Research in Astronomy and Astrophysics, 24:045024 (16pp), 2024 April https://doi.org/10.1088/1674-4527/ad34bb

Accelerating Asteroidal Period and Pole Inversion from Multiple Lightcurves Using Parallel Differential Evolution and Cellinoid Shape Model

Yong-Xiong Zhang[1], Wen-Xiu Guo[2,3], Xiao-Ping Lu[2,4], Hua Zheng[3], Hai-Bin Zhao[5], Jun Tian[5], and Wei-Lin Wang[1]
[1] School of Engineering, Guangzhou College of Technology and Business, Guangzhou 510850, China
[2] School of Computer Science and Engineering, Macau University of Science and Technology, Taipa, Macao 999078, China; xplu@must.edu.mo
[3] School of Mathematics and Statistics, Shaoguan University, Shaoguan 512099, China
[4] State Key Laboratory of Lunar and Planetary Sciences, Macau University of Science and Technology, Macao 999078, China
[5] Key Laboratory of Planetary Sciences, Purple Mountain Observatory, CAS, Nanjing 210023, China
Received 2023 October 18; revised 2024 March 11; accepted 2024 March 15; published 2024 April 24

Abstract

Determining asteroid properties provides valuable physical insights but inverting them from photometric lightcurves remains computationally intensive. This paper presents a new approach that combines a simplified Cellinoid shape model with the Parallel Differential Evolution (PDE) algorithm to accelerate inversion. The PDE algorithm is more efficient than the Differential Evolution algorithm, achieving an extraordinary speedup of 37.983 with 64 workers on multicore CPUs. The PDE algorithm accurately derives period and pole values from simulated data. The analysis of real asteroid lightcurves validates the method's reliability: in comparison with results published elsewhere, the PDE algorithm accurately recovers the rotational periods and, given adequate viewing geometries, closely matches the pole orientations. The PDE approach converges to solutions within 20,000 iterations and under one hour, demonstrating its potential for large-scale data analysis. This work provides a promising new tool for unveiling asteroid physical properties by overcoming key computational bottlenecks.

Key words: methods: numerical – minor planets – asteroids: general – techniques: photometric – astrometry

1. Introduction

Asteroids are important objects for understanding the origin and evolution of the solar system. As small bodies orbiting the Sun, they provide insights into early planetary formation processes and conditions. Additionally, some asteroids pose a potential hazard to Earth through impact events. Characterizing and understanding asteroids, especially those that pass near Earth's orbit, is therefore crucial. Most asteroids reside between the orbits of Mars and Jupiter in the main asteroid belt. However, gravitational perturbations can send some asteroids into the inner solar system, becoming near-Earth asteroids. Key physical properties of asteroids, such as rotational periods, shapes, and pole orientations, can be determined through analysis of photometric lightcurves and radar images. While radar data are limited, lightcurves from optical photometric monitoring are abundant. Deriving accurate asteroid physical parameters from lightcurve data remains an open challenge. Further work to develop efficient lightcurve inversion techniques would significantly advance our understanding of the asteroid population.

The utilization of lightcurve data for studying asteroid shapes was initially met with skepticism in early research endeavors. Despite being the pioneer in employing lightcurve data for asteroid shape inversion, Russell (1906) was skeptical about its ability to construct satisfactory asteroid shape models. However, advancements in observational techniques eventually made the inversion of physical parameters of asteroids using lightcurve data increasingly viable. Surdej & Surdej (1978) implemented Lambert's law and the Lommel–Seeliger law to simulate a synthetic lightcurve from a spinning triaxial ellipsoid. Meanwhile, Karttunen (1989) and Karttunen & Bowell (1989) advanced a method for generating lightcurves using a triaxial ellipsoid anchored on the Lumme–Bowell scattering law Lumme & Bowell (1981a, 1981b). However, the symmetric and regular nature of the ellipsoid resulted in a symmetric synthetic lightcurve, posing challenges for simulating asymmetrical and irregular real asteroids. In response to these limitations, Cellino et al. (1985, 1987, 1989) introduced a new model based on a composite of eight ellipsoidal octants, moving away from the regular triaxial ellipsoid. The model facilitated the generation of lightcurves more indicative of real-world observations. Still, Cellino et al. (1989) did not provide an inversion method for these lightcurves. The model was subsequently named the Cellinoid shape model by Lu et al. (2014), who also provided a detailed derivation proof and inversion method.

