

Article

Model Predictive Control-Based Fast Charging for Vehicular Batteries

Jingyu Yan ^{1,2}, Guoqing Xu ^{1,2,3,*}, Huihuan Qian ^{1,2}, Yangsheng Xu ^{1,2} and Zhibin Song ¹

¹ Shenzhen Institutes of Advanced Technology, The Chinese Academy of Science, Shenzhen 518055, China; E-Mails: jyyan@mae.cuhk.edu.hk (J.Y.); hhqian@mae.cuhk.edu.hk (H.Q.); ysxu@mae.cuhk.edu.hk (Y.X.); zb.song@siat.ac.cn (Z.S.)

² Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong, China

³ Department of Electrical Engineering, Tongji University, Shanghai 200092, China

* Author to whom correspondence should be addressed; E-Mail: gq.xu@siat.ac.cn; Tel.: +84-755-86392148.

Received: 13 June 2011; in revised form: 2 August 2011 / Accepted: 4 August 2011 /

Published: 17 August 2011

Abstract: Battery fast charging is one of the most significant and difficult techniques affecting the commercialization of electric vehicles (EVs). In this paper, we propose a fast charge framework based on model predictive control, with the aim of simultaneously reducing the charge duration, which represents the out-of-service time of vehicles, and the increase in temperature, which represents safety and energy efficiency during the charge process. The RC model is employed to predict the future State of Charge (SOC). A single mode lumped-parameter thermal model and a neural network trained by real experimental data are also applied to predict the future temperature in simulations and experiments respectively. A genetic algorithm is then applied to find the best charge sequence under a specified fitness function, which consists of two objectives: minimizing the charging duration and minimizing the increase in temperature. Both simulation and experiment demonstrate that the Pareto front of the proposed method dominates that of the most popular constant current constant voltage (CCCV) charge method.

Keywords: battery fast charging; model predictive control; state of charge; genetic algorithm; electric vehicles

1. Introduction

It is necessary to refuel all vehicles after a period of travel. For conventional fuel-driven vehicles, the fuel can be replenished quickly and safely at filling stations, with a minimal out-of-service time and without energy loss. However, the replenishment of energy in electric vehicles requires charging the battery pack. In general applications, such as MP3 players, shavers, cell phones, the charging process may last several hours, leading to a long out-of-service time. The charging method is also crucial; an unsuitable method leads to high increase in temperature in a battery and even the risk of fire and explosion.

Therefore, charging control is a significant issue in battery management systems, with the aim of feeding external energy into batteries in a fast, safe, and efficient way. Fast charging helps to reduce out-of-service time and promote the commercialization of EVS. Safe charging not only assures the safety of users by preventing battery burning and explosions during the charging process, but also prolongs the battery life by preventing overcharging and overheating damage. Efficient charging can convert as much electrical energy as possible from a charger to electrochemical energy stored in a battery so as to enhance energy efficiency.

1.1. Literature Review

Charging methods have been studied since the invention of rechargeable batteries. The earliest and the most common charging method is named the constant trickle current (CTC) [1] method, which has a simple circuit structure and very low cost. Thus, the constant trickle current method has been adopted in most electronic products for many years. Because this method provides a very small current, the charging time is extremely long so that it usually works overnight. An easy way to reduce the charging time is to increase the constant current (CC). Charging with 1 C current can fill an empty battery in one hour. However, one drawback in the CC charging method is that it requires a very precise state of charge (SOC) estimator which has the ability to determine whether the battery is fully charged and whether to stop the charging process. Unfortunately, the precise SOC estimator is not easy to implement. Another disadvantage is that CC charging cannot avoid over-voltage to the battery. A battery can be simply modeled with an open circuit voltage (OCV) source and a series-connected resistor. The voltage on the resistor will cause the terminal voltage to be always higher than the OCV during CC charging, finally leading to over-voltage when the OCV approaches its full value. Taking a Li-ion battery as an example, the over-voltage during charging will degrade the crystallographic structure of the cathode and cause oxidative decomposition of the electrolytic solvents [2].

The constant current constant voltage (CCCV) method has been proposed to overcome the disadvantages of CC charging. A CC period is applied until the charging voltage reaches a predetermined value, and then the charging process goes into the CV period. In the CV period, the charging voltage is fixed to a cutoff value; therefore, the charging current will be automatically reduced along with the increase of SOC. Generally, the CV period requires a long charging period [3]. In order to enhance charging performance of CCCV, various combinations of CC and CV periods have been proposed in recent years, such as CCCVCV [4], (CCCV)² [5], and so forth. A smooth control

circuit (SCC) is proposed to ensure the stable transition from the constant-current (CC) to the constant-voltage (CV) stage [6]. A further review of CCCV family is given in [7].

In recent years, since microprocessor control units (MCUs) and even computers have been applied to monitor and control the charging process at charging station for EVs, so some intelligent methods can be put into practice. One kind of intelligent charger controls the charging process by detecting the tuning points. The tuning points are selected as the threshold points, stationary points, and inflection points of battery voltage, temperature, and lapsed time [8–10]. These intelligent methods only change the charging behavior at tuning points and the periods between tuning points are still under open-loop control. Fuzzy control has been also applied to solve the charging control problem [3,11]. Optimal control based on power loss model was adopted to enhance the charging efficiency [12]. In [13–15], neural networks and genetic algorithms are also introduced to design the membership functions and inference rules of the fuzzy controller. Optimization methods, such as the ant colony algorithm and evolutionary algorithm, have also been introduced to optimize a best multi-stage CC charging profile [16,17]. The optimization does not aim to regulate the real-time charging current according to instant system measurements, but rather to find a best charging profile covering the whole charging process. Therefore, its performance is seriously dependent on the accuracy of the model used in optimization. Suffering from system noise and the time-variance properties of batteries, the robustness of these methods is weak. In [18], to increase the charging speed, a grey prediction technique is utilized to develop a grey-predicted control system. In that study, the control system does not take increase in temperature into consideration and only applies a one-step prediction. The small predictive horizon weakens the prediction of future system behavior.

1.2. Overview of Proposed Charging Controller

Modeling of batteries has been comprehensively studied in recent years, and some are demonstrated to be accurate and efficient enough to model the behaviors of batteries [19–23]. These models make it possible to apply model predictive control (MPC) to manage the battery charging process.

