Trajectory Planning Based on Dual Torque Feedforward Control for Robot Astronaut in Space Station

Jiang Zhihong, Xu Jiafeng, Li Hui*, and Huang Qiang *Member, IEEE*

Abstract— **The robot astronaut is a multi-body dynamics system with multi-degrees of freedom, nonlinearity and strong coupling. Its motion control and operation target are in micro-gravity environment, in which the robot is in a natural floating state. In space station, the dynamics of the robot is influenced by the internal force, contact force, inertial force and joint friction. At this point, the collision between the robot and the space station due to the movement impact will greatly influence the robot stability as well as cause serious damage. In this paper, we propose a control strategy based on dynamics and force compliant dual feed forward torque compensation, which greatly eliminates the impact force during the collision and realizes the smoothly motion of the robot in space station. Simulation results validated the effectiveness of the control strategy.**

I. Introduction

Robot astronaut has a similar appearance to the astronaut and can replace the astronaut to complete various tasks without changing the environment of the space station. Therefore, it is of great significance to use the robot astronaut to assist or even replace the astronaut to perform the space station. The space environment is characterized by microgravity, astronauts are naturally floating in this environment.

The robot astronaut is a highly nonlinear mechanical system coupled in parallel by several subsystems[1]. It is affected by internal forces, inertia forces, and contact stresses in the space station. Under this condition, when the robot astronaut moves in the space station, due to the microgravity and narrow space of the space station, uncertain dynamics interactions and collision forces are serious threats to control stability, even small disturbances may cause significant The movements of the robots make the robot astronaut's dynamics very complicated and it is difficult to maintain a stable movement.

Through analyzing the robot astronaut's force impact with the space station during the movement of the space station, this paper performs a generalized multi-point impact dynamics modeling of the robot and divides the robot legs into a master leg and a slave leg. Then a torque dual feed forward control strategy based on force compliance compensation and dynamics feedforward compensation is proposed. This control strategy includes different control methods for the master leg and slave leg. To ensure the stability of the robot astronaut[2], the master leg is used position control and the adaptive impedance controller based on forgetting factor function was used to control the slave leg. Finally the experiments were performed to verify that the control strategy proposed in this paper greatly reduces (or even eliminates) the impact force of the robot during move in the space station.

Supported by the National Natural Science Foundation of China (61733001, 61573063, 61503029, U1713215).
✉ Author for correspondence, Lecturer, Ph. D.
E-mail: lihui2011@bit.edu.cn

II. Modeling For Robot Astronaut

A. Robot Astronaut Principle Prototype

Essentially, the robot astronaut is a humanoid robot with limbs and torso, and a head that provides visual feedback. In order to make the movement of the robot in the working space more in line with the engineering principles of human movement, the prototype of the robot limbs is designed as a redundant robot arm with 7 degrees of freedom[1]. The existence of redundant degrees of freedom makes the movement of the robot more flexible and can avoid singularities, which are of great significance for path optimization and control optimization. The prototype model of the robot astronaut is shown in Figure 1.

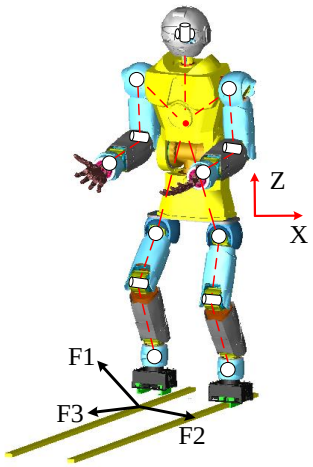

Figure 1. The principle prototype of robot astronaut

B. Dynamics Modeling of Robot Astronaut

The dynamic equation structure based on Newton-Euler equation[4] can be described as

$$\tau = M(\Theta)\ddot{\Theta} + V(\Theta,\dot{\Theta}) + G(\Theta) \tag{1}$$

For a robotic astronaut with four limbs, its dynamics model can be seen as two parts: ‘the right leg fixed - the left leg swing’ and ‘the left leg fixed - the right leg swing’, and

their combination can fully describe the dynamics characteristics of the robot astronaut in motion.

