Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 51-6 (2018) 354–359

Fuzzy Classification of Hand's Motion

Lukas Peter * Filip Maryncak * Antotnino Proto * Martin Cerny *

* VSB-Technical University of Ostrava, Department of Cybernetics and Biomedical Engineering, Ostrava 70800, Czech Republic, (e-mail:lukas.peter@vsb.cz)

Abstract: The goal of this work was to create the measurement circuit that would be able to measure and classify signal of myopotentials to classify specific gestures of hand.Realization of the system for classification of hand's gestures is described in this paper. Hardware prototype of four measuring channels was created by combination of 2nd order filters and right amount amplification. For digitizing the data, the Arduino Nano microcontroller was selected and programmed using defined communication protocol. The computer software is programmed in C# programming language. Signal processing and drawing to user interface is in real time. The one of five possible gestures that user made is chosen using fuzzy logic and designed system of scaling.

© 2018, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: electromyography, fuzzy, classiffication of myopotentials, hands gestures.

1. INTRODUCTION

Nowadays the wearable technologies are trying to simplify everyday activities of their users are trying to collect data during the day and night. Under the continuous collection of data, you can imagine a smart watch, that measure your heart rate, count your steps and based on that data evaluates your daily activity. But what you could control applications in your computer or mobile phone just by making a gesture of your hand? This is exactly the subject matter of this paper (Merletti and Parker, 2004).

First of all, the genesis of myopotentials must be understood. It is a complex system that includes both neural and muscular system. The theoretical schematic for measuring myopotentials was designed based on this knowledge. The design was then used for creating PCB to measure myopotentials (Svecova et al., 2017). Also the galvanic isolation between users and the power source must be created because of usage of active reference electrode (Merletti and Farina, 2016).

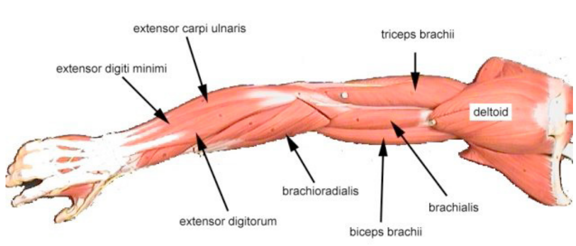

Fig. 1. Muscles of the forearm. For one movement are used two and more muscles. With right position of electrodes for measuring EMC is possible to recognize what kind of muscles are used for each movement.

With fully functional prototype for measuring myopotentials, it could now be possible to create algorithm for gesture classification. The communication protocol was created between Arduino Nano, that used for digitizing the signal and octet stuffing, and computer. That the algorithm for adaptive segmentation was designed. Adaptive segmentation allows to detect differences in both amplitude and frequency in real time for multiple channels. After the right gesture segmentation, the two phase gesture classification is used to determine the correct gesture (Benatti et al., 2015).

2. HARDWARE PROTOTYPE

The first major goal was to create prototypes with which it could be possible to measure myopotentials without noise and with right amplification (Chen et al., 2007).

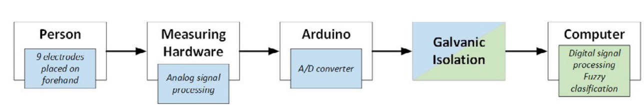

Fig. 2. Block scheme of measuring circuit. Myopotencial signals are measured and preprocessed by developed four channel analog hardware. There used one active electrode so galvanic isolation is needed. It was developed measuring software where was used fuzzy classification for recognition of gesture of hand's movement.

The myopotentials are measured from four channels. Each channel consists of two electrodes in differential mode, one instrumental amplifier, one high-pass filter, one low-pass filter, one notch filter with adjustable quality and one operational amplifier for final amplification (Slanina et al., 2017).

2.1 Galvanic Isolation

The whole prototype's board is powered by the Arduino Nano. Therefore the supply voltage is USB 5 V. The

2405-8963 © 2018, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2018.07.179

analogue part was necessary to be powered symmetrically, for this purpose was added a simple circuit to create ±2,5 V and virtual ground.

