Proceedings of 2020 IEEE
International Conference on Mechatronics and Automation
October 13 - 16, Beijing, China

Design of Graphical User Interface for Motor Selection of the Lower Limb Exoskeleton

Jian Guo[1] and Fan Bu[1]

[1]*Tianjin Key Laboratory for Control Theory & Applications in Complicated Systems and Intelligent Robot Laboratory*

Tianjin University of Technology
Binshui Xidao Extension 391, Tianjin,China
jianguo@tjut.edu.cn; Ryan_w77@yeah.net

Shuxiang Guo[1,2]*

[2]*Department of Intelligent Mechanical Systems Engineering,*

Faculty of Engineering,
Kagawa University,
Takamatsu, Kagawa, Japan
**corresponding author: guo@eng.kagawa-u.ac.jp*

***Abstract* - The weight of motor is related to torque in exoskeleton. Designers need to install the motor on the exoskeleton. The weight and moment of inertia of the exoskeleton will change when the motor type is changed. For the purpose of determine the motor type, this paper developed a MATLAB Graphical User Interface (GUI) to complete forward kinematics solution, dynamic analysis through program writing and data interaction. The forward kinematics is solved by Denavit-Hartenberg (DH) parameters method, the dynamics is solved by Lagrange equation. General kinematics can be decomposed into multiple polynomials by Taylor series. Input polynomial coefficients into GUI to obtain motor torque curve data. The designer can then determine the motor power by the product of torque and rotational speed.**

Index Terms - Exoskeleton leg robot, Graphical user interface, Kinematic, Dynamics

I. Introduction

Legs play an important role in human movement. But in life, it is difficult to avoid the loss of walking ability due to accidents or diseases, such as stroke [1]. Due to leg disabilities, nearly 10 million people cannot walk [2]. More and more designers are beginning to pay attention to the development of lower exoskeleton [3]. The first step in the development of exoskeleton is to choose an electric motor. The motor is too small to drive the body [4]. If the capacity of the motor is too large, it will cause energy loss and shorten the charging interval. Correct choice of exercise is an important step in the development of lower limb exoskeleton.

In the 1960s, General Electric created the world's first electric-assisted robot for the development and testing of wearable electric-assisted exoskeletons: hydraulically driven Hadman [5]. In the 21st century, with the explosive growth of leg exoskeleton research, Berkeley University produced BLEEX (Berkeley Lower Extremity Exoskeleton) in 2004. Japan masters the core technique of robotics. The HAL series of robots was developed by the University of Tsukuba and has now reached the fifth generation. Honda has developed the U3X robot to reduce the workload of workers in the automobile production workshop. The development of exoskeleton robots is obvious. Exoskeleton robots are gradually full of life, and developing a suitable exoskeleton is a new challenge [6].

The shape design of the lower limbs is an important part of the lower limb assisted robot. The shape design and material selection of the lower limb robot will affect the parameters of the lower limb. Different materials and shapes determine the weight and rotational inertia of the lower limbs, which directly affects the choice of motor [7]. The different power of the motor, the weight and capacity of the motor are also different, which will affect the movement of the exoskeleton together with the material and shape. Therefore, reliable simulation software is an important condition to help the external skeleton [8].

In the above selection of robot motor torque, it is generally roughly obtained by multiplying torque by rotational speed. Or it can be solved by ADAM and other software, and the result is not good under the condition of MATLAB interaction [9].

The article consists of five parts. The first part is Introduction. In the second part, the objects of simulation design are introduced. The third part uses DH parameter method to solve kinematics. The fourth part introduces the Lagrange equation. At the same time, in the third and fourth parts, we will also add programming and interface design. In the fifth part of this article, this article will show the simulation results [10].

II. Model Structure

The main joints of the human lower extremities include the hip, knee and ankle joints. The ankle joint has little effect on human activity. It is set to fix the joint and will not be discussed. The human hip joint is a complex joint. Then the knee. Thigh and calf are about two links.

