Computation of the Speed of Four In-Wheel Motors of an Electric Vehicle Using a Radial Basis Neural Network

Merve Yildirim
Department of Electrical and Electronics Engineering
University of Firat, Elazig, Turkey
merveyildirim@firat.edu.tr

Mehmet Cem Catalbas
Department of Electrical and Electronics Engineering
University of Firat, Elazig, Turkey
catalbas@firat.edu.tr

Arif Gulten
Department of Electrical and Electronics Engineering
University of Firat, Elazig, Turkey
agulten@firat.edu.tr

Hasan Kurum
Department of Electrical and Electronics Engineering
University of Firat, Elazig, Turkey
hkurum@firat.edu.tr

Abstract—**This paper presents design and speed estimation for an Electric Vehicle (EV) with four in-wheel motors using Radial Basis Neural Network (RBNN). According to the steering angle and the speed of EV, the speeds of all wheels are calculated by equations derived from the Ackermann-Jeantand model using CoDeSys Software Package. The Electronic Differential System (EDS) is also simulated by Matlab/Simulink using the mathematical equations. RBNN is used for the estimation of the wheel speeds based on the steering angle and EV speed. Further, different levels of noise are added to the steering angle and the EV speed. The speeds of front wheels calculated by CoDeSys are sent to two Induction Motor (IM) drives via a Controller Area Network-Bus (CAN-Bus). These speed values are measured experimentally by a tachometer changing the steering angle and EV speed. RBNN results are verified by CoDeSys, Simulink, and experimental results. As a result, it is observed that RBNN is a good estimator for EDS of an EV with in-wheel motor due to its robustness to different levels of sensor noise.**

Keywords-electric vehicle; electronic differential system; in-wheel motor; radial basis neural network; speed estimation

I. INTRODUCTION

Electric Vehicles (EVs) are increasingly being used due to their reduced pollution emissions and fuel consumption [1, 2]. Further, recent efficient electric motors and developments in drive and battery technology have resulted to an increase in the popularity of EVs [3]. EV motors having a differential gear are fitted into the wheels of the vehicle to reduce the mass caused from the batteries and drive-trains [4]. In traditional vehicles, mechanical differentials are utilized in slippery and sloping roads to distribute power and torque equally to the traction wheels [1]. On the contrary, the Electronic Differential System (EDS) is used in EVs to eliminate mechanical losses, maintenance, and repair costs of gears caused by the powertrains. According to the curve of the road, the outer wheel speed of the EV must be higher than the inner wheel speed for safe driving [5].

An EDS for EV is modelled by NN based on the vehicle speed and steering angle in [1]. Using a fuzzy logic control method to estimate the slip rate of each wheel, a new Electronic Differential (ED) control for two in-wheel motors of EV is investigated in [4]. The designed ED control is validated by Matlab/Simulink results. In [6], authors estimate the speeds of four wheels for EV by using the Ackermann-Jeantand model [6]. The estimated speeds are also verified by Matlab/Simulink and experimental results. In [7], the speeds of the four wheels are calculated by NN PID ED based on the steering angle and speed of EV. An EDS for rear wheels of an EV used in Brushless DC motors is investigated by Fuzzy Logic Speed Controller using the Ackermann-Jeantand model [8]. The proposed EDS is verified by Matlab/Simulink results. In [9], authors proposed an EDS for the rear wheels of EV driven by fault tolerant permanent magnet motors. The speeds of the wheels are calculated by Ackermann-Jeantand model. Authors in [10] design an EDS for the rear wheels of an EV using DC motors based on the steering angle and vehicle speed. The results of the designed EDS are verified by experimental results. NN control is used for estimating the rear wheels of an EV in [11]. The simulation results are tested by two 37-kW IMs. In another paper [12], an EDS for two rear wheels of EV is presented and analyzed versus the speed and torque observed for the DC motor.

In this paper, an EDS for all four wheels of an EV is modeled and the speeds of all wheels are calculated by the CoDeSys Software Package using mathematical equations derived from the Ackermann-Jeantand model. Then, the EDS simulation is also conducted in Matlab/Simulink using these equations. The speeds of the wheels are estimated by an Radial Basis Neural Network (RBNN) based on the steering angle and the EV speed. Further, white noise at specified amplitudes is added to the EV speed and the steering angle in order to investigate the behavior under sensor noise. According to the

This work is supported by TUBITAK (The Scientific and Technological Research Council of Turkey) with the project, 113M090.

noise levels, the speeds estimated by the RBNN are compared with Simulink results. Furthermore, the speeds of the front wheels calculated by CoDeSys are sent to two IM drives and measured by a tachometer experimentally. RBNN results are also verified by CoDeSys, Simulink, and experimental results.

