Hindawi
Journal of Sensors
Volume 2020, Article ID 8881263, 13 pages
https://doi.org/10.1155/2020/8881263

Research Article

Research on Collision Point Identification Based on Six-Axis Force/Torque Sensor

Zhijun Wang,[1,2] **Lu Liu**,[1,2] **Wenkai Yan**,[1,2] **Jing He**,[3] **Bingyan Cui**,[1,2] **and Zhanxian Li**[1,2]

[1]*College of Mechanical Engineering, North China University of Science and Technology, Tangshan 063210, China*
[2]*Hebei Province Research Institute of Industrial Robot Industry Technology, Tangshan 063210, China*
[3]*Tangshan Industrial Vocational Technical College, Tangshan 063020, China*

Correspondence should be addressed to Zhijun Wang; zjwang@ncst.edu.cn

Received 18 August 2020; Revised 15 November 2020; Accepted 17 November 2020; Published 12 December 2020

Academic Editor: Giovanni Diraco

The collision detection algorithm of the robot body previously needed to rely on the surface geometry information of the colliding object and no deformation was allowed during the collision process. To solve this problem, a new robot body collision detection algorithm that uses the force information of the six-axis force/torque sensor at the base to self-constrain is proposed which does not rely on the geometric information of the colliding object surface, and the deformation also allows deformation during the collision. In terms of sensor data preprocessing, a gravity and dynamic force compensation algorithm for the six-axis force/torque sensor at the base is proposed to ensure that the reading of the six-axis force/torque sensor at the base always maintains the value of 0 when the robot is working. Then, the robot is considered to have collided with the outside world when the sensor reading exceeds the set threshold. And a precision factor is proposed to analyze the influence of force and collision distance on the accuracy of the algorithm. Finally, the new algorithm proposed in this paper is compared with the traditional algorithm that relies on the geometric information of the colliding body surface. The experimental results indicate that the accuracy of the collision point detection algorithm proposed in this paper is close to that of the traditional method, but it does not need to rely on the geometric information of the collision body surface, and there is no requirement for whether there is deformation during the contact process. It can be concluded that the collision distance is the most important factor affecting the accuracy of the algorithm, followed by the conclusion of the magnitude of the collision force through the calculation of the precision factor. The results show that this method can effectively detect the collision point of the machine body, and the maximum error at the farthest point of the robot is 8.712%, which lays a certain foundation for the subsequent research on human-machine collaboration in small collaborative robots.

1. Introduction

With the development of robot technology, a variety of sensors such as vision, touch, and force were applied to robot systems. Force perception is an important function for intelligent robots to interact with the external environment, especially for operations such as grasping, contour detection, obstacle avoidance, human-computer interaction, and force feedback control [1–3]. The perception of force information also plays a vital role in the hexacopter equipped researched by Ibrahim et al. [4] and the new full-mobile robot and control system researched by Kilin et al. [5]. The perception of force information is conducive to improving the robustness of the hexarotor and fully movable robot control. Six-axis force/torque sensors are widely used in industrial robot collision detection and feedback control [6–8]. In some smarter robot systems, six-axis force/torque sensors are used to implement functions such as tactile sensing, safety control, and collision detection [9–14]. It can be seen that collision detection and feedback control are two key functions of force perception, and collision position detection is one of the important contents of collision detection.

There have been many researches on collision detection algorithms using six-axis force/torque sensors in the past few years. Bicchi et al. [15] proposed a solution for contact position detection based on a six-axis force/torque sensor as

early as 1990, which allows for practical devices that provide simple, relevant contact information in practical robotic applications. Kazanzides et al. proposed a method for detecting collision points by a linear regression method in a surgical robot system in [16], but did not give a specific calculation process. Zhou et al. proposed a mathematical model for determining the contact position between the fingertip and the object to be grasped/manipulated using measurement data provided by a force/torque sensor installed at the end of the fingertip in 1996, and the errors were analyzed [17]. The real-time selection algorithm described above was used to implement contact point measurement based on a six-axis force/torque sensor. The experimental results show that the test results still have good stability in the case of large noise and interference. Leng et al. [18, 19] used constraint equations and compatibility solution analysis to ensure collision point detection. In addition, gravity compensation and dynamic compensation are described, and the theory is proved to be effective in a gravity environment in experiments. The geometric and natural constraints of the collision process have also been analyzed by many scholars, and the gravity compensation method has also been applied to the collision detection of robots [10, 20].

Although the methods mentioned above provide a certain theoretical basis for the detection of robot collision points, there are still obvious shortcomings. The previous method has limited application range due to the reliance on geometric constraints to solve. Therefore, the collision objects must be regular in shape or geometrically modelable (such as a probe with a certain length, sphere, ellipse, and plane), and the collision surface must be smooth and cannot be deformed. The previous method of relying on geometric constraints to calculate the collision point has great limitations because most of the actual collision environment will be accompanied by deformation. In Figure 1, the collision types of robots can be divided into three types according to the number of contacts on the collision surface: single-contact collision (Figure 1(a)), flat or surface uniform contact collision (Figure 1(b)), and multicontact collision (Figure 1(c)). However, single-point collisions can be divided into structured collisions (which can model the collision body surface) and unstructured collisions as shown in (e) and (d) in Figure 1. In Figure 1(f), deformation is also an important issue to be considered in the collision process. In fact, collisions with uniform contact on the surface and collisions with multiple contacts can be considered as special combinations of single-contact collisions. Therefore, this article focuses on the analysis of single-contact collision and deformation in an unstructured environment.

In this paper, a robot collision point detection system based on six-axis force/torque sensor is proposed and experimental studies were carried out. We propose an error factor to search for the best in order to improve the detection accuracy. Compared with the traditional method that relies on the geometric information of the colliding body surface, the method proposed not only has no requirements on the surface of the colliding object, but also has good accuracy in the case of shape changes. The algorithm proposed solves the limitations of traditional geometric constraint methods

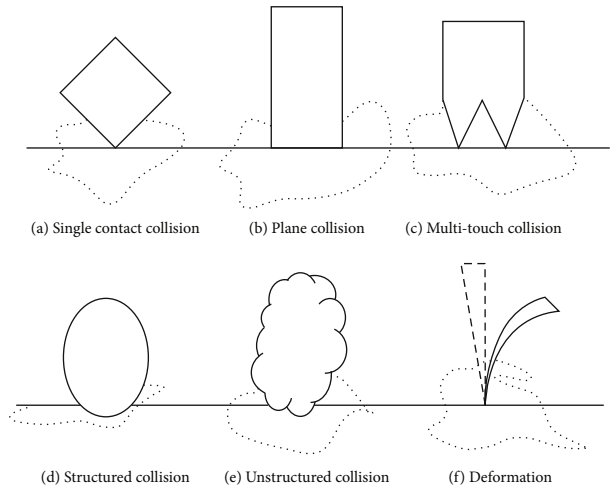


Figure 1: Collision type classification.

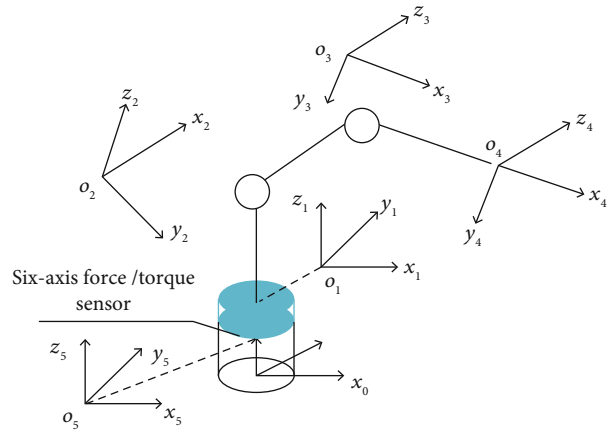


Figure 2: Schematic diagram of sensor installation location.