The hip joint consists of the femoral head and acetabulum. It belongs to the ball and socket joint and is a typical ball and socket joint. Its main functions are: internal rotation and external rotation. The knee joint consists of the lower end of the femur, the upper end of the tibia and the patella. It is the largest and most complex joint of the human body and belongs to the pulley joint.

Fig.1 Anatomical structure of hip joint

TABLE I
RELATIVE PARAMETERS OF THE LEG

Joint	Joint	range of motion (Degree)
Hip Joint	Adduction/Abduction	-30~40
	Flexion/Extension	-120~65
	Internal Rotation/External Rotation	-35~60
Knee Joint	Flexion/Extension	-60~0
	Valgus/Varus	-34~20

Simplify the simulation object in the simulation process. The knee joint is simplified as a rotational joint with one degree of freedom. The ankle is a fixed joint. The hip joint is considered as a rotating joint with three degrees of freedom. Mainly through the combination of two rotating joints. In this way, the lower limb exoskeleton with 4 degrees of freedom is obtained.

Then, the length of the connecting rod is set. Set the midpoint of the left and right hip joints as i_0, the two rotation joints of the right hip joint as i_1 and i_2, the axis of the rotation joint of the knee joint as i_3, and the center of the ankle joint as i_4. Among them, the distance between i_0 and i_1 is l_1, and the length of thigh is l_2 and the length of leg is l_3.

According to the Denavit-Hartenberg (DH) parameters method, every joint should have a corresponding XYZ coordinate system. In order to explain the design of the simulation system in this paper more directly, this paper will also directly stipulate the coordinate system on each point. See Fig.2 for details.

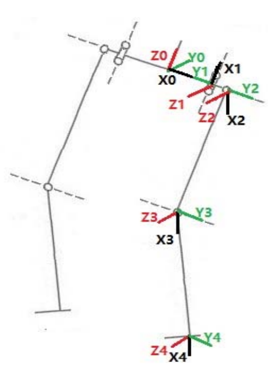

Fig.2 DH coordinate system setting

III. KINEMATIC ANALYSIS AND SIMULATION

Kinematics is an important part of robot simulation. Kinematics analysis is to analyze the relationship between the rotation angle of the rotary joint and the position of the connecting rod. Through kinematic analysis, the rotation angle of hip joint and knee joint can be transformed into the position of connecting rod. The rotation angle of hip joint and knee joint can be obtained by encoder, and the length of connecting rod is also known. If it is a two-dimensional plane, it can be obtained by the transformation of polar coordinate system and rectangular coordinate system. But for the three-dimensional plane is more complex. It can be solved by three-dimensional spherical coordinate system. But for robotics, there is a mature Denavit-Hartenberg (DH) parameters method to solve the problem. The kinematic simulation of this paper is also solved by DH parameter method [11].

There are two parts in kinematic solution. Forward kinematics analysis and inverse kinematics analysis. Inverse kinematics analysis is to calculate the joint rotation angle through the end position, length and origin of the connecting rod.

There are four important parameters in DH parameter method: $a_i, \alpha_i, d_i, \theta_i$. a_i is distance of (z_{i-1}, z_i) along x_{i-1}. α_i is angle of (z_{i-1}, z_i) about x_{i-1}. d_i is distance of (x_{i-1}, x_i) along z_i is angle of (x_{i-1}, x_i) about z_i. According to Fig.2, the following table can be obtained.

TABLE II
DH PARAMETER TABLE

Joint	θ_i(°)	d_i(mm)	a_i(mm)	α_i(°)
1	θ_1	0	l_1	90°
2	θ_2	0	0	90°
3	θ_3	0	l_2	0
4	θ_4	0	l_3	0

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & l_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & l_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

The operation of T matrix is the core of DH parameter method. T matrix contains a lot of information. The transformation matrix can be obtained with the parameters in Table II and formula 1.

