Application of Genetic Algorithm for Optimization of Data in Surface Myoelectric Prosthesis for the Transradial Amputee

Jumelyn L. Torres and Noel B. Linsangan

***Abstract*—Genetic algorithm (GA) is a method that can be used to discover and manage a population of useful patterns in which this study implements; specifically, in optimization. This algorithm is a powerful tool to find the best solution in problems such as prediction and data fitting due to its ability for fast adaptation in the problem environment. Continuous or discrete parameters can be optimized by GA even without requiring derivative information by simultaneously searching from a wide sampling of the cost surface even if it deals with large number of parameters. The paper makes use of this algorithm to optimize the surface electromyography (SEMG) signal from the skeletal muscle force of a transradial amputee in controlling a surface myoelectric prosthesis. The SEMG signals patterns are acquired from the two devices: the microcontroller unit and the EMG simulator. The signals from these two devices are processed and optimized using GA. The optimized signal is used to test the surface myoelectric prosthesis. Moreover, the data acquired from these signals is treated using t- test to show the significant difference of their means.**

***Index Terms*—Genetic algorithm, optimization, surface electromyography signal (SEMG), surface myoelectric prosthesis, T-test.**

I. Introduction

Surface myoelectric prostheses are prosthetic devices that are being controlled by surface electromyography (SEMG) signal which is an electrical manifestation of skeletal muscle contractions [1] produced naturally by extant muscle tissue. The SEMG signal is the comprehensive effect of the activity of neural stem and shallow muscle. It reflects functions of the nerve and muscle which can be recorded through the surface electrodes from the surface of the human skeletal muscle [2]. This signal provides valuable information that has been essential to its application in clinical diagnosis and as a source for controlling assistive devices, and schemes for functional electrical stimulation such as myoelectric prostheses.

There are some existing studies (Fukuda *et al*. 2003) that have been conducted on using SEMG signals to control prosthetic hands. According to these studies that, in many cases, some parts of the muscles remain near the amputated part and the EMG signals measured from them can be used as a control signal to the prostheses. It has been stated in the studies that the SEMG patterns have nonlinear and non-stationary characteristics. Moreover, that SEMG patterns can change in accordance to the differences of individuals, different locations of electrodes, and time variation caused by fatigue or sweat [3]. Using SEMG signals, skeletal muscle force can be estimated to control the myoelectric prostheses. Kumar stated that generally EMG signal amplitude increases with an increase in the skeletal muscle force; however, various factors affect this relationship. Since varying motor unit recruitments, crosstalk, and biochemical interaction within the muscular fibres contribute to generate EMG signals, hence EMG signals are random, complex and dynamic in nature [4], so there is a need that their patterns must be classified or discriminated to have a smooth and accurate operation on the prostheses. To address such issues of the SEMG signals, genetic algorithm (GA) can be applied. Liu *et al*. stated that GA is considered to be an effective search and optimization method (2011) [5]. It is a powerful tool because of its ability for fast adaptation in the problem environment [6].

The study designed and implemented SEMG signals in controlling a myoelectric prosthesis for a transradial amputee and utilized GA tool in Matlab to optimize the acquired data of the surface EMG signal patterns. Using Matlab has been found to be quite acceptable to students in learning GA since it can be accessed conveniently and provides many toolboxes such as GA tool to support an interactive environment for modeling and simulating a wide variety of dynamics systems [7]. The GA tool in Matlab is a graphical user interface that enables the user to use GA without working at the command line [8]. This tool provides the basic process of a genetic algorithm such as creating the initial population, evaluating the fitness of the individuals, selecting the best individuals, applying the genetic operators such as crossover and mutation and providing the stopping criteria. It also provided statistical comparisons between the signal generated from a simulator and the signal generated from the signal processor through the surface electrodes by using the t-test analysis showing the significant difference of their means [9].

II. Methodology

Controlling the prosthesis must be précised so the need of discriminating the EMG signals is so important. The extracted patterns of the signals were optimized to give more precise movements of the prosthesis. To address the challenge, we conducted the following methods which

Manuscript received November 13, 2013; revised March 14, 2014. This work has been supported by Engineering Research and Development for Technology (ERDT) Scholarship Grant of Department of Science and Technology (DOST), Philippines.