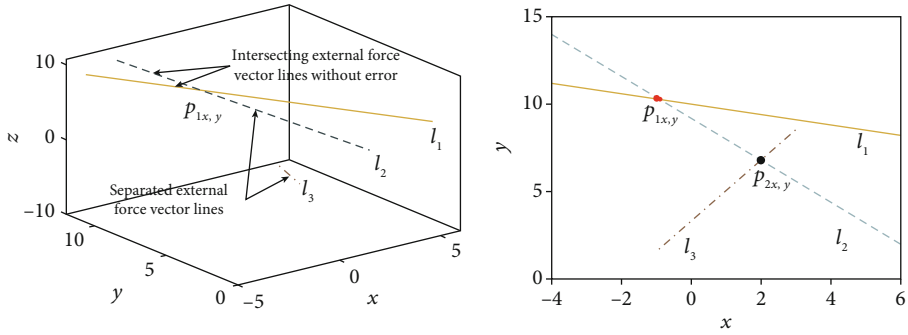
that require strict surface information and do not allow deformation. This paper effectively promotes the application of force perception in robot systems and also lays the foundation for the research of multicontact detection.

The remainder of this paper is organized as follows. Collision point detection algorithms and minimum error search strategies are proposed in Section 2. Section 3 proposes gravity compensation and dynamic force compensation algorithms. Section 4 conducts experimental research on the proposed collision point detection algorithm and verifies the correctness and accuracy of the algorithm. Finally, the paper is summarized in Section 5.

2. Collision Point Detection Algorithm

2.1. Mathematical Model for Collision Point Detection. Six-axis force/torque sensor is installed at the base of the robot in order to measure the collision points of the robot body extensively, which is shown in Figure 2.

Six-axis force/torque sensors can resolve forces and torque into components on three coordinate axes. Therefore,



(a) Two external force vector lines in space (b) Schematic diagram of space linear projection

Figure 3: Diagram of external force lines in space.

assuming that the collision force and moment are $\mathbf{F}(F_x, F_y, F_z)$ and $\mathbf{M}(M_x M_y M_z)$, the relationship between force and moment and contact position can be expressed as follows:

$$M = P \times F, \tag{1}$$

where $\mathbf{P}(P_x, P_y, P_z)$ is the vector of the position of the sensor to the contact point. Equation (1) can be expressed in matrix form as follows:

$$\begin{bmatrix} F_y & -F_x & 0 \\ 0 & F_z & -F_y \\ -F_z & 0 & F_x \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} M_x \\ M_y \\ M_Z \end{bmatrix}. \tag{2}$$

The equation system has multiple solutions for $|F| = 0$, so other constraints are needed to determine the unique contact position.

In fact, the solution P of Eq. (2) is distributed on a spatial straight line when $\mathbf{F}(F_x, F_y, F_z)$ and $\mathbf{M}(M_x M_y M_z)$ are constants. This space curve is usually called the external force vector line, which is defined as $L_c(p)$. The parameter form can be expressed as follows:

$$L_c(p): \frac{x}{F_x} = \frac{y + M_z/F_x}{F_y} = \frac{z + M_y/F_x}{F_z} L_c(p). \tag{3}$$

The external force vector line is determined by the collision force, and $\mathbf{F}(F_x, F_y, F_z)$ is the direction vector of the straight line. Generally, the direction of the collision object's movement and the direction of the collision force cause two sets of external force vector lines to intersect at different times, and the intersection point is the collision contact point. Therefore, the position of the contact point can be obtained by solving the intersection of the two sets of external force vector lines, which can be expressed as follows:

$$\{P(x, y, z) | (x, y, z) \in L_{c1}(p) \cap L_{c2}(p)\}. \tag{4}$$

2.2. Projection Method for Collision Points. The external force vector line calculated using the data of two adjacent frames of the sensor under ideal conditions without error is shown as l_1 and l_2 in Figure 3(a) and intersects at point $p_{1x,y}$ in space.

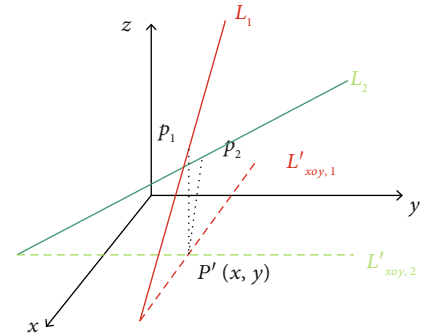


Figure 4: Basic principle diagram of projection method.

However, the two external force vector lines may not intersect in space due to the measurement error of the sensor, which is shown as l_2 and l_3 in Figure 3(a). Therefore, a projection method is proposed in this paper to solve the contact coordinates. The two external force vector lines l_2 and l_3 in space will not intersect at a certain point in space due to the existence of errors, but the projected straight lines on a certain plane will intersect at point $p_{2x,y}$, as shown in Figure 3(b).

The projection point coordinates of the contact point in three coordinate planes can be obtained by projecting the intersection point, and then the three-dimensional coordinate value of the contact point can be obtained by combining the original external force vector line equation, which is shown in Figure 4.

The projection of $L_c(p)$ on the XOY plane is set to L'_{XOY}, the projection on the XOZ plane is L'_{XOZ}, and the projection on the YOZ plane is L'_{YOZ}. Then, we can get

$$\begin{cases} L'_{XOY}: F_y F_z x - F_x F_z y = -F_x M_x - F_y M_y \\ L'_{XOZ}: F_y F_z x - F_x F_y z = F_x M_x + F_z M_z \\ L'_{YOZ}: F_x F_z y - F_x F_y z = -F_y M_y - F_z M_z. \end{cases} \tag{5}$$

$P'(x, y)$ is the intersection of the projection lines $L'_{XOY,1}$ and $L'_{XOY,2}$, andP_1, P_2 are the corresponding points of the projection points on L_{c1} and L_{c2}. P_1 and P_2 can be solved by the following formulas:

$$\begin{cases} \eta L_1'\left(P'\right) \\ \eta L_2'\left(P'\right) \\ L_{c1}(p) \end{cases} \text{and} \begin{cases} \eta L_1'\left(P'\right) \\ \eta L_2'\left(P'\right) \\ L_{c2}(p) \end{cases}, \tag{6}$$

where η is a determination factor. η is used to select an optimal projection plane from the three coordinate planes, and the specific rules are as follows:

$$\begin{cases} L_n' = L_{XOY,n}', \max\left(\theta_{XOY}, \theta_{XOZ}, \theta_{YOZ}\right) = \theta_{XOY} \\ L_n' = L_{XOZ,n}', \max\left(\theta_{XOY}, \theta_{XOZ}, \theta_{YOZ}\right) = \theta_{XOZ} \\ L_n' = L_{YOZ,n}', \max\left(\theta_{XOY}, \theta_{XOZ}, \theta_{YOZ}\right) = \theta_{YOZ}, \end{cases} \tag{7}$$

where $n = \{1, 2\}$, and θ_{XOY} represents the angle between $L_{XOY,1}'$ and $L_{XOY,2}'$, which can be calculated with the following formula:

$$\begin{cases} \theta_{XOY} = \dfrac{F_{y1}}{F_{x1}} - \dfrac{F_{y2}}{F_{x2}} \\ \theta_{XOZ} = \dfrac{F_{z1}}{F_{x1}} - \dfrac{F_{z2}}{F_{x2}} \\ \theta_{YOZ} = \dfrac{F_{y1}}{F_{z1}} - \dfrac{F_{y2}}{F_{z2}}. \end{cases} \tag{8}$$

The solving formulas for $P_1(x_1, y_1, z_1)$ and $P_2(x_2, y_2, z_2)$ can be expressed as Eq. (9) and Eq. (10) if XOY is the final selected projection plane after the above calculation.

$$\begin{bmatrix} F_{y1}F_{z1} & -F_{x1}F_{z1} & 0 \\ F_{y2}F_{z2} & -F_{x2}F_{z2} & 0 \\ -F_{z1} & 0 & -F_{x1} \end{bmatrix} \begin{bmatrix} \mathrm{x} \\ y \\ z_1 \end{bmatrix} = \begin{bmatrix} -F_{x1}M_{x1} - F_{y1}M_{y1} \\ -F_{x2}M_{x2} - F_{y2}M_{y2} \\ M_{y1} \end{bmatrix}, \tag{9}$$

$$\begin{bmatrix} F_{y1}F_{z1} & -F_{x1}F_{z1} & 0 \\ F_{y2}F_{z2} & -F_{x2}F_{z2} & 0 \\ -F_{z2} & 0 & -F_{x2} \end{bmatrix} \begin{bmatrix} \mathrm{x} \\ y \\ z_2 \end{bmatrix} = \begin{bmatrix} -F_{x1}M_{x1} - F_{y1}M_{y1} \\ -F_{x2}M_{x2} - F_{y2}M_{y2} \\ M_{y2} \end{bmatrix}. \tag{10}$$

As it is impossible to confirm which point the real collision point is closer to, the contact point is temporarily defined as the midpoint of P_1 and P_2, which can be expressed as follows:

$$P = \frac{1}{2}(P_1 + P_2). \tag{11}$$

2.3. Least Error Search. This paper proposes a method to search the optimal solution among multiple sets of data during the collision process in order to further reduce the error and improve the robustness of contact position detection. Assuming that the external force vector lines generated by multiple sets of force sensor data during the collision are shown in Figure 5, the contact set obtained by the projection method using two sets of adjacent data is shown in Figure 6, where the purple point in the point set represents the real contact position, and the orange point is the optimal solution searched by the minimum error method.

