Recurrent Neural Network Models for Myoelectric-based Control of a Prosthetic Hand

Teodor-Adrian Teban
Dept. of Automation and Applied Inf.
Politehnica University of Timisoara
Timisoara, Romania
adrian.teban@student.upt.ro

Radu-Emil Precup
Dept. of Automation and Applied Inf.
Politehnica University of Timisoara
Timisoara, Romania
radu.precup@aut.upt.ro

Elena-Cristina Lunca
Dept. of Automation and Applied Inf.
Politehnica University of Timisoara
Timisoara, Romania
luncaelenacristina@yahoo.com

Adriana Albu
Dept. of Automation and Applied Inf.
Politehnica University of Timisoara
Timisoara, Romania
adriana.albu@aut.upt.ro

Claudia-Adina Bojan-Dragos
Dept. of Automation and Applied Inf.
Politehnica University of Timisoara
Timisoara, Romania
claudia.dragos@aut.upt.ro

Emil M. Petriu
School of EECE
University of Ottawa
Ottawa, Canada
petriu@uottawa.ca

***Abstract*—This paper proposes a set of recurrent neural networks (RNNs) capable of replicating the non-linear mechanism of a prosthetic hand based on surface myoelectric sensors. The experimental results of the RNN show a good result of the system for the training data and an acceptable result on the validation data. A comparison between the developed RNNs and a similar size non-recurrent neural network is included.**

Keywords— finger angles, flexion percentages, myoelectric sensors, prosthetic hands, recurrent neural networks

I. INTRODUCTION

The creation of lower arm prosthetic systems revolves around the pattern recognition of myoelectric signals (MESs) [1]–[3] coming from the muscles. The important task is to correlate the movement of the arm with the muscle activation in order to mimic the nonlinear dynamic system. Neural networks (NNs) [4], [5], maximal Lyapunov exponent inserted in NN [6] and evolving fuzzy models [7] are commonly used to model this nonlinear system. Practical achievements of MES-based control of prosthetic arms and hands are described in [8]–[10]. Recent results on fuzzy control and modeling are presented in [11]–[13]. The functional electrical stimulation of the arm by using electric impulses to stimulate the neural pathway is another method to regain movement in case of nerve damage without the hand being amputated.

This paper suggests a system architecture for prosthetic hand myoelectric-based control systems and variable structure RNN models of the human hand dynamics, i.e., the finger dynamics to be used as reference models in myoelectric-based control systems. The inputs of this nonlinear system are the MESs obtained from eight sensors placed on human subject's arm, and the outputs are the flexion percentages that correspond to the midcarpal joint angles. For the sake of simplicity the flexion percentages are also called flex percentages and finger angles in the rest of the paper.

The system architecture is based on the general architecture given in [13] and on the results reported in [14], but this paper is focused on Proportional-Integral-Derivative (PID) controllers, which are intended to be first used in the real-time control of the prosthetic system. The angles involved in the finger dynamics are in fact the midcarpal joint angles, and they are referred to as follows as finger angles.

The proposed variable structure recurrent neural network (RNN) is used instead of a non-recurrent NN due to its ability to learn from previous steps (t_{n-1}, t_{n-2}, ...) meaning better performance for this particular nonlinear dynamic system. The RNN considered in this paper is a Long-Short Term Memory (LSTM) [15], [16] and it will be compared with a regular neural network (NN) in order to evaluate the strengths and weaknesses of this approach.

The paper demonstrates that an RNN is capable of replicating a nonlinear mechanism of a real human hand. This output is used to drive a prosthetic hand by means of the system architecture presented in the paper. The dynamics of finger flexion is partially taken into account, the prosthetic hand delays being ignored for now. This appears in the proposed application, as fingers flexion are used as references of prosthetic hand.

Details on the experimental setup (for example, what sensors have been used, what sensitivity, frequency band, electrodes, etc.) are not given in the sequel as the paper is mainly focused on modeling. However, minimum technical details related to the sensor placement are presented in the next section. No specific detail on the use of the obtained results in the control of prosthetic hand is given, so the dynamics of the flexion of the fingers is ignored.