The reference electrode is connected to the virtual ground, therefore the galvanic isolation must be present while using the device. It is for safety of the user and also to meet the requirements of IEC 60601-1 (Boyali et al., 2015). The most convenient way is to isolate the whole USB on its way from computer (power source) to Arduino. The isolator ADuM4160 was used for data lines D+ and D-. Because the ADuM4160 did not provide enough power to supply the rest of the board, the DC-DC converter was used to isolate the power lines. The DC-DC converter provides an isolated power source to supply the prototype board and Arduino while the ADuM4160 provides galvanic isolation of the data lines.

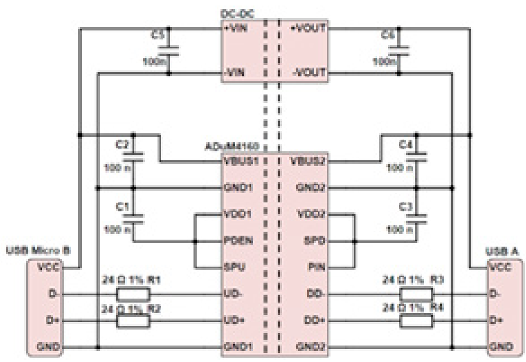

Fig. 3. Galvanic isolation.

2.2 The Instrumental Amplifier

The INA126 from Texas Instruments was used as instrumentation amplifier. It acts as a differential amplifier and has easily adjustable amplification.

The amplification was set on 9. Effect of polarization of electrodes was appearing with higher value of amplification.

2.3 Filtration

The myopotentials have a frequency spectrum between 20 Hz and 500 Hz so filtering the unneeded frequencies is in place. The topology of second order Sallen-Key was used for high-pass filter as well as for the low-pass filter.

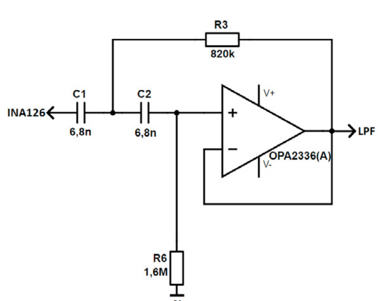

Fig. 4. Active high pass filter with cut-off frequency 20 Hz

The cut-off frequency was calculated thanks to equation 1.

$$f_c = \frac{1}{2\pi\sqrt{C1C2R3R6}} \quad (1)$$

2.4 Low Pass Filter

As mentioned before, the second order Sallen-Key topology was used for low-pass filter as well. The cut-off frequency was set close to 500 Hz (495 Hz). The difference is caused because of values of electric components (Phinyomark et al., 2012).

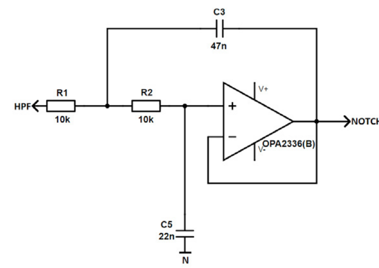

Fig. 5. Active low pass filter with cut-off frequency 500 Hz

The notch filter for 50 Hz must be used to reduce noise from electrical network and surroundings. Notch filter with very narrow frequency characteristic was used to preserve the most of the precious biological signal. It is a combination of two operational amplifier and twin T connection. It has also adjustable quality (Q) with the trimmer.

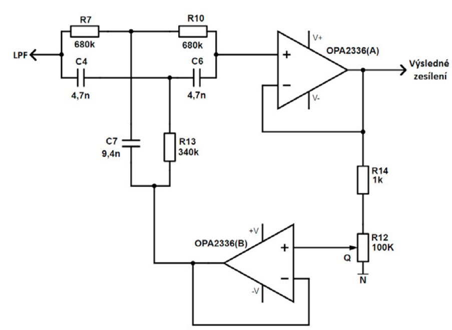

Fig. 6. Notch filter with cut-off frequency 50 Hz

2.5 Final Amplification

The final amplification was used to amplify to signal to get most of the resolution from A/D converter. It was used a classic non-inverting wiring. The amplification was set to $A = 341$. So the total amplification is 350 with a combination with instrumental amplifier.

The amplified biological signal goes to the Arduino Nano where is converted to a digital signal and sent to the computer via galvanic isolation (Tomczyński et al., 2015).

3. SOFTWARE

The software part consists of communication protocol between Arduino Nano and the computer, plotting the measured signal, adaptive segmentation of the gestures, calculating the features and finally two phase classification of the gestures.