The authors are with the Computer Engineering Department at the Mapua Institute of Technology, Philippines (e-mail: jumelyn.torres@yahoo.com, nblinsangan@mapua.edu.ph).

include the hardware and software designs and the optimization process using the GA tool of Matlab in which this study focused more.

A. Hardware Design

Fig. 1 shows the block diagram for the hardware design. The knob potentiometer, one each to operate each finger of the hand and form the hand movements, serves as the input signals to be processed in the microcontroller unit then is converted to analog signal which is displayed on the monitor. Another input signals are SEMG signals generated from the surface electrodes. These signals are the one we optimized using the genetic algorithm.

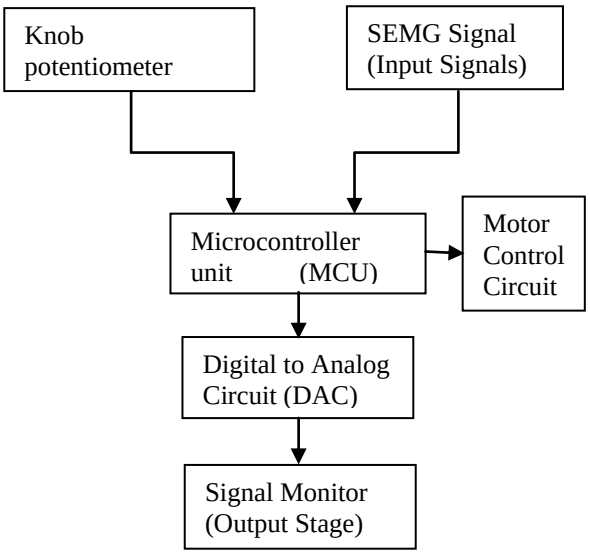

Fig. 1. Block diagram of hardware design.

B. Software Design

Fig. 2 shows the development of the software design. It involves the extraction of SEMG signal patterns to classify the hand movements such as close (hold) position, open (normal). The signals generated from the surface electrodes are optimized using the GA Tool in Matlab.

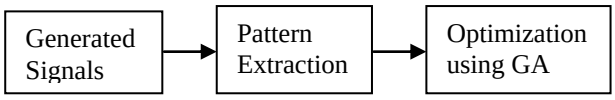

Fig. 2. Block diagram of software design.

C. Optimization Process

The optimization process using GA, shown in Fig. 3, basically involves as follows:

1) *creation of initial population*: random population should be generated to serve as the initial population
2) *evaluation of fitness of individuals*: function value should be defined to evaluate the indiviadual fitness
3) *selection*: best individuals should be selected as parents that will contribute to the population of the next generation
4) *application of genetic operators such as crossover and mutation*: crossover point and mutation point must be generated to produce a new generation or a new individual
5) *providing stopping criteria*: termination condition/s must be determined

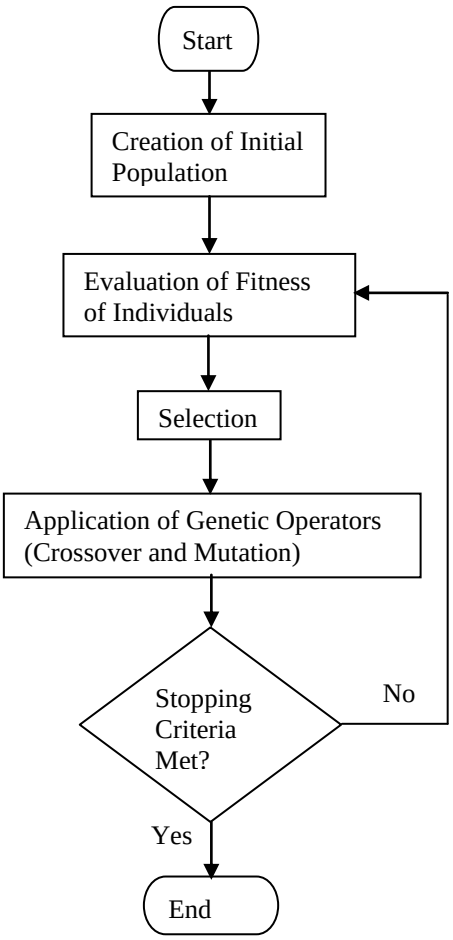

Fig. 3. Flowchart of the optimization process using genetic algorithm.