$$ {}^{0}_{1}T = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & l_1\cos\theta_1 \\ \sin\theta_1 & 0 & -\cos\theta_1 & l_1\sin\theta_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2) $$

$$ {}^{1}_{2}T = \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_2 & 0 \\ \sin\theta_2 & 0 & -\cos\theta_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3) $$

$$ {}^2_3T = \begin{bmatrix} cos\,\theta_3 & -sin\,\theta_3 & 0 & l_2\,cos\,\theta_3 \\ sin\,\theta_3 & cos\,\theta_3 & 0 & l_2\,sin\,\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4) $$

$$ {}^3_4T = \begin{bmatrix} cos\,\theta_4 & -sin\,\theta_4 & 0 & l_3\,cos\,\theta_4 \\ sin\,\theta_4 & cos\,\theta_4 & 0 & l_3\,sin\,\theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5) $$

Through matrix point multiplication, multiple t-matrices can be connected, thus the whole lower limb system of robot can be reduced to a T-matrix, which simplifies the operation of computer simulation software. The solution of T matrix is as follows:

$$ {}^0_4T = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T = \begin{bmatrix} R & d \\ 0 & 1 \end{bmatrix} \quad (6) $$

The DH parameter method is far from simple. ${}^0_4T$ is a 4×4 matrix, and ${}^0_4T$ can be divided into four parts, as shown in formula 6, Where R is the attitude matrix and d is the end position [6].

$$ d = \begin{bmatrix} x_4 \\ y_4 \\ z_4 \end{bmatrix} \quad (7) $$

In formula 7, $x_4 y_4 z_4$ exactly represents the position of the robot leg end, which is exactly what the forward kinematics needs to solve.

The simulation design is divided into two parts: front end and back end. The front-end is the human-computer interface, the main function is to collect the data information that people input. The back end is mainly used for data calculation, and the information calculated by the back end needs to be returned to the front end. Sequential information interaction can save researchers' time. Through simple and visual information input, results can be obtained. And the program is open-source, through the upper derived forward kinematics, to calculate. Next, the interface design is introduced.

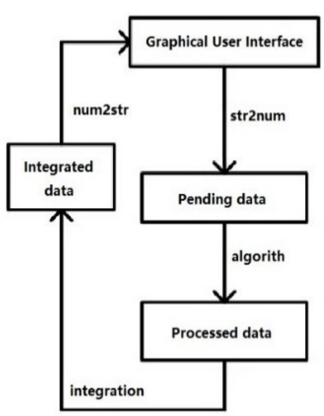

Fig.3 Data flow diagram

In Fig.3, the left side of the figure is the tool provided by MATLAB, the main role is to input kinematics data. Among them, axis1 is used to display the simulation diagram. And you can use text to prompt for input. The three boxes on the left of axis 1 are the control of the hip joint and the control of the knee joint in two directions. The twelve frames on the right of axis 1 are the motion descriptions of the two motors of the hip joint, the length of the thigh and the length of the lower leg, and the weight and rotational inertia. The researcher can set the parameters and press the calculation button. At (x, y, z), we obtain ankle coordinates. By pressing the dynamics button, the motor output torque curve is obtained. And the design of MATLAB is open source, we can access the source program and transplant the system algorithm.

The data sent to the background through the GUI is string data, which needs to be extracted after pressing the button, and type conversion. 2 in num2str means "to", which is a unique transformation statement in MATLAB. In data output, you need to convert array data to string data. After the transformation, x, y and z need to be spliced, so that the kinematic calculation can be completed.

The simulation results need to be tested. At present, Adams and ANSYS with better simulation are not suitable for simple robot simulation [12]. Finally, choose MATLAB's Robotic Toolbox for simulation testing [13].

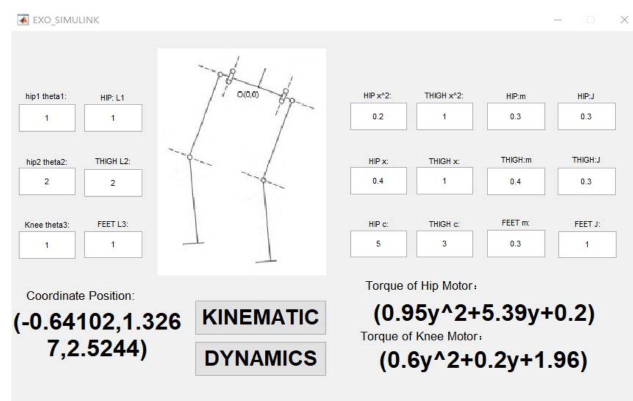

Fig.4 Graphical user interface

IV. Dynamics Analysis And Simulation

Next is the analysis and simulation of dynamics, which is also an essential part of robot control. The introduction of dynamics enables the control robot to follow the required trajectory more accurately.

