Design and simulation of PID controller for lower limb exoskeleton robot

Munadi, M. S. Nasir, M. Ariyanto, Norman Iskandar, and J. D. Setiawan

Design And Simulation Of PID Controller For Lower Limb Exoskeleton Robot

Munadi[1,a)], M. S. Nasir[1)], M. Ariyanto[1)], Norman Iskandar[1)], J. D. Setiawan[2)]

[1]*Robotic, Automation, and Computation Lab., Dept. of Mechanical Engineering, Diponegoro University, Indonesia*
[2]*Mechatronic, Instrumentation, and Reliability Lab., Dept. of Mechanical Engineering, Diponegoro University, Indonesia*

[a)]Corresponding author: muna_096@yahoo.com

Abstract. Lower limb exoskeleton robot is a robot that functions as a walking tool of movement for the lower of the human body (usually made for patients with stroke and paraplegic). However, as time goes by lower limb exoskeleton robot is no longer reserved for patients with stroke and paraplegic, some of them are made to military requirements or the need to lift heavy stuff. In this case, the lower limb exoskeleton robot is used for rehabilitation aids for people with paraplegic who suffered an accident on the lower part of the human body (lower limb). This paper presents the design and simulation of the lower limb exoskeleton robot using CAD software to design, and Matlab/Simulink for simulation. The first step after the design of the lower limb exoskeleton robot is completed, we export the file of robot design to the SimMechanics (using first generation). After we obtain the results of the export file, we give a Proportional-Integral-Derivative (PID) controller at each revolute joint. Then, we use a Matlab/Simulink to analyze the response system of PID controller. For the stability analysis of controller, we plot the transfer function of system into the root locus method. Based on the results of simulation, the PID controller has been able to making each revolute joint of the lower limb exoskeleton robot to become stable.

INTRODUCTION

Stroke is the loss of brain function due to impaired blood supply in the brain. Without blood to supply oxygen and nutrients, the brain cells will quickly begin to die. When it happens, the affected person usually has one or more limbs on one side of the body, because the contralateral affected area of the brain can no longer function. When the lower limb is affected, rehabilitation treatment is needed to restore body function and regain the ability to walk independently. A number of individual rehabilitation treatment trials have been proposed to improve the ability to walk again [1].

Paraplegia is paralysis of both lower body members and both legs that are caused by severe injury to the lower level spinal cord [2]. The injury causes the entire impulse of the brain to be unacceptable to the underlying muscle tissue, and otherwise the impulse from below the broken level is unacceptable to the brain. As a result paraplegia sufferers lose motor and sensory function under damaged areas, lose power, become weak and wither. The patients also lose the ability to control urination and defecate (bladder and bowel control), so the sufferers of paraplegia become very dependent on others.

The development of cases of paraplegia is increasing rapidly in line with the increase of natural disasters (tsunami, earthquake) and the occurrence of accidents (work, traffic, households). One of the robots that can help paraplegia therapy is a lower limb exoskeleton robot. The lower limb exoskeleton robot is helpful for paraplegia sufferers, and also serves for therapy. It is caused by the technology develops more human-friendly, smarter and safer. So, the development of self-standing lower limb exoskeletons for physical assistance is one of the most addressed mobility assistance options.

Recently published reviews give us the opportunity to obtain a general acknowledgment of the exoskeleton robot [3-7]. Panich [8] described the design of the leg-exoskeleton robot for rehabilitation, including kinematic analysis. Veneman et al. [9] introduced a newly developed gait rehabilitation device, called LOPES, combines a freely

translatable and 2-D-actuated pelvis segment with a leg exoskeleton containing three actuated rotational joints: two at the hip and one at the knee, in which the joints are impedance controlled to allow bidirectional mechanical interaction between the robot and the training subject. Kim et al. [10] proposed a design of the walking assistive lower limb exoskeleton for paraplegic patients, and they validated a hardware using the centre of pressure. Wong et al. [11] presented the mechanical analysis of a lower limb exoskeleton for rehabilitation for both legs. They developed a wearable exoskeleton system based on the average height of Malaysian to enhance the wearability of the system to lower limbs. About the control strategy, Beyl et al. [12] and Pan et al. [13] emphasized the control algorithms. Meanwhile, Gmerek [14] proposed the mechanical and hardware design of a robot for the rehabilitation of upper extremities that is called ARR-1 (Arm Rehabilitation Robot). Therefore, we will proposed a design and control strategy for the lower limb exoskeleton robot based on the average height and weight of Indonesian. We will explain about the design process using CAD software, then the robot design is directly exported to SimMechanics (first generation) to be simulated in Matlab/ SimMechanics for the robotic system motion.