If $P''(x'', y'', z'')$ is the result of the collision point calculation, then $\zeta(\zeta_{M_x}, \zeta_{M_y}, \zeta_{M_z})$ is introduced as a precision factor to reduce the calculation error. The Eq. (2) can be expressed as follows:

$$\begin{bmatrix} F_y & -F_x & 0 \\ 0 & F_z & -F_y \\ -F_z & 0 & F_x \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} M_x + \zeta_{M_x} \\ M_y + \zeta_{M_y} \\ M_Z + \zeta_{M_z} \end{bmatrix}, \tag{12}$$

then

$$\begin{cases} \zeta_{M_x} = F_z y'' - F_y z'' - M_x \\ \zeta_{M_y} = F_x z'' - F_z x'' - M_y \\ \zeta_{M_z} = F_y x'' - F_x y'' - M_z. \end{cases} \tag{13}$$

Assuming $|\zeta| = \sqrt{\zeta^2_{M_x} + \zeta^2_{M_y} + \zeta^2_{M_z}}$, the smallest point $|\zeta|$ in the multiple sets of data is the optimal result of collision point detection. Finally, the improved contact position detection method of the system can be summarized as the flow shown in Figure 7.

3. Dynamic Force Compensation Algorithm

3.1. Gravity Compensation Algorithm. The influence of the gravity of the robot body and the end load on the sensor during the movement should be eliminated first when the six-axis force/torque sensor is installed at the base of the robot, and the dynamic force compensation of the sensor should be performed so that the sensor is not changed in the state of robot motion. After dynamic force compensation is performed on the sensor, the reading is constant to zero when the robot is in motion and is not impacted by external forces. It indicates that the robot collided unexpectedly with the outside world, when the sensor reading exceeds the threshold. Robot gravity compensation is shown in Figure 8.

The D-H parameters are used to establish the joint coordinate system $\{i\}$ of the robot. The general formula of the homogeneous transformation matrix ${}_i^{i-1}T$ of the connecting rod can be expressed as follows:

$${}_i^{i-1}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & \mathrm{a}_{i-1} \\ \cos\alpha_{i-1}\sin\theta_i & \cos\alpha_{i-1}\cos\theta_i & -\sin\alpha_{i-1} & -d_i\sin\alpha_{i-1} \\ \sin\alpha_{i-1}\sin\theta_i & \sin\alpha_{i-1}\cos\theta_i & \cos\alpha_{i-1} & d_i\cos\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{14}$$

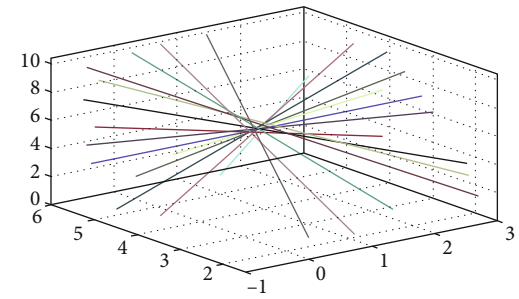


Figure 5: Multiple sets of external force vector lines during collision.

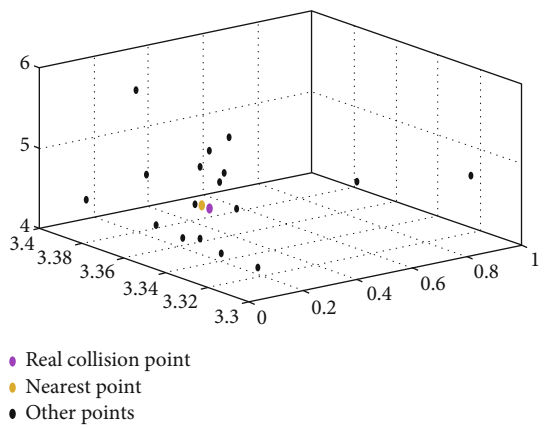


Figure 6: Result of collision point solution.

Equation (14) can be expressed as follows:

$$ {}_{i}^{i-1}T = \left[\begin{array}{c|c} {}_{i}^{i-1}R & {}_{i}^{i-1}\boldsymbol{p} \\ \hline 0\,0\,0 & 1 \end{array}\right], \tag{15} $$

where $\mathbf{{}_{i}^{i-1}P} = [p_x\, p_y\, p_z]^T$, ${}_{i}^{i-1}R$is the rotation matrix of coordinate system i relative to coordinate system $i-1$, which can be expressed as follows:

$$ {}_{i}^{i-1}\mathbf{R} = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}, \tag{16} $$

where ${}_{i}^{i-1}R$is a unit orthogonal matrix, ${}_{i-1}^{i}\mathbf{R} = {}_{i}^{i-1}\mathbf{R}^T = {}_{i}^{i-1}\mathbf{R}^{-1}$.

The pose transformation matrix between adjacent links can be obtained in turn by bringing the parameters of each link of the robot into Eq. (14). The transformation matrix of the coordinate system $\{i\}$ relative to the coordinate system {0} can be expressed as follows:

$$ {}_{i}^{0}T = {}_{1}^{0}T_{2}^{1}T_{3}^{2}T \cdots {}_{i}^{i-1}T = \begin{bmatrix} {}_{i}^{0}R & {}_{i}^{0}p \\ 0 & 1 \end{bmatrix}, \tag{17} $$

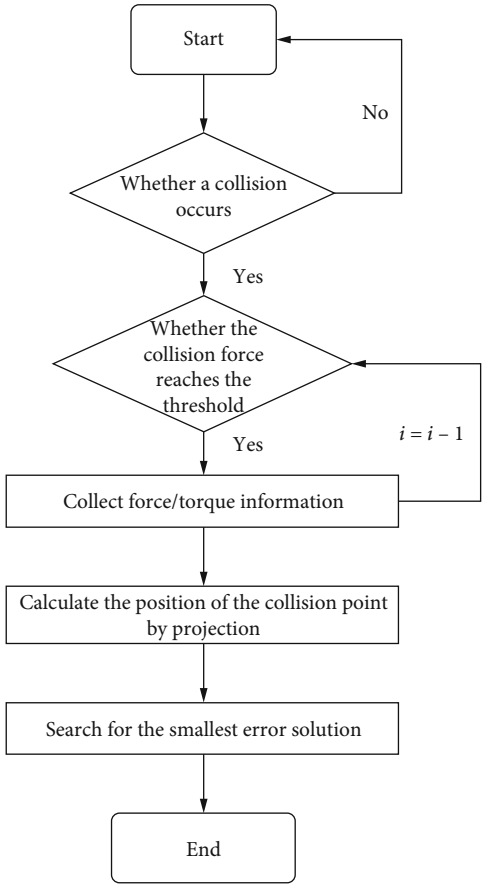


Figure 7: Flow chart of collision point position detection.

