International Journal of Engineering & Technology, 7 (4.36) (2018) 404-408

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Energy and Path Optimization of Robot Arm Simulator Via Multi-Objective Evolutionary Algorithm

Z. Mohamed[1,2]*, N.S. Khusaini [2,3], M.A.M Anuar[1,2], R. Ramly[1], M.A Anuar[1,2], K.S.M Sahari[4],

[1]Faculty of Mechanical Engineering Universiti Teknologi MARA, Shah Alam Selangor, Malaysia
[2] Sports Engineering & Artificial Intelligent Center, Faculty of Mechanical Engineering Universiti Teknologi MARA, Shah Alam Selangor, Malaysia
[3]Faculty of Mechanical Engineering, Universiti Teknologi MARA, Kampus Permatang Pauh, Pulau Pinang, Malaysia
[4]Department of Mechanical Engineering, Universiti Tenaga Nasional, Kajang Selangor, Malaysia
**Corresponding author E-mail: zulkifli127@salam.uitm.edu.my*

Abstract

The performance of robot arm motion generated via neural network are presented in this paper. The robot arm motion for obstacle avoidance simultaneously optimizing three functions; minimum distance, minimum time and minimum energy are generated. Four different initial and goal position had been chosen to test and analyze the performance of generated neural controller. The same neural controllers can be employed to a different range of initial and goal position. The motion generated yield good results in the simulator. In this research a new approach for intelligent robot arm path and motion generation are successfully implemented.

Keywords: *Robot arm, Genetic algorithm, Neural network, Multi-objective evolutionary algorithm.*

1. Introduction

Human being is said to have unique traits compared to other living creatures. Human being has the ability to think constructively, and fulfill daily needs by performing various kind of task, which may be complicated and need further scrutiny. However, as time goes on, smart humanoid robots are developed to do simple daily routines which are normally done by humans. For instance, taking order in the restaurant and as a receptionist in the hotel.

As the technology progresses, researchers are looking into possibilities to create humanoid robots with the capability and ability to mimic simple human movement. The movement is related to reaching and moving an object from one place to another. As easy as it may seem, there are certain requirements that should be taken into consideration. For instance, if the humanoid robot is programmed to perform simple movement such as to move an object, it needs to identify at least seven different elements namely shape, size, position, color (for object's identification purposes), as well as optimum distance, speed and energy (for the robot's hand movement). This is crucial to ensure smooth and accurate travelling path, especially, if there is a lot of obstacles around the chosen or specific object. Hence, there is a need to clearly determine the object's position, the robot's hand initial position (starting point), as well as the travelling path.

To ensure a successful implementation, motion generation characteristics such as optimum speed and distance, as well as obstacle avoidance should be clearly determined for the robot's hand to mimic or perform as how human's hand performed. Motion generation characteristics is the combination of two or more motion characteristics which will affect the robot's performance, depending on the requirements which is stated by the researcher.

The approach to the development of motion generation characteristics has been conducted in previous research using two different methods namely Rapid-Exploring Random Trees (RRT) and Rapid-Exploring Dense Trees (RDT). For RRT algorithm, the main focus is to optimize the degrees of freedom when the robotic arm named ARMAR-III is performing the given task [1]–[4]. On the other hand, RDT is related to fine-tuning the parameters to detect crash automatically until the final or best solution is obtained [3]. The finding has proven that the performance for RDT is superior than RRT.

In other works, [5] utilize a segmented positioning method and it had been implemented in an established service robot manipulator motion. Higher accuracy had been achieved by utilizing the robot vision system. A minimum time industrial robot arm motion generation had been proposed by [6]. In this study, the robot manipulators' dynamic model is utilized to regulate the maximum kinematic constraints. On the other hand, a single objective robot arm motion had been introduced by [7]. However, the minimum distance, minimum velocity and minimum acceleration objective function for robot arm motion generation had been implemented as separate entity.

Opportunely, a motion generation via multi-objective characteristics has been proposed by [8] and [9]. Five objective functions are considered for the study. The five objective functions are constructed to achieve minimum angular displacement, minimum, Cartesian distance, minimum angular velocity, minimum Cartesian velocity, as well as minimum energy. A planar robot is used to test the performance and capability of the proposed method. A similar approach is applied by [10] for generating parallel kinematics machine motion. Three different objective functions were chosen in their work; namely minimum shaking force, minimum

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

used of electric energy by the actuators, as well as maximum torque.

Interestingly, [5] had conducted a study of an enhanced version of a multi objective genetic algorithm (MOGA). In order to develop a multi-degree of freedom planar robot arm motion, an approach called as non-dominated sorting differential evolution algorithm (NSDE) is used. The NSDE is used to optimize three different objectives simultaneously. The three different objectives that had been optimized are, avoidance of singularity, obstacles and joint limit. In other work which is conducted by [11], two types of multi objective evolutionary algorithms (MOEA) has been proposed. The two types of MOEA are known as multi-objective differential evolution (MODE) and non-dominated sorting genetic algorithm (NSGA-II). The robot end effector motion can be generated by these two methods. The robot need to do a pick and place task while avoiding three obstacle and simultaneously minimizing the distance and energy.

As for this research, the evolved neural controller for generating the robot motion of the robot arm, three different optimization functions had been introduced based on the needs or requirements of this study. The three different functions are optimum time, distance and energy. Those three criteria are chosen based on human arm motion criteria and it covers range of the required robot's motion for the execution of task. A simulator were developed in MATLAB environment to test and compare the performance of generated neural controller. On the other hand, the distinctiveness of this research is the usage of a single neural controller to generate the robot's arm motion in an extensive range of the initial and intended locations. In laymen's term, the humanoid robot itself intelligently choose the best neural controller depending on the task which is required to be executed.

2. Problem formulation

In this research, a robot arm simulator had been developed in MATLAB GUI environment to implement and test and analyse the performance of proposed controller. The robot arm motion is generated for pre-determine initial and goal positions as in Figure 1. In order for the robot to execute the task, there are numbers of optimal path, trajectories and velocities characteristics need to be considered. The generated neural controller is tested with four different set of initial and goal position as tabulated in Table 1. In this experiment, the behaviour of the generated neural controller will be compared and discussed.

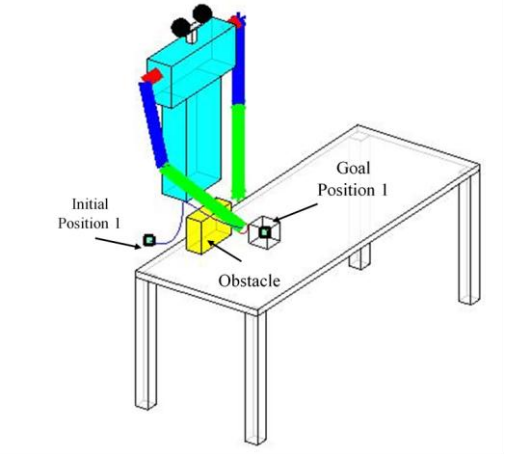


Fig.1: Robot arm simulator

Table 1 Simulation parameters

Environment Setup		*Initial Position*			*Goal Position*		
		x_{init}	y_{init}	z_{init}	x_g	y_g	z_g
Trajectory 1 (NC1L Neural Controller)	Position 1	20	0	-50	15	38	-36
	Position 2	20	0	-50	5	38	-36
	Position 3	30	0	-50	15	38	-36
	Position 4	20	0	-50	10	30	-30

3. Evolutionary algorithm

3.1. Neural networks

A feed-forward artificial neural network (FFNN) with single hidden layer is utilized in this research. Figure 2 show the applied FFNN with 3 inputs and 3 outputs. FFNN is chosen based on its simplicity, flexibility and very robust compared to other type of FFNN such as back-propagation neural networks (Montana & Davis 1989).

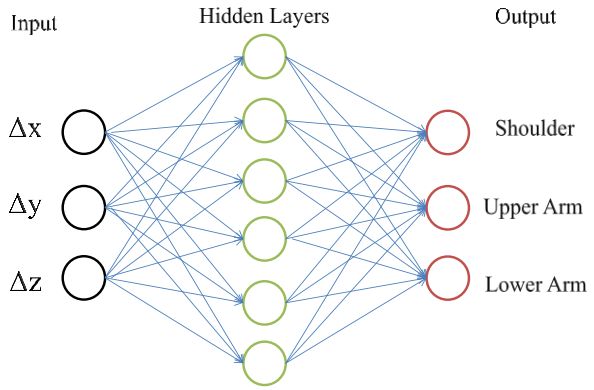


Fig. 2 Single layer FFNN

A set of FFNN solutions are generated, for the robot arm to travel from its starting position to desired goal position (as in Figure 1). Each neural controller receives three different inputs, representing the elbow angle and two shoulder angle, which is the differences between the robot hand starting and goal positions coordinate in x, y and z axis (Cartesian). A sigmoid function had been introduced to determine the hidden and output unit of the FFNN (Equation 1).

$$y = \frac{1}{1+e^{-x}} \tag{1}$$

Genetic Algorithm is utilized to optimize all the weight connections of the neural controller.

3.2. Multi objective genetic algorithm (MOGA)

Solving optimization problem with several objectives definitely need to be analysed with an irrefutable method. Evolutionary algorithm has been proven to be a reliable approach when dealing with multiple objectives problem. MOGA had been chosen for solving complex optimization problem and provide better solution compare to deterministic method such as Jacobian [12]. MOGA has the capability to optimize more than one objective function and produce multiple solution in a single run.

A non-dominated sorting GA (NSGA) was utilized to generate the motion neural controller. NSGA has better performance than other multi-objective evolutionary algorithms (MOEA) and it had been proven by [13]. NSGA can successfully determine a multiple Pareto optimal solution for the robot arm. The details explanation of MOEAs are discussed by [14]. Table 2 below shows the overall summary of GA parameters utilized in this paper.

Table 2 Summary of GA parameters

No. of Sub-populations	**3**
No. of Individuals	650, 650, 500
No. of Maximum Generations	80

4. Optimization Functions

There is a lot of robot motion characteristics to date, which can be tested. However, the focus of this study is on executing every day task. Hence, three objective functions were chosen with respect to this study. The three objective functions are optimum time (OT), optimum distance (OD) and optimum energy (OE).

4.1. Optimum time (OT)

The optimum time for the robot's hand to move from its reference position (initial) to the desired position is chosen as the first criterion. Moving from one point to another is very significant to be chosen as one of the optimization function, where the robot arm can move freely in any direction or from one point to another. Therefore, the objective function is to optimize the number of steps involved for the robot to reach the desired goal position.

$$f_1 = number\ of\ step \quad (2)$$

4.2. Optimum distance (OD)

Optimum distance can be defined as the shortest path from the initial position to the goal position. Hence, it is selected as one of the objective functions since it is related to the robot's arm movement for one place to another. The optimization function for distance is as follows:

$$f_2 = abs\ (\Sigma rt_i - sd) \quad (3)$$

where

- Σrt_i - sum of robot hand moving trajectory distance
- sd - shortest distance initial to the goal position

4.3. Optimum energy (OE)

To minimize energy and power consumption, the robot's hand should move with constant acceleration, hence moving in a more stable manner. For this study, the velocity from the motion of the robot's arm will increase gradually from the initial point and decrease gradually towards the goal position. In order for the robot to have these motion characteristics, two rejection functions had been applied. Therefore, the optimum energy objective function is as follows:

$$f_3 = \Sigma a_{hand} + (v_{hand_end} * w) + (nvc * w) \quad (5)$$

where

- Σa_{hand} - summation of robot hand acceleration
- v_{hand_end} - robot hand velocity when approaching the desired goal position,
- w - weight variable
- ***nvc*** – no. of velocity fluctuation.

The minimization and optimization of velocity fluctuation is important in order to have optimum energy usage throughout the trajectories. The weight function (w) had been introduce to adjust the priority between v_{hand_end}, Σa_{hand} and ***ncv***.

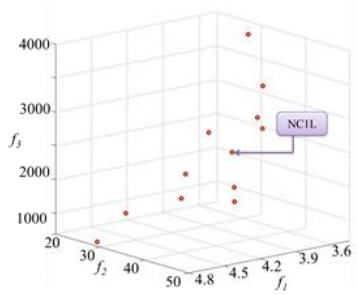


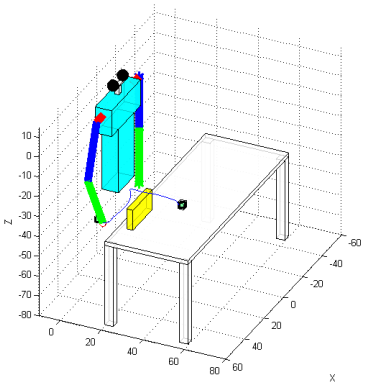
Fig. 3: Selected neural controller for optimizing all three objective functions

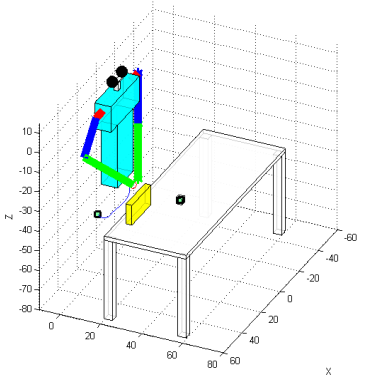
5. Result

The 80th generation Pareto front optimizing all the three criteria for the left hand (Figure 3), show a good and clear comparison among objective functions. The Pareto front shows 12 neural controllers and the best neural controller had been chosen from the generated Pareto front, *NC1L* (Neural Controller for Left Hand – 1) as in Figure 3. The neural controller matrix had been implemented in the simulated environment.

The robot arm motion for position 1 (refer Table 1) is shown in Figure 4. The neural controller shows good performance optimizing all three-objective function, optimum time, distance and energy simultaneously. The position is similar to the position when generating the neural controller. It can be observed from the motion the speed of the robot arm is constant and produce the optimum distance and time.The same generated neural controller is further tested in a different goal position in *x-axis* direction. The goal position is shifted 15 cm from its original position (Figure 5) and the robot arm manage to execute the task successfully.

For position 3, the robot arm is required to start the motion at different initial position from the generated neural controller. The initial position is shifted 10 cm from its original position in *x-axis* direction and it is shown in Figure 6. The robot arm is successfully completed the task in optimized manner.





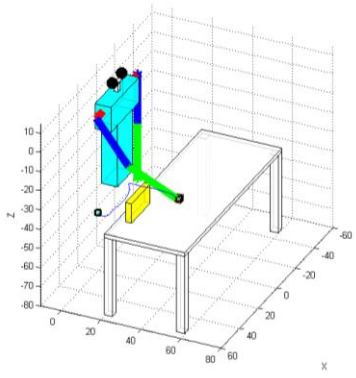


Fig. 4: Robot arm motion for position 1, initial *(20,0,-50)* and goal *(15,38,-36)*

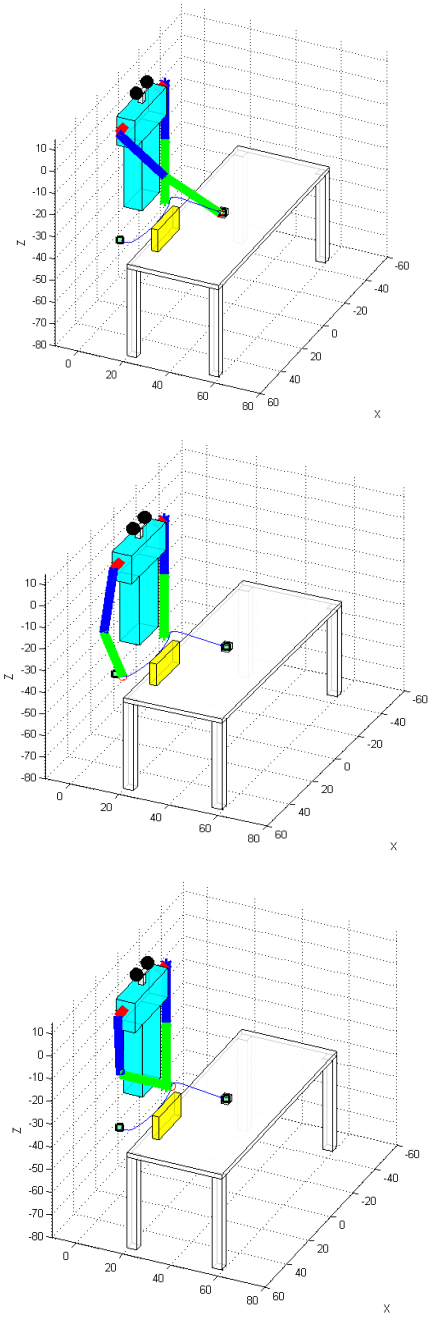
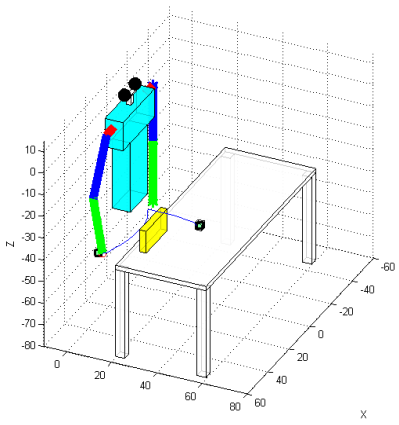


Fig. 5: Robot arm motion for position 2, initial *(20,0,-50)* and goal *(**5**,38,-36)*



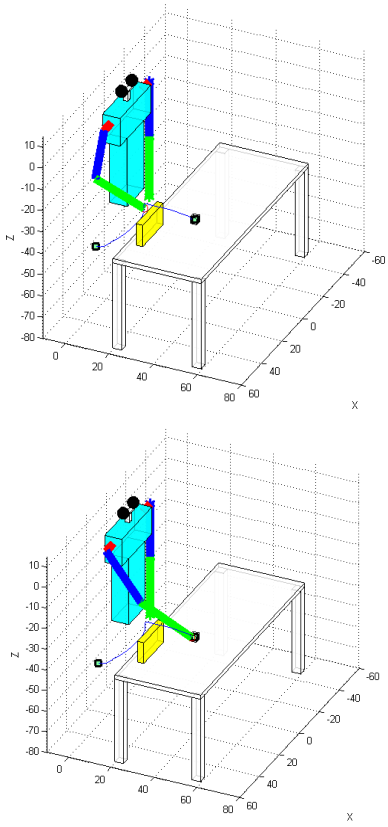
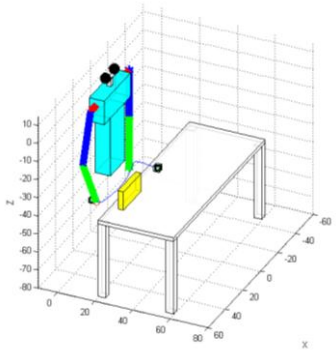


Fig. 6: Robot arm motion for position 3, initial *(**30**,0,-50)* and goal *(15,38,-36)*

Finally, the goal position is shifted in all three axes to a totally new goal position (Figure 7). The generated neural controller successfully maintains its performance and able to reach the goal position form it's reference position while avoiding the obstacle and table.



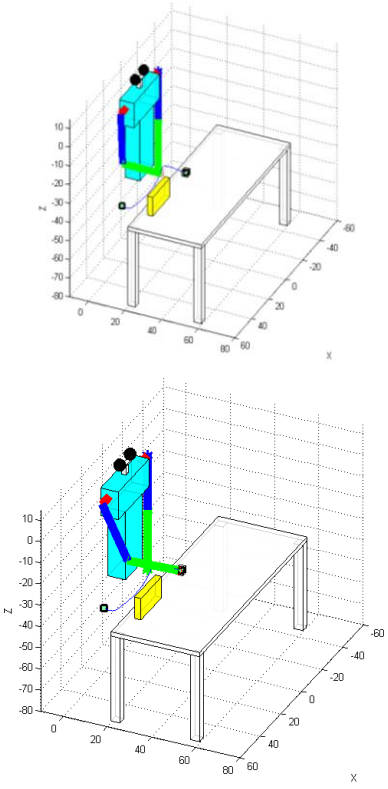


Fig. 7: Robot arm motion for position 4, initial *(20, 0,-50)* and goal ***(10, 30,-30)***

6. Conclusion

In this paper, an evolutionary-based arm motion generation in dynamic environment had been proposed. The performance in simulated environment had been tested and the results show an optimized trajectory of the robot arm. Four different initial and desired goal positions of the robot arm had been tested on the generated neural controller and the arm's motion show good performance. The robot arm has successfully reach the goal position utilizing only one neural controller and simultaneously optimizing distance, speed and energy.

Acknowledgement

Research reported in this International Journal of Engineering Technology was supported by Universiti Teknologi MARA of the Fundamental Research Grants under award number 600-RMI/FRGS 5/3 (0114/2016) and 600-IRMI/PERDANA 5/3 BESTARI (099/2018).

References

[1] N. Vahrenkamp, C. Scheurer, T. Asfour, J. Kuffner, and H. Str, (2008) "Adaptive Motion Planning for Humanoid Robots," in 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 22–26.

[2] T. Asfour, K. Regenstein, P. Azad, J. Schr, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, (2006) "ARMAR-III : An Integrated Humanoid Platform for Sensory-Motor Control," Humanoids 2006 - 6th IEEE-RAS Int. Conf. Humanoid Robot., pp. 169–175.

[3] N. Vahrenkamp and H. Str, (2009) "Humanoid Motion Planning for Dual-Arm Manipulation and Re-Grasping Tasks," no. Section IV, pp. 2464–2470.

[4] N. Vahrenkamp, M. Do, T. Asfour, and R. Dillmann, (2010) "Integrated Grasp and motion planning," 2010 IEEE Int. Conf. Robot. Autom., pp. 2883–2888.

[5] Y. Liu, Y. Jiang, and L. Li, (2011) "Multi-objective performance optimization of redundant robots using differential evolution," Proc. 2011 6th Int. Forum Strateg. Technol., no. 2, pp. 410–414.

[6] S. Kim, C. Kim, and J. Park, (2006) "Human-like Arm Motion Generation for Humanoid Robots Using Motion Capture Database," 2006 IEEE/RSJ Int. Conf. Intell. Robot. Syst., pp. 3486–3491.

[7] Z. Mohamed, M. Kitani, and G. Capi, (2013) "Humanoid Robot Arm Performance Optimization using Multi Objective Evolutionary Algorithm," Int. J. Control Autom. Syst., pp. 1–8.

[8] E. J. S. Pires, J. A. T. Machado, and P. B. D. M. Oliveira, (2004) "Robot Trajectory Planning Using Multi-objective Genetic Algorithm Optimization," pp. 615–626.

[9] E. J. S. Pires, P. B. de Moura Oliveira, and J. A. T. Machado, (2007) "Manipulator trajectory planning using a MOEA," Appl. Soft Comput., vol. 7, no. 3, pp. 659–667.

[10] R. Ur-Rehman, S. Caro, D. Chablat, and P. Wenger, (2010) "Multi-objective path placement optimization of parallel kinematics machines based on energy consumption, shaking forces and maximum actuator torques: Application to the Orthoglide," Mech. Mach. Theory, vol. 45, no. 8, pp. 1125–1141.

[11] S. Ramabalan, R. Saravanan, and C. Balamurugan, (2008) "Multi-objective dynamic optimal trajectory planning of robot manipulators in the presence of obstacles," Int. J. Adv. Manuf. Technol., vol. 41, no. 5–6, pp. 580–594.

[12] A. H. F. Dias and J. A. de Vasconcelos, (2002) "Multiobjective genetic algorithms applied to solve optimization problems," IEEE Trans. Magn., vol. 38, no. 2, pp. 1133–1136.

[13] G. Capi, (2007) "Multiobjective Evolution of Neural Controllers and Task Complexity," IEEE Trans. Robot., vol. 23, no. 6, pp. 1225–1234.

[14] K. Deb, (2009) Multi-Objective Optimization using Evolutionary Algorithms. West Sussex, UK: John Wiley & Son.