DESIGN OF LOWER LIMB EXOSKELETON ROBOT

The paraplegic patients have difficulty walking. They cannot move their legs independently. One way to help paraplegic patients in order to walk is using a robot. The robot is a powered lower limb exoskeleton designed to regain locomotion capability, in which it is properly designed to hold with human leg by belts. Then, this robot is designed for the Indonesian, in which the robot is able to support the maximum weight of 80 kg and height 165 cm. Figure-1 shows the design of lower limb exoskeleton robot which has six DoFs. The characteristics of the human leg have a total of 7 degrees of freedom from 14 DoFs, in which there are 3 DoFs for the hip joint, there is 1 DoF for the knee joint, and there are 3 DoFs for the ankle joint. So, for a pair of legs there are 14 DoFs. But in this study, we develop the lower limb exoskeleton robot has 6 DoFs. For each joint, the electric motor is used as the actuator due to its higher efficiency and lower weight.
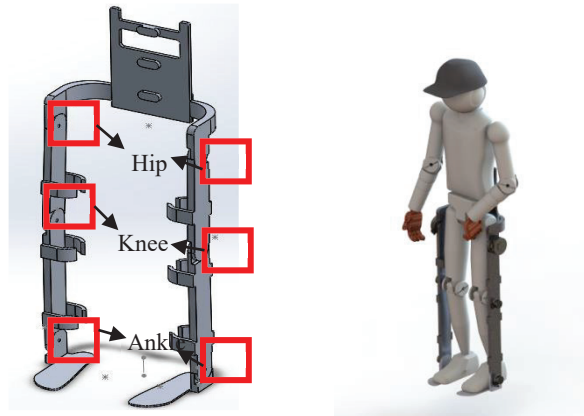

FIGURE 1. Design of the lower limb exoskeleton robot.

For the hardware, the each joint of the robot will be integrated with flat DC motors with encoder, integrated sensors, and inertial measurement units (IMUs). The encoders at each joint of the robot are integrated to measure the motion angle of hip, knee and ankle joints, in which the range of motion is listed in Table I. The integrated sensors in each flat DC motor to measure the motor velocity and temperature. The IMUs at the segments of thigh will be used to measure the acceleration, velocity, and orientation of the corresponding segments in the world frame.

TABLE 1. Range of motion at each joint

DoF	Hip	Knee	Ankle
Flexion	100º	100º	20º
Extention	20º	5º	20º

The weight of the lower limb exoskeleton robot is very important. The structure is not only light but also ensured for its durability. Thus, aluminum alloy 2024 is selected to be the structure of the robot. The aluminium alloy 2024 is an aluminium alloy, with copper as the primary alloying element. It is used in applications requiring high strength to

weight ratio, as well as good fatigue resistance. **So**, it will not add a load for the paraplegic patients when they are walking. The material properties of aluminium alloy 2024 is shown in Table II.

TABLE 2. Material properties of aluminium alloy 2024

Property	Value	Unit
Elastic Modulus	73,000	MPa
Poisson's Ratio	0.33	-
Mass Density	2800	Kg/m^3
Tensile Strength	186,126	MPa
Yield Strength	75,829	MPa

MODELLING OF THE ROBOT USING SIMMECHANICS

We design the robot using a CAD software. After we get a final design, we can export the image to SimMechanics toolbox from Matlab, so the dynamic model was implemented using SimMechanics. For controlling the robot, we have to use a controller. In this simulation, we will use a PID controller as an actuator control at each joint.
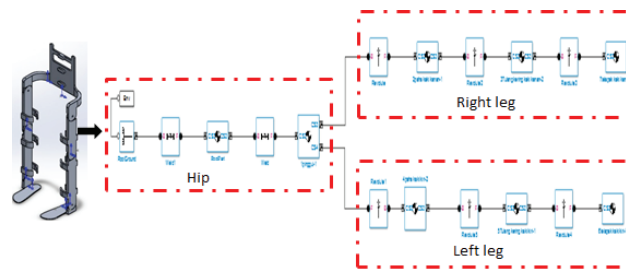
FIGURE 2. SimMechanics model of the robot.

