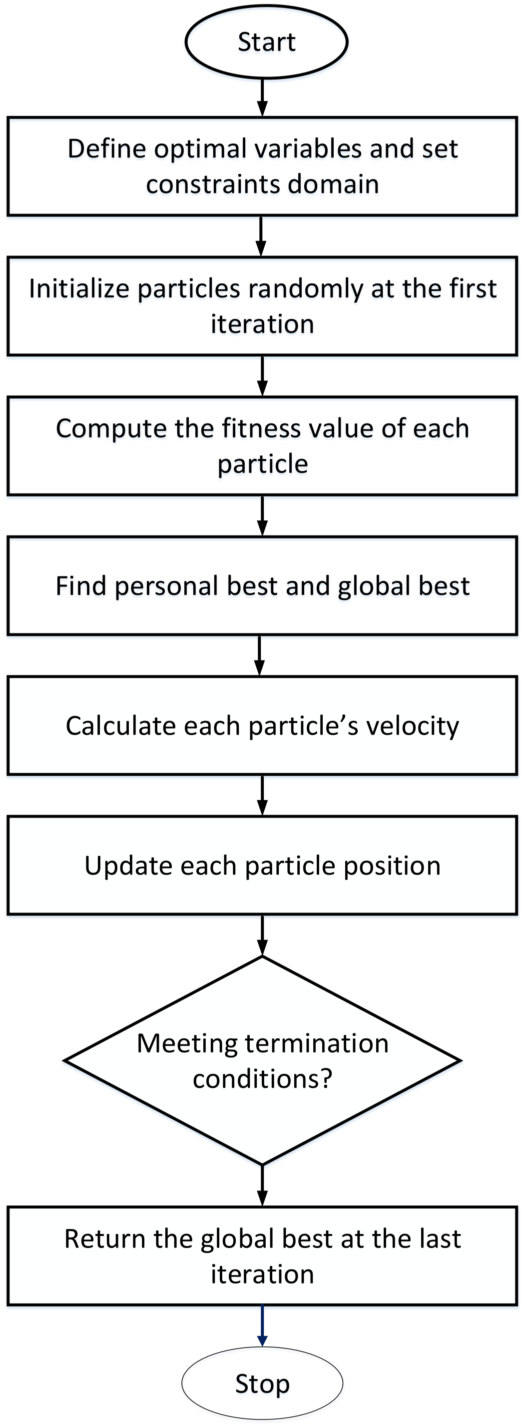

Fig. 7.1 PSO Flowchart [74]

7.2 Capability of PSO in solving complicated problems

Naturally, finding the global optimum for multimodal functions is more challenging than doing so for unimodal functions. In order to investigate the potential of PSO, it is applied to optimize the Schwefel function, which has a lot of local optimums [75].

This function is defined as below:

$$f(x) = 418.9829d - \sum_{i=1}^{d} x_i \sin(|x|)^{0.5}. \tag{7.3}$$

$$-500 \leq x_j \leq 500.$$

This function has a global optimum $f(x^*) = 0$ at x^*= (420.9687,…,420.9687)

The effectiveness of the PSO algorithm depends on the numbers of the following: the particles (population size), the dimensions of each particle, and the iterations. As illustrated in Fig. 7.1, the larger the numbers of particles and iteration are, the more effectively particles can search the feasible area, so the greater the computational time is required and the more accurate final answers will be. The following figures show the effects of these parameters.

As shown in Fig. 7.2, increasing the dimension of each particle makes the problem exponentially bigger and much more challenging to solve.

