Contact Force Rendering Method Based on Robot Dynamics

Qing Wei[1,2], Nailong Liu[1,2], Zhaoming Liu[1,2], Long Cui[1]

1. State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
E-mail: weiqing@sia.cn, liunailong@sia.cn, liuzhaoming@sia.cn, cuilong@sia.cn

2. University of Chinese Academy of Sciences, Beijing 100049, China

Abstract: Force perception is the primary means for operators to perceive the environment accurately, and a contact force rendering method based on the robot dynamics is proposed for rigid contact tasks. Firstly, the normal contact force is calculated by the rigid contact model and the robot dynamics, and then the tangential force is calculated according to the friction model. In order to improve the computational stability, the flexible joint model is designed, and the additional force and torque are used to compensate the pose error introduced by the sampling calculation. Finally, the proposed algorithm is verified in our self-developed robot simulation system, experiments show that the algorithm has high real-time performance, and the stable contact between the robot with the environment can be achieved through the master hand motion filtering and feedback force variable stiffness rendering in human-computer interaction.

Key Words: Contact Force Rendering, Virtual Robot, Rigid Contact

1 INTRODUCTION

As the most promising method to solve the delay problem of teleoperation, virtual reality has already played an important role in the field of telerobot pre-simulation and auxiliary operation[1-3]. In all kinds of teleoperation tasks, vision is responsible for the overall perception of the environment, and force is an effective supplement to vision, which completes the precise perception of the environment in contact tasks. In contact tasks, more than 70% of the environmental perception information is derived from force[4]. Especially in rigid contact tasks, in which tiny position errors may cause very large contact force. So, it is difficult to complete the remote contact task safely and effectively without force guided [5-7].

In order to solve the problem of contact force rendering in virtual environment, Jiancheng and Faulring put forward a contact force rendering method based on admittance control and geometric constraint[8, 9]. The algorithm has high computational stability, but its algorithm and structure are complex, and its implementation cost is high. And in the contact stage, the operation target will have a significant equivalent quality[10], so the operator's physical burden will also be larger. Huijun and Juan proposed a contact force calculation method based on the environmental dynamics model[11,12]. The algorithm has high realtime performance, but the dynamic parameters of the environment need to be identified on-line. So a vibration suppression algorithm should be need when the environmental stiffness is large, and the process is difficult and with high risk. And in the calculation process, the contact force between two sampling points is calculated according to the current state. Therefore, the sum of the impulse calculated by this method is greater than the penetration impulse, which destroys the passivity of the physical collision, and will introduce excess jitter, and even leads to system instability. Yim proposed a virtual contact force calculation method based on measured data modeling[13]. The contact force calculated by this method has high fidelity, but it is only suitable for accurate environment and requires additional equipment, which is difficult to achieve in many cases. Hongyi proposed a contact force calculation method based on collision detection[14], the method can be applied to complex geometry scene, and the real-time performance of the algorithm will not decrease with the complexity of the model, but it requires a lot of memory to achieve accurate collision detection. Park proposed a force calculating method based on the object dynamics model [15]. This method can simulate physical contact, but it is only suitable for free moving objects. In addition, the artificial potential field method[16,17] and virtual fixture [18,19] can also be used to calculate the virtual guiding force according to the distance between the tool and the obstacle or operation target or the predetermined trajectory, but it is generally not used for contact force calculation.

In order to render the contact force of the virtual robot steadily in rigid contact tasks, a rendering method based on the dynamic model of the robot is proposed in this paper. In this method, the dynamic parameters of the environment are replaced by the robot dynamics, which can be calculated beforehand, so the delay and security problems introduced by online parameter identification are eliminated. In the process of physical simulation, in order to improve the stability of computation, a flexible joint model is designed, and the additional force and torque are used to compensate the pose error introduced by the sampling data calculation. Finally, the proposed algorithm is verified in our self developed robot simulation system, experiments show that the algorithm has high real-time performance, and the stable contact between the robot with the environment can be achieved through the master hand motion filtering and feedback force variable stiffness rendering in human computer collaboration.

This work is supported by International Thermonuclear Experimental Reactor(ITER) Project under Grant 2012GB102005.