Before designing the controller, we have to describe a plant in the mathematical equation. Then, this mathematical equation or mathematical modeling will be converted in-Laplace to be changed to the s domain, so we get the transfer function that will be used as the plant of the robot. But, since the model of the robot uses the application of SimMechanics, then mathematical modeling is not needed anymore. It is because the mathematical modeling is automatically formed. Figure-2 shows the SimMechanics model of the robot that has been resulted in SimMechanics from a CAD software. Based on Figure-2, the plant of the robot can be said as a model if it has the inputs, the control systems, and the outputs. It is shown in Figure-3. Furthermore, by using a step input or a sine wave input, we obtain a motion implementation of robot that is described on the virtual reality of the robot, which it is shown in Figure-3.
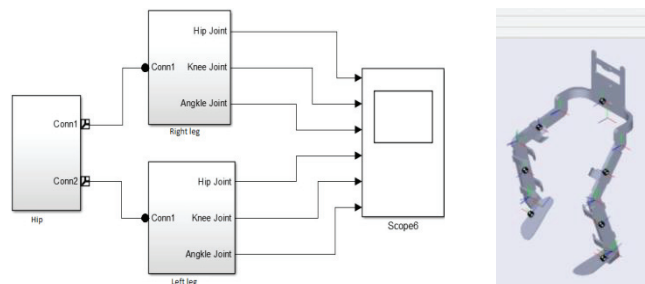

FIGURE 3. SimMechanics system and SimMechanics virtual model of robot.

SIMULATION AND RESULTS

The Hip Joint Of The Left Leg

This section will present the simulation results of an actuator as the plant at each revolute joint of the left leg. For simulation, each revolute joint is given a 20 degrees for the step input as the reference position to be achieved. Figure-4 shows a modelling a hip joint of left leg wihout a controller, and Figure-5 shows the response system.
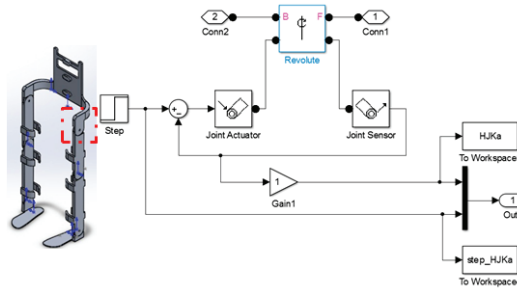

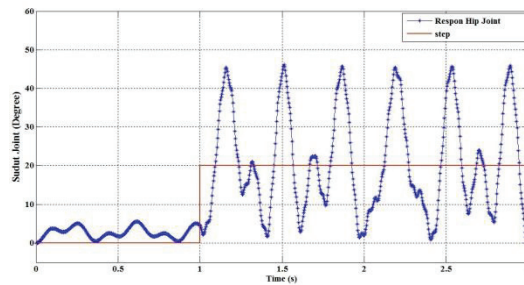**FIGURE 4.** Modelling a hip joint of left leg without a controller.

FIGURE 5. Response system generated by hip joint of left leg without a controller.

Based on the response system of the hip joint (left leg) on Figure-5, it shows that the system is not stable, so we can give a PID controller. Figure-6 shows the modelling is equipped with a PID controller. Before the control design begins to be simulated, the PID block must be tuned in advance of obtaining the ideal gain value. It is done by double click on the PID block, then click on the tune button. So, we obtain the graph that is shown in Figure-7.
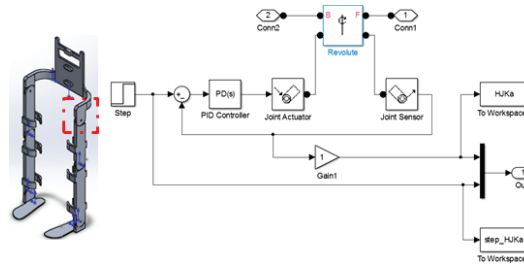

FIGURE 6. Modelling a hip joint of left leg using the PID controller.
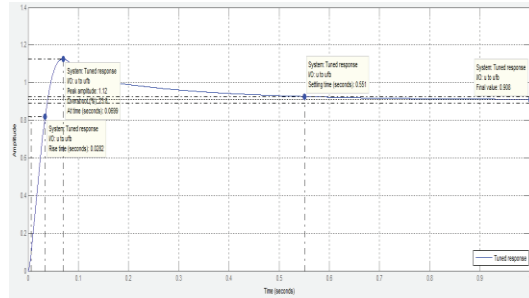
FIGURE 7. Tunning result the PID controller of the hip joint of left leg.

