



Fast Two-dimensional Positioning Method of Crab Pulsar Based on Multiple Optimization Algorithms

Jianfeng Chen¹, Jin Liu^{1,2} , Xin Ma², and Xiaolin Ning^{2,3}

¹ College of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081, China; liujin@wust.edu.cn

² School of Instrumentation Science and Opto-electronics Engineering, Beihang University, Beijing 100191, China

³ Hefei National Laboratory, Hefei 230088, China

Received 2024 July 2; revised 2024 August 12; accepted 2024 August 31; published 2024 September 26

Abstract

In the two-dimensional positioning method of pulsars, the grid method is used to provide non-sensitive direction and positional estimates. However, the grid method has a high computational load and low accuracy due to the interval of the grid. To improve estimation accuracy and reduce the computational load, we propose a fast two-dimensional positioning method for the crab pulsar based on multiple optimization algorithms (FTPCO). The FTPCO uses the Levenberg–Marquardt (LM) algorithm, three-point orientation (TPO) method, particle swarm optimization (PSO) and Newton–Raphson-based optimizer (NRBO) to substitute the grid method. First, to avoid the influence of the non-sensitive direction on positioning, we take an orbital error and the distortion of the pulsar profile as optimization objectives and combine the grid method with the LM algorithm or PSO to search for the non-sensitive direction. Then, on the sensitive plane perpendicular to the non-sensitive direction, the TPO method is proposed to fast search the sensitive direction and sub-sensitive direction. Finally, the NRBO is employed on the sensitive and sub-sensitive directions to achieve two-dimensional positioning of the Crab pulsar. The simulation results show that the computational load of the FTPCO is reduced by 89.4% and the positioning accuracy of the FTPCO is improved by approximately 38% compared with the grid method. The FTPCO has the advantage of high real-time accuracy and does not fall into the local optimum.

Key words: (stars:) pulsars: individual (Crab pulsar) – space vehicles – methods: data analysis – astrometry

1. Introduction

X-ray Pulsar Navigation (XPNAV) system is an autonomous navigation system (Zhang et al. 2021). XPNAV use X-ray pulsar to estimate the position and velocity of the spacecraft. When the spacecraft receives pulsar signals from multiple pulsars, the position and velocity of the spacecraft can be estimated accurately (An et al. 2023; Sun et al. 2023; Wang et al. 2023b).

In recent years, with the deepening of research in the field of XPNAV (Wang et al. 2019; Qian & Pan 2021; Zhang et al. 2024), two main situations exist for pulsar period estimation: one is transfer orbit or uniform linear motion, where the velocity error within the pulsar observation period can be considered constant (Liu et al. 2023a), the other is nonlinear orbit, such as circular orbit, where the positional drift on the pulsar Line Of Sight (LOS) caused by both position and velocity can cause the distortion of the accumulative pulsar profile (Liu et al. 2023b, 2024).

The degree of nonlinearity in spacecraft orbit affects pulsar velocity estimation or period estimation based on the distortion of the pulsar profile. For linear orbit, the chi-square method (Xu et al. 2019), the fast Fourier transform (Zhang et al. 2019) the frequency subdivision-based period search algorithms

(Shen et al. 2015) and a frequency domain method based on the Fourier series (Song et al. 2022) can achieve good performances in pulsar velocity estimation or period estimation. Given limited star-borne computing resources, methods such as the fast butterfly epoch folding (Zhang et al. 2015), fast Fourier transform based on compressive sensing and period estimation using compressed sensing (Liu et al. 2019), ultra-fast estimation method of pulsar period based on empirical mode decomposition-compressed sensing (Liu et al. 2020). However, in these methods, the actual complex deep space environment is not considered. To address the challenge of phase evolution in nonlinear orbit, Liu proposed an innovative nonlinear least squares joint estimation method (Liu et al. 2014). Wang introduced an approximate value of pulse phase evolution and combined it with linear polynomial fitting to optimize the accuracy of spacecraft orbit (Wang & Zhang 2016). Based on this, Wang further established a phase shift model. He successfully estimated the hyper-parameters in the pulsar phase model (Wang et al. 2023a), which provided important support for the performance improvement of XPNAV systems. Besides, Wang proposed an absolute navigation scheme for final approach phase using relative measurements of X-ray pulsar and orbiter (Wang et al. 2017).

Through observability analysis, absolute navigation using relative information is proved observable under the condition of reasonable pulsar selection. Inspired by the observable analysis, Liu applied an observable analysis method to determine two vectors with good observability in two velocity directions. The expansion of the two-dimensional search in these directions and the reduction of computational complexity using dictionary folding (Liu et al. 2023b), effectively improved the efficiency of the navigation system. To accurately estimate the period of X-ray pulsars and enhance navigation performance, Ma proposed an energy-based pulsar period estimation method using Hilbert curve and double CNNs (Ma et al. 2024). For deep space navigation, the noise is time-varying, and its statistics cannot be accurately determined. Gui proposed a variational Bayesian implicit unscented Kalman filter method (Gui et al. 2023), to solve this problem, it can obtain high-accuracy navigation. All of the above methods improve the accuracy of XPNVAV.

In recent years, swarm intelligence optimization algorithms (Muller et al. 2023) have gained significant attention and are applied in various fields such as path planning (Li & Yu 2023) and health monitoring (Achouri et al. 2023). Classical swarm intelligence algorithms include genetic algorithm (Carpintero & Melita 2018), Particle Swarm Optimization (PSO; Wang et al. 2018), and ant colony optimization (Li et al. 2022), new optimization algorithms continue to emerge, which provide more choices for problem-solving. Among them, the nutcracker optimizer algorithm (Abdel-Basset et al. 2023a) is based on simulating the behavior of nutcrackers in their natural environment, which achieves efficient search and optimization processes through simulates communication and cooperation among individuals in the group. The nutcracker optimizer algorithm demonstrates excellent performance in solving complex problems. The spider wasp optimizer algorithm (Abdel-Basset et al. 2023b) is inspired by the hunting, nest-building and mating behaviors of female spider wasps, which achieved effective solutions for multi-objective optimization problems. The spider wasp optimizer algorithm exhibits high search accuracy and robustness. The Newton–Raphson-based optimizer (NRBO; Sowmya et al. 2024) is inspired by Newton–Raphson’s approach. It explores the entire search process using two rules: the Newton–Raphson search rule and the trap avoidance operator and a few groups of matrices to explore the best results further. The NRBO demonstrates outstanding performance in mathematical modeling and engineering optimization, which provides strong support for solving complex problems. When faced with complex navigation problems, a single optimization algorithm is limited and struggles to achieve efficient solutions. Therefore, the application of multiple swarm intelligence optimization algorithms to the navigation problem is expected to improve the performance and stability of the XPNVAV system.

In nonlinear orbit, for fast positioning, we propose a Fast Two-dimensional Positioning method for the Crab pulsar based on multiple optimization algorithms (FTPCO) to reduce the computational load. First, the Levenberg–Marquardt (LM) algorithm or PSO significantly reduces the computational load of the non-sensitive direction search compared to the grid method. For the search for direction on the plane perpendicular to the non-sensitive direction, the Three-Point Orientation (TPO) method is employed to fast determine the two most accurate directions, namely the sensitive direction and the sub-sensitive direction. For the two directions, the NRBO is used for two-dimensional positioning instead of the traditional grid method. The NRBO efficiently solves the optimization problem for multi-dimensional and complex functions, which significantly saves time. In this way, the overall computational load is greatly reduced with the optimization algorithm as compared to the grid methods.

2. Theoretical Basis for Non-sensitive Direction

In this section, we theoretically prove the existence of the non-sensitive direction using orbital dynamics models and singular value decomposition. The singular value decomposition is performed according to the observability matrix. When the singular values are smaller, the observability is smaller. We prove the existence of the non-sensitive direction. Two arbitrary directions perpendicular to the non-sensitive direction are used to construct a chi-square model. Besides, we demonstrate that the distortion of the pulsar profile exhibits a ridge shape, with the approximate flat plane on both sides of the ridge. Based on this, the TPO method can be used to determine the sensitive direction.

