Hindawi
International Journal of Aerospace Engineering
Volume 2023, Article ID 1711773, 12 pages
https://doi.org/10.1155/2023/1711773

*Research Article*

# Satellite Fast Maneuver Control Technology Based on Parallel System

**Youtao Gao**,[1] **Jinghe Guo**,[1] **Zhicheng You**,[2] **Zezheng Dong**,[2] **and Yi Cheng**[2]

[1]*School of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China*
[2]*Shanghai Institute of Satellite Engineering, Shanghai 201109, China*

Correspondence should be addressed to Youtao Gao; ytgao@nuaa.edu.cn

Received 23 July 2023; Revised 14 September 2023; Accepted 19 September 2023; Published 13 October 2023

Academic Editor: Chuang Liu



An on-orbit thrust estimation method of satellite based on parallel system, which can achieve high-efficiency and high-precision thrust estimation, is proposed. A complete satellite maneuvering parallel system framework is constructed. Initially, a real-time artificial model, which is consistent with the actual system, is established. The injection time of maneuvering control is estimated optimally based on the modification of the artificial model with the specific injection time treated as the optimization parameter. The jet time, with the minimum maneuvering error, is obtained. Then, a maneuver strategy is designed and fed back to the actual system. The method based on a parallel system with jet time as the optimal parameter has higher control accuracy than the previous maneuvering control, which only considers the speed increment. Simulation results show that the terminal error of the first maneuver using the parallel system method is less than 100 meters for a maneuvering mission of tens of kilometers.

## 1. Introduction

The number of satellites in space increases rapidly. More spacecrafts maneuver to avoid collisions. In addition, due to perturbations, many spacecrafts perform high-frequency maneuvering control to remain in an ideal orbit. High-precision spacecraft control is important for a spacecraft orbit maneuver task. Therefore, a ground flight control and simulation system should be built to simulate the spacecraft's operational status and verify the rationality and accuracy of the control scheme [1]. The appropriateness of the control strategy depends on the ability of obtaining the real-time status of spacecraft on-orbit.

Thrust is the main state that affects satellite maneuver. In the case of satellite fast maneuver, the value of thrust directly affects the maneuver result. The actual output thrust of jet thruster will decrease due to the change of fuel consumption and gas pressure [2]. Therefore, we need to estimate the thrust parameters. The current thrust parameter estimation method is usually based on empirical formulas derived from ground-based thruster testing combined with telemetry data collected on orbit [3]. A limitation of this method is that it does not establish a direct relationship between thrust and maneuver outcomes, and the accuracy of the empirical formulas is affected by unknown factors in the space environment. In addition, control strategy also can affect the maneuver result. The existing satellite maneuver control strategies primarily only focus on calculating velocity increments [4, 5]. These strategies typically overlook the specific correlation between thrusting time and satellite maneuver error. Most research presume that satellites can instantaneously acquire velocity increments, disregarding the fact that actual jet-based satellite thrusters possess finite thrust and require time to accelerate [6, 7]. However, due to the noninstantaneous nature of satellite maneuvers, the required velocity increments gradually change during the maneuver process. And the dynamic models used for velocity increment calculations often omit higher-order spatial disturbances [8, 9]. These issues ultimately cause that satellites cannot accurately reach the target position.

In this paper, we propose to construct a parallel satellite fast maneuver system to address the above problems. In 2004, the parallel system (Artificial system, Computational experiments, Parallel execution, ACP) method was proposed [10]. ACP provided a new approach to address control issues in spacecraft systems with missing parameter information and inaccurate physical models. For the "virtual" and "soft" aspects of the spacecraft system, the ACP method utilizes quantitative and repeatable computational experiments to "harden" them through artificial system [11]. Through parallel execution, the artificial system conducts simulations of the decision scheme, guiding the control and management strategy of the actual system based on the outcomes [12]. This method has been applied in various fields. Ge et al. proposed an ACP simulation framework for data-driven evolutionary modeling, providing a theoretical basis for parallel simulation technology [13]. Zhou et al. established a mechanism and technical architecture for parallel simulation and deduction, providing methodological support for the implementation of parallel simulation and deduction system [14]. Yuan et al. proposed a flight control simulation parallel system, which was tested through the flight control work of the Chang'e-5 mission [15]. Chen et al. applied the parallel system approach to missile defense system, constructing a parallel system structure for missile defense command and control [16]. Jin et al. developed a disturbance-resistant attitude controller using the ACP method and demonstrated its effectiveness [17]. The research and application of spacecraft simulation and parallel systems have received increasing attention. The European Space Agency (ESA) has developed the SMP2 space simulation platform [18]. NASA's Johnson Space Center in the United States has also adopted the open-source Trick Universal Simulation Environment [19]. The corresponding concept to the parallel system is the Dynamic Data Driven Application System (DDDAS) [20, 21], which has been extensively studied and applied in military command [22, 23], engineering applications [24, 25], and other fields. The parallel system constructed in this paper serves two primary purposes: thrust estimation and the formulation of spacecraft orbit maneuver control strategies, with a particular emphasis on thrust time estimation. Unlike traditional sequential simulation models, the parallel system has the capability to simultaneously operate both artificial and actual systems, enabling real-time telemetry data acquisition from on-orbit spacecraft.

For thrust estimation, an artificial system is first constructed. Then, computational experiments combined with particle swarm optimization (PSO) are used to achieve thrust estimation with the telemetry data from on-orbit spacecraft. Finally, the artificial system achieves dynamic evolution of the model based on thrust estimation results. For maneuver control strategy problem, we utilize the adjusted artificial model to estimate the required thrusting time in order to minimize postmaneuver errors relative to the target position. This strategy can reduce the burden of end-guidance. The parallel system provides an effective solution for satellite maneuver control.

## 2. Parallel System of Satellite Fast Maneuver

As illustrated in Figure 1, the satellite fast maneuver parallel system includes two components: the artificial system and the actual system, which interact with each other. The artificial model primarily consists of three parts: model parameter adjustment, artificial satellite model, and control strategy formulation.

Initially, an artificial simulation model is constructed for the spacecraft fast maneuver system. The J2000 inertial coordinate system is selected as the reference coordinate system. In this coordinate system, the satellite orbital maneuver dynamics model which includes perturbation factors is as follows:

$$\begin{aligned}
\ddot{x} &= -\frac{\mu}{R^3}x\left\{1+\frac{3}{2}J_2\left(\frac{R_e}{R}\right)^2\left[1-5\left(\frac{z}{R}\right)^2\right]\right\}+\frac{F_x}{m},\\
\ddot{y} &= -\frac{\mu}{R^3}y\left\{1+\frac{3}{2}J_2\left(\frac{R_e}{R}\right)^2\left[1-5\left(\frac{z}{R}\right)^2\right]\right\}+\frac{F_y}{m},\\
\ddot{z} &= -\frac{\mu}{R^3}z\left\{1+\frac{3}{2}J_2\left(\frac{R_e}{R}\right)^2\left[3-5\left(\frac{z}{R}\right)^2\right]\right\}+\frac{F_z}{m},\\
m &= m_0-\dot{m}\cdot(t-t_0),
\end{aligned} \tag{1}$$

where $\mu = 3.986004418\times10^{14}(\mathrm{m}^3/\mathrm{s}^2)$represents the gravitational constant of the Earth, $R_e = 6378136.3\mathrm{m}$ represents the equatorial radius of the Earth, $J_2 = 1.082629989052\times10^{-3}$ represents the second-degree zonal harmonic coefficient, and $F_x$, $F_y$, and $F_z$ represent the components of spacecraft thrust in the three-axis directions of the J2000 coordinate system. $\dot{m}$ represents the total fuel mass flow rate. The spacecraft's nonpropulsive propagation model adopts the high-precision high-order perturbation (HPOP) model.

