International Journal of Automotive Technology, Vol. 9, No. 6, pp. 759–764 (2008)
DOI 10.1007/s12239–008–0090–x

1229–9138/2008/043–15

ESTIMATION OF BATTERY STATE-OF-CHARGE USING ν-SUPPORT VECTOR REGRESSION ALGORITHM

Q.-S. SHI*, C.-H. ZHANG and N.-X. CUI

School of Control Science and Engineering, Shandong University, Jinan 250061, China

(Received 8 January 2008; Revised 18 June 2008)

ABSTRACT–Accurately estimating the SOC of a battery during the electric vehicle drive cycle is a vital issue that currently remains unresolved. A support vector regression algorithm (SVR), which has good nonlinear approximation ability, a quick convergence rate and global optimal solution, is proposed to estimate the battery SOC. First, the training data and the test data required in the estimation operation are collected using the ADVISOR software, followed by normalization of the data above. Then, cross validation and grid search methodologies are used to determine the parameters in the ν-SVR model. Finally, simulation experiments have been carried out in the LIBSVM simulator. The simulation results show that, compared to the BP neural network algorithm, the ν-Support Vector Regression algorithm performs better in estimating the battery SOC.

KEY WORDS : State-of-charge, Electric vehicles, ν-support vector regression

1. INTRODUCTION

Increasing concerns throughout the world surrounding energy conservation and environmental protection have resulted in the revival of the electric vehicle (Chan and Chau, 2001; Wu *et al*., 2008). As electric vehicles (EVs) gain popularity, battery state of charge (SOC) estimation becomes an increasingly important issue in terms of both extending the lifetime of the battery and displaying the usable charge to the user before recharging. However, the SOC cannot be measured directly, but rather must be estimated based on measurable battery parameters such as voltage and current.

Much research has been performed in recent years to improve SOC determination. Conventionally, estimation methods of the battery SOC can be partitioned into four categories. The first method is based upon empirical formulae (Song and Kim, 1994; Riutort *et al*., 1995; Caumont *et al*., 2000) and mathematical modeling (Salameh *et al*., 1992; Fan and White, 1991; Bergveld *et al*., 1999) using analysis of battery characteristics of the constant current discharge. Parameters of this type are generally obtained only in the steady state, and cannot reflect the dynamic behavior of the battery in EVs. Moreover, the outcome of these calculations are strongly dependent on the type of battery used and is therefore not easily transferred from one type of battery to another. The second method is based on coulometric measurements in which the accumulated error is corrected by the stabilized open-circuit voltage (OCV). In this method, the stabilized OCV refers to the battery terminal voltage measured after the battery is sufficiently rested (Aylor and Johnson, 1992). However, practical EV operation may not permit the vehicle to be stopped for such a long enough period of time for the stabilized OCV to be measured. The third method is based on an impedance measurement (Karden *et al*., 1997; Barsoukov *et al*., 1999). In this method, a small amplitude AC signal is injected into the battery in order to measure its internal impedance. As a result, the impedance obtained does not include the dynamic characteristics of a large discharge current, which is often imposed on the EV battery. Furthermore, the equipment necessary to carry out such an impedance measurement is too expensive and bulky for EVs. Recently, a new method using neural networks has been proposed for estimation of the battery SOC. The utilization of artificial neural networks for SOC determination is presented in (Peng *et al*., 2000; Shen *et al*., 2005). Since artificial neural networks establish a relationship between input/output data of any kind, this method can be used for all battery systems, provided that training data for the net is available. One problem is that the network is a black box and can only be verified using another data set. Another drawback is that the error is strongly dependent on the structure of the neural network, the training data and the training method.

Support Vector Regression, which is used to estimate the battery SOC in this paper, is a learning algorithm motivated by results of statistical learning theory. It has been successfully applied to many highly nonlinear systems, such as travel time prediction (Wu *et al*., 2004), nonlinear noise filtering (Zhang *et al*., 2006), image superresolution (Ni and Nguyen, 2007) and so on. The Support Vector Regression algorithm (SVR) has good qualities such as nonlinear approximation ability, a quick convergence rate and a global

**Corresponding author*. e-mail: zchui@sdu.edu.cn

optimal solution. Simulation results indicate that the SVR performs well at estimating the SOC of batteries in electric vehicle drive cycles.