$$\tau_{robot} = \tau_{left} + \tau_{right} \tag{2}$$

The limbs of the robot are coupled and interact with each other. The movement of each part will affect other parts. It is difficult to model directly. Therefore, the model needs to be processed in a specific method. This paper uses the idea of equivalent substitution. Taking the left leg as an example, the robot can be seen as four arms connected in parallel on a waist joint. Therefore, the waist joint becomes a link connecting the four limbs. The movement of the waist can be forward recursive from the fixed base.

$$\begin{bmatrix} \upsilon \\ \omega \end{bmatrix}_{waist} = {}^{0}J(\Theta)\dot{\Theta} \tag{3}$$

And the waist movement can be divided into three separate recursive paths to get the motion of other joints.

$$\begin{bmatrix} \upsilon \\ \omega \end{bmatrix}_{end-i} = {}^{waist}J(\Theta)\dot{\Theta}\cdots\cdots(i=1,2,3) \tag{4}$$

In reverse recursion, the robot is equivalent to a robotic arm, the waist is equivalent to the end link, the right leg and arms are equivalent to three different actuators connected in parallel on the end link. The forces and moments generated by the actuators are integrated through coordinate transformation into external force acting on the end link.

$$\tau_{waist} = \begin{bmatrix} f \\ n \end{bmatrix}_{waist} = \sum_{i=1}^{3} \begin{bmatrix} {}^{waist}R_{end-i} f_{end-i} \\ {}^{waist}R_{end-i} n_{end-i} \end{bmatrix} \tag{5}$$

C. Master-slave leg collision mechanics modeling

When the robot is moving along the handrail fixed on the space station, the robot astronaut will easily make rigid contact with the environment (including multi-point impact),as shown in Figure 2. which can resulting in a large contact force.

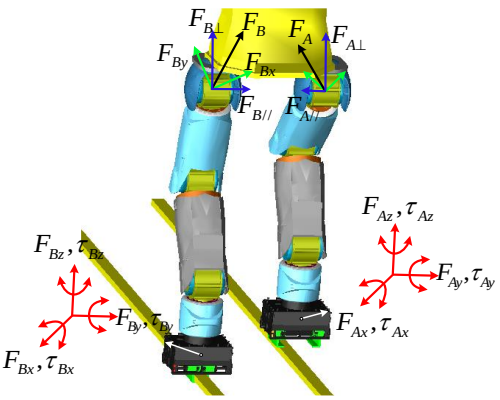

Figure 2. Multipoint impact model for robot astronaut

In the microgravity environment, even small disturbances have a great influence on the robot's posture. Therefore, establishing a multi-point impact dynamics model is one of the foundations of the robot astronaut's stability control in the space station.

When the robot grabs the handrail of the space station, the robot's legs form a rigid closed chain with the space station. At this time, because the rigidity of the robot and the space station are generally different, it often generates a large impact force, causing the robot to oscillate or even lose stable. In order to solve the internal force caused by the impact, this paper defines the legs as the master leg and the slave leg, and adopt different control strategies respectively.

Forces acting on the waist can be derived from the contact force and torques using a statics analysis[5]. The recursion of the force from the $i+1$ th joint to the i th joint is satisfied:

$${}^{i}f_i = {}_{i+1}^{i}R\,{}^{i+1}f_{i+1} \tag{6}$$

$${}^{i}n_i = {}_{i+1}^{i}R\,{}^{i}n_{i+1} + {}^{i}P_{i+1} \times {}^{i}f_{i+1} \tag{7}$$

The force applied on leg A (F_A) is derived and projected onto the XOY plane; then it is decomposed into vertical and parallel forces:

$$F_{A\perp} = F_{Ax}\sin\theta_A - F_{Ay}\cos\theta_A \tag{8}$$

$$F_{A//} = F_{Ax}\cos\theta_A + F_{Ay}\sin\theta_A \tag{9}$$

Similarly, the force applied on leg B (F_B) is decomposed as follows:

$$F_{B\perp} = F_{Bx}\sin\theta_B - F_{By}\cos\theta_B \tag{10}$$

$$F_{B//} = F_{Bx}\cos\theta_B + F_{By}\sin\theta_B \tag{11}$$

The Leg with large internal forces need to be adjusted because it create conflicting forces in the closed chain. As a result, leg with larger vertical forces is defined as slave leg, and the other is defined as the master leg. Comparing $F_{A//}$ with $F_{B//}$: If $F_{A//} < F_{B//}$, then leg A is the master and leg B is the slave and vice versa.