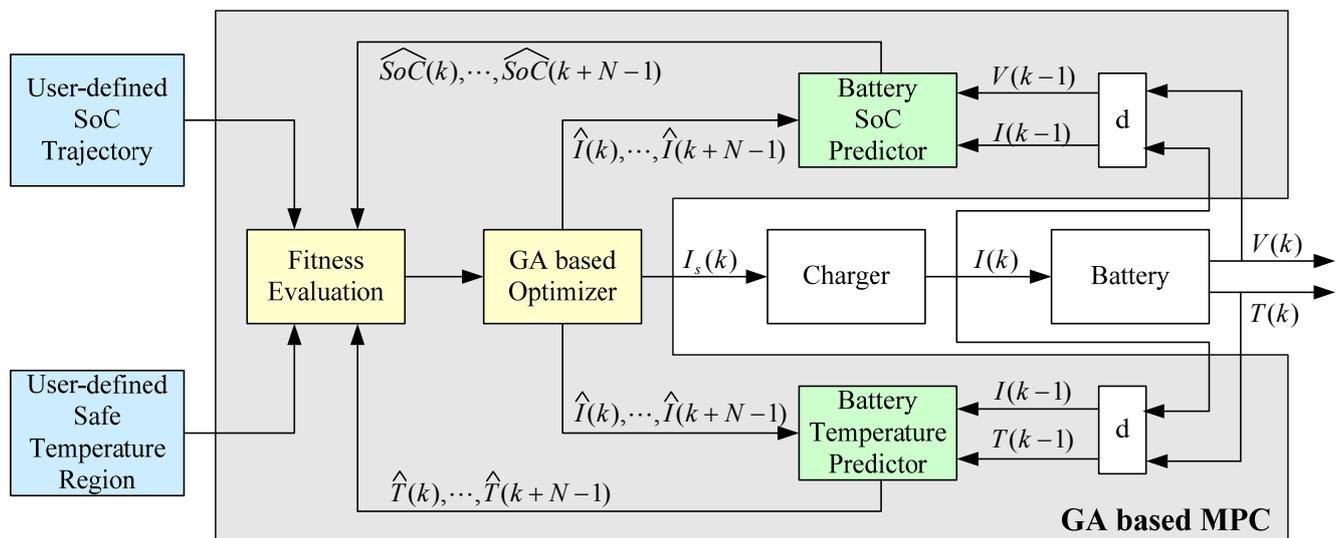
MPC is an advanced control method and has been widely applied in many fields [24,25]. The working principle of MPC is to apply system models to predict system responses to possible future control sequences and to find the best future control sequence by optimizing the user-defined objective function. Only the first element of the control sequence is applied to the controlled system at each time. The system states are then sampled again and the calculations are repeated at the next control time. The prediction horizon is constantly shifted forward. Because the receding horizon strategy updates predictions based on the instant measurement of system inputs and outputs at each control time, its robustness has been demonstrated to be strong [26]. MPC may be categorized as either linear MPC or nonlinear MPC, according to the linearity of the system models and constraints. For linear MPC, the best control sequence can be analytically obtained by solving Diophantine equations. However, for nonlinear MPC, the best control sequence usually is calculated numerically under an optimization framework [27].

Available battery models establish the basis for applying a MPC framework to solve the charge control problem. Three advantages of MPC explain the rationale. Firstly, it has the ability to predict future battery states under a possible future charging sequence and hence evaluate charging performance

over a comparatively long period. Secondly, it calculates the best charging sequence using an optimization method, which can solve multi-objective problems. Thirdly, it utilizes a receding horizon strategy and only applies the first element of the best charging sequence at each control moment. The prediction error caused by model inaccuracy will not accumulate because instantaneous measurements correctly update each initial prediction value. As shown in Figure 1, the fast charging control framework proposed in this work consists of the following components:

1. A SOC predictor, predicting the SOC of battery when fed by a sequence of future charging current;
2. A temperature predictor, predicting the future battery temperature under a sequence of charging current;
3. A fitness evaluator, evaluating the performance of the sequence of charging current;
4. An optimizer, finding the best sequence of charging current using genetic algorithm (GA).

Figure 1. Fast charging control framework based on model predictive control.



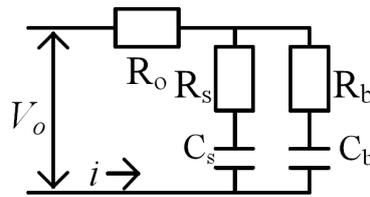
2. Predictive Models

To utilize predictive control, we apply battery models to predict future battery states under a sequence of future charging current. The battery states of interest in this study are SOC and temperature. The former indicates the charging duration and whether overcharge damage is possible, while the latter represents the safety and energy efficiency during the charging process. Charging control is necessary to: (1) minimize charging duration; (2) minimize temperature increases; and (3) restrict temperature to a safe range.

2.1. RC Model for SOC Prediction

Figure 2 describes the RC equivalent circuit model [28], which is applied in this paper. This model consists of a bulk capacitor C_b which simulates energy storage and a surface capacitor C_s which represents the dynamic property of the battery. Output resistance R_0 , surface resistance R_s and bulk resistance R_b are utilized to model the internal resistance of the battery.

Figure 2. The battery RC Model.



In this paper, we denote k and T_s as the time index and sampling time, $V_b(k)$ and $V_s(k)$ as the voltages of bulk capacitor and surface capacitor respectively, $I(k)$ as the charging current, $V_o(k)$ as the terminal voltage. The voltage of bulk and surface capacitor are selected as the state vectors. Charging current is selected as the system input and terminal voltage is selected as the system output. System discrete time-variant state space equations are expressed as follows:

$$x(k + 1) = A(k)x(k) + B(k)u(k) \tag{1}$$

$$y(k) = C(k)x(k) + D(k)u(k) \tag{2}$$

where:

$$x(k) = [V_b(k), V_s(k)]^T \tag{3}$$

$$u(k) = I(k) \tag{4}$$

$$y(k) = V_o(k) \tag{5}$$

$$A(k) = \begin{bmatrix} 1 - \frac{T_s}{C_b(k)[R_b(k) + R_s(k)]} & \frac{T_s}{C_b(k)[R_b(k) + R_s(k)]} \\ \frac{T_s}{C_s(k)[R_b(k) + R_s(k)]} & 1 - \frac{T_s}{C_s(k)[R_b(k) + R_s(k)]} \end{bmatrix} \tag{6}$$

$$B(k) = \begin{bmatrix} \frac{-R_s(k)T_s}{C_b(k)[R_b(k) + R_s(k)]} \\ \frac{-R_b(k)T_s}{C_s(k)[R_b(k) + R_s(k)]} \end{bmatrix} \tag{7}$$

$$C = \begin{bmatrix} \frac{R_s(k)}{R_b(k) + R_s(k)} & \frac{R_b(k)}{R_b(k) + R_s(k)} \end{bmatrix} \tag{8}$$

$$D = -R_o(k) - \frac{R_b(k)R_s(k)}{R_b(k) + R_s(k)} \tag{9}$$

SOC in the RC model is estimated using the voltages of the two capacitors based on the relationship between SOC and the open-circuit voltage (OCV). Since C_b represents the bulk energy in the battery, it contributes the majority of the battery SOC, as expressed as follows:

$$\overline{SOC} = \frac{1}{21} [20\overline{SOC}_{C_b}(k) + \overline{SOC}_{C_s}(k)] \tag{10}$$

where:

$$\overline{SOC}_{C_b}(k) = F_{OCV-SOC}(V_b(k)) \quad (11)$$

$$\overline{SOC}_{C_s}(k) = F_{OCV-SOC}(V_s(k)) \quad (12)$$

$F_{OCV-SOC}(\cdot)$ is the function mapping OCV to SOC. It is usually predetermined by the manufacturer's datasheet or experimental data.