2 ALGORITHM

The robot is driven by joint torques, let τ are joint torques of the robot, and F is the contact force of the end tool. Without considering the mass of the robot, the formula(1) can be obtained according to the principle of conservation of energy.

$$F\delta x = \tau\delta\theta \tag{1}$$

In which, δx is the tiny increment of the robot's end in cartesian space, and $\delta\theta$ is the tiny increment in the joint space.

According to the definition of the robot Jacobian matrix, the formula(2) can be obtained.

$$\delta x = J(\theta)\delta\theta \tag{2}$$

In which, $J(\theta)$ is the Jacobian matrix of the robot.

Therefore, without considering the mass of the robot, the contact force at the end of the robot can be expressed as a function of the joint torque, as formula(3).

$$F = J^{-\mathrm{T}}(\theta)\tau \tag{3}$$

But the actual robot always has mass, so the robot joint torques not only to overcome the end contact force, but also to compensate the gravity and change the motion state of robot. The dynamic model of the robot is as follow:

$$\tau = M(\theta)\ddot{\theta} + V(\theta,\dot{\theta}) + G(\theta) \tag{4}$$

In which, $M(\theta)$ is the mass matrix, $V(\theta,\dot{\theta})$ is the vector of centrifugal force and coriolis force, and $G(\theta)$ is the vector of gravity.

In the rigid contact operation task, for safety, we need to make the robot flexible. So the stiffness of the end tool and environment is much larger than the robot's closed-loop stiffness. Therefore, it can be considered that there is no penetrating motion between the robot and the environment, and the joint torque of the robot can be expressed as follow:

$$\tau_d' = M(\theta)\ddot{\theta}_d + V(\theta,\dot{\theta}_d) + G(\theta) \tag{5}$$

In which, θ is the current joint angle vector， $\dot{\theta}_d$ and $\ddot{\theta}_d$ are the expected joint velocity and acceleration, which depends on the position increment of the main hand after the collision and the path planning algorithm of the robot.

Therefore, the traditional mass damping spring model can be improved, and the static contact force can be directly calculated according to the dynamic model of the robot as follow.

$$F_s = -J^{-\mathrm{T}}(\theta)\cdot(M(\theta)\ddot{\theta}_d + V(\theta,\dot{\theta}_d)) \tag{6}$$

But in the actual process of contact, a lthough the robot will not penetrate the contact surface, bur the tangential motion still exists. So, the contact force directly calculated in formula (6) is effective only in the normal, and the tangential contact force should be calculated according to the friction model. The contact force can be decomposed as formula(7).

$$f = F_n + F_t \tag{7}$$

In which, F_n and F_t are the normal contact force and the tangential friction, respectively.

In addition, the normal direction of the contact surface will change with the contact position, so the change of the normal velocity also needs to include in the contact force calculation. According to the momentum theorem, formula(8) can be acquired.

$$\int_t^{t+T} F_{sn}dt = \int_t^{t+T} F_n dt + M_x(\theta)\int_t^{t+T} \dot{v}_n dt \tag{8}$$

In which, $M_x(\theta)$ is the mass matrix in Cartesian space, F_{sn} is the normal component of F_s , $\dot{v}_{\mathrm{n}}$ is the normal component of the acceleration of the robot end. And T is the sampling period.

According to the definition of the Jacobian matrix, the normal acceleration can be acquired.

$$\dot{v}_{\mathrm{n}} = ((\dot{J}(\theta)\dot{\theta} + J(\theta)\ddot{\theta})\cdot n)n \tag{9}$$

From formula(3) and (9), we can obtain formula(10).

$$M_x(\theta) = J^{-\mathrm{T}}(\theta)M(\theta)J^{-1}(\theta) \tag{10}$$

And formula(11) is used in the actual calculation.

$$\int_t^{t+T} \dot{v}_n dt = -(1+e)v_n \tag{11}$$

In which, e is the equivalent rebound coefficient of the collision process, which depending on the elastic coefficient of the robot and the environment.

The tangential friction can be expressed as:

$$F_t = \begin{cases} \mu\,\mathrm{sgn}(v_t)\|F_n\|, & \|v_t\| \geq v_\sigma \\ \dfrac{v_t}{v_\sigma}\mu\,\mathrm{sgn}(v_t)\|F_n\|, & \text{else} \end{cases} \tag{12}$$

In which, n is the unit normal vector for the contact surface, μ is the coefficient of friction, v_t is the speed on the tangent and the threshold v_σ is used to weaken the friction jitter caused by the calculation error.