The field has made significant strides over the past decades in inverting the physical parameters of asteroids. Lumme et al. (1990) employed the method of spherical harmonics to determine the pole position of an asteroid. A method for calculating asteroid shapes and related parameters was

proposed and validated on a large number of real observed lightcurves by Kaasalainen et al. (1992a), Kaasalainen et al. (1992b), Kaasalainen & Torppa (2001), Kaasalainen et al. (2001), Kaasalainen (2001), Kaasalainen et al. (2002), Kaasalainen et al. (2003), Kaasalainen et al. (2005) and Kaasalainen & Lamberg (2006). Innovative approaches have been introduced to accelerate lightcurve simulation and inversion, as evidenced by Kaasalainen et al. (2012) usage of Lebedev quadratures. Ďurech et al. (2010, 2011) successfully applied Kaasalainen's method to obtain the relevant physical parameters of hundreds of asteroids. Furthermore, Muinonen & Lumme (2015) and Muinonen et al. (2015) utilized the disk-integrated brightness method and the Lommel–Seeliger ellipsoid model to effectively invert physical parameters such as the rotational period and shape of an asteroid from both sparse and dense photometric data. The inversion tests conducted by Cellino et al. (2015) on both simulated and real photometric data using the Muinonen method further contributed to the field. Inversions on asteroids like (6) Hebe and (7) Iris were performed using the Cellinoid model and Hipparcos data by Lu & Ip (2015), Lu et al. (2016, 2017), Lu et al. (2018) and Lu & Jewitt (2019). These studies also advocated the use of the Lebedev quadrature method to accelerate the inversion process. In addition, Muinonen et al. (2020) offered the first Markov chain Monte Carlo method to account for the uncertainties of the spin, shape, and scattering parameter solutions with ellipsoids and general convex shapes. Martikainen et al. (2021) developed a method for deriving reference absolute magnitudes and phase curves from the Gaia data, which allows for comparative studies involving hundreds of asteroids. The study found wide variations in the derived photometric slope values within each assumed Tholen class. Muinonen et al. (2022) provided error models for four classes of lightcurves and used linear or linear-exponential phase functions for phase angles below 50°. Tian et al. (2022) measured the YORP effect of (1685) Toro and (85989) 1999 JD6 by inverting their lightcurves, providing new detections of this effect and insights into the evolution of small asteroids.

As established by Karttunen (1989) and Karttunen & Bowell (1989), the fluctuation in an asteroid's luminosity is primarily governed by its form rather than the scattering principles. Furthermore, as shown by Kaasalainen & Torppa (2001) and Kaasalainen et al. (2001), the physical attributes of asteroids can be inferred from their lightcurves. They utilized inversion techniques that use a "convex hull" to approximate the shape, assuming a universal albedo. As some asteroids are diminutive bodies formed during the early epoch of the solar system and have not undergone frequently collisional evolution, their surfaces remain relatively unaltered. However, Bottke et al. (2005) showed that small asteroids (less than tens of kilometers) are likely multi-generations of collisional fragments rather than primordial planetesimals. The influence of variations in albedo is less pronounced than that of irregular shape changes. Therefore, these lightcurve inversion methodologies are capable of deriving a comprehensive shape model. To streamline the inversion process, the assumption of universal albedo is frequently employed in these techniques. The convex hull may correspond to a triaxial ellipsoidal form, a convex polyhedral shape, or a Cellinoid shape. As highlighted by Lu et al. (2017), when compared to the Cellinoid shape model, the convex polyhedral shape model is excessively intricate. More specifically, this model incorporates more than 50 parameters, necessitating the collection of a substantial volume of observational data to accommodate these parameters. Even when employing methods based on low-degree spherical harmonics models (Muinonen et al. 2020, 2022), around 15 parameters are needed, which is more complex than the Cellinoid model.