The main analysis of dynamics: the relationship between the force acting on an object and the speed of the object [14]. This part of the analysis is mainly: the conversion of motor torque and exoskeleton inertia, gravitational potential energy and speed. Accurate analysis makes the control speed more accurate. Newton's first law can explain the solution to this part of the problem:

$$ \vec{F} = \sum_i \frac{dm_i \vec{v}_i}{dt} $$

In the power assisted robot, the motor usually works by rotating. Newton's law is linear thrust, so it should be extended to Euler's second law of motion. Euler's second law will lead to the calculation of tensor, so it is difficult to solve the kinematic problem through Euler's second law.

$$\vec{\tau} = I\vec{\alpha} + \vec{\omega} \times I\vec{\omega}$$

In order to solve the complex calculation of Euler's second law, it can be solved by Euler-Lagrange equation.

For this part of the analysis, Ignore the link between the two rotating joints of the hip joint. The rest is thigh and calf, hip joint fixation. Two link mechanism is formed. The output torque of hip joint motor is M_1, the output torque of knee joint motor is M_2, θ_1 is the rotation angle of hip joint motor, $\dot{\theta_1}$ is the speed of rotation, and $\ddot{\theta_1}$ is the acceleration of rotation. O is hip joint, A is knee joint, B is ankle joint.

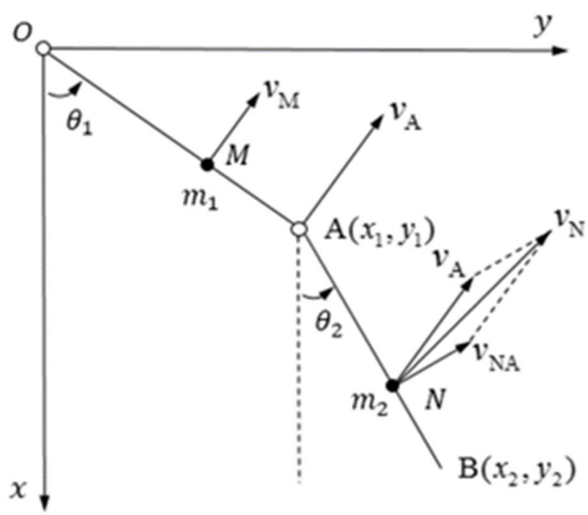

Fig.5 Dynamic analysis

In the whole system, all motion energy is output by the motor. It is mainly converted into kinetic energy and potential energy, and potential energy is mainly used to work against gravity. The kinetic energy is divided into two parts, one is the kinetic energy of linear motion, the other is the energy of the rotation of the connecting rod. The energy formula can be expressed as follows:

$$E = \frac{1}{2}m_1 v_M^2 + \frac{1}{2}m_2 v_N^2 + \frac{1}{2}J_1\omega_1^2 + \frac{1}{2}J_2\omega_2^2 \quad (8)$$

The part of potential energy is mainly divided into thigh and calf raising, work done to overcome gravity:

$$V = -m_1 g l_{MA} \cos\theta_1 - m_2 g(l_1 \cos\theta_1 + l_{NB} \cos\theta_2) \quad (9)$$