When the object is a flexible object, we can use the proton spring model to describe the object[12], and its dynamic model can identify online easily[11]. The contact force model can be expressed as:

$$f = m\ddot{x} + b\dot{x} + kx \tag{13}$$

In which, $\boldsymbol{f}$ is the contact force, x is the deformation of the environment, m 、 b 、 k are dynamic parameters for the object(equivalent mass, damping coefficient and elastic coefficient).

So, the dynamic equations of the contact between the robot and the environment can be obtained.

$$\tau_d = M(\theta)\ddot{\theta} + V(\theta,\dot{\theta}) + G(\theta) + J^{\mathrm{T}}(\theta)f \tag{14}$$

And, the joint acceleration vector of the robot is as follows:

$$\ddot{\theta} = M^{-1}(\theta)(\tau_d - V(\theta,\dot{\theta}) - G(\theta) - J^{\mathrm{T}}(\theta)f) \tag{15}$$

In the ideal case, the simulation of the robot can be realized by using the numerical integration according to the acceleration of the robot. However, in the physical based robot simulation method, all the links of the robot are driven by the forces and torques of the joints. Due to the limitation of the computation time, the discrete integral cannot simulate the real motion process of each link, and calculation error would appears. In order to eliminate the

pose error of links, we need to exert force and torque on every link. In order to improve the solution stability and reduce the penetration phenomenon, flexibility joints model should be used[20]. According to the designed joint damping k_v and stiffness k_p the expected joint torque τ_d can be calculated, and the additional restoring force and torque can be calculated according to the following method.

If the theoretical position and centroid speed of link i calculated from the joint state are ${}^{i}x_i$ and $[{}^{i}v_i \quad {}^{i}\omega_i]^{\mathrm{T}}$. And the actual position and velocity of the link are ${}^{i}x_i'$ and $[{}^{i}v_i' \quad {}^{i}\omega_i']^{\mathrm{T}}$, respectively, so the external force and torque added to the links are as follows:

$$\begin{bmatrix} {}^{i}F_i \\ {}^{i}N_i \end{bmatrix} = k_{v_i} \begin{bmatrix} m_i({}^{i}v_i - {}^{i}v_i')/\mathrm{T} \\ {}^{C_i}I_i({}^{i}\omega_i - {}^{i}\omega_i')/\mathrm{T} \end{bmatrix} + k_{p_i}({}^{i}x_i - {}^{i}x_i') \quad (16)$$

In the operation task, in order to reduce the burden of the operator and let the operator to accurately perceive physical contact force, dynamic torque compensation is used to the master manipulator. So the equivalent quality of the robot is much greater than that of the master manipulator. When a collision occurs, the normal velocity's change will cause large contact force, but restricted by the reaction time, the operator's arm stiffness and damping will not immediately increase. In this case, the master hand will pop up, and may cause an accident when the teleoperation system containing a real robot. In order to achieve smooth contact with the environment, it is necessary to render the virtual feedback force with variable stiffness [21]. Let f_b is the contact force feedback to the master manipulator.

$$f_b = S(t)f \quad (17)$$

In which, $S(t)$ is the variable stiffness function, which is an increasing function with time. Let the collision occurred at $t=0$, and the $S(t)$ meets: $S(t)=0$ when $t \le 0$, $S(t)=1$ when $t \ge t_0$. As humans are more sensitive to the force reaction, so t_0 is generally below 0.1s.

3 EXPERIMENTS AND ANALYSIS

In order to adapt to the unstructured environment teleoperation tasks, the structure light is used to model the slave environment on-line in our system. The experiment platform is shown in figure 1, contains a black and white multi line structured light scanner (CP-300 3D CaMega), a 3dof master manipulator Omega.3 (Force Dimension), a 6dof robot (Stäubli TX60L), a 3dof holder and a set of 3D projection system. In addition, there is a graphical workstations running virtual reality system and a server running master program. All devices communication through network, besides the master manipulator to the server through USB2.0. The master control software and virtual reality software are developed by C++ language, and the development environment is VS2010.