Zhang et al. (2023) introduced a parallel acceleration-based three-step reduced voting (TRV) method aimed at expediting the inversion process. This method notably enhances computational efficiency, enabling the rapid determination of asteroid rotational periods. The TRV algorithm is a parallelization scheme of the Levenberg–Marquardt (LM) algorithm, which significantly accelerates the inversion process. However, given that the LM algorithm optimizes locally, additional iterations are necessary to ensure inversion accuracy. To address this limitation, Li et al. (2023) devised a hybrid optimization algorithm that combines the genetic algorithm with the LM algorithm, all based on a Cellinoid shape model. This combination substantially boosts the efficiency of inversion. Initially, the hybrid algorithm employs the genetic algorithm to search the vicinity of the global optimal solution, then the LM algorithm refines the accuracy of the optimal solution. However, despite the significant efficiency improvement, this algorithm is confined to the comparison of serial execution. Collectively, these studies illustrate that evolutionary algorithms are effective for the inversion of such problems, and parallel computing can enhance inversion efficiency. The asteroid package for the shape model used in this work will consider the astrometric tools provided by PyMsOfa (Ji et al. 2023) in the future. This paper introduces a novel Parallel Differential Evolution (PDE) algorithm, developed based on the Cellinoid model. This approach combines the benefits of evolutionary algorithms and parallel computing for improved efficiency and precision.

The inversion process is depicted in Figure 1. The inversion process begins with the collection of asteroid lightcurve data from ground-based telescopic observations. These lightcurves capture the changing brightness of an asteroid over time as its irregular shape rotates. This paper then applies a novel PDE algorithm to rapidly invert these lightcurve data into the asteroid's rotation period and pole orientation. The algorithm maintains a population of candidate solutions that are evolved through iterative mutation, crossover, and selection operators executed in parallel. By evaluating candidates concurrently at

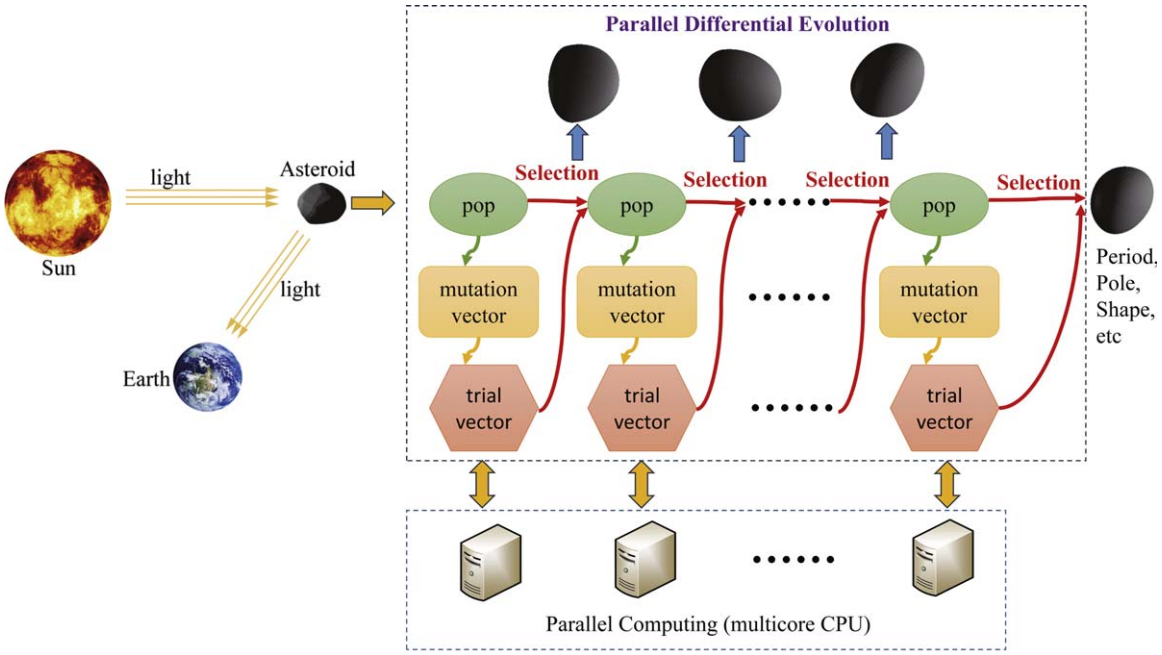

Figure 1. Overview of the proposed Parallel Differential Evolution algorithm.

each generation on a multicore CPU devices, the algorithm significantly accelerates the global search. The improved computational efficiency enables fast determination of the asteroid's physical parameters from large photometric data sets. The PDE algorithm provides an efficient and robust optimization framework for asteroid inversion utilizing multicore CPU parallel computing hardware.

