



Research Article

Forward solution algorithm of Fracture reduction robots based on Newton-Genetic algorithm

Jian Li^{a,b,c}, Xiangyan Zhang^{a,b}, Yadong Mo^{a,b}, Guang Yang^{a,b}, Yun Dai^{a,b}, Chengyu Lv^{a,b,*}, Ying Zhang^{a,b}, Shimin Wei^{a,b}

^a Laboratory of Robotics Mechanism and Cross Innovation, Beijing University of Posts and Telecommunications, Beijing 100876, China

^b School of Intelligent Engineering and Automation, Beijing University of Posts and Telecommunications, Beijing 100876, China

^c School of Electronic and Electrical Engineering, Ningxia University, Ningxia 750021, China

ARTICLE INFO

Article history:

Received 5 November 2024

Revised 23 December 2024

Accepted 16 January 2025

Available online 28 January 2025

Keywords:

Fracture reduction robot

Newton method

Genetic algorithm

Forward kinematics

Medical application

ABSTRACT

The Fracture Reduction Robot (FRR) is a crucial component of robot-assisted fracture correction technology. However, long-term clinical experiments have identified significant challenges with the forward kinematics of the parallel FRR, notably slow computation speeds and low precision. To address these issues, this paper proposes a hybrid algorithm that integrates the Newton method with a genetic algorithm. This approach harnesses the rapid computation and high precision of the Newton method alongside the strong global convergence capabilities of the genetic algorithm. To comprehensively evaluate the performance of the proposed algorithm, comparisons are made against the analytical method and the Additional Sensor Algorithm (ASA) using identical computational examples. Additionally, iterative comparisons of iteration counts and precision are conducted between traditional numerical methods and the Newton-Genetic algorithm. Experimental results show that the Newton-Genetic algorithm achieves a balance between computation speed and precision, with an accuracy reaching the 10^{-4} mm order of magnitude, effectively meeting the clinical requirements for fracture reduction robots in medical correction.

© 2025 The Author(s). Published by Elsevier B.V. on behalf of Shandong University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

External fixation technology has emerged as a significant method for limb fracture reduction and deformity correction, utilizing the tension-stress principle to traction bone and promote soft tissue regeneration [1–3]. Compared to traditional internal fixation techniques, external fixation offers several advantages, including the ability to correct conditions beyond the scope of conventional surgeries, shortened treatment durations, reduced risks of intraoperative and postoperative infections [4], and higher success rates of correction procedures. Since the introduction of external fixation surgeries for limb fractures, this technique has gained attention for its favorable treatment outcomes and low rates of postoperative complications [5,6]. To enhance the precision and success rates of external fixation surgeries while mitigating risks such as physician fatigue and exposure to X-rays, researchers worldwide have increasingly focused on utilizing FRR [7–14] to assist in surgical procedures. The FRR are one of the important applications of external fixation technology.

* Correspondence to: Beijing haidian district west TuCheng Road 10, Beijing university of posts and telecommunications, China.

E-mail address: chengyu1159@163.com (C. Lv).

<https://doi.org/10.1016/j.birob.2025.100216>

2667-3797/© 2025 The Author(s). Published by Elsevier B.V. on behalf of Shandong University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Among various configurations of fracture reduction robots, the parallel six-axis fracture reduction robot based on the Stewart platform principle is a commonly encountered design. Its structure typically comprises upper and lower fixed rings (static and moving platforms), six telescopic rods, several connecting hinges, steel pins, and a fixation frame. The affected limb is connected to the upper and lower fixed rings via steel pins and the fixation frame. In comparison, the lengths of the six telescopic rods are adjusted to control the relative positions of the upper and lower fixed rings, allowing the trajectory of the dynamic platform to closely follow the prescribed path for effective limb fracture reduction and deformity correction.

After the physician measures the deformity parameters on the X-ray film, the FRR is positioned on the patient's affected limb. It is essential to determine the initial pose of the moving platform relative to the static platform based on the lengths of the six connecting rods. Accurate determination of the initial posture is critical for ensuring the effectiveness of the external fixator's orthopedic prescription and maintaining medical safety, thereby significantly influencing the success rate of external fixation surgery. With the advancement of intelligent fracture reduction robots, a fast and precise pose calculation method is crucial for the robot's overall motion planning. This paper proposes a feasible solution to enhance the accuracy of initial pose

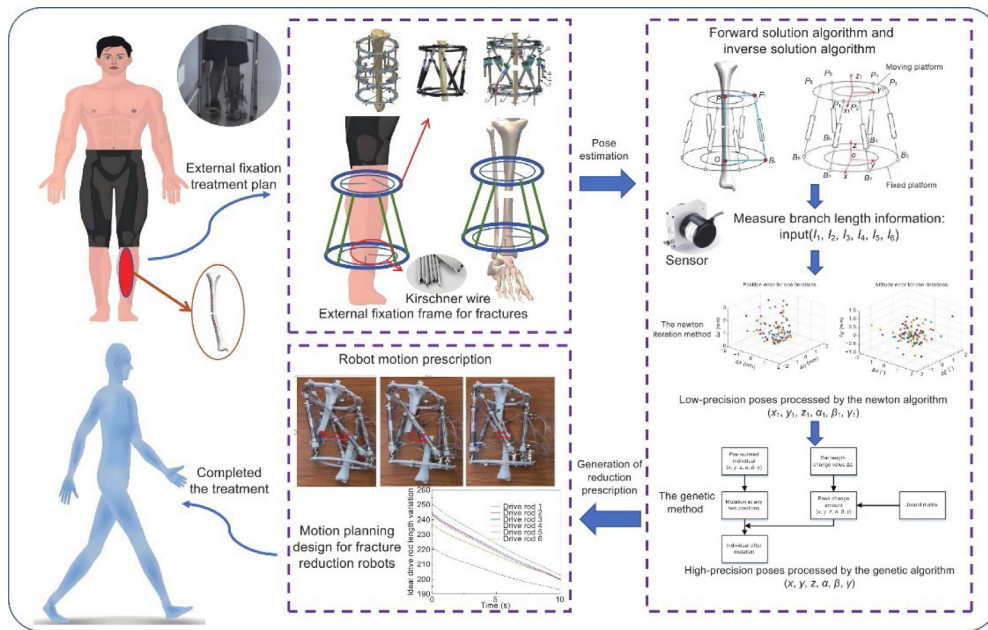


Fig. 1. The correction process for the external fixation fracture reduction robot.

estimation while striving to reduce computation time as much as possible. The illustration of the correction process for the external fixation fracture reduction robot is shown below in Fig. 1.

Currently, parallel Fracture Reduction Robot utilize a variety of forward solution algorithms [15,16], including numerical methods, analytical methods, additional sensor algorithms, and intelligent optimization algorithms. The primary challenge in solving the forward kinematics of a 6-DOF motion platform lies in addressing a set of strongly coupled nonlinear equations, which complicates the solution process. Consequently, no single algorithm has been universally recognized for guaranteeing high success rates, accuracy, and efficiency. Numerical methods [17–19] offer high computational efficiency but involve extensive calculations. Additionally, their results are highly dependent on the selection of initial values; inappropriate initial values can lead to increased iteration counts or even cause the solution to diverge. Analytical methods [20] primarily reduce the number of unknowns in the constraint equations through various elimination techniques, ultimately deriving a higher-order equation with only one unknown. Although analytical methods can obtain all forward solutions of the mechanism, their derivation process is complex, and determining the correct solution in practice can be problematic. Moreover, analytical methods are only applicable to specific parallel structures. Mahmoodi et al. [21] proposed an additional sensor algorithm that directly solves the forward kinematics of the platform by installing an angle sensor. However, this approach is highly dependent on the accuracy of the assembly and the sensors. Intelligent optimization algorithms, such as neural networks, genetic algorithms, and ant colony algorithms, present alternative solutions. Nevertheless, most single intelligent optimization algorithms suffer from long computation times and low computational efficiency. Specifically, the single genetic algorithm [22] demonstrates good global performance by mitigating the dependency on initial values inherent in traditional numerical algorithms, but it requires excessive computation time and a large number of iterations.