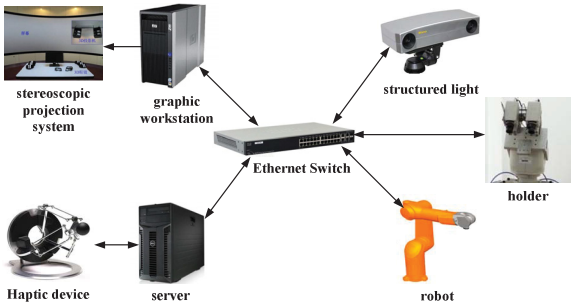
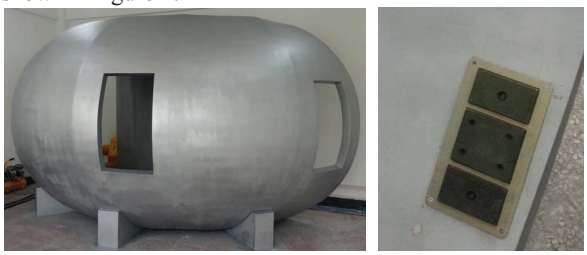


Fig.1 Experiment platform

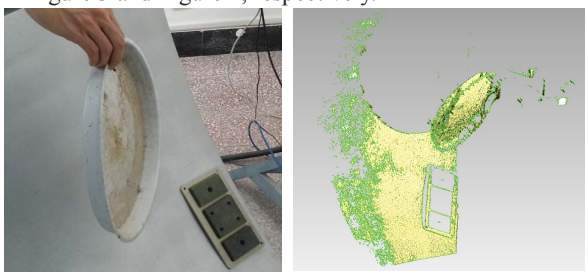
In the experiment, the stiffness of the robot operating environment (a stainless steel fusion reactor) and the operating target (three insulating beryllium tiles) is larger, as shown in figure 2.



a. nuclear fusion reactor b. beryllium tile

Fig.2 Work environment of the robot

When an obstacle occurs in the environment, the scanned point cloud data and the reconstructed 3D scene are shown in Figure 3 and Figure 4, respectively.



a.the environment with an obstacle B.the scanned point cloud

Fig.3 Point cloud when an obstacle appeared

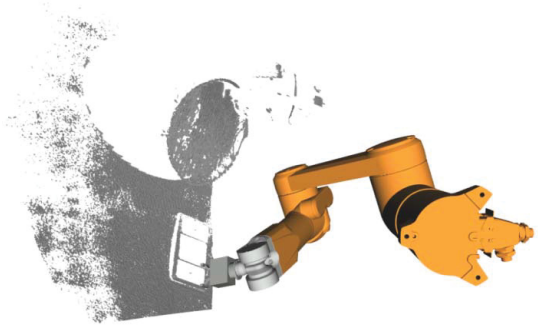


Fig.4 Virtual reality scene with an obstacle

Since the human beings have the persistence of vision phenomenon, so the operator will feel the video is continuous when the video refresh frequency is more than 24Hz. However, if we want the operator feel the force continuous, the refresh rate of the feedback force shuold be more than 200hz. And in order to ensure the operator feel the force smoothly, the refresh rate should be more than 1000Hz[22]. Therefore, the graphics processing and the physical simulation using two separate threads in the program, physical simulation and graphics refresh rate are shown in Figure 5 and Figure 6 (in order to ensure the stability of the image display,the vertical sync is enabled in the graphics rendering). From Figure 5 and Figure 6 we can see that the graphic refresh rate is more than 57Hz, and the force feedback rate is more than 1100Hz. Therefore, both the graphic rendering rate and force feedback frequency are meet the continuous perception demand of the operator.