When the tuning process is done, then click the run button to get the simulation result of the hip joint. The result shows that the response of output is close to 20 degrees, it is according to the desired input, and Figure-8 shows output results. It shows that at 0 to 1 seconds, when the system without the step input, the system is very good output, without experiencing oscillation. Next, when the system is given by the input step 20 degrees, there is overshoot at first second, but the response will be stable again in third second according to the input. The response can already be said to work for a system, because it is in accordance with the input step.
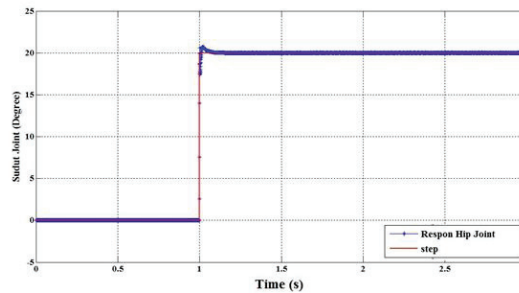
FIGURE 8. Output result for the hip joint of left leg using the PID controller.

Next, we will discuss about the stability analysis. The analysis begins with the linierization process of the plant to obtain the transfer function of the plant that is shown in Equation 1.

$$\frac{\begin{matrix} 7806,3(s+23,66)(s+3,552)(s+1,908)(s^2+10,28s+26,44) \\ (s^2+237,5s+5,738e04)(s^2+956,3s+6,211e05) \end{matrix}}{\begin{matrix} (s+24,01)(s+5,239)(s+5,14)(s+4,164)(s+1,91)(s^2+73,78s+3101) \\ (s^2+207,9s+1,081e05)(s^2+979,5s+7,535e05) \end{matrix}} \qquad (1)$$

The following steps are be taken to get the equation of linierization using Matlab:

a. Right click on the arrow of the step block, then direct to the linear analysis point, then click the input point, so that it looks like the arrow down direction on step arrow. Next, we click the open loop that is shown in Figure-9.
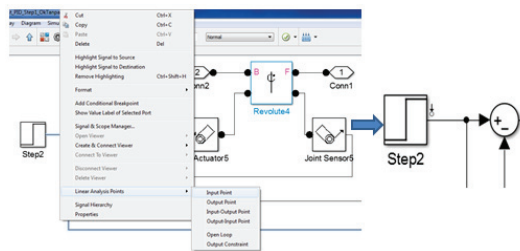

FIGURE 9. Step to insert the input point.

b. The next step is to right-click on the arrow of the joint sensor block, and direct to the linear analysis point, then click the output point and also output contraint, making it look like the upward arrow arrow on the arrow of the joint sensor block. This process is shown in Figure-10.
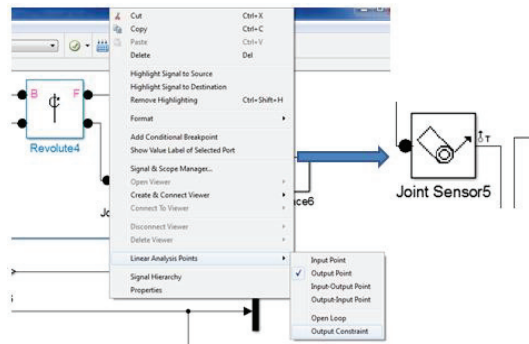

FIGURE 10. Step to insert the output point and output constraint.

c. Click analysis on the menu bar, then direct to control design, then click the linear analysis. Furthermore, click run linearize, so the result is shown in Figure-11. In additional, clik the menu bar of linier analysis tool, select report, and show result details, so we obtain the transfer function which is shown in Equation 1.
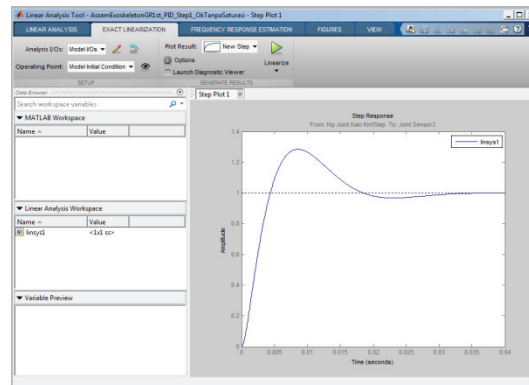

FIGURE 11. Ploting result of linier analysis.