II. ELECTRONIC DIFFERENTIAL SYSTEM FOR EV

The Ackermann-Jeantand model is employed for EDS design which is generally preferred at low speeds due to the effect of centrifugal force and centripetal forces is utilized [4]. A position encoder is used for the steering angle (δ). Once δ is zero, EV drives on a straight road. If δ is different from zero, it means that EV drives on the curved road. According to the turning direction, the speed of the outer wheel has to be higher than that of the inner wheel [4, 6, 13]. In this situation, the EDS is activated. The equations derived from this model are as follows:

The inner and outer steering angles of the front wheel are respectively given by:

$$\delta_{1,2} = \arctan\left[\frac{L \cdot \tan(\delta)}{L \pm ((K/2) \cdot \tan(\delta))}\right] \tag{1}$$

where K is the distance between the left and right kingpin, L is the distance between the front and rear wheel.

To estimate the speeds, the turning radii of the front inner and outer wheels and rear inner and outer wheels can be respectively defined by:

$$R_{1,2} = \frac{L}{\sin(\delta_{1,2})} \tag{2}$$

$$R_{3,4} = \frac{L}{\tan(\delta)} \pm \frac{d_r}{2} \tag{3}$$

where d_r is the distance between rear wheels.

The gravity centre radius of the EV is:

$$R_{cg} = \sqrt{(R_3 + (d_r/2))^2 + (l_r)^2} \tag{4}$$

where l_r is the distance between the rear wheel and gravity centre.

The angular speeds of the front inner and outer wheels, and the rear inner and outer wheels can be respectively expressed by:

$$w_{1,2} = \frac{V \cdot R_{1,2}}{(R_{cg}) \cdot r} \tag{5}$$

$$w_{3,4} = \frac{V \cdot R_{3,4}}{(R_{cg}) \cdot r} \tag{6}$$

where r is the radius of the wheel and V is speed of EV.

The equations derived from Ackermann-Jeantand geometry are given into CoDeSys Software Package. L, l_r, d_r, r, and K parameters taken from a vehicle model are shown in Table I. These parameters are used as the constant values in the CoDeSys programmer.

TABLE I. EDS MODEL PARAMETERS

Parameters	***L***	***l_r***	***d_r***	***r***	***K***
Values (m)	2.285	0.835	1.35	0.395	1.219

III. MODELLING OF EDS VIA RBNN

The implementation of RBNN is similar to performing exact interpolation of various data points in a multidimensional space [14]. The RBNNs are rather popular due to their simple structure, fast training, and implementation time. The network architecture of an RBNN is similar to that of a classical NN [15]. The structure of RBNN includes the input layer, hidden layer and output layer [16, 17]. The input vector x is used as an input in all radial basis functions. μ_j is the vector determining the center of basic function ϕ_j and has elements μ_{ji}. The weight parameter is represented by w_{kj} and bias term is equal to w_{k0}. The hidden layer includes nonlinear radial basis activation function and the output layer is linear combination of hidden layer. Each node in the hidden layer uses an RBF as a nonlinear activation function. $\phi_0(x)=1$ corresponds to the bias in the output layer. The RBF network can give an optimal solution to the adjustable weights in the minimum Mean Squared Error (MSE) sense by linear optimization method [18]. The smoothness level of interpolation function is controlled by the parameter of spread (σ). The relationship between input, output, and nonlinear transfer function is given by :

$$\phi_j(x) = \exp\left(-\frac{\left\| x - \mu_j \right\|}{2\sigma_j^2}\right) \tag{7}$$

$$y_k(x) = \sum_{j=1}^{M} w_{kj}\phi_j(x) + w_{k0} \tag{8}$$

RBNNs are generally utilized for classification and curve fitting. Our test system has two inputs: the steering angle and the speed of EV and the outputs are the speeds of front and rear wheels. The overall scheme is shown in Figure 1.