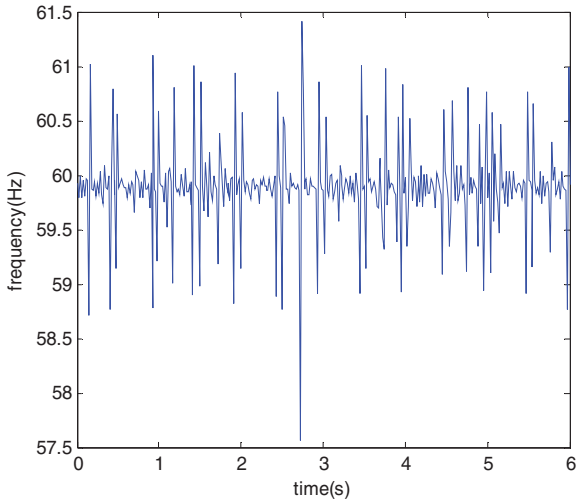


Fig.5 GUI refresh rate

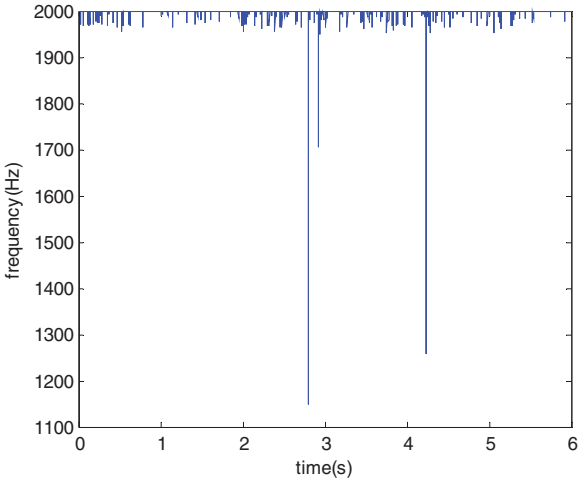


Fig.6 Physical simulation frequency

The inertia and centroid of each link can be calculated by numerical integration method before the task, according to the distribution of vertices and triangles in its 3D model. The virtual robot is controlled by the master manipulator to perform the tile replacement task, and the computed normal contact force between the robot and the environment is shown in figure 7.

From the figure we can see that the calculated contact force contains significant jitter. This is due to the first and second order differential of the joints is used when calculating the virtual contact force, so it's very sensitive to joint jitter. However, the jitter of the operator is inevitable when operating the master manipulator, so the contact force calculated from the dynamical model jittering heavily. Therefore, in order to complete the contact task smoothly, it is necessary to smooth the feedback force.

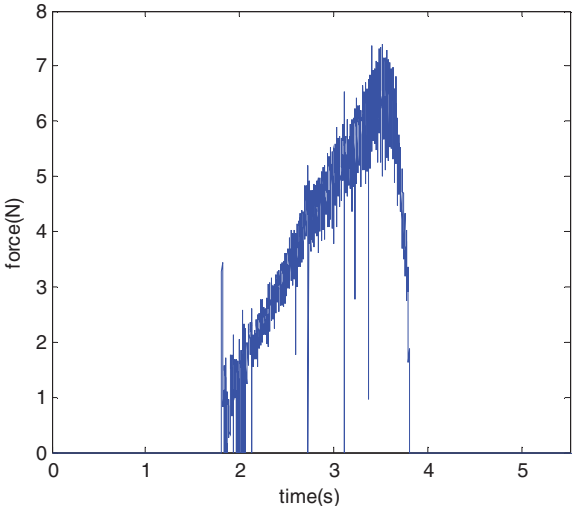


Fig.7 The calculated normal contact force

Previous studies have shown that the contact force is a low frequency signal with high energy, and the high frequency dynamics part of force feedback may cause the unstable of the contact force rendering and perception[23], so a low pass filtering is needed for the calculated contact force. In order to determine the cut-off frequency of the filter, we first sampled the calculated contact force at the speed of 2000Hz, and then the fast fourier transform(FFT) is carried out, figure 8 showing the 0-50Hz band of the transformation result.

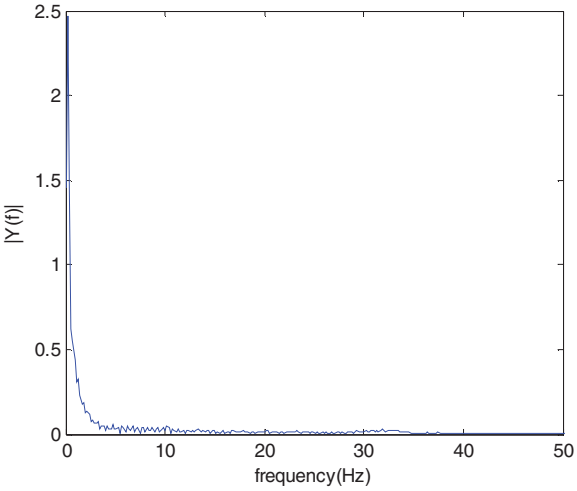


Fig.8 FFT result of the contact force

According to the results of fast Fourier transform, a low pass

filter with cut-off frequency of 35Hz is selected to filter the calculated contact force, and the filtering result is shown in figure 9.