The paper presents an RNN to model the relation between the activation of eight muscles in harm and the flexion of the five fingers. The purpose of the model is to generate the flexion references for the fingers of a prosthetic hand, controlled with PID controllers. The proposed RNN uses 100 past inputs values in addition to the averaging inputs during 10 and 100 samples. The obtained results are

978-1-5386-4444-7/18/$31.00 ©2018 IEEE

compared with the ones obtained with a regular NN using the same inputs, averaged inputs and neurons of one layer as those for the RNN. The results are also compared to the ones obtained using the validation data for training.

This paper is organized as follows: sensors setup will be described in the next section and covers the type, number of sensors and placement. The overall system architecture is discussed in Section III and concerns the communication between different modules. The RNN architecture is described in section IV treating topics as structure, number of parameters and type of neurons. Section V contains the results and comparison with other NN model, with the conclusions and future improvements being summarized in Section VI.

II. Sensors and Their Placement

The myoelectric sensors are in essence differential amplifiers that transform the electric signals received from the muscles into signals that can be read by a microcontroller as in Fig. 1. The signal is amplified 200 times and then rectified from max (–5V; +5V) to (0V; 5V). The final step is to smooth the signal from fast variations to an average in order to reduce the load on the microcontroller. This type of sensors is used in many versions of lower arm prosthetics [17]–[19] because they are relatively easy to use and do not require surgery in order to place them.

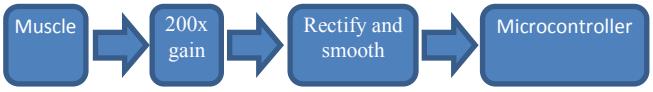

Fig. 1. Myoelectric sensor.

The sensors are placed directly above the muscle or as close as possible in order to read the signal from the targeted muscle and with the smallest amount of interferences from other muscles. The placement of the 8 sensors is done as follows: four sensors on the flexor digitorum superficialis as in Fig. 2, two sensors on the extensor digitorum and one on extensor digiti minimi as in Fig. 2, and one sensor on abductor pollicis longus as in Fig. 3.

Fig. 2. Placement of sensors 1 to 7 on the hand.

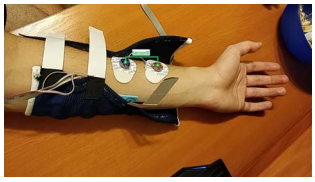

Fig. 3. Placement of sensor 8 on the hand.

III. System Architecture

The system is composed of three major modules: acquisition module, neural network module and actuators module as shown in Fig. 4.

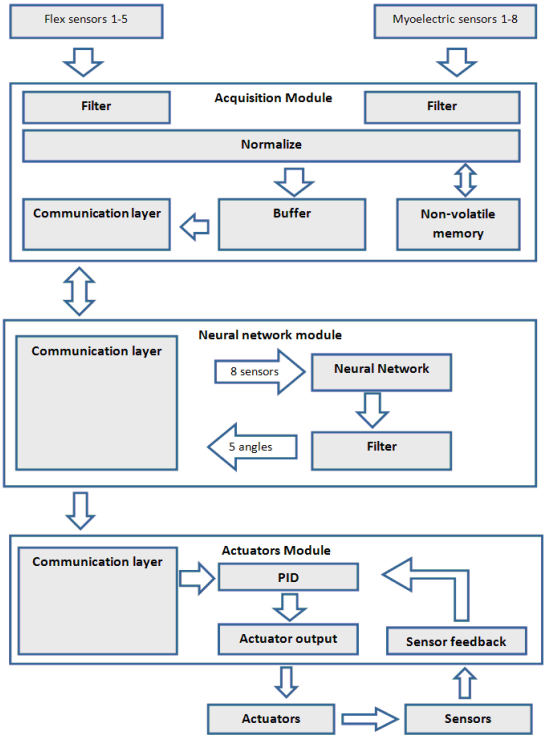

Fig. 4. System architecture for prosthetic hand control.