In clinical practice, manual external fixation techniques for treating fractures often lack sufficient precision. The introduction of intelligent fracture reduction robots has significantly improved the accuracy of bone alignment in affected limbs. While different robots exhibit varying degrees of precision, most high-precision

fracture reduction robot systems achieve positional errors within 0.1–1 mm and orientation errors within 0.1–1° [23]. To further enhance the therapeutic efficacy and reliability of these robots, developing even higher-precision robots is an inevitable trend. Achieving such precision necessarily requires more accurate algorithms; therefore, it is essential to identify algorithms that can accommodate both current and future demands for increased precision.

Considering the motion characteristics of the FRR, this paper introduces a hybrid algorithm that combines the Newton method with a genetic algorithm. Initially, the high efficiency of the Newton method is leveraged during the early stages of iteration to rapidly approach a solution. Based on the accuracy requirements, the optimal number of Newton method iterations is determined. The resulting pose from the Newton method is then incorporated into the initial population of the genetic algorithm. By utilizing the global convergence properties of the genetic algorithm, the hybrid approach effectively resolves the forward kinematics problem with enhanced precision.

The remainder of the paper is organized as follows. The second section outlines the kinematic modeling method for the parallel fracture reduction robot, including the establishment of the coordinate system and the calculation of inverse kinematics. The third section focuses on the computational solving of the Newton-Genetic forward kinematics algorithm. In the fourth section, a comparative analysis of the algorithms is presented, accompanied by a detailed examination of the experimental results.

2. Kinematics analysis of FRR

2.1. Establishment of FRR structure and coordinate system

The FRR is based on the principles of the Stewart platform, resulting in a structure that closely mirrors that of a traditional Stewart platform [24]. The overall robot system is illustrated in Fig. 1. The main components of the FRR include upper and lower fixed rings, six telescopic rods, and twelve Hooke hinges. Each telescopic rod is equipped with a quick-inserting airport, facilitating rapid assembly and adjustment. The robot's pose is determined by coordinating the lengths of the individual actuators. The mechanical configuration of the FRR is shown in Fig. 2.

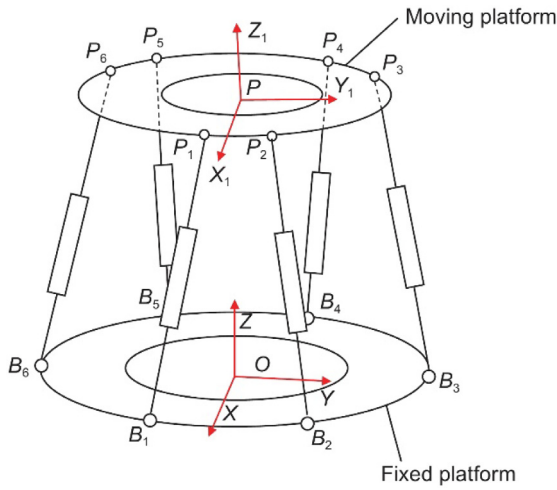


Fig. 2. Structure diagram of bone external fixator.

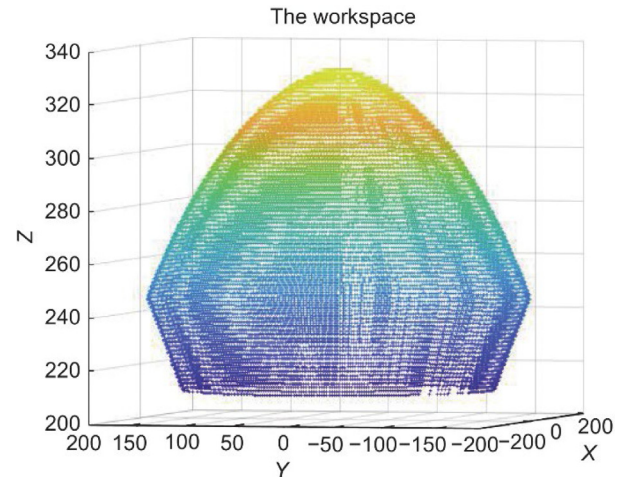


Fig. 3. Three-dimensional map of working space of the FRR.

The hinge points on the static platform for the telescopic rods are designated as B_1 through B_6 , while the hinge points on the moving platform are designated as P_1 through P_6 .

A coordinate system, $O-XYZ$, is established on the static platform with the origin O fixed at the center. The X -axis points towards the midpoint between B_1 and B_2 , the Z -axis is perpendicular to the static platform, and the Y -axis is defined according to the right-hand rule. The moving platform's coordinate system, $P-X_1Y_1Z_1$, is translated from the $O-XYZ$ system along the Z -axis, with point P located at the center of the moving platform. This coordinate transformation ensures precise alignment and accurate pose calculation, which are critical for the robot's motion control and the success of fracture reduction procedures.

According to the rectangular coordinate system established above, the relative pose of the moving platform coordinate system relative to the static platform coordinate system can be represented by a set of vectors. According to the Euler angle description method, the moving platform coordinate system can be realized by three rotation movements around the coordinate axis. In this paper, the matrix transformation of $Z-Y-X$ is used to represent, and the overall rotation matrix R is:

$$R = R_Z(\alpha) \cdot R_Y(\beta) \cdot R_X(\gamma)$$

$$= \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ c\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix} \quad (1)$$

where α , β and γ are Euler angles rotating around Z , Y and X axes, respectively. The position vector of the moving platform in the static platform coordinate system is, and the homogeneous coordinate transformation matrix is introduced to represent the pose of the moving platform relative to the static platform. Then the coordinate transformation matrix T is:

$$T = \begin{bmatrix} R & P \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma & x \\ c\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma & y \\ -s\beta & c\beta s\gamma & c\beta c\gamma & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where the symbol c represents the function \cos ; the symbol s represents the function \sin .

2.2. The working space of the FRR

The workspace of a robot is a key indicator of its motion capabilities and overall performance [25–29]. In the context of

fracture reduction, understanding the workspace parameters of the FRR is crucial for physicians to accurately select the most suitable device for each patient. The workspace of the FRR depends on several factors, including the stroke length of the telescopic rods, the rotation range of the hinges, potential interference between the rods, and the overall structural dimensions of the FRR. These parameters collectively determine the FRR's effectiveness in performing precise corrective maneuvers during surgery.

2.2.1. Length constraint of telescopic rod

During the movement of the telescopic rod, its working length must be limited to a certain set range. Assuming that the maximum length and the minimum length of the telescopic rod are L_{\max} and L_{\min} , respectively, the length constraint of the telescopic rod is:

$$L_{\min} \leq L_i \leq L_{\max}, \quad i = 1, 2, 3, 4, 5, 6 \quad (3)$$

2.2.2. Hinge rotation range constraint

The rotation range of the Hooke hinge of the FRR must always satisfy the following constraint:

$$\sigma_i \leq \sigma_{\max}, \quad i = 1, 2, 3, 4, 5, 6 \quad (4)$$

2.2.3. Interference constraint of rod

When the FRR operates, the shortest distance d_{ij} between the centerlines of any two telescopic rods must be greater than or equal to the diameter d of each rod, ensuring no interference occurs:

$$d_{ij} \geq d \quad (5)$$

where i, j are the numbers of any two telescopic rods respectively.

The working space of the FRR, obtained through the scanning method, is shown in Fig. 3.