After the linearization, the transfer function is obtained in Equation 1, and the root locus of the transfer function is shown in Figure-12. The root locus graph shows that the system has been successfully controlled. It is indicated by the location of the pole which is entirely on the left of the imaginary axis, so the system becomes stable. There are some zero values that are on the left side of the imaginary axis, indicating that the system still has overshoot with a bit of oscillation.
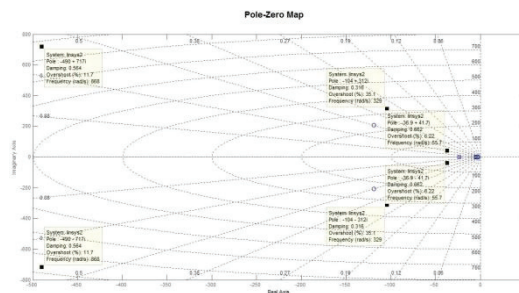

FIGURE 12. Root locus stability of the hip joint of left leg.

The Knee Joint Of The Left Leg

The same steps are done in the analysis of knee joint. Figure-13 shows modelling a knee joint of left leg using the PID controller. Before the control design begins to be simulated, the PID blocks within a plant must be tuned in advance to obtain the ideal gain control value. The response with the most ideal value is shown in Figure-14.
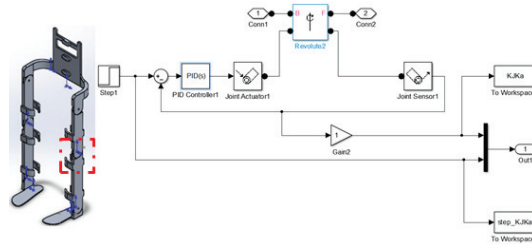

FIGURE 13. Modelling a knee joint of left leg using the PID controller.

Figure-15 shows the automatic tunning result in Matlab/Simulink. When the tuning is done, continued to press the run button, the result of knee joint actuator simulation with the final value at angle 20 degrees, and obtained data in the form of angular position response. In Figure-15, it is seen that in 0 to 1 seconds when it is not given a step, the system is very good without experiencing oscillation. Then, when it is given the step input of angle 20 degrees, seen the overshoot and undershoot at the time of first second, but after that the stable response again until the 3rd second. The response can already be said to work for a system, because the response is in accordance with the step value is stable at angle 20 degrees.
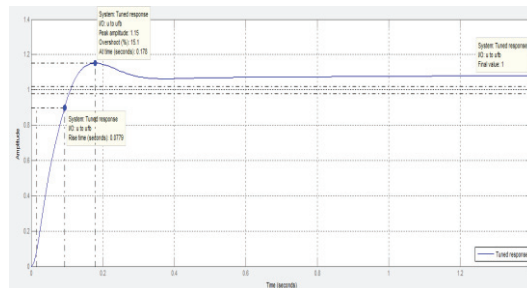

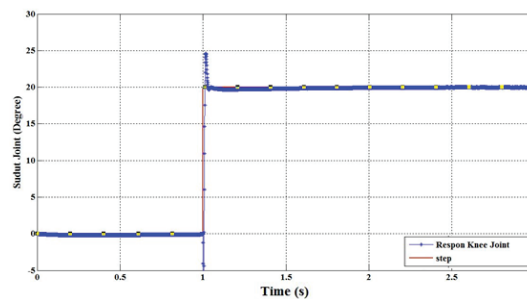**FIGURE 14.** Tunning result the PID controller of the knee joint of left leg.

FIGURE 15. Output result for the knee joint of left leg using the PID controller.