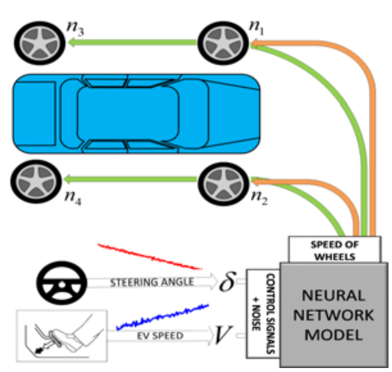

Fig. 1. Overall scheme

The Signal-to-noise ratio is given by

$$SNR_{dB} = 20\log_{10}\left(\frac{A_{signal}}{A_{noise}}\right) \qquad (9)$$

where A_{signal} and A_{noise} are RMS amplitudes of the signal and noise as decibel, respectively.

Gaussian White Noise (GWN) is added to NN systems for performance analysis. GWN in this case represents sensor interference in steering angle and EV speed [19]. Different levels of GWN are used to obtain the optimal solution. MSE gives an idea about the behavior of the RBNN training process. Low MSE means a successful data fitting. Therefore, the number of epochs is selected at nearly fixed value of MSE. In this study, the number of epochs is taken as 400. The MSE versus number of epochs is shown in Figure 2.

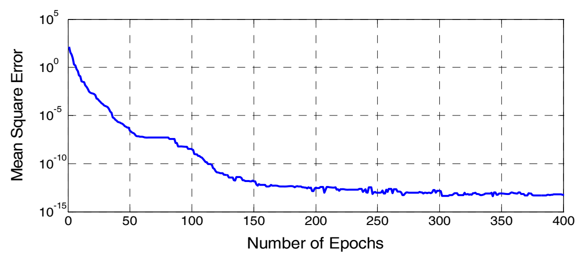

Fig. 2. MSE versus number of epochs.

IV. MATLAB/SIMULINK MODEL OF EDS

EDS is simulated by Matlab/Simulink using the equations derived from the Ackermann Jeantand model. The simulation model and subsystem of the model are shown in Figures 3 and 4, respectively.

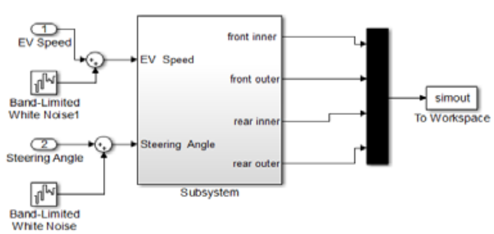

Fig. 3. Matlab/Simulink model of EDS.

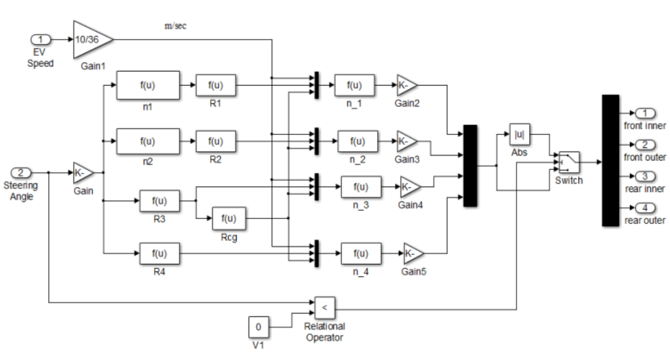

Fig. 4. Subsystem of EDS model.

The maximum speed of an EV having 21-inch wheel size is 50.806 km/h. Therefore, this speed is used as maximum value in the simulation. Maximum manoeuvrability can be calculated by minimum circle radius of outer wheel trace at 10 km/h vehicle speed. Turning radius is given by

$$\varsigma = \frac{L}{\sin(\delta_1 - \delta_2)} \qquad (10)$$

Turning radius changes between 7 m and 9 m in passenger cars. Since the movement area where is placed in the wheels is another factor limited the steering of the wheel, the steering angle is taken as maximum value (15˚). The speeds of four wheels are calculated by Simulink changing the steering angle from 1° to 10° and EV speed from 0 to 40 km/h. Then, the relationship between input parameters *(δ,V)* and four wheel speeds are trained to NN model. The speeds are estimated by changing the steering angle from 10° to 15° and the EV speed from 40 to 50 km/h. The RBNN results are verified by Simulink results for all wheel speeds as shown in Table II, III, IV, and V.