2.2. Thermal Model in Simulation

Reference [28] proposed a simple single-node lumped-parameter thermal model for batteries. It models the thermal process in three stages. In the first stage, the Joule effect generates heat in the battery. In the second, the battery's heat is conducted and convected to the surrounding air. Finally, the surrounding air exchanges heat with the ambient. In the RC model, the heat generation is expressed by:

$$Q_g(k) = T_s[I_0^2(k)R_0(k) + I_s^2(k)R_s(k) + I_b^2(k)R_b(k)] \quad (13)$$

where:

$$I_0(k) = I(k) \quad (14)$$

$$I_b(k) = \frac{V_b(k) - V_s(k) + R_s(k)I(k)}{R_s(k) + R_b(k)} \quad (15)$$

$$I_s(k) = I_b(k) - I(k) \quad (16)$$

Meanwhile, the heat passing from the battery to the surrounding air is expressed by:

$$Q_p(k) = \frac{T(k-1) - T_{air}(k-1)}{R_{eff}} \quad (17)$$

where T_{air} represents the effective temperature of surrounding air and R_{eff} stands for the effective thermal resistance. Based on the assumption that 50% of the heat from the battery is spent to warm the air [28], T_{air} is expressed by:

$$T_{air}(k-1) = T_{amb} - \frac{0.5 \times Q_p(k-1)}{\dot{m}_{air} C_{air}} \quad (18)$$

where T_{amb} is the ambient temperature, $Q_p(k-1)$ is the passing heat in previous step, \dot{m}_{air} is the airflow rate, and C_{air} is its heat capacity.

Strictly speaking, the value of R_{eff} depends on the thermal control method. For example, it becomes smaller if the cooling fans are open. However, in this study, we assume the charging process is conducted in an environment with only natural convection. In this case, R_{eff} is fixed as a constant with the value calculated by:

$$R_{eff} = \frac{1}{hA} + \frac{t}{kA} \quad (19)$$

where h is the heat transfer coefficient in the natural environment, A is the total module surface area exposed to the air, t is the thickness of the module case, and k is the thermal conductivity of the module case material.

We denote $Q_g(k)$ as the heat generation which heats the battery and $Q_p(k)$ as the heat loss which cools the battery by heating the surrounding air. Therefore, the battery temperature $T(k)$ can be calculated by:

$$T(k) = T(k-1) + \frac{Q_g(k) - Q_p(k)}{m_{bat} C_{bat}} \quad (20)$$

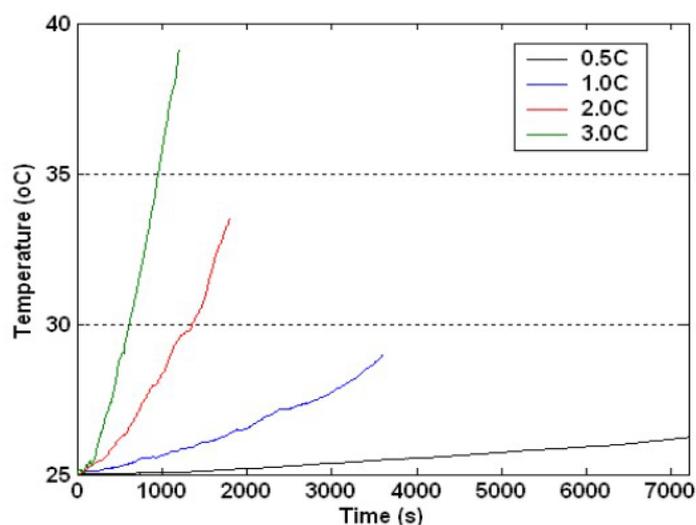
where m_{bat} is the mass of the battery, C_{bat} is heat capacity of the battery.

2.3. Thermal Model in Experiment

The simulation model requires many physical parameters of the battery and working environment. However, in practice, these parameters usually cannot be obtained accurately. Meanwhile, the theoretical model only considers the ideal situation so that it may not be suitable for the complex and nonlinear electrochemical process inside the battery. Furthermore, heat generation is only caused by heating of the resistors, which does not fully represent the actual situation. In reality, heat generation during charge is also affected by the charging acceptance rate. In [5], it pointed out that “close-to-fully discharged batteries can be recharged with very high currents for a short period of time” and [29] concluded that for A123, a kind of LiFePO₄, “at a low state of charge, nearly all the charging current is absorbed by the chemical reaction. Above 80% of SOC, more and more energy goes into heat”.

Figure 3 shows the temperature curves of our experimental LiFePO₄ cells during charge processes with different charging rates. The increase in temperature agrees with the two results reported in the literature above.

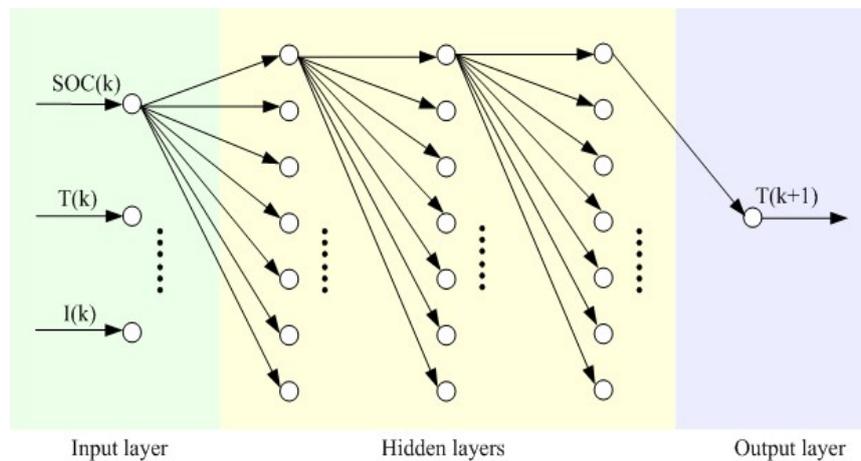
Figure 3. Increase in temperature under different charging rates.



The increase in temperature obviously also has a close relationship with the charging rate. High charging rates will cause high increase in temperature. Therefore, we select the SOC, the charging rate and the current temperature to predict the future temperature based on a neural network (NN).

Figure 4 shows the structure of the NN. The training set consists of the real data recorded in the charging process, as shown in Figure 3. The input vector contains three parameters of interest: $SOC(k)$, $T(k)$ and $I(k)$, and the output or the predictive result is $T(k+1)$. T is directly measured by a temperature sensor while I and SOC is measured and estimated respectively by the programmable charger IT6154, measurement of current of which is accurate enough to calculate SOC using coulombic counting method. Since the whole charging process is recorded, the input vector at time k and the output at time $k+1$ are known to train the neural network.

Figure 4. Neural network for predicting battery temperature.



The testing set also consists of the real data recorded in charging process, including the batteries used in training set and those only used in testing set. The predictive error is defined as the expression:

$$\left| \frac{\hat{T}(k+1) - T(k+1)}{T(k+1)} \right| \times 100\%$$

where $\hat{T}(k+1)$ is the predictive result by the neural network and $T(k+1)$ is the measured result during the charging process. The training method is back propagation (BP) learning. When the time step is set to 30s, the average prediction error is 3.36% and the maximum is 5.43%.