The split of the system is done in order to reduce the complexity of the neural network to a simple Multi Input-Multi Output (MIMO) system specialized only in mapping the non-linear input of the MESs to the flex angle of the fingers. In this case the NN performance can be easily calculated without being influenced by external factors.

The acquisition module is based on an Atmega8 microcontroller and is responsible for the real-time acquisition of the MES signals and flex sensor signals. The acquisition period is 10 ms and the values go through the first layer of filtering which removes unwanted noise. The filtered values then go through the normalization phase which sets the same range for each sensor (0% to 100% for flex sensors and 0 to 255 for myoelectric sensors). Sensor ranges for the flex sensors from 0% to 100% for each finger represent the finger fully opened and fully closed. For the MESs the range 0 to 255 is just the maximum value on 8 bits, which represents the electric impulse that can be received by the targeted muscle. 8 bits were chosen in order to avoid unnecessary noise caused by the cables and also to lighten the load on the neural network. The obtained values are then sent through a serial interface to the neural network module to be processed.

The acquisition module works in two modes: training and normal. In the training mode, it sends the flex sensor measurements and the MESs measurements because the system is used on a healthy hand and creates a correlation

between the sensors which is then used to train the neural network. In normal mode, the flex sensors are disconnected and no values are sent to the NN.

The neural network module, into normal mode, takes as an input the 8 sensors and outputs the flexion percentage of each finger that corresponds to the midcarpal joint angles. This output is then filtered again to remove noise and smoothen, and after that it is sent to the actuators. The training of the NN is performed separately on a high performance device and had as inputs the measured MESs and the flexion percentages.

The actuators module is in charge of transforming the finger closing percentage sent by the RNN into prosthetic finger movement. The control of this movement is achieved by an appropriately designed PID controller, which controls the actuators (servos) and has two inputs: the flexion percentages from the RNN and sensor feedback (force, angle) from the prosthetic hand.

IV. Recurrent Neural Network Architecture

The LSTM takes as input the t_n value but also the past x values, where in this case x = 100 (1 s), with the sampling period of 10 ms. The chosen 1 s of recurrence is a covering value of fast and average speed contraction of the fingers, which are more dependent of the past values. Slow movements (over 1 s) are more reliant of sensor averages as presented the sequel.

The internal structure of the network (for only one time stamp) makes use of four layers shown in Fig. 5: input layer, LSTM layer, a hidden layer and an output layer

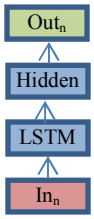

Fig. 5. LSTM network layers.

For each time stamp n, the input layer is composed of $24 = 8 \times 3$ neurons, which take the input vector

$$[\mathbf{Z}^T \ \mathbf{A}^T \ \mathbf{S}^T]^T \in \Re^{24} \tag{1}$$

composed of three concatenated vectors of eight elements (T indicates matrix transposition):

$$\mathbf{Z} = [z_{1,n} \ z_{2,n} \ z_{3,n} \ z_{4,n} \ z_{5,n} \ z_{6,n} \ z_{7,n} \ z_{8,n}]^T, \tag{2}$$

where $z_{j,n}$ is the output of myoelectric sensor j, j=1…8,

$$\mathbf{A} = [a_{1,n} \ a_{2,n} \ a_{3,n} \ a_{4,n} \ a_{5,n} \ a_{6,n} \ a_{7,n} \ a_{8,n}]^T, \tag{3}$$

with

$$a_{j,n} = (\sum_{i=0}^{9} z_{j,n-i}) / 10 \tag{4}$$

as the average of the past ten samples (100 ms) of myoelectric sensor j, j=1…8, and

$$\mathbf{S} = [s_{1,n} \ s_{2,n} \ s_{3,n} \ s_{4,n} \ s_{5,n} \ s_{6,n} \ s_{7,n} \ s_{8,n}]^T, \tag{5}$$

where

$$s_{j,n} = (\sum_{i=0}^{99} z_{j,n-i}) / 100 \tag{6}$$

is the average of the past 100 samples (1 s) of myoelectric sensor j, j=1…8.

The **Z**, **A** and **S** vectors were chosen based on experimental configurations in order to obtain the best accuracy, the combination between the instant value of the sensor. The 100 ms and 1s averages exhibit the best results.