This work primarily focuses on the following key contributions.

1. This paper introduces an innovative PDE algorithm for asteroid lightcurve inversion. The parallel implementation on multicore CPUs leads to significant speed improvements over former serial optimization methods, including the Hybrid Optimization Algorithm (Li et al. 2023), and reduces the iteration count compared to local optimization algorithms designed for parallel computation, such as TRV (Zhang et al. 2023).
2. The PDE algorithm is shown to quickly and accurately determine asteroid rotational periods from lightcurve data. Experimental findings indicate that rotational periods determined through this method exhibit a high degree of concordance with the reference values established in the DAMIT database (Ďurech et al. 2010), demonstrating the effectiveness of the Cellinoid model despite its simplicity. Furthermore, when the observation geometries are favorable, the calculated pole positions are found to be generally consistent with the pole orientations listed in the DAMIT database (Ďurech et al. 2010).
3. The computational efficiency of the PDE algorithm is demonstrated by its ability to converge to solutions within 20,000 iterations and under one hour for all test cases. This marks a significant advancement in the analysis of asteroid photometric data, offering a rapid and reliable method that aligns with reference values from the DAMIT database (Ďurech et al. 2010). Consequently, this algorithm provides a time-efficient solution for managing the increasing volume of asteroid observations.

The rest of this paper is structured as follows: Section 2 describes the Cellinoid shape model, brightness calculation, and the Mean Square Error (MSE). Section 3 provides details on the PDE algorithm, delving into the differential evolutionary methodology and the parallel acceleration scheme. Section 4 offers a performance analysis of the PDE algorithm, validation of the method through synthetic lightcurve analysis, a detailed case study on asteroid (15) Eunomia, and experiments on 16 other real asteroids. This section also includes a discussion on a methodology for inverting observational data that accounts for observational uncertainties. Lastly, Section 5 presents the conclusion of this paper, summarizing the key findings and implications.

2. Methodology

This section provides an in-depth overview of the Cellinoid shape model, a crucial tool for understanding the physical characteristics of asteroids. The focus then shifts to the brightness calculations, essential for interpreting asteroid

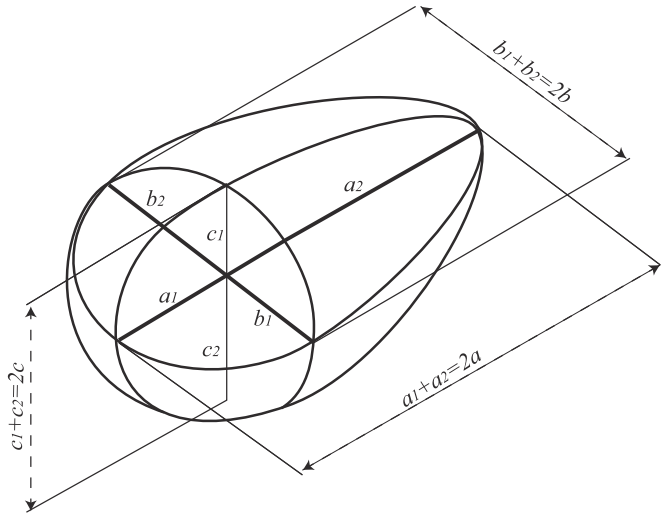

Figure 2. Cellinoid shape model.

lightcurve data. Concluding the section is a comprehensive explanation of the MSE, a statistical measure that provides a quantitative assessment of the accuracy of predictions; it is integral to optimizing model parameters.

2.1. Cellinoid Shape Model

As described by Lu et al. (2014), the Cellinoid model is a geometric construct that effectively represents the shape of an asteroid. This model is characterized as a composite structure, comprised of eight distinct octant surfaces. Each of these surfaces originates from ellipsoids, which are in turn defined by a set of six semiaxes: a_1, a_2, b_1, b_2, c_1, and c_2. This intricate configuration allows the Cellinoid model to account for a variety of asteroid shapes, making it a versatile tool in asteroid studies. A visual representation of this model is provided in Figure 2.