2.3. Inverse kinematic model

The inverse kinematics of the parallel mechanism is comparatively simpler than the forward kinematics. In the FRR, the lengths of the telescopic rods define the pose of the moving platform. During the correction process, inverse kinematics calculations are performed to determine the precise lengths of the six

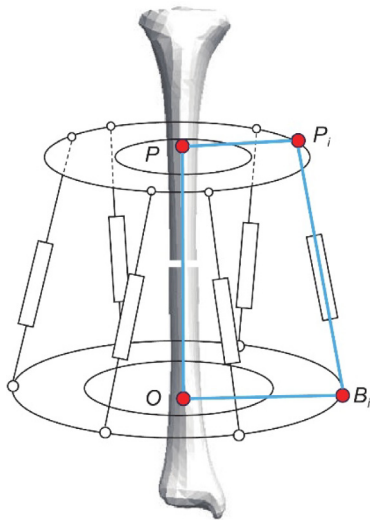


Fig. 4. Schematic diagram of kinematics solution of the FRR.

telescopic rods needed to achieve a desired position. The mechanism for solving the inverse kinematics of the FRR is depicted in Fig. 4.

The vector $P_i = [P_{ix}, P_{iy}, P_{iz}]^T$ of the hinge point on the moving platform in the moving coordinate system is expressed as P_{oi} in the static coordinate system:

$$P_{oi} = RP_i \quad (6)$$

where R is the rotation matrix that converts coordinates from the moving platform to the static platform. The i vector of the telescopic rod can be expressed as:

$$L_i = P + P_{oi} - B_i \quad (7)$$

where P represents the translation vector of the moving platform's center relative to the static platform, and B_i is the position vector of the i th hinge point on the static platform in the static coordinate system.

The length of the telescopic rod i can be based on the modulus length of the vector L_i :

$$L_i = \sqrt{L_i^T L_i} \quad (8)$$

This length is calculated as the Euclidean norm of vector L_i .

3. Forward kinematics solution method based on Newton-Genetic hybrid algorithm

The forward kinematics of the parallel degree-of-freedom platform is relatively complex due to the strong coupling of its motion [27,28]. The traditional Newton algorithm offers high accuracy and fast computation speed, but it is heavily dependent on the choice of initial values—large deviations in the initial value can lead to iterative divergence. While the traditional genetic algorithm addresses the issue of dependence on initial values, it requires many iterations to achieve a high-precision solution, resulting in low computational efficiency. Considering the practical use of the FRR, real-time performance is not required for its forward kinematics solution. Therefore, it is essential to strike a balance between success rate, solution accuracy, and computational efficiency. Given the strengths and weaknesses of both the Newton and genetic algorithms, this paper proposes a hybrid approach that combines these two algorithms. The flowchart of the Newton-Genetic algorithm is presented in Fig. 5. In the initial phase of the algorithm, the core step is to use

the Newton–Raphson method to process the preliminary results. Through the Newton–Raphson operation, the processed data will progressively converge towards the true value. Thanks to the fast convergence speed of the Newton method, a small number of iterations can effectively leverage its computational efficiency to quickly optimize the initial values. The processed initial values will then serve as the basis for generating the initial population of the genetic algorithm.

After generating the initial population, the genetic algorithm will perform weighting and offsetting operations on the population to further optimize its quality and diversity, taking into account the actual operational conditions. Specifically, these weighting and offsetting operations will adjust the initial population based on computational requirements and environmental constraints, generating a more advantageous solution space. Ultimately, through multiple generations of evolution in the genetic algorithm, the individuals in the population will continuously evolve, leading to a higher-precision computational result, ensuring both the quality and reliability of the solution.

This process fully integrates the advantages of both the Newton method and the genetic algorithm, ensuring rapid convergence of the initial solution while further improving the precision and robustness of the solution through the global search capability of the genetic algorithm.

3.1. The Newton iteration method part of the hybrid algorithm

The Newton method is widely used for solving nonlinear equations. The Newton iteration method features second-order convergence, providing a fast convergence speed, but it has stringent requirements for initial values and a limited convergence range [29]. The iteration formula is:

$$T_{k+1} = T_k - (F'(T_k))^{-1}F(T_k) \quad (9)$$

where T_k and T_{k+1} represent the solution vectors at the current and next iterations, respectively. $F(T_k)$ is the value of the nonlinear function at the current estimate T_k , representing the residual vector, and $F'(T_k)$ is the Jacobian matrix at T_k , which contains the partial derivatives of the system of nonlinear equations.

The Newton iteration method is a classical approach to solving the forward kinematics of parallel mechanisms. In this method, the exact solution is obtained through iterative approximation. First, an initial assumed pose of the moving platform is provided. The inverse kinematics solution is then used to determine the corresponding assumed rod lengths. The difference between the actual rod lengths of the six telescopic rods and the assumed values is used to form the rod length error vector. The Jacobian matrix J is employed to map the rod length error vector to the pose error vector, allowing the correction of the pose based on the known rod length information. The corrected pose is then fed back into the process, iterating until the pose error falls below a predetermined threshold.

In the forward solution of the FRR, the length of the six telescopic rods of the FRR is $L = (l_1, l_2, l_3, l_4, l_5, l_6)$, and the initial assumed position is $X_0 = (x_0, y_0, z_0, \alpha_0, \beta_0, \gamma_0)$. According to the inverse kinematics solution, the rod length vector L_1 under the assumed position is obtained.

The difference vector of rod length is:

$$\Delta L = L - L_1 \quad (10)$$

The kinematic Jacobian matrix J is expressed as follows.

$$J = \begin{bmatrix} \frac{\partial l_1}{\partial x} & \frac{\partial l_1}{\partial y} & \frac{\partial l_1}{\partial z} & \frac{\partial l_1}{\partial \alpha} & \frac{\partial l_1}{\partial \beta} & \frac{\partial l_1}{\partial \gamma} \\ \frac{\partial l_2}{\partial x} & \frac{\partial l_2}{\partial y} & \frac{\partial l_2}{\partial z} & \frac{\partial l_2}{\partial \alpha} & \frac{\partial l_2}{\partial \beta} & \frac{\partial l_2}{\partial \gamma} \\ \frac{\partial l_3}{\partial x} & \frac{\partial l_3}{\partial y} & \frac{\partial l_3}{\partial z} & \frac{\partial l_3}{\partial \alpha} & \frac{\partial l_3}{\partial \beta} & \frac{\partial l_3}{\partial \gamma} \\ \frac{\partial l_4}{\partial x} & \frac{\partial l_4}{\partial y} & \frac{\partial l_4}{\partial z} & \frac{\partial l_4}{\partial \alpha} & \frac{\partial l_4}{\partial \beta} & \frac{\partial l_4}{\partial \gamma} \\ \frac{\partial l_5}{\partial x} & \frac{\partial l_5}{\partial y} & \frac{\partial l_5}{\partial z} & \frac{\partial l_5}{\partial \alpha} & \frac{\partial l_5}{\partial \beta} & \frac{\partial l_5}{\partial \gamma} \\ \frac{\partial l_6}{\partial x} & \frac{\partial l_6}{\partial y} & \frac{\partial l_6}{\partial z} & \frac{\partial l_6}{\partial \alpha} & \frac{\partial l_6}{\partial \beta} & \frac{\partial l_6}{\partial \gamma} \end{bmatrix} \quad (11)$$

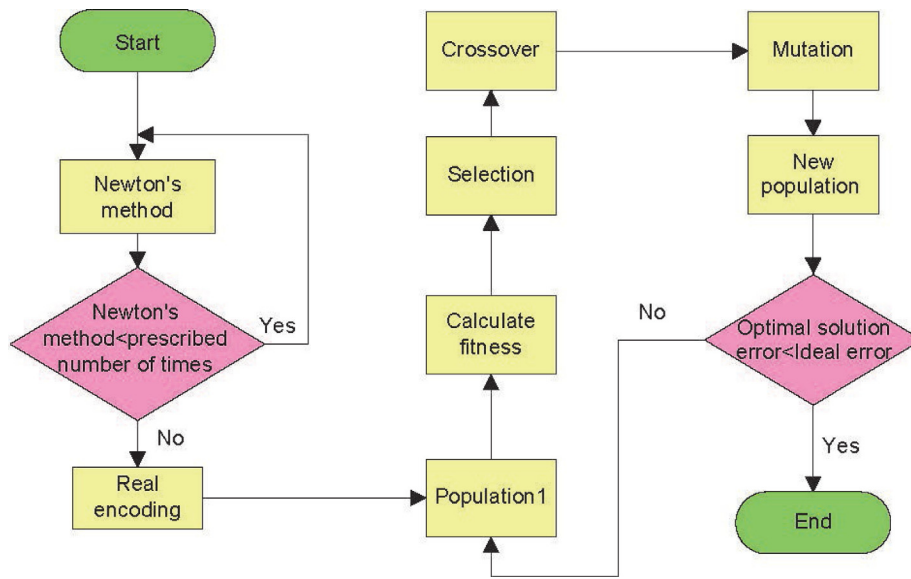


Fig. 5. Mixing algorithm flow chart.

where each element, such as $\frac{\partial l_i}{\partial x}$, represents the partial derivative of the i th telescopic rod length with respect to a specific pose parameter, including x, y, z (translation) and α, β, γ (rotation). These partial derivatives indicate how sensitive the rod length is to changes in the position or orientation of the moving platform.