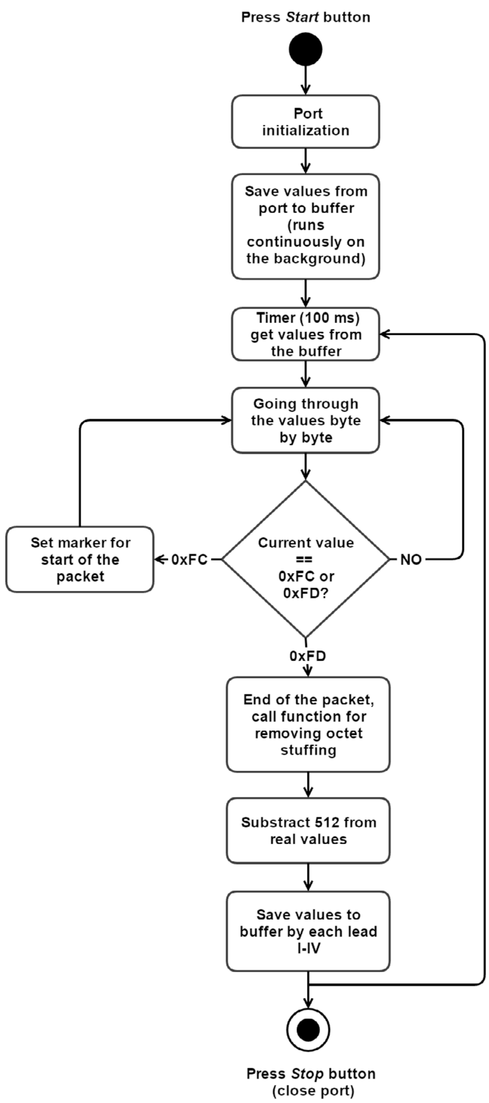

Fig. 7. Diagram of the data receiving.

Table 1. Octet stuffing

Unique values	**Octet stuffing**	**Meaning**
0xFC	0xFE 0xDC	start of packet
0xFD	0xFE 0xDD	end of packet
0xFE	0xFE 0xDE	octet stuffing mark

3.1 Communication Protocol

The signal is digitized by 10-bit A/D converter which is included in Arduino Nano. The values are continually read from analogue pins, checked for unique values and then put into packets. Three forbidden values are used in packet to determine where packet starts, where packet ends and to mark the octet stuffing.

The unique values can be situated in specified positions in the packed only. Based on this knowledge, it is possible to determine the start of the packet and the end of the packet when processed in the computer.

When the COM port is successfully opened, the incoming measured data starts streaming into the PC. The octet stuffing is removed in the first place. Then the values are divided into four channels as per connection on the prototype board. The measured signal is plotted in real time as shown in Fig. 10.

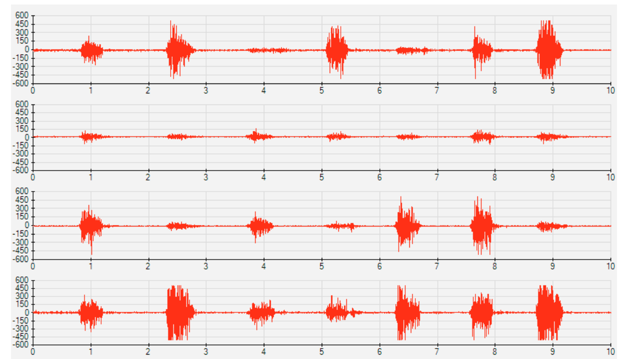

Fig. 8. Plotted signal from four channels of EMG measurement. Myopotencials were measured from four muscles. It can be seen that each movement of the hand activates different muscles.

3.2 Adaptive Segmentation

One of the most crucial part was to know when and where did the gesture occurred in the signal. This was what adaptive segmentation was used for.