III. Track Planning Based on Dual Torque Feedforward Control Strategy

The existence of space microgravity makes it impossible for the robot to walk like a robot on the ground. Therefore, the robot astronaut forms a certain fixed relationship with the space station body through the "claw" on the foot. The existence of such a fixed relationship makes the robot astronaut's movement to have a rigid collision with the space station during the movement, which greatly affects the stability of the movement. Therefore, the robotic astronaut's trajectory planning also needs to be optimized to suit the special arrangement of the space station.

A. Gait Planning under Dynamic Equilibrium Theory

ZMP[6], full name Zero Moment Point, refers to the zero-moment point of the humanoid robot. As long as the zero-moment point of the robot falls within the range of supporting feet, the robot can walk stably[7]. The relationship between the landing point of the ZMP and the stability of the robot is shown in Figure 3.

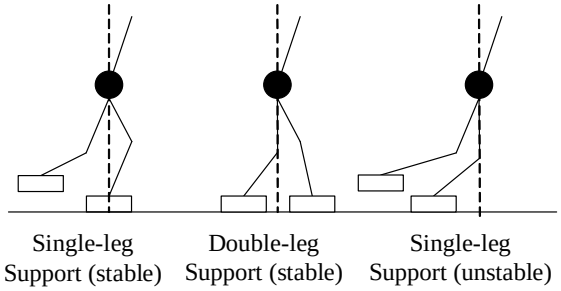

Figure 3. Dynamic equilibrium theory

The robot astronaut can be seen as a humanoid robot formed by multiple subsystems connected in parallel on the body. The subsystems are coupled and interact with each other, so their dynamic behavior is very complicated. Although the robot astronaut have a fixed relationship formed by the claws with the space station, when the robot moves in an unstable posture, the robot's feet will be subject to large bending moments and torques, which may cause imbalance of movement. And this will greatly reduce the useful life of the robot, so it is necessary to optimize the movement of the robot astronaut. In combination with the gait of the human walking[8], we can generate the robot gait as shown in the figure below.

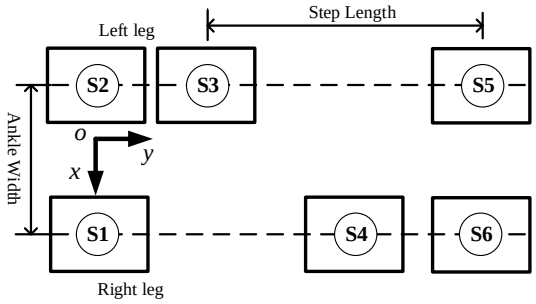

Figure 4. A complete gait from S1 to S6 of robot astronaut

As shown in the Figure 4. , our planned robot gait [9] can be divided into four steps. In the first step, the left leg falls in S2 and the right leg falls in S1. In this step, the robot legs will natural bend and prepare for further movement. Then comes to the second step, the left leg moves from S2 to S3 while the right leg still in S1. In the third step, the right leg moves from S1 to S4 and the left leg still in S3. Repeating the third step, the robot goes forward again and the left leg moves to S5 while the right leg in S4. Finally is the fourth step, the right leg stops at S6, and the robot astronaut moves from S1, S2 to S6, S5.

Analyzing from another aspect, the above four steps can be described as four stages: “double leg support period”, “right leg support period”, “left leg support period”, “double leg support period”. And we can get the walk mode in the follow Figure 5.