The position vector of the center of mass c_i of the joint i relative to the joint coordinate system i is as follows:

$$ c_i = \left[c_{ix}\, c_{iy}\, c_{iz}\right]^T. \tag{18} $$

If the gravity of each joint of the robot is G_i, then the gravity vector of each link can be expressed in the base coordinate system as follows:

$$ \mathbf{G}_i = [0\,0\,G_i]. \tag{19} $$

The component of the gravity of each joint at its center of mass will change with the robot's posture. It can be known from Eq. (18) that the rotation matrix of the joint i attachment coordinate system $\{i\}$ relative to the base coordinate system is ${}_{i}^{0}R$. The centroid coordinate system $\{c_i\}$ of the connecting rod i has the same direction as the coordinate system $\{i\}$. Therefore, the gravity vector of each joint on the coordinate system $\{c_i\}$ can expressed as follows:

$$ {}^{c_i}G_i = {}^{i}G_i = {}_{i}^{0}R^{T\,0}G_i. \tag{20} $$

The homogeneous transformation matrix of the joint centroid coordinate system $\{c_i\}$ and the joint coordinate system $\{i\}$ can be shown as

$$ {}^{i}_{c_i}\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & c_{ix} \\ 0 & 1 & 0 & c_{iy} \\ 0 & 0 & 1 & c_{iz} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{21} $$

The homogeneous transformation matrix of the centroid coordinate system $\{c_i\}$ of each link with respect to the {0} system can be expressed as follows:

$$ {}^{0}_{c_i}T = {}^{0}_{i}T\,{}^{i}_{c_i}\mathbf{T} = \begin{bmatrix} {}^{0}_{c_i}R & {}^{0}_{c_i}p \\ 0 & 1 \end{bmatrix}, \tag{22} $$

where ${}^{0}_{c_i}P = \begin{bmatrix} p_{cx} & p_{cy} & p_{cz} \end{bmatrix}^{\mathbf{T}}$. The force vector and moment vector of the robot's own gravity on the {0} system in the stationary state can be expressed as follows:

$$ \begin{aligned} {}^{0}\mathbf{f} &= \begin{bmatrix} 0\,0 & \sum_{i=1}^{n} G_i \end{bmatrix}^{\mathbf{T}}, \\ \begin{bmatrix} {}^{0}\mathbf{m} \end{bmatrix} &= \begin{bmatrix} -\sum_{i=1}^{n}\left(G_i \times p_{cy}\right) & -\sum_{i=1}^{n}\left(G_i \times p_{cx}\right) & 0 \end{bmatrix}^{\mathbf{T}}. \end{aligned} \tag{23} $$

The force sensor coordinate system $\{S\}$ has the same attitude as the base coordinate system {0} and is offset by - h on the Z-axis; then, we can get the transformation matrix of ${}^{0}\mathbf{f}$ and ${}^{0}\mathbf{m}$ in the coordinate system {0} and the coordinate system $\{S\}$ which can be expressed as follows:

$$ \begin{bmatrix} {}^{S}f \\ {}^{S}m \end{bmatrix} = \begin{bmatrix} {}^{S}_{0}R & \mathbf{0} \\ S\left({}^{S}p_0\right){}^{S}_{0}R & {}^{S}_{0}R \end{bmatrix} \begin{bmatrix} {}^{0}\mathbf{f} \\ {}^{0}\mathbf{m} \end{bmatrix}, \tag{24} $$

where

$$ {}^{S}_{0}R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{25} $$

$$ S\left({}^{S}p_0\right) = \begin{bmatrix} 0 & -P_z & p_y \\ P_z & 0 & p_x \\ -p_y & p_x & 0 \end{bmatrix} = \begin{bmatrix} 0 & h & 0 \\ h & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{26} $$

3.2. Dynamic Force Compensation Algorithm. It is obviously not enough to only compensate the sensor for gravity during the robot movement. At this time, the inertial force and the coercive force generated during the movement of the robot need to be considered. Therefore, the robot dynamic force compensation algorithm is proposed in this section.

We assume that the mass of each joint arm of the robot is $\mathbf{m_i}(i = 1, 2, \cdots, 6)$, the position vector of the centroid c_i of the joint i with respect to the coordinate system $\{i\}$ is $\mathbf{r_i}(i = 1, 2, \cdots, 6)$, the displacement of joint i is $\theta_{\mathbf{i}}$, the speed is $\dot{\theta}_{\mathbf{i}}$, and the acceleration is $\ddot{\theta}_{\mathbf{i}}$. Then, we can get

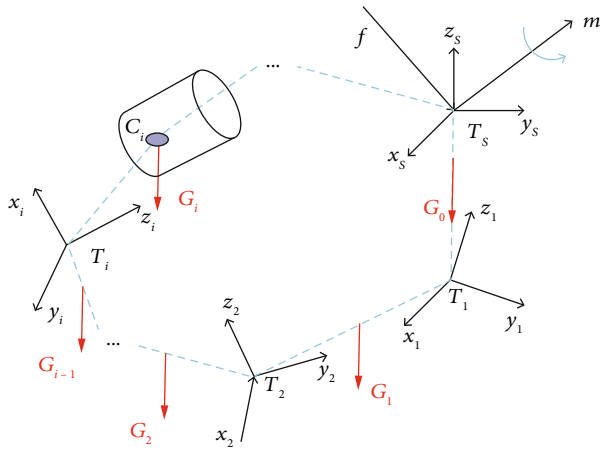


Figure 8: Gravity analysis.

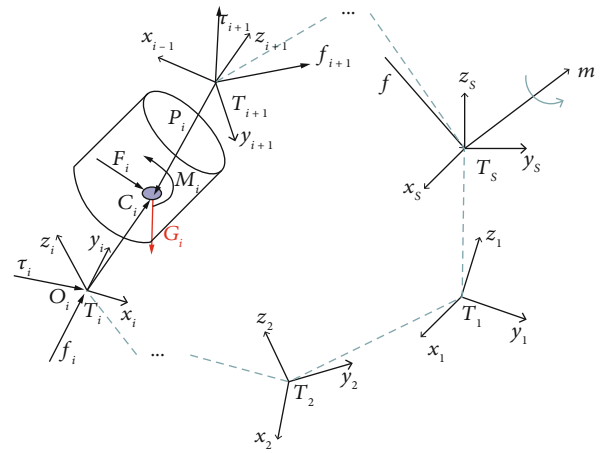


Figure 9: Link force diagram.

$$ \omega_{i+1} = {}^{i+1}_{i}R\omega_i + \dot{\theta}_{i+1}Z_{i+1}, \tag{27} $$

$$ \dot{\omega}_{i+1} = {}^{i+1}_{i}R\dot{\omega}_i + {}^{i+1}_{i}R\omega_i \times \dot{\theta}_{i+1}e_{i+1} + \ddot{\theta}Z_{i+1}, \tag{28} $$

$$ \dot{v}_{i+1} = {}^{i+1}_{i}R\left[\dot{v}_i + \dot{\omega}_i \times {}^{i}_{i+1}p + \omega_i \times \left(\omega_i \times {}^{i}_{i+1}p\right)\right], \tag{29} $$

$$ \dot{v}_{c_{i+1}} = \dot{v}_{i+1} + \dot{\omega}_{i+1} \times r_{i+1} + \omega_{i+1}(\omega_{i+1} \times r_{i+1}), \tag{30} $$

where ω_i is the angular velocity of link i, $\dot{\omega}_i$ is the angular acceleration of link i, $\dot{v}_i$ is the linear acceleration of link i, $\dot{v}_{ci}$ is the linear acceleration of the center of mass of link i, and $\omega_0 = \dot{\omega}_0 = v_0 = \dot{v}_0 = [0\,0\,0]^T$, $Z_i(i = 1, 2, \cdots, 6) = [0\,0\,1]^T$. The schematic diagram of dynamic force compensation for the sensor is shown in Figure 9. Then, we can get the force/torque relationship between the links, which can be expressed as follows:

$$ \mathbf{F_i^{i-1}} = \mathbf{F_i^{i+1}} + \mathbf{m_i}\dot{\mathbf{v}}_{\mathbf{c_i}} - {}^{c_i}\mathbf{G}_i, \tag{31} $$