In the 3D coordinates of this model, the volume equation can be derived as Equation (1):

$$V = \frac{\pi}{6}(a_1 + a_2)(b_1 + b_2)(c_1 + c_2), \tag{1}$$

In addition, it is assumed that the volume density ρ is uniformly distributed throughout the Cellinoid shape. Under this assumption, the mass M of the Cellinoid shape can be calculated by multiplying the density over the volume, as shown in Equation (2):

$$M = \rho V, \tag{2}$$

After deriving the expression for the total mass M of the Cellinoid shape using Equation (2), the center of mass position can be calculated. The coordinates $(\bar{x}, \bar{y}, \bar{z})$ of the mass center are given by Equation (3):

$$G(\bar{x}, \bar{y}, \bar{z}): \quad \bar{x} = \frac{3}{8}(a_1 - a_2), \quad \bar{y} = \frac{3}{8}(b_1 - b_2), \quad \bar{z} = \frac{3}{8}(c_1 - c_2). \tag{3}$$

In addition to mass properties, Lu et al. (2014) also analyzed the rotational dynamics of the Cellinoid shape model. They calculated the moment of inertia tensor for the Cellinoid based on its geometry and mass distribution. By diagonalizing this inertia tensor, the principal axes of rotation were determined. The axis corresponding to the maximum moment of inertia was found to be the stable rotational axis for the Cellinoid shape model under free rotational motion. This provides insights into the spin dynamics and rotational stability of asteroids approximated by this Cellinoid shape model.

The matrices A and B are presented by Lu et al. (2014), as depicted in Equation (4).

$$A = \begin{pmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{pmatrix},$$
$$B = \begin{pmatrix} \bar{y}^2 + \bar{z}^2 & -\bar{x}\bar{y} & -\bar{x}\bar{z} \\ -\bar{x}\bar{y} & \bar{x}^2 + \bar{z}^2 & -\bar{y}\bar{z} \\ -\bar{x}\bar{z} & -\bar{y}\bar{z} & \bar{x}^2 + \bar{y}^2 \end{pmatrix}. \tag{4}$$

The inertia tensor matrix $A - MB$ can be diagonalized through an eigendecomposition. This decomposition expresses the tensor as a product of an orthogonal matrix Q, a diagonal matrix Λ, and the transpose Q^T, as shown in Equation (5):

$$A - \mathrm{MB} = Q\Lambda Q^T, \quad \Lambda = \begin{pmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{pmatrix}. \tag{5}$$

where Λ is a diagonal matrix containing the principal moments of inertia as diagonal elements, and Q is an orthogonal matrix whose columns are the principal axes of rotation. The eigendecomposition allows the principal moments and axes to be extracted from the inertia tensor.

The definition of the rotation matrix R, as represented in Equation (6), incorporates the pole coordinates (λ, β) of the asteroid and the rotational phase (θ) within the ecliptic coordinate system.

$$R = \begin{pmatrix} \cos\theta\cos\beta\cos\lambda - \sin\theta\sin\lambda & \cos\theta\cos\beta\sin\lambda + \sin\theta\cos\lambda & -\cos\theta\sin\beta \\ -\sin\theta\cos\beta\cos\lambda - \cos\theta\sin\lambda & \cos\theta\cos\lambda - \sin\theta\cos\beta\sin\lambda & \sin\theta\sin\beta \\ \sin\beta\cos\lambda & \sin\beta\sin\lambda & \cos\beta \end{pmatrix} \tag{6}$$

Hence, the 3D coordinates in the ecliptic coordinate system can be transformed to the coordinates in the Cellinoid model's system using Equation (7):

$$P' = \mathrm{QR}(P - \mathrm{Loc}_{\mathrm{Asteroid}}) \tag{7}$$

Here, Loc_{Asteroid} signifies the 3D coordinates of the asteroid within the ecliptic coordinate system, P denotes the coordinates of any given point within the same ecliptic system, and P' corresponds to the coordinates in the system where P is translated to the Cellinoid model's locale.