2.4. Model Based Prediction

To predict battery states under a given sequence of future charging current, two problems should be solved: (1) how to initialize the prediction at time k ; and (2) how to realize the multi-step prediction based on the one-step prediction model.

In order to initialize the prediction at time k , the RC model must know the last system state $x(k-1)$, the last system input $u(k-1)$, and the model parameters at time k . Among them, $u(k-1)$ and $y(k-1)$ can be obtained from the current sensor and voltage measurement, respectively, while $x(k-1)$ can be obtained by closed-loop estimators such as the extended Kalman filter [30], sigma-point Kalman filter [31,32], H_∞ filter [33], and so on. The closed-loop filters are able to eliminate accumulated errors and the estimation result will gradually converge to the real value, or to a limited error range. The model parameters are stored in a look-up table indexed by SOC and temperature. The table is constructed using the off-line testing data. The slow changes in SOC and

temperature give the reason for updating the model parameters at time k according to $SOC(k-1)$ and $T(k-1)$, especially when T_s is comparatively small. $SOC(k-1)$ is calculated according to Equation (10), and $T(k-1)$ is measured directly from a temperature sensor fixed on the surface of the battery.

Given the value of $I(k)$, the above preparation allows a one-step prediction of $SOC(k)$ and $T(k)$. However, a single-step prediction is usually insufficient to predict system behavior over a long process. Generally speaking, the prediction horizon in MPC is more than one step. Therefore, a multi-step predictor is necessary to predict system states exerted by a sequence of future charging current. A simple way to realize the multi-step predictor is an iterative prediction, which uses the predictive future system states at time $k+j$ as the initial state for the next time $k+j+1$.

In summary, at control time k , the SOC model and temperature model can iteratively map a sequence of future charging current $[I(k), I(k+1), \dots, I(k+p-1)]$ to future battery states $[SOC(k), SOC(k+1), \dots, SOC(k+p-1)]$ and $[T(k), T(k+1), \dots, T(k+p-1)]$. The multi-step prediction uses an open-loop manner in which each step suffers from the prediction error in the previous step and finally reverts to the errors in the initial values at time $k-1$. Fortunately, these initial values are estimated in a closed-loop manner so that initial errors are comparatively small. In reality, any battery management system requires such an estimator to obtain the real-time SOC and measure the battery temperature. The results can be used as the initial values.

3. Formulation under MPC Framework

Given a sequence of future inputs, the length of which is denoted as the prediction horizon P , the future system states can be predicted based on the dynamic system model. The future system behavior under the sequence of inputs can then be evaluated based on a performance index. At each control time k , the basic idea of MPC is to find an optimal sequence of inputs $[\hat{u}(k), \hat{u}(k+1), \dots, \hat{u}(k+p-1)]$, which optimizes the performance index, and then apply the first element of the input sequence $\hat{u}(k)$ to the system as the current control variable. In the charging control problem, MPC optimizes a sequence of future charging current $[I(k), I(k+1), \dots, I(k+p-1)]$, which has the best performance index based on the predicted battery states $[\overline{SOC}(k), \overline{SOC}(k+p-1), \dots, \overline{SOC}(k+p-1)]$ and $[\hat{T}(k), \hat{T}(k+1), \dots, \hat{T}(k+p-1)]$.

3.1. Performance Indexes

The performance index reflects control objectives. The first objective is to minimize charging duration. However, in a limited prediction horizon, it is impossible to directly predict the whole charging process, so we turn to the tracking of a user-defined SOC trajectory instead. A fast-rising SOC trajectory requires a fast charging speed while a flat one requires a slow speed.

The expected SOC trajectory can directly copy from any real charging trajectory controlled by any charging scheme. In addition, the expected trajectory can be set as a real trajectory with revisions based on some special considerations. To track the expected trajectory, the performance index J_1 is expressed by:

$$J_1 = \left| \overline{SOC}(k+p-1) - SOC^*(k+p-1) \right| \quad (21)$$

where SOC^* is the user-defined, expected SOC trajectory. Because SOC is required to achieve the expected point in the final step, evaluation of the SOC tracking performance is based on the final prediction state only. How this is achieved is not very important from the point of view of the charging process.

The second objective of charging control is to minimize the increase in temperature, which partially reflects energy efficiency and system safety. Assuming that the expected trajectory is copied from a CCCV charging process, without consideration of temperature, the best charging current sequence is the same sequence applied in the CCCV process. However, if we take increase in temperature into consideration to evaluate charging performance, as expressed in J_2 , the MPC will try to track the expected SOC curve using a process in which increase in temperature is minimized:

$$J_2 = \max \{ \hat{T}(k+j) - T(k-1) \mid j = 0, 1, \dots, p-1 \} \quad (22)$$

3.2. Constraints

Because a charging sequence may cause damage to the battery or lead to dangerous events, it is necessary to design some constraints to protect the battery during the charging process. The first constraint is that the SOC must not exceed 100%, to avoid over-charge damage. In practice, to reserve some tolerance, 98% SOC is treated as the full state. The second constraint is that temperature must be kept in a user-defined range to avoid overheating caused by overcharge or by a large charging current which exceeds the instant charging acceptance level. The two constraints are expressed by:

$$C_1 : \overline{SOC}(k+j) \leq 98\% \quad (23)$$

and:

$$C_2 : \hat{T}(k+j) \leq \bar{T}(k+j) \quad (24)$$

where $j = 0, 1, \dots, p-1$ and \bar{T} is the user-defined safe range, which is designed either as a constant, indicating the highest temperature during the whole charging process, or as a time indexed function specifying the temperature limitation along with charging duration.

4. Optimization Using Genetic Algorithm

The MPC charging control problem is formulized to minimize the performance indexes J_1 and J_2 , subject to the constraints C_1 and C_2 . Essentially, the control problem is transformed into a constrained multi-objective optimization problem.

Generally, a multi-objective optimization problem can either be solved by multi-objective optimizers directly, or by transforming it into a single-objective problem which can be solved by single-objective optimizers [34]. In this paper, we apply the latter method because SOC tracking is more important than temperature management in the charging process, so the two objectives can be merged into one index by summing with different weights. To solve the optimization problem, a genetic algorithm (GA) is applied because of its strong global searching ability without the requirement of derivative information of objective functions. Since the two constraints should be strictly satisfied in the charging process to assure the safety and health of the battery, solutions that fail to satisfy anyone of the constraints will be assigned the worst fitness.