III. Results and Discussion

The generated signal from the signal processor using Independent Component Analysis (ICA) (Lazaro, 2013) [10] and the simulated signals using data analysis such as the poly or linear regression are plotted using Microsoft Excel, which are represented by the dashed lines and smooth line respectively, shown in the figures below.

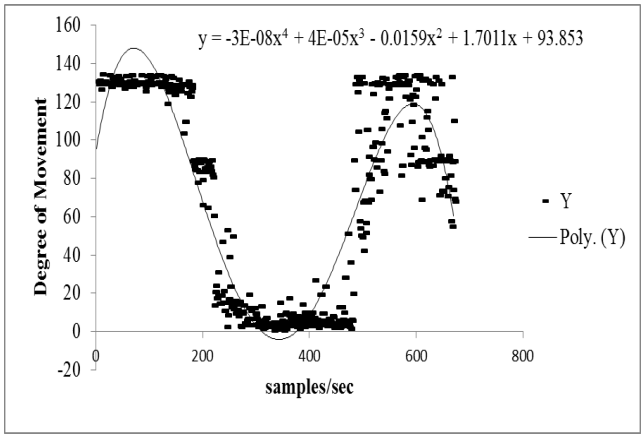

Fig. 4. Signal for hold position.

Fig. 4 shows the signal patterns for hold position. Equation (1) is the function of the simulated signal which is used as fitness function for hold position; it is derived using poly regression in data analysis of Excel.

$$y = -3E\text{-}08x^4 + 4E\text{-}05x^3 - 0.0159x^2 + 1.7011x + 93.853 \quad (1)$$

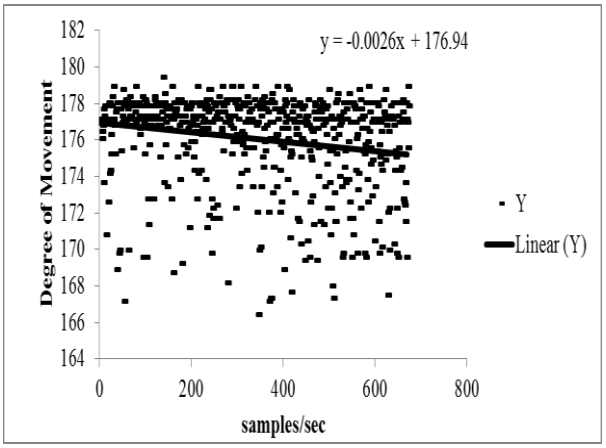

Fig. 5. Signal for open position.

Fig. 5 shows the signal patterns for an open position. Equation (2) is the function of the simulated signal which is used as fitness function for open position. Using EXCEL data analysis, linear regression is chosen over poly regression for the open (normal) position.

$$y = -0.0026x + 176.94 \qquad (2)$$

Defining the necessary fitness function for each signal is important to evaluate the individuals in a population. It is used to select individuals for reproduction [11].

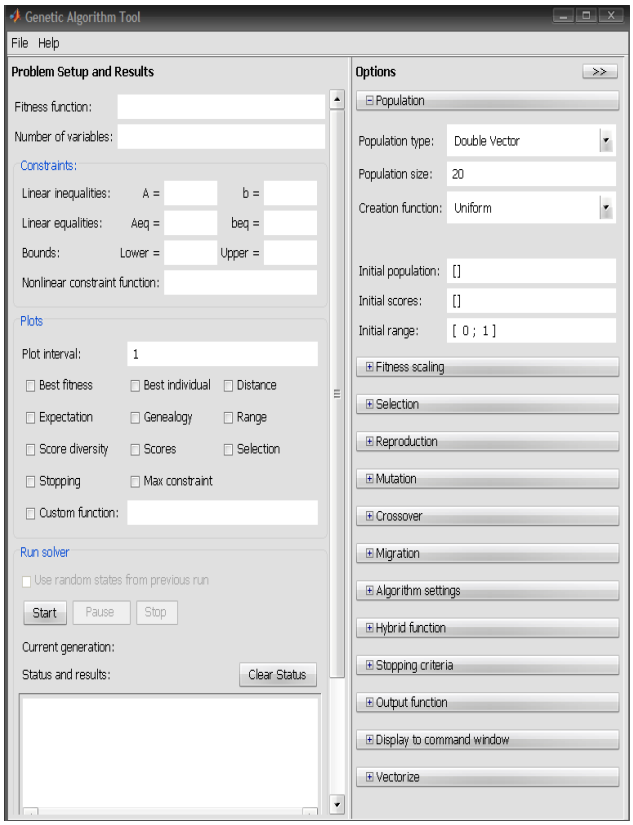

Fig. 6. GA tool solver in Matlab.