The relationship between the telescopic rod error vector ΔL and the pose error vector ΔX is established with the help of the Jacobian matrix:

$$\Delta L^T = J \Delta X^T \tag{12}$$

This is the core step of the Newton iteration method. Repeat the above operation and iterate to approximate the real value.

In this paper, the advantages of the fast iteration speed of the Newton method are utilized to obtain pose estimates with a specified level of accuracy. However, due to the local convergence characteristics of the Newton method [30], if the initial deviation is large, the iterative process can cause the estimated values to diverge further from the actual solution, especially as the number of iterations increases. This divergence can result in the subsequent genetic algorithm needing to perform extensive operations to correct the deviation. To prevent this, the number of Newton iterations is limited to a predefined maximum.

In practical applications, the accuracy requirement for the FRR is generally at the millimeter level. With the advancement of the FRR, achieving higher precision in the forward kinematics solution has also become an essential requirement. Therefore, the target accuracy for the forward solution is set to 10^{-5} mm. The Newton iteration process is terminated once the specified number of iterations is reached, and the output is then used as the initial population for the genetic algorithm.

To determine the optimal number of Newton iterations, 100 sample poses were randomly generated within the defined workspace of the FRR. The final iteration accuracy was recorded for different iteration counts of 1, 2, and 3. The iteration accuracy diagrams for these specified numbers of iterations are shown in Figs. 6–8.

In Figs. 6–8, the position error and attitude error decrease with the increase of the number of iterations. The table of position error and attitude error with the increase of the number of iterations is shown in Table 1.

After three iterations of the Newton method, the maximum position error is 0.036 mm, and the maximum orientation error is 0.008° . The Newton iteration method demonstrates high initial

Table 1
Position error and attitude error increase with the increase of iteration times.

Iteration times	Position error (mm)	Attitude error ($^\circ$)
1	2.3	2.2
2	0.01	0.09
3	0.039	0.008

efficiency and the fastest convergence speed. In the Newton-Genetic hybrid algorithm, the number of Newton iterations is set to 3. Additionally, the Jacobian matrix obtained during the third iteration is recorded and used in the subsequent genetic algorithm.

3.2. The genetic algorithm part of hybrid algorithm

The genetic algorithm is a stochastic global optimization method [31]. It generates successive populations by selecting, crossing, and mutating individuals from an initial random population, simulating the process of biological evolution to obtain high-quality solutions.

In this paper, finding the forward kinematics solution of the FRR is essentially a constrained optimization problem. The goal is to find the optimal numerical solution within a specified range of conditions. For constrained optimization problems, the search efficiency of real-valued encoding is higher compared to binary encoding when using genetic algorithms [32–35].

The initial population for the genetic algorithm is generated based on the results from the Newton method. The Newton iteration method provides a pose with a certain level of accuracy, and from this pose, 100 initial poses are generated randomly with position and orientation ranges set to 0.05 and 0.001 , respectively. This pose set is used as the initial population for the genetic algorithm.

After the population is generated, it is necessary to determine the fitness function to evaluate individuals within the population. The selection of the fitness function significantly affects the efficiency of the algorithm. The fitness function is defined as:

$$F(p) = 1/(f(p) + \lambda) \tag{13}$$

where $F(p)$ represents the fitness value for an individual p , $f(p)$ is the objective function value for the individual, and λ is a small

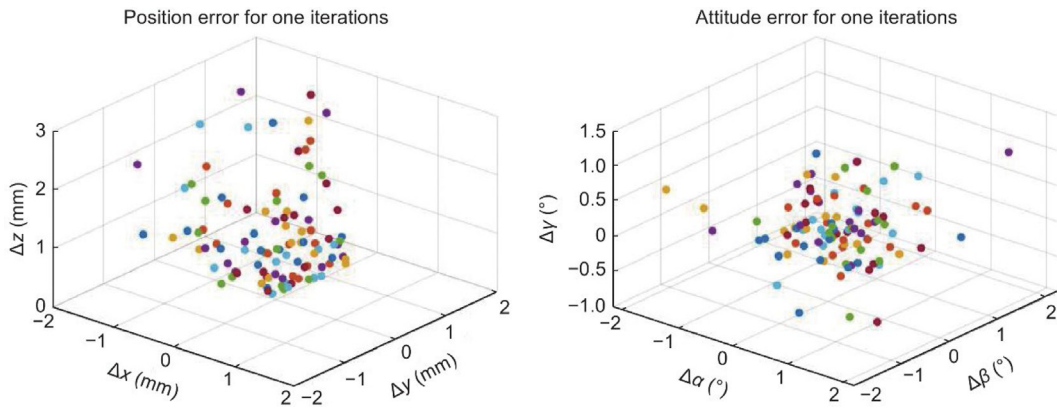


Fig. 6. The position error and attitude error after one iteration of the Newton method.

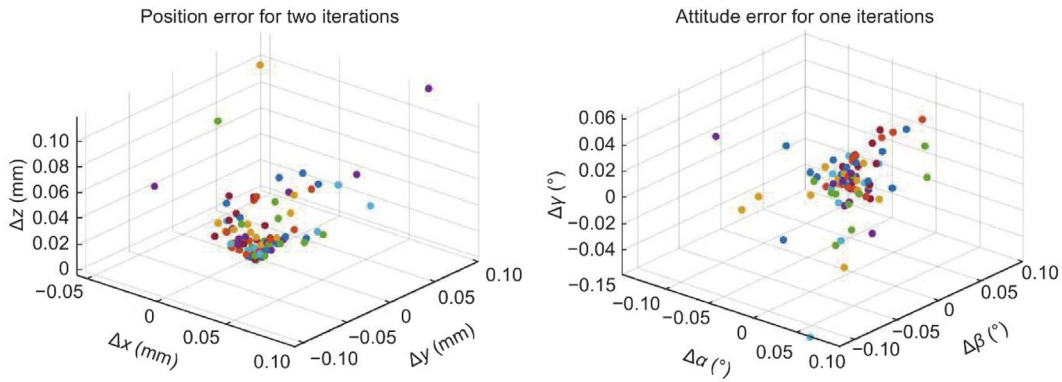


Fig. 7. The position error and attitude error after two iterations of the Newton method.