The Ankle Joint Of The Left Leg

This section presents the design of the PID controller for the angular position on the left leg joint ankle as shown in Figure-16. Next, the PID blocks within a plant of the system must be tuned in advance to obtain the ideal controller gain value. The results as shown in Figure-17.
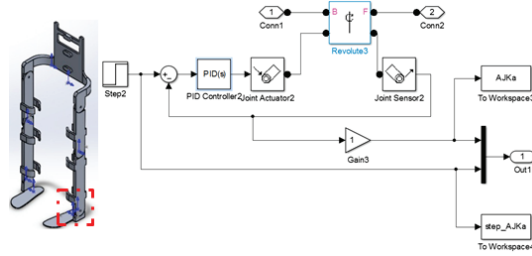
FIGURE 16. Modelling an ankle joint of left leg using the PID controller.

When the tuning is done, then proceed with the click of the run button, the simulation result of the ankle joint actuator with the final value of 20 degrees, and obtained data in the angular position response shown in Figure-18. Based on the picture, seen that at 0 to 1 seconds, when it is not given an input step, the response system follows the input without experiencing oscillations. Then, when it is given the input step, it will go to 20 degrees. Seen the existence of overshoot and undershoot at 1 to 1.3 seconds, and the next the system shows a stable response again until 3 seconds to the angle 20 degree. Then the system is steady-state.
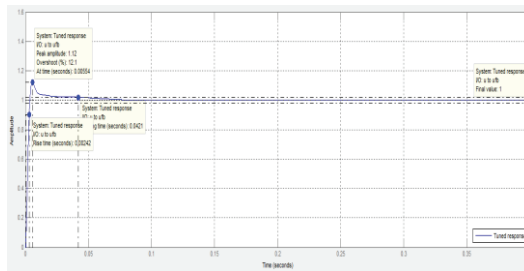

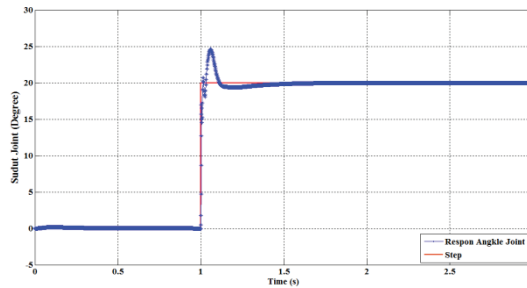**FIGURE 17.** Tunning result the PID controller of the ankle joint of left leg.

FIGURE 18. Output result for the ankle joint of left leg using the PID controller.

The Hip, Knee, And Ankle Joint Of The Right Leg

Furthermore, this section presents the design of the PID controller for the angular position of the hip joint of right leg shown in Figure-19. The PID block located within a plant of a system must be tuned to obtain the ideal gain control value, that is shown in Figure-20.
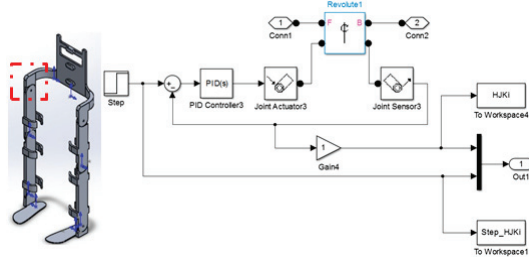
FIGURE 19. Modelling a hip joint of right leg using the PID controller.
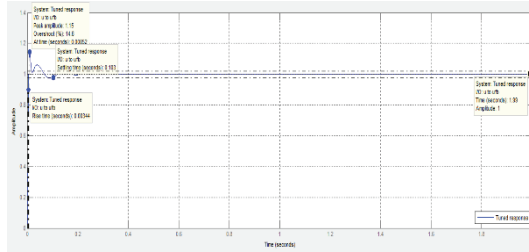

FIGURE 20. Tunning result the PID controller of the hip joint of right leg.

Same with the process for the other joint, when the tuning is done, then by clicking the run button, we get the result of a simulation of the hip joint actuator with the final value at the angle of 20 degrees. The angular position response produced is shown in Figure-21. Based on the figure, we can see that in 0 to 1 seconds when the system is not yet given an input step, then the system is still in stable condition, without experiencing oscillation. Then, when the system is given an input step, it will go to the angle of 20 degrees. Seen there is overshoot and undershoot at the first second, after which the response looks unstable in 1.25 to 2.8 seconds. Next stabilized again in seconds 2.8 to 3 seconds, which produces steady-state conditions at an output angle of 20 degrees in accordance with the input.
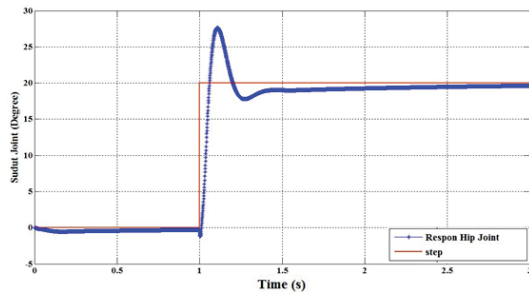