Method of adaptive segmentation was dividing the signal into quasi-stationarity segments of variable length, depending on the occurrence of non-stationarities in the signal (diagram of the segmentation of signal can be seen on the figure 10). The key factors for choosing the right method of adaptive segmentation were:

- Fast algorithm
- High precision
- Multiple channel segmentation

Based on the key factors, the algorithm using two connected windows and detecting differences of amplitude and frequency was used. Two windows are moving along the signal in each channel. In each window the differences are computed. Amplitude and frequency difference computing was based on equation 2 for Amplitude difference and on equation 3.

$$ADIF = \sum_{i=1}^{WL} |x_i| \tag{2}$$

$$FDIF = \sum_{i=1}^{WL} |x_i - x_{i-1}| \tag{3}$$

Where WL is window length and it was set to 400 samples. Combining equation 2 and equation 3 the total difference was calculated:

$$DIF = 1 \cdot |ADIF_1 - ADIF_2| + 7 \cdot |FDIF_1 - FDIF_2| \tag{4}$$

If the difference is higher than the calculated threshold, the segment border is marked on the place of local maximum of the difference.

$$THR = \frac{1}{BL} \cdot DIF \tag{5}$$

Where BL is size of incoming data or the number of values in current sample.

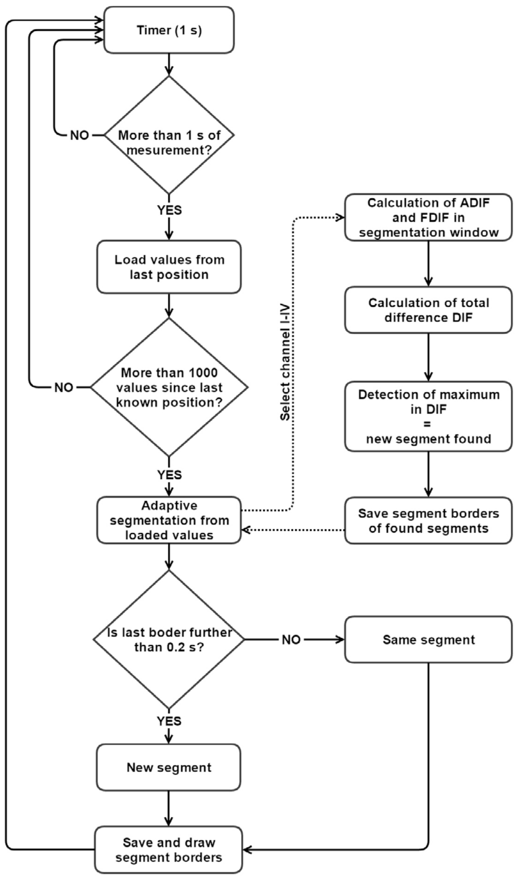

Fig. 9. Diagram of the segmentation of the measured signal signal.

3.3 Feature Extraction

Three features were chosen to be extracted from each segment window and they are Root Mean Square (RMS), Logarithmic Band Power (LBP) and first derivation (DIFF). The RMS is calculated by:

$$x_{RMS} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \tag{6}$$

The LBP is calculated by:

$$x_{LBP} = log(1 + \frac{1}{N}\sum_{i=1}^{N} x_i^2) \tag{7}$$

And first derivation is calculated by:

$$x_{DIFF} = \frac{1}{N}\sum_{i=1}^{N} |x_{i-1} - x_i| \tag{8}$$

These three features ale calculated for each channel. So there are twelve feature values for each detected segment.

If there is not find segments based on described algorithm in some channel it is automatically compute as:

- Compare segments from all of rest channels for one gesture
- Evaluate the beginnings of the segments
- Evaluate the ends of the segments

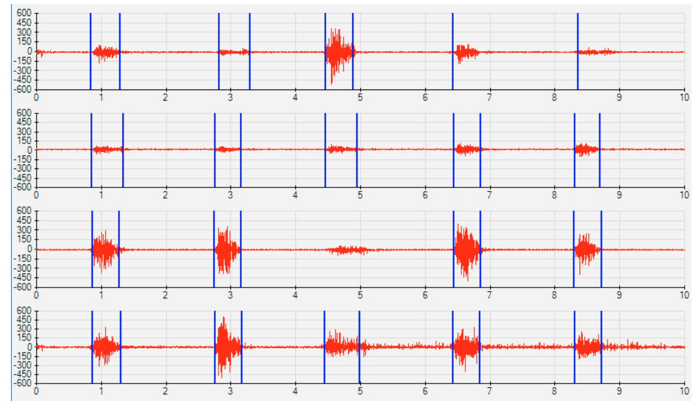

Fig. 10. Plotted segmented signal. For the third gesture there is missing one of evaluated segment in third channel. In this case it is used algorithm based on comparison all of rest segments in same time to evaluate the beginning and the end of missing segment.