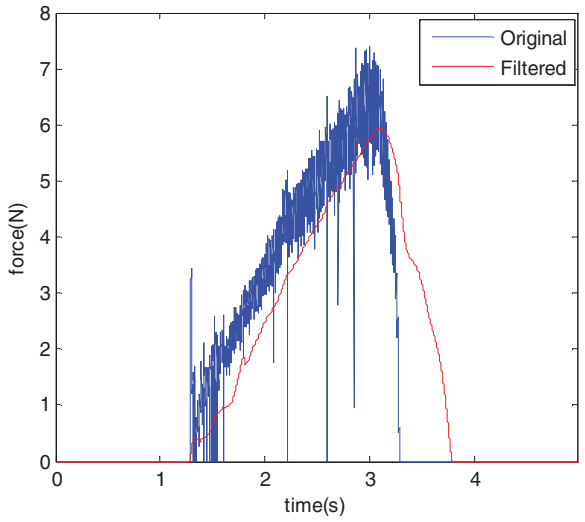


Fig.9 Contact force after low pass filtered

From figure 9, we can see that although low-pass filtering can smooth the contact force directly calculated, but the delay introduced by it will cause visual, motion and force feedback do not correspond in time dimension. Therefore, the operator can't adjust their arm in real-time to control the change of contact force, which will cause high energy surge and position jitter in contact process, and can not provide operators with good operation experience.

The jitter data in the calculated force can be obtained by the result of the filtering(shown in Figure 10).

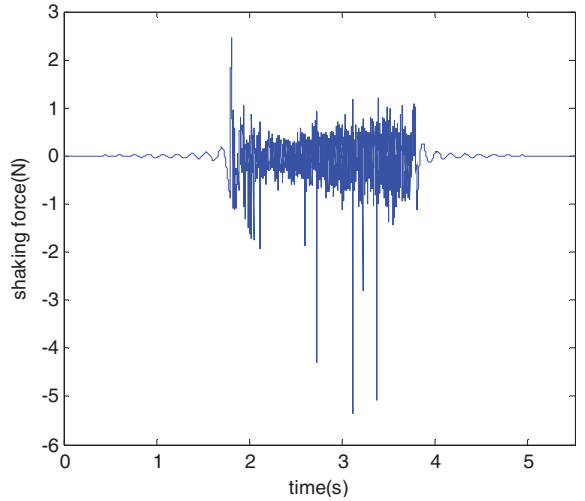


Fig.10 Jitter data of the contact force

From the contact force jitter data shown in figure 10, we can see that: 1) the greater the contact force, the more serious the contact force jitter; 2) the contact force jitter at the start of the collision is greater than that of the collision separation. I think there are two reasons for this phenomenon: 1) the operator is a control system with low frequency and high flexibility, the damping and stiffness of the arm is very small in free movement; 2) affected by the influence of the reaction delay, the operator can not immediately adjust the damping and stiffness of his arm to produce a reasonable force to counteract the feedback force at the moment of collision occurrence.

As the main reason of the contact force jitter is the jitter of the master manipulator, so the other way to smooth the contact force is the master hand motion filtering. The master hand motion jitter mainly comes from two aspects, one is the operator factors (such as motion error adjustment and arm muscle fatigue, etc.), the other is the device factors (such as electromagnetic interference and non rigid characteristics, etc.). The expected motion of the operator is generally low frequency, and the modal difference is relatively large to the noise. Therefore, it is possible to use a simpler filtering algorithm to separate them. Eliminate rough values and filter the motion of the master hand using a sliding average algorithm. The result is as figure 11.