The three-axis components of spacecraft thrust in the J2000 coordinate system are as follows:

$$\begin{bmatrix}F_x\\F_y\\F_z\end{bmatrix} = M_{Jv}\cdot M_{vo}\cdot\begin{bmatrix}f_x\\f_y\\f_z\end{bmatrix}. \tag{2}$$

$f_x$, $f_y$, and $f_z$ represent the spacecraft's body-axis components of thrust in the three-axis directions. The specific calculation method is as follows:

$$f_x = \sum_{i=0}^{N} F_i\cdot\cos\alpha_i. \tag{3}$$

$N$ represents the number of thruster installations, $F_i$ represents the thrust output of a single thruster, and $\alpha_i$ represents the corresponding installation angle of the thruster.

$$F = \mathrm{Isp}\cdot g\cdot\dot{m}. \tag{4}$$

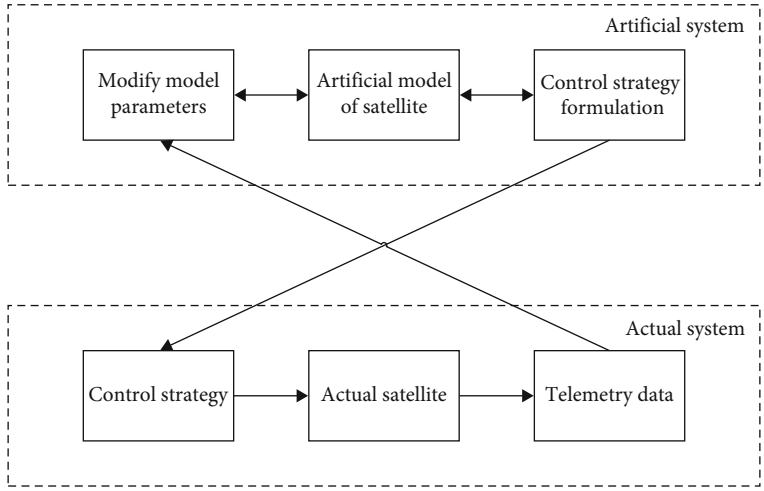


Figure 1: Parallel system of satellite maneuver.

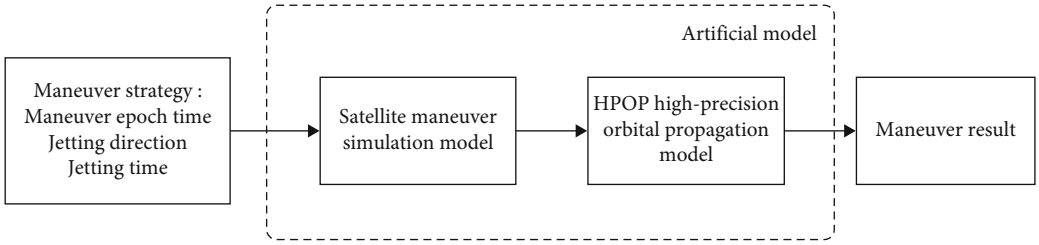


Figure 2: Artificial system of satellite maneuver.

$F$ represents the thrust of the thruster, Isp represents the specific impulse of the spacecraft propellant, $g = 9.80665(\mathrm{m/s^2})$ represents the gravitational acceleration, and $\dot{m}$ represents the fuel mass flow rate. Various methods are available for measuring satellite propellants, including gas law method, volume excitation method, gas injection method, and radioactive method [26]. These methods enable the measurement of fuel mass changes before and after satellite maneuvers. The fuel mass flow rate remains approximately constant during the maneuvering process. Therefore, the average rate of mass change during satellite maneuvering can be used as the fuel mass flow rate.

In summary, an artificial model for satellite maneuvering has been constructed. The actual maneuvering process can be simulated by inputting the control strategy of the actual maneuvering task. The artificial model is depicted in Figure 2.

## 3. Artificial Model Correction

On the basis of the satellite maneuvering artificial model construction, the thrust parameters of the artificial model are adjusted. This adjustment is performed by taking into account the maneuver strategy information and orbit position telemetry data from the actual system. The adjustment process involves utilizing the particle swarm optimization algorithm for computational experiments.

Particle swarm optimization algorithm is evolutionary. It utilizes particle positions as solutions. Numerous particles start from random solutions and iteratively search for the optimal position, which corresponds to the best solution. The advantages of particle swarm optimization algorithm include fast convergence, high efficiency, simple principles, and easy implementation. It incorporates individual local information and collective global information, making it less prone to being trapped in local optimal solutions.

Furthermore, the iteration process of the particle swarm algorithm is essentially a large number of repeated quantitative computational experiments, which aligns well with the characteristics of parallel systems.

The specific process of thrust estimation is as follows: the thrust is considered by the particle position, and the random thrust parameters are generated within a certain range. Then, these thrust parameters are inputted into the satellite maneuvering simulation model, which simulates the maneuvering process based on the actual control strategy. When the simulation of the maneuvering phase is completed, a high-precision orbit prediction model is utilized for orbit propagation, to result in a series of simulated position data. These data correspond to the same epoch as the actual telemetry position data. Subsequently, the difference between the two sets of data is used as the objective function.

$$J(F) = \frac{\sum_{i=1}^{N}\left[(X_i - x_i)^2 + (Y_i - y_i)^2 + (Z_i - z_i)^2\right]}{N}. \quad (5)$$

The symbols $X_i$, $Y_i$, and $Z_i$ represent the actual telemetry position data, whereas $x_i$, $y_i$, and $z_i$ represent the simulated system-calculated position data at the corresponding orbit