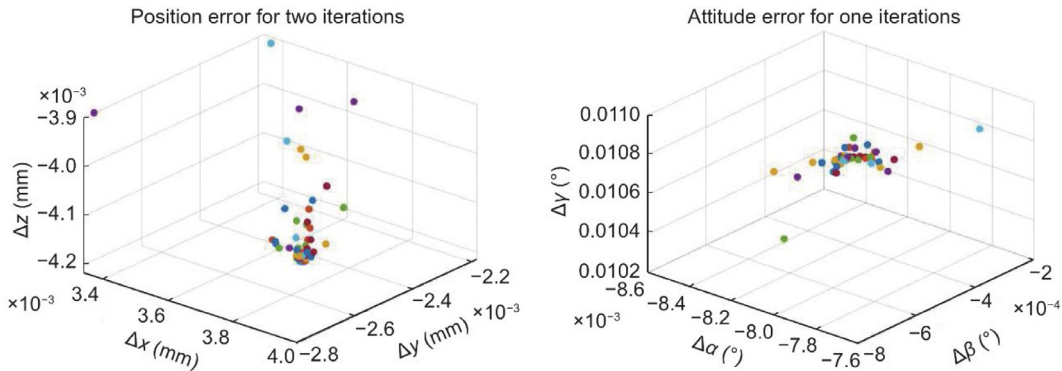


Fig. 8. The position error and attitude error after three iterations of the Newton method.

positive constant to ensure that the fitness function does not become zero.

In selecting a new generation of individuals, this paper abandons the traditional roulette selection algorithm and adopts a random traversal sampling method. This improves the probability of selecting individuals from the low-fitness population, ensuring that these individuals still have a chance to be selected, which promotes genetic diversity. This approach prevents individuals with high fitness values from monopolizing the selection, thereby maintaining a diverse gene pool in the new population.

The crossover operation is a core process that drives the generation of optimal solutions. The parent individuals, P_1 and P_2 , produce offspring through crossover, which enables the offspring to adapt better to the environment. To ensure a rich variety of offspring, a random coefficient α is introduced, ranging from 0 to

1. The offspring generation formula is:

$$P' = \alpha P_1 + (1 - \alpha)P_2 \tag{14}$$

where P' is the offspring individual, P_1 and P_2 are the parent individuals, and α determines the contribution of each parent to the offspring.

Mutation operations determine the local search capability of the genetic algorithm and also play a key role in maintaining population diversity. To speed up the convergence of the genetic algorithm and enhance the efficiency of the overall hybrid algorithm [36], specific mutation parameters are determined by incorporating the Jacobian matrix from the third iteration of the Newton method.

The position and orientation errors can be determined using the difference in rod lengths and the Jacobian matrix. Since the

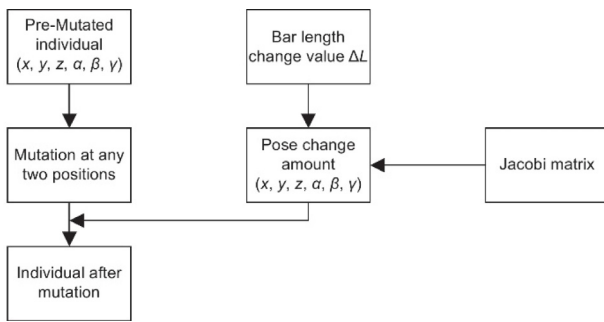


Fig. 9. Variation operation flow chart.

rod length differences are known, these errors can be estimated using the Jacobian matrix obtained from the third iteration of the Newton method. The specific mutation operation process is illustrated in Fig. 9. This flowchart illustrates the mutation process in a genetic algorithm that incorporates the Jacobian matrix for enhanced efficiency and precision. The process begins with a selected individual from the population (pre-mutated individual), represented by its position and orientation parameters $(x, y, z, \alpha, \beta, \gamma)$. Two parameters are randomly selected for mutation, introducing new candidate solutions and increasing population diversity. The mutation operation results in changes to the rod lengths, quantified as ΔL , which serves as a key metric for evaluating the mutation's effect. Subsequently, the Jacobian matrix derived from the third iteration of the Newton method is used to map the rod length changes to corresponding pose adjustments (including position and orientation updates). This step guides the mutated individual closer to the optimal solution. Finally, the individual is refined by applying the pose adjustments calculated through the Jacobian matrix, producing a new individual. This process effectively combines randomness and directional guidance, maintaining population diversity while significantly enhancing local search capability and global convergence speed, ultimately achieving higher computational precision and efficiency.

In the genetic algorithm section, selecting appropriate algorithm parameters is crucial for achieving a dynamic balance between computational resources and solution accuracy. The key parameters include population size, number of generations, crossover rate, and mutation rate. In this study, since the initial solution of the genetic algorithm has already been optimized through the preliminary numerical processing of the Newton iteration method, significant accuracy improvements have been achieved at the outset. Therefore, the population size and number of generations can be appropriately reduced to lower computational costs. However, if higher solution accuracy is required, increasing the population size and number of generations will enhance the algorithm's global search capability.

The proper setting of the mutation rate is essential for the genetic algorithm to escape local optima. Particularly in cases where the initial results from the Newton iteration method are suboptimal, the mutation rate becomes a critical parameter for ensuring the robustness of the algorithm. A higher mutation rate can significantly improve the global search ability of the genetic algorithm, allowing it to explore the solution space more comprehensively and mitigating the limitation of the Newton method's tendency to converge to local optima.

From the perspective of clinical needs, ultra-high precision solutions currently do not provide significant advantages for the operational performance of actual robots. Therefore, in the configuration of global algorithm parameters, highly time-consuming

parameters can be adjusted appropriately, such as reducing population size or limiting the number of generations. This adjustment effectively accelerates the solving process while meeting the combined requirements of clinical solution efficiency and accuracy.

4. Simulation experiment

4.1. Comparative experiments of different types of algorithms

To provide a more comprehensive explanation of the algorithm's application in clinical computational scenarios, a simulation experiment was designed to replicate critical computational tasks that may arise during the actual clinical use of a fracture reduction robot. The experimental scenario assumes that when the fracture reduction robot is first applied to the patient's affected limb, the robot's pose information cannot be directly obtained. However, accurately determining the robot's pose is crucial for subsequent fracture reduction procedures, surgical precision, and patient safety.

In this experiment, the lengths of the robot's six motion chains were acquired as the foundational input data for solving the robot's pose using different algorithms. Three distinct methods were used for pose calculation: the analytical method, the additional sensor-based algorithm, and the Newton-Genetic hybrid algorithm. These methods represent different computational approaches: the analytical method relies on known geometric relationships for direct computation, the additional sensor-based algorithm utilizes external sensors to obtain pose information, while the Newton-Genetic hybrid algorithm combines the fast convergence of the Newton method with the robust global search capability of the genetic algorithm to enhance computational accuracy and robustness.

Subsequently, the solution sets obtained from the three methods were analyzed and compared to evaluate their performance in terms of computational accuracy, convergence speed, and resource efficiency. This experiment provides a comprehensive understanding of the characteristics and limitations of each method, offering valuable references for assessing the clinical applicability of fracture reduction robots.

4.1.1. Analytical method to solve experimental examples

The algebraic elimination method [37] is employed to solve for the initial values of different rod lengths. In this approach, 6 of the 9 constraint equations are replaced by variables and converted into linear equations. The Groebner basis algorithm, implemented in a computer algebra system, is then used to construct the Sylvester resultant to obtain the one-dimensional polynomial equations for the forward solution of the FRR's position. Finally, each specific solution is determined by solving these equations.

The algebraic elimination method is capable of outputting all possible solutions for the mechanism's configuration, which may include non-real solutions. After an initial data filtering process, only the real solutions are retained. One solution from the 40 groups of experiments is presented for demonstration, with specific results shown in Table 2 (values are rounded to two decimal places).

Based on the specific conditions of the application example, further filtering and refining of the data yields the final solution. The speed and accuracy of the algorithm are summarized in Table 3.

Table 2
The real number solution of the analytic method.

No.	x	y	z	α	β	γ
1	-16.82	25.97	17.51	55.83	-24.17	12.94
2	-12.04	-28.47	17.31	54.48	-28.35	-14.59
3	0.00	0.00	67.09	0.00	0.00	0.00
4	30.51	-5.24	17.09	8.11	-58.92	-4.60

Table 3
Analytical algorithm performance parameters.