2.1. Existence of Non-sensitive Direction

In this section, we prove the existence of non-sensitive direction by using the singular values of the observability matrix. First, we provide the error expressions at multiple time instants are provided. Then the observability matrix is constructed. Finally, the singular values of the observability matrix are analyzed. A small singular value demonstrates the presence of the non-sensitive direction. Namely, the error in the non-sensitive direction does not cause significant distortion.

In the inertial J2000.0 coordinate system, the orbital dynamics model can be shown as:

$$\dot{\mathbf{X}}(t) = \mathbf{f}(\mathbf{X}(t), t) \quad (1)$$

Where, $\mathbf{X}$ is the state.

Equation (1) discretisation can be shown as:

$$\mathbf{X}_i = \Phi_{i,i-1} \mathbf{X}_{i-1} \quad (2)$$

where, $\Phi_{i,i-1} \approx \mathbf{F}_{i-1} \cdot \Delta T + \mathbf{I}_{6 \times 6}$, $\mathbf{I}$ is the identity matrix; ΔT is the sampling period; $\mathbf{F}_{i-1}$ is the Jacobian matrix of the state transition equation at time t_{i-1} . The effects of solar perturbation

and atmospheric perturbation are ignored. Hence, F_{i-1} can be expressed as:

$$F_{i-1} = \frac{\partial f(X(t_{i-1}), t_{i-1})}{\partial X^T(t_{i-1})} \bigg|_{X(t)=\hat{X}(t_{i-1})} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \\ \mathbf{S}_{i-1} & \mathbf{0}_{3 \times 3} \end{bmatrix} \quad (3)$$

where $\mathbf{S}_{i-1}$ is the component of the earth's gravitational acceleration in different directions.

Set X_0 be the initial state of the spacecraft. The spacecraft state at time i , X_i , can be shown as:

$$X_i = \Phi_{i,i-1} \cdot \Phi_{i-1,i-2} \cdots \Phi_{1,0} \cdot X_0 \quad (4)$$

$$\mathbf{r}_i^r = [\mathbf{n}, \mathbf{0}_{1 \times 3}] \cdot \Phi_{i,i-1} \cdot \Phi_{i-1,i-2} \cdots \Phi_{1,0} \cdot X_0 \quad (5)$$

where, $\mathbf{n}$ is the pulsar LOS direction; $\mathbf{r}_i^r$ is the projection of the position at time i onto the LOS direction. Set the error at time i be δX_i . Then the positional error on the LOS direction at this time is $\delta \mathbf{r}_i^r$, which expression can be given as:

$$\delta \mathbf{r}_i^r = [\mathbf{n}, \mathbf{0}_{1 \times 3}] \cdot \Phi_{i,i-1} \cdot \Phi_{i-1,i-2} \cdots \Phi_{1,0} \cdot \delta X_0. \quad (6)$$

The specific expression at time t_1 is shown as:

$$\begin{aligned} \delta \mathbf{r}_1^r &= [\mathbf{n}, \mathbf{0}_{1 \times 3}] \cdot \Phi_{1,0} \cdot \delta X_0 \\ &= [\mathbf{n}, \mathbf{0}_{1 \times 3}] \cdot \begin{bmatrix} \mathbf{I}_{3 \times 3} & \Delta T \cdot \mathbf{I}_{3 \times 3} \\ \Delta T \cdot \mathbf{I}_{3 \times 3} \cdot \mathbf{S}_0 & \mathbf{I}_{3 \times 3} \end{bmatrix} \cdot \delta X_0 \\ &= \mathbf{n} \cdot \delta \mathbf{r}_0 \end{aligned} \quad (7)$$

where $\delta \mathbf{r}_0$ is the initial positional error at time t_0 .

Similarly, the specific expressions positional error at time t_2 , t_3 and t_4 are obtained, as shown in Equations (8)–(10)

$$\delta \mathbf{r}_2^r = \mathbf{n} \cdot (\mathbf{I}_{3 \times 3} + \Delta T^2 \mathbf{S}_0) \cdot \delta \mathbf{r}_0 \quad (8)$$

$$\delta \mathbf{r}_3^r = \mathbf{n} \cdot [\mathbf{I}_{3 \times 3} + (2\mathbf{S}_0 + \mathbf{S}_1)\Delta T^2] \cdot \delta \mathbf{r}_0 \quad (9)$$

$$\delta \mathbf{r}_4^r = \mathbf{n} \cdot [\mathbf{I}_{3 \times 3} + \Delta T^2(3\mathbf{S}_0 + 2\mathbf{S}_1 + \mathbf{S}_2) + \mathbf{S}_0 \mathbf{S}_2 \Delta T^4] \cdot \delta \mathbf{r}_0. \quad (10)$$

According to Equations (7)–(10), we can obtain a system of equations. The observability matrix, $\mathbf{O}$, is expressed as follows:

$$\mathbf{O} = \begin{bmatrix} \mathbf{n} \\ \mathbf{n} \cdot (\mathbf{I}_{3 \times 3} + \Delta T^2 \mathbf{S}_0) \\ \mathbf{n} \cdot [\mathbf{I}_{3 \times 3} + (2\mathbf{S}_0 + \mathbf{S}_1)\Delta T^2] \\ \mathbf{n} \cdot [\mathbf{I}_{3 \times 3} + \Delta T^2(3\mathbf{S}_0 + 2\mathbf{S}_1 + \mathbf{S}_2) + \mathbf{S}_0 \mathbf{S}_2 \Delta T^4] \end{bmatrix} \quad (11)$$

$$\leftrightarrow \begin{bmatrix} \mathbf{n} \\ \mathbf{n} \cdot \Delta T^2 \mathbf{S}_0 \\ \mathbf{n} \cdot \Delta T^2 \mathbf{S}_1 \\ \mathbf{n} \cdot (\Delta T^2 \mathbf{S}_2 + \mathbf{S}_0 \mathbf{S}_2 \Delta T^4) \end{bmatrix}$$

where the value of $\mathbf{S}_0$, $\mathbf{S}_1$, and $\mathbf{S}_2$ are not strongly correlated. The singular value decomposition of the observability is as follows:

$$\mathbf{O} = \mathbf{U} \Sigma \mathbf{V}^T \quad (12)$$

where $\mathbf{U}$ and $\mathbf{V}$ represent the left and right singular matrices, respectively. The orbital inclination relative to the LOS of a

pulsar (OILOS) is defined as the angle between the orbital plane and the pulsar LOS. Figure 1 shows the singular values.

The maximum and middle singular values are more significant than the minimum singular value on the orbit. A larger singular value indicates better observability of the position on the corresponding singular direction, and vice versa. Therefore, we estimate the position along the two directions of maximum and middle singular values. In contrast, the position along the direction of the minimum singular value is discarded, which means giving up the positioning in the non-sensitive direction.

2.2. TPO Method

As the grid method fitting the “ridge” model needs N^2 sampling points, its computational load is high. We establish a chi-square increment model, $\Delta \chi^2$, with respect to initial positional error, $\delta \mathbf{r}_0$, and based on this, propose a TPO method. Next, to simplify the problem, we neglect noise interference and consider only the distortion of the pulsar profile.

In the “ridge” model, as shown in Figure 2, the sub-sensitive direction divides the “ridge” model into two planes. When $\delta \mathbf{r}_0$ is small, $\chi^2(\delta \mathbf{r}_0) - \chi^2(\mathbf{o}) = \Delta \chi^2$, where, $\chi^2(\mathbf{o})$ is the chi-square value for zero distortion. Its directly proportional to $\delta \mathbf{r}_0$. Suppose that the sub-sensitive direction error be $\delta \mathbf{r}_N^0$, and the error on the sensitive direction be $\delta \mathbf{r}_I^0$. So $\delta \mathbf{r}_0 = \delta \mathbf{r}_N^0 + \delta \mathbf{r}_I^0$

$$\Delta \chi^2(\delta \mathbf{r}_0) = \Delta \chi^2(\delta \mathbf{r}_N^0 + \delta \mathbf{r}_I^0) \approx \Delta \chi^2(\delta \mathbf{r}_I^0). \quad (13)$$

The line formed by $\delta \mathbf{r}_N^0 + \delta \mathbf{r}_I^0$ and $\delta \mathbf{r}_I^0$ is parallel to $\delta \mathbf{r}_N^0$.