Utilizing the GA Tool Solver in Matlab as shown in Fig. 6, after parameters are set; the fitness values of a hold signal vs number of generations are generated as shown in Fig. 7 and as well as of the open signal shown in Fig. 8. The points above of the plot denote the averages of the fitness values in each generation, while the points at the bottom denote the best fitness values. The plot also displays the best and mean values in the current generation numerically at the top. As the generation leads to termination, the changes in the best fitness values slow down since it's getting closer to the optimal point.

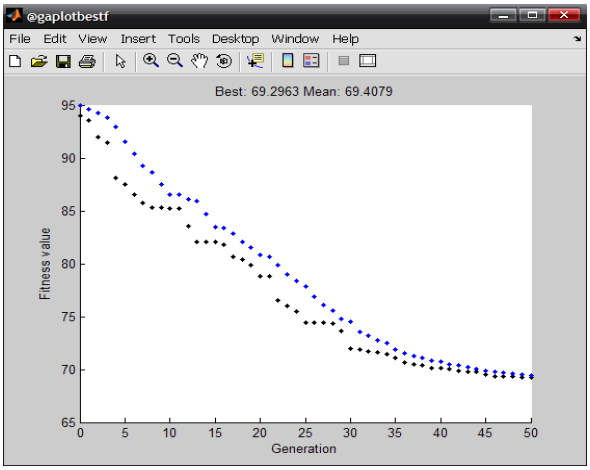

Fig. 7. Best Fitness of hold signal.

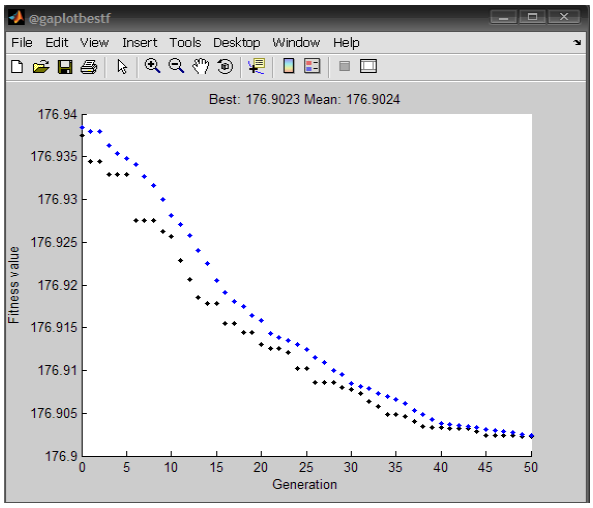

Fig. 8. Best Fitness of open signal.

Running the solver again but changing the Hybrid Function from none to unconstrained minimization function (fminunc); another set of fitness values per generation is shown in Fig. 9 and Fig. 10 for hold signal and open signal, respectively. This function uses the final point from the first run that is close to the optimal point and uses it as the initial point for fminunc.

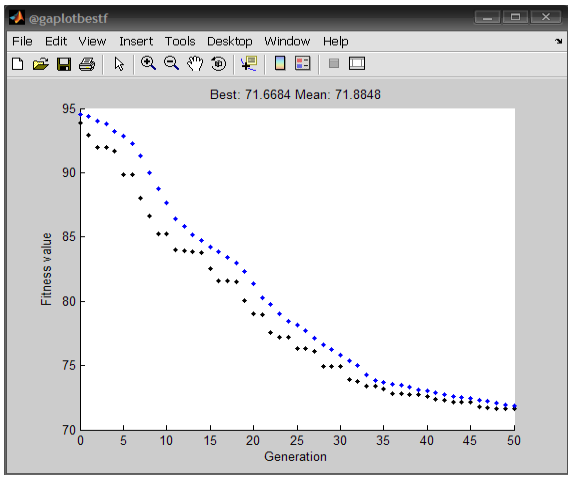

Fig. 9. Best Fitness of hold signal with hybrid function.