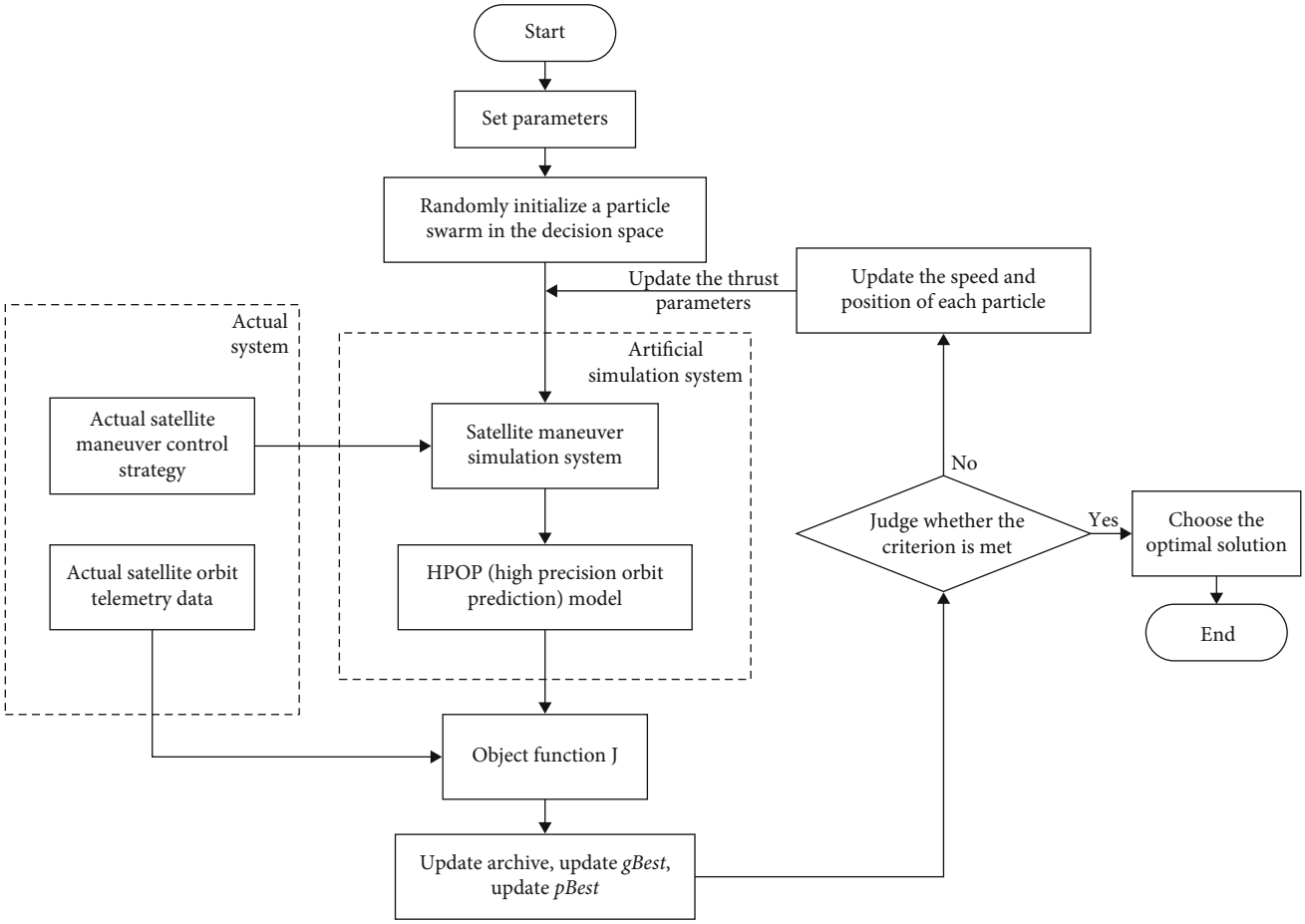


Figure 3: Flow chart of thrust estimation.

epoch time. These values are used as the population fitness. Through multiple iterations, the population's position with the lowest fitness value is determined. This position corresponds to the thrust parameters that minimize the error between the simulation results and the actual position results.

The particle update formulas are as follows:

$$\begin{aligned} \widehat{v}_i &= \omega \cdot v_i + c_1 \cdot \text{rand}\,() \cdot (p\text{best}_i - F_i) + c_2 \cdot \text{rand}\,() \\ &\quad \cdot (g\text{best}_i - F_i), \\ \widehat{F}_i &= F_i + \widehat{v}_i, \\ \omega^{(t)} &= (\omega_{\text{ini}} - \omega_{\text{end}}) \cdot \frac{(G_k - g)}{G_k} + \omega_{\text{end}}, \end{aligned} \tag{6}$$

where $v_i$ is the particle velocity, rand () are random numbers between $(0, 1)$, $F_i$ is the particle's current position (i.e., the thrust value), $p\text{best}_i$ is the particle's historical best thrust value, $g\text{best}_i$ is the population's best thrust value, $c_1$ and $c_2$ are the learning factors, $\omega$ is the inertia factor, $\omega_{\text{ini}}$ is the initial inertia weight, $\omega_{\text{end}}$ is the inertia weight at the maximum number of iterations, $G_k$ is the maximum number of iterations, and $g$ is the current iteration number. The dynamic inertia factor allows the algorithm to have better optimization performance. A higher value provides better global optimization capability, and a lower value enhances local optimization capability.

During the two consecutive maneuver tasks, the thrust value of the satellite thrusters remains approximately constant. Therefore, the currently estimated thrust value can be used to simulate the next maneuver in advance. Then, a maneuver strategy can be devised. After the completion of the next maneuver, the thrust value can be updated again based on telemetry data. This approach enables the real-time and long-term consistency between the simulation system and the actual system.

The specific flowchart of thrust estimation is shown in Figure 3.

## 4. Fast Maneuver Control Strategy

After rectifying the artificial model, it can be parallelly executed with the actual system. The simulation results from the artificial system can be used to predict the outcomes of actual satellite maneuvering tasks. This approach helps validate and evaluate the maneuvering strategies of the actual tasks. Furthermore, the artificial system can be utilized for quantitative and repeatable computational experiments. The actual system is controlled by actively formulating maneuvering strategies based on the requirements of the actual tasks, completing the closed loop of the parallel systems.

The fast satellite maneuvering involves satellite moving a certain distance forward relative to its premaneuver orbit, such as approaching a target. The formulation of maneuvering strategies can use the method of setting a virtual satellite. Placing a "virtual satellite" at the desired position for satellite maneuvering, the satellite maneuvering problem is transformed into a rendezvous problem between the satellite and the "virtual satellite." For satellites in near-circular orbits, in the case of maneuver tasks with relatively small distances of forward movement, the relative motion equations can be described using the CW equation. The required velocity increment for the satellite to rendezvous with the virtual satellite can be solved. This condition serves as the foundation for formulating maneuvering strategies. The relative motion equations are as follows:

$$\begin{aligned} \delta\ddot{x} - 3n^2\delta x - 2n\delta\dot{y} &= 0, \\ \delta\ddot{y} + 2n\delta\dot{x} &= 0, \\ \delta\ddot{z} + n^2\delta z &= 0, \end{aligned} \tag{7}$$

where $\delta x$, $\delta y$, and $\delta z$ represent the relative position vector components along the three axes of the virtual satellite's STW coordinate system and $n$ represents the angular velocity of the virtual satellite.

Continuing the solution of equation (7), the relative position vector and relative velocity vector can be obtained as follows:

$$\begin{aligned} \delta r(t) &= \phi_{rr}(t)\delta r_0 + \phi_{rv}(t)\delta v_0, \\ \delta v(t) &= \phi_{vr}(t)\delta r_0 + \phi_{vv}(t)\delta v_0. \end{aligned} \tag{8}$$

In the above equation,

$$\begin{aligned} \phi_{rr}(t) &= \begin{bmatrix} 4 - 3\cos nt & 0 & 0 \\ 6(\sin nt - nt) & 1 & 0 \\ 0 & 0 & \cos nt \end{bmatrix}, \\ \phi_{rv}(t) &= \begin{bmatrix} \frac{1}{n}\sin nt & \frac{2}{n}(1 - \cos nt) & 0 \\ \frac{2}{n}(\cos nt - 1) & \frac{1}{n}(4\sin nt - 3nt) & 0 \\ 0 & 0 & \frac{1}{n}\sin nt \end{bmatrix}, \\ \phi_{vr}(t) &= \begin{bmatrix} 3n\sin nt & 0 & 0 \\ 6n(\cos nt - 1) & 0 & 0 \\ 0 & 0 & -n\sin nt \end{bmatrix}, \\ \phi_{vv}(t) &= \begin{bmatrix} \cos nt & 2\sin nt & 0 \\ -2\sin nt & 4\cos nt - 3 & 0 \\ 0 & 0 & \cos nt \end{bmatrix}. \end{aligned} \tag{9}$$