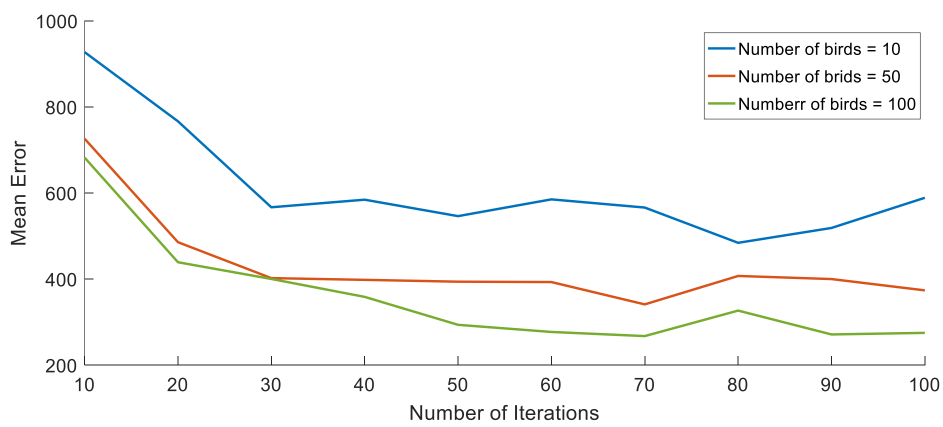

Fig. 7.2 Comparison of different iteration and particle numbers on optimization of Schwefel function (Dimension = 5)

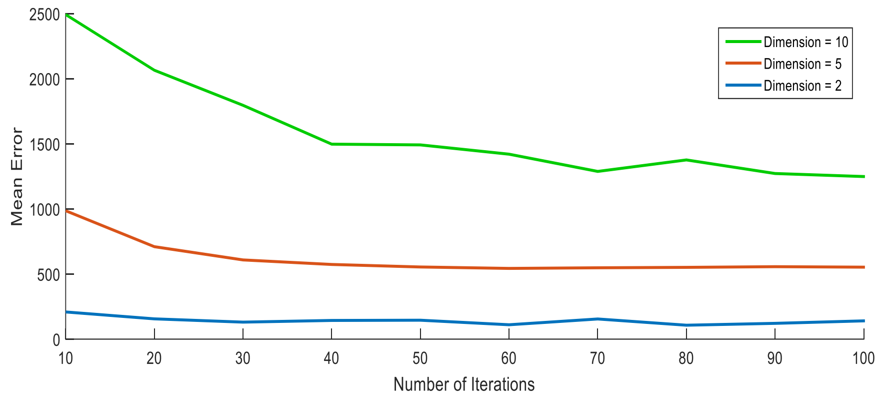

Fig. 7.3 Comparison of different iteration and dimension numbers on optimization of Schwefel function (Bird Number = 10)

7.3 Applying PSO-MPC to the control problem

In the problem at hand, PSO is implemented in the heart of the MPC. PSO aims to minimize the performance index over the control horizon (N_c). Consequently, it should determine the optimal power distribution at each second of the control horizon $\{r_1, r_2, r_3, \dots, and\ r_{N_c}\}$. As mentioned, in PSO, each particle is a solution. That is, to solve the control problem with the length of N_c, each particle has N_c dimensions. For instance, to solve this control problem for 100 seconds, it should calculate $r_1, r_2, r_3, \dots, and\ r_{100}$. In other words, each particle has 100 dimensions. At the last iteration, PSO chooses the particle that has the minimum performance index as the final answer.

Within the MPC frame, only the first dimension (r_1) is applied. Consequently, the engine receives the power from the battery according to r_1 in the first second. The car proceeds for

Table 7-1 Dimension = 100, Iteration = 100

Number of particles	Mean Error	Runtime
10	$3*10^5$	0.5 s
100	$1*10^2$	4 s
1000	$2*10^{-9}$	45 s
10000	$9*10^{-10}$	450 s

one second, then, the system will be updated; since there is more information about the route and the demanded power in the following seconds can be predicted more accurately. The PSO-MPC will be implemented in the next N_c seconds; this cycle will be repeated until the whole length of the prediction horizon (N_p) is covered. Table 7.1 shows the effect of the number of particles on the PSO-MPC's performance. As mentioned, since DP-MPC finds the global optimum, the evaluation of the PSO-MPC's performance is based on the DP-MPC solution. Obviously, the greater the number of particles is, the more-effectively the algorithm can search the solution area, leading to a more-optimal solution being found. On the other hand, a large number of particles increases the computational cost. Consequently, for the problem at hand, which requires a sufficiently fast controller to find the optimum in few milliseconds, PSO-MPC is not fast enough to do so. Decreasing the number of particles increases the convergence speed of the PSO-MPC, but the answer is not accurate enough. The computational cost of the PSO-MPC is a function of the following parameters:

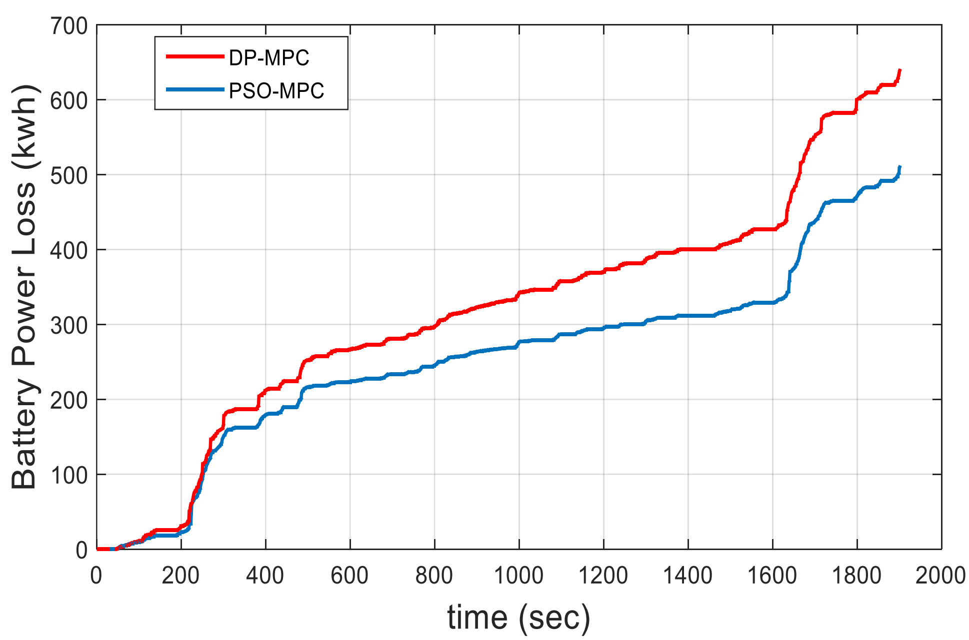

Fig. 7.4 PSO-MPC with an initial guess

$$Computational\ Cost = O(number\ of\ birds * dimension * iteration * time\ horizon) \tag{7.4}$$

Moreover, the standard PSO initializes with a random population, so in every run, it finds a different answer for the same problem. Thus, it is not stable enough to optimize control-based driving problems, which require high levels of reliability and numerical stability.

In order to enhance the stability and reduce the computational cost of the PSO-MPC, random initialization is replaced by initialization with the DP-MPC's solution. Fig. 7.4 shows the simulation result.

As shown, the battery power loss decreases by 20% over 1900 seconds. Although this combination is successful in terms of optimization, it shows some difficulties for online superfast optimization. Table 7.2 compares the run times of these methods.

Table 7-2 Comparing computational cost of different methods

Method	Time (seconds)	Final Battery Power Loss (kwh)
DP-MPC Grid Size = 10	8	640
DP-MPC Grid Size = 20	52	477
DP-MPC Grid Size = 30	165	452
PSO-MPC with Initial Guess (Grid Size = 10)	23	540

Chapter 8
Conclusion and Future Works

In this thesis, several widely used topologies have been reviewed and the best-fit topology has been chosen based upon the BEV-HSC characteristics. An accurate control-oriented model has been developed and utilized to define the optimization control problem to maximize the battery lifespan.

A number of EMSs of the BEV-HSC have been proposed using a few deterministic and probabilistic approaches for Toyota Rav4EV. Several simulations have examined the performance of the presented methods compared to performance of widely-used EMSs in the literature.

