Control of Eight-Leg Walking Robot Using Fuzzy Technique Based on SimScape Multibody Toolbox

Ammar A. Aldair\textsuperscript{1}, Auday Al-Mayyahi\textsuperscript{2} and Basil H. Jasim\textsuperscript{3}
\textsuperscript{1,2,3} Electrical Engineering department, University of Basrah, Basrah, Iraq.

Abstract. In this article, a simulation platform is used to design and test the proposed controller and to study the performance of the eight legs walking robot based on SimScape Multibody toolbox. We show that, by using the SimScape Multibody toolbox, the intelligent controller can be designed without needing for the mathematical model of the legged robot. This toolbox is used to simulate and visualize the motion of the legged robot with the proposed controller. The fuzzy technique is used for designing a control system for the walking robot. The particle swarm optimization (PSO) algorithm is used to obtain the optimal parameters of output membership function of the proposed fuzzy controller. The effectiveness and robustness of the designed controller are studied by changing the carried load by the walking robot. The obtained results demonstrate efficiently the validity of the proposed controller for guiding the walking robot when the disturbance is applied.

1. Introduction

A walking robot can be used in different applications such as daily-life and industrial applications. Additionally, the walking robot is capable to serve in different fields such as military, aerospace applications, assisting disabled people, study the natural phenomenon that has a high risk and so on. Such types of robots can exhibit extended flexibility in terms of motion in comparison with wheeled robots. It can feasibly and smoothly jump and cross barriers in a rough environment. There is high nonlinearity and uncertainties inherently exist in the modeling system of a walking robot that has comprised multiple legs. Consequently, it is required to design a robust controller to guide the motion of the walking robot. Undoubtedly, the cost of multiple legs walking robots is very expensive. Hence, a reliable controller needs to be properly designed and its performance should be thoroughly studied before buying a physical robot. Moreover, simulation programs allow researchers to evaluate and estimate the overall performance of a robot and optimize the path planning of its process. For such reasons, the usage of a simulator program becomes advantageous as it can save time and cost-effectively. Different programs are utilized for the analysis of dynamic and kinematic characteristics of robot systems, for offline programming, to design different control algorithms [1]. In the literature, many researchers used different programs for simulating the motion of the robot system. The authors in reference [2] used Microsoft robotics studio (MSRS) and NET framework 4.5 to simulate the motion of the controlled mobile robot while in reference [3], the authors designed and developed the operation of the 2-R robot system by using the SolidWorks program and Matlab/Simulink software packages. In reference [4], the Microsoft robotics studio was utilized to create a model of the mobile robot, and then to integrate it into a working environment study its behavior. Another research work was introduced the SolidWorks and SimScape Multibody packing to model and study the performance of a platform comprising parallel robotics with a designed controller [5]. Other researchers used simulation programs such as 20-sim Mathematica, Dymola/Modelica, Matlab/Simulink and so on to design a control algorithm, visualize the results and analyze simulation results [6-10]. From the above literature review, it can be observed that most researchers used two different strategies to simulate the motion of a robot system i.e. tools based on general simulation systems and special tools for robot
systems. The first strategy (Tools based on general simulation systems) is used special libraries, blocks or toolboxes to build the robot system and design control system for performing different missions. By using this strategy, only the analysis of simulation results can be obtained while the visual motion of a robotic system cannot be displayed. The second strategy (Special tools for robot systems) is used as a special simulator (such as Mathematica, Dymola/Modelica and so on) to simulate the motion of special types of robots that perform special tasks such as path planning, line following and/or obstacle avoidance. The advantage of this strategy is the visual motion of the robotic system can be displayed by using a graphical user interface (GUI) environment in addition to the analysis of simulation results. However, this strategy needs high experience in programming in order to be able to design and develop control algorithms for a specific robotic system. The SimScape Multibody Toolbox combines the advantages of the two strategies and overcomes the disadvantages might exist. By using this toolbox, the motion of the robotic system can be demonstrated on a special GUI environment and special blocks or libraries which can be used to design a control system for a particular robot to perform specific tasks. In this work, the SimScape Multibody toolbox is used to simulate and visualize the motion of the eight legs walking robot with proposed controller. The fuzzy technique is used for designing the controller system for the walking robot. The particle swarm optimization (PSO) algorithm is deployed to obtain the optimal values of the fuzzy controller. The effectiveness and robustness of the designed controller are studied by changing a carried weight of the walking robot. The obtained simulation results show the validity of the proposed controller for quadruped robot and the overall performance has been improved.

2. Build the Robot Model Using SimScape Multibody Toolbox

The simulation of the robot system with a controller is a very important issue to study the effectiveness of the control system to drive the robot under any disturbances. Therefore, many researchers used different simulation program to perform this issue [11, 12]. In most of these researches, the authors need to use two programs: one for simulating the controlled system and the other to visualize the motion of the robot in different environments. In this paper, the just Simscape Multibody toolbox is used to design the controller system for the legged robot and visualize the motion of the controlled robots with different applied disturbances. The Simscape Multibody toolbox contains many libraries and Simulink blocks by which any architecture of the robot can build such as mobile robots, robotic manipulators with the different numbers of joints, and walking robots. It also contains simulation and control interfaces to help users to obtain required simulation results and display feasible the motion of a simulated robot. The basic structure to build a two-joint robotic arm is shown in Fig.1.

![Figure 1 SimScape Multibody model of two-link robot arm.](image)

Figure 2 depicts the visual display of the two-link robot arm in the Mechanics Explorer platform. In this working environment, the manipulator can perform the motion in a 3-dimensional space. To understand the used Simulink blocks that are shown in Fig. 1, their functions are described in Table 1.
Table 1 Description of SimScape blocks used in Fig. 1

| Name                      | Description                                           |
|---------------------------|-------------------------------------------------------|
| Solver Configuration      | Defines the configuration values for the simulation   |
| World Frame               | Construction of the reference point of the mechanical model. World Frame. |
| Mechanism Configuration   | Initial configuration of the mechanical and simulation parameters |
| Rotational Joint          | Rotational Joint to interpret the motion at angles between the actuators and the fixed base. |
| Solid Block               | The Solid blocks provide solid properties             |
| PS-Simulink              | PS-Simulink Converter                                  |

Figure 2 Visual display of two-link robot arm in the Mechanics Explorer display.
3. Using SimScape Multibody Toolbox for Designing Control System to Eight Legs Walking Robot

As aforementioned uncertainties and high nonlinearity, inherently exist in the mathematical model of the eight legs walking robot. Thus, the derivation of a suitable mathematical model for this robot is a complex problem. In this work, the SimScape toolbox is utilized to simulate the motion of the controlled walking robot. The undertaken study has eight legs in which each leg has three rotating joints. Therefore, there will be 24 rotational joints in total. The structure of each group of joints is identical to each other. Hence, only the control system for one joint is introduced and the others are exactly the same architecture. The SimScape model of uncontrolled one leg is shown in Fig. 3. The Simscape block of each joint needs to be modified to add the effect of the designed control system as shown in Fig. 4. From this Figure, it notes that the Simscape block diagram of each joint has two inputs and three outputs. The inputs are the ‘B’ which represents the port that connects the current link with the previous one. The second input is ‘t’ which represents the control signal supplied to the torque of the joint. The outputs are: ‘F’ is the connection point between the current link and the next one. The ‘q’ and ‘w’ are the angle and speed of a current link, respectively.

![Figure 3 The SimScape model of uncontrolled one leg.](image)

![Figure 4 the modified SimScape block of the joint.](image)

As explained earlier, the complexity and the nonlinearity inherently exist in eight legs walking robot, therefore, the design of a robust controller for this system is quite challenging. The fuzzy controller is a robust controller and most researchers use it to design a controller for a nonlinear system. In the majority of applications, researchers found that the performance of a fuzzy controller can produce better results in comparing with a classic PID controller because a fuzzy logic controller demonstrates a decent performance for controlling nonlinear and uncertain systems. There are main advantages of using fuzzy logic systems such as it can be applied to plants that are difficult to model mathematically and the controller can be designed to apply heuristic rules that reflect the experience of human experts. On the other hand, a fuzzy logic system has several parameters that should be properly adjusted. Many algorithms are used to select the optimal values of the fuzzy system such as neural network, genetic algorithm, ant colony optimization, and particle swarm optimization [13]. In this paper, the fuzzy controller is proposed to control the gait of the eight legs walking robot. The particle swarm optimization algorithm is used to adjust the centers of output membership functions of the fuzzy controller. Fig. 5 shows the connection of the fuzzy controller with one joint of the walking robot. The SPS block is used to convert simulink input signal into a physical signal. The output of a fuzzy controller \( q \) is applied to control the torque of the robot joint. The fuzzy controller has two inputs i.e. the error \( e(t) \) is the difference between the desired angle and an actual angle of the joint) and the error rate \( \dot{e}(t) \). The Mamdani inference and centroid methods are used to construct the fuzzy
controller. Five triangular membership functions are defined for each input. The Linguistic terms of each input are assigned as: Negative Big (NB), Negative Small (NS), Zero (Z), Positive Small (PS) and Positive Big (PB). The parameters of input membership functions are assumed fix (not tuned by PSO algorithm). Seven triangular membership functions are assumed for the output of the fuzzy controller assigned as Negative Big (NB), Negative Medium (NM) Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM) and Positive Big (PB). The PSO algorithm tunes just the centers of the output membership functions. Table 2 shows the fuzzy rules that are used in this work.

![Figure 5](image1.png)

**Figure 5** the connection of the fuzzy controller with the one joint of the walking robot.

| Table 2 The rules inference for each fuzzy controller |
|-----------------------------------------------------|
| **Output (q)** | **Change in Error (e)** |
| NB | NS | ZE | PS | PB |
| NB | NS | NS | NS | ZE | PB |
| NM | NS | NB | ZE | PS | PB |
| NS | NB | ZE | PS | PS | PS |
| ZE | NS | NB | PS | PB | PB |
| PS | NS | NS | PB | PB | PB |

Fig. 6 shows the flowchart that is used to tune the centers of output membership functions using the PSO algorithm. The PSO algorithm memorizes the best position for the $i^{th}$ particle $p_i^{best}$ and the best global position for the swarm $G_{best}$. The velocity $v_i$ and position $p_i$ is updated according to the following equations:

$$v_i(t + 1) = w \cdot v_i(t) + c_1 \cdot r_1(t) \cdot (p_i^{best} - p_i(t)) + c_2 \cdot r_2(t) \cdot (G_{best} - p_i(t))$$  \hspace{1cm} (1)$$

$$p_i(t + 1) = p_i(t) + v_i(t)$$  \hspace{1cm} (2)$$

where $c_1$ and $c_2$ are the social and cognitive accelerations respectively, $r_1(t)$ and $r_2(t)$ are random numbers of $t^{th}$ iteration selected from interval $[1.0]$, $w$ is the momentum constant.

The position $p_i$ represents the parameter which should be adjusted (in this work it represents the centre of output membership function of a fuzzy controller).

The following objective function is used to improve the gait of the walking robot:

$$OF = \sum_{t=0}^{T} e^2(t)$$  \hspace{1cm} (3)$$

where $T$ is the maximum number of iterations,
$e(t)$ is the $t^{th}$ error between the desired angle and the actual angle of the joint. The learning process of the PSO algorithm will be stopped when the maximum number of iterations is reached or the cost function is equal to or less than small value $\varepsilon$.

Fig. 6 the flowchart that is used to tune parameters of fuzzy controller using the PSO algorithm
4. Simulation and Results

In this paper, the SimScape Multibody toolbox is used to design and simulate the eight legs walking robot as shown in Fig. 7. Each leg has three joints, so that, the total number of joints is twenty-four joints. A different fuzzy controller is designed for each joint in the walking robot, therefore, twenty-four fuzzy controllers are designed in this work in total. The fuzzy controller with the PSO algorithm is designed by using Simulink toolbox. Simulink to PS converter block is used to connect the Simulink blocks with the SimScape blocks as shown in Fig. 5 given previously. The effectiveness and robustness of the designed controllers are tested by adding extra weights to the body of the walking robot.

Fig. 8 shows the walking robot standing at the initial position before running the simulation. Fig. 9 demonstrates the position-walking robot after a few meters. Fig. 10 depicts the position of the walking robot when it reaches the end of the road.
Because the designed walking robot has eight legs and each leg has three joints, therefore, it is difficult to illustrate the performances of each joint. In this work, only the performance of the right front leg is shown. Fig. 11 shows the fuzzy control efforts of the right front leg for three links: ankle, hip and knee. In addition, the comparisons between the desired angle and the actual angle (for the first two of gaits) for the three joints are shown in Figs. (12-14).

Figure 10 Position of walking robot at end of the road.

Figure 11 Fuzzy control effects for ankle, hip and knee.

Figure 12 Comparison between the reference angle and actual angle of the ankle.
To study the robustness and effectiveness of the designed controller, two scenarios of disturbances are applied to the controlled system. In the first scenario, one Kg of additional weight is added on the back of the walking robot as shown in Fig. 15.

Figure 13 Comparison between the reference angle and actual angle of the Hip.

Figure 14 Comparison between the reference angle and actual angle of the Knee.

To study the robustness and effectiveness of the designed controller, two scenarios of disturbances are applied to the controlled system. In the first scenario, one Kg of additional weight is added on the back of the walking robot as shown in Fig. 15.

Figure 15 Visual display of eight legs walking robot in the Mechanics Explorer display with 1 Kg.
Fig. 16 shows the fuzzy control efforts of the right front leg for three links: ankle, hip and knee when 1 Kg is added. Furthermore, the comparisons between the desired angle and the actual angle (for the first two of gaits) for the three joints are shown in Figs. (17-19).

Figure 16 Fuzzy control effects for ankle, hip and knee when 1Kg is added.

Figure 17 Comparison between the reference angle and actual angle of the ankle when 1Kg is added.

Figure 18 Comparison between the reference angle and actual angle of the Hip when 1Kg is added.
In the second scenario, 2 Kg is added on the platform of a walking robot to test the performance of the designed controller as shown in Fig. 20. While Fig. 21 shows the fuzzy control efforts of the right front leg for three links i.e. ankle, hip and knee when 2 Kg is added. In addition, the comparisons between the reference angle and the actual angle (for the first two of gaits) for the three joints are shown in Figs. (22-24).

Figure 19 Comparison between the reference angle and actual angle of the Knee when 1Kg is added.

Figure 20 Visual display of eight legs walking robot in the Mechanics Explorer display with 2 Kg.

Figure 21 the fuzzy control effects for ankle, hip and knee when 2 Kg is added.
Figure 22 Comparison between the reference angle and actual angle of the ankle when 2 Kg is added.

Figure 23 Comparison between the reference angle and actual angle of the Hip when 2 Kg is added.

Figure 24 Comparison between the reference angle and actual angle of the Knee when 2 Kg is added.
5. Conclusion

The Fuzzy controllers are designed for the eight legs walking robot and the optimal parameters of the fuzzy controllers are adjusted using the PSO algorithm. The studied walking robot has twenty-four joints, therefore, twenty-four fuzzy controllers are designed for each joint. The Simscape toolbox is used to implement the walking robot and the simulation toolbox is used to construct the proposed controllers with the optimization algorithm. The obtained simulation results have shown that the proposed controllers have successfully guided the walking robot to follow the desired gaits. The robustness of the proposed controllers has been tested by adding extra weights on the back of the walking robot. Two different weights are added i.e. 1 Kg and 2 Kg. In both cases, the controlled system has been stable and flow the desired gaits with rejection to disturbance. Therefore, the overall designed system demonstrate feasible suitability and effectiveness in controlling the walking robot.

References

[1] Kumar P. and Narayan Y. 2011 Simulation in Robotics Proceedings of the National Conference On Recent Advances In Manufacturing Engineering & Technology, January 10-1.
[2] Mandita F., Kunsanto G. and Sadewa E. 2016 Simulation of Controlling Mobile Robot Using Microsoft Robotics Studio (MSRS) and NET Framework 4.5 International Journal of Emerging Technology and Advanced Engineering, Vol. 6, No. 12, pp. 71-78.
[3] Gouasmi M., Ouali M., Fernini B. and Meghatria M. 2012 Kinematic Modelling and Simulation of a 2-R Robot Using SolidWorks and Verification by Matlab/Simulink, International Journal of Advanced Robotic Systems, Vol. 9, No. 2, pp. 1-13.
[4] Yusof Y., Abu Hassan F., Mohd N. and Azizan W. 2012 Development of an Educational Virtual Mobile Robot Simulation, Proceedings of the World Congress on Engineering 2012 Vol II London, UK.
[5] Olaya J., Pintor N., Aviles O. and Chaparro J. 2017 Analysis of 3 RPS Robotic Platform Motion in SimScape and MATLAB GUI Environment, International Journal of Applied Engineering Research Vol. 12, No. 8, pp. 1460-1468.
[6] Martin H., Johan A. and Anders R. 2009 Optimal Robot Control using Modelica and Optimica, 7th International Modelica Conference, Como, Italy.
[7] Price D., Walsh S. and Nahavandl S. 2016 “Unifying Manufacturing Simulation Models Using HLA”, ASME, Geelong, Australia.
[8] Liao H., Noguchi M., Maruyama T., Muragaki Y., Kobayashi E., Iseki H. and Sakuma I.. 2013 Medical Image Analysis, Science Direct, Tokyo, Japan.
[9] Aranha C., Carvalho S. and Gonc L. 2014 Cambio: Realistic Three Dimensional Simulation of Humanoids based on Computer Vision and Robotics, Universidade Federal do Rio Grande do Norte, Brazil.
[10] Afzani N., Johari M., Haron H., Syukor A. and Jaya M. 2013 Robotic modeling and simulation of palletizer robot using Workspace, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia.
[11] Kubela T., Pochlyy A. and Singule V. 2010 Advanced Tools for Multi-body Simulation and Design of Control Structures Applied in Robotic System Development, Solid State Phenomena. Vol. 164, pp 387-391.
[12] Takaya K., Asai T., Kroumov V. and Smarandache F. 2016imulation Environment for Mobile Robots Testing Using ROS and Gazebo 20th International Conference on System Theory, Control and Computing (ICSTCC), October 13-15, Sinaia, Romania.
[13] Ammar A. Aldair, Eman A., Turki Y. 2019 Design of ABCF Control Scheme for Full Vehicle Nonlinear Active Suspension System with Passenger Seat, Iranian Journal of Science and Technology - Transactions of Electrical Engineering. 2019 Vol. 43(Suppl 1), pp.289-302.