The LSTM layer contains the neurons connected from the input layer and also the connection to the previous layers. It is configured as 300 neurons LSTM layer, and has a connection to the past 100 time steps. The size of the LSTM layer and hidden layer were raised from 50 (roughly $\sqrt{inputlayer \times timesteps}$) to 300 by parameter tuning for an increase of 7% accuracy without adding an unpractical number of parameters and training time.

The hidden layer is connected only to the current LSTM layer and also contains 300 neurons for a new abstractization of the inputs for the nonlinear behavior. Also this layer enables the LSTM to output values in the first second without having to wait for the first 100 samples to be received. This is useful in case of NN module failure and reboot in which the system down time is shorter with 1 s.

The two types of RNN operation are described in Fig. 6 for the first 1 second (a MIMO system structure) and Fig. 7 for normal functioning (a Multi Input-Single Output system structure). Therefore, the overall description of Fig. 6 and Fig. 7 corresponds to a variable structure RNN.

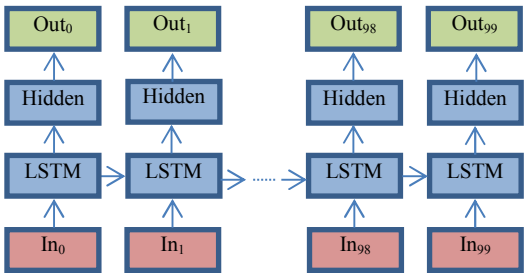

Fig. 6. LSTM network in the first second.

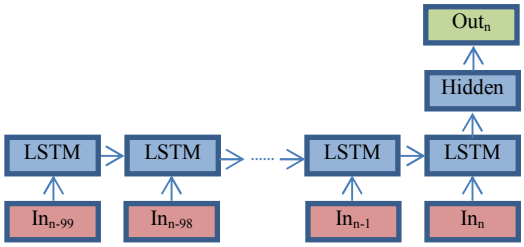

Fig. 7. LSTM network after the first second.

The final layer is the output which contains five neurons, and the output vector **O**:

$$\mathbf{O} = [o_{1,n} \;\; o_{2,n} \;\; o_{3,n} \;\; o_{4,n} \;\; o_{5,n}]^T \tag{7}$$

gathers the flexion percentage of each finger at time stamp n.

The LSTM networks parameters n_p are calculated using the formula [15]

$$n_p = 4[\mathrm{SO}(\mathrm{SI}+1)+\mathrm{SO}^2], \tag{8}$$

where SI is the input layer size and SO is the output layer size. The 4 in front of the right-hand term comes from the recurrent layer which, for each neuron, contains three gates (read / write / forget) and the fourth parameter for the cell state.

This model contains a total number of 1,112,705 parameters which were trained. They are distributed as follows: 390,000 in the first LSTM layer, 721,200 in the hidden LSTM layer and 1,505 in the output layer.

V. Recurrent Neural Network Results

The RNN model results are compared to the real hand movements (captured by the flex sensor) using the performance criterion presented by root mean square error (RMSE) for comparison to express the model accuracy:

$$\mathrm{RMSE}_j = \sqrt{\frac{1}{D}\sum_{n=1}^{D}(o_{l,n}-y_{l,n})^2}\cdot 100, \tag{9}$$

where $l = 1...5$ represents the targeted finger, D is the number of data samples in the dataset, D = 18,490 for the validation dataset, and $y_{l,n}$ is the flex percentage of the finger $l = 1...5$ in the real hand. The training and the validation data sets have major influence on the obtained results.

The training of the networks was done on a training data set of 110,374 samples, equivalent to 1,103.74 s. Each time sample is composed of 24 inputs, as shown in (1) to (6), and five outputs corresponding to the flexion percentage of each finger.

The RMSE value for the RNN on the validation data is 8 to 9% depending on the training performance and was compared with a regular NN with 2 layers of 300 neurons (Fig. 8), exhibiting an RMSE of 13 to 14%. For a performance benchmark, another instance of the RNN trained also with the validation data was created in order to simulate a more complete data set, which reached an RMSE of 2%.