The NMPC has solved the problem at hand. The Newton/GMRES method, which is a fast optimizer, solves the optimization problem at the heart of the NMPC while the exterior penalty method handled the problem constraints. Also, the maximum potential of applying MPC for this hybrid system has been investigated by DP. The sensitivity analysis of DP-MPC parameters has been done. DP has yielded an affective baseline for comparison since it has provided the global optimum upon choosing a proper number of discretization.

In order to design a more effective EMS, a method to handle uncertainties has been sought. First, the future power demand has been predicted by a Markovian chain where the TPM has been created using real drive cycles. Then, SDP has been applied to find the optimum policy using a policy iteration algorithm. The simulation results have showed the effectiveness of the proposed algorithm. These results were based on a very short drive cycle (around 2000 seconds) which means that the long-term implementation of SDP can save a considerable amount of the stored charge in the battery, which in turn leads to extending the lifespan of the battery. On the other hand, SDP tries to protect the battery against fluctuations of the demanded power during sudden accelerations and brakes by assigning a higher share of the power demand to the supercapacitor.

Also, from an economical point of view, this approach is preferable since it only relies on offline calculations and does not require any extra equipment. Moreover, since it does not involve online calculations, it is suitable for real-time applications and the computational cost is not an issue.

Finally, the optimal control problem was solved by PSO-MPC. PSO is a fast bio-inspired optimization technique that search the solution area effectively. A sensitivity analysis of PSO-MPC parameters has been presented. Simulation results demonstrate that although PSO-MPC is successful in terms of optimality, it is not fast enough for real-time application.

8.1 Contributions

To the best knowledge of the author

- In this thesis, the Newton/GMRES-based NMPC approach has been utilized for the EMS of the BEV-HSC for the first time.
- DP-MPC has been implemented in the problem at hand for the first time.
- This investigation is the first utilization of SDP with no knowledge of future for the EMS of the BEV-HSC.
- TPM has been created using real driving information
- Also, this study is the first execution of PSO-MPC to this particular problem.

8.2 Future Works

- To implement SDP in this problem, the power demand is assumed as a Markovian state. That means the next power demand is predicted based on the current power demand and not the previous ones. Designing a new EMS controller by predicting the future power demands more accurately using previous ones is still in hand.
- Applying hardware-in-the-loop experiments can add more validity to the proposed approaches.
- The proposed EMSs can be improved in terms of robustness and stability.

- Designing more advanced methods to enhance the accuracy of power demand prediction can improve the MPC perfromance.
- Developing a faster version of PSO could make it more applicable for real-time applications.