2. PRELIMINARY OF SUPPORT VECTOR REGRESSION ALGORITHM

Support Vector machines are originally developed for pattern recognition, which represent decision boundaries in terms of a typically small subset of all training examples, called the Support Vectors (SVs). In order for this property to carry over to the case of SV Regression, Vapnik devised the so-called ε-insensitive loss function and proposed the ε-SVR algorithm (Vapnik, 1995). It seeks to estimate functions

$$f(\mathbf{x}) = \sum_{i=1}^{l} (-\alpha_i + \alpha_i^*) k(\mathbf{x}_i, \mathbf{x}) + b, \tag{1}$$

based on independent identically distributed data

$$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l) \in X \times R. \tag{2}$$

Here, X is the space in which the input vector exists, $X \in R^n$, y_i is the output value, b is the bias, and $k(\mathbf{x}_i, \mathbf{x})$ is the kernel function. In this paper, the current, voltage and the required power of the batteries are selected as inputs, and the SOC is the output. The goal of the learning process is to find a function f with a small risk

$$R[f] = \int_X l(f, \mathbf{x}, y) dP(\mathbf{x}, y), \tag{3}$$

where P is the probability measure which is assumed to be responsible for the generation of the data in Equation (2), and l is a loss function. Since we do not know P, Equation (3) cannot be minimized directly. Instead, we are given the sample Equation (2), and we try to obtain a small risk by minimizing the regularized risk functional

$$\frac{1}{2}\|\mathbf{w}\|^2 + C \cdot R_{emp}^{\varepsilon}[f] \tag{4}$$

Here, $\|\mathbf{w}\|^2$ is a term which characterizes the model complexity, $R_{emp}^{\varepsilon}[f] = \frac{1}{l}\sum_{i=1}^{l} |y_i - f(\mathbf{x}_i)|_{\varepsilon}$ measures the ε-insensitive training error, and C is a constant determining the trade-off.

The minimization of Equation (4) is equivalent to the following constrained optimization problem:

$$\min \tau(\mathbf{w}, \xi^{(*)}) = \frac{1}{2}\|\mathbf{w}\|^2 + C \cdot \frac{1}{l}\sum_{i=1}^{l} (\xi_i + \xi_i^*) \tag{5}$$

subject to $((\mathbf{w} \cdot \mathbf{x}_i) + b) - y_i \le \varepsilon + \xi_i$, (6)

$$y_i - ((\mathbf{w} \cdot \mathbf{x}_i) + b) \le \varepsilon + \xi_i^*, \tag{7}$$

$$\xi_i^{(*)} \ge 0 \tag{8}$$

Here and below, it is understood that i=1, …, l, and that bold face Greek letters denote l-dimensional vectors of the corresponding variables; ξ is a slack variable, and (*) is shorthand implying both the variables with and without asterisks.

By using Lagrange multiplier techniques, one can show that this leads to the following dual optimization problem:

$$\max\ W(\boldsymbol{\alpha}, \boldsymbol{\alpha}^*) = -\varepsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) - \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) y_i - \frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)(\mathbf{x}_i \cdot \mathbf{x}_j) \tag{9}$$

subject to $\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0$, (10)

$$\alpha_i^{(*)} \in [0, \frac{C}{l}], i = 1, \ldots, l \tag{11}$$

where, α is the dual variable.

It should be noted that the parameter ε can be useful if the desired accuracy of the approximation can be specified beforehand. In some cases, however, we just want the estimate to be as accurate as possible, without having to commit ourselves to a specific level of accuracy a priori.