The calculation method for the relative position and velocity vectors of the two stars in the virtual star's STW coordinate system at the initial time is as follows:

$$\begin{aligned} \delta r_0 &= Q_{Xx}(r - r_V), \\ \delta v_0 &= Q_{Xx}(v - v_V - \Omega_V \times \delta r), \end{aligned} \tag{10}$$

where $r$, $r_V$, $v$, and $v_V$ represent the position and velocity vectors of the satellite and virtual satellite in the inertial frame, $\Omega_V$ is the angular velocity of the virtual satellite, and $Q_{Xx}$ is the transformation matrix from the inertial frame to the virtual satellite's STW coordinate system. Therefore, by providing the initial relative position vector and the rendezvous time with the virtual satellite, the initial relative velocity vector required for the rendezvous orbit can be obtained. Thus, the velocity increment for the satellite maneuver is given as follows:

$$\Delta v_0 = \delta v_0^+ - \delta v_0^- = \begin{bmatrix} \delta u_0^+ \\ \delta v_0^+ \\ \delta w_0^+ \end{bmatrix} - \begin{bmatrix} \delta u_0^- \\ \delta v_0^- \\ \delta w_0^- \end{bmatrix}. \tag{11}$$

By evaluating equation (8) at the rendezvous time $t_f$, we obtain

$$\delta v_0^+ = -\phi_{rv}^{-1}(t_f)\phi_{rr}(t_f)\delta r_0. \tag{12}$$

When formulating actual satellite maneuver control strategies, the velocity increment in the satellite body frame should be determined. Therefore, a coordinate transformation should be performed on the velocity increment.

The control strategy for the actual satellite involves the thrusting time. A correlation between the velocity increment and the thrusting time is established. The relationship between velocity increment along one axis and thrust is given by the following:

$$\int_0^{t_x} \frac{F_x}{m - \dot{m} \cdot t_x} dt = \Delta v_x. \tag{13}$$

The following can be derived from the above equation:

$$\Delta v_x = \frac{F_x}{\dot{m}} \ln \frac{m}{m - \dot{m} \cdot t_x}. \tag{14}$$

The equation relating the thrusting time and the velocity increment can be solved as follows:

$$t_x = \frac{m - m \cdot e^{-(\Delta v_x \cdot \dot{m}/F_x)}}{\dot{m}}. \tag{15}$$

The required thrusting time for the maneuver can be calculated using the above equation.

The problem lies in the fact that the CW equation is derived from the ideal two-body model and does not include factors such as perturbations.

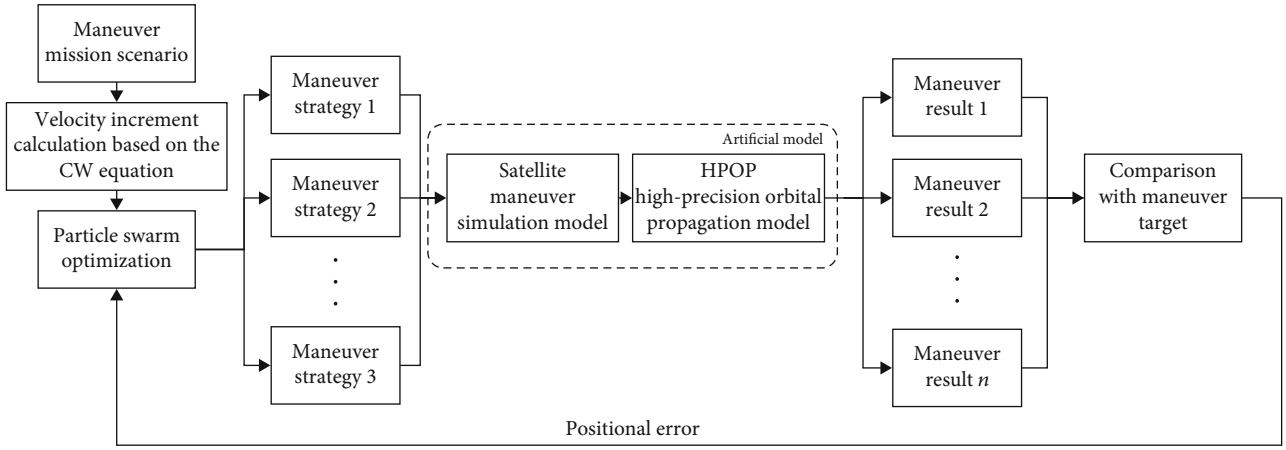


Figure 4: Flow chart of maneuver strategy.

Subsequently, using a constant thrust for the satellite does not provide an instantaneous velocity increment. This condition leads to errors in the calculated thrusting time based on the CW equation.

A maneuver control strategy formulation method, combining an artificial model with the particle swarm algorithm, is proposed to address this issue.

The thrusting time is considered the particle position and the initial thrusting time obtained from the preliminary solution of the CW equation as a reference. A range of ±50 seconds is set as the particle position range. Specific thrusting times are randomly generated within this range using the particle swarm algorithm.

These generated thrusting times are then used in computational experiments with the artificial model of the satellite. The artificial model consists of two components: the satellite maneuver simulation model with modified thrust parameters and the HPOP high-precision orbital recurrence model. The difference between the simulation results and the maneuver target position is used as the objective function for iterative optimization. As a result, the thrusting time that minimizes the error with the maneuver target position is obtained. The maneuver strategy estimation process is shown in Figure 4.

During the execution of satellite maneuvering tasks, the maneuvering strategy is determined using the aforementioned method. This condition enables the actual satellite to reach the target position with greater accuracy in a single maneuver, while reducing the terminal guidance pressure and minimizing fuel consumption. In addition, this approach allows feedback from the artificial system to the actual system, achieving a closed loop in parallel systems.

When performing maneuvering tasks, the actual system operates according to the jetting strategy provided by the simulation system. Subsequently, the telemetry data of the maneuvering results are fed back to the artificial system to adjust its parameters once again. By synchronously advancing and running in parallel with the actual system, the artificial system evolves in synchronization, maintaining long-term consistency throughout the satellite's lifespan.

After accumulating a certain amount of data, estimating the maneuvering lifespan of the satellite, specifically the relationship between thrust magnitude and thrusting time, becomes possible. This condition further enables a certain degree of independent operation for the artificial system. The artificial system can utilize this independence to perform various maneuvering simulation experiments that are infeasible in actual scenarios. Conducting numerous simulation experiments and exploring different conditions are aimed at uncovering emergent phenomena and providing ample experimental data support for the execution of satellite maneuvering tasks.

## 5. Simulation Results

To validate the method, the following application scenario is simulated with reference to the actual satellite maneuver mission: Assuming a geostationary orbit satellite (with a total mass $m = 1527\text{kg}$), the satellite maintains nadir-pointing orientation. Starting from the initial orbit (with epoch parameters $T$: $2022-1-1-000000$, $A = 42166.3\text{km}$, $e = 0$, $i = 0.12°$, $\Omega = 90.45°$, $\omega = 10.75°$, and $\theta = 204.0°$), the satellite undergoes orbit maneuver in the $-z$ direction of the satellite orbit coordinate system with a constant thrust $F = 25N$. The thrusting time is set as $t = 300\text{s}$, and the fuel mass flow rate is $\dot{m} = 0.0085\text{kg/s}$. Incorporate solar radiation pressure and three-body gravitational perturbation into the simulation scenario and introduce a range measurement noise of 100 meters.