References

[1] A. Mozaffari, N. L. Azad, J. K. Hedrick, “A Nonlinear Model Predictive Controller with Multi Agent Online Optimizer for Automative Coldstart Hydrocarbon Emissions Reduction,” *IEEE Transactions on Vehicular Technology Journal*, Vol.65, Issue 6, pp.4548-4563. 2016.
[2] C. Manzie, H. Watson, and S. Halgamuge, “Fuel Economy Improvements for Urban Driving: Hybrid vs. Intelligent Vehicles,” *Transportation Research Part C: Emerging Technologies*, vol. 15, no. 1, pp. 1–16, 2007.
[3] J. Hooker, “Optimal Driving for Single-vehicle Fuel Economy,” *Transportation Research Part A*: General, vol. 22, no. 3, pp. 183–201, 1988.
[4] Q. Gong, Y. Li, and Z.-R. Peng, “Optimal Power Management of Plugin HEV with Intelligent Transportation System,” *in Advanced intelligent mechatronics*, IEEE/ASME international conference, pp. 1–6, 2007.
[5] P. Golchoubian, N. L. Azad, “An Optimal Energy Management System for Electric Vehicles Hybridized with Supercapacitor,” *ASME Dynamic Systems and Control Conference,* DSCC, Columbus, Ohio, USA. Oct. 2015.
[6] HACKADAY Website, “http://hackaday.com/2017/01/19/will-supercapacitors-ever-replace-batteries/
[7] A. Styler, G. Podnar, P. Dille, M. Duescher, C. Bartley, and I. Nourbakhsh, “Active Management of a Heterogeneous Energy Store for Electric Vehicles,” *Integrated and Sustainable Transportation Systems. (FISTS)*, IEEE Forum on, pp. 20–25, Vienna, Austria, June 2011.
[8] fhttps://iee.ucsb.edu/mixing-batteries-and-supercapacitors-powerful-combination
[9] J. M. Miller, “Energy Storage Technology Markets and Applications: Ultracapacitors in Combinationatin with Lithium_ion,” *7th International Conference on Power Electronics*, pp. 16-22, Daegu, Korea, Oct. 2007.
[10] CreateLab. The Chargecar Project, http://www.chargecar.org, 2012.
[11] R. Carter, A. Cruden, and P. J. Hall, “Optimizing for Efficiency or Battery Life in Battery/Supercapacitor Electric Vehicle,” *Transactions on Vehicular Technology*, vol. 61, no. 4, pp. 1526–1533, May 2012.
[12] M. E. Choi, J. S. Lee, and S. W. Seo, “Real-time Optimization for Power Management Systems of a Battery/Supercapacitor Hybrid Energy Storage System in Electric Vehicles,” *Transactions on Vehicular Technology*, vol. 63, no. 8, pp. 3600–3611, Oct. 2014.
[13] L. Cheng, W. Wang, S. Wei, H. Lin, Z. Jia, “An Improved Energy Management Strategy for Hybrid Energy Storage System in Light Rail Vehicles,” *Energies*, 11:423, 2018.
[14] S. Tajeddin, M. Vajedi, N. L. Azad, “A Newton/GMRES Approach to Predictive Ecological Adaptive Cruise Control of a Plug-in Hybrid Electric Vehicle in Car-followin Scenarios,” *IFAC-papersOnLine*, 49-21, 2016.
[15] B. Asadi, and A. Vahidi, “Predictive Cruise Control: Utilizing Upcoming Traffic Signal Information for Improving Fuel Economy and Reducing Trip Time,” *Control Systems Technology, IEEE Transactions,* 19(3), 707–714, 2011.
[16] N. Kohut, K. Hedrick, and F. Borrelli, “Integrating Traffic Data and Model Predictive Control to Improve Fuel Economy,”*12th IFAC Symposium on control*, 2009.
[17] Li-hua. Luo, Hong Liu, Ping Li, and Hui Wang, “Model Predictive Control for Adaptive Cruise Control with Multi-objectives: Comfort, Fuel-economy, Safety and Car-following,” *Journal of Zhejiang University SCIENCE A*, 11(3), 191–201, 2010.
[18] P. Golchoubian, N. L. Azad and K. Ponnambalam, “Stochastic Nonlinear Model Predictive Control of Battery-Supercapacitor Hybrid Energy Storage Systems in Electric Vehicles,” *American Control Conference*, Seattle, USA, May 2017.
[19] R. Bellman, “Dynamic Programming,” Princeton University Press, 1957.
[20] J. Rust, “Numerical Dynamic Programming in Economics,” *Handbook of Computational Economics*, edition 1, volume 1, chapter 14, 619-729 Elsevier, 1996.
[21] C. C. Lin, H. Peng and J. W. A. Grizzle, “A Stochastic Control Strategy for Hybrid Electric Vehicles,” *Proceedings of the American Control Conference*, Boston, USA, pp. 4710–4715, 2004.
[22] R. C. Eberhart, and J. Kennedy “A New Optimizer Using Particle Swarm Theory,” *Proceedings of the Sixth International Symposium on Micromachine and Human Science*, Nagoya, Japan. pp. 39-43, 1995.
[23] J. Cao and A. Emadi. “A New Battery/Ultracapacitor Hybrid Energy Storage System for Electric, Hybrid, and Plug-in Hybrid Electric Vehicles,” *IEEE Transactions on Power Electronics*, 27(1):122-132, Jan. 2012.