An example for each finger is shown as follows: thumb (l = 1, Fig. 9), index (l = 2, Fig. 10), middle (l = 3, Fig. 11), ring (l = 4, Fig. 12) and little finger (l = 5, Fig. 13). Each figure outlines a comparison between the expected result (blue), regular NN (red), RNN (green) and RNN when the test data was also used for training the RNN (magenta). All figures illustrate the results for only 1 to 2 s of data out of the 184.9 s in order to fit in the paper.

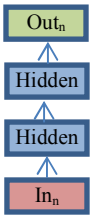

Fig. 8. Regular NN with a similar size (2 layers of 300 neurons).

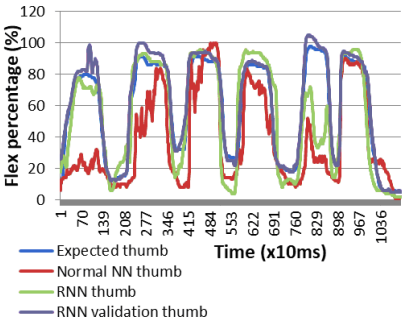

Fig. 9. Thumb finger results for expected result (blue), regular NN (red), RNN (green) and RNN trained with validation data (magenta).

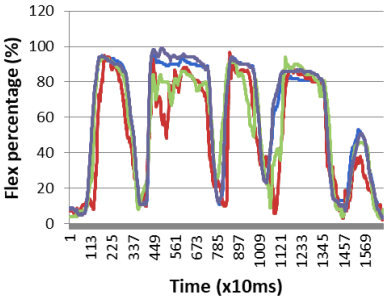

Fig. 10. Index finger results for expected result (blue), regular NN (red), RNN (green) and RNN trained with validation data (magenta).

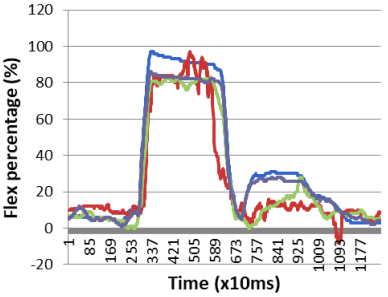

Fig. 11. Middle finger results for expected result (blue), regular NN (red), RNN (green) and RNN trained with validation data (magenta).

In general the RNN outperforms a regular NN for each finger but, as illustrated in Fig. 9. However, there is a situation in which both networks encounter an untrained case (fifth movement) and do not replicate the finger movement as it was intended. After the case was covered through training the RNN with the validation data (magenta line in Fig. 9), it is shown that the new RNN gives a more accurate result. The result also improves in the case in which the output of one finger is influenced by another muscle (Fig. 14). Here, the flexing of the little finger (black) influences the index output for booth he RNN (green) and normal NN (red) because the lack of training data for that scenario.

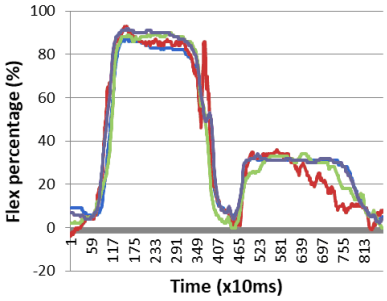

Fig. 12. Ring finger results for expected result (blue), regular NN (red), RNN (green) and RNN trained with validation data (magenta).

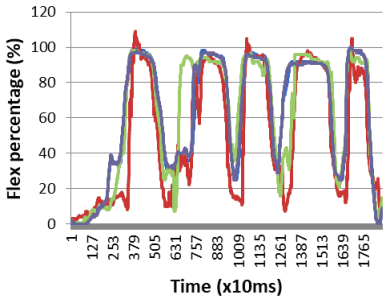

Fig. 13. Little finger results for expected result (blue), regular NN (red), RNN (green) and RNN trained with validation data (magenta).