$$\mathbf{M_i^{i-1}} = \mathbf{M_i^{i+1}} - \mathbf{r_{i+1,c_i}} \times \mathbf{F_i^{i+1}} + \mathbf{r_{i,c_i}} \times \mathbf{F_i^{i-1}} + \mathbf{I_i}\dot{\boldsymbol{\omega}}_\mathbf{i} + \omega_\mathbf{i} \times \mathbf{I_i}\omega_\mathbf{i}, \tag{32}$$

where $\mathbf{F_i^{i-1}}$ is the acting force of rod i-1 on rod i; $\mathbf{F_i^{i+1}}$ is the acting force of rod $i+1$ on rod i; $\mathbf{M_i^{i-1}}$ is the resistance of rod i-1 to rod i; $\mathbf{M_i^{i+1}}$ is the resistance of rod $i+1$ to rod i; $\mathbf{r_{i,c_i}}$ is the sagittal diameter from the origin O_i of the coordinate system attached to the joint i to the center of massc_i; $\mathbf{r_{i+1,c_i}}$ is the sagittal diameter from the origin O_{i+1} of the coordinate system attached to the joint $i+1$ to the center of massc_{i+1};$\mathbf{I_i}$ is the inertia tensor of the rod i with respect to its center of mass c_i ; $\omega_\mathbf{i} \times \mathbf{I_i}\omega_\mathbf{i}$ is the scientific force term.

For n-degree-of-freedom robots, the value of $\mathbf{F}_n^{n+1}$is related to the end load of the robot. $\mathbf{F}_n^{n+1} = 0$ when the end is unloaded. From Eqs. (27)–(32), F_0^1 and M_0^1 can be calculated. Take them into the Eqs. (24)–(26) to get the specific dynamic force compensation values ${}^S\mathrm{f}$ and ${}^S\mathbf{M}$ that are required by the six-axis force/torque sensor at the base. If the force/torque sensor readings of the six-axis force/torque sensor on the base are ${}^D\mathbf{f}$ and ${}^D\mathbf{M}$ when the robot is in a certain posture, the force/torque information after gravity compensation can be expressed as follows:

$$\begin{bmatrix} F \\ M \end{bmatrix} = \begin{bmatrix} {}^Df - {}^Sf \\ {}^DM - {}^SM \end{bmatrix}. \tag{33}$$

After dynamic force compensation, the robot can keep the six-axis force/torque sensor reading constant to 0 under the condition of no external force during the movement. Once the reading of the six-axis force/torque sensor at the spindle exceeds the set threshold, the robot can be considered to collide with the outside. The threshold value needs to be determined according to the characteristics of different robot systems. The collision between the robot and the outside world is a continuous process, so each frame of data read by the sensor at the base can be used as input to the collision point detection algorithm after dynamic force compensation.

4. Simulation Experiment

4.1. Simulation Verification Experiment. A three-degree-of-freedom robot was built to perform simulation experiments to verify the correctness and effectiveness of the algorithm proposed in this paper, as shown in Figure 10. The links and joints of the manipulator are center symmetrical to reduce the amount of calculation, and the center of gravity of each link is located on its own central axis. The structural parameters of each link of the three-degree-of-freedom robot used in this paper are shown in Table 1, and the material is set to alloy ($\rho = 2.7 \times 10^3 \mathrm{kg/m^2}$). In Figure 10, $l_1 = 120\mathrm{mm}$, $l_2 = 150\mathrm{mm}$, $l_3 = 200\mathrm{mm}$, $l_4 = 200\mathrm{mm}$, and $h = 30\mathrm{mm}$.

The date of six-axis force/torque sensor at the base will change continuously over time. The data collected by the sensor at the base is preprocessed and brought into the proposed collision point detection algorithm. The robot model is imported into ADAMS, and two sets of simulation experiments are designed to verify the proposed dynamic force compensation algorithm and collision point detection algorithm. The magnitude, direction, and position of the collision force applied in these two sets of experiments are known, and the data in the experimental results are based on the sensor coordinate system O_S. The first set of experiments was designed to verify the proposed gravity compensation algorithm and collision point detection algorithm, and an equal amount of increased collision force was applied at the same collision point $p_c = (204.32, 68.96, 518.46)$. The robot was set to be stationary in this set of experiments, and $\theta_1 = -\pi/6$, $\theta_2 = -\pi/6$, $\theta_3 = -\pi/4$, and $h = 30\mathrm{mm}$. The results of the experiment are shown in Table 2, where the collision direction is obtained by calculating the difference of the coordinate system data of the sensor before and after the force is applied in the ADAMS. Calculation force and calculate position can be obtained by Eq. (24) and Eq. (11) and a series of coordinate transformations.

In order to verify the effectiveness and accuracy of the dynamic force compensation algorithm and collision point detection algorithm in the experiment 2, the following assumptions were added on the basis of experiment 1: $\dot{\theta}_1 = \dot{\theta}_2 = \dot{\theta}_3 = 0$; $\ddot{\theta}_1 = 0.3491\mathrm{rad}/s^2$, $\ddot{\theta}_2 = 0.1746\mathrm{rad}/s^2$, and $\ddot{\theta}_3 = 0.4363\mathrm{rad}/s^2$. The robot moves according to the set parameters in experiment 2 and performs the collision test at different points with the same collision force when running to the 3rd second. Specific experimental data is shown in Table 3.

4.2. Error Calculation Method. The error in the above table is calculated to verify the effectiveness of the proposed algorithm. Equation of them is shown as follows:

Force direction error:

$$\Delta F = |F_C - F_D|, \tag{34}$$

where F_C is the collision force direction, and F_D is the calculate force direction.

Relative error of force in all directions:

$$\varphi F = \frac{|F_C - F_D|}{\|F_C\|}. \tag{35}$$

Force error:

$$\Delta F_D = \frac{\left|\sqrt{f_{Cx}^2 + f_{Cy}^2 + f_{Cz}^2} - \sqrt{f_{Dx}^2 + f_{Dy}^2 + f_{Dz}^2}\right|}{\sqrt{f_{Cx}^2 + f_{Cy}^2 + f_{Cz}^2}}. \tag{36}$$

In Eq. (36), f_{Cx}, f_{Cy}, f_{Cz} and f_{Dx}, f_{Dy}, f_{Dz} are the component forces of the measured force and the calculated force, respectively.

Position error:

$$\Delta P = |P_C - P_D|, \tag{37}$$

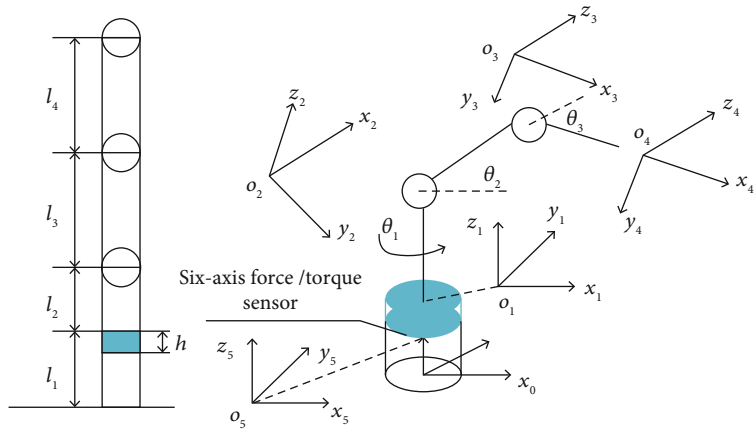


Figure 10: Robot structure.

Table 1: Connecting rod parameters.

Link	Quality(kg)	Center of mass coordinates in each link coordinate system (mm)
0	1.598	(0,0,60)
1	1.686	(0,0,70.024)
2	2.489	(84.685,0,0)
3	3.201	(104.684,0,0)

where P_C is the known collision position, and P_D is the calculated collision position.