Using the particle swarm algorithm to estimate thrust, the learning factor $c_1 = 2$, $c_2 = 2$, $c_2 = 2$, initial inertia weight $\omega_{\text{ini}} = 0.5$, final inertia weight $\omega_{\text{end}} = 1$, population size popsize = 200, number of iterations $N = 100$, and particle position range $[10, 30]$ are set.

Figure 5 illustrates the schematic of particle positions in each generation during the thrust estimation process. The transition from blue to yellow in the figure represents the number of iterations in the particle swarm optimization (PSO) algorithm. As the number of iterations increases, the particles in the swarm gradually converge toward the

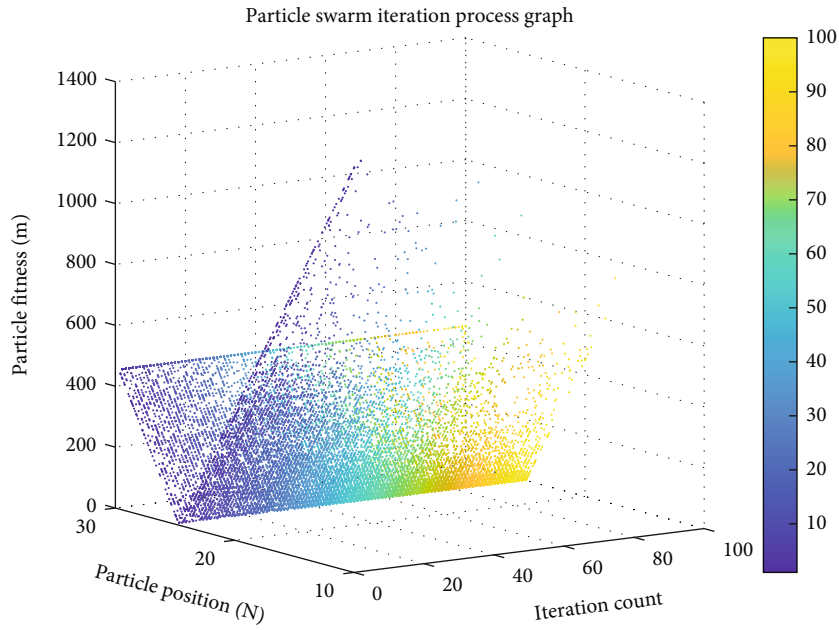


Figure 5: Iterative graph of particle swarm optimization algorithm.

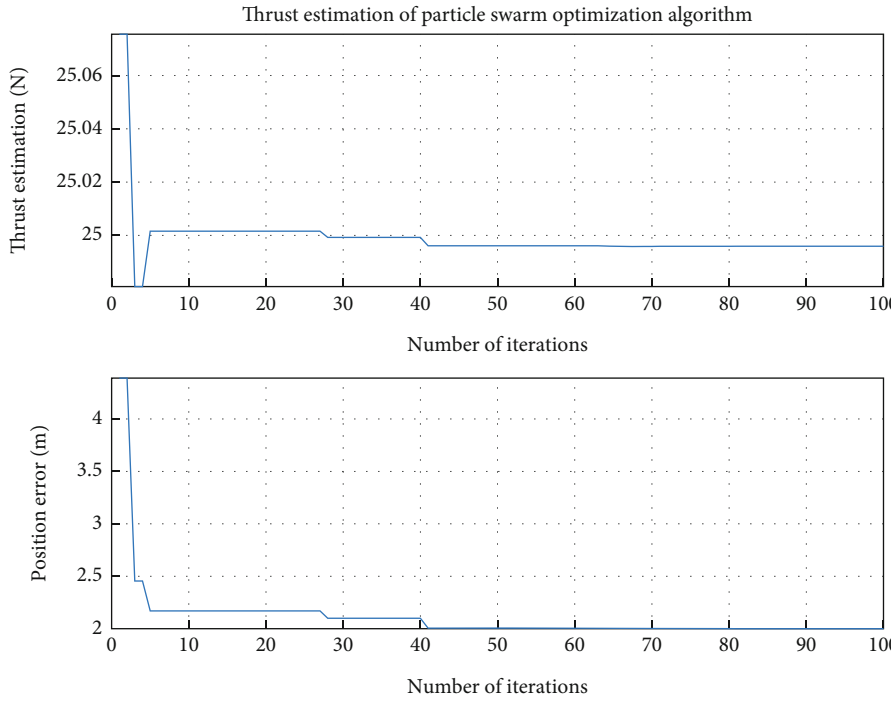


Figure 6: Thrust estimation result.

position with the minimum fitness, without being trapped in a local optimal solution.

Figure 6 shows the final thrust estimation results, with a computation time of 78.364 seconds. The optimal estimated thrust value is 24.9689 N, with an error of 0.1244% compared with the actual thrust value. Under this thrust value, the error between the simulated model's propagated orbital position and the telemetry data is 2.0564 meters. The estimation results validate the feasibility of the method, demonstrating its ability to achieve fast and accurate thrust estimation. Subsequently, thrust estimation using telemetry data at different time intervals yielded the following results, as shown in Figure 7 and Table 1.

As the epoch time of telemetry data extends, the computation time for orbital propagation increases considerably. Nevertheless, the accuracy of thrust estimation improves mainly due to the increase in satellite displacement as time progresses, relatively reducing the impact of measurement

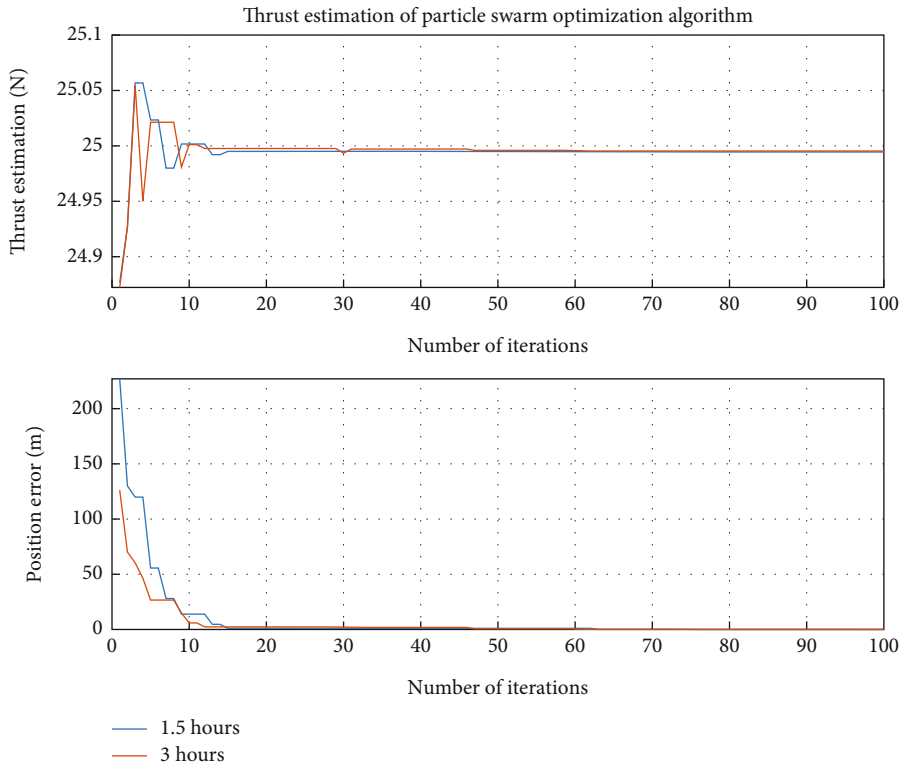


Figure 7: Thrust estimation results of different telemetry data.