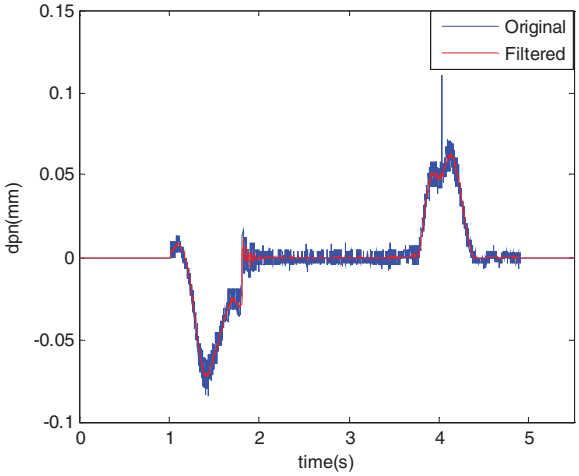


Fig.11 Position increment of the master hand

From figure 11, we can see that the jitter data in the position increment of the master hand can be eliminated using simple filtering algorithm, and do not introduce obvious delay. The virtual contact force calculated according to the position after filtering is as follows:

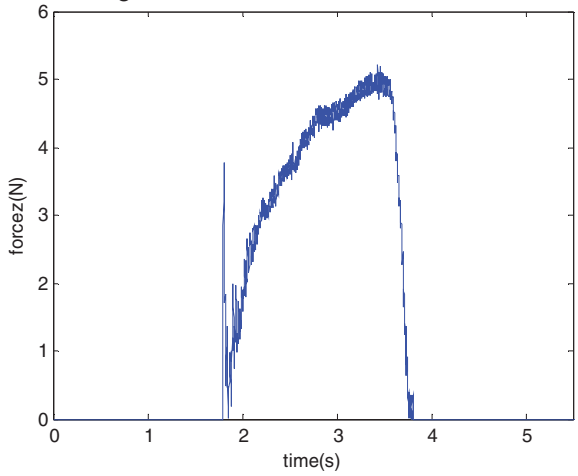


Fig.12 The calculated normal contact force

From figure 12, we can see that the amplitude of contact force jitter has been significantly reduced after master hand motion filtering, and the energy is more dispersed, so there is only very small jitter of force can be sensed through the master hand. From figure 11, we can see that there is no obvious beating of the main hand position even at the moment of collision occurrence. So the contact task can carry out smoothly through this method.

4 CONCLUSION

Force perception is the primary means for operators to perceive the environment accurately in contact tasks, in order to render the contact force of the virtual robot steadily in rigid contact tasks, a virtual force rendering method based on the dynamic model of the robot is proposed in this paper. In the proposed algorithm, the rigid contact model is used, and considers that there is no penetrating motion between the robot and the environment. So the normal contact force can be calculated directly according to the dynamic parameters and velocity planning algorithm of the robot, and then the tangential force is calculated according to the friction model. Since the whole process of computation only requires the robot dynamics, and does not need to identify the parameters online, so the algorithm has high real-time performance and security. In order to improve the stability of the algorithm, the algorithm reduces the closed loop stiffness of the virtual robot, and compensates the pose error introduced by the sampling calculation through additional force and torque. Finally, the proposed algorithm is verified in our self developed robot simulation system, in order to meet the continuous perception demand of the operator, we use two different threads to handle physical simulation and graphics refresh, respectively. Experiments show that the algorithm has high real-time performance, and the stable contact between the robot with the environment in human computer collaboration can be achieved through master hand motion filtering and feedback force variable stiffness rendering.

REFERENCES

[1] Bo L, Fuyi C, Jinfeng L. The Teleoperation Simulation System Based on VR .Proceedings of the 2014 International Conference on Virtual Reality and Visualization, Shenyang, China, IEEE, 7-11, 2007.

[2] Cichon T, Roßmann J. Robotic teleoperation: Mediated and supported by virtual testbeds. Proceedings of the 2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR), Shanghai , Peoples R China, IEEE, 55-60, 2017.

[3] Rakshit S M, Banerjee S, Hempel M, et al. Fusion of VR and teleoperation for innovative near-presence laboratory experience in engineering education. Proceedings of the 2017 IEEE International Conference on Electro Information Technology (EIT), Lincoln, NE, USA, IEEE, 376-381, 2017.