Relative error of position in all directions:

$$\varphi P = \frac{|P_C - P_D|}{\|P_C\|}. \tag{38}$$

Position error (%):

$$\Delta P_D = \frac{\Delta P_x + \Delta P_y + \Delta P_z}{\sqrt{P_{Cx}^2 + P_{Cy}^2 + P_{Cz}^2}}. \tag{39}$$

4.3. Analysis of Results. Two sets of experiments are designed to reflect the effectiveness and accuracy of the algorithm proposed in this paper. Experiment 1 and experiment 2 take the magnitude of collision force and collision distance as two independent variables. Figures 11 and 12 show the relative error and absolute error of each direction force and the position of the collision point in experiment 1 with the increase of the collision force. Figures 13 and 14 show the relative error and absolute error of each direction force and the position of the collision point in experiment 2 as the collision distance increases. Figure 15 shows the overall error of the force and the location of the impact point.

4.3.1. Error Analysis of Experiment 1 Results. In Figures 11 and 12, the absolute error $(\Delta Fx, \Delta Fy, \Delta Fz)$ and relative error$(\varphi Fx, \varphi Fy, \varphi Fz)$ of the force in all directions are kept within a reasonable range when the collision point is fixed, and there is no obvious change trend with the increase of the force. However, the absolute position error $(\Delta Px, \Delta Py, \Delta Pz)$ and relative position error $\varphi Fx, \varphi Fy, \varphi Fz$ in all directions have a clear tendency to decrease with increasing force. Therefore, if the method proposed in this paper is used to detect the collision point, the position accuracy of the minimum detectable collision external force calculation must be achieved.

4.3.2. Error Analysis of Experiment 2 Results. Figures 13 and 14 show the force error and position error of the system in different directions under the same collision external force at different collision points. If the denominator is 0 when performing the calculation in Eq. (38), the calculation result is output as 0. When using the fifth set of data in Table 3 to calculate φPy, this value is eliminated as a singular value, because the difference between the calculation results is very large. It can be seen that the absolute error $(\Delta Fx, \Delta Fy, \Delta Fz)$ and relative error $(\varphi Fx, \varphi Fy, \varphi Fz)$ of the component force in all directions fluctuate within a certain range as the distance of the collision point changes when the collision force is constant. But the absolute error$(\Delta Px, \Delta Py, \Delta Pz)$ and relative error $\varphi Fx, \varphi Fy, \varphi Fz)$ in each direction of the collision position continue to increase as the distance of the collision point increases. So using this method for collision point detection needs to meet the robot's farthest collision accuracy to meet the requirements.

4.3.3. The Overall Error Analysis of the Experiment. The relative error results of the force and collision points in the two sets of experiments calculated by Eq. (36) and Eq. (39) are shown in Figure 15.

It can be concluded that the error of the component force in each direction fluctuates within 5%, and the maximum relative error of the resultant force is 4.8952% from Figures 12, 14, and 15. The accuracy of the dynamic force compensation algorithm proposed in this paper meets the requirements. However, it can be seen that as the distance of the collision point increases, the error keeps increasing from Figures 14 and 15. The relative error of the collision point coordinate

Table 2: Experimental results of different external forces.

Number	Collision force (N)	Collision direction (N)	Collision position (mm)	Calculation force (N)	Calculate position (mm)	Force direction error (N)	Force error (%)	Position error (mm)	Position error (%)
		-32.0642	204.32	-31.2463	217.9682	0.8179		13.6582	
1	40	20.4271	68.96	19.9976	74.1460	0.4295	1.5330	5.1860	7.102
		12.4347	518.46	13.2315	539.4948	0.7968		21.0348	
		-46.3017	204.32	-45.2675	216.6647	1.0342		12.3447	
2	60	33.9705	68.96	34.2894	75.0908	0.3279	0.5075	6.1308	6.864
		17.3827	518.46	18.4023	538.5227	1.0196		20.0672	
		-62.6532	204.32	-60.4687	216.3556	2.1845		12.0356	
3	80	49.3346	68.96	48.6750	74.4850	0.6596	2.6612	5.5250	6.541
		6.3780	518.46	6.1794	539.4948	0.1986		19.1684	
		-81.4014	204.32	-78.6781	192.4465	2.7233		11.8735	
4	100	56.3467	68.96	54.7614	73.076	1.5853	2.8481	4.1160	6.173
		14.1018	518.46	15.7934	537.1331	1.6916		18.6731	
		-95.9861	204.32	-98.7495	194.1437	2.7634		10.1763	
5	120	69.7804	68.96	70.6451	64.8687	0.8647	2.1875	4.0913	5.684
		17.8147	518.46	17.1665	500.8108	0.6482		17.6492	
		-114.0861	204.32	-117.1907	214.2127	3.1046		9.8927	
6	140	80.6483	68.96	79.9509	65.4977	0.6974	1.5464	3.4623	5.493
		8.9562	518.46	9.2236	535.9493	0.2674		17.4893	
		-126.5287	204.32	-124.5523	214.2584	1.9764		9.9384	
7	160	96.1673	68.96	97.4040	72.9644	1.2367	0.4302	4.0044	5.208
		18.5024	518.46	19.4785	533.7612	0.9761		15.3012	
		-143.1024	204.32	-145.1510	211.9638	2.0486		7.6438	
8	180	105.8671	68.96	106.8833	74.1596	1.0162	1.3090	5.1996	4.431
		26.7182	518.46	27.5855	506.4224	0.8673		12.0376	
		-161.4531	204.32	-160.3099	196.2409	1.1432		8.0791	
9	200	115.6146	68.96	114.9674	65.5742	0.6472	0.6092	3.3858	4.173
		23.7941	518.46	24.4413	530.4273	0.6472		11.9673	
		-177.0267	204.32	-174.6555	211.1113	2.3712		6.7913	
10	220	127.3468	68.96	126.2738	63.8338	1.0942	1.2027	5.1262	3.675
		29.0575	518.46	28.1632	509.7417	0.8943		8.7183	

at the position 598.61 mm farthest from the force sensor reaches the maximum value of 8.712%.

4.4. Error Source Analysis. It can be seen from Section 4.3 that the magnitude of the force and the distance from the collision position to the sensor have an impact on the calculation accuracy. The accuracy factor ϖ is proposed to verify the impact of collision force and collision distance on the accuracy of the collision point detection algorithm proposed in this paper, which can be expressed as follows:

$$\varpi_{\Delta P_{\mathrm{m}}} = \frac{1}{9}\sum_{i=1}^{10}\left(\Delta P_{mi} - \Delta \bar{P}_m\right)^2, (m = x, y, z), \tag{40}$$

$$\varpi_{\Delta P_D} = \frac{1}{9}\sum_{i=1}^{10}\left(\Delta P_{Di} - \Delta \bar{P}_D\right)^2. \tag{41}$$

The data of experiment 1 and experiment 2 are brought into Eqs. (40) and (41), and the specific calculation results are shown in Table 4.

The sensitivity of data to variables is reflected by the size of ϖ. In experiment 1, the experiment of collision point detection was performed using the magnitude of the force as a variable, and in experiment 2, the experiment was performed using the position of the collision force as a variable. It can be seen from Table 4 that the values of $\varpi_{\Delta P_x}$, $\varpi_{\Delta P_y}$, $\varpi_{\Delta P_z}$, and $\varpi_{\Delta P_D}$ in experiment 1 are all more smaller than those in experiment 2. Therefore, the impact of the collision position on the accuracy of the algorithm is much greater than the effect of the force. The location of the collision is the first factor affecting accuracy. So it is necessary to mainly consider the algorithm accuracy of the furthest part of the robot working space when using this algorithm for single external force collision point detection.

Table 3: Experimental results of different collision points.