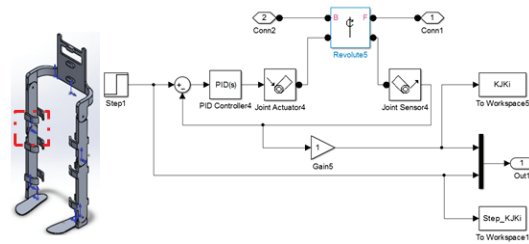**FIGURE 21.** Output result for the hip joint of right leg using the PID controller.

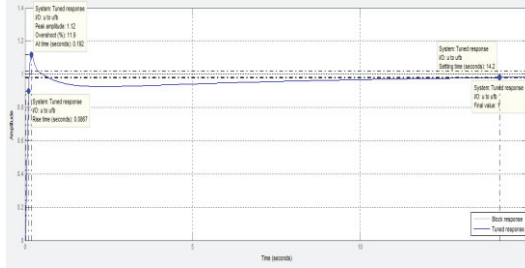**FIGURE 22.** Modelling a knee joint of right leg using the PID controller.

FIGURE 23. Tunning result the PID controller of the knee joint of right leg.
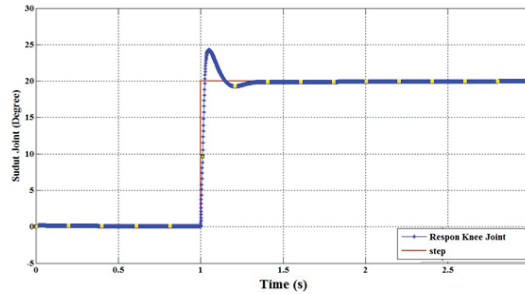

FIGURE 24. Output result for the knee joint of right leg using the PID controller.

The design of the PID controller for the angular position of the knee joint, as shown in Figure-22. The tunning system controller to obtain the best gain controller value. The results are shown in Figure-23. Furthermore, when the tuning process has been completed, then by pressing the run button, the result of the actuator simulation on the hip joint with the final value of the steady-state output at an angle of 20 degrees in accordance with the input. It is shown in Figure-24.
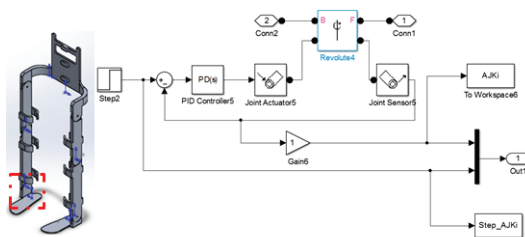

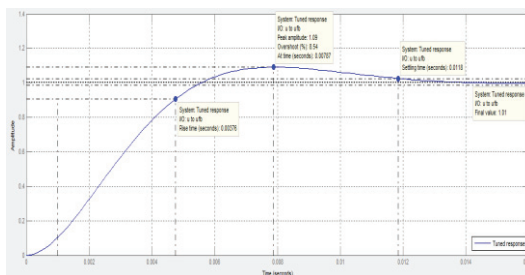**FIGURE 25.** Modelling an ankle joint of right leg using the PID controller.

FIGURE 26. Tunning result the PID controller of the ankle joint of right leg.
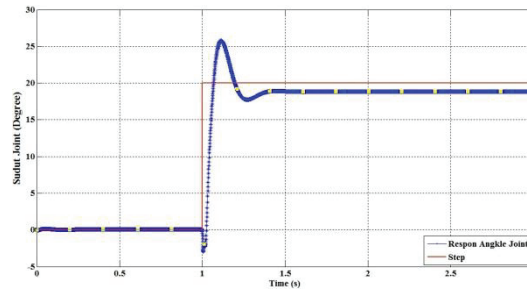
FIGURE 27. Output result for the ankle joint of left leg using the PID controller.

Next, the last joint is the ankle joint of the right foot. The design of the PID controller is shown in Figure-25. The PID block in a plant of the system must be tuned that is shown in Figure-26. When the tuning process has been completed, then we get the run button on Matlab, we get the simulation result of the ankle joint actuator with the output value of angle 20 degrees, according to the input. The angular position response is shown in Figure-27. It shows that the controller achieve a study-state condition.

CONCLUSIONS

In this paper, a lower limb exoskeleton robot for paraplegic patients has been designed. The design was optimized for assisted walking based on the average height and weight of Indonesian. The hip, knee, and ankle joints were considered in the proposed lower limb exoskeleton robot. The control algorithm was designed to cover all actuator joints, in which we used the PID controller. We have successfully used SimMechanic for a control modeling of the robot control without having to define its transfer function. The implementation of a control method will be tested and evaluated for paraplegic persons in the near future.

REFERENCES

1. K.L. Hollands , T.A. Pelton , S.F. Tyson , M.A. Hollands , and P. M. van Vliet , "Interventions for coordination of walking following stroke: Systematic review," in *Gait Posture*, 35 (3), (2012), pp. 349-359.
2. H. Kawamoto, Sankai, & H, "Power assist system HAL-3 for gait disorder person," in *Proceedings of the 8th International Conference on Computers Helping People with Special Needs*, (2002), pp. 196–203.
3. Khairul Anam and Adel Ali Al-Jumaily, "Active exoskeleton control systems: State of the art," in *Procedia Engineering,* 41, (2012), pp. 988 – 994.
4. Slavka Viteckova, Patrik Kutilek, Marcel Jirina, "Wearable lower limb robotics: A review," in *Biocybernetics and Biomedical Engineering*, 33, (2013), pp. 96 – 105.
5. Mohd Azuwan Mat Dzahir, and Shin-ichiroh Yamamoto, "Recent trends in lower-limb robotic rehabilitation orthosis: Control scheme and strategy for pneumatic muscle actuated gait trainers," in *Robotics*, 3, (2014), pp. 120-148.
6. Tingfang Yana, Marco Cempinia, Calogero Maria Oddoa, Nicola Vitiello, "Review of assistive strategies in powered lower-limb orthoses and exoskeletons," in *Robotics and Autonomous Systems*, 64, (2015), pp. 120–136.
7. Bing Chen, Hao Ma, Lai-Yin Qin, Fei Gao, Kai-Ming Chan, Sheung-Wai Law, Ling Qin, Wei-Hsin Liao, "Recent developments and challenges of lower extremity exoskeletons," in *Journal of Orthopaedic Translation*, 5, (2016), pp. 26-37.
8. Surachai Panich, "Design and simulation of leg-exoskeleton suit for rehabilitation," in *Global Journal of Medical Research*, 12 (3), (2012).
9. Jan F. Veneman, Rik Kruidhof, Edsko E. G. Hekman, Ralf Ekkelenkamp, Edwin H. F. Van Asseldonk, and Herman van der Kooij, "Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation," in *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15 (3), (2007), pp. 379 – 386.
10. Jung-Hoon Kim, Jeong Woo Han, Deog Young Kim and Yoon Su Baek, "Design of a walking assistance lower limb exoskeleton for paraplegic patients and hardware validation using CoP," in *International Journal of Advanced Robotic Systems*, 10, (2013), pp. 1–13.

11. Z. Y. Wong, A. J. Ishak, S. A. Ahmad, Y. Z. Chong, "Mechanical analysis of wearable lower limb exoskeleton for rehabilitation," in *Journal of Engineering Science and Technology Special Issue on Applied Engineering and Sciences*, (2014), pp. 107 – 114.

12. Pieter Beyl, Michael Van Damme, Ronald Van Ham, Bram Vanderborght and Dirk Lefeber, "Design and control of a lower limb exoskeleton for robot-assisted gait training," in *Applied Bionics and Biomechanics*, 6 (2), (2009), pp. 229–243.

13. Dalei Pan, Feng Gao, Yunjie Miao, Rui Cao, "Co-simulation research of a novel exoskeleton-human robot system on humanoid gaits with fuzzy-PID/PID algorithms," in *Advances in Engineering Software*, 79, (2015), pp. 36 – 46.

14. Artur Gmerek, "Mechanical and hardware architecture of the semi-exoskeleton arm rehabilitation Robot," in *Archive of Mechanical Engineering*, 4, (2013), pp. 557-574.