2.2. Brightness Calculation

The examination of scattering behavior is a critical aspect of asteroid modeling. A variety of physical parameters such as the single-scattering albedo Ω_0, the asymmetry factor g, the volume density of the surface material D, and the surface roughness ϱ were incorporated into the studies by several researchers, as cited (Lumme & Bowell 1981a, 1981b). These collective efforts culminated in the creation of a complex scattering model, designed to mimic the reflection of sunlight on asteroid surfaces. In a different approach, Hapke (1984) considered the opposition effect and shadowing within surface particle interactions. While these models are capable of representing the physical characteristics of light reflection, their practical application is impeded by uncertain physical parameters. To further explore this issue, Kaasalainen et al. (2005) carried out photometric research on an artificial asteroid in laboratory experiments. Their results indicate that the primary source of brightness variation is shape variation, not the scattering law. Additionally, they pinpointed a difficulty in differentiating between the scattering law and random error, highlighting the necessity for a simpler scattering law for efficient derivation of shape models. Addressing this need, Kaasalainen et al. (2001) introduced a practical method for simulating scattering behavior. This method consists of a linear combination of both the single scattering factor S_{LS} (Lommel–Seeliger) and the multiple scattering factor S_L (Lambert). This refined approach simplifies the process while still providing important insights into the scattering behavior of celestial bodies. By diminishing the complexity of the scattering law, it becomes easier for researchers to derive accurate shape models, thereby improving comprehension of asteroid surfaces and their interactions with sunlight. This improved understanding has wider implications for celestial mechanics and can provide relevant context for other aspects of asteroid research, such as trajectory prediction. In summary, the development and use of simpler, more efficient scattering laws have the potential to significantly enhance understanding of the complex and dynamic behavior of asteroids. The scattering law can be computed as Equation (8).

$$\begin{aligned} S(\mu, \mu_0, \alpha) &= f(\alpha)[S_{\mathrm{LS}}(\mu, \mu_0) + \gamma\, S_L(\mu, \mu_0)] \\ &= f(\alpha)\left(\frac{\mu\mu_0}{\mu + \mu_0} + \gamma\, \mu\mu_0\right), \end{aligned} \tag{8}$$

Here, α represents the solar phase angle (Sun–object–observer). Meanwhile, $f(\alpha)$ is the solar phase function, which is characterized by four parameters, as indicated in Equation (9):

$$f(\alpha) = A \exp\left(-\frac{\alpha}{D}\right) + K\alpha + B. \tag{9}$$

In addition, μ and μ_0 are defined as follows in Equation (10):

$$\mu = \eta \cdot E, \quad \mu_0 = \eta \cdot E_0, \tag{10}$$

where η symbolizes the outward unit normal vector of the surface. Additionally, the unit vectors E and E_0 correspond to the directions toward Earth and the Sun respectively, as perceived from the asteroid's perspective.

Utilizing the Cellinoid model, the brightness of the asteroid can be determined via surface integration, as demonstrated in Equation (11):

$$L(E, E_0) = \iint_{C^+} S(\mu, \mu_0, \alpha) ds, \tag{11}$$

where C^+ refers to the portion of the Cellinoid shape model's surface that is both visible and illuminated.

Moreover, a Cellinoid shape can be discretized utilizing a triangularization scheme. Consequently, the surface integral in Equation (11) can be approximately computed as Equation (12):

$$L(E, E_0) \approx \sum_{i=1}^{8}\left(\sum_{j=1}^{N}[S(\mu, \mu_0, \alpha) \triangle s_{i,j}]\right), \tag{12}$$

Here, i signifies the index of the octants, while j denotes the index of the triangular facets within each octant. Meanwhile, $\triangle s_{i,j}$ represents the area of the j-th facet in the i-th octant.

2.3. Mean Square Error

The MSE is an essential metric used in this research to evaluate the difference between the observed and simulated lightcurves generated using the Cellinoid shape model. The MSE is a common measure of prediction error in regression analyses and is particularly useful in this context as it provides a quantifiable measure of the accuracy of the model's predictions.

The MSE is calculated as the average of the squared differences between the observed and predicted values. This calculation is expressed mathematically as follows:

$$\mathrm{MSE} = \frac{1}{M}\sum_{i=1}^{M}(L_i - \tilde{L}_i)^2 \tag{13}$$

Here, L_i denotes the observed brightness for the i-th lightcurve, whereas $\tilde{L}_i$ stands for the calculated brightness aligned with the

observation geometry. The summation extends across all M data points.