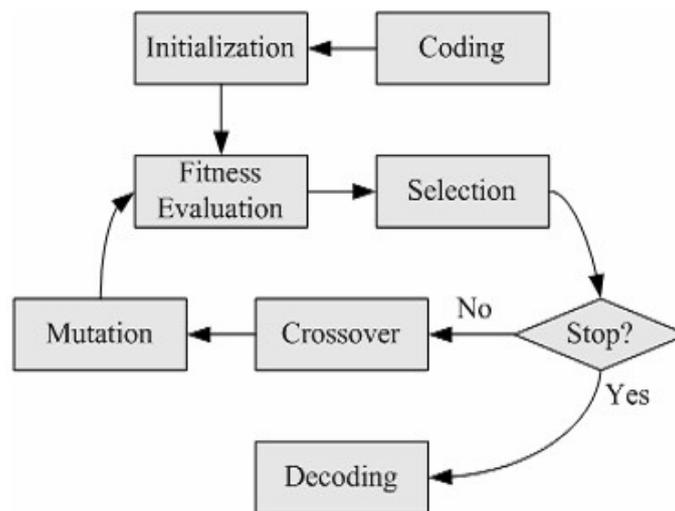
From what discussed above, the fitness function to be minimized in GA is expressed by:

$$F = \begin{cases} \omega_1 J_1 + \omega_2 J_2 & (C_1 \text{ and } C_2 \text{ are held}) \\ +\infty & (\text{otherwise}) \end{cases} \quad (25)$$

where ω_1 and ω_2 are the weights of J_1 and J_2 .

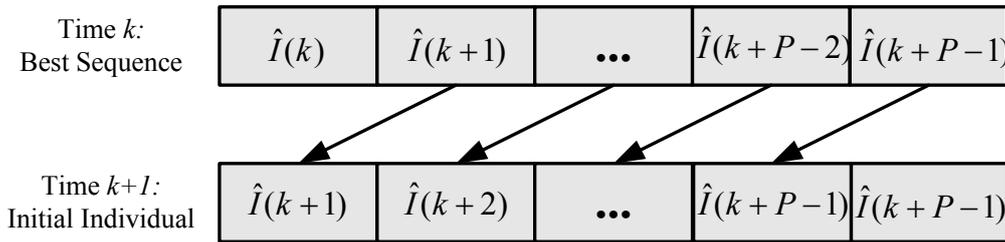
The minimization problem is solved by a standard GA, the scheme of which is illustrated in Figure 5 and briefly described in the following steps.

Figure 5. Scheme of the standard GA.



1. Coding. The standard GA generally codes a candidate solution as a string of characters which are usually binary digits, referred to as a chromosome. The candidate solution is termed an individual. Accordingly, the set made up of a number of individuals is termed a population. In this paper, we apply a real-value coding method, which codes a candidate solution as a set of floating decision variables. The real-value coding method is proven to have superior performance to the binary-coded method in control optimization problems [35].
2. Initialization. The standard GA starts with an initial population. Usually, individuals in the initial population are produced randomly. In MPC, the initialization process is executed at each control time to start the GA. Since the best control sequence optimized at time k contains good candidates from $k+1$ to $k+p-1$, one initial individual is specially designed by shifting it one time step and filling the last charging current with the same value as $\hat{I}(k+p-1)$, as shown in Figure 6. This individual introduces historical best charging sequence into the current optimization process, thus it is helpful to improve optimization performance to be at least very similar with the previous optimized performance.

Figure 6. Initialization of one special individual by introducing the best control sequence optimized in previous step into the present step.



3. Fitness evaluation. We evaluate the fitness of each individual in each generation according to the Equation (25). The smaller the fitness, the better the individual. However, to facilitate the following selection step, the raw fitness is usually scaled to assign suitable selection pressure to each individual. In this work, the scale function is expressed as follows:

$$F_{scale} = \frac{1}{1 + F^r} \tag{26}$$

where r is power of raw fitness. A large r will quickly increase the selection pressure to a worse individual, accelerate the convergence speed, and increase the risk of premature especially for multi-peak landscape, and *vice versa*.

4. Selection. Individuals are selected from the previous generation to the current generation based on the scaled fitness F_{scale} , following the survival of the fittest rule. Many selection methods have been developed to avoid genetic drift and premature phenomena. In this work, the roulette wheel selection method is adopted [36]. The elitism strategy is also applied in selection to assure that the best solution will never be lost.

5. Crossover. In the crossover step, the standard GA exchanges information between two parent individuals and produces two child individuals. In this work, the arithmetical crossover method is used. Given two parents x_1 and x_2 , the children y_1 and y_2 are produced by linear combinations of parents with a random coefficient λ :

$$y_1 = \lambda x_1 + (1 - \lambda) x_2 \tag{27}$$

$$y_2 = \lambda x_2 + (1 - \lambda) x_1 \tag{28}$$

6. Mutation. After the crossover step, a subset of individuals is selected with a mutation probability of p_m . To explore the search space, we use Gaussian mutation, which adds a random value from a Gaussian distribution with variance σ to each item of the selected individual.

7. Termination. Many terminating conditions have been proposed to stop the iteration process. For example, when the distances among individuals are smaller than a predetermined value, an individual satisfies a minimum criterion, or the maximum number of generations is reached. The last method is applied here.

5. Performance Demonstration

5.1. Settings

In order to evaluate the proposed MPC based charging control strategy, simulations are conducted based on a well established “7 Ah Soft Lithium-Ion battery” provided in Advisor [21]. RC model parameters and OCV all are time-variant variables depending on SOC and temperature. The curves of OCV and R_0 are shown in Figure 7 as examples. Interpolating method is applied to create the continuous values space. The constant parameters in this simulation environment are given in Table 1.

Figure 7. Battery time-variant properties. Taking OCV and R_0 as examples. Data source: Advisor.

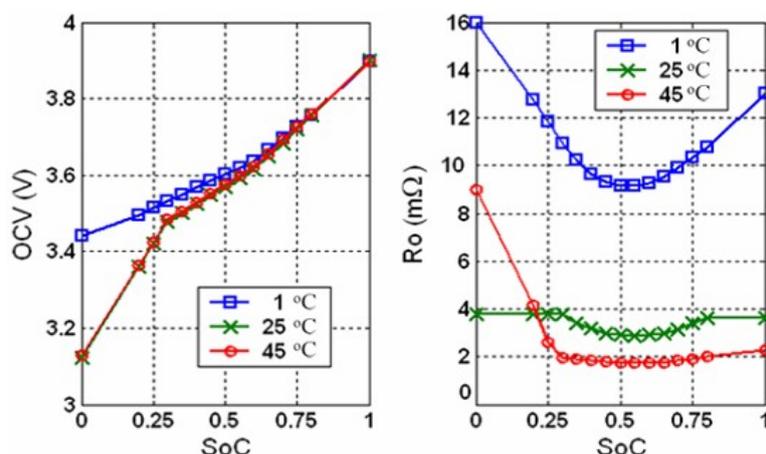


Table 1. Parameters setting in the simulation and experiment.