[24] M. Ortuzar, J. Moreno, and J. Dixon, "Ultracapacitor-based Auxiliary Energy System for an Electric Vehicle: Implementation and Evaluation," *IEEE Transactions on Industrial Electronics,* 54(4):2147-2156, Aug. 2007.
[25] A. Di Napoli, F. Crescimbini, F. Giuli Capponi, and L. Solero, "Control Strategy for Multiple Input DC-DC Power Converters Devoted to Hybrid Vehicle Propulsion Systems," *IEEE International Symposium on Industrial Electronics,* 3:1036-1041, vol.3, 2002.
[26] J. P. Trovao, P. G. Pereirinha, H. M. Jorge, and C. H. Antunes, "A Multi-level Energy Management System for Multi-source Electric Vehicles - An Integrated Rule-based Meta-heuristic Approach," *Applied Energy,* 105:304 - 318, 2013.
[27] D. Tarsitano, L. Mazzola, S. Arrigoni, F. L. Mapelli, and F. Cheli, "Energy Management Algorithms Comparison for an Electric Bus with a Hybrid Energy Storage System by Means of Dynamic Programming," *2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall),* pp. 1-5, Sept. 2014.
[28] Z. Song, H. Hofmann, J. Li, X. Han, and M. Ouyang, "Optimization for a Hybrid Energy Storage System in Electric Vehicles using Dynamic Programing Approach," *Applied Energy,* 139:151 - 162, 2015.
[29] A. Castaings, W. Lhomme, R. Trigui, and A. Bouscayrol, "Comparison of Energy Management Strategies of a Battery/Supercapacitors System for Electric Vehicle Under Real-time Constraints," *Applied Energy,* 163:190 - 200, 2016.
[30] M. Vajedi and N. L. Azad, "Ecological Adaptive Cruise Controller for Plug-in Hybrid Electric Vehicles using Nonlinear Model Predictive Control," *IEEE Transactions on Intelligent Transportation Systems,* 17(1):113 - 122, Jan. 2016.
[31] B. Hredzak, V. G. Agelidis, and M. Jang, "A Model Predictive Control System for a Hybrid Battery-Ultracapacitor Power Source," *IEEE Transactions on Power Electronics,* 29(3):1469-1479, March 2014.
[32] M. Jalalmaab, "Stochastic Power Management Strategy for In-Wheel Motor Electric Vehicles," *MASc Thesis,* Universiy of Waterloo, 2014.
[33] S. Ermon, Y. Xue, C. Gomes, and B. Selman, "Learning Policies for Battery Usage Optimizaiton in Electric Vehicles," *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Bristol, UK, pp. 195–210, Sept. 2012.
[34] O. Laldin, M. Moshirvaziri, and O. Trescases, "Predictive Algorithm for Optimizing Power Flow in Hybrid Ultracapacitor/Battery Storage Systems for Light Electric Vehicles," *IEEE Transactions on Power Electronics*, vol. 28, no. 8, pp. 3882–3895, Aug. 2013.
[35] S. Wang, S. Shan, and X. Luo, "Particle-Swarm Optimization Algorithm for Model Predictive Control of MIMO with Constraints," *10th Word Congress on Intelligent Control and Automation*, pp. 2576-2581, 2012.
[36] Q. Zou, J. Ji, S. Zhang, M. Shi, Y. Luo, "Model Predictive Control Based on Particle Swarm Optimization of Greenhouse Climate for Saving Energy Consumption," *IEEE World Automation Congress (WAC)*, Kobe, Japan, pp. 123-128, 2010.
[37] R. Susuki, F. Kawai, C. Nakazawa, T. Matsui, E. Aiyoshi, "Parameter Optimization of Model Predictive Control using PSO," *SICE Annual Conference*, pp. 1981–1988, Aug. 2008.
[38] L. Re, P. Ortner, and D. Alberer, "Chances and Challenges in Automotive Predictive Control*," Lecture Notes in Control and Information Sciences, Springer, London*, vol. 402, pp. 1-22, 2010.
[39] L. Wang, "Model Predictive Control System Design and Implementation Using MATLAB," *Springer*, 2009.
[40] T. S. Kim, C. Manzie, and R. Sharma, "Model Predictive Control of Velocity and Torque Split in a Parallel Hybrid Vehicle," *Systems, Man and Cybernetics*, IEEE International Conference on, pp. 2014-2019, IEEE, Oct. 2009.
[41] H. Borhan, A. Vahidi, A. M. Phillips, and I. V. Kolmanovsky, "Predictive Energy Management of a Power-Split Hybrid Electric Vehicle," in *American Control Conference*, pp. 3970-3976, IEEE, 2009.
[42] H. Borhan, A. Vahidi, A. M. Phillips, M. Kuang, I. V. Kolmanovsky, and S. D. Cairano, "MPC-Based Energy Management of a Power-Split Hybrid Electric Vehicle," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 593-603, 2012.
[43] S. Tajeddin, "Automatic Code Generation of Real-Time Nonlinear Model Predictive Control for Plug-in Hybrid Electric Vehicle Intelligent Cruise Controllers," *MASc Thesis,* University of Waterloo, 2016.
[44] F. Allgower and A. Zheng, "Nonlinear Model Predictive Control," vol. 26. Birkhauser, 2012.
[45] C. E. Beal and J. C. Gerdes, "Model Predictive Control for Vehicle Stabilization at the Limits of Handling," *Control Systems Technology,* IEEE Transactions on, vol. 21, no. 4, pp. 1258-1269, 2013.