In cases where the observed lightcurve is uncalibrated, a modified form of the MSE function is employed, as detailed in Equation (14).

$$\mathrm{MSE} = \frac{1}{M}\sum_{i=1}^{M}\left(\frac{L_i}{\langle L_i\rangle} - \frac{\tilde{L}i}{\langle \tilde{L}i\rangle}\right)^2 \tag{14}$$

In Equation (14), $\langle L_i\rangle$ and $\langle \tilde{L}_i\rangle$ correspond to the mean observed and synthetic brightness, respectively. Similar to previous instances, the summation is conducted across all M data points.

3. Parallel Differential Evolution Algorithm

With the fundamentals of the Cellinoid shape model and the mean square error metric established, the focus now shifts to utilizing these components within an optimization framework to reconstruct asteroid models from observed brightness data. Specifically, a PDE algorithm is developed to efficiently search the high-dimensional parameter space and determine the Cellinoid parameters that best reproduce the lightcurve observations. The following sections describe the differential evolutionary algorithm and the parallel acceleration scheme implemented to expedite convergence. First, the basic differential evolutionary algorithm is outlined, including details of the mutation, crossover, and selection operations that underlie this global optimization technique. Building upon this foundation, the parallelization approach is then presented, introducing concurrent fitness evaluation across subpopulations and a periodic exchange of solutions between subpopulations to enhance the search diversity and rate of convergence. Together, these methods provide an optimized framework for accelerating the asteroidal properties inversion process by fitting Cellinoid shapes to lightcurve observations using an accelerated, parallelized differential evolutionary algorithm.

3.1. Differential Evolutionary Algorithm

Differential Evolution (DE) is an evolutionary algorithm uniquely suited for optimizing problems that involve real-valued parameters. This makes it ideal for tasks such as fitting the Cellinoid model to lightcurve data. DE optimizes candidate solutions by iteratively refining a population through mutation, crossover, and selection operations, all of which are inspired by biological evolution. In DE, each member of the population is a vector representing a potential solution. In the context of fitting Cellinoid models, an individual vector would contain the six shape parameters, two pole orientation angles, period, rotational phase angle, and phase function coefficients.

The DE algorithm operates on a population of potential solutions, enhancing these solutions iteratively through a series of mutation, crossover, and selection operations. These operations are elaborated as follows:

3.1.1. Mutation

The first step in the DE algorithm is mutation. During this operation, new trial vectors are generated by adding the weighted difference between two population vectors to a third vector. This can be mathematically represented as:

$$V = X_{r0} + F \cdot (X_{r1} - X_{r2}) \tag{15}$$

In Equation (15), X_{r0}, X_{r1}, and X_{r2} are vectors randomly selected from the current population, and F is a scaling factor that controls the amplification of the differential variation $(X_{r1} - X_{r2})$. The mutation operation introduces diversity into the population and enables the exploration of the search space.

3.1.2. Crossover

The crossover operation is designed to further enhance the diversity of the population and explore the search space. In Equation (16), components of the mutant vector V (created during the mutation stage) are mixed with the components of the target vector X_i to generate a trial vector U. This combination is governed by a crossover probability recombination:

$$U_j = \begin{cases} V_j & \text{if } \mathrm{rand}_j \leqslant \text{recombination or } j = \mathrm{rand}n[i] \\ X_{i,j} & \text{otherwise} \end{cases} \tag{16}$$

Here, rand_j is a uniformly distributed random number between 0 and 1, and randn$[i]$ is a randomly selected index.

3.1.3. Selection

In the selection operation, as shown in Equation (17), the trial vector U competes against the target vector X_i from the original population. The vector with the lower objective function value (Equation (13) or Equation (14)) remains and proceeds to the next generation:

$$X_i = \begin{cases} U & \text{if } f(U) \leqslant f(X_i) \\ X_i & \text{otherwise} \end{cases} \tag{17}$$

Here, $f(U)$ and $f(X_i)$ represent the objective function values of the trial and target vectors, respectively. This selection operation ensures the quality of solutions in the population improves over generations.

The DE algorithm repeats these mutations, crossover, and selection operations until a stopping criterion is met, such as a maximum number of generations or a satisfactory objective function value. Through these iterative operations, the DE algorithm efficiently navigates the high-dimensional parameter space and finds the optimal set of parameters that minimize the objective function, in this case, the MSE between observed and simulated lightcurve data.