Number	Collision force (N)	Collision direction (N)	Collision position (mm)	Calculation force (N)	Calculate position (mm)	Force direction error (N)	Force error (%)	Position error (mm)	Position error (%)
		92.6873	40.03	94.5629	40.4627	1.8756		0.4327	
1	100	10.3486	0	9.6745	0.0436	0.6741	2.2044	0.0436	1.5470
		36.0828	148.03	37.5501	149.926	1.4673		1.8960	
		62.3486	0	60.3813	1.3468	1.9673		1.3468	
2	100	73.9483	38.49	72.7500	37.6433	1.1983	2.3799	0.8467	1.7856
		25.3831	167.03	24.3152	167.8972	1.0679		0.8672	
		-82.9436	46.83	-84.1905	47.3161	1.2469		0.4861	
3	100	48.6725	0	49.6071	2.3791	0.9346	1.946	2.3791	2.3915
		-27.4107	184.16	-29.0514	85.8391	1.6407		1.6791	
		76.8942	64.07	78.7615	66.7183	1.8673		2.6483	
4	100	14.6879	36.83	16.4195	38.9973	1.7316	0.7565	2.1673	2.8540
		-62.2218	264.19	-61.6539	261.1763	0.5679		3.0137	
		-16.4834	82.64	-18.0505	79.1709	1.5671		3.4691	
5	100	84.6781	6.76	85.6615	9.4081	0.9834	1.5283	2.6481	3.9679
		-50.5759	306.82	-51.4226	313.3137	0.8467		6.4937	
		-45.0167	116.79	-46.3635	122.1008	1.3468		5.3108	
6	100	-62.8643	69.72	-62.2297	76.4891	0.6346	0.8760	6.7691	4.8264
		63.4159	298.71	64.4505	302.4713	1.0346		3.7613	
		-86.0348	190.46	-87.7972	180.7866	1.7624		9.6734	
7	100	32.1864	80.73	33.0516	92.2237	0.8652	2.2500	11.4937	7.5780
		39.5228	422.67	-40.6704	408.1766	1.1476		14.4934	
		72.1894	213.41	70.7132	226.9001	1.4762		13.4901	
8	100	30.1609	124.08	29.6686	135.7148	0.4923	0.5796	11.6348	7.7792
		-62.2817	486.49	-63.2758	503.8037	0.9941		17.3137	
		-80.7934	219.46	-82.5168	235.4646	1.7234		16.0046	
9	100	52.8792	97.61	54.3396	116.1037	1.4604	4.8925	18.4937	8.5263
		26.00417	519.73	27.1388	505.4114	1.1346		14.3186	
		-68.4937	234.18	-70.6343	253.8643	2.1406		19.6843	
10	100	-24.6791	126.49	-23.6394	139.9582	1.0397	1.8862	13.4682	8.7119
		-68.5533	536.18	-69.5181	555.1773	0.9648		18.9973	

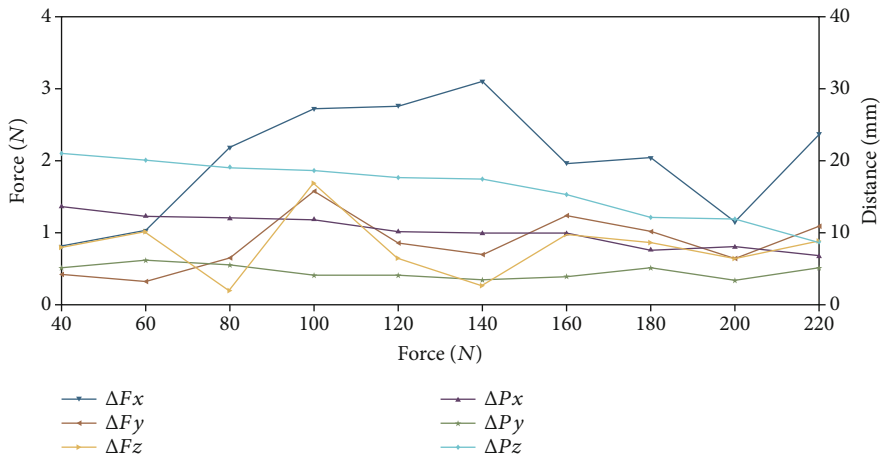


Figure 11: Absolute error relationship in all directions in experiment 1.

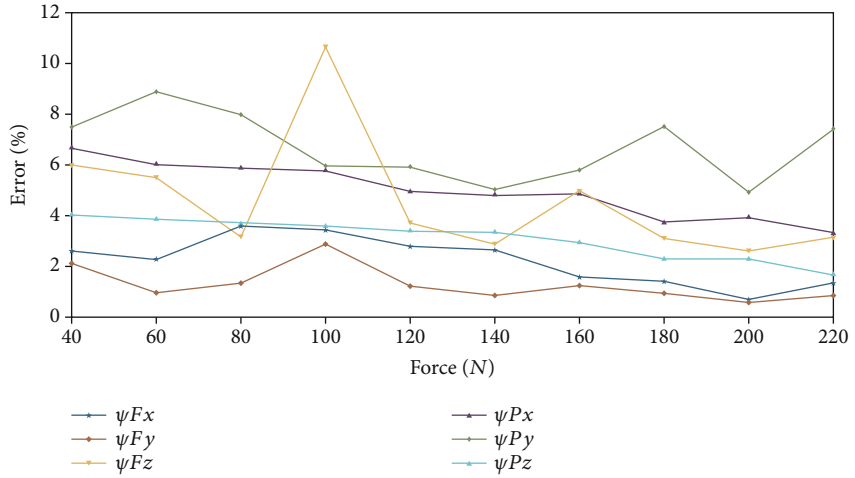


Figure 12: Relative error relationship in each direction in experiment 1.

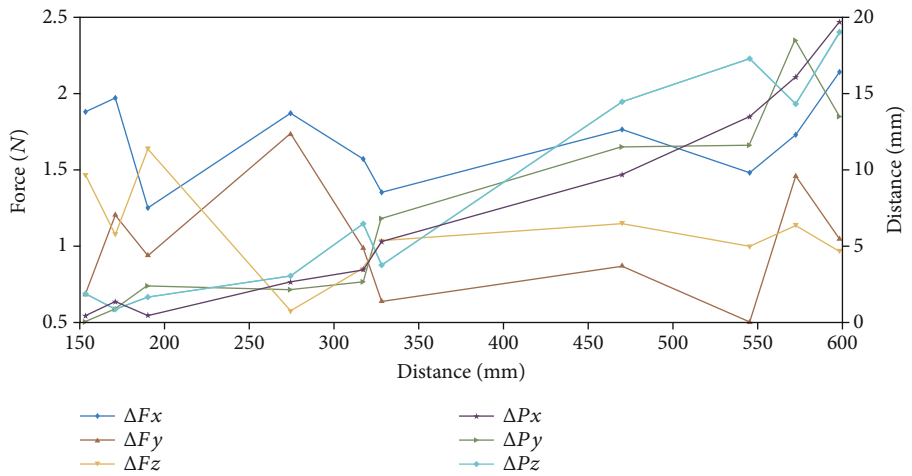


Figure 13: Absolute error relationship in each direction in experiment 2.

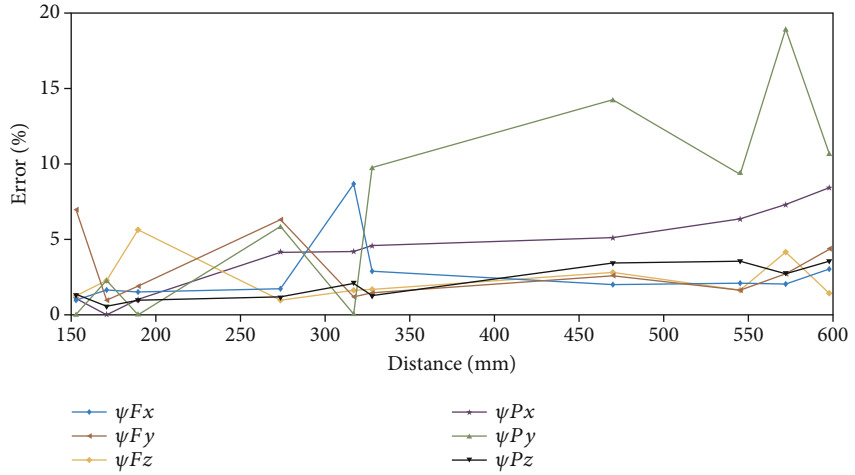


Figure 14: Relative error relationship in each direction in experiment 2.