Name	Running speed (s)	Position error (mm)	Attitude error (°)
Algebraic elimination	7.3	10^{-30}	10^{-30}

Table 4
Operating parameters of additional sensor algorithm.

Name	Running speed (s)	Position error (mm)	Attitude error (°)
ASA	0.0125	5.2×10^{-2}	δ

4.1.2. Application of additional sensor algorithm to solve experimental examples

The ASA utilizes a set of sensors installed at various locations on the FRR to simplify subsequent calculations using measured data. In this study, three attitude sensors are positioned on the moving platform, and the actuators of the FRR to capture the robot's motion information.

In this paper, three attitude sensors are arranged on the moving platform and actuator of the FRR to obtain the motion information of the robot.

The FRR model is first established and imported into ADAMS, where a virtual simulation environment is created by adding the necessary constraints. The robot is driven by setting different branch lengths, with a simulation duration of 40 s, divided into 40 time steps. During the simulation, the attitude sensors directly collect the motion data, which is then recorded and imported into MATLAB for further pose calculation of the robot based on the collected information. The speed and accuracy of the ASA are summarized in Table 4.

Among them, δ is the system error value of the attitude sensor (the system error of the attitude parameter measured by the attitude sensor will generally be less than 0.1°).

4.1.3. Application of Newton-Genetic algorithm to solve experimental examples

The Newton-Genetic algorithm is applied to solve the forward kinematics of the FRR. Initially, an assumed pose is provided, and the final pose is determined through iterative calculations. The speed and accuracy of the algorithm are summarized in Table 5.

4.1.4. Comparative analysis of different types of algorithms

Among the three algorithms used to solve the forward kinematics of the FRR, each demonstrates unique advantages and limitations. The analytical method offers high precision in solutions; however, it faces challenges in determining a specific solution from multiple possible outcomes. In practical applications involving the FRR, the desired forward solution should ideally be obtained directly.

The additional sensor algorithm simplifies the problem by using sensors to measure certain parameters, which significantly reduces the complexity of the calculations and improves efficiency. However, this approach is heavily dependent on the accuracy of the sensors, and the assembly requirements are stringent. Low-precision assembly can result in failure to obtain valid solutions.

Table 5
Operating parameters of Newton-Genetic algorithm.

Name	Running speed (s)	Position error (mm)	Attitude error (°)
Newton-GA	4.25	6.16×10^{-4}	6.53×10^{-4}

Compared to these methods, the Newton-Genetic algorithm does not require determining specific solutions nor the installation of additional sensors. For solving the forward kinematics of the FRR, this hybrid algorithm reduces the complexity of the problem and provides a set of accurate poses. For the FRR's requirements in medical applications, the Newton-Genetic algorithm delivers high-precision poses while maintaining a lower difficulty in sensor assembly, ultimately enhancing the effectiveness and safety of medical correction procedures.

4.2. Comparison test of Newton-Genetic algorithm and common numerical method

The motion trajectory of the moving platform within the workspace is planned, and a total of 40 poses at different time points are selected during the motion process. The corresponding rod lengths are determined using inverse kinematics. The Newton-Genetic algorithm, genetic algorithm, and particle swarm optimization algorithm are then used to calculate the forward kinematics for these 40 sets of rod lengths. The poses obtained through inverse kinematics at different times serve as the reference values for evaluating the accuracy of the forward kinematics calculations. Several sets of iterative error plots comparing the position and orientation results of the different algorithms are presented in Fig. 10. To provide a clearer illustration of the variation in iteration error of the Newton-Genetic Algorithm following the Newton method, a smaller scale coordinate system is utilized in Figs. 10(c) and 10(d) to display the trend of the changes.

The Newton-Genetic algorithm demonstrates a faster iteration speed compared to the simple genetic algorithm and the particle swarm optimization algorithm. When the iterative error curve stabilizes, the Newton-Genetic algorithm achieves a noticeably smaller error compared to the genetic algorithm, with higher accuracy. After 20 iterations, the Newton-Genetic algorithm achieves an accuracy error of less than 10^{-4} mm.

Due to the poor convergence performance of the particle swarm optimization algorithm during the iteration process, only the genetic algorithm and the Newton-Genetic algorithm are retained for further analysis and comparison in the subsequent numerical evaluation. The forward kinematics calculations were performed on 40 sets of rod length data, and the results obtained from the two algorithms were statistically analyzed. The analysis results are presented in Fig. 11.

After 250 iterations, the average position and orientation errors of the genetic algorithm are significantly higher than those of the Newton-Genetic algorithm. The genetic algorithm's average position error is 0.5630 mm, and the average orientation error is 0.1403° . In contrast, the Newton-Genetic algorithm achieves smaller errors, fully meeting the accuracy requirements for adjusting the medical FRR.

These results demonstrate that the Newton-Genetic algorithm has faster convergence and better accuracy compared to the traditional genetic algorithm, thus verifying its practicality for the intended application.

5. Conclusion

In the clinical application of a fracture reduction robot for correcting affected limbs, achieving a high-precision forward kinematics solution is crucial for ensuring the success of subsequent

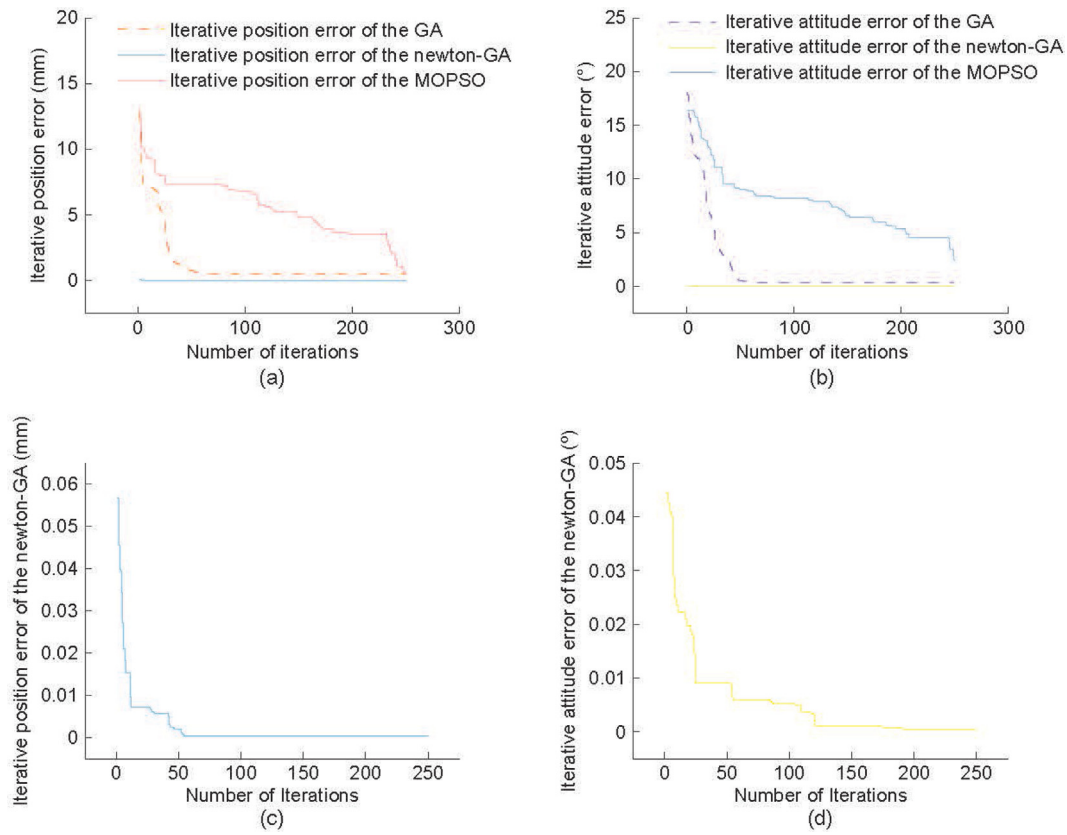


Fig. 10. The iterative error comparison diagram of the three algorithms: (a) is the iteration diagram of position error; (b) is the iterative diagram of attitude error; (c) is the iteration diagram of position error of the Newton-GA; (d) is the iteration diagram of attitude error of the Newton-GA.