TABLE II. COMPARISON OF SIMULINK AND RBNN RESULTS FOR FRONT INNER WHEEL.

δ (°)	13		14		15	
V (km/h)	**Simulink Results**	**RBNN Results**	**Simulink Results**	**RBNN Results**	**Simulink Results**	**RBNN Results**
45	43.334	43.300	43.300	43.253	43.282	43.210
46	44.297	44.226	44.263	44.171	44.244	44.117
47	45.260	45.119	45.225	45.052	45.205	44.985
48	46.223	45.961	46.187	45.879	46.167	45.792
49	47.186	46.729	47.149	46.626	47.129	46.514
50	48.149	47.393	48.112	47.264	48.091	47.122

TABLE III. COMPARISON OF SIMULINK AND RBNN RESULTS FOR FRONT OUTER WHEEL.

δ (°)	13		14		15	
V (km/h)	**Simulink Results**	**RBNN Results**	**Simulink Results**	**RBNN Results**	**Simulink Results**	**RBNN Results**
45	48.715	48.671	49.083	49.018	49.465	49.359
46	49.797	49.706	50.174	50.050	50.564	50.385
47	50.880	50.702	51.265	51.039	51.663	51.361
48	51.962	51.636	52.356	51.958	52.762	52.261
49	53.045	52.478	53.446	52.779	53.861	53.053
50	54.128	53.195	54.537	53.465	54.961	53.701

TABLE IV. COMPARISON OF SIMULINK AND RBNN RESULTS FOR REAR INNER WHEEL

δ (°)	13		14		15	
V (km/h)	**Simulink Results**	**RBNN Results**	**Simulink Results**	**RBNN Results**	**Simulink Results**	**RBNN Results**
45	41.783	41.752	41.514	41.474	41.241	41.184
46	42.711	42.645	42.436	42.355	42.157	42.052
47	43.640	43.507	43.359	43.203	43.074	42.883
48	44.568	44.320	44.281	43.997	43.990	43.657
49	45.497	45.059	45.204	44.715	44.907	44.350
50	46.425	45.697	46.126	45.327	45.823	44.933

TABLE V. COMPARISON OF SIMULINK AND RBNN RESULTS FOR REAR OUTER WHEEL.

δ (°)	13		14		15	
V (km/h)	Simulink Results	RBNN Results	Simulink Results	RBNN Results	Simulink Results	RBNN Results
45	47.899	47.856	48.115	48.053	48.331	48.233
46	48.963	48.874	49.184	49.066	49.405	49.237
47	50.028	49.853	50.254	50.035	50.479	50.192
48	51.092	50.769	51.323	50.935	51.553	51.071
49	52.156	51.595	52.392	51.738	52.627	51.845
50	53.221	52.294	53.461	52.405	53.701	52.475

V. EXPERIMENTAL RESULTS

The experimental setup is established to obtain the results of designed EDS as shown in Figure 5. The speeds of the front wheels calculated by CoDeSys are sent to three-phase IM drives (VFD C2000) via CAN-Bus. An encoder is used for the steering angle. Three-phase IM drives, HY-TTC 60 and encoder which are fed from 24 V power supply communicate with each other on CAN-Bus. HY-TTC 60 processor which is an advanced model of 16-bit controller family produced by TTControl Company and consists of power and control cards is a master unit on CAN-Bus line. Two drives connected to IM motors and encoder are slave units on the CAN-Bus line. Characteristic line impedances are used in beginning and end of the line. By changing the steering angle from 0° to 15° degree and taking the EV speed which is fixed as 50.806 km/h, the speeds of front wheels calculated by CoDeSys using the Ackermann-Jeantand model are sent to IM drives. The speeds of front wheels are measured by a tachometer experimentally. CoDeSys results are verified by Simulink, experimental, and RBNN results as given in Table VI. According to the difference of the comparison, MSE of all wheel speeds estimated by RBNN are shown in Table VII based on the noise level and function spread. Once examining the Table VII, the large number of the spread is preferred to estimate the speeds due to having less error based on different noise levels. Hence, the spread value and noise level are selected as 10 and 40 dB, respectively. RBNN results of the front inner and outer, rear inner and outer wheel speeds are illustrated based on the noise level in Figures 6-9, respectively.