Scholkopf *et al*. (2000) modified the ε-SVR algorithm, and proposed the ν-SVR algorithm, which automatically minimized ε. The size of ε is trade off against model complexity and slack variables via a constant $\nu \ge 0$. The primal form is

$$\min\ \tau(\mathbf{w}, \xi^{(*)}, \varepsilon) = \frac{1}{2}\|\mathbf{w}\|^2 + C \cdot (\nu\varepsilon + \frac{1}{l}\sum_{i=1}^{l}(\xi_i + \xi_i^*)) \tag{12}$$

subject to $((\mathbf{w} \cdot \mathbf{x}_i) + b) - y_i \le \varepsilon + \xi_i$, (13)

$$y_i - ((\mathbf{w} \cdot \mathbf{x}_i) + b) \le \varepsilon + \xi_i^*, \tag{14}$$

$$\xi_i^{(*)} \ge 0, \varepsilon \ge 0 \tag{15}$$

According to the Wolfe dual, the dual form of the ν-SVR optimization problem can be described as follows: For $\nu \ge 0$, $C < 0$, maximize

$$W(\boldsymbol{\alpha}, \boldsymbol{\alpha}^*) = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i) y_i - \frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) k(\mathbf{x}_i, \mathbf{x}_j) \tag{16}$$

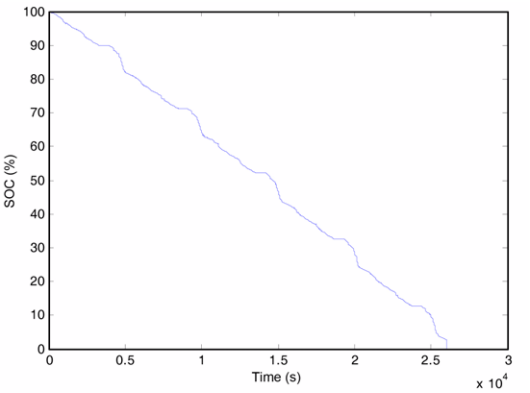
subject to $\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0$ (17)

$$\alpha_i^{(*)} \in [0, \frac{C}{l}] \tag{18}$$

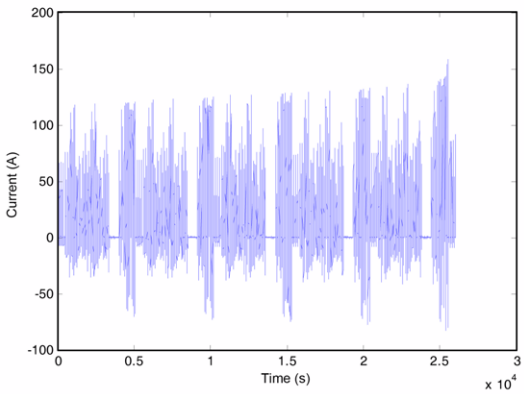
$$\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) \le C \cdot \nu. \tag{19}$$

The regression estimate then takes the form

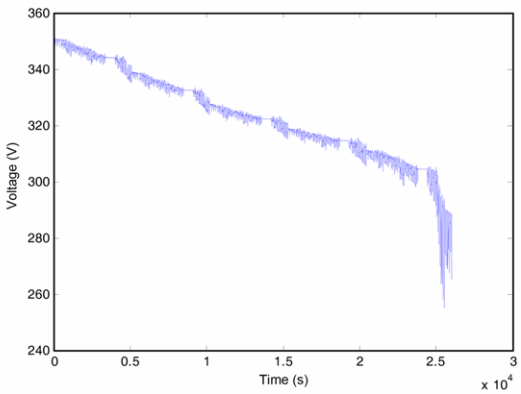
$$f(\mathbf{x}) = \sum_{i=1}^{l}(-\alpha_i + \alpha_i^*) k(\mathbf{x}_i, \mathbf{x}) + b$$



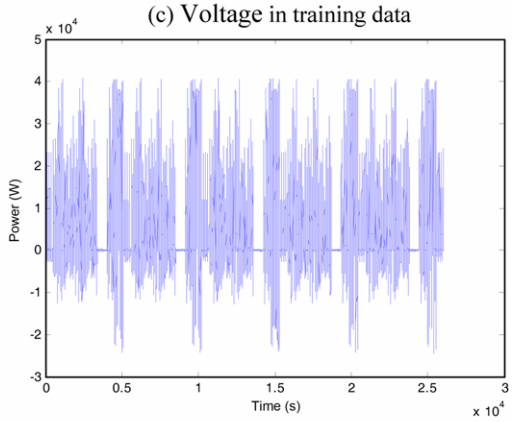
(a) SOC profile in training data



(b) Current in training data



(c) Voltage in training data



(d) Required power in training data

Figure 1.7 Training data vector.

where b can be computed by taking into account that Equation (13) and (14) become equalities with $\xi_i^{(*)} = 0$ for points with $\alpha_i^{(*)} \in [0, \frac{C}{l}]$, respectively, due to KKTconditions.