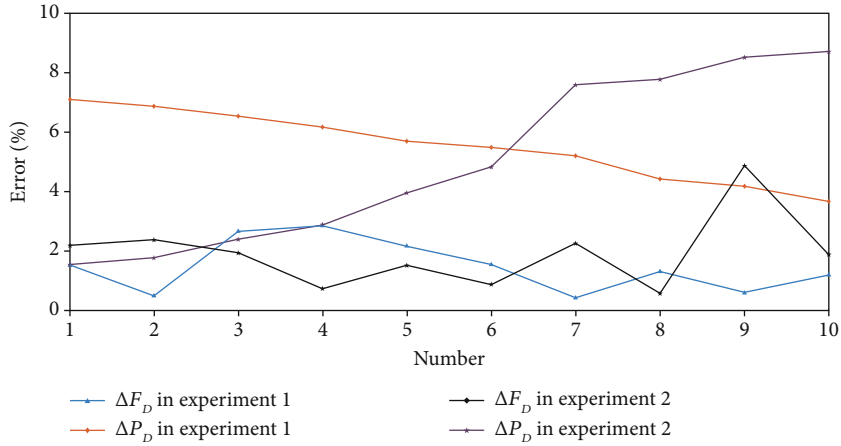


Figure 15: Relative error of force and position.

Table 4: Accuracy factor calculation result.

	Experiment 1	Experiment 2
$\omega_{\Delta P_x}$	5.0584	49.1524
$\omega_{\Delta P_y}$	0.8656	40.6191
$\omega_{\Delta P_z}$	16.5605	51.3511
$\omega_{\Delta P_D}$	1.3672	8.3715

5. Conclusion and Future Work

The detection of collision by the robot can make the robot interact with the environment, so that the robot can adapt to the environment and realize the safety of human-computer interaction. Collision sensing includes collision position, collision direction, and force size. Most of the existing methods use geometric constraints. Their disadvantages are that they cannot be applied to unstructured environments and cannot cope with the deformation of collision objects. In order to make up for the above problems, a robot collision point detection algorithm for single point and single external force is proposed in this paper, and the gravity compensation and dynamic force compensation algorithms of the six-axis force/torque sensors at the base are described, then perform simulation experiments to verify the algorithm. Compared with the traditional algorithms [18, 19] that use the geometric information of the collision surface to calculate the collision position, the algorithm proposed in this paper has a minimum error of only 1.5470%, and the maximum error is approximated by traditional methods. However, the advantage compared with the traditional method is that the proposed algorithm does not rely on the geometric information of the collision body surface, and the generation of deformation has no effect on the result. The experimental results prove the effectiveness of the proposed algorithm and show that the accuracy of the proposed gravity and dynamic force compensation algorithm does not change with the change of the force and the position of the collision point. The accuracy of the collision point detection algorithm is obviously affected by the magnitude of the collision force and the collision position. The accuracy of the algorithm in the furthest part of the robot's workspace should be considered first when applying this algorithm.

The application of the algorithm proposed in this paper in robot force drag teaching and application in medical collaborative robots needs to continue to be explored, which could be as follows: (1) how to use contact information as a basis for decision-making to achieve task division in human-machine collaboration and (2) how to use the trajectory of the touch point as the basis to realize the force teaching of the complex curved surface of the robot. Aiming at the application direction of the above two points, experimental research will be carried out in the follow-up to increase the practicality of the algorithm proposed in this paper.

Data Availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 51505124), the Science and Technology Research Project of Hebei Province (ZD2020151), the Foster Fund Projects of North China University of Science and Technology (No. JP201505), and the Natural Science Foundation of Hebei Province (E2016209312).

References

[1] Z. J. Wang, Z. X. Li, J. He, J. Yao, and Y. Zhao, "Optimal design and experiment research of a fully pre-stressed six-axis force/-torque sensor," *Measurement*, vol. 46, no. 6, pp. 2013–2021, 2013.

[2] Z. J. Wang, J. T. Yao, Y. D. Xu, and Y. Zhao, "Hyperstatic analysis of a fully pre-stressed six-axis force/torque sensor," *Mechanism and Machine Theory*, vol. 57, no. 2, pp. 84–94, 2012.

[3] J. Yao, L. Zu, H. Ruan, D. Cai, Y. Xu, and Y. Zhao, "A heavy load miniature six-component force sensing mechanism with hybrid branches," *Measurement*, vol. 157, p. 107623, 2020.

[4] I. N. Ibrahim, K. Izhevsk, B. Pavol, A. A. M. Aiman, K. Izhevsk, and A. Karam, "Navigation control and stability investigation of a hexacopter equipped with an aerial manipulater," in *2017 21st International Conference on Process Control (PC)*, Strbske Pleso, Slovakia, June 2017.

[5] A. Kilin, P. Bozek, Y. Karavaev, A. Klekovkin, and V. Shestakov, "Experimental investigations of a highly maneuverable mobile omniwheel robot," *International Journal of Advanced Robotic Systems*, vol. 14, no. 6, 2017.

[6] J. Shen and N. Gans, "Robot-to-human feedback and automatic object grasping using an RGB-D camera-projector system," *Robotica*, vol. 36, no. 2, pp. 241–260, 2018.

[7] S. Zhou, J. Sun, and F. Gao, "Influence of flexible spherical joints parameters on accuracy of the six-axis force/torque sensor with three-three orthogonal parallel mechanism," *Mechanism and Machine Theory*, vol. 145, p. 103697, 2020.

[8] R. Xue, Z. Du, Z. Yan, and B. Ren, "An estimation method of grasping force for laparoscope surgical robot based on the model of a cable-pulley system," *Mechanism & Machine Theory*, vol. 134, pp. 440–454, 2019.

[9] J. T. Yao, H. Y. Zhang, J. L. Zhu, Y. Xu, and Y. Zhao, "Isotropy analysis of redundant parallel six-axis force sensor," *Mechanism & Machine Theory*, vol. 91, pp. 135–150, 2015.

[10] C. He, S. Wang, H. Sang, J. Li, and L. Zhang, "Force sensing of multiple-DOF cable-driven instruments for minimally invasive robotic surgery," *International Journal of Medical Robotics & Computer Assisted Surgery*, vol. 10, no. 3, pp. 314–324, 2014.

[11] T. D. Tang, "Algorithms for collision detection and avoidance for five-axis NC machining: a state of the art review," *Computer-Aided Design*, vol. 51, no. 6, pp. 1–17, 2014.

[12] R. C. Luo, M. H. Li, H. L. Jhu, and J. W. Chen, "Multi-sensor based object grasping with eye-in hand laser ranger," *Conference of the IEEE Industrial Electronics Society*, pp. 2289–2294, 2009.

[13] W. Shang, B. Zhang, S. Cong, and Y. Lou, "Dual-space adaptive synchronization control of redundantly-actuated cable-driven parallel robots," *Mechanism and Machine Theory*, vol. 152, p. 103954, 2020.

[14] A. A. Alqumsan, S. Khoo, and M. Norton, "Robust control of continuum robots using Cosserat rod theory," *Mechanism and Machine Theory*, vol. 131, pp. 48–61, 2019.

[15] A. Bicchi, J. K. Salisbury, and D. L. Brock, "Contact sensing from force measurements," *The International Journal of Robotics Research*, vol. 12, no. 3, pp. 249–262, 2016.

[16] P. Kazanzides, J. F. Zuhars, B. D. Mittelstadt, and R. H. Taylor, "Force sensing and control for a surgical robot," *IEEE International Conference on Robotics & Automation*, 2002, Nice, France, May 1992, 2002.

[17] X. Zhou, Q. Shi, and Z. Li, "Contact localization using force/-torque measurements," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1339–1344, Minneapolis, MN, USA, USA, April 1996.

[18] L. Yuquan, C. Zhengcang, Z. Wei et al., "Collision position sensing for manipulator using force/torque sensor," in *Proceedings of the 34th Chinese Control Conference*, pp. 5809–5814, Hangzhou, China, July 2015.

[19] Y. Q. Leng, Z. C. Chen, X. He, Y. Zhang, and W. Zhang, "Collision sensing using force/torque sensor," *Journal of Sensors*, vol. 2016, no. 3, p. 10, 2016.

[20] K. J. Xu, C. Li, and Z. N. Zhu, "Dynamic modeling and compensation of robot six-axis wrist force/torque sensor," *IEEE Transactions on Instrumentation & Measurement*, vol. 56, no. 5, pp. 2094–2100, 2007.