TABLE VI. COMPARISON OF CODESYS, SIMULINK, EXPERIMENTAL, AND RBNN RESULTS FOR FRONT WHEEL SPEEDS

δ(°)	Codesys Results (km/h)		Simulink Results (km/h)		Exp. Results (km/h)		RBNN Results (km/h)	
	n_1	n_2	n_1	n_2	n_1	n_2	n_1	n_2
0	50.807	50.807	50.803	50.803	50.803	50.803	50.805	50.805
1	50.577	51.050	50.582	51.054	50.562	51.054	50.575	51.048
2	50.360	51.306	50.361	51.305	50.351	51.285	50.359	51.305
3	50.158	51.576	50.160	51.577	50.139	51.536	50.157	51.575
4	49.969	51.859	49.969	51.858	49.959	51.848	49.968	51.858
5	49.794	52.155	49.798	52.160	49.778	52.149	49.793	52.154
6	49.633	52.465	49.637	52.461	49.607	52.431	49.632	52.463
7	49.487	52.787	49.486	52.783	49.496	52.742	49.486	52.786
8	49.355	53.123	49.356	53.124	49.346	53.084	49.354	53.122
9	49.239	53.471	49.235	53.476	49.225	53.446	49.238	53.470
10	49.137	53.834	49.134	53.838	49.134	53.828	49.136	53.832
11	49.051	54.209	49.054	54.210	49.044	54.159	49.049	54.207
12	48.980	54.598	48.984	54.602	48.954	54.572	48.978	54.596
13	48.926	55.001	48.923	55.004	48.903	54.963	48.922	54.997
14	48.888	55.417	48.883	55.416	48.873	55.386	48.882	55.412
15	48.867	55.847	48.863	55.848	48.843	55.808	48.858	55.839

TABLE VII. MEAN SQUARE ERRORS OF FOUR WHEEL SPEEDS

Noise (dB)	*30 dB*			
Spread (σ)	n_1	n_2	n_3	n_4
0.5	12.004	15.369	10.822	14.685
1	31.400	36.197	30.987	35.445
5	5.627	6.663	5.340	6.480
10	0.741	0.848	0.757	0.815
Spread (σ)	*40 dB*			
0.5	11.987	15.346	10.807	14.662
1	30.909	35.763	30.484	35.006
5	5.397	6.369	5.094	6.205
10	0.323	0.368	0.320	0.364

Fig. 5. The experimental setup

Estimation results of RBNN for the front and rear wheel speeds are compared with Simulink results. Differences between Simulink and RBNN results are illustrated for the front inner and outer wheel speeds based on the noise levels in Fig. 10-11, respectively. As shown in the figures, the errors of all wheel speeds increase with rising of steering angle and EV speed.

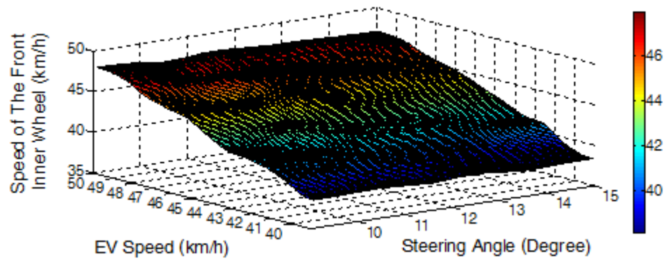

Fig. 6. RBNN results of front inner wheel (40 dB).

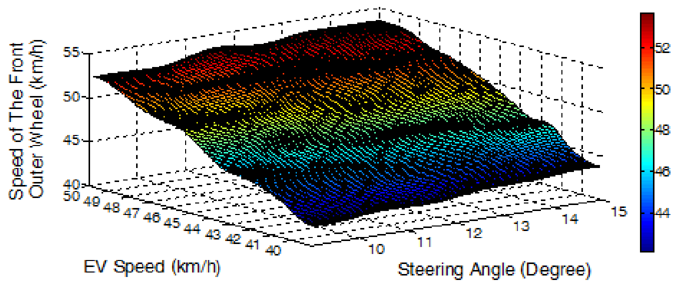

Fig. 7. RBNN results of front outer wheel (40 dB).