In this paper, the ν-SVR algorithm is adopted to estimate the battery SOC in electric vehicles. An SVM should first be designed to incorporate thousands of training data points and reduce them to a set of SVs. Here, we choose the battery current, voltage and required power as the input vector, and the battery SOC as the output vector. Then, we use the right training data and proper kernel functions to correctly optimize the SVM. During the estimation process, we use LIBSVM software to determine the SVs with the source code modified so that its input parameters can be automatically varied and the resulting SVM can be tested for accuracy. The detailed estimation processes are described below in section 3.

3. ELECTRIC VEHICLE BATTERY SOC ESTIMATION PROCEDURE BASED ON ν-SVR ALGORITHM

During the process of battery SOC estimation in electric vehicles, LIBSVM software is used (Chang and Lin, 2001). Steps used in training a SVR model for SOC estimation are presented followed by the test procedures and results. Training steps include: training data selection and preprocessing and finding the optimal SVR parameters. Similarly, test steps also consist of selection and scaling operation of test data.

3.1. Training Data Selection

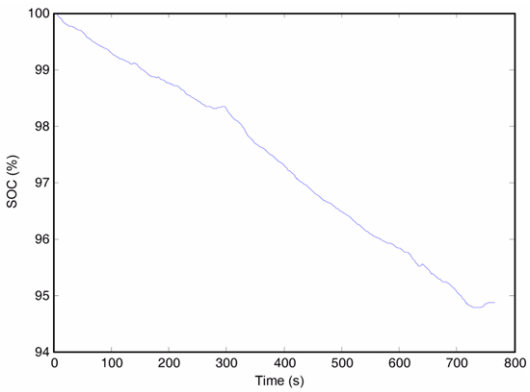
Using the electric vehicle systems analysis tool ADVISOR (Markel *et al.*, 2002), we collect the training data required in the estimation process, which consists of the current, voltage, required power and SOC, as shown in Figure 1(a) to Figure 1(d). The data are obtained from a hybrid drive cycle, which covers the main operation modes of the vehicle. This cycle is comprised of a lot of typical cycles, such as JA-1015, UDDS, FTP and US06.

3.2. Training Data Scaling

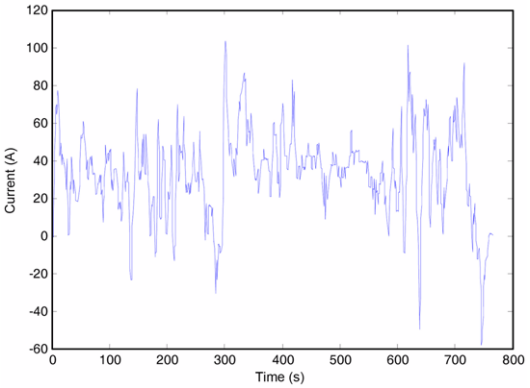
Data scaling before applying SVR is very important. The main advantage of this is to avoid attributes in greater numerical ranges dominating those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during calculation. Since the parameters may have different ranges, computation could be difficult, and very few SVR training runs may converge.

Table 1. Example of training data.

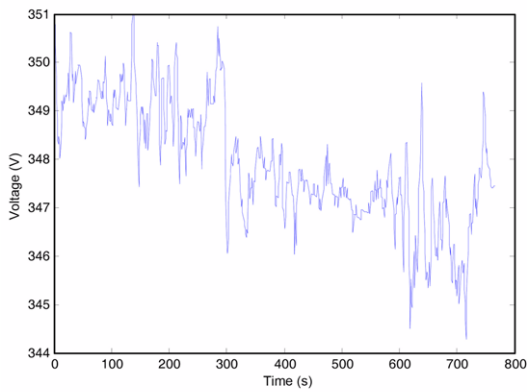
Element	Current	Voltage	Required power	SOC
Unscaled	9.57	335.5	3184.5	76.75
Scaled	–0.34	–0.34	0.81	0.84