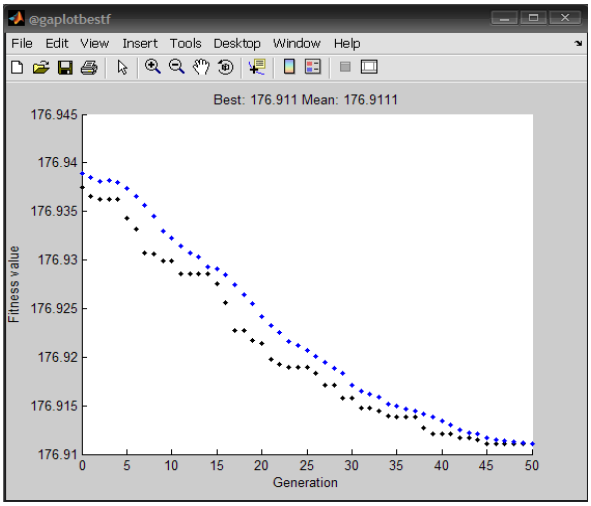

Fig. 10. Best fitness of open signal with hybrid function.

TABLE I: DATA FOR HOLD SIGNAL

Iterations	Signals	
	Generated EMG Signal (X_1)	*Optimized Signal* (X_2)
1	130.343	92.9061
2	129.1557	92.002
3	129.1557	92.002
4	131.2665	91.6567
5	129.8153	89.811
6	129.8153	89.811
7	126.5172	88.0163
8	130.2111	86.645
9	134.1689	85.2596
10	129.0237	85.2596
11	129.0237	83.9665
12	129.5515	83.922
13	129.9472	83.8604
14	129.2876	83.7984
15	130.343	82.5559
16	129.1557	81.5604
17	129.5515	81.5603
18	129.0237	81.5334
19	129.6834	80.0484
20	129.0237	79.0257
21	129.0237	78.9822
22	133.905	77.5859
23	129.1557	77.2268
24	129.1557	77.2268
25	127.7045	76.3032
26	127.7045	76.3032
27	129.6834	76.0752
28	132.058	74.9375
29	129.1557	74.9375
30	129.0237	74.9374

TABLE II: DATA FOR OPEN SIGNAL

Iterations	Signals	
	Generated EMG Signal (X_1)	*Optimized Signal* (X_2)
1	177.1529	176.937
2	176.4499	176.936
3	176.0984	176.936
4	176.8014	176.936
5	177.6801	176.934
6	173.638	176.933
7	177.3286	176.931
8	177.8559	176.931
9	177.1529	176.93
10	177.1529	176.93
11	177.1529	176.929
12	170.826	176.929
13	176.9772	176.929
14	176.9772	176.929
15	176.9772	176.928
16	172.5835	176.926
17	178.0316	176.923
18	178.0316	176.923
19	174.1652	176.922
20	174.3409	176.921
21	176.2742	176.92
22	175.2197	176.919
23	177.8559	176.919
24	178.3831	176.919
25	177.1529	176.919
26	177.1529	176.918
27	177.1529	176.917
28	178.9104	176.917
29	176.8014	176.916
30	175.2197	176.916

Using the t-test analysis provided by the Microsoft EXCEL, the data on Table I and Table II in which the generated EMG signal serves as the control group and the data from the optimized signal as the experimental group are utilized. The t-results are derived with 0.05 as level of significance which is also called as alpha. When p-value is greater than the alpha then the null hypothesis is not rejected otherwise it is rejected [8]. For this study, the null hypothesis is $\overline{X_1} = \overline{X_2}$ or the difference between the two means is zero, meaning there is no significant difference between the means of these two sets [12].

The *p*-value of the data for hold signal resulted to 1.15939E-27 which is less than 0.05. Therefore, the null hypothesis is rejected which mean there is a significant difference.

The *p*-value of the data for open signal resulted to

0.149835047 which is greater than 0.05. Therefore, the null hypothesis is not rejected. There is no significant difference between the generated EMG signal and the optimized signal, showing that there are no changes at all even if it is optimized. Based on the graph shown in Fig. 5, the values are just almost confined in points 175 and 176.