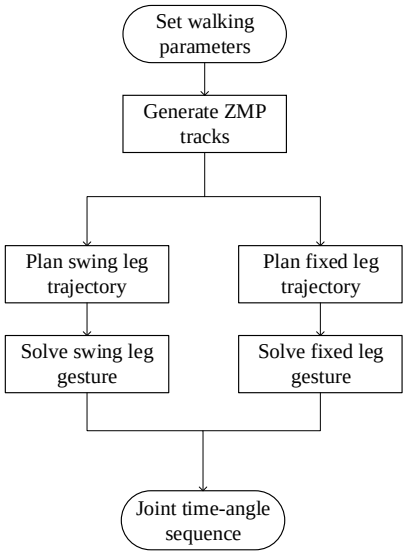

Figure 5. Walk mode generation flow chart

B. Master-slave Adaptive Impedance Controller

When the robot does not grab the handrail, the legs of the robot are symmetrically distributed in the desired position, as shown in Figure 6,

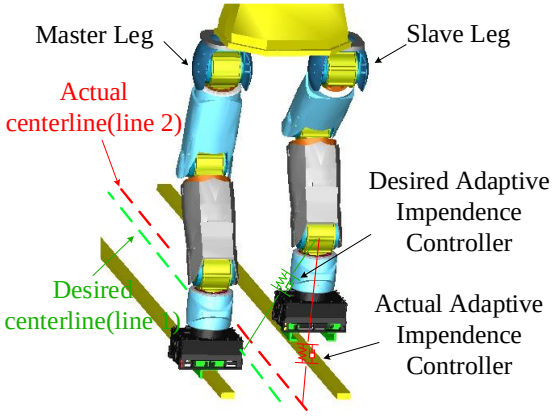

Figure 6. Principle of master-slave selection

The green line (line 1) is the desired centerline line[10]. However, due to uncontrollable multi-point collisions and errors due to mechanical deformation, the robot is usually at the red line (line 2) in an asymmetrical state. Because the red line and the green line do not coincide, the distance between them causes internal force in the impedance controller[9]. This internal force is the source of the impact force in the closed chain of the robot.

To solve this problem, the path planning strategy of the main leg is to control the robot toward a desired symmetrical posture. At the same time, under the premise of the stable movement of the robot, the equilibrium position of the virtual spring in the impedance control is gradually adjusted by the forgetting factor function, and the conflicting force is eliminated.

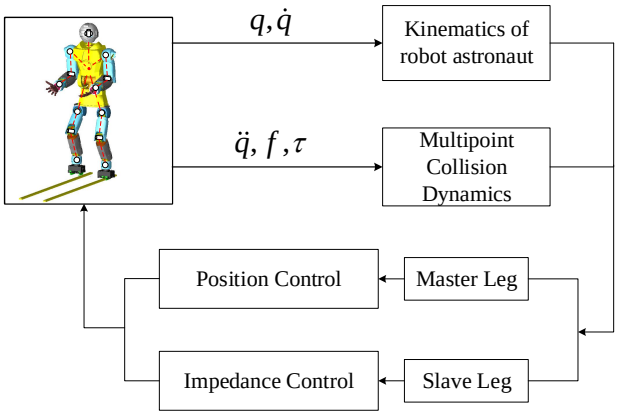

Figure 7. Master-slave adaptive impedance controller law

According to the Symmetrical configuration of the legs, the impedance equation of the legs can be written as[12]

$$\mathrm{M_S}(\ddot{\chi}_{des} - \ddot{\chi}) + \mathrm{B_S}(\dot{\chi}_{des} - \dot{\chi}) + \mathrm{K_S}(\chi_{des} - \chi) = \mathrm{Q}_{des} - \mathrm{Q}_{ext} \quad (12)$$

Where Q_{des} represents the expectation forces and torques of the slave leg, and Q_{ext} represents the detecting forces and torques by the six-dimensional force sensor at the end effector. Q_{ext} is written as

$$\mathrm{Q}_{ext} = [{}^{7}f_7^T, {}^{7}n_7^T]^T \quad (13)$$

Set e as the error between the equilibrium position χ_{des} and the actual position χ of the virtual spring

$$e_s(\mathrm{k}) = \chi_{des}(\mathrm{k}) - \chi(\mathrm{k}) \quad (14)$$

At the same reason

$$\Delta\mathrm{F(k)} = \mathrm{F}_{des}(\mathrm{k}) - \mathrm{F}_{ext}(\mathrm{k}) \quad (15)$$