[46] P. Shakouri, A. Ordys, and M. R. Askari, "Adaptive Cruise Control with Stop & Go Function using the State-dependent Nonlinear Model Predictive Control Approach," *ISA Transactions*, vol. 51, no. 5, pp. 622-631, 2012.
[47] M. Huang, H. Nakada, K. Butts, and I. Kolmanovsky, "Nonlinear Model Predictive Control of a Diesel Engine Air Path: A Comparison of Constraint Handling and Computational Strategies," *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 372-379, 2015.
[48] T. Ohtsuka. "Continuation/GMRES Method for Fast Algorithm of Nonlinear Receding Horizon Control," *In Decision and Control, Proceedings of the 39th IEEE Conference on*, volume 1, pages 766-771 vol.1, 2000.
[49] Moritz Diehl, H. Georg Bock, Johannes P. Schloder, Rolf Findeisen, Zoltan Nagy, and Frank Allgower, "Real-time Optimization and Nonlinear Model Predictive Control of Processes Governed by Differential-algebraic Equations" *Journal of Process Control*, 12(4):577 - 585, 2002.
[50] C.-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*, vol. 20, no. Nov. 2003.
[51] C.-c. Lin, H. Peng, and J. Grizzle, "Power Management Strategy for a Parallel Hybrid Electric Truck," *IEEE Transactions on Control Systems Technology*, vol. 11, pp. 839-849, Nov. 2003.
[52] Q. Gong, Y. Li, and Z.-R. Peng, "Trip-based Optimal Power Management of Plug-in Hybrid Electric Vehicles," *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 6, pp. 3393-3401, 2008.
[53] M. P. O. Keefe and T. Markel, "Dynamic Programming Applied to Investigate Energy Management Strategies for a Plug-in HEV," *22nd International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium and Exhibition*, Yokohama, Japan, Oct. 2006.
[54] X. Lin, M. Hu, S. Song, and Y. Yang, "Battery-Supercapacitor Electric Vehicles Energy Management using DP based Predictive Control Algorithm," In *Computational Intelligence in Vehicles and Transportation Systems (CIVTS), 2014 IEEE Symposium on*, pp. 30-35, Dec. 2014.
[55] H. Zhang, Y. Qin, X. Li, X. Liu, and M. Engineering, "Driver-Oriented Optimization of Power Management in Plug-In Hybrid Electric Vehicles," in *EIC Climate Change Technology Conference*, no. 1569730735, pp. 1–12, 2013.
[56] D. F. Opila, R. McGee, J. A. Cook, and J. W. Grizzle, "Performance Comparison of Hybrid Vehicle Energy Management Controllers on Real-world Drive Cycle Data," *2009 American Control Conference*, pp. 4618–4625, 2009.
[57] V. Schwarzer, R. Ghorbani, and R. Rocheleau, "Drive Cycle Generation for Stochastic Optimization of Energy Management Controller for Hybrid Vehicles," *2010 IEEE International Conference on Control Applications*, pp. 536–540, Sept. 2010.
[58] www.swarmintelligence.org/tutorial.php
[59] K. Y. Lee, and J. Park, "Application of Particle Swarm Optimization to Economic Dispatch Problem: Advantages and Disadvantages," *2006 IEEE PES Power Systems Conference and Exposition*, pp. 188-192, 2006.
[60] L. S. Coelho, and F.A. Guerra, "Applying Particle Swarm Optimization to Adaptive Controller," *Soft Computing in Industrial Applications*, vol. 39, Springer, Berlin, Heidelberg, pp. 82-91, 2007.
[61] L. S. Coelho, R. A. Krohling, "Predictive Controller Tuning using Modified Particle Swarm Optimization based on Cauchy and Gaussian Distributions," *Soft Computing: Methodologies and Applications*, Springer Engineering series in Advances in Soft Computing, pp. 287–98, 2005.
[62] X. H. Wang and J. M. Xiao, "PSO-based Model Predictive Control for Nonlinear Processes," *Lecture Notes in Computer Science*, ser. 3611. Berlin, Germany: Springer-Verlag, pp. 196–203, 2005.
[63] M. Han, J. Fan, and J. Wang, "A Dynamic Feedforward Neural Network based on Gaussian Particle Swarm Optimization and its Application for Predictive Control," *IEEE Transactions on Neural Networks*, vol. 22, no. 9, pp. 1457–1468, Sept. 2011.
[64] M. Batra, J. McPhee, and N. Azad, "Parameter Identification for a Longitudinal Dynamics Model Based on Road Tests of an Electric Vehicle," *ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, Charlotte, North Carolina, USA, Aug. 2016.
[65] P. Golchoubian, and Nasser L. Azad, "Real-Time Nonlinear Model Predictive Control of a Battery-Supercapacitor Hybrid Energy Storage System in Electric Vehicles," *IEEE Transaction on Vehicular Technology*, vol. 66, no. 11, pp. 9678-9688, 2017.
[66] R. T. Meyer, R. A. DeCarlo, and S. Pekarek, "Hybrid Model Predictive Power Management of a Battery-Supercapacitor Electric Vehicle," *Asian J. Control*, vol. 18, no. 1, pp. 150–165, 2016.