Table 1: Comparison of thrust estimation with different telemetry data.

| Telemetry position data | Estimated thrust value ($N$) | Positional error ($m$) | Thrust estimation error | Computation time ($s$) |
|---|---|---|---|---|
| 5 minutes | 24.9689 | 2.0564 | 0.1244% | 79.905 |
| 1.5 hours | 24.9927 | 2.8744 | 0.0292% | 80.247 |
| 3 hours | 24.9944 | 6.8502 | 0.0224% | 130.815 |

noise in telemetry data. Furthermore, the particle swarm calculation time is obviously less than the time between two maneuver tasks, which proves the feasibility of using modified thrust parameters to simulate the next maneuver.

Scenarios with different actual thrust values (24 N-16 N) are simulated to verify the robustness of this method under various relative disturbances. The specified variation range indicates the actual fluctuation range of the thruster's performance over the entire lifespan of the spacecraft. The simulation results are shown in Figure 8 and Table 2.

Figure 8 and Table 2 indicate that this method can rapidly and accurately estimate under different thrust values. As the actual thrust value decreases, the relative disturbance increases, and the estimation accuracy decreases slightly but still remains within 0.1%. This demonstrates the robustness of the method.

After completing the thrust estimation, the formulation of maneuver control strategies can be carried out. Taking the satellite mentioned above, the initial condition for maneuvering is set to 12 hours after the first maneuver. The target orbit positions are $T: 2022-1-1-120500$, $A = 42165.9\text{km}$, $e = 0$, $i = 0.12^\circ$, $\Omega = 90.267^\circ$, $\omega = 250.56^\circ$, and $\theta = 125.728^\circ$. A virtual satellite is set in the target orbit. The relative positions and velocities between the two satellites in the virtual satellite's coordinate system are calculated at the initial time.

$$\delta r_0 = \begin{bmatrix} -0.01 \\ -29.118 \\ 0 \end{bmatrix} (\text{km}),$$
$$\delta v_0 = \begin{bmatrix} -0.000527 \\ -0.00000128 \\ 0 \end{bmatrix} (\text{km/s}). \tag{16}$$

Calculate the required velocity increment in the satellite's orbit coordinate system.

$$\Delta v = \begin{bmatrix} 0.000534 \\ 0.00000229 \\ 0 \end{bmatrix} (\text{km/s}). \tag{17}$$

The thrusting time required in the $x$-axis direction is calculated as $t_x = 32.026s$ and in the $y$-axis direction as $t_y = 0.0768s$. These values are used as references to set the particle range. The thrusting time range in the $x$-axis direction is $[0, 100]$, and the thrusting time range

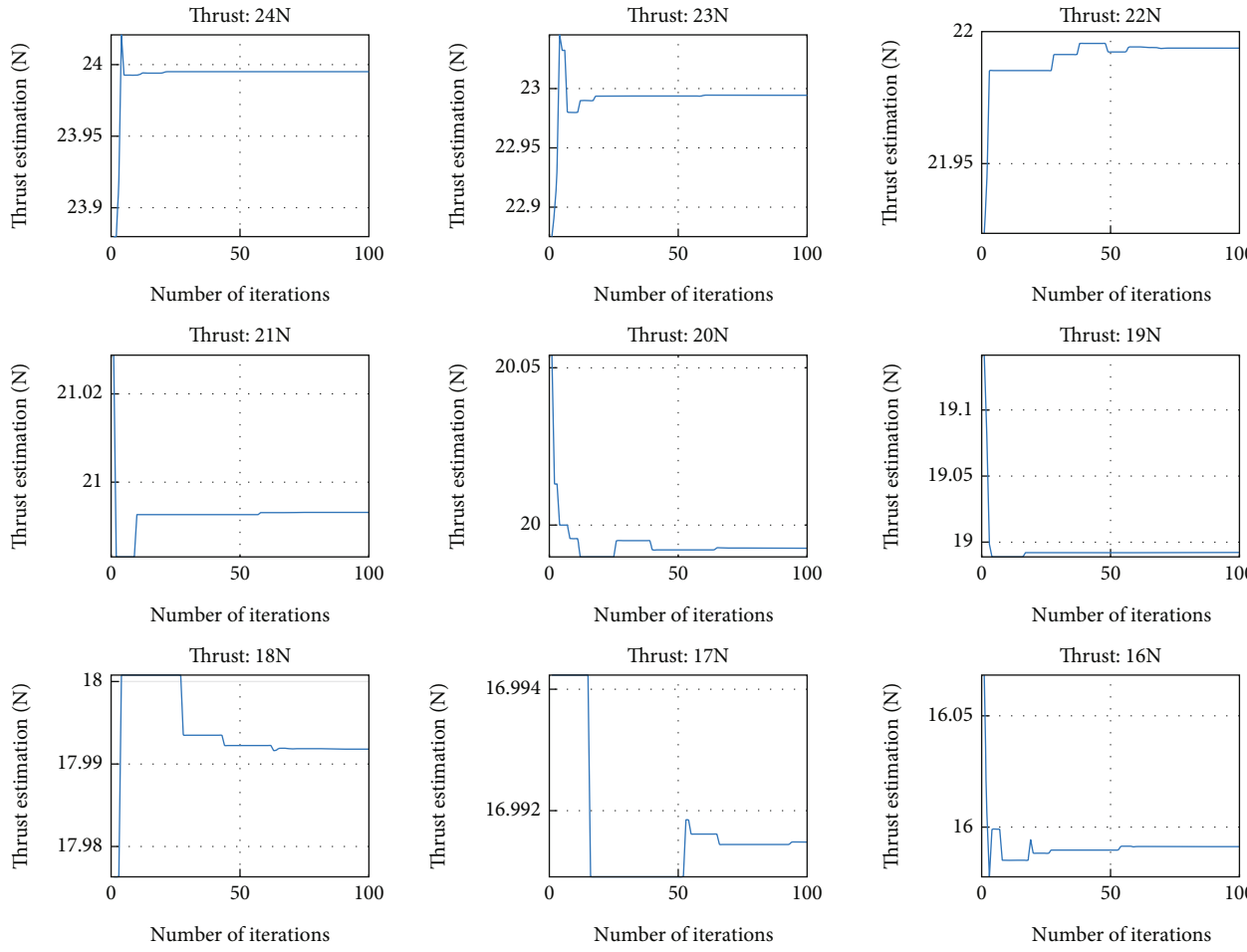


Figure 8: Thrust estimation results of different thrust.

Table 2: Comparison of estimation errors with different actual thrust values.

| Actual ($N$) | 24 | 22 | 20 | 18 | 16 |
|---|---|---|---|---|---|
| Estimation ($N$) | 23.9951 | 21.9938 | 19.9928 | 17.9921 | 15.9912 |
| Error | 0.0203% | 0.0280% | 0.0361% | 0.0439% | 0.0549% |

in the $y$-axis direction is $[0, 100]$. The particle swarm algorithm is used for estimation.