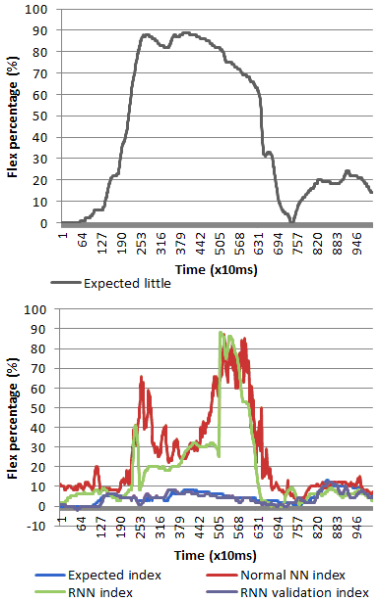

Fig. 14. Index finger influenced by little finger.

After the training, the RNN was not influenced any more and the output (magenta) follows the desired output of the index finger (blue). Nevertheless, the results presented in this section prove that necessity of using a recurrent neural network over a regular one is justified.

VI. Conclusion

This paper proposed new variable structure RNNs for modeling the nonlinear dynamic output of MESs. The obtained models are intended for prosthetic hand control. The results obtained by the RNNs outperform the non-recurrent single layer NN [4] in terms of mean absolute error between 9% and 14% and the regular NN considered here with similar size due to the recurrent layer which adds the evolution in time of the inputs.

The comparison between the training data and testing data outputs show an over fitting to the train data. This will be compensated by adding more training samples (the current training data contains a time window with the width of approximately 1103 s of recordings) and also more samples with sensors slightly moved from the ideal position.

Since the increase of the RNN size by adding more neurons and layers offered small improvements in performance, future research will focus on preprocessing the RNN inputs and also on the neural network by adding more features. Useful results from other linear and nonlinear system models and applications will be embedded including large-scale complex systems [20], evolving and self-learning systems [21], [22], automotive and aviation systems [23]–[25], mobile robots [26]–[29], fuzzy modeling and control [30]–[34], other NN architectures [35]–[39], neuron phase constancy and the use of fractional order models for developing ladder network models [40]–[42].

The development of the actuator module in terms of algorithmic design, hardware and software implementation is also a direction of future research. Fig. 4 highlights one controller type to be used in the initial simulations and next experiments.

Acknowledgment

This work was supported by Accenture and the NSERC of Canada.

References

[1] J. G. Hincapie and R. F. Kirsch, “Feasibility of EMG-based neural network controller for an upper extremity neuroprosthesis,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 17, no. 1, pp. 80–90, Feb. 2009.

[2] R. N. Khushaba, A. Al-Ani, and A. Al-Jumaily, “Orthogonal fuzzy neighborhood discriminant analysis for multifunction myoeletric hand control,” *IEEE Trans. Biomed. Eng.*, vol. 57, no. 6, pp. 1410–1419, June 2010.

[3] A. Balbinot, A. S. Júnior, and G. W. Favieiro, “Decoding arm movements by myoeletric signal and artificial neural networks,” *Intell. Control Autom.*, vol. 4, no. 1, pp. 87–93, Feb. 2013.

[4] R. J. Smith, F. Tenore, D. Huberdeau, R. Etienne-Cummings, and N. V. Thakor, “Continuous decoding of finger position from surface EMG signals for the control of powered prostheses,” in *Proc. 30th*

Ann. Int. Conf. IEEE Eng. Med. Biol. Soc., Vancouver, BC, Canada, 2008, pp. 197–200.

[5] T.-A. Teban, R.-E. Precup, T. E. A. de Oliveira, and E. M. Petriu, "Recurrent dynamic neural network model for myoelectric-based control of a prosthetic hand," in *Proc. 2016 IEEE Intl. Syst. Conf.*, Orlando, FL, USA, 2016, pp. 1–6.

[6] Y. Guo, G. R. Naik, S. Huang, A. Abraham, and H. T. Nguyen, "Nonlinear multiscale maximal Lyapunov exponent for accurate myoelectric signal classification," *Appl. Soft Comput.*, vol. 36, pp. 633–640, Nov. 2015.