[67] P. Golchoubian, “Real-time Energy Management of a Battery Electric Vehicle Hybridized with Supercapacitor”, *MASc Thesis*, Universiy of Waterloo, 2017.
[68] C. T. Kelley, “Iterative Methods for Linear and Nonlinear Equations,” *Raleigh N. C.: North Carolina State University*, 1995.
[69] S. Tajeddin and N. L. Azad, “Ecological Cruise Control of a Plug-in Hybrid Electric Vehicle: A comparison of different GMRES-based Nonlinear Model Predictive Controls,” *American Control Conference (ACC)*, Seattle, WA, pp. 3607-3612, 2017.
[70] Donald E Kirk. “Optimal control theory: an introduction,” *Courier Corporation*, 2012.
[71] Charge car: Driving data. [Online]: http://www.chargecar.org/data
[72] M. L. Puterman, “Markov Decision Processes: Discrete Stochastic Dynamic Programming,*” John Wiley & Sons*, p. 672, 2009.
[73] C. Lin, H. Peng, J.W. Grizzle and J. Kang, “Power Management Strategy for a Parallel Hybrid Electric Truck”, *IEEE Trans Control Systems Technology*, vol. 11, issue: 6, Nov. 2003.
[74] Ye L., Yang M., Xu L., Zhuang X., Dong Z., Li S. “Nonlinearity Analysis and Parameters Optimization for an Inductive Angle Sensor,” *Sensors*, 14:4111–4125, 2014.
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4003933/
[75] https://www.sfu.ca/~ssurjano/schwef.html