Next, we propose the TPO method. The specific steps are as follows:

We take three points from the right side of the plane. It is required that the three points must be on the same side. Set $\Delta \chi^2(\delta \mathbf{r}_H)$, $\Delta \chi^2(\delta \mathbf{r}_M)$, $\Delta \chi^2(\delta \mathbf{r}_N)$ be the chi-square values of points $\delta \mathbf{r}_H$, $\delta \mathbf{r}_M$, $\delta \mathbf{r}_N$ respectively. Assume that $\Delta \chi^2(\delta \mathbf{r}_M) < \Delta \chi^2(\delta \mathbf{r}_N)$, there exists $\delta \mathbf{r}_{M'}$ that make $\Delta \chi^2(\delta \mathbf{r}_M) = \Delta \chi^2(\delta \mathbf{r}_{M'})$ holds. We can get $\mathbf{n}_{\text{sub}}$ is parallel to the direction of $\delta \mathbf{r}_{MM'}$ formed by $\delta \mathbf{r}_M$ and $\delta \mathbf{r}_{M'}$. Its expression is shown in Equation (14)

$$\delta \mathbf{r}_{MM'} // \mathbf{n}_{\text{sub}}. \quad (14)$$

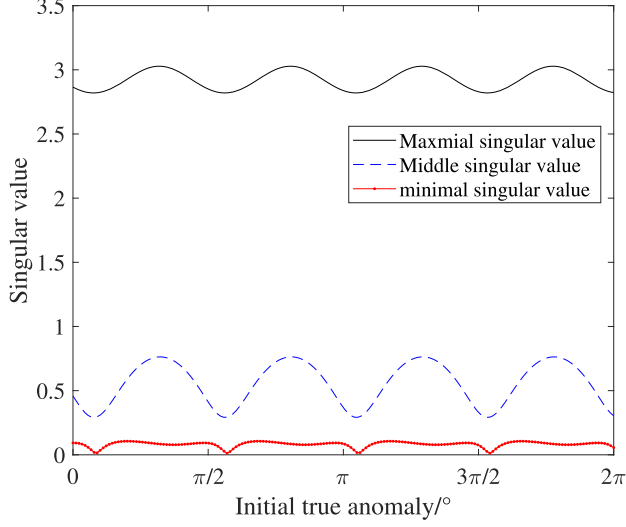
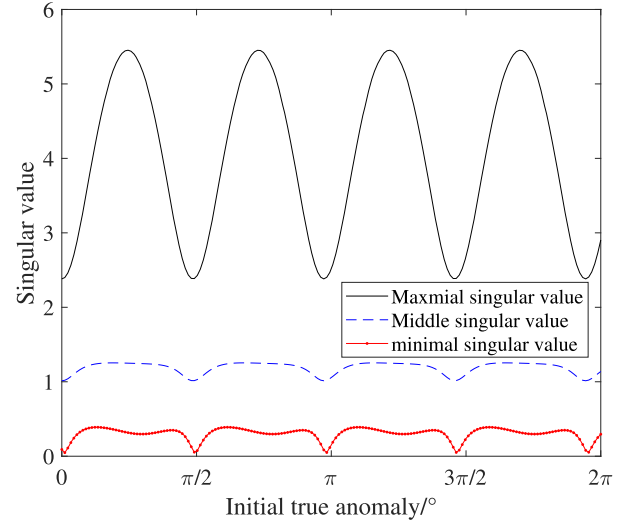
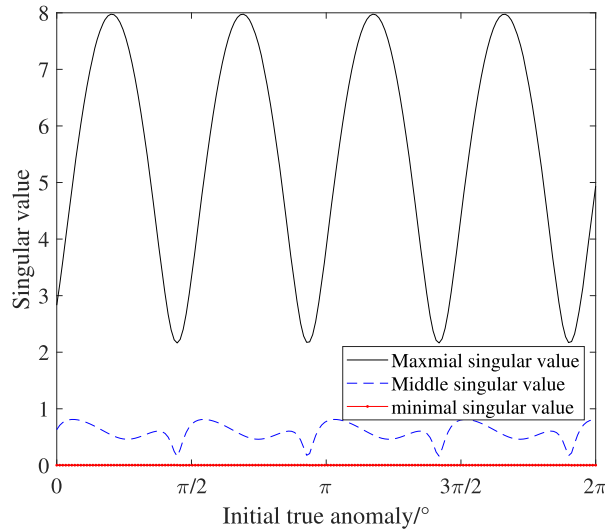
According to the bi-interpolation method, the position of a point $\delta \mathbf{r}_{M'}$ can be estimated

$$\frac{\delta \mathbf{r}_N - \delta \mathbf{r}_{M'}}{|\delta \mathbf{r}_N|} = \frac{\Delta \chi^2(\delta \mathbf{r}_N) - \Delta \chi^2(\delta \mathbf{r}_{M'})}{\Delta \chi^2(\delta \mathbf{r}_N) - \Delta \chi^2(\delta \mathbf{r}_H)}. \quad (15)$$

The direction of $\delta \mathbf{r}_{MM'}$ can be determined by $\delta \mathbf{r}_M$ and $\delta \mathbf{r}_{M'}$. $\delta \mathbf{r}_{MM'}$ is the sub-sensitive direction $\mathbf{n}_{\text{sub}}$.

3. Acceleration Algorithm Based on Multiple Optimization Algorithms

The computing resources of spacecraft are limited and the computational load of the accumulative pulsar profile is high. Therefore, the computational load of accumulative pulsar

(a) OILOS of 10° (b) OILOS of 40° (c) OILOS of 90° **Figure 1.** Singular values at different OILOS.

profiles must be reduced. The grid method requires a high computational load. We use the LM algorithm, PSO, and NRBO to reduce the computational load.

First, we use coarse search and the LM algorithm or PSO to fast search for the non-sensitive direction. Then, we use the TPO to search for sensitive and sub-sensitive directions on the plane perpendicular to the non-sensitive direction. Finally, we perform two-dimensional positioning in both directions with NRBO. In this section, we introduce the FTPCO in the order of the flow shown in Figure 3.

3.1. Fast Search Method for Non-sensitive Direction

Although sampling points in the grid method are further reduced, there is still room for reduction. To further reduce computational load, we initially establish an objective function based on the positional error as the coarse search method. Then, we utilize the chi-square value of the distortion of the pulsar profile as the objective function. A fine search method is required as the chi-square value serves as the objective function. As the fine search method employs the grid method, the computational load of the grid method is significant. Hence,

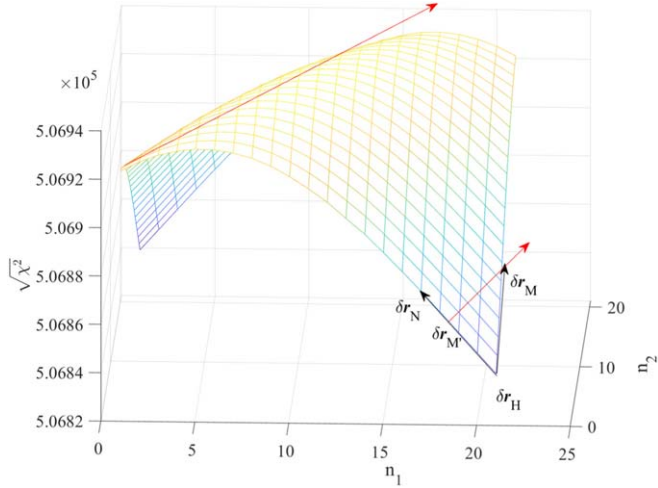


Figure 2. Schematic diagram of the TPO method.

we employ the LM algorithm or PSO substitute to the grid method for fine search. Since the chi-square surface is single-peak or multi-peak problems, the LM algorithm and PSO are employed for single-peak and multi-peak problems, respectively.