[7] R.-E. Precup, T.-A. Teban, T. E. A. de Oliveira, and E. M. Petriu, "Evolving fuzzy models for myoelectric-based control of a prosthetic hand," in *Proc. 2016 IEEE Intl. Conf. Fuzzy Syst.*, Vancouver, BC, Canada, 2016, pp. 72–77.

[8] DARPA, "Proto 2," *Popular Sci. Mag.*, Aug. 2007.

[9] Advanced Arm Dynamics, "Philly, welcome to the bionic age," *Metro*, July 2011.

[10] Advanced Arm Dynamics, "Advanced surgery helps 13-year-old move prosthetic arm," *CBS Minnesota*, Sep. 2015.

[11] M. Tabakov, K. Fonal, R. A. Abd-Alhameed, and R. Qahwaji, "Fuzzy bionic hand control in real-time based on electromyography signal analysis," in *Computational Collective Intelligence. ICCCI 2016*, N. T. Nguyen, L. Iliadis, Y. Manolopoulos, and B. Trawiński, Eds. Cham: Springer, Lecture Notes in Computer Science, vol. 9875, pp. 292–302, 2016.

[12] M. Tabakov, K. Fonal, R. A. Abd-Alhameed, and R. Qahwaji, "Bionic hand control in real-time based on electromyography signal analysis," in *Transactions on Computational Collective Intelligence XXIX*, N, T. Nguyen and R. Kowalczyk, Eds. Cham: Springer, Lecture Notes in Computer Science, vol. 10840, pp. 21–38, 2018.

[13] R.-E. Precup, T.-A. Teban, E. M. Petriu, A. Albu, and I.-C. Mituletu, "Structure and evolving fuzzy models for prosthetic hand myoelectric-based control systems," in *Proc. 26th Mediter. Conf. Control Autom.*, Zadar, Croatia, 2018, pp. 625–630.

[14] P. Wide, E. M. Petriu, and M. Siegel, "Sensing and perception for rehabilitation and enhancement of human natural capabilities," in *Proc. 2010 IEEE International Workshop on Robotic and Sensors Environments*, Phoenix, AZ, USA, 2010, pp. 75–80.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[16] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000.

[17] H.-J. Liu and K.-Y. Young, "Upper-limb EMG-based robot motion governing using empirical mode decomposition and adaptive neural fuzzy inference system," *J. Intell. Robot. Syst.*, vol. 68, no. 3, pp. 275–291, Dec. 2012.

[18] M. Borghetti, E. Sardini, and M. Serpelloni, "Sensorized glove for measuring hand finger flexion for rehabilitation purposes," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 12, pp. 3308–3314, Dec. 2013.

[19] Z. Ma, P. Ben-Tzvi, and J. Danoff, "Hand rehabilitation learning system with an exoskeleton robotic glove," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 12, pp. 1323–1332, Dec. 2016.

[20] F. G. Filip, "Decision support and control for large-scale complex systems," *Ann. Rev. Control*, vol. 32, no. 1, pp. 61–70, Apr. 2008.

[21] J. Vaščák, "Adaptation of fuzzy cognitive maps by migration algorithms," *Kybernetes*, vol. 41, no. 3–4, pp. 429–443, Mar. 2012.

[22] S. Blažič, I. Škrjanc, and D. Matko, "A robust fuzzy adaptive law for evolving control systems," *Evolv. Syst.*, vol. 5, no. 1, pp. 3–10, Mar. 2014.

[23] S. Formentin, P. De Filippi, M. Corno, M. Tanelli, and S. M. Savaresi, "Data-driven design of braking control systems," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 1, pp. 186–193, Jan. 2013.

[24] Z. C. Johanyák, "A modified particle swarm optimization algorithm for the optimization of a fuzzy classification subsystem in a series hybrid electric vehicle," *Tech. Vjesnik – Tech. Gaz.*, vol. 24, no. 2, pp. 295–301, June 2017.

[25] L. Fozo, R. Andoga, K. Beneda, and J. Kolesár, "Effect of operating point selection on non-linear experimental identification of iSTC-21v and TKT-1 small turbojet engines," *Period. Polyt. Trans. Eng.*, vol. 45, no. 3, pp. 141–147, Oct. 2017.