[4] Ma Yunfei, Wang Aimin, Song Aiguo, et al. Prospects of Internet—Based Force Telepresence Teleoperation Control System. MEASUREMENT & CONTROL TECHNOLOGY, Vol.21, No.3, 1-3, 2002.

[5] Smisek J, Mugge W, Smeets J B J, et al. Haptic guidance on demand: A grip-force based scheduling of guidance forces. IEEE Transactions on Haptics, Vol.PP, No.99, 1-1, 2017.

[6] Coles T R, Meglan D, John N W. The Role of Haptics in Medical Training Simulators: A Survey of the State of the Art [J]. IEEE Transactions on Haptics,Vol 4, No.1,51-66, 2011.

[7] Trejo F, Hu Y. Towards an analytic haptic model for force rendering of soft-tissue dissection. Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 001098-001103, 2016.

[8] Shi Jiancheng, Liu Jianhua, NingRuxin, Hou Weiwei. Haptic Rendering Technology Based on Admittance Control in Virtual Assembly. Journal of Computer-Aided Design & Computer Graphics, Vol 24, No.02, 227-235, 2012.

[9] Faulring E L, Lynch K M, Colgate J E, et al. Haptic Display of Constrained Dynamic Systems via Admittance Displays. IEEE Transactions on Robotics, Vol23, No.1, 101-111, 2007.

[10] Treadway E, Yang Y, Gillespie R B. Decomposing the performance of admittance and series elastic haptic rendering architectures .Proceedings of the 2017 IEEE World Haptics Conference (WHC), 346-351, 2017.

[11] Li H, Song A. Virtual-Environment Modeling and Correction for Force-Reflecting Teleoperation With Time Delay. IEEE Transactions on Industrial Electronics, Vol.54, No.2, 1227-1233, 2007.

[12] WU Juan , SONG Aiguo , LI Jianqing. A Quick Physics-Based Deformation Model and Real-Time Force Reflection Algorithm . CHINESE JOURNAL OF SENSORS AND ACTUATORS, Vol.18, No.01, 90-94+111, 2005.

[13] Yim S, Jeon S, Choi S. Data-Driven Haptic Modeling and Rendering of Viscoelastic and Frictional Responses of Deformable Objects. IEEE Transactions on Haptics, Vol.9, No.4, 548-559, 2016.

[14] Xu H, Barbič J. 6-DoF Haptic Rendering Using Continuous Collision Detection between Points and Signed Distance Fields. IEEE Transactions on Haptics, Vol.10, N0.2, 151-161, 2017.

[15] Park G, Choi S. A Physics-Based Vibrotactile Feedback Library for Collision Events. IEEE Transactions on Haptics, Vol.10, No.3, 325-337, 2017.

[16] [16] Penin L F, Matsumoto K, Wakabayashi S. Force reflection for ground control of Space robots. IEEE Robotics & Automation Magazine, Vol,7, No.4, 50-63, 2000.

[17] Li Xiaopeng. Research on Virtual Reality of Virtual Force Guidance for Telerobotic System [D]; changchun: Jilin University , 2014.

[18] Rosenberg L B. Virtual fixtures: Perceptual tools for telerobotic manipulation. Proceedings of the 1993 IEEE Virtual Reality Annual International Symposium, Seattle, WA, IEEE, 76-82, 1993.

[19] JIANG Zainan, ZHAO Jingdong, LIU Hong. Haptic Flexible Virtual Fixture for Teleoperation. ROBOT, Vol.33, No.6, 685-690, 2011.

[20] Todorov E. Convex and analytically-invertible dynamics with contacts and constraints: Theory and implementation in MuJoCo. Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), 6054-6061, 2014.

[21] Singh H, Ryu J H. Circumventing the fundamental tradeoff between stability and performance in haptic rendering - successive force augment approach. Proceedings of the 2016 16th International Conference on Control, Automation and Systems (ICCAS), 1292-1297, 2016.

[22] McNeely W A, Puterbaugh K D, Troy J J. Six degree-of-freedom haptic rendering using voxel sampling [C].Proceedings of the 26th annual conference on Computer graphics and interactive techniques, New York, NY, USA, ACM Press, 401-408, 1999.

[23] Choi S, Tan H. Perceived Instability of Virtual Haptic Texture. I. Experimental Studies. Presence, Vol.13, No.4, 395-415, 2004.