3.1.1. Coarse Search for Non-sensitive Direction

The distortion of the accumulative pulsar profile is related to the orbital error. We use the orbital error to make a prediction for the accumulative pulsar profile. Due to the avoidance of the accumulation of pulsar profile, the computational load significantly decreases.

The specific steps are as follows:

Step 1: Establish the orbital error model.

Step 1.1: Establish the positional error model as follows:

$$\Delta \mathbf{r}_n(\theta_i, \varphi_j) = (\cos \theta_i \cos \varphi_j \cdot \mathbf{n} + \cos \theta_i \sin \varphi_j \cdot \mathbf{n}_1 + \sin \theta_i \cdot \mathbf{n}_2) \cdot \delta r_c \quad (16)$$

where $i = 0, 1, 2, \dots, I-1$, $j = 0, 1, 2, \dots, J-1$. $\theta_i = i\pi/I$, $\varphi_j = 2\pi j/J$. δr_c is the search step for coarse search.

Step 1.2: Calculate the orbital recursive error. The difference between the error-containing state and the predicted orbital state is the orbital recursive error $s(\theta_i, \varphi_j, t)$, whose expression can be given as:

$$s(\theta_i, \varphi_j, t) = F(\mathbf{x} + [\Delta \mathbf{r}_n(\theta_i, \varphi_j), 0_{1 \times 3}], t) - F(\mathbf{x}, t) \quad (17)$$

where F is the orbital prediction model and $\mathbf{x}$ is the initial state.

Step 1.3: Design the objective function. The orbital error and the accumulative pulsar profile have strong correlations. Therefore, we can design the objective function according to the orbital error. In the coarse method, the objective function with respect to orbital error, $f^o(\theta_i, \varphi_j)$, is the integral value of the absolute value of the orbital error. Its expression shown as

Equation (18):

$$f^o(\theta_i, \varphi_j) = \int_0^T |s(\theta_i, \varphi_j, t)| dt \quad (18)$$

where T is the observation time.

Step 2: Grid search for θ_i, φ_j . Find θ_i, φ_j that minimizes the objective function with respect to orbital error $f^o(\theta_i, \varphi_j)$

$$[\theta^c, \varphi^c] = \underset{\theta, \varphi}{\operatorname{argmin}} \int_0^T |s(\theta_i, \varphi_j, t)| dt \quad (19)$$

where θ^c, φ^c is the solution of Equation (19).

3.1.2. LM-based Fine Search for Non-sensitive

The computational load of coarse search is low, but we cannot obtain the optimal solution. Therefore, the LM algorithm is introduced to further solve the optimal value problem.

The specific steps are as follows:

Step 1: Initialize the parameters. θ, φ is the R.A. and decl. It is incorporated into the trust-region framework, which is as follows:

$$(J^T J + \lambda I) * P^{\text{LM}} = -J^T f^p(\theta_i, \varphi_j) \quad (20)$$

$$\lambda(\Delta - \|P^{\text{LM}}\|) = 0 \quad (21)$$

where λ is the damping parameter, Δ is the trust radius, $\Delta > 0$, and J is the Jacobian of $f^p(\theta_i, \varphi_j)$. $f^p(\theta_i, \varphi_j)$ is the objective function with respect to the distortion of the pulsar profile.

Step 2: Establish the objective function. Establishment of the objective function based on the accumulative pulsar profile.

Step 2.1: Establish the positional error model as follows:

$$\Delta \mathbf{r}_n(\theta_i, \varphi_j) = (\cos \theta_i \cos \varphi_j \cdot \mathbf{n} + \cos \theta_i \sin \varphi_j \cdot \mathbf{n}_1 + \sin \theta_i \cdot \mathbf{n}_2) \cdot \delta r_f \quad (22)$$

where, $i = 0, 1, 2, \dots, I-1$, $j = 0, 1, 2, \dots, J-1$. $\theta_i = i\pi/I$, $\varphi_j = 2\pi j/J$. δr_f is the search step for fine search.

Step 2.2: Calculate the orbital recursive error. We refer to the difference between the error-containing orbital state and the predicted orbital state as the orbital recursive error $s(\theta_i, \varphi_j, t)$, whose expression can be given as:

$$s(\theta_i, \varphi_j, t) = F(\mathbf{x} + [\Delta \mathbf{r}_n(\theta_i, \varphi_j), 0_{1 \times 3}], t) - F(\mathbf{x}, t) \quad (23)$$

where F is the orbital prediction model and $\mathbf{x}$ is the initial state.

Step 2.3: Estimate phase shift. Divide the accumulation time T_{obs} of the pulsar into M segments. Set the initial orbital error be $s(\theta_i, \varphi_j, 0)$. Based on the orbital dynamics model, the orbital error at time $m T_{\text{obs}}/M$ ($m = 0, 1, 2, \dots, M-1$) can be estimated to be $s(\theta_i, \varphi_j, m)$. The phase shift of the pulsar ε_m can be given as Equation (24)

$$\varepsilon(\theta_i, \varphi_j, m) = n \cdot s(\theta_i, \varphi_j, m) / (c \cdot T_0). \quad (24)$$

where, T_0 is the pulsar intrinsic period.