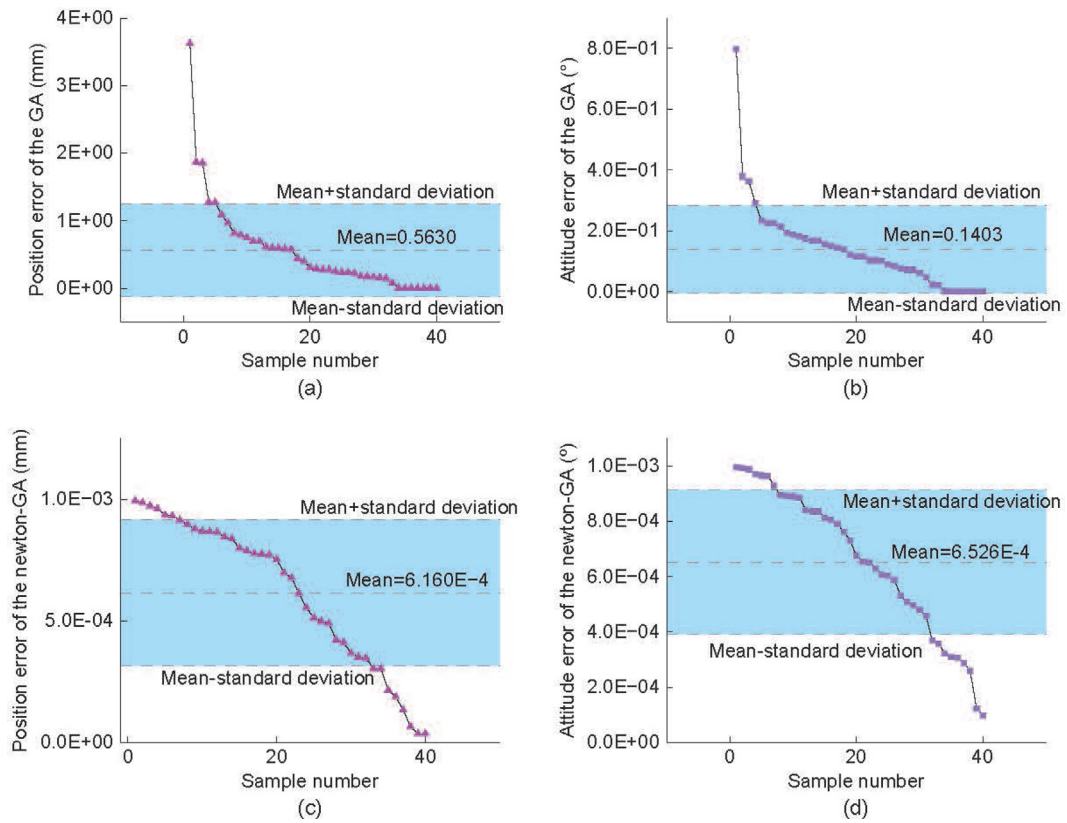


Fig. 11. Position and attitude error statistics obtained by the two algorithm: (a) is the final position error graph of GA algorithm; (b) is the final attitude error graph of GA algorithm; (c) is the final position error graph of Newton-GA algorithm; (d) is the final attitude error graph of Newton-GA algorithm.

treatment steps and patient safety. The Newton-Genetic algorithm proposed in this study offers superior accuracy and faster computation speed compared to traditional genetic algorithms, significantly improving the success rate of solving forward kinematics for a six-axis fracture reduction robot. This novel approach addresses the challenge of solving complex kinematic problems and contributes to the intelligent development of external fixators [38–44].

The Newton-Genetic iterative algorithm demonstrates clear advantages in clinical computations. Its simple code logic makes it flexible and adaptable, while also requiring fewer computational resources. In contrast to traditional methods such as analytical solutions, sensor-based algorithms, and numerical approaches, the Newton-Genetic algorithm avoids the need for additional sensors, reducing both hardware dependency and operational complexity for clinicians. This leads to a more streamlined workflow and lower operational burden.

Moreover, the algorithm provides more precise results with fewer iterations compared to numerical methods. Given the clinical need for sub-millimeter accuracy and the increasing demand for high-precision medical correction, the Newton-Genetic iterative algorithm strikes an optimal balance between computation speed and accuracy, offering a promising computational solution for six-degree-of-freedom parallel fracture reduction robots.

CRediT authorship contribution statement

Jian Li: Writing – original draft, Methodology, Conceptualization. **Xiangyan Zhang:** Visualization, Formal analysis, Data curation. **Yadong Mo:** Validation, Software, Investigation. **Guang Yang:** Supervision, Resources, Project administration. **Yun Dai:** Writing – review & editing. **Chengyu Lv:** Visualization, Resources, Methodology. **Ying Zhang:** Validation, Supervision, Software. **Shimin Wei:** Writing – review & editing, Validation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research work is supported by the National Natural Science Foundation of China (52005120), the interdisciplinary Team of Intelligent Elderly Care and Rehabilitation in the “Double first-class” Construction of Beijing University of Posts and Telecommunications in 2023 (2023SYLTD04), the BUPT innovation and entrepreneurship support program (2024-YC-T038), the BUPT Excellent Ph.D. Students Foundation (CX2023315).