The significant difference showed how the optimization changed the patterns of the signal for the hold position. However, in the open signal, the optimization has of no great effect since the patterns are too close. The average distance between individuals determines the diversity of the population. If the average distance is large, then the diversity is high. Otherwise, the diversity is low. Diversity of the population affects the performance of the genetic algorithm like what happened in the data of the open position [7].

IV. Conclusion

The study significantly improved the operation of the prosthesis by using the Genetic Algorithm Solver in Matlab. The generated signals from the prototype has been optimized by creating an initial population; evaluating the fitness of the individuals; selecting the parents to be used for the next generation; applying crossover, mutation and hybrid functions and providing the stopping criteria. The best fitness data showed how its value improved in one generation after another. In the early generations, its values rapidly improved as the individuals are farther from the optimal point but slow down as the populations become closer the optimal point.

Appendix

The following procedures indicate and show how to optimize a function by using Genetic Algorithm (GA) Tool of Matlab:

1) Before using the GA Tool, create first the M-file of the function to be optimized. Fig. 11 shows an example of an M-file for the function to be optimized.

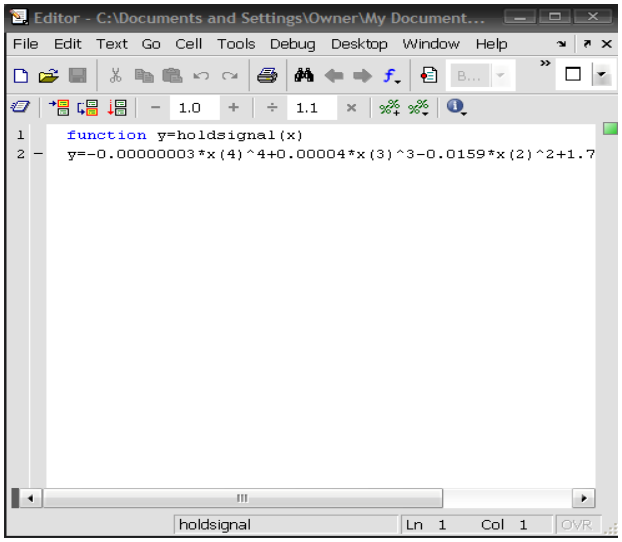

Fig. 11. An example of M-file.

2) The created M-file of the function to be optimized is used as the fitness function that serves as the object function. Number of variables must be deternined also before running the GA solver. Fig. 12 shows the genetic algorithm tool of Matlab.

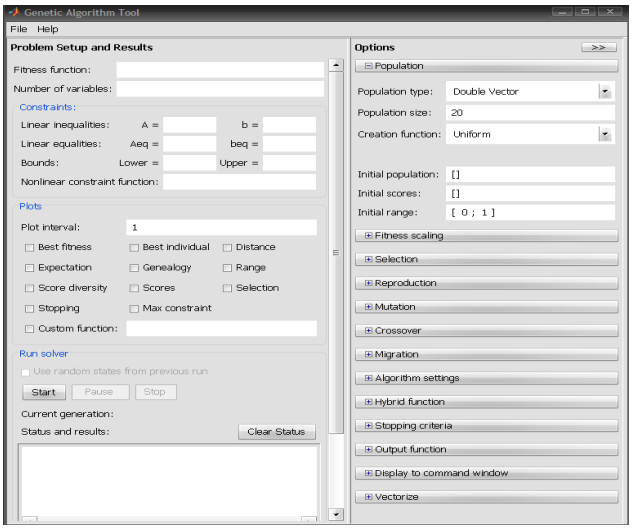

Fig. 12. Genetic algorithm tool of Matlab.

3) Next, enter the necessary inputs such as the fitness function using @before the filename of the M-file created for a specific function. The default conditions or parameters used such as population size, selection, mutation, crossover, hybrid function and stopping criteria are shown in Fig. 13 and Fig. 14. The paramaters can be altered as to satisfy whatever the user desires for the optimization of the function.