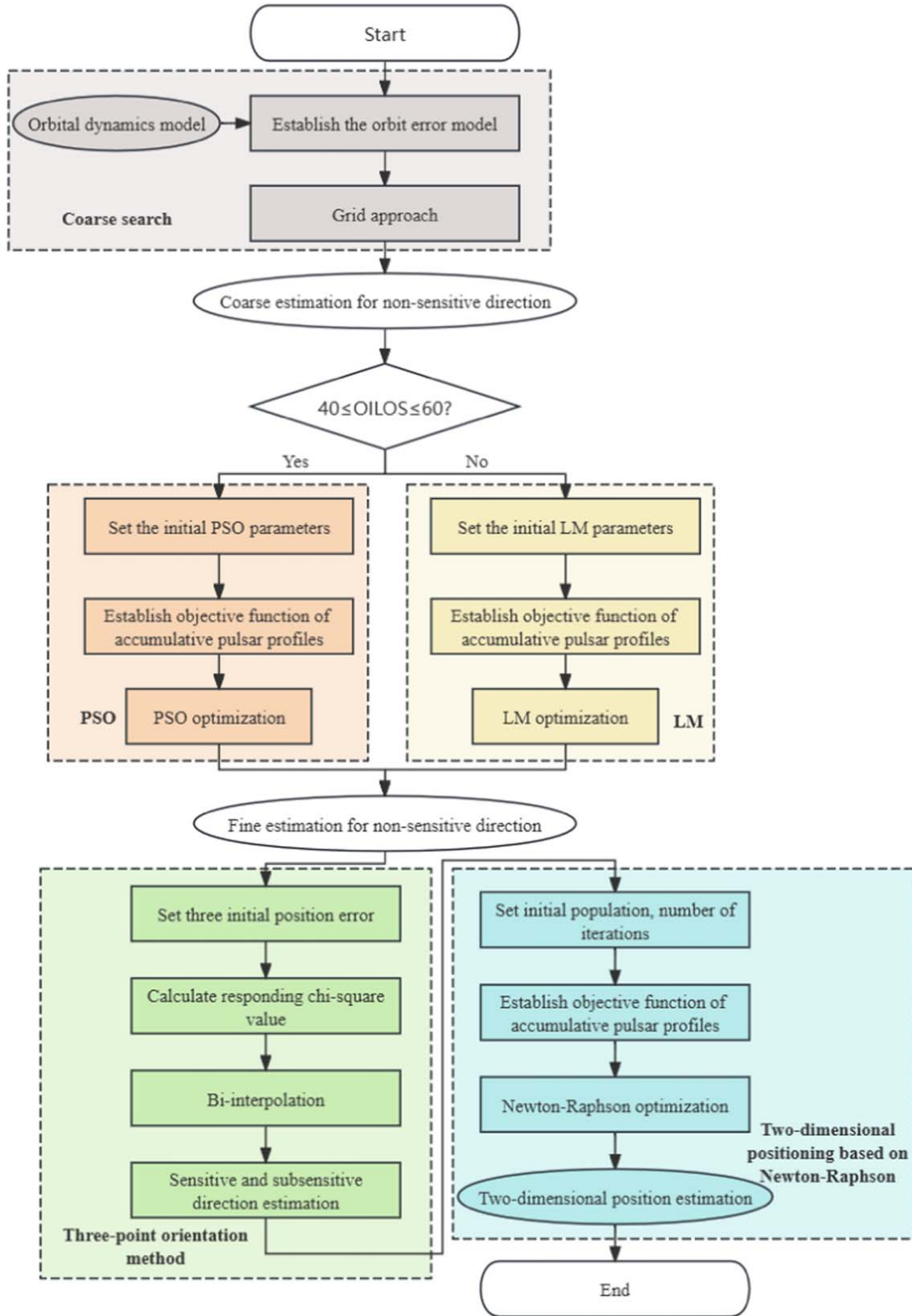


Figure 3. Singular values at different OILOS.

Step 2.4: Accumulate profile. Set the accumulative pulsar sub-profile within the time interval $m T_{\text{obs}}/M \sim (m+1) T_{\text{obs}}/M$ to be $h_m(\varepsilon)$, where $m = 0, 1, 2, \dots, M-1$. The accumulative pulsar profile $\tilde{h}(\tilde{\varepsilon}, \theta_i, \varphi_j)$ is the superposition of the above accumulative sub-profiles. It can be represented as:

$$\tilde{h}(\tilde{\varepsilon}, \theta_i, \varphi_j) = \frac{1}{M} \sum_{m=0}^{M-1} h(\varepsilon + \varepsilon(\theta_i, \varphi_j, m)). \quad (25)$$

Step 2.5: Calculate the χ^2 value of the accumulative pulsar profile $\tilde{h}(\tilde{\varepsilon}, \theta_i, \varphi_j)$. The chi-square value is shown in Equation (26):

$$f^p(\theta_i, \varphi_j) = \chi^2(\theta_i, \varphi_j) = \int_0^1 \left(\tilde{h}(\tilde{\varepsilon}, \theta_i, \varphi_j) - \int_0^1 \tilde{h}(\tilde{\varepsilon}, \theta_i, \varphi_j) d\tilde{\varepsilon} \right)^2 d\tilde{\varepsilon} \quad (26)$$

Step 3: LM optimization. Finally, we can obtain the optimal value

$$[\theta^f, \varphi^f] = \underset{\theta, \varphi}{\operatorname{argmax}} [f^p(\theta_i, \varphi_j)]. \quad (27)$$

At this time, the non-sensitive direction value is the optimal solution, which the non-sensitive direction $\mathbf{n}_{\text{non}}$ can be obtained.

3.1.3. PSO-based Fine Search for Non-sensitive

Due to the multi-peak problem in some orbits, the LM algorithm cannot accurately search for the optimal value. It is necessary to use the PSO to substitute the LM algorithm for optimization. The PSO is a swarm intelligence algorithm, which converges to the optimal solution faster than the genetic algorithm for high-dimensional optimization problems. It has excellent optimization performance for the objective function.

The specific steps are as follows:

Step 1: Set the initial population and number of iterations.

Step 2: Construct the objective function. The objective function is the same as in Section 3.1.2.

Step 3: PSO optimization. Optimized with PSO, and θ^f, φ^f obtained. The non-sensitive direction expression is:

$$\mathbf{n}_{\text{non}} = [\cos \theta \cos \varphi \cdot \mathbf{n} + \cos \theta \sin \varphi \cdot \mathbf{n}_1 + \sin \theta \cdot \mathbf{n}_2]. \quad (28)$$

3.1.4. Coarse-to-fine Search

We combine coarse search and the LM algorithm or PSO, which can greatly reduce the computational load. The specific steps are as follows:

Step 1: Coarse search. According to the coarse search method provided in Section 3.1.1, we can search for θ^c, φ^c .

Step 2: Use θ^c, φ^c obtained from the coarse search as the initial value for the LM optimization algorithm or the PSO.

Step 3: According to Sections 3.1.2 or 3.1.3, the optimal value, θ^f, φ^f , can be found through the optimization algorithm without global search, which reduces the number of

computations of the objective function. When the LM algorithm or the PSO is combined with the coarse search method, we can obtain non-sensitive directions with high accuracy. Therefore, two-dimensional positioning can be performed in two mutually perpendicular directions on the plane where the non-sensitive direction, results in high-precision positioning.

3.2. TPO Method

We carry out the positioning on the sensitive plane perpendicular to the non-sensitive direction, which makes the selection of direction particularly important. Random choice of the search direction makes the optimal value not easily obtain. Therefore, we use the TPO method to search the sensitive direction and sub-sensitive direction.

The specific steps are as follows:

Step 1: Set the initial positional error for three points. On the given plane, randomly select two perpendicular directions and establish the initial positional error model $\Delta \mathbf{r}_{k,l}$

$$\Delta \mathbf{r}_{k,l} = \left(k - \frac{N+1}{2} \right) \delta r_l \cdot \mathbf{n}_1^{\text{non}} + \left(l - \frac{N+1}{2} \right) \delta r_l \cdot \mathbf{n}_2^{\text{non}} \quad (29)$$

where δr_l is the search step for TPO method; N is the number of sampling in the direction; k and l is the variables, $k = 1, 2, \dots, N$, $l = 1, 2, \dots, N$. Only three points need to be sampled.

Step 2: Calculate the corresponding chi-square value.

Step 2.1: Generate the distortion model. The initial positional error model $\Delta \mathbf{r}_{k,l}$ is substituted into the orbital dynamics model to obtain the radial displacement compensation $\Delta \mathbf{r}_m^r$ at different times, $m = 0, 1, 2, \dots, M-1$. A distortion model $g_{k,l}(\Delta \mathbf{r}^r)$ is built based on the probability density of the displacement compensation.

Step 2.2: Accumulate the distortion of the pulsar profile. The distortion of the pulsar profile $\tilde{h}_{k,l}(\varepsilon)$ can be considered as the cyclic cross-correlation of the pulsar standard profile and the distortion template

$$\tilde{h}_{k,l}(\varepsilon) = h(\varepsilon)^* g_{k,l}(\Delta \mathbf{r}^r). \quad (30)$$

Step 2.3: Calculate the chi-square value of the distortion of the pulsar profile.

Step 3: Bi-interpolation method. Through the theoretical analysis of Section 2.2, the sub-sensitive direction $\mathbf{n}_{\text{sub}}$ can be obtained.

Step 4: Estimate sensitive direction. The sensitive direction $\mathbf{n}_{\text{sen}}$ can be calculated by the following formula:

$$\mathbf{n}_{\text{sen}} = \mathbf{n}_{\text{sub}} \times \mathbf{n}_{\text{non}} \quad (31)$$

where $\times$ represents the cross product.

3.3. Two-dimensional Positioning Method Based on NRBO

In Section 3.2, two directions can be obtained through the TPO method. The two-dimensional positioning in these two directions can improve positioning accuracy. However, the computational load of the two-dimensional grid positioning method is high, so we use NRBO for two-dimensional positioning. The NRBO can reduce the computational load while ensuring accuracy, which greatly saves the computing resource of star-borne.

The specific steps are as follows:

Step 1: Set the initial population size and the number of iterations.

Step 2: Establish the objective function.

Step 2.1: Two-dimensional initial positional error. The two directions $\mathbf{n}_{\text{sub}}$ and $\mathbf{n}_{\text{sen}}$ obtained by the TPO method are used as the positional direction, then the initial positional error is:

$$\Delta \mathbf{r}_{k,l}^n = \left(k - \frac{N+1}{2} \right) \delta r_{\text{NRBO}} \cdot \mathbf{n}_{\text{sub}} + \left(l - \frac{N+1}{2} \right) \delta r_{\text{NRBO}} \cdot \mathbf{n}_{\text{sen}} \quad (32)$$

where δr_{NRBO} is the search step for two-dimensional positioning; N is the number of sampling in the direction; k and l is the variables, $k = 1, 2, \dots, N$, $l = 1, 2, \dots, N$. The total sampling points are determined by the initial population and the number of iterations.

Step 2.2: Accumulate the pulsar profile. The phase shift estimation is stored in the dictionary. Suppose the k th phase shift vector $\varepsilon_{k,l}^m$ matches $\varepsilon_{k,l}^m$ in the k th group of the relative phase shift dictionary. We choose the k th accumulative sub-profiles from the k th group of the segmented complete accumulative profile dictionary. Set $h(\varepsilon_{k,l}^m)$ be the accumulative pulsar sub-profile in the period $m T_{\text{obs}}/M \sim (m+1) T_{\text{obs}}/M$, where $m = 0, 1, 2, \dots, M-1$. The accumulative pulsar profile $\tilde{h}_{k,l}(\tilde{\varepsilon})$ is the overlay of the above accumulative pulsar sub-profile, which can be expressed as:

$$\tilde{h}_{k,l}(\tilde{\varepsilon}) = \frac{1}{M} \sum_{m=0}^{M-1} h(\varepsilon + \varepsilon_{k,l}^m). \quad (33)$$

Step 2.3: Calculate the chi-square value. Calculate the chi-square value of the accumulative pulsar profile $\tilde{h}_{k,l}(\tilde{\varepsilon})$. Select the largest chi-square value in the population

$$\chi_{k,l}^2 = \int_0^1 \left(\tilde{h}_{k,l}(\tilde{\varepsilon}) - \int_0^1 \tilde{h}_{k,l}(\tilde{\varepsilon}) d\tilde{\varepsilon} \right)^2 d\tilde{\varepsilon}. \quad (34)$$

Step 3: NRBO optimization. With NRBO optimization, we are able to obtain the optimal chi-square value

$$\chi_{k+1,l+1}^2 = \max(\chi_{k,l}^2). \quad (35)$$

Step 4: Estimate two-dimensional positioning. Update the iteration to get the optimal value. This enables the estimation of two-dimensional positioning accuracy.

The NRBO greatly satisfies the computational load requirement and positioning accuracy. Therefore, the FTPCO can satisfy star-borne needs.

4. Computational Complexity Analysis

For the search direction, we use both the LM algorithm and the PSO. We mainly use the LM algorithm. Therefore, we only discuss the computational complexity of the LM algorithm. As the LM algorithm is a trust region method, it is distinct from our traditional methods, thus we only consider each iteration number. The computational load of the objective function mainly arises from the accumulative profiles. The computational load for each accumulative profile is D Multiplications and Accumulations Computation (MAC). In the context of the fine search needs as well as N^2 sampling points at an interval of 0.1 the grid method. The fine search reduces sampling points. When the compression rate is δ , the computational load required for the accumulative profile of the grid method is $\delta N^2 D$ MAC. The LM algorithm only needs the computation of chi-square values of the accumulative profile G time. Thus the computational load is GD MAC. Typically, given an initial position by the coarse search, the LM algorithm can fast iterate to find the optimal value, hence $G \ll \delta N^2$. For the two-dimensional positioning, we use the NRBO. Thus we analyze the computational complexity of the NRBO. The NRBO is a swarm intelligence optimization algorithm. For simple objective functions, it can iterate fast to find the optimal value. However, the objective function in this context is more complex, so we only need to consider the computational load of the objective function. The computational load of the NRBO depends on the initial population size and the number of iterations. Assume that the initial population size is P and the number of iterations is E . The computational load for the NRBO is $PE D$ MAC. The computational load of the grid method can be expressed as $\delta N^2 D$ MAC. Usually, $PE < \delta N^2$. For instance, give $\delta = 0.02$, $N = 210$, and $G = 16$. The computational load for the accumulative profile is 3.3×10^7 MAC. The computational load of the LM algorithm and the grid method is 5.3×10^8 MAC and 2.9×10^{10} MAC, respectively, which shows a computational load reduction of 98%. For the two-dimensional positioning, we have $\delta \approx 0.3$, $N = 21$, $P = 10$, and $E = 10$. The computational load of the NRBO and the grid method is 3.3×10^9 MAC and 4.4×10^9 MAC respectively, which reduces the computational load by 25%.

We use FTPCO to decrease the computational load by approximately 71%, which fully illustrates the feasibility of employing multiple optimization algorithms in XPNV.

5. Simulation Results and Analysis

In this section, we investigate the performance of the FTPCO. First, we compare the optimization algorithm with the

grid method to search the non-sensitive direction, which illustrates the superiority of the LM algorithm and PSO in searching for the non-sensitive direction. Then, we use the TPO method on the plane perpendicular to the non-sensitive direction to search for the sensitive and sub-sensitive directions. Finally, the feasibility of NRBO is verified.

5.1. Simulation Conditions

The navigation pulsar is the Crab pulsar PSR B0531+21, and its basic parameters are shown in Table 1. The six elements of the Earth satellite orbit are shown in Table 2. The initial parameters of the algorithms used at each stage are shown in Table 3. The simulation platform is an HP laptop with an Intel Core i5-8300H @2.30 GHz processor and 8G memory.

5.2. Singular Values and Positioning Accuracy

In this subsection, we investigate the relationship between the singular value and the positioning accuracy. From Table 4, we can see that when the positioning in the sensitive direction, the sub-sensitive direction and the non-sensitive direction, the positioning accuracy of the three directions is directly proportional to the singular value demonstrated in Figure 1, which satisfies the theoretical derivation in Section 2.1. When the singular value increases, the observability improves, which leads to improved positioning accuracy. Therefore, we select to perform the two-dimensional positioning in the sensitive and sub-sensitive directions and abandon the non-sensitive directions.

5.3. Comparison of the Grid Method and LM or PSO

In this subsection, we compare the grid method and the LM algorithm for OILOS. As shown in Figure 4, the differences in R.A. and decl. obtained from the grid method and the LM algorithm are presented. When the decl. is at 0° or 180° , the R.A. does not affect the position. We can see from the figure, their difference is approximately zero. When OILOS is 70° and 80° , the change in R.A. does not affect the position. Figure 5 shows the chi-square values corresponding to different OILOS, consistent with the analysis results of singular values in theoretical Section 2.1. The computational load of coarse search is very low and can be ignored. Table 5 shows a comparison between the grid method and the LM algorithm, from which we can see that the computational time required for the LM algorithm is far shorter than that for the grid method. The reason is that sampling points for the LM algorithm are far less than those for the grid method. Hence, the computational load is lower. When OILOS is at 40° , 50° , and 60° , the computational load is very low to use the LM algorithm, but the error is large. The reason is the presence of too many peak values, which is a multi-peak problem. It is easy to fall into local optimization. Therefore we consider the PSO.

Table 1
Simulation Parameters of the Pulsar Observation

Parameter Name	Value
Pulsar period/(ms)	33
Photon flux of pulsar/(ph cm ⁻² s ⁻¹)	1.54
Background noise flux density/(ph cm ⁻² s ⁻¹)	0.005
Number of pulsar intervals	330,000
Detector area/(m ²)	1
Time resolution/(μs)	0.1
Observation time/s	2600
Number of observation time segments	100
Number of sampling at positions n_{sub} and n_{sen}	21
positional sampling interval on n_{sub} and n_{sen} /m	300

Table 2
Orbital Parameters

Orbital Elements	Value
Semimajor axis/(km)	6500
Eccentricity	0
Inclination/(°)	120
Argument of perigee/(°)	0
The R.A. of ascending node/(°)	0
True anomaly/(°)	0

Table 3
Initial Parameters of Three Optimization Algorithms

Algorithm	Number of Initial Populations	Maximum Number of Iterations
LM	1	100
PSO	25	10
NRBO	12	12

Table 4
Singular Values and Positioning Accuracy

OILOS	Positioning Accuracy		
	Sensitive Direction (m)	Sub-sensitive Direction (m)	Non-sensitive Direction (m)
10°	232.926	244.203	289.404
40°	241.195	510.502	1253.748
90°	183.777	1396.271	1517.419

As shown in Table 6, compared with the LM algorithm, the PSO has a high computational load and low error. Furthermore, both of them compare to the grid method, with the difference

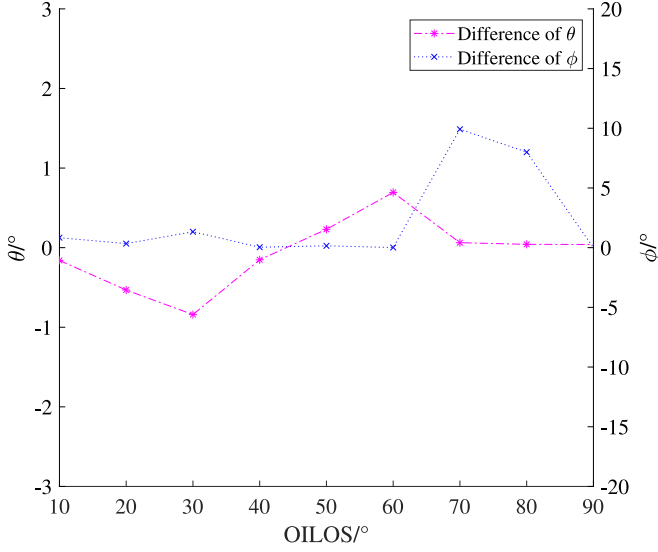


Figure 4. Differences in R.A. and decl. under different OILOS.

between the two fluctuating within the range of 0.3° . The computational load decreases by 71%–98%, which is consistent with the analysis results of computational complexity in Section 4.

5.4. Comparison of the Grid Method and TPO

We search for sensitive and sub-sensitive directions on the sensitive plane perpendicular to the non-sensitive direction. The grid method requires 441 sampling points to identify the sensitive and sub-sensitive directions, while the TPO method only needs three points. Therefore, we utilize the latter. To highlight the superiority of the TPO method, as shown in Table 7, we conduct a comparison between the grid method and the TPO method. From Table 7, it can be seen that as the OILOS increases, the computational load of the grid method exhibits a clear upward trend. We use the dictionary folding method when we use the grid method to find the direction. When the OILOS is small, there are repeated calculations in the phase shift dictionary. The dictionary folding method works to eliminate these repeated calculations, which results in a low computational load. As the OILOS increases, the repeated counting in the phase-shifted dictionary decreases, which makes dictionary folding ineffective and therefore the computational load is high. The grid method presents a significant deficiency in the computational load, especially for large OILOS, with the computational load vastly exceeding that of the TPO method. The reason is that sampling points in the TPO method remain at 3, far less than that of the grid method. As a result, the computational load of the TPO method is reduced by 98% compared to the grid method. Moreover, the directional difference between the sensitive directions obtained via the grid method and the TPO method remains within 2.5° . Compared with the grid method, the TPO method ensures accuracy and greatly reduces the computational load.

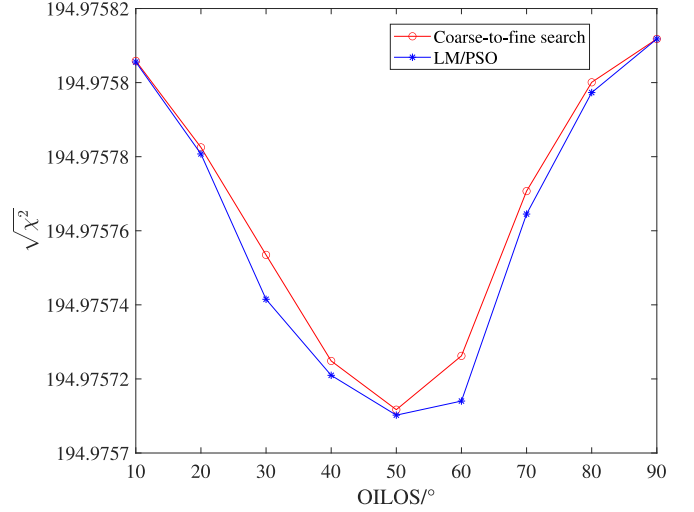


Figure 5. Chi-square values under different OILOS.

Table 5
LM Algorithm vs. Grid Method

OILOS	Grid Method		LM Algorithm		Difference ($^\circ$)
	Fine Sample	Time (s)	Fine Sample	Time (s)	
10 $^\circ$	861	717.799	24	19.571	0.181
20 $^\circ$	861	728.047	24	38.405	0.535
30 $^\circ$	861	743.031	24	46.583	0.875
40 $^\circ$	861	738.201	22	38.641	7.072
50 $^\circ$	861	740.351	20	30.905	7.792
60 $^\circ$	861	735.305	20	26.461	5.846
70 $^\circ$	861	917.824	24	32.139	0.126
80 $^\circ$	861	1106.901	24	26.915	0.096
90 $^\circ$	861	1085.352	24	15.722	0.040

Table 6
PSO vs. Grid Method

OILOS	Grid Method		PSO		Difference ($^\circ$)
	Fine Sample	Time (s)	Fine Sample	Time (s)	
40 $^\circ$	861	738.201	250	210.754	0.153
50 $^\circ$	861	740.351	200	156.739	0.232
60 $^\circ$	861	735.305	200	167.248	0.694

5.5. Comparison of the Grid Method and NRBO

In this subsection, we investigate the impact of the key parameter of the NRBO and the grid method. As the population size and the number of iterations increase, the accuracy improves;

Table 7
TPO vs. Grid Method

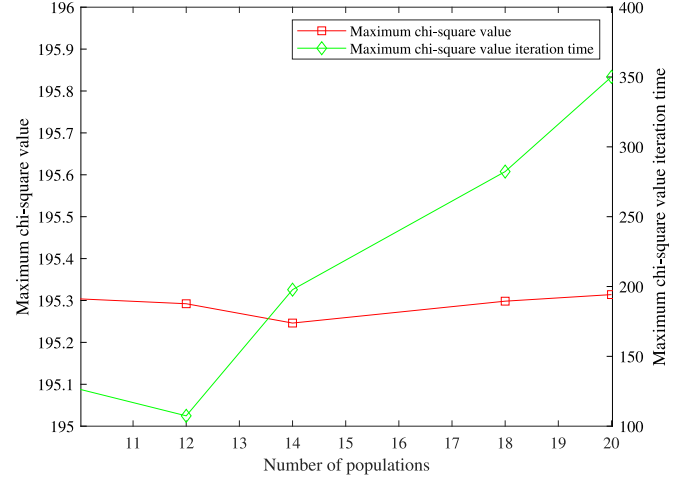
OILOS	Time (s)		Directional Difference (°)
	Grid Method	TPO	
10°	602.743	13.625	1.775
20°	1168.026	29.799	1.495
30°	2245.466	40.626	2.364
40°	2555.887	50.361	2.304
50°	4006.691	51.812	1.107
60°	4226.576	69.446	1.196
70°	4524.115	98.824	0.558
80°	5018.303	105.999	0.421
90°	4368.096	109.921	0.370

however, this results in an increased computational load. Therefore, it is crucial to select an appropriate number of populations and iterations. As shown in Figure 6, when the number of populations is 12, the computational load remains low, and the computational load of the NRBO positioning is approximately equal to that of the grid positioning method. Furthermore, the accuracy can be effectively maintained under these conditions.

Then, we select OILOS of 10°, 40°, and 90° as simulation orbits. As shown in Table 8, we compare the one-dimensional grid positioning method, the two-dimensional grid positioning method and the NRBO positioning method. As shown in Table 8, although the one-dimensional positioning method requires less time, its positioning accuracy is low. The reason is that the positioning accuracy of the non-sensitive direction is low. Consequently, we concentrate our comparison on the two-dimensional grid positioning method and the NRBO positioning method. Within an acceptable range of positioning accuracy, the NRBO positioning algorithm reduces the computational time by 39–53 s. The reason is that the NRBO significantly reduces sampling points compared to the two-dimensional grid positioning method. Furthermore, since the dictionary folding method can only be used in the grid method, this method is dismissed in FTPCO. The NRBO makes positioning simple and efficient. Consequently, NRBO can substitute the grid method for two-dimensional positioning, which thereby reduces the computational load.

5.6. Computational Load

In this section, the computational load is presented, as shown in Table 9. The computational load of FTPCO is reduced by 1000–5000 s compared to the grid method. The above results suggest that the FTPCO effectively enhances real-time performance, due to the significant reduction in sampling points.

**Figure 6.** NRBO parameters.

6. Conclusion

We propose FTPCO to reduce the computational load. First, we theoretically demonstrate the feasibility of optimization algorithms in XPNAV. Then, we substitute the grid method with three optimization algorithms, which offer non-sensitive, sensitive, and sub-sensitive directions for positioning. Finally, we employ the NRBO instead of the grid method for two-dimensional positioning. The main reason for high positioning accuracy is that it is not easy to fall into a local optimum.

FTPCO has the following advantages:

- (1) Real-time performance. Simulation results demonstrate that compared to the traditional grid method, the computational load of the non-sensitive direction search, the sensitive direction search and the two-dimensional positioning, is reduced by 71%–98%, 97.8%, and 24.2%, respectively. To sum up, the computational load of FTPCO has declined by 89.4%.
- (2) High directional accuracy. The directional accuracies of the non-sensitive directions obtained by the LM algorithm and the PSO are 0.4°–0.9°. The directional accuracies of sensitive and sub-sensitive directions obtained by the TPO method are 0.3°–1.7°. Therefore, the FTPCO reduces the computational load significantly and maintains high directional accuracy.
- (3) High positioning accuracy. The NRBO is not easy to fall into the local optimal solution. Its two-dimensional positioning error is approximately 200–1000 m, which meets the requirements of navigation. Compared to the traditional methods, the positioning accuracy of the NRBO is improved by 38% due to high direction accuracy.

Table 8
NRBO Positioning vs. Grid Method

OILOS	Two-dimensional Positioning						Time Difference (s)
	One-dimensional Positioning						
			Grid Positioning Method		NRBO Positioning Algorithm		
	Positioning Accuracy (m)	Time (s)	Positioning Accuracy (m)	Time (s)	Positioning Accuracy (m)	Time (s)	
10°	1891.022	43.530	278.933	164.107	278.783	124.469	39.638
40°	4424.226	46.229	641.112	188.370	601.791	134.619	53.752
90°	2705.880	50.081	1086.750	197.171	1110.614	153.806	43.365

Table 9
Computational Load

Time (s)									
OILOS	Grid Method				FTPCO				Promotion Rate
	Non-sensitive Direction	Sensitive Direction	Two-dimensional Positioning	Total Time	LM/PSO	TPO	NRBO		(%)
							Positioning	Total Time	
10°	717.799	602.743	164.107	1484.649	19.571	13.625	124.469	157.665	0.894
40°	738.201	2555.887	188.370	3482.458	38.641	50.361	134.619	223.621	0.936
90°	1085.352	4368.096	197.171	5650.619	15.722	109.921	153.806	279.449	0.951

The FTPCO provides new ideas and references for the design of future XPNV systems.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (Nos. 61873196 and 62373030) and the Innovation Program for Quantum Science and Technology (No. 2021ZD0303400).

ORCID iDs

Jin Liu  <https://orcid.org/0000-0002-1098-722X>

References

- Abdel-Basset, M., Mohamed, R., Jameel, M., & Abouhawwash, M. 2023a, *Knowledge-Based Systems*, 262, 110248
- Abdel-Basset, M., Mohamed, R., Jameel, M., & Abouhawwash, M. 2023b, *Artific. Intell. Rev.*, 56, 11675
- Achouri, F., Khatir, A., Smahi, Z., Capozucca, R., & Ouled Brahim, A. 2023, *J. Brazilian Soc. Mech. Sci. Eng.*, 45, 621
- An, K., Gu, D.-F., Shao, K., Song, J.-N., & Zhu, J.-B. 2023, *AcAau*, 202, 186
- Carpintero, D., & Melita, M. 2018, *A&A*, 620, 11
- Gui, M.-Z., Yang, H., Ning, X.-L., et al. 2023, *AdSpR*, 71, 756
- Li, X.-H., & Yu, S.-H. 2023, *JMST*, 28, 832
- Li, Y., Li, J.-X., Sun, Y., & Li, H.-S. 2022, *Biomimetics*, 7, 168
- Liu, J., Fang, J.-C., Wu, J., Kang, Z.-W., & Ning, X.-L. 2014, *IET Radar, Sonar and Navigation*, 8, 1154
- Liu, J., Han, X.-X., Ning, X.-L., Chen, X., & Kang, Z.-W. 2020, *Hangkong Xuebao/Acta Aeronautica et Astronautica Sinica*, 41, 8
- Liu, J., Jia, Y.-Z., Wang, Y.-D., & Ma, X. 2023a, *Yuhang Xuebao/Journal of Astronautics*, 44, 1094
- Liu, J., Wang, Y.-D., Ning, X.-L., & Hu, J.-T. 2023b, *AcAau*, 213, 373
- Liu, J., Yang, Z.-H., Kang, Z.-W., & Chen, X. 2019, *AcAau*, 160, 90
- Liu, J., Zhang, H.-Z., Ning, X.-L., & Ma, X. 2024, *ChJA*, in press
- Ma, X., Xie, T.-H., Li, J.-R., et al. 2024, *AcAau*, 219, 79
- Muller, H., Mus, A., & Lobanov, A. 2023, *A&A*, 675, 13
- Qian, L., & Pan, Z. 2021, *RAA*, 21, 185
- Shen, L.-R., Li, X.-P., Sun, H.-F., Fang, H.-Y., & Xue, M.-F. 2015, *Frontiers of Information Technology and Electronic Engineering*, 16, 858
- Song, M.-Z., Wang, Y.-D., Zheng, W., & Wu, Y.-L. 2022, *AdSpR*, 70, 203
- Sowmya, R., Premkumar, M., & Jangir, P. 2024, *Engineering Applications of Artificial Intelligence*, 128
- Sun, H.-F., Su, J.-Y., Deng, Z.-W., et al. 2023, *ChJA*, 36, 386
- Wang, D.-S., Tan, D.-P., & Liu, L. 2018, *Soft Computing*, 22, 387
- Wang, S., Cui, P.-Y., Gao, A., Yu, Z.-S., & Cao, M.-L. 2017, *AcAau*, 138, 68
- Wang, Y.-C., Li, M.-T., Pan, Z.-C., & Zheng, J.-H. 2019, *RAA*, 19, 133
- Wang, Y.-D., Zhang, S.-N., Ge, M.-Y., et al. 2023a, *ITAES*, 59, 3395
- Wang, Y.-D., & Zhang, W. 2016, *ITAES*, 52, 2210
- Wang, Y.-D., Zheng, W., Zhang, S.-N., et al. 2023b, *ChJA*, 36, 44
- Xu, N., Sheng, L.-Z., Chen, C., et al. 2019, *Optik*, 198, 163253
- Zhang, H., Jiao, R., Lu-Ping, X., Xu, C.-Y., & Mi, P. 2019, *PASP*, 131, 045002
- Zhang, J., Liu, J., Ma, X., Kang, Z.-W., & Wang, Z.-N. 2021, *JAerE*, 34, 998
- Zhang, W.-J., Ma, X., Chen, X., et al. 2024, *ITIM*, 73, 1
- Zhang, X.-Y., Shuai, P., & Huang, L.-W. 2015, *Yuhang Xuebao/Journal of Astronautics*, 36, 1056