Through this simple but powerful evolutionary process, the DE implementation is able to robustly optimize solutions for the complex, high-dimensional Cellinoid fitting problem.

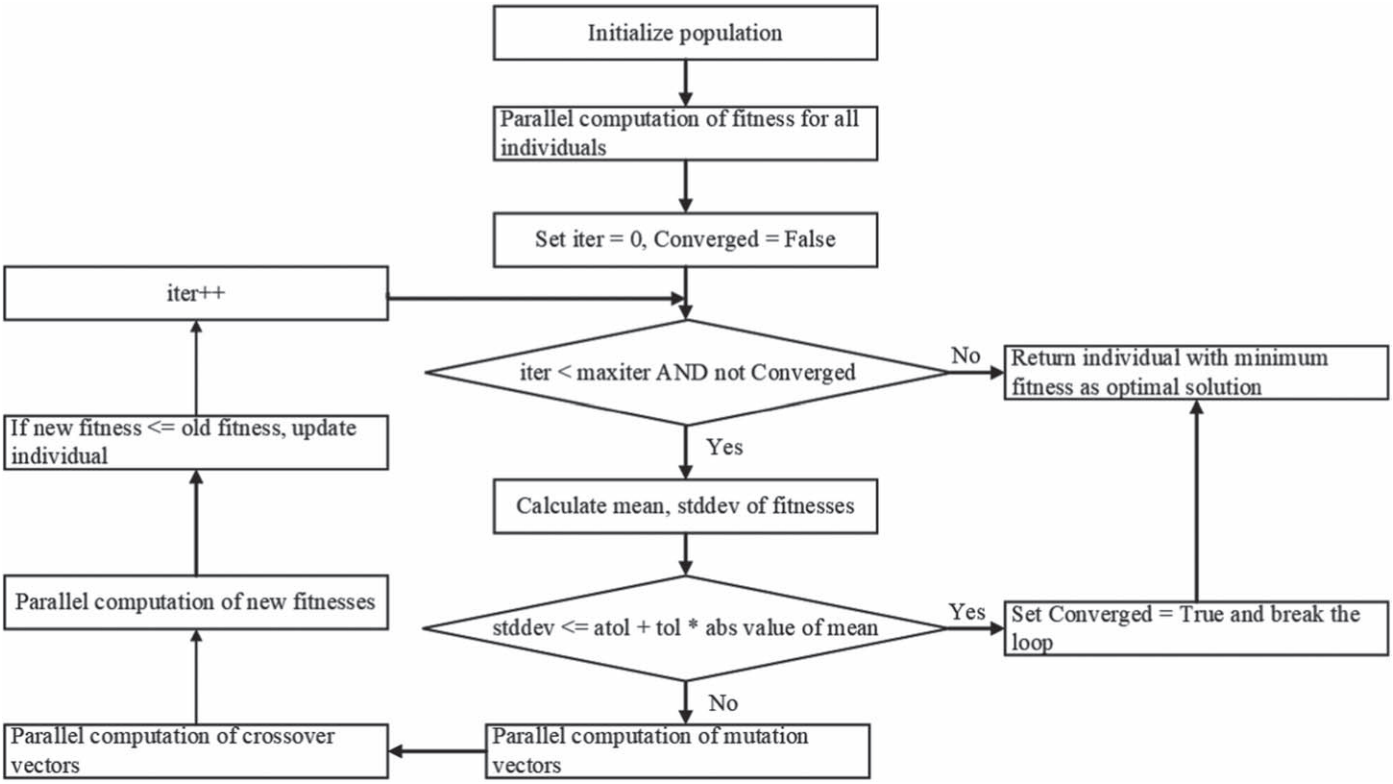

Figure 3. Flowchart of the PDE algorithm.

3.2. Parallel Acceleration Scheme

The PDF algorithm is designed to enhance the computational efficiency of the DE algorithm for physical properties inversion of asteroids. It takes advantage of parallel processing to simultaneously evaluate and evolve multiple individuals in the population. This algorithm is particularly advantageous when dealing with large populations or complex objective functions that require substantially computational resources. The procedure of this algorithm is shown in Figure 3, