MODELING AND SIMULATION FOR ONE LEG OF QUADRUPED ROBOT USING GSA PID CONTROLLER

HASRUL CHE SHAMSUDIN, MOHAMMAD AFIF AYOB

Electrical Engineering Department, Politeknik Sultan Haji Ahmad Shah, Semambu, 25350 Kuantan, Pahang, Malaysia
Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Malaysia
E-mail: hasrulcheshamsudin@yahoo.com, mohammadaffayob@gmail.com

Abstract- Designing a robot leg with great performance is one of the interest parts in designing legged robot. This is due to the nonlinearities and input couplings presented in the dynamics of the robot leg. Feedback control system such as classical PID controller is widely used to control the response of the system. This study considered the problems of modelling and control of a 3 degree of freedom for one leg of quadruped robot. The drawbacks exist in conventional PID such as tuning three-term control can be time consuming and system unstable behaviour. Therefore, we introduce a variant tuning technique to calculate the best PID values. The research work was divided into several developmental stages; Firstly, the complete mathematical model of a 3 DOF of one leg using trigonometry method including the dynamics of servomotors actuators in the state variable form is to be developed. Then, the GSA PID Controller is applied to the robot leg. To perform the simulation, Solid Works 2013 x64 Edition is used to develop the 3D modelling of the leg while Sim Mechanics with First Generation Format was applied to export the models to Simulink. The simulation was done on desktop with 64 bit Windows 7 Professional, 8 GB installed memory and Intel(R) Xeon(R) 3.7 Ghz processor. Lastly, the performance of GSA PID controller is to be compared with the conventional controller which is PID. The simulation results show that the best PID values can be obtain very quickly and the output performance of GSA PID is better than the performance of conventional method of PID controller.

Keywords- Sim Mechanics, Legged robot, GSA PID.

I. INTRODUCTION

Legged robotic technology is constantly evolving due to its flexibility and adapt to different rough terrain in a better way [1]. Robot platform will help advance knowledge in locomotion by demonstrating the feasibility of gait patterns and motions. Kinematic models of the robot need to be developed to help determine important parameters and characteristics [2]. For 12 DOF quadruped robots, each leg has hip, knee and ankle with three degrees of freedom per leg as shown in Fig. 1. Kinematics deals with the movement of end effector of the robot manipulator relative to the base of the manipulator as a function of time [3], [4], [5]. Kinematic analysis of the robot manipulator can be divided into two categories which is forward kinematics and inverse kinematics. To improve legged robot walk gait, researchers concentrate on studying the technique to improve robot leg controller. This is due to the nonlinear and dynamic input physically present in the robot leg. Every controller produces different output performance for robot movement. Robot leg performance can be analyzed by assessing the initial to the end position movement. Moreover, parameter uncertainty and external disturbances must be considered while designing robot controller. Therefore, robot controller must be able to operate linear and nonlinear systems. Good controller should minimize the error occur between real and intended position by achieving certain specifications such as to reduce resulting overshoot, fast rise time or eliminate the steady state error [6], [7]. PID controller is a good control technique in control applications due to its robustness and easy to be implemented. Although PID is most widely used controller to gain the response of the system, it has several drawbacks. Generally, PID controller has to balance all three gains impact to the whole system and may compromise the transient response, such as settling time, overshoot and oscillation. For manual tuning case, several procedures and times is required to tune the PID controller to provide desired system performance. Since basic PID algorithm presents some challenges in control applications, many researchers addressed modifications to the PID form. The study of Fuzzy PID shows that the robot performance increase by about 3 seconds in advance while the overshoot of the system decrease by 40 percent, comparing to the robot with traditional PID controller [8]. On the other hand, there are studies that proposed heuristic approach in order to improve PID controller performance such as by using Genetic Algorithms and PSO [9], [10]. Gravitational Search Algorithm (GSA) has been originally proposed by Rashedi et al. in 2009. The idea of GSA came from the Newtonian laws of gravitation and motion where all objects move as a result of attraction with each other by gravitational forces. Objects with heavier mass have stronger attraction and move slower than the objects with relatively small mass. The results in [11] provide that GSA performed considerably better compared to Particle Swarm Optimization (PSO) [12] and Central Force Optimization (CFO) [13]. Therefore, this
research attempts to use GSA as an optimization technique to control the servomotor response of robot leg. GSA will be used to optimize the PID values by comparing the desired angle of servomotor and the actual position. The rest of this paper is organized as follows; robot leg modelling will be described in the next section including kinematics of the leg and the servomotor modelling. Then, PID structure with the gravitational search algorithms used for continuous problem is explained. After that, experimental results obtained and discussions will be showed before the conclusion was made to conclude the overall study.

II. ROBOT LEG MODELLING

2.1. Forward Kinematic

The forward kinematic equations describe the functional relationship between the joint variables and the position and orientation of the end-effector. In this paper, geometric approach is used to solve the joint angle equations required. A complex mathematical process is reduced using basic trigonometric in the modelling one leg of quadruped robot. Fig. 2 shows the 3D model for one leg and the respective skeleton outline. Eq. 1 to Eq. 3 presents the forward kinematic model.

\[ x = [87\cos\theta_2 + 115\cos\theta_2 + 3]\cos\theta_1 \]  
\[ y = [87\cos\theta_2 + 115\cos\theta_2 + 3]\sin\theta_1 \]  
\[ z = 105 + 87\sin\theta_2 - 115\cos\theta_2 + 3 \]  

2.2. Inverse Kinematic

Purposes of inverse kinematic is to find the joint variables of the robot manipulator for a given position and orientation of the robot hand. The computation of the inverse kinematics of the robot manipulator is a difficult task because of the nonlinearities and multiple solution problems. For reference, Eq. 4 to Eq. 18 shows the inverse kinematic equation for the robot leg.

\[ r = \sqrt{x^2 + y^2 - 87\cos\theta_2 + 115\cos\theta_2 + 3} \]  
\[ h = z - 105 - 87\sin\theta_2 - 115\sin\theta_2 + 3 \]  
\[ k_1 = 87 + 115\cos\theta_3 = d\cos\gamma \]  
\[ k_2 = 115\sin\theta_3 = d\sin\gamma \]  
\[ d = k_1^2 + k_2^2 \]  
\[ \gamma = \arctan2(k_2, k_1) \]  

Add Eq. 4 with Eq. 5:

\[ x^2 + y^2 + (z - 105)^2 = r^2 + h^2 \]  
\[ \cos\theta_3 = \frac{x^2 + y^2 + (z - 105)^2 - 87^2 - 115^2}{2(87)(115)} \]  
\[ \sin\theta_3 = \sqrt{1 - \cos^2\theta_3} \]  
\[ \theta_3 = \arctan2(\sin\theta_3, \cos\theta_3) \]  

Using Sum-Difference Formulas on Eq. 4 and Eq. 5, then substitutes with Eq. 6 and Eq. 7:

\[ r = d\cos(-\theta_2 - \gamma) \]  
\[ h = d\sin(-\theta_2 - \gamma) \]  
\[ -\theta_2 - \gamma = \arctan2(h, r) \]  
\[ -\theta_2 = \arctan2(h, r) + \gamma \]  
\[ \theta_1 = \arctan2(y, x) \]
2.3. Servomotor Modelling

DC servomotors convert electrical energy to mechanical energy. The motor has rotary movement, and when combined with mechanical part it can provide translation movement for the desired joint. Servomotor can be considered as a linear SISO system having 3rd order transfer function. Relation between shaft position and armature voltage is derived from the physical laws [14] as shown in Eq. 3.

\[ \frac{\theta(s)}{V(s)} = \frac{K_e}{J_m L_a s^2 + (L_a R_m + R_a I_a) s + K_T K_e} \times [1] \] (19)

### Table 1: DC Servomotor Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moment of inertia ( J_m )</td>
<td>0.01 Kg.m²</td>
</tr>
<tr>
<td>Friction Coefficient ( B_m )</td>
<td>0.1 N.m/s</td>
</tr>
<tr>
<td>Back EMF constant ( K_b )</td>
<td>0.3164 V.ms⁻¹</td>
</tr>
<tr>
<td>Torque constant ( K_T )</td>
<td>0.3164 V.ms⁻¹</td>
</tr>
<tr>
<td>Resistance ( R_a )</td>
<td>1 ohm</td>
</tr>
<tr>
<td>Inductance ( L_a )</td>
<td>0.5 H</td>
</tr>
</tbody>
</table>

III. GRAVITY BASED ALGORITHM TO OPTIMIZE PID TUNING

3.1. PID Structure

The response of leg robot joint will be controlled by PID algorithm as shown in Fig. 4. The error signal is the difference between the desired input and the actual output angle. If the error between the output and the input values is large, then large input signal is applied to the physical system. If the error is small, a small input signal is used. Any change in the control signal, is directly proportional to change in the error signal for a given proportional gain \( K_p \). On the other hand, a derivative term is added to form the PD controller, which tends to adjust the response to the set point. The output of PD controller is calculated based on the sum of both current error and change of error with respect to time. The main function of the third term is the integral, I control tends to reduce the effect of steady state error that may be caused by the proportional gain, where a smaller integration time result is the faster change in the controlled signal output. Each term of the three components are amplified by an individual gain. As a result, the sum of the three terms is applied as an input to control the response of DC servomotor angle.

3.2. GSA PID

GSA optimization process is realized by a group of agents inspired by the Newton’s laws of gravitation and motion. Fig. 5 shows the general principle of GSA. The optimization processes started with randomly positioning the agents with random velocity values and initialize the gravitational constant. The next step is to evaluate the Fitness for each agent according to the objective function which to minimize the error between desired and actual angle as shown in Eq. 20 and Eq. 21. Then, the gravitational constant is updated since the effect of gravity is decrease. The best value of the population is calculated using Eq. 22.

\[ \text{Fitness} = \sum_{i=1}^{n} \sqrt{(y_d - y_i)^2} \] (20)

If \( \text{Fitness}(t,i) < \text{GlobalBestFitness}(t) \) (21)

then

\[ \text{GlobalBestFitness}(t) = \text{Fitness}(t,i) \] (22)

\[ \text{best}(t) = \min_{j \in \{1, \ldots, n\}} \text{Fitness}_j(t) \] (23)

Mass, \( M_i \), for each agent is calculated using Eq. (23), and acceleration, \( a \), is calculated using...
where $M_{aj}$ is the active gravitational mass related to agent $j$, $e$ is a small constant, $R_{ij}$ is the distance between agent $i$ and $j$ and $\text{rand}_j$ is a uniform random variable in the interval $[0,1]$. Then, the velocity, $v_i^d$, and position, $x_i^d$, of $i$th agents is calculated as follows:

$$v_i^d(t + 1) = \text{rand}_i \times v_i^d(t) + a_i^d(t)$$

$$x_i^d(t + 1) = x_i^d(t) \times v_i^d(t + 1)$$

where $\text{rand}_i$ is a uniform random variable in the interval $[0,1]$. This updating process is repeated as long as the stopping criterion is not satisfied.

IV. SIMULATION RESULTS

This section discusses the simulation and results. MATLAB and SIMULINK are used to evaluate the performance of the proposed controller. Two types of control algorithms which are PID and GSA PID were implemented to control the 3 DOF robot legs using an independent joint control mechanism as shown in Fig. 6. Table 2 shows the desired input angle from the initial position for each joint.

<table>
<thead>
<tr>
<th>Joint</th>
<th>Initial Position</th>
<th>Desired Angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hip</td>
<td>0°</td>
<td>-20°</td>
</tr>
<tr>
<td>Knee</td>
<td>0°</td>
<td>-30°</td>
</tr>
<tr>
<td>Ankle</td>
<td>0°</td>
<td>45°</td>
</tr>
</tbody>
</table>

In order to study the performance of the proposed controller, simulation studies have been conducted to check the efficiency of the system. The simulation was done for one leg with no controller, using PID controller in Simulink Library and with PID GSA controller. For simulation with PID controller, Tune button is pressed several times for tuning PID value automatically. Fig. 7 until Fig. 10 shows the output response of the DC servomotor of the 3 DOF robot leg for ... It is show that PID GSA controller gives better response. Table 3 summarizes the performance of ... with no controller, PID controller and GSA PID controller. Although the rise time is slower than PID controller Tune Once but GSA PID controller improvement for P.O is about 37 times with almost the same of setting time. The behaviour for other joints is shown in Fig. 11.
CONCLUSIONS

In this study, robot leg was split to two different aspects which are modelling and control.

Modelling process includes kinematic analysis and DC motor modelling. For controller part, it is proven that the GSA PID time response behaviour is more efficient than classical PID controller with minimum time to tune the three term values as expected. The system characteristics (Tr, O.S and Ts) for other two joints are as follows; (0.3395s, 1.0387%, 1.1048s) and (0.3387s, 1.0786%, 1.1057s). On the other hand, offline simulation was used in the proposed design. For future works, it can be extended to an on-line controlling simulation for the leg robot.

ACKNOWLEDGMENTS

The authors would like to thanks Ministry of Education (MOE), Malaysia for providing financial support under research grant FRGS/1/2014/TK03/JPP/03/1

REFERENCES


Modeling And Simulation for One Leg of Quadruped Robot Using GSA Pid Controller


★★★