Then, Eq. (12) can be rewritten as:

$$\mathrm{M_S(k)}\ddot{e}_s(\mathrm{k}) + \mathrm{B_S(k)}\dot{e}_s(\mathrm{k}) + \mathrm{K_S(k)}e_s(\mathrm{k}) = \Delta\mathrm{F(k)} \quad (16)$$

The spring damping system should be operated under critical damping conditions:

$$\mathrm{B}_s(\mathrm{k}) = 2\sqrt{\mathrm{K}_s(\mathrm{k}) - \mathrm{M}_s(\mathrm{k})} \quad (17)$$

Substituting Eq. (17) into Eq. (16),

$$\ddot{e}_s(\mathrm{k}) + 2\sqrt{\frac{\mathrm{K_S(k)}}{\mathrm{M_S(k)}}}\dot{e}_s(\mathrm{k}) + \frac{\mathrm{K_S(k)}}{\mathrm{M_S(k)}}e_s(\mathrm{k}) = \frac{\Delta\mathrm{F(k)}}{\mathrm{M_S(k)}} \quad (18)$$

Further simplified

$$\mathrm{H(k)} = \sqrt{\frac{\mathrm{K_S(k)}}{\mathrm{M_S(k)}}} \quad (19)$$

And the get the impedance equation for dual legs

$$\ddot{e}_s(\mathrm{k}) = \frac{\Delta\mathrm{F(k)}}{\mathrm{M_S(k)}} - e_s(\mathrm{k})\mathrm{H}^2(\mathrm{k}) - 2\dot{e}_s(\mathrm{k})\mathrm{H(k)} \quad (20)$$

C. *Forgetting Factor Function for Adaptive Impedance Controller*

Classical impedance control methods (as described above) can't eliminate the inherent internal forces of the closed chain. This section introduces the forgetting factor function [13][14][15][15] into the impedance controller; see Eq. (21). As the robot astronaut moves, the provided forgetting factor function can be used to adjust the virtual spring position to the expected motion state in the impedance controller, thereby eliminating the conflicting forces in the rigid closed chain.

$$u_{n+1}(t) = (1 - r(n))u_n(t) + r(n)u_0(t) + [\Gamma_1 \quad \Gamma_2][\begin{matrix}\dot{e}_n(t) \\ e_n(t)\end{matrix}] \quad (21)$$

where $r(n) \in [0,1)$ describes the forgetting factor function, n is the amount of iterations and $e_n(t)$ is the deviation of the controlled variable.

Based on the forgetting factor function, Eq. (12), which describes the slave leg impedance control, can be written as:

$$\mathrm{M_S}\ddot{\tilde{e}}_s + \mathrm{B_S}\dot{\tilde{e}}_s + \mathrm{K_S}\tilde{e}_s = \mathrm{Q}_{des} - \mathrm{Q}_{ext} \quad (22)$$

Here, $\tilde{e}_s(n)$ describes the difference between the desired posture and the actual posture in Eq. (14), which can be written as:

$$\tilde{e}_s(n) = \tilde{\chi}_{des}(n) - \chi_p(n) \quad (23)$$

$\tilde{\chi}_{des}(n)$ defines the expected posture using the forgetting factor function:

$$\tilde{\chi}_{des}(n+1) = (1 - r(n))\chi_a^{\#}(n) + r(n)\chi_{des}^{\#}(n) \quad (24)$$

Where $\chi_{des}^{\#}(n)$ describes the expected posture of the slave leg, while $\chi_a^{\#}(n)$ describes the posture of the slave leg calculated from the master leg. Here, $\chi_a^{\#}(n)$ can be calculated by:

$$\chi_a^{\#}(n) = T_a^{\#}(n) \cdot \chi_{des}(n) \quad (25)$$

Where $T_a^{\#}(n)$ is the coordinate transformation from the master leg to the slave leg. The following equation is obtained

$$\tilde{e}_s(n) = (1 - r(n))T_a^{\#}(n) \cdot \chi_a(n) + r(n)\chi_{des}^{\#}(n) - \chi_p(n) \quad (26)$$

The forgetting factor function $r(n) \in [0,1)$ is a monotonically decreasing function during movement. First, $r(n)$ is closed to 1, so the equilibrium position of the virtual spring is closer to the expected position of the follower. Then, $r(n)$ gradually approaches 0, and the equilibrium position is close to the position corresponding to the main leg. When

motion stops, the main leg determines the equilibrium position.