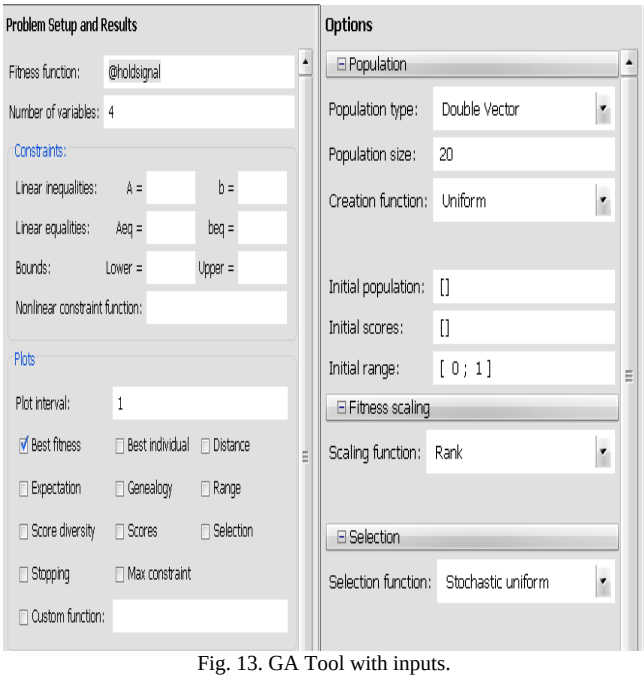

Fig. 13. GA Tool with inputs.

4) After setting all the necessary inputs in the GA Tool, the optimization is started by clicking the start button in the Run Solver. To show the plots of the function, just click any of the plots provided by the tool such as Best fitness and to show the generated values, click the History to new window in the Output function and choose any of those options for level of display in the Display to command window such as iterative as shown in Fig. 14. As the GA terminates the results are generated like the one shown in Fig. 15.

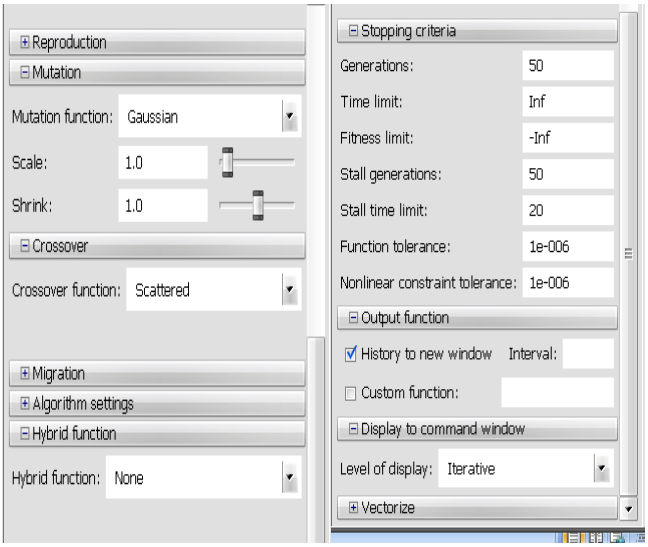

Fig. 14. GA tool with inputs.

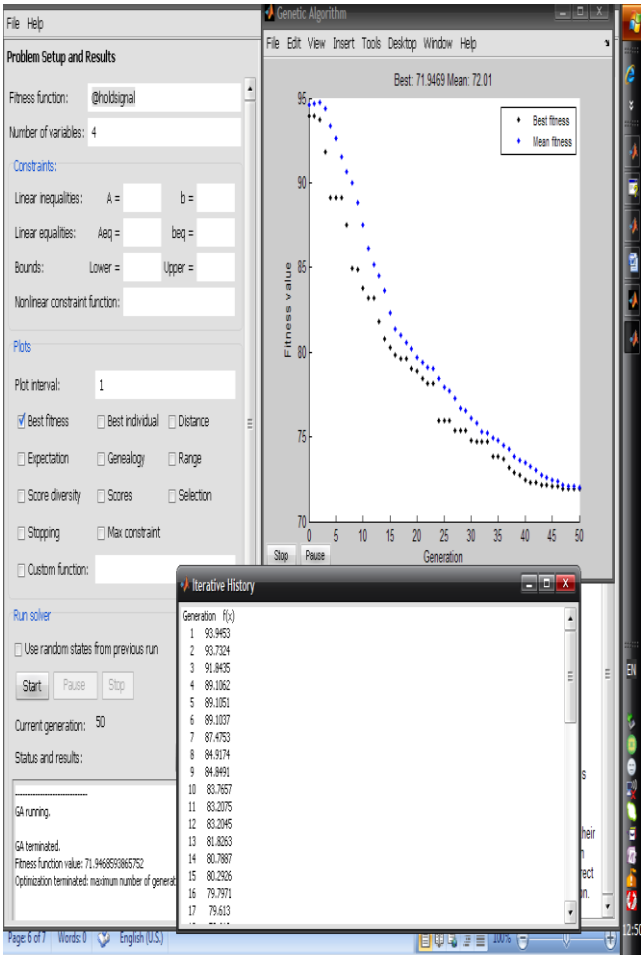

Fig. 15. Sample results of the GA solver.

ACKNOWLEDGMENT

The authors are grateful to all the people who took part in completing this work. Special thanks to Jose Lazaro, Jr. for his help with data collection. This work was supported by the Engineering Research and Development for Technology (ERDT) Scholarship Grant of the Department of Science and Technology (DOST), Philippines.

REFERENCES

[1] H. Huang, H. Xie, J. Guo, and H. Chen, "Ant colony optimization-based feature selection method for surface electromyography signals classification," *Comput. Biol. Med.*, pp. 1-9, 2011.

[2] Z. Xishi, "Study of surface electromyography signal based on wavelet transform and radial basis function neural network," *IEEE International Seminar on Future Biomedical Information Engineering*, vol. 38, pp. 160-163, 2008.

[3] O. Fukuda, T. Tsuji, M. Kaneko, and A. Otsuka, "A human-assisting manipulator teleoperated by EMG signals and arm motions," *IEEE Tran. Robot. Autom.*, vol. 19 no. 2, pp. 210 -222, Apr. 2003.

[4] P. Kumar *et al.*, "A hybrid adaptive data fusion with linear and nonlinear models for skeletal muscle force estimation," *IEEE 5th Cairo, Int. Conf. on Biomed. Eng.*, December 16-18, 2010, pp. 9-12.

[5] G. Liu and J. Chen, "The application of genetic algorithm based on matlab in function optimization," *IEEE Int. Conf. on Electrical and Control Engineering*, September 16-18, 2011, pp. 5034-5037.

[6] M. Borkowski, "Application of genetic algorithm to pattern extraction," *IEEE 5th Int. Conf. on Intelligent Systems Design and Applications*, September 8-10, 2005, pp. 222-227.

[7] Y. J. Cao and Q. H. Wu, "Teaching genetic algorithm using matlab," *Int. J. Elect. Enging. Educ.*, vol. 36, pp. 139-153, 1999.

[8] *Genetic Algorithm and Direct Search Toolbox User's Guide*, The Mathworks, Inc., 2004.

[9] L. Paler-Calmorin and M. Calmorin, *Methods of Research and Thesis Writing*, pp. 135-144, 1995.

[10] J. Lazaro Jr., "Surface electromyography signal for control of myoelectric prosthesis of the upper-Limb using independent component analysis," 2013.

[11] R. Malhotra, N. Singh, and Y. Singh, "Genetic algorithms: Concepts, design for optimization of process controllers," *Computer and Science*, vol. 4 no. 2, pp. 39-54, March 2011.

[12] J. McClave, P. G. Benson, and T. Sincich, *Statistics for Business and Economics*, 8th Ed., pp. 394-445, 2001.

Jumelyn L. Torres is a faculty member of the Computer Engineering (CpE) Department at Mapua Institute of Technology, Manila, Philippines. From this institution, she finished bachelor of science in electronics and communications engineering (BSECE), graduated as a silver medalist in 2000 and also finished master in science in computer engineering (MSCOE) in 2013. She joined the teaching industry in 2002 right after passing the licensure exam for BSECE.

Noel B. Linsangan is the present program chair of the Computer Engineering (CpE) program at the Mapúa Institute of Technology, Manila, Philippines. Mapúa is the largest engineering school in the country. He obtained his baccalaureate degree in computer engineering from the same school in 1988 and his graduate degree, Master in engineering in COMPUTER Engineering from the Pamantasan ng Lunsodng Maynila (University of the City of Manila) in 2000. Right after graduation from college he joined the faculty of the computer Engineering program and in 2000 he was appointed as its program chair. He is part of the team that pioneered the implementation of OBE in Mapúa and also one of the point persons of the Institute when its EE, ECE, and CpE programs were preparing for ABET accreditation.