Figure 9 represents a schematic of the computational experiment using the artificial model. The orbit of the maneuvered satellite is obtained by generating a large number of different jet thrust strategies within the specified range and performing orbit propagation simulations. The figure shows the relative motion trajectory between the maneuvered satellite and the original satellite within a 12-hour period. On this basis, the particle swarm algorithm is used to iterate and search for the motion trajectory with the smallest target position error, ultimately obtaining the optimal jet thrust strategy.

Figure 10 represents the results of jet timing estimation. The final estimated result for the thrusting time in the $x$-axis direction is $t_x = 30.513s$ and $t_y = 2.075s$ in the $y$-axis direction, with a positional error of $79.827m$ between the maneuvered satellite and the target position after 12 hours.

Traditional methods assume that the satellite can instantaneously obtain velocity increments during maneuvering and typically use the CW equation or the Lambert problem to calculate these velocity increments. Simulating the results obtained from the traditional methods and compared with the method (ACP) is proposed in this paper for further analysis. The simulation results are shown in Figure 11 and Table 3.

The results demonstrate that, compared with traditional methods, this approach can accurately determine the required thrusting time for maneuvering tasks. This condition enables the satellite to reach the target position with greater accuracy after a single maneuver.

## 6. Conclusion

The aim of this study is to address the challenge of varying and unmeasurable thrust in fast satellite orbit maneuvers. We established a parallel system framework for rapid satellite orbit maneuvers and proposed a thrust estimation

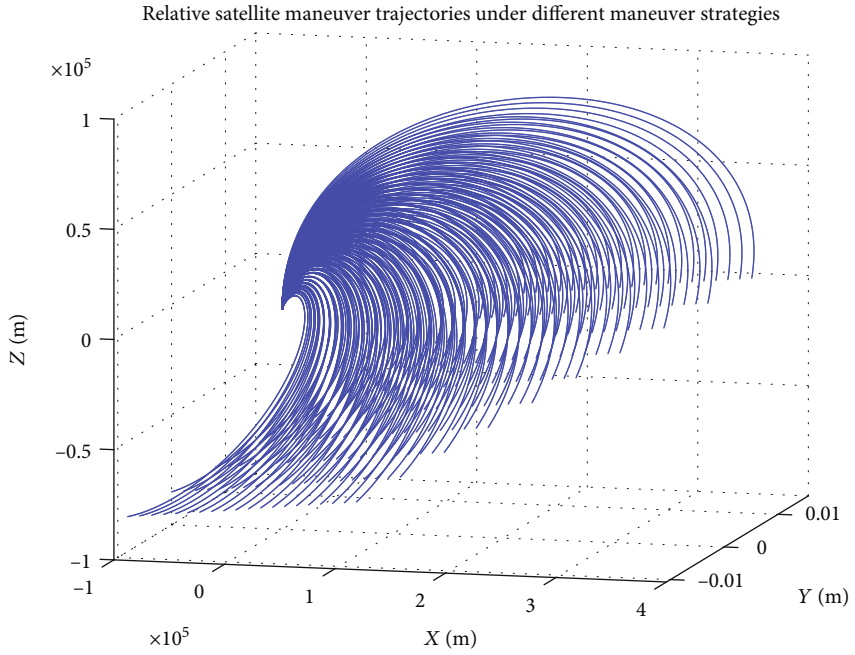


Figure 9: Relative trajectories of different maneuver strategies.

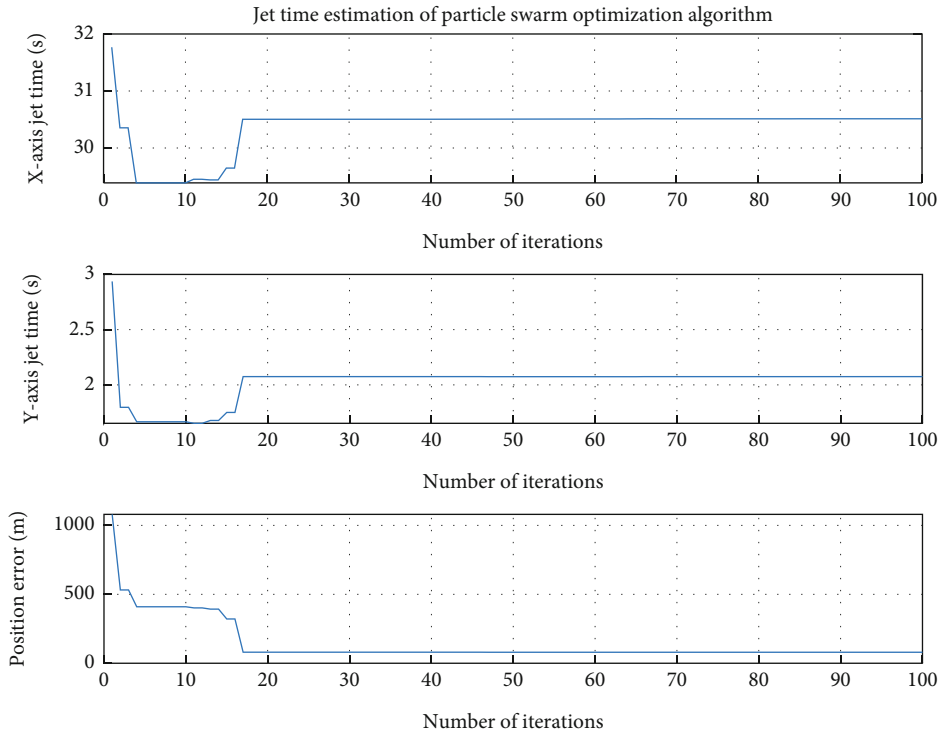


Figure 10: Estimated results of maneuver strategy.

method based on the particle swarm optimization (PSO) algorithm. The main contributions and improvements to existing technologies of this study are reflected in the following.

We successfully achieved thrust estimation by introducing a parallel system framework. By interacting with the actual system, parallel systems can evolve to be consistent with the actual system. It performs well in the estimation of the real thrust. Compared to traditional methods, this method can effectively solve the problem of constantly changing thrust. The estimation accuracy of thrust is better than 0.1%.

We propose an innovative jet time estimation method that combines the intelligent strategies of artificial systems

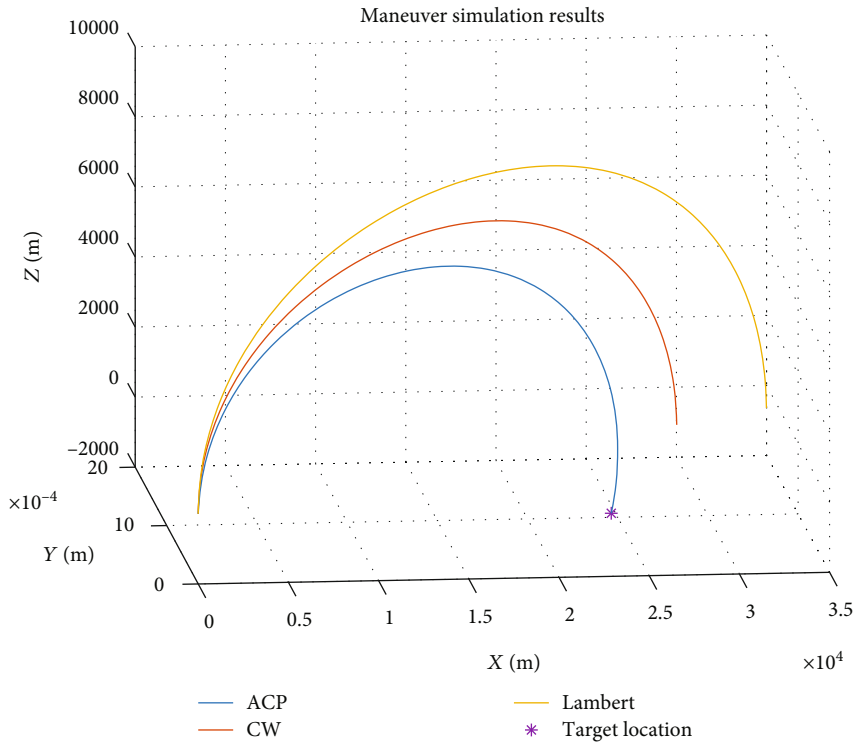


Figure 11: Simulation results of orbit maneuver.