References

- [1] H.Q. Yang, L. Qu, Ilizarov bone transport technique, *Zhongguo Gu Shang* 35 (10) (2022) 903–907.
- [2] V.P. Giannoudis, E. Ewins, D.M. Taylor, P. Foster, P. Harwood, Clinical and functional outcomes in patients with distal tibial fracture treated by circular external fixation: A retrospective cohort study, *Strat. Trauma Limb Reconstr.* 16 (2) (2021) 86–95.
- [3] M.J. Moses, N. Tejwani, The role of external fixation in the management of upper extremity fractures, *J. Am. Acad. Orthop. Surg.* 31 (2023) 860–870.
- [4] J. Meng, T. Sun, F. Zhang, et al., Deep surgical site infection after ankle fractures treated by open reduction and internal fixation in adults: A retrospective case-control study, *Int. Wound J.* 15 (6) (2018) 971–977.
- [5] G.A. Hosny, Limb lengthening history, evolution, complications, and current concepts, *J. Orthop. Traumatol.* 21 (2020) 3.
- [6] H. Zhang, F. Sun, Y. Li, Application of smart healthcare in comparative analysis of effect of early external fixator and plate internal fixation treatment on postoperative complications and lower limb function recovery of patients with unstable pelvic fracture, *Front. Public Heal.* 10 (2022) 887123.
- [7] J. Lei, G. Song, Six-dimensional constraints and force feedback for robot-assisted teleoperated fracture reduction, *Robot.* 42 (7) (2024) 2328–2344.
- [8] S. Lu, W. Yan, Z. Xu, T. Cheng, A framework of robot skill learning from complex and long-horizon tasks, *IEEE Trans. Autom. Sci. Eng.* 19 (4) (2021) 3628–3638.
- [9] J. Lei, Z. Wang, Force/position tracking control of fracture reduction robot based on nonlinear disturbance observer and neural network, *Int. J. Med. Robot.* (2024).
- [10] Z. Wu, Y. Dai, Y. Zeng, Intelligent robot-assisted fracture reduction system for the treatment of unstable pelvic fractures, *J. Orthop. Surg. Res.* 19 (1) (2024).
- [11] Q. Zha, Z. Xu, H. Yang, G. Zhang, X. Cai, W. Zhang, Y. Liu, X. Shen, Y. Li, Development of a robot-assisted reduction and rehabilitation system for distal radius fractures, *Front. Bioeng. Biotechnol.* 11 (2024) 1342229.
- [12] F. Alruwaili, M.S. Saeedi-Hosseiny, L. Guzman, S. McMillan, I.I. Iordachita, M.H. Abedin-Nasab, A 3-armed 6-DOF parallel robot for femur fracture reduction: Trajectory and force testing, in: 2022 Int. Symp. Med. Robot., ISMR, GA, USA, 2022, pp. 1–6.
- [13] H. Sheng, W. Xu, B. Xu, H. Song, D. Lu, W. Ding, M.H. Mildred, Application of intelligent computer-assisted taylor 3D external fixation in the treatment of tibiofibular fracture: Retrospective case study, *JMIR Med. Inf.* 9 (5) (2021) e21455.
- [14] W. Kou, P. Zhou, J. Lin, S. Kuang, L. Sun, Technologies evolution in robot-assisted fracture reduction systems: a comprehensive review, *Front. Robot. AI* 10 (2023) 1315250.
- [15] Z. He, Y. Song, B. Lian, T. Sun, Kinematic calibration of a 6-DoF parallel manipulator with random and less measurements, *IEEE Trans. Instrum. Meas.* 72 (2023) 1–12, <http://dx.doi.org/10.1109/TIM.2022.3221149>, Art. no. 7500912.
- [16] M. Liu, Q. Gu, B. Yang, Z. Yin, S. Liu, L. Yin, W. Zheng, Kinematics model optimization algorithm for six degrees of freedom parallel platform, *Appl. Sci.* 13 (2023) 3082.
- [17] C.C. Nguyen, Z.-L. Zhou, S.S. Antrazi, C.E. Campbell, Efficient computation of forward kinematics and Jacobian matrix of a Stewart platform-based manipulator, in: IEEE Proc. SOUTHEASTCON '91, Williamsburg, VA, USA, 1991, pp. 869–874.
- [18] K. Kiu, J.M. Fitzgerald, F.L. Lewis, Kinematic analysis of a Stewart platform manipulator, *IEEE Trans. Ind. Electron.* 40 (2) (1998) 282–293.
- [19] Z. Zhou, C. Gosselin, Analysis and design of a novel compact three-degree-of-freedom parallel robot, *J. Mech. Robotics: Trans. ASME* 15 (5) (2023).
- [20] S. Cheng, An analytical method for the forward kinematics analysis of 6-SPS parallel mechanisms, *J. Mech. Eng.* 46 (9) (2010) 75–80, <http://dx.doi.org/10.3901/JME.2010.09.026>.
- [21] A. Mahmoodi, M. Sayadi, M.B. Menhaj, Solution of forward kinematics in Stewart platform using six rotary sensors on joints of three legs, *Adv. Robot.* 28 (1) (2014) 27–37.
- [22] P. Wang, et al., Research of the forward kinematics of parallel robot using genetic algorithm, *Mech. Sci. Technol.* (2005).
- [23] J. Zhao, et al., Evolution and current applications of robot-assisted fracture reduction: A comprehensive review, *Ann. Biomed. Eng.* 48 (1) (2020) 203–224.
- [24] T. Liu, Y. Lu, C.Z. Wang, Study on orthopaedic path planning of Taylor spatial frame, *Int. J. Med. Robot. Comput. Assist. Surg.* 20 (1) (2024) e2606.
- [25] J. Lei, J. Wang, Orientation workspace analysis and parameter optimization of 3-RRPS parallel robot for pelvic fracture reduction, *J. Mech. Robot. Trans. ASME* 15 (5) (2023).
- [26] N. Petr, W. Dominik, Investigation of the Stewart platform workspace using MATLAB-simulink and simecape multibody library, in: 21st Int. Conf. Res. Educ. Mechatronics, REM, Cracow, Poland, 2020, pp. 1–5.
- [27] F. Tavassolian, et al., Forward kinematic analysis of spatial parallel robots using a parallel evolutionary neural networks, *Iran J. Sci. Technol. Trans. Mech. Eng.* 47 (2022) 1079–1092.
- [28] T. Li, M. Gong, K. Hu, L. Zhao, B. Zhao, A novel 3-DOF translational parallel robot and its fuzzy controller design, *J. Intell. Fuzzy Syst.* 41 (2021) 4211–4224.
- [29] H.-B. Q. W. Li-Hang, J. Xue, et al., Forward kinematics of Stewart parallel manipulator based on step-adjusting Newton method, *Guangxue Jingmi Gongcheng/ Opt. Precis. Eng.* 26 (12) (2018) 2982–2990.
- [30] M. Geng, Direct position analysis of parallel mechanism based on quasi-Newton method, *J. Mech. Eng.* 51 (9) (2015).

- [31] J. Tang, S. Wang, T. Bai, S. Lu, J. Xiong, Intelligent ATM replenishment optimization based on hybrid genetic algorithm, in: 24th Int. Conf. Adv. Commun. Technol., ICACT, PyeongChang, Korea, 2022, pp. 469–475.
- [32] P. Ye, J. You, H. Shen, A novel semi-analytical algorithm for forward kinematics of 6-DOF parallel mechanisms, in: M. Okada (Ed.), *Advances in Mechanism and Machine Science*, IFToMM WC 2023, in: *Mech. Mach. Sci.*, vol. 147, Springer, Cham.
- [33] E.P. dos Santos, C.R. Xavier, P. Goldfeld, F. Dickstein, R. Weber dos Santos, Comparing genetic algorithms and Newton-like methods for the solution of the history matching problem, in: G. Allen, J. Nabrzyski, E. Seidel, G.D. van Albada, J. Dongarra, P.M.A. Soot (Eds.), *Computational Science – ICCS 2009*, in: *Lect. Notes Comput. Sci.*, vol. 5544, Springer, Berlin, Heidelberg, 2009, pp. 37–48.
- [34] A.-D. Li, B. Xue, M. Zhang, Multi-objective feature selection using hybridization of a genetic algorithm and direct multisearch for key quality characteristic selection, *Inform. Sci.* 523 (2020) 245–265.
- [35] M. Alshehri, P. Sharma, R. Sharma, O. Alfarraj, Motion-based activities monitoring through biometric sensors using genetic algorithm, *Comput. Mater. Contin.* 66 (3) (2020) 2525–2538.
- [36] S. Harifi, R. Mohamaddoust, Zigzag mutation: a new mutation operator to improve the genetic algorithm, *Multimedia Tools Appl.* 82 (2023) 45411–45432.
- [37] Huang, Forward kinematics analysis of the general 6-6 platform parallel mechanism based on algebraic elimination, *Chin. J. Mech. Eng.* 45 (2009) 56.
- [38] H. Tian, et al., Influence of different position modal parameters on milling chatter stability of orthopedic surgery robots, *Sci. Rep.* 14 (2024) 10581.
- [39] Z. Cao, et al., Robot-assisted internal fixation of calcaneal fractures versus conventional open reduction internal fixation: a systematic review and meta-analysis, *J. Robot. Surg.* 18 (1) (2024) 1–10.
- [40] H. Du, et al., Experimental research based on robot-assisted surgery: Lower limb fracture reduction surgery planning navigation system, *Heal. Sci. Rep.* 7 (4) (2024).
- [41] C. Li, et al., Advances of surgical robotics: image-guided classification and application, *Natl. Sci. Rev.* 9 (2024) 9.
- [42] X. Long, et al., The current status and global trends of clinical trials related to robotic surgery: a bibliometric and visualized study, *J. Robot. Surg.* 18 (1) (2024).
- [43] J. Wang, et al., Editorial for the special issue on design, sensing and control in medical robots, *Biomim. Intell. Robot.* 4 (3) (2024) 100177, <http://dx.doi.org/10.1016/j.birob.2024.100177>.
- [44] L. Chen, et al., Adaptive patient-cooperative compliant control of lower limb rehabilitation robot, *Biomim. Intell. Robot.* 4 (2) (2024) 100155, <http://dx.doi.org/10.1016/j.birob.2024.100155>.