[26] H.-X. Lang, M. T. Khan, K.-K. Tan, and C. W. de Silva, "Application of visual servo control in autonomous mobile rescue robots," *Int. J. Comput. Communic. Contro*l, vol. 11, no. 5, pp. 685–696, Oct. 2016.

[27] B. Kovács, G. Szayer, F. Tajti, M. Burdelis, and P. Korondi, "A novel potential field method for path planning of mobile robots by adapting animal motion attributes," *Robot. Auton. Syst.*, vol. 82, pp. 24–34, Aug. 2016.

[28] E. Osaba, X.-S. Yang, F. Diaz, E. Onieva, A. Masegosa, and A. Perallos, "A discrete firefly algorithm to solve a rich vehicle routing problem modelling a newspaper distribution system with recycling policy," *Soft Comput.*, vol. 21, no. 18, pp. 5295–5308, Sep. 2017.

[29] M. Gânsari and C. Buiu, "Software system integration of heterogeneous swarms of robots," *Control Eng. Appl. Informat.*, vol. 19, no. 3, pp. 49–57, Sep. 2017.

[30] P. Angelov and R. Yager, "A new type of simplified fuzzy rule-based systems," *Int. J. Gen. Syst.*, vol. 41, no. 2, pp. 163–185, Feb. 2012.

[31] R.-E. Precup, M. L. Tomescu, S. Preitl, E. M. Petriu, J. Fodor, and C. Pozna, "Stability analysis and design of a class of MIMO fuzzy control systems," *J. Intell. Fuzzy Syst.*, vol. 25, no. 1, pp. 145–155, Mar. 2013.

[32] R.-E. Precup, R.-C. David, and E. M. Petriu, "Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 527–534, Jan. 2017.

[33] S. Vrkalovic, T.-A. Teban, and I.-D. Borlea, "Stable Takagi-Sugeno fuzzy control designed by optimization," *Int. J. Artif. Intell.*, vol. 15, no. 2, pp. 17–29, Oct. 2017.

[34] G. Navarro, D. K. Umberger, and M. Manic, "VD-IT2, Virtual Disk cloning on disk arrays using a type-2 fuzzy controller," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1752–1764, Dec. 2017.

[35] I. Dumitrache, N. Constantin, and M. Drăgoicea, *Retele neurale: identificarea si conducerea proceselor*. Bucharest: Matrix Rom, 1999.

[36] A. Alique, R. E. Haber, R. H. Haber, S. Ros, and C. Gonzalez, "Neural network-based model for the prediction of cutting force in milling process. A progress study on a real case," in *Proc. 15th IEEE Intl. Symp. Intell. Control*, Patras, Greece, 2000, pp. 121–125.

[37] M. Wagarachchi and A. Karunananda, "Optimization of artificial neural network architecture using neuroplasticity," *Int. J. Artif. Intell.*, vol. 15, no. 1, pp. 112–125, Mar. 2017.

[38] J. Saadat, P. Moallem, and H. Koofigar, "Training echo state neural network using harmony search algorithm," *Int. J. Artif. Intell.*, vol. 15, no. 1, pp. 163–179, Mar. 2017.

[39] V. Fioriti and M. Chinnici, "Node seniority ranking in networks," *Stud. Informat. Control,* vol. 26, no. 4, pp. 397–402, Dec. 2017.

[40] C. M. Ionescu, "Phase constancy in a ladder model of neural dynamics," *IEEE Trans. Syst. Man Cybern. A., Syst. Humans*, vol. 42, no. 6, pp. 1543–1551, Nov. 2012.

[41] C. M. Ionescu, D. Copot, and R. De Keyser, "Respiratory impedance model with lumped fractional order diffusion compartment," *IFAC Proc. Vol.*, vol. 46, no. 1, pp. 260–265, Feb. 2013.

[42] D. Copot, R. De Keyser, E. Derom, M. Ortigueira, and C. M. Ionescu, "Reducing bias in fractional order impedance estimation for lung function evaluation," *Biomed. Signal Process. Control*, vol. 39, pp. 74–80, Jan. 2018.