To construct Lagrange function:

$$L = E - V \quad (10)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q_j}}\right) - \frac{\partial L}{\partial q_j} = M_j \quad (11)$$

q in the above formula is the coordinate. The output torque of the motor, M_1, M_2, can be obtained by taking L into the Lagrange equation:

$$M_1 = (m_1 l_{MA}^2 + m_2 l_1^2 + J_1)\ddot{\theta_1} + m_2 l_1 \cos(\theta_2 - \theta_1)\,\ddot{\theta_2}$$

$$-m_2 l_1 l_{NB} \sin(\theta_2 - \theta_1) + (m_1 l_{MA} + m_2 l_1) g \sin\theta_1 \quad (12)$$

$$M_2 = m_2 l_1 l_{NB} \cos(\theta_2 - \theta_1)\,\ddot{\theta_1} + (m_2 l_{NB}^2 + J_2)\ddot{\theta_2}$$

$$+m_2 l_1 l_{NB} \sin(\theta_2 - \theta_1)\,\dot{\theta}_1^2 + m_2 l_{NB} g \sin\theta_2 \quad (13)$$

It should be noted here that θ_1 is a continuous function, and the same is true for θ_2 minus θ_1, so $sin(\theta_2 - \theta_1)$ needs to be expanded by a binary Taylor function, which can reduce the amount of computer operations by sorting out the formula in advance.

At present, the motion of each joint can be represented by a polynomial. Therefore, only the coefficients of the polynomial need to be input, and the polynomial of the motor torque can be obtained through calculation and sorting. In addition, general polynomials are infinitely differentiable, so Taylor functions can be used here.

Inverse dynamics solution is the process of obtaining M_1 and M_2 by knowing the output torque of the motor. It is the process of solving the differential equation. The real-time requirement of simulation is not high, so it can be solved directly by the Runge-Kutta method built in MATLAB. The solution statement is "ode45". The differential equation solution will not be described here.

V. Experiments and Results

In this paper, the experimental results are tested through the Robotic Toolbox, version 9.10.

A. Experimental setup

In order to reflect the accuracy of the simulation, the human gait data is quoted.

TABLE III
Clinical Gait Analysis [8]

Time(s)	Hip Angle (°)	Knee angle (°)
0.01	13.368	24.548
0.14	11.999	23.623
0.27	0.379	20.110
0.4	-8.323	18.528
0.53	0.052	49.299
0.66	22.131	79.426
0.79	36.628	66.754
0.92	32.056	15.361
1.05	29.253	17.528
1.18	25.053	28.011
1.25	16.979	25.678

According to the data to test the input of simulation, detect the error of data output, so as to analyze the simulation accuracy. The following table shows the data of human walking at 4km/s. It is necessary to convert the data into radian system, and process it according to DH parameter method. Facilitate the next step.

Through the Robotic Toolbox, the same model of the lower limb assist robot as the simulation is established [16].

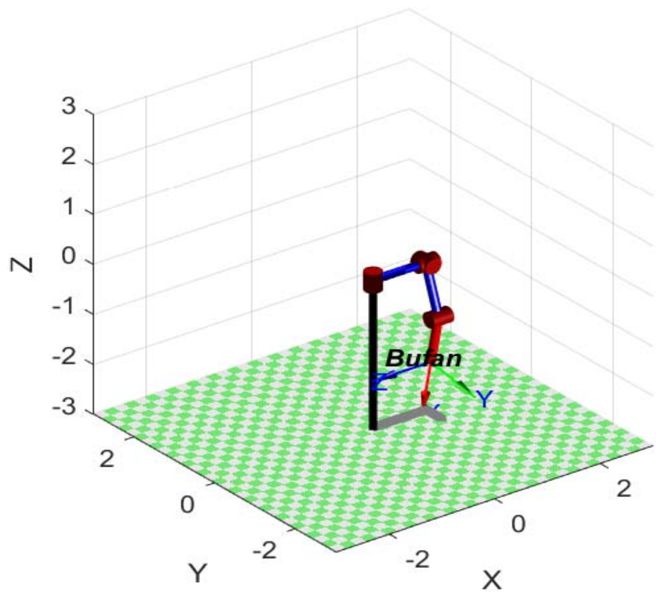

Fig.6 Robotic Toolbox Model

In order to obtain more data points, test the error. The motion curve of hip joint and ankle joint can be obtained by curve fitting. In this paper obtains Fig.7 and Fig.8 by 6th order curve fitting.

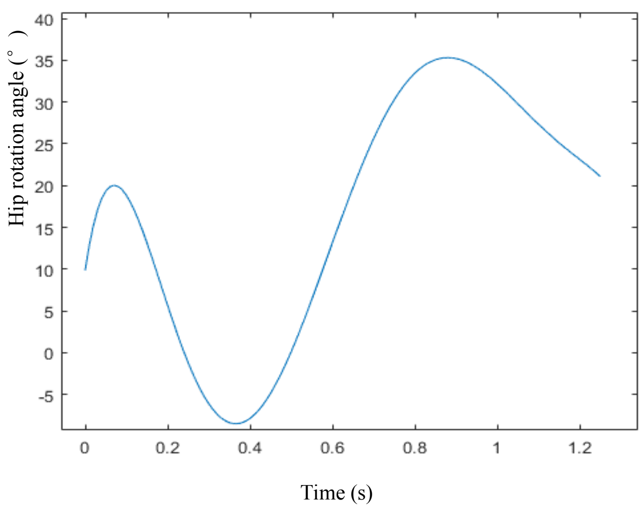

Fig. 7 Fitting function of hip joint rotation angle.

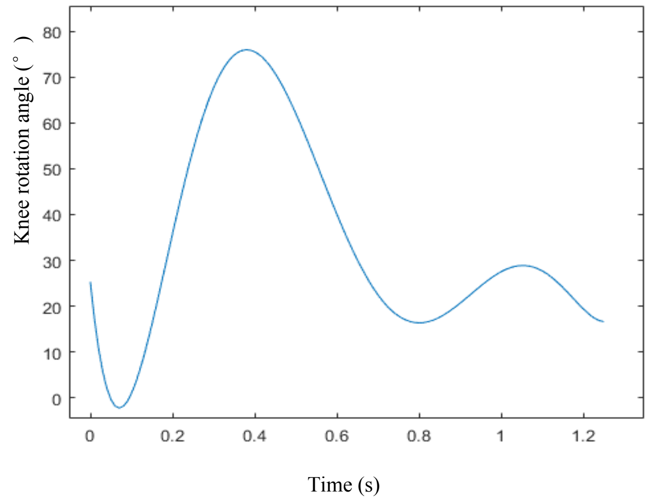

Fig. 8 Fitting function of knee joint rotation angle.

Through the function of fkine, the kinematics is solved, and the data is saved for the comparison of simulation results [19].

B. Experimental results and analysis

According to the motion data of Fig.7 and Fig.8, several motion angles of the hip joint and the ankle joint are selected, and the terminal coordinates can be obtained through the calculation of GUI and Robotic Toolbox, and the coordinates are on the XOZ plane.

Set the length of thigh and calf to 1m, bring in two curves in Fig.7 and Fig.8, and solve them in the Robotic Toolbox and GUI respectively. According to the data obtained, the comparison of drawings is carried out [21]. Fig.10 is available:

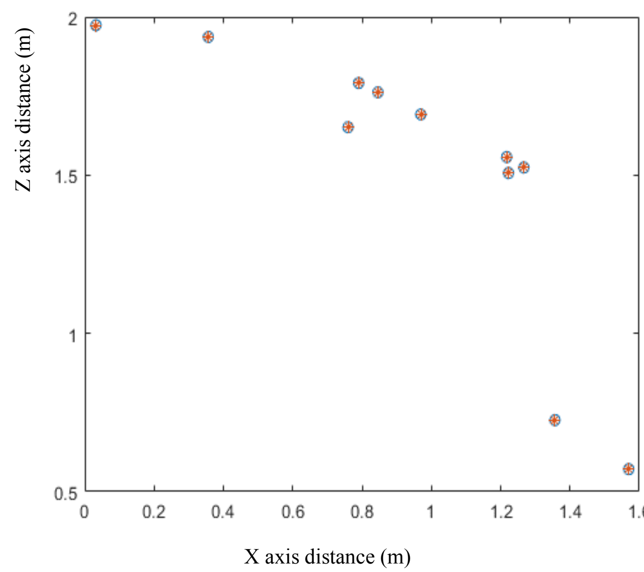

Fig. 9 Distance between ankle and origin.

'*' represents the kinematics result of Robotic Toolbox, 'O' represents the GUI kinematics result. It can be seen from Fig.9 that the Robotic Toolbox and GUI results are completely coincident, which verifies the accuracy of the kinematics of GUI.

After inputting other parameters such as the motion trajectory and weight of the ankle and hip joints in the dynamics simulation, the torque that can be output by the motor of the Lagrange solution can be obtained. The torque multiplied by the rotation speed is the power required. Generally, robots require large torque and low speed, so they need mechanisms such as reducers. The energy loss needs to be considered in the calculation of power [22].

VI. CONCLUSION AND FUTURE WORK

In this paper, a new type of exoskeleton assisted lower limb robot simulation is proposed, which is used to help developers more convenient, fast and visual simulation. The simulation can be used for kinematic and dynamic analysis. Through the quality and length of the preliminary selection of motor power. The simulation mechanism has four degrees of freedom movement, which can basically realize the human leg movement. It is convenient for the developer to select the motor and determine the movement range of the lower limb robot. In the experiment, it is proved that the simulation is reliable and accurate through the verification of Robotic Toolbox. In conclusion, the research draws the following conclusions:

1) The new simulation design can directly understand the situation of the lower limb robot through the interface meter.
2) The simulation can help designers to determine the motor parameters and the movement of lower limbs.
3) The simulation design is based on the open source platform, which can be simply transplanted, and add new functions, laying the foundation for future development and improvement.

In the future work, we will add more complex functions and turn them into better simulation software. For example: joint simulation with other software, as well as various inverse kinematics and inverse dynamics solutions. The Jacobian matrix is added to make the simulation more accurate for speed control. At the same time, simplify the algorithm, use less storage space and calculation times to get the accurate answer.

REFERENCES

[1] Wu. H, Jia. T, Li. N, Wu. J, Yan. L, Study on the control algorithm for lower-limb exoskeleton based on ADAMS Simulink cosimulation. *Journal of Vibro-engineering*, Vol. 19, no. 4, pp. 2976–2986, June 2017.

[2] Shuxiang Guo, Wu Zhang, Jian Guo, et al. "Design and kinematic simulation of a novel exoskeleton rehabilitation hand robot", *IEEE International Conference on Mechatronics and Automation,* pp. 1125-1130, 2016.

[3] Shuxiang Guo, Zhi Wang, Jian Guo. Study on Motion Recognition for a Hand Rehabilitation Robot Based on sEMG Signals, *International Conference on Mechatronics and Automation,* pp. 1061-1066, 2019.

[4] Shuxiang Guo, Zhi Wang, Jian Guo, Qiang Fu and Nan Li, "Design of the Speech Control System for a Upper Limb Rehabilitation Robot Based on Wavelet De-noising," *International Conference on Mechatronics and Automation*, pp. 2300-2305, 2018.

[5] Swaraj Zodey, Sharad K. Pradhan, Matlab Toolbox for Kinematic Analysis and Simulation of Dexterous Robotic Grippers, *Procedia Engineering*, Vol 97, no.4 , Pages 1886-1895. May 2014

[6] S. Middleton, C. Levi, S. Dale et al. Triage, treatment and transfer of patients with stroke in emergency department trial (the T-3 Trial): *A cluster randomised trial protocol*, Implement. Sci. Vol.11, no.1, pp. 567–578, January 2016.

[7] Leiyu Zhang, Jianfeng Li, et al. Design and Kinematic Analysis of Co-Exoskeleton with Passive Translational Joints for Upper-Limb Rehabilitation, *International Journal of Humanoid Robotics*, Vol. 15, no. 5, pp. 24, January 2018.

[8] Y. Long, Z.-J. Du, C.-F. Chen, W. Wang, L. He, X. Mao, G.-Q. Xu, G. Zhao, W. Dong, Design kinematics and dynamics modeling of a lower limb walking assistant robot. *in Proceedings of the 4th International Conference on Robotics and Mechatron-ics*, Vol. 8, no. 2, pp. 26–28, October 2016.

[9] Anjali S. Nair, D. Ezhilarasi, Performance Analysis of Super Twisting Sliding Mode Controller by ADAMS–MATLAB Co□simulation in Lower Extremity Exoskeleton. *International Journal of Precision Engineering and Manufacturing-Green Technology*, Vol. 7, no. 4, pp. 743–754, March 2020.

[10] Dang X., Z. Zhou, L. S., Liao L. P., Liang D, Modelling and Simulation of Forward Kinematics for Planar 3-DOF Parallel Robot Based on Simulink. *Applied Mechanics and Materials*, Vol. 4, no. 4, pp. 567–578, April 2013.

[11] V.M. Akhil, M. Ashmi, P.K. Rajendrakumar, K.S. Sivanandan, Human Gait Recognition Using Hip, Knee and Ankle Joint Ratios, *Innovation and Research in Biomedical Engineering*, Vol. 41, No. 3, pp. 133-140, 2020.

[12] Khusainov R, Shimchik I, Afanasyev I, Magid E, 3D Modelling of Biped Robot Locomotion with Walking Primitives Approach in Simulink Environment. *Lecture Notes in Electrical Engineering*, Vol. 11, no. 13, pp. 287–304, April 2016.

[13] Cao Yanfei, Shi Tingna, Yan Yan, Braking Torque Control Strategy for Brushless DC Motor With a Noninductive Hybrid Energy Storage Topology, *IEEE Transactions on Power Electronics,* Vol. 35, no. 8, pp. 8417-8428, February 2020.

[14] Deng Mingdi, Li Zhijun, Kang Yu, et al. A Learning-Based Hierarchical Control Scheme for an Exoskeleton Robot in Human-Robot Cooperative Manipulation, *IEEE Transactions on Cybernetics,* Vol. 50, no. 1, pp. 112-125, January 2020.

[15] Rifai Hala, Mohammed Samer, Djouani Karim, et al. Toward Lower Limbs Functional Rehabilitation Through a Knee-Joint Exoskeleton, *IEEE Transactions on Control Systems Technology,* Vol. 25, no. 2, pp. 712-719, November 2017.

[16] Thieme H, Mehrholz J, Pohl M, et al, Mirror therapy for improving motor function after stroke. *Stroke*, Vol. 44, no. 1, pp. 112-119 2013.

[17] Giovacchini Francesco, Vannetti Federica, Fantozzi Matteo, et al, A light-weight active orthosis for hip movement assistance, Robotics and Autonomous Systems, Vol. 73, SI, pp. 123-134, September 2015.

[18] Shuxiang Guo, Lin Yan, Jian Guo, "Force feedback-based robotic catheter training system for the vascular interventional surgery," *IEEE International Conference on Mechatronics and Automation*, pp. 2197-2202, 2016

[19] Jian Guo, Nan Li, Shuxiang Guo, et al. "A LabVIEW-based human-computer interaction system for the exoskeleton hand rehabilitation robot" *IEEE International Conference on Mechatronics and Automation,* pp. 571-576, 2017.

[20] Jian Guo, Nan Li, Shuxiang Guo, et al. "A VR-based Upper Limb Rehabilitation Hand Robotic Training System" *IEEE International Conference on Mechatronics and Automation,* pp. 2364-2369, 2018.

[21] Ziyi Yang, Shuxiang Guo, Yi Liu, et al, An intention-based online bilateral training system for upper limb motor rehabilitatione, *Microsystem Technologies*, June 2020. Microsyst Technol (2020). https://doi.org/10.1007/s00542-020-04939-x

[22] Songyuan Zhang, Qiang Fu, Shuxiang Guo, Yili Fu. A Telepresence System for Therapist-in-the-Loop Training for Elbow Joint Rehabilitation. *Appl. Sci*. 1710, April 2019.