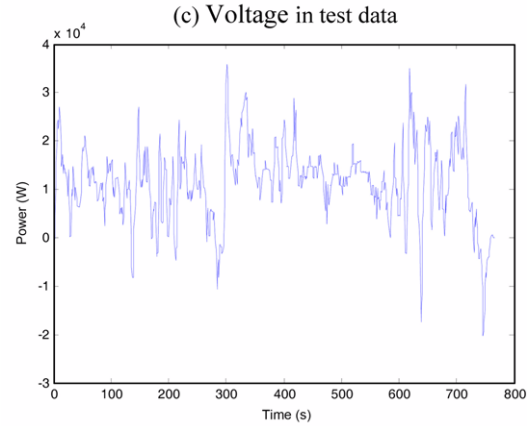
(a) SOC profile in test data



(b) Current in test data



(c) Voltage in test data



(d) Required power in test data

Figure 2. Test data in the NYCC.

By scaling operation, all the input elements are in the range of [–1, 1], using

$$\mathbf{x}_i = \frac{2(\mathbf{x}_i^* - \mathbf{x}_{\min})}{\mathbf{x}_{\max} - \mathbf{x}_{\min}} - 1, \tag{20}$$

where $\mathbf{x}_i$ is the i-th scaled factor, $\mathbf{x}_i^*$ is the i-th unscaled factor, and $\mathbf{x}_{\max}$ and $\mathbf{x}_{\min}$ are the maximum and minimum values, respectively, of the unscaled factors. A representative training vector is shown in Table 1.

3.3. Finding Optimal SVR Parameters

During the estimation process, the radial basis function (RBF) is chosen as the kernel function, the form of which is as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2), \gamma > 0 . \tag{21}$$

There are three parameters in ν-SVR when using RBF kernels: C, ν and γ. It is not known beforehand which C, ν and γ are the best for one problem; consequently a parameter search must be done. The goal is to identify optimal (C, ν, γ) so that the regression model can accurately predict unknown data.

Here, cross-validation and grid-search methods are adopted to identify optimal values of (C, ν, γ). In cross-validation, we first divide the training set into m subsets of equal size. Sequentially one subset is tested using the regression model trained on the remaining m-1 subsets. Thus, each instance of the whole training set is predicted once so the cross-validation accuracy is the percentage of data which are correctly estimated.

Before using the grid-search method, we should preset the variable range (C, ν, γ). Generally, the values of C, ν and γ are set as exponentially growing sequences (for example, $C=2^1, 2^2, \cdots, 2^{10}$; $\nu=2^4, \cdots, 2^0$; $\gamma=2^{10}, 2^9, \cdots, 2^0$). The value of C should not be too large, while g should not be too small. Finally, a value of 16 is used for C, 0.5 for ν and 0.5 for γ in the optimal ν-SVR model.

3.4. Selection and Preprocessing of Test Data

According to the criteria that test data should be different from the training data, test data chosen for this work are obtained from running the New York City Cycle (NYCC), the Central Business District test cycle (CBD) and the Manhattan drive cycle. Here, in order to make the paper more readable, only the test data in the NYCC are shown, which are depicted in Figure 2(a) to Figure 2(d). Scaling of test data is done in the same way as training data.

4. SIMULATION RESULTS AND ANALYSIS

To evaluate the performance of the SVR algorithm in battery SOC estimation, the neural network method (NN) is simultaneously adopted in the estimation process as a baseline. The NN takes a three-layer structure: three nodes in the input layer, five in the hidden layer and one in the output layer. The comparison simulation results are pre-

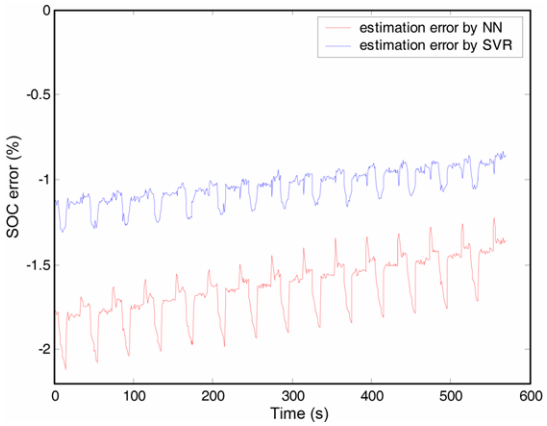


Figure 3. Estimation errors in CBD.

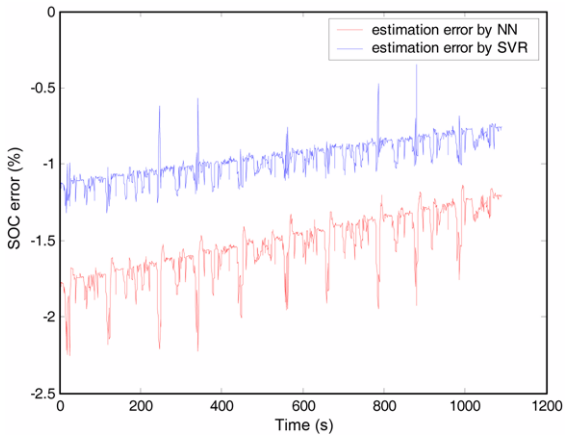


Figure 4. Estimation errors in Manhattan.

sented in Figure 9, Figure 10 and Figure 11. The estimation error using the above two methods is also given, as shown in Figure 10. To allow for comparison, the average percentage error (APE) is also introduced (Shen *et al*., 2005). It is defined as:

$$APE = \frac{1}{N}\sum_{j=1}^{N}\frac{\left|SOC_{ra}(j) - SOC_{re}(j)\right|}{\left|SOC_{ra}(j)\right|} \times 100\%, \qquad (22)$$

where N is the number of the data sets for each test, and SOC_{ra} and SOC_{re} refer to the estimated SOC from the trained model and the actual SOC from simulation experiments.

From Figure 3 to Figure 5, we can see that the estimation errors of SOC using the Support Vector Regression algorithm are smaller than those using a neural network in the CBD cycle, the Manhattan cycle and the NYCC cycle. To express the estimation performance more concisely, the APE values using a NN and ν-SVR are listed in Table 2. It can be seen that the APE values using the NN method are 1.64%, 1.54% and 1.64%, while the APE values using the ν-SVR method are 1.04%, 0.94% and 1.02%, respectively.

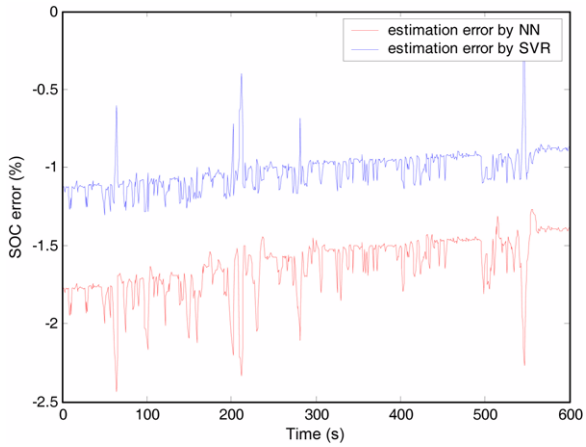


Figure 5. Estimation errors in NYCC.

Table 2. APE value in different drive cycles.

Drive cycle	APE value	
	NN	ν-SVR
CBD	1.64%	1.04%
Manhattan	1.54%	0.97%
NYCC	1.64%	1.02%

5. CONCLUSION

This paper presents a method for estimating the battery SOC of Electric Vehicle using the ν-SVR algorithm. We collect the sample data, including training data and test data, using the electric vehicle simulation software ADVISOR. In sample data, the battery current, voltage and required power are used as input vectors and the SOC as the output vector. Cross-validation and grid-search methodologies are used to find the optimal ν-SVR parameters. Comparative simulations are performed between the NN method and the ν-SVR method. Our results show that the ν-SVR algorithm offers a smaller APE than the NN, that is, the ν-SVR is more accurate than the NN in estimating the battery SOC.

ACKNOWLEDGEMENT–This work was supported and funded by the National Natural Science Foundation of China under Grant 60874016 and 50477042, and the Outstanding young scientists incentive funding of Shandong Province (2007BS01012). The authors would like to thank Prof. Chih-Jen Lin for the discussion about usage of LIBSVM software.

REFERENCES

Aylor, J. H., Thieme, A. and Johnson, B. W. (1992). A battery state-of-charge indicator for electric wheelchairs. *IEEE Trans. Industrial Electronics* **39**, **5**, 398–409.

Barsoukov, E., Kim, J. H., Yoon, C. O. and Lee, H. (1999). Universal battery parameterization to yield a nonlinear equivalent circuit valid for battery simulation at arbitrary

load. *J. Power Sources* **83**, **1–2**, 61–70.

Bergveld, H. J., Kruijt, W. S. and Notten, P. H. L. (1999). Electronic-network modeling of rechargeable NiCd cells and its application to the design of battery management system. *J. Power Sources* **77**, **2**, 143–158.

Caumont, O., Moigne, P. L., Rombaut, C., Muneret, X. and Lenain, P. (2000). Energy gauge for lead-acid batteries in electric vehicles. *IEEE Trans. Energy Conversion* **15**, **3**, 354–360.

Chan, C. C. and Chau, K. T. (2001). *Modern Electric Vehicle Technology*, **1**. Oxford University Press. Oxford. 1–14.

Chang, C. L. and Lin, C. (2001). LIBSVM: A Library for Support Vector Machines (Version2.3) [EB/OL] http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.Pdf

Fan, D. and White, R. E. (1991). Mathematical modeling of a nickel-cadmium battery-effects of intercalation and oxygen reactions. *J. Electrochemical Society* **138**, **10**, 2952–2960.

Karden, E., Mauracher, P. and Schoepe, F. (1997). Electrochemical modeling of lead-acid batteries under operating conditions of electric vehicles. *J. Power Sources* **64**, **1**, 175–180.

Markel, T., Brooker, A., Hendricks, T., Johnson, V., Kelly, K., Kramer, B., Okeefe, M., Sprik, S. and Winke, K. (2002). ADVISOR: A systems analysis tool for advanced vehicle modeling. *J. Power Sources* **110**, **2**, 255–266.

Ni, K. S. and Nguyen, T. Q. (2007). Image superresolution using support vector regression. *IEEE Trans. Image Processing* **16**, **6**, 1596–1610.

Peng, J. C., Chen, Y. B., Eberhart, R. and Lee, H. H. (2000). Adaptive battery state of charge estimation using neural network. *Proc. Int. Electric Vehicle Symp.*, CD-ROM.

Riutort, J. F., Rulliere, E. and Toutain, E. (1995). An improved coulometric gauge for electric vehicles. *Proc. Electric Vehicle Technology Conf*., **3**, 7–15.

Salameh, Z. M., Casacca, M. A. and Lynch, W. A. (1992). A mathematical model for lead-acid batteries. *IEEE Trans. Energy Conversion*, **7**, 93–98.

Scholkopf, B., Smola, A. J., Williamson, R. C. and Bartlett, P. L. (2000). New support vector algorithms. *Neural Computation* **12**, **5**, 1207-1245.

Shen, W. X., Chau, K. T. and Chan, C. C. (2005). Neural network-based residual capacity indicator for nickel-metal hydride batteries in electric vehicles. *IEEE Trans. Vehicular Technology* **54**, **5**, 1705-1712.

Song, S. K. and Kim, K. H. (1994). A dynamic state of charge model for electric vehicles batteries. *Proc. Int. Electric Vehicle Symp*., **2**, 519–527.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer Verlag. New York. 1–20.

Wu, C. H., Ho, J. M. and Lee, D. T. (2004). Travel time prediction with support vector regression. *IEEE Trans. Intelligent Transportation Systems* **5**, **4**, 276–281.

Wu, J., Zhang, C. H. and Cui, N. X. (2003). PSO algorithm-based parameter optimization for HEV powertrain and its control strategy. *Int. J. Automotive Technology* **9**, **1**, 53–59.

Zhang, J., Peng, Q. C., Shao, H. Z. and Shao, T. G. (2006). Nonlinear noise filtering with support vector regression. *Intelligent Systems Design and Applications, 6th Int. Conf.*, **1**, 172–176.