	Symbol	Description	Value	Unit
Battery (simulation)	C	Battery nominal capacity	7	Ah
	m_{bat}	Battery mass	0.37824	kg
	C_{bat}	Battery heat capacity	795	J/kgK
	R_{eff}	Effective thermal resistance	7.8146	K/W
	T_{amb}	Ambient temperature	20	°C
	\dot{m}_{air}	Airflow rate	5.8333	g/s
	C_{air}	Air heat capacity	1009	J/kgK
MPC	T_s	Control period	30	s
	p	Prediction horizon	5	--
	ω_1	Weight of SOC tracking J_1	100	--
	ω_2	Weight of temperature rising J_2	1	--
GA	$MaxGen$	Maximum generation number	30	--
	$Popsiz$	Population size	50	--
	r	Power of raw fitness in scaling	2	--
	R_λ	The range of crossover coefficient	[0.1, 0.9]	--
	p_m	Mutation probability	0.2	--
	σ	Variance of Gaussian mutation	1	--

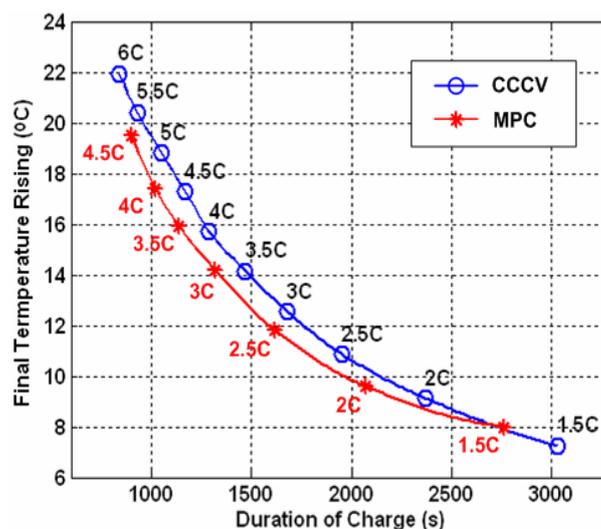
Remark: The typical value of J_1 is around 0.08 while that of J_2 is around 2. Therefore, the real weight ratio of J_1 to J_2 is around 4:1.

In this work, we select the control period T_s to be 30 seconds using a trial and error method according to experimental results. A shorter period requires the charger to change the charging current more frequently. It will increase working load to the charger and lead to more energy loss caused by switch circuits. The longer period will decrease the frequency to tune charging current in the charging process so that the performance of MPC will be limited. In addition, the long control period implies that the prediction based on the model is over a long time which essentially requires a more accurate battery model. Finally, the 30 seconds control period is long enough to allow GA to finish the optimization process.

5.2. Evaluation Method

Fast charging speed and low heat generation are both objectives in charging control. However, the two goals conflict with each other. Fast charging essentially requires large current and hence leads to high heat generation, and vice versa. From the viewpoint of multi-objective optimization, the conflicting objectives are usually evaluated by Pareto curve. For charging control, the x -axis is set as the charging duration and the y -axis as the final temperature increase $T(\text{end}) - T(0)$. As shown in Figure 8, the curve with the circle marks represents the Pareto front of the CCCV family, where the CC period applies 1.5 C to 6 C current for fast charging. The CCCV Pareto front splits the objective space into two sections. Any charging result located in the upper right section is worse than the CCCV family for both objectives, while any result in the bottom left section is better than CCCV for both objectives. Therefore, we evaluate the performance of the charging controller according to the location of results.

Figure 8. Pareto fronts of CCCV and MPC charging methods in simulation. The expected trajectories of MPC are modified from CCCV by multiplying 1.05.



5.3. Simulation Results

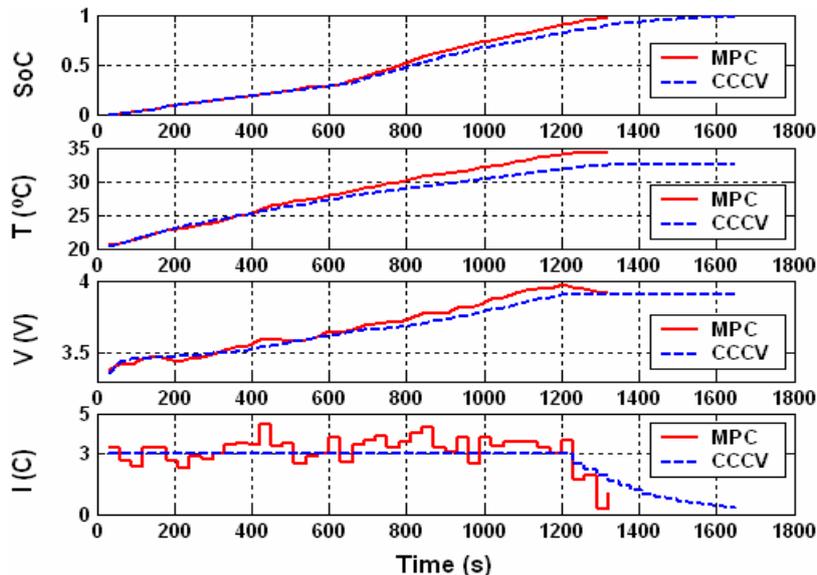
To facilitate comparison with the conventional CCCV family, we set the expected SOC trajectory by multiplying by 1.05. The new trajectory is intended to accelerate the charging speed. Since the working temperature of Li-ion batteries, especially for vehicular batteries, is from 20 °C to 40 °C, the

ambient temperature is set as 20 °C and the safe range of temperature is set below 40 °C. The search space of the charging current is fixed from 0 C to 6 C.

The MPC is applied to track the modified SOC trajectories from 1.5 C to 4.5 C. A faster charging speed cannot keep the rising temperature within a safe range. The Pareto front of the MPC is illustrated in Figure 8 by the curve marked with stars. The results for the MPC clearly dominate those of the CCCV family. Only the result of the trajectory revised from 1.5 C has a similar performance to CCCV. The reason is that a 1.5 C current is comparatively so small that it limits the applicable current sequence to a small range. If the fixed search range can be accordingly reduced, e.g., from 0 C to 3 C, the result will be improved.

As an example, the charging processes of CCCV and MPC are compared in the case of 3 C, shown in Figure 9. At the beginning of charge, the internal resistance is large when the SOC is very low. In this process, the rising temperature dominates the fitness function. The optimized charging profile applies a smaller current than CCCV. However, the increase in the SOC tracking error gradually requires a higher current to speed up. Meanwhile, internal resistance is reduced significantly along with the increase in SOC. Therefore, in the middle period, the charging current for MPC is larger than for CCCV. In the final stage, the current in the CV period is decreased to prevent over-voltage of the terminal voltage. Since the trajectory of the MPC is modified from the CCCV, the same trend is retained in MPC, keeping the terminal voltage under 4 V.

Figure 9. Curves during CCCV and MPC charging processes using modified 3 C profile.



5.4. Experimental Results

Since the maximum charging current of the programmable electric charger IT6154 is 9 A, in the experiment we use a single Li-ion cell with 2.3 Ah to demonstrate the fast charge performance of MPC. To protect the cells, we limit the charging current to below 3 C, *i.e.*, 6.9 A. The ambient temperature is 25 °C. The expected SOC trajectories of MPC are 1.10 times those of the CCCV results.