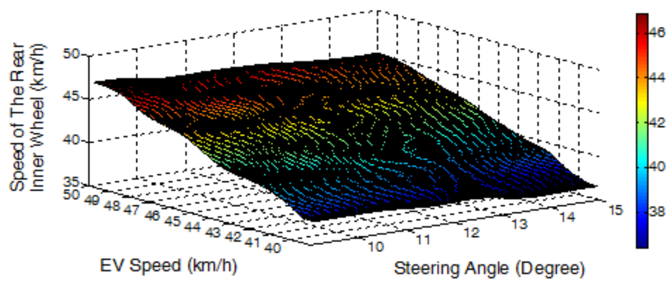

Fig. 8. RBNN results of rear inner wheel (40 dB).

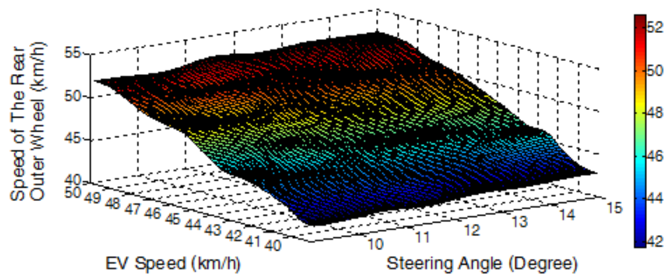

Fig. 9. RBNN results of rear outer wheel (40 dB).

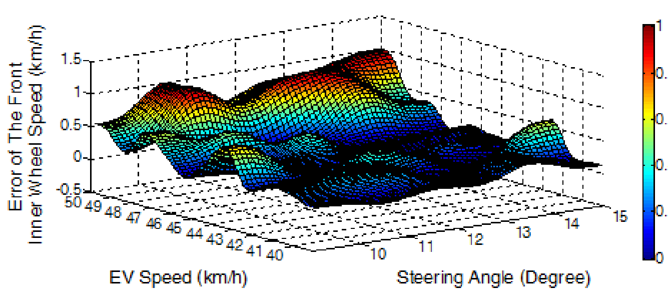

Fig. 10. Difference between Simulink and RBNN results of front inner wheel.

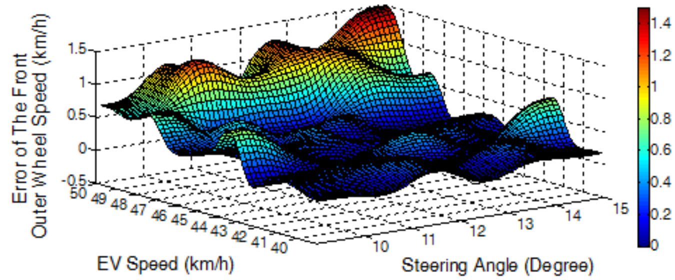

Fig. 11. Difference between Simulink and RBNN results of front outer wheel.

VI. CONCLUSION

In this paper, EDS modelling and estimation parameters for front and rear wheels of an EV are realized. According to the steering angle and speed of the EV, the speeds of the four wheels are calculated by mathematical equations derived from the Ackermann-Jeantand model using the CoDeSys Software Package. Matlab/Simulink modeling of EDS is also realized by using the equations obtained from the Ackermann-Jeantand model. The speeds of the all wheels are calculated by Simulink changing the steering angle from 1˚ to 15˚ and the EV speed from 0 km/h to 50 km/h. Different levels of white noise are added to the steering angle and EV speed as sensor noise. RBNN is used to estimate the speeds changing the steering angle from 10˚ to 15˚ and the EV speed from 40 km/h to 50 km/h. The steering angle is taken by an encoder. The calculated front wheel speeds are sent to IM drives via CAN-Bus for taking experimental results by a tachometer. RBNN results are verified by comparing with Simulink, experimental and CoDeSys results. Furthermore, RBNN results are compared with Simulink results based on noise and function of spread at different levels. To conclude, RBNN is show to be appropriate for the estimation of EDS parameters due to its minor error and its robustness to different levels of sensor noise.