Table 3: Comparison of maneuver strategy.

| | $X$-axis jet time ($s$) | $Y$-axis jet time ($s$) | Position error ($m$) |
|---|---|---|---|
| CW | 32.026 | 0.0768 | 4693.018 |
| Lambert | 38.6838 | 0.956 | 8560.454 |
| ACP | 30.513 | 2.075 | 79.827 |

with actual maneuvering tasks. The simulation results have verified the superiority of this method in the actual maneuvering process. In contrast to the traditional Lambert method, this approach can enable the satellite to achieve extremely close accuracy to the target position after a single maneuver.

This research makes a progress in the field of fast satellite orbit maneuvers. We not only proposed an innovative solution to the thrust estimation problem but also enhanced the accuracy and efficiency of satellite maneuvering processes through the incorporation of intelligent maneuvering strategies. Compared to existing techniques, our research approach demonstrates higher applicability and reliability in solving practical problems.

## Data Availability

The underlying data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work is supported by the Fundamental Research Funds for the Central Universities of China (Grant No. NT2022024).

## References

[1] X. B. Zhang, L. X. Lin, and X. H. Suo, "Simulation technology of satellite control system," *Computer Simulation*, vol. 17, no. 2, pp. 57–59, 2000.

[2] S. Q. Yang, Y. S. Jia, and J. Cui, "A new method for calculating the injection time of satellite gas cylinder thruster," *Shanghai Aerospace*, vol. 35, no. 4, pp. 48–53, 2018.

[3] L. Yuan, H. L. Zhou, and B. Sun, "Numerical simulation and test of pressure drop characteristics of hydrazine propulsion system in satellite unit," *Shanghai Aerospace*, vol. 24, no. 2, pp. 42–46, 2007.

[4] W. Wang, J. P. Yuan, and J. J. Luo, "Optimal single impulse maneuver for spacecraft cluster flight," *Chinese Journal of Theoretical and Applied Mechanics*, vol. 47, no. 5, pp. 799–806, 2015.

[5] W. Ding, X. Li, and H. Yang, "Research on minimum time interception problem with a tangent impulse under relative motion models," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 6827982, 14 pages, 2019.

[6] H. Shen and P. Tsiotras, "Using Battin's method to obtain multiple-revolution Lambert's solutions," *Advances in the Astronautical Sciences*, vol. 116, pp. 1067–1084, 2003.

[7] D. L. Li and F. M. Huang, "Research on precise interception and rendezvous strategy based on Lambert problem," *Flight Dynamics*, vol. 100, no. 2, pp. 57–59, 2008.

[8] A. Y. Huang, Y. Z. Luo, and H. N. Li, "Global optimization of ergodic rendezvous orbit of near-earth constellation," *Chinese Journal of Astronautics*, vol. 43, no. 6, pp. 772–780, 2022.

[9] S. Oghim, H. Leeghim, and D. Kim, "Real-time spacecraft intercept strategy on J2-perturbed orbits," *Advances in Space Research*, vol. 63, no. 2, pp. 1007–1016, 2019.

[10] F. Y. Wang, "Artificial society, computational experiment, parallel system– discussion on computational research of complex socio-economic system," *Complex Systems and complexity Science*, vol. 9, no. 4, pp. 25–35, 2004.

[11] J. Zhang, P. D. Xu, and F. Y. Wang, "A data-driven representation and computing framework for parallel systems and digital twins," *Acta Automatica Sinica*, vol. 46, no. 7, pp. 1346–1356, 2020.

[12] F. Y. Wang, "Parallel control: data-driven computational control method," *Journal of Automation*, vol. 39, no. 4, pp. 293–302, 2013.

[13] C. L. Ge, Y. C. Zhu, Y. Q. Di, Z. W. Hu, and X. G. Meng, "Research on the theoretical framework of equipment parallel simulation," *Journal of Command and Control*, vol. 3, no. 1, pp. 48–56, 2017.

[14] F. Zhou, S. J. Mao, Y. C. Wu, and Y. P. Li, "Parallel simulation deduction method driven by real-time situation data," *Journal of the Chinese Academy of Electronic Sciences*, vol. 15, no. 4, pp. 323–328, 2020.

[15] L. Yuan, M. Cheng, and L. Wang, "Spacecraft flight control simulation and parallel system," *Journal of Astronautics*, vol. 42, no. 8, pp. 982–988, 2021.

[16] M. Chen, Q. Sun, and D. K. Wang, "Application of parallel system method in command and control for anti-missile system," *Command, Control and Simulation*, vol. 39, no. 1, 2017.

[17] L. Jin, J. C. Li, Z. Q. Chang, J. W. Lu, and L. Cheng, "Attitude control of flapping-wing micro air vehicle based on ACP theory," *Acta Automation*, vol. 27, no. 3, pp. 1–13, 2023.

[18] J. Barreto, L. Hoffmann, and A. Ambrosio, "Using SMP2 standard in operational and analytical simulators," in *SpaceOps 2010 Conference*, p. 2267, Huntsville, AL, USA, 2010.

[19] J. M. Penn and A. S. Lin, "The trick simulation toolkit: a NASA/opensource framework for running time based physics models," in *AIAA Modelling and Simulaiton Technologies Conference*, p. 1187, San Diego, CA, USA, 2016.

[20] M. Schluse and J. Rossmann, "From simulation to experimental digital twins: simulation-based development and operation of complex technical systems," in *Proceedings of the 2016 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 1–6, Edinburgh, England, 2016.

[21] G. N. Schroeder, C. Steinmetz, C. E. Pereira, and D. B. Espindola, "Digital twin data modeling with AutomationML and a communication methodology for data exchange," *IFAC Papers Online*, vol. 49, no. 30, pp. 12–17, 2016.

[22] J. R. Surdu and K. Kittka, "Deep green: commander's tool for COA's concept," in *Computing, communications and control technologies: CCCT*, pp. 45–51, Orlando, FL, USA, 2008.

[23] "Air Force Research Laboratory. What is Skyborg? [EB/OL]," (2020-02-21) [2021-04-11], https://afresearchlab.com/technology/vanguards/successstories/skyborg.

[24] J. David, A. Lobov, and M. Lanz, "Attaining learning objectives by ontological reasoning using digital twins," *Procedia Manufacturing*, vol. 31, pp. 349–355, 2019.

[25] F. Caputo, A. Greco, M. Fera, and R. Macchiaroli, "Digital twins to enhance the integration of ergonomics in the workplace design," *International Journal of Industrial Ergonomics*, vol. 71, pp. 20–31, 2019.

[26] D. A. Da and T. P. Zhang, "Review of liquid propellant measurement technology for in-orbit satellite [J]," *Propulsion Technology*, vol. 18, no. 4, pp. 89–94, 1997.