Figure 10 shows the Pareto-fronts of the CCCV and MPC charging results in experiments. All the results for MPC are superior to those for CCCV. Figure 11 illustrates the details of the 2 C charging

processes. Unlike the simulation model, where the temperature is high in the lower SOC range of the specified battery type, the real LiFePO₄ cell has good charging acceptance at low SOC. Therefore, a high charging current is utilized to increase the charging speed. Before reaching around 80% SOC, another high charging period occurs. This is because MPC predicts the high temperature increase beyond 80%, so it applies a high charging current at this stage and switches to a low charging current when SOC is close to 1.

Figure 10. Pareto fronts of CCCV and MPC charging methods in experiments. The expected trajectories of MPC are modified from CCCV by multiplying by 1.10.

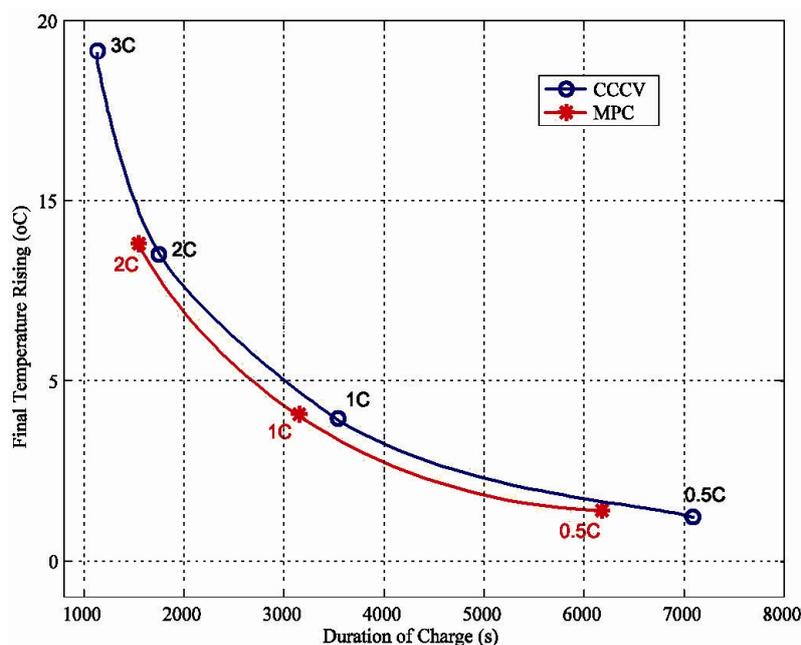
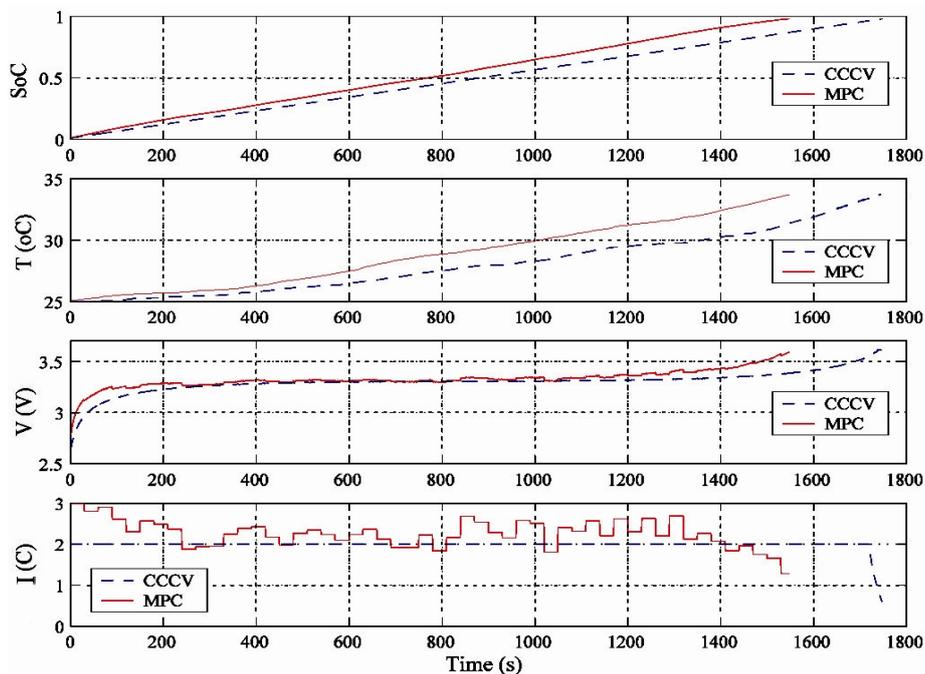


Figure 11. Experimental curves during the CCCV and MPC charging processes using a modified 2 C profile.



6. Conclusions

One of the most important obstacles for the mass commercialization of electric vehicles is the comparatively too slow process of refueling their energy. However, simply speeding up the charging process will increase the temperature rise, resulting in a low energy efficiency of charging process and maybe even damage to the batteries being charged. Therefore, the battery charge problem actually is a multi-objective issue which should be evaluated under the Pareto-optimal concept.

On the other hand, the charging stations of electric vehicles are usually designed as advanced equipment rather than a single chip based charger for portable devices. Thus, computers can be applied in charging stations to control the charging process. Although the proposed framework contains advanced algorithms such as NN and GA, it is practicable to be implemented on these computers, as shown in our experiment.

To accelerate charging speed and reduce the temperature increase, we introduce the MPC framework to the charging control process. Given a future charging sequence, the RC model and proposed modified ESC model are applied to predict the SOC in simulation and experiment, respectively. Meanwhile, besides the single-node lumped-parameter thermal model used in simulation, we also establish a neural network to model the thermal behavior of the applied battery in the experiment. A standard genetic algorithm is applied to optimize the charging current under the multi-objectives and constraints. The simulation and experimental results strongly demonstrate the availability and efficacy of MPC, with the conclusion that the Pareto front of the MPC dominates that of CCCV.

Acknowledgments

This work was supported by the Shenzhen Key Laboratory of Electric Vehicle Powertrain Platform and Safety Technology.

References

1. Cope, R.C.; Podrazhansky, Y. The Art of Battery Charging. In *Proceedings of the 14th Annual Battery Conference on Applications and Advances*, Long Beach, CA, USA, 12–15 January 1999; pp. 233–235.
2. Zhang, S.S.; Xu, K.; Jow, T.R. Study of the charging process of a LiCoO₂-based Li-ion Battery. *J. Power Sources* **2006**, *160*, 1349–1354.
3. Hsieh, G.C.; Chen, L.R.; Huang, K.S. Fuzzy-controlled Li-ion battery charge system with active state-of-charge controller. *IEEE Trans. Ind. Electron.* **2001**, *48*, 585–593.
4. Wong, Y.S.; Hurley, W.G.; Wolfle, W.H. Charge regimes for valve-regulated lead-acid batteries: Performance overview inclusive of temperature compensation. *J. Power Sources* **2008**, *183*, 783–791.
5. Notten, P.H.L.; Veld, J.H.G.; Beek, J.R.G. Boostcharging Li-ion batteries: A challenging new charging concept. *J. Power Sources* **2005**, *145*, 89–94.
6. Lin, C.H.; Hsieh, C.Y.; Chen, K.H. A Li-ion battery charger with smooth control circuit and built-in resistance compensator for achieving stable and fast charging. *IEEE Trans. Circuits Syst. I.* **2010**, *57*, 506–517.

7. Hussein, A.A.H.; Batarseh, I. A review of charging algorithms for nickel and lithium battery chargers. *IEEE Trans. Veh. Technol.* **2011**, *60*, 830–838.
8. Park, S.Y.; Miwa, H.; Clark, B.T.; Ditzler, D.S.K.; Malone, G.; D'souza, N.S.; Lai, J.S. A Universal Battery Charging Algorithm for Ni-Cd, Ni-MH, SLA, and Li-Ion for Wide Range Voltage in Portable Applications. In *Proceedings of IEEE Conference on Power Electronics Specialists*, Rhodes, Greece, 15–19 June 2008; pp. 4689–4694.
9. IKeya, T.; Sawada, N.; Murakami, J.; Kobayashi, K.; Hattori, M.; Murotani, N.; Ujiie, S.; Kajiyama, K.; Nasu, H.; Narisoko, H.; *et al.* Multi-step constant-current charging method for an electric vehicle nickel/metal hydride battery with high-energy efficiency and long cycle life. *J. Power Sources* **2002**, *105*, 6–12.
10. Diaz, J.; Martin-Ramos, J.A.; Pernia, A.M.; Nuno, F.; Linera, F.F. Intelligent and universal fast charger for Ni-Cd and Ni-MH batteries in portable applications. *IEEE Trans. Ind. Electron.* **2004**, *51*, 857–863.
11. Cheng, M.W.; Wang, S.M.; Lee, Y.S.; Shiao, S.H. Fuzzy Controlled Fast Charging System for Lithium-Ion Batteries. In *Proceedings of International Conference on Power Electronics and Drive Systems*, Taipei, Taiwan, 2–5 November 2009; pp. 1498–1503.
12. Wang, J. Charging Strategy Studies for PHEV Batteries Based on Power Loss Model. In *Proceedings of SAE World Congress & Exhibition*, Detroit, MI, USA, April 2010; SAE Paper No: 2010-01-1238.
13. Ullah, Z.; Burford, B.; Dillip, S. Fast intelligent battery charging: Neural-fuzzy approach. *IEEE Aerosp. Electron. Syst. Mag.* **1996**, *11*, 26–34.
14. Khosla, A.; Kumar, S.; Aggarwal, K.K. Fuzzy Controller for Rapid Nickel-Cadmium Batteries Charger through Adaptive Neuro-Fuzzy Inference System (ANFIS) Architecture. In *Proceedings of International Conference of the North American Fuzzy Information Processing Society*, Chicago, USA, 24–26 July 2003; pp. 540–544.
15. Aliev, R.A.; Aliev, R.R.; Guirimov, B.; Uyar, K. Dynamic data mining technique for rules extraction in a process of battery charging. *Appl. Soft Comput.* **2008**, *8*, 1252–1258.
16. Liu, Y.H.; Teng, J.H.; Lin, Y.C. Search for an optimal rapid charging pattern for lithium-ion batteries using ant colony system algorithm. *IEEE Trans. Ind. Electron.* **2005**, *52*, 1328–1336.
17. Saberi, H.; Salmasi, F. Genetic Optimization of Charging Current for Lead-Acid Batteries in Hybrid Electric Vehicles. In *Proceedings of International Conference on Electrical Machines and Systems*, Seoul, Korea, 8–11 October 2007; pp. 2028–2032.
18. Chen, L.R.; Hsu, R.C.; Liu, C.S. A design of a grey-predicted Li-ion battery charge system. *IEEE Trans. Ind. Electron.* **2008**, *55*, 3692–3701.
19. Salamch, Z.M.; Casacca, M.A.; Lynch, W.A. A mathematical model for lead-acid batteries. *IEEE Trans. Energy Convers.* **1992**, *7*, 93–98.
20. Bergveld, H.J.; Kruijt, W.S.; Notten, P.H.L. *Battery Management System: Design by Modeling*; Kluwer Academic Publishers: Dordrecht, The Netherlands, 2002.
21. Johnson, V.H. Battery performance models in ADVISOR. *J. Power Sources* **2002**, *110*, 321–329.
22. Plett, G.L. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 2. Modeling and identification. *J. Power Sources* **2004**, *134*, 262–276.

23. Szumanowski, A.; Chang, Y. Battery Management System Based on Battery Nonlinear Dynamics Modeling. *IEEE Trans. Veh. Technol.* **2008**, *57*, 1425–1432.
24. Camacho, E.F.; Bordons, C. *Model Predictive Control*; Springer Verlag: London, UK, 2004.
25. Qin, S.J.; Badgwell, T.A. A survey of industrial model predictive control technology. *Control Eng. Pract.* **2003**, *11*, 733–764.
26. Bemporad, A.; Morari, M. Robust Model Predictive Control: A Survey. In *Robustness in Identification and Control*; Springer Verlag: London, UK, 1999; pp. 207–226.
27. Yan, J.Y.; Ling, Q.; Chen, W. Nonlinear Model Predictive Control Based on Evolutionary Algorithms: Framework, Theory and Application. In *A Giordano and G. Costa, Soft Computing: New Research*; Alessia, J.G., Ginevra, E.C., Eds.; Nova Publishers: New York, NY, USA, 2009.
28. Pesaran, A.A. Battery thermal models for hybrid vehicle simulations. *J. Power Sources* **2002**, *110*, 377–382.
29. Chow, M.; Lukic, S.; Wang, L.; Govindaraj, A. Research report on A123 battery modeling. Available online: <http://www.adac.ncsu.edu/projects/Battery%20Model/Docs/Research%20Repot%20on%20A123%20Battery%20Modeling.ppt> (accessed on 4 August 2011).
30. Plett, G.L. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs-Part 3. State and parameter estimation. *J. Power Sources* **2004**, *134*, 277–292.
31. Plett, G.L. Sigma-point Kalman filtering for battery management systems of LiPB-based HEV Battery packs: Part 1: Introduction and state estimation. *J. Power Sources* **2006**, *161*, 1356–1368.
32. Plett, G.L. Sigma-point Kalman filtering for battery management systems of LiPB-based HEV Battery packs: Part 2: Simultaneous state and parameter estimation. *J. Power Sources* **2006**, *161*, 1369–1384.
33. Yan, J.Y.; Xu, G.Q.; Xie, B. Battery State-of-Charge Estimation Based on H Filter for Hybrid Electric Vehicle. In *Proceedings of International Conference on Control, Automation, Robotics and Vision*, Hanoi, Vietnam, 17–20 December 2008; pp. 464–469.
34. Deb, K. *Multi-Objective Optimization Using Evolutionary Algorithms*; Wiley: Hoboken, NJ, USA, 2001.
35. Jiang, B.; Wang, B.W. Parameter estimation of nonlinear system based on genetic algorithm. *Control Theory Appl.* **2000**, *17*, 150–152.
36. Goldberg, D.E.; Deb, K. A comparative analysis of selection schemes used in genetic algorithms. *Found. Genet. Algorithms* **1991**, *1*, 69–93.