The forgetting factor function is written as:

$$r(n) = \frac{1}{\ln(\frac{n}{10} + 2.9)} \tag{27}$$

IV. Dynamics Co-simulation Experiment

A. Experimental System Description

The experiments in this chapter are all based on the ADAMS and MATLAB co-simulation platform[16][17]. The following figure shows the simulation experiment system build in MATLAB.

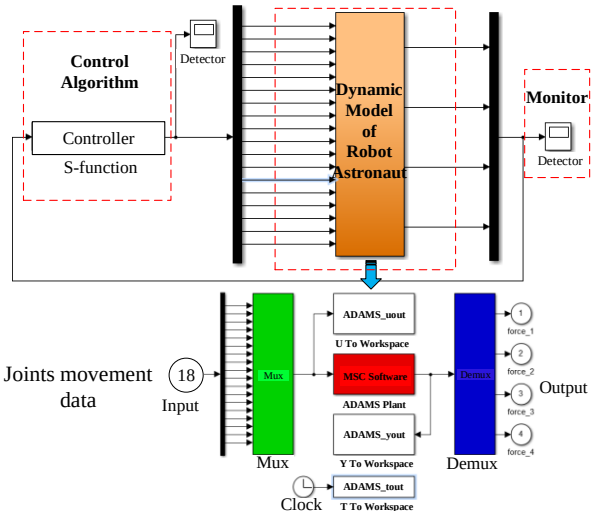

Figure 8. Simulation experiment system in MATLAB

It can be seen that the simulation system receives 18 joints motion data and outputs robot motion and joint dynamics information.

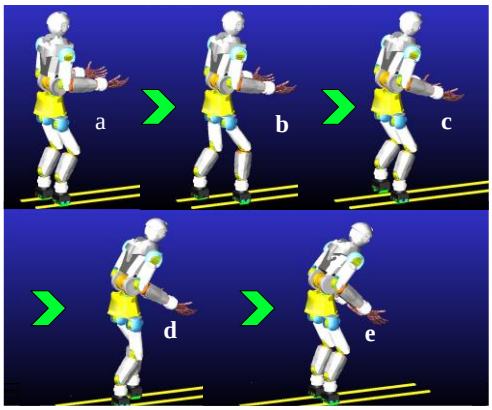

Figure 9. A complete experiment process from a to e in ADAMS. a: the robot naturally bents two legs; b: the right leg steps forward; c-d: the left leg steps forward; e: the right leg goes half step and align with the left leg.

The above figure is a process diagram of the simulation experiment in ADAMS. It can be seen from the figure that the robot walks a complete course of a gait.

B. Contact forces between end effector and handrails

Due to mechanical deformation, accidental multi-point collisions, joint positioning errors and visual inaccuracies, the robot's actual posture and position deviate from the ideal state (Figure 6). These deviations create conflicting forces in the rigid closed chains formed by the legs, which can result in greater force/torque between the end effector and the handrail. As can be seen from Figure 10.(a) and 10.(c), the force and torque between the slave leg and the handrail controlled by the conventional position-based control method start to appear at the stage of grasping the handrail, reaches the maximum at the moment of fully grasp, and then the other leg starts relax the end effector, and the contact force and moment are then reduced to the normal value. The peak force/torque is very large, which causing great damage to the robot.