VII. REFERENCES

[1] J. S. Lee, Y. J. Ryoo, Y. C. Lim, P. Freere, T. G. Kim, S. J. Son, E. S. Kim, “A neural network model of electric differential system for electric vehicle”, IEEE 26th Annual Conference of the Industrial Electronics Society (IECON), Vol. 1, pp. 83-88, 2000

[2] T. Ma, A. M. Osama, “Optimal charging of plug-in electric vehicles for a car-park infrastructure”, IEEE Transactions on Industry Applications, Vol. 50, No. 4, pp. 2323-2330. 2014

[3] M. Yildirim, M. Polat, H. Kurum, “A survey on comparison of electric motor types and drives used for electric vehicles”, IEEE 16th International Power Electronics and Motion Control Conference and Exposition (PEMC), pp. 218-223, 2014

[4] Y. E. Zhao, J. W. Zhang, X. Q. Guan, “Modelling and simulation of the electronic differential system for an electric vehicle with two-motor-wheel drive”, International Journal of Vehicle Systems Modelling and Testing, Vol. 4, No. 1-2, pp. 1209-1214, 2009

[5] J. L. Febin Daya, P. Sanjeevikumar, F. Blaabjerg, P. W. Wheeler, J. O. Ojo, “Implementation of wavelet-based robust differential control for electric vehicle application”, IEEE Transactions on Power Electronics, Vol. 30, No. 12, pp. 6510-6513, 2015

[6] Y. Zhou, S. Li, X. Zhou, Z. Fang, “The control strategy of electronic differential for EV with four in-wheel motors”, IEEE Control and Decision Conference (CCDC), pp. 4190-4195, 2010

[7] L. Zhai, S. Dong, "Electronic differential speed steering control for four in-wheel motors independent drive vehicle", 9th World Congress on Intelligent Control and Automation (WCICA), pp. 780-783, June 2011

[8] H. Kahveci, H. I. Okumus, M. Ekici, "An electronic differential system using fuzzy logic speed controlled in-wheel brushless DC motors", IEEE Fourth International Conference on Power Engineering, Energy and Electrical Drives (POWERENG), pp. 881-885, May 2013

[9] R. Wan, G. Liu, D. Zhang, W. Gong, "A fault-tolerant electronic differential system of electric vehicles", in IEEE Vehicle Power and Propulsion Conference (VPPC), pp. 1-5, Oct. 2013

[10] S. Sharma, R. Pegu, P. Barman, "Electronic differential for electric vehicle with single wheel reference", IEEE 1st Conference on Power, Dielectric and Energy Management at NERIST (ICPDEN), pp. 1-5, Jan. 2015

[11] A. Haddoun, M. E. H. Benbouzid, D. Diallo, R. Abdessemed, J. Ghouili, K. Srairi, "Modeling, analysis, and neural network control of an EV electrical differential", IEEE Transactions on Industrial Electronics, Vol. 55, No. 6, pp. 2286-2294, 2008

[12] D. Foito, G. Manuel, V. F. Pires, "Road motion control electric vehicle with speed and torque observer", IEEE International Conference on New Concepts in Smart Cities: Fostering Public and Private Alliances (SmartMILE), pp. 1-6, 2013

[13] M. Yildirim, E. Oksuztepe, B. Tanyeri, H. Kurum, "Electronic differential system for an electric vehicle with in-wheel motor", in 9th International Conference on Electrical and Electronics Engineering (ELECO), pp. 1048-1052, 26-28 Nov. 2015

[14] M. J. L. Orr, Introduction to radial basis function networks, Technical Report, Center for Cognitive Science, University of Edinburgh, 1996.

[15] T. Bow, Pattern recognition and image preprocessing, CRC Press, Sing, 2002.

[16] C. M. Bishop, Neural networks for pattern recognition, Oxford university press, 1995.

[17] A. Rubaai, M. D. Kankam, "Adaptive tracking controller for induction motor drives using online training of neural networks", IEEE Transactions on Industry Applications, Vol. 36, No. 5, pp. 1285-1294, 2000

[18] K. Du, NS Madisetti, Neural networks in a softcomputing framework, Springer Science & Business Media, 2006

[19] S. Stergiopoulos, Advanced signal processing: theory and implementation for sonar, radar, and non-invasive medical diagnostic systems, CRC Press, 2009

[20] M. Yildirim, E. Oksuztepe, B. Tanyeri, H. Kurum, Design of electronic differential system for an electric vehicle with in-wheel motor, IEEE Power and Energy Conference at Illinois (PECI), pp. 1-5, 2016