- Line up values of the beginnings
- Line up values of the ends
- Choose the first value from the beginnings (the earliest beginning)
- Choose the last value from the ends (the latest end)
- Use these values as segment values for the missing segment for one gesture

3.4 Gestures and Electrode Placement

The algorithm is programmed to classify five gestures (Fig. 11).

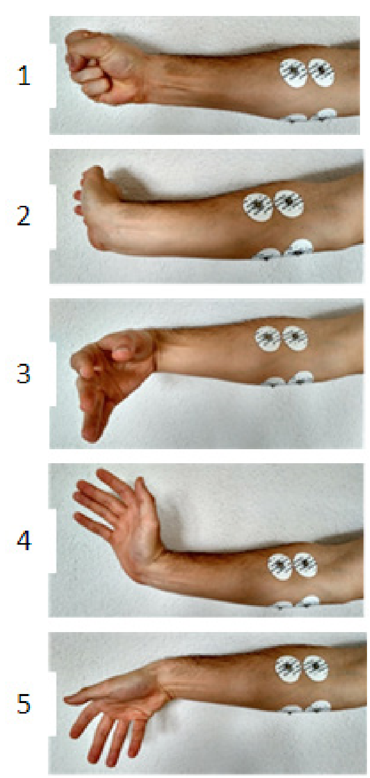

Fig. 11. Gesture demonstration. 1 - Fist; 2 - Right; 3 - Left; 4 - Up; 5 - Down

Each gesture has different values as well as features in a different channel. Based on that, the algorithm using fuzzy logic was designed for gesture classification.

3.5 Fuzzy Set Design

Because the first phase of classification is based on fuzzy logic, the fuzzy sets must be designed in the first place.

For each gesture 102 instances were measured. All three features were calculated for each instance and from these features, the standard deviation and expected value is calculated. Using these two parameters for each gesture and feature in each channel, twelve fuzzy sets were designed (4 channel x 3 features).

Each fuzzy set contains the member function of all five gestures. These member functions are based on expected value standard deviation.

After the member functions are cast into the fuzzy set, the fuzzy set is adjusted so everywhere in the fuzzy set the sum of the member function at certain point equals 1.

Fig. 12. Adjusted fuzzy set.

3.6 Results

The adaptive segmentation and plotting the graph runs smoothly as well as the classification when it comes to computing memory. It uses only a small amount of processor and RAM.

The gesture classification can be described in diagram of gesture classification (Fig. 13).

Segment borders are plotted in real time directly on the measures signal. The classification is also done and shown in real time, right after the gesture is done. There is an option to save the raw values in.Csv format for further analysis. There is also an option to save just two features from finding segments (Fig. 14).

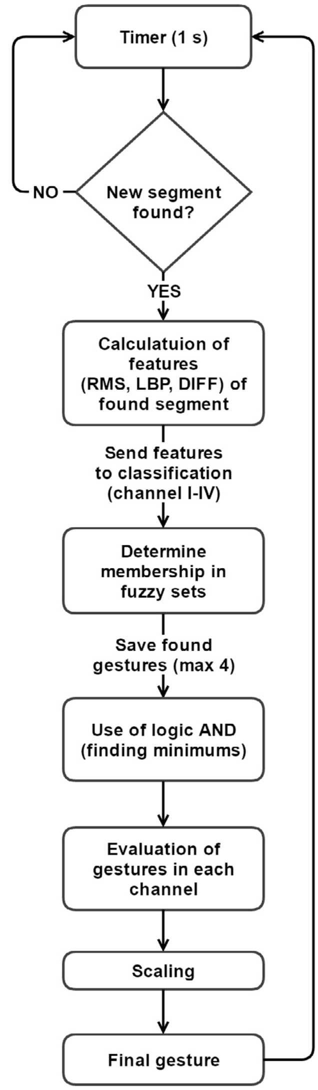

Fig. 13. Diagram of the gesture classification.

Fig. 14. Record of real time segmentation and gesture classification.

ACKNOWLEDGEMENT

The work and the contributions were supported by the project SV4506631/2101 'Biomedicínské inženýrské systémy XII'.

4. CONCLUSION

The goal of this work was the creation of the fuzzy logic algorithm to classify gesture of hand movements. This was a completely experimental creation of a new classification algorithm. The basis for gesture recognition is segmentation, that is, the recognition that there was a gesture in the signal. For this purpose, the adaptive segmentation method was used using two connected windows. These two linked windows float after the signal and look for changes in the difference that is higher than the threshold. This algorithm works perfectly and uses only a very small percentage of the operating performance. The adaptive segmentation and plotting the graph runs smoothly as well as the classification when it comes to computing memory. It uses only a small amount of processor and RAM.

Segment borders are plotted in real time directly on the measures signal. The classification is also done and shown in real time, right after the gesture is done. There is an option to save the raw values in.Csv format for further analysis. There is also an option to save just two features from finding segments.

The algorithm was tested by counting the successful classification for each gesture. Each gesture was performed 40 times.

The results shown, that the Fist has 100% successful rate of classification. Next most successful gesture is Left with 85% rate. Other gestures are around 60%. This can be caused by too much overlap in fuzzy sets. Overall success rate 73% is sufficient at this first experimental stage.

There are some considerable options that can improve further success rate of classification.

By using more electrodes, we get more values that can define the gesture. More values mean more data to work with but it also means higher demands on computing power.

By classifying less gestures, the fuzzy set is going to overlap less, so it will definitely increase accuracy but in the cost of less gestures.

This project was done only for one individual. The input data for fuzzy sets as well as the testing. If would be more data from more individuals collected, the accuracy of classification could improve.

By using a different algorithm such as artificial neural network, fuzzy k-NN classification or Bayes theorem.

The accuracy of the classification is depending more on the software part than the hardware part. So the future of this work should focus more improving the classification algorithm than the hardware prototype.

REFERENCES

Simone Benatti, Filippo Casamassima, Bojan Milosevic, Elisabetta Farella, Philipp Schönle, Schekeb Fateh, Thomas Burger, Qiuting Huang, and Luca Benini. A versatile embedded platform for emg acquisition and gesture recognition. *IEEE transactions on biomedical circuits and systems*, 9(5):620–630, 2015.

Ali Boyali, Naohisa Hashimoto, and Osamu Matsumoto. Hand posture and gesture recognition using myo armband and spectral collaborative representation based classification. In *Consumer Electronics (GCCE), 2015 IEEE 4th Global Conference on*, pages 200–201. IEEE, 2015.

Xiang Chen, Xu Zhang, Zhang-Yan Zhao, Ji-Hai Yang, Vuokko Lantz, and Kong-Qiao Wang. Multiple hand gesture recognition based on surface emg signal. In *Bioinformatics and Biomedical Engineering, 2007. ICBBE 2007. The 1st International Conference on*, pages 506–509. IEEE, 2007.

Roberto Merletti and Dario Farina. *Surface electromyography: physiology, engineering and applications*. John Wiley & Sons, 2016.

Roberto Merletti and Philip A Parker. *Electromyography: physiology, engineering, and non-invasive applications*, volume 11. John Wiley & Sons, 2004.

Angkoon Phinyomark, Sirinee Thongpanja, Huosheng Hu, Pornchai Phukpattaranont, and Chusak Limsakul. The usefulness of mean and median frequencies in electromyography analysis. In *Computational intelligence in electromyography analysis-A perspective on current applications and future challenges*. InTech, 2012.

Zdenek Slanina, Sarka Mikolajkova, and David Vala. Human vehicle interaction. In *AIP Conference Proceedings*, volume 1836, page 020050. AIP Publishing, 2017.

Lucie Svecova, David Vala, and Zdenek Slanina. Emg as objective method for revealed mistakes in sport shooting. In *International Conference on Intelligent Information Technologies for Industry*, pages 439–448. Springer, 2017.

Jakub Tomczyński, Piotr Kaczmarek, and Tomasz Mańkowski. Hand gesture-based interface with multichannel semg band enabling unknown gesture discrimination. In *Robot Motion and Control (RoMoCo), 2015 10th International Workshop on*, pages 52–57. IEEE, 2015.