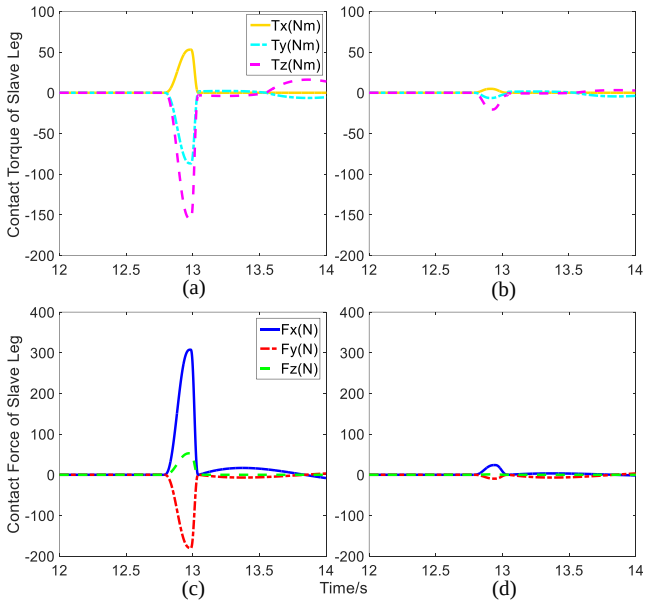

Figure 10. Contact forces of dual legs by using different control method

In order to overcome this problem, an adaptive algorithm is proposed to eliminate the conflicting force of the slave leg during the movement. It is based on the Impedance Control and Forgetting Factor functions. Figures 10.(b) and 10.(d) show the contact force and the torque of slave leg controlled by the adaptive impedance controller.

TABLE I. Contact Force/Torque

Controller	Force(N)			Torque(Nm)		
	F_x	F_y	F_z	τ_x	τ_y	τ_z
Classical	307.56	180.69	53.01	53.00	86.82	154.97
Adaptive	23.87	84.98	10.87	4.82	6.42	20.62

It can be seen from the figure 10.(b) and 10.(d) that the contact force and torque are always controlled within a suitable size range. TABLE I. lists the maximum contact forces under the control of the two different methods. Obviously, the contact force and torque of the leg controlled using the classical method is much greater than the contact

force and torque of the leg when using the adaptive method. These results show that our proposed adaptive impedance controller can reduce the connection force/torque between the end effector and the handrails and eliminate the collision force in the rigid closed chain.

C. Torques of joints

In this experiment, joints 1, 3, 5, and 6 have the highest output torque (excluding initial stage torque), and the rest of the joints have smaller output torques or no significant changes in output torques. Figures 11.(a) and 11.(c) show the joint output torques of dual legs by using the conventional position-based control method. Figures 11.(b) and 11.(d) show joint output torques of dual legs by using the adaptive impedance controller.

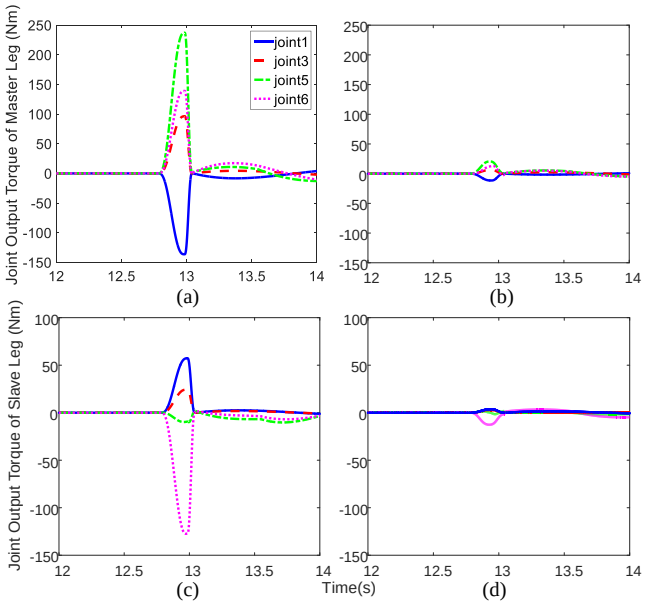

Figure 11. Joint torques of dual legs by using different control method

It can be seen from the figure that when the robot legs do not grasp the handrails, the output torques under different control methods are similar. During the process of the robot grabbing the handrail, the output torques under classical method increases sharply until it reaches the maximum (completely grasped). While the joint torques under adaptive impedance control are always controlled within a reasonable range in the process of grabbing the handrail. This shows that the adaptive impedance control method can effectively eliminate the impact moment during robot movement and improve the robot's movement stability.

V. CONCLUSION

Due to mechanical deformation, accidental multi-point collisions, joint positioning errors, the robot astronaut always generates internal stress during movement, which causes great damage to robot astronaut and space station. Through analyzing this problem, we carried out the dynamic modeling and collision mechanics modeling of the robot astronaut. On this basis, we proposed a dual-feedforward torque compensation control strategy based on dynamics and force impendence. In the simulation experiments, this control strategy greatly reduces (or even eliminates) the conflicting force during robot movement. It is verified that the control strategy has good robustness and control accuracy. It has certain application value for the stability control of robot astronaut.

ACKNOWLEDGMENT

The authors would like to acknowledge the National Natural Science Foundation of China (61733001, 61573063, 61503029, U1713215) for its support and funding of this paper.

REFERENCES

[1] JohnMcPhee, ChadSchmitke, and ScottRedmond. "Dynamic Modelling of Mechatronic Multibody Systems With Symbolic Computing and Linear Graph Theory." *Mathematical Modelling of Systems* 10.1(2004):1-23.

[2] Huang, Q., Yokoi, K., Kajita, S., & Kaneko, K. (2001). Planning walking patterns for a biped robot. *Robotics & Automation IEEE Transactions on,17*(3), 280-289.

[3] Craig, J. J. (1986). "Introduction to Robotics: Mechanics and Control". Third Edition. Beijing: Machinery Industry Press, 2006:48-59.

[4] Siciliano, B., & Sciavicco, L. (2000). Modelling and control of robot manipulators. *Industrial Robot*, 11(1), 1828.

[5] Craig J J . Introduction to Robotics: Mechanics and Control[M]. Upper Saddle River: Pearson Prentice Hall, 2005.

[6] Vukobratovic, M., & Kircanski, M. (1984). "A dynamic approach to nominal trajectory synthesis for redundant manipulators." Systems Man & Cybernetics *IEEE Transactions on*, SMC-14(4), 580-586.

[7] Harada, K., Kajita, S., Kaneko, K., & Hirukawa, H. (2004). "An analytical method on real-time gait planning for a humanoid robot." *IEEE/RAS International Conference on Humanoid Robots* (Vol.2, pp.640-655 Vol. 2).

[8] Zhu Haiyan. Study on trajectory planning and control method of humanoid robot arm with seven degrees of freedom. Zhejiang University of Technology, 2012.

[9] Rehnmark F, Spain I, Bluethmann W, et al. An experimental investigation of robotic spacewalking. 4th *IEEE/RAS International Conference on Humanoid Robots. IEEE*, 2004, 366-384.

[10] Wei, Bo, et al. "Adaptive Impedance Controller for a Robot Astronaut to Climb Stably in a Space Station." *International Journal of Advanced Robotic Systems* 13.3(2016):1.

[11] HOGAN,N. "Impedance control, An approach to manipulation : Part I, II."*Asme Journal of Dynamic Systems Measurement & Control*107.4(1985):481-9.

[12] Hashtrudi-Zaad, K., & Salcudean, S. E. (2001). Analysis of control architectures for teleoperation systems with impedance/admittance master and slave manipulators. int j robot res. *International Journal of Robotics Research, 20*(6), 419-445.

[13] Latawiec, Krzysztof J., et al. "Adaptive finite fractional difference with a time-varying forgetting factor." *International Conference on Methods and MODELS in Automation and Robotics* IEEE, 2012:64-69.

[14] Albu, F. "Improved variable forgetting factor recursive least square algorithm." *International Conference on Control Automation Robotics & Vision* IEEE, 2012:1789-1793.

[15] Bouakrif, Farah. "Iterative Learning Control with Forgetting Factor for Robot Manipulators with Strictly Unknown Model." 26.3(2011):264-271.

[16] Wang Guoqiang, Zhang Jinping. "Virtual Prototyping Technology and Its Practice on Adams". First Edition. Xi'an: Northwestern Polytechnical University Press, 2002:8-90.

[17] Liu Jun. "Dynamic System Modeling and Identification". First Edition. Changsha: National University of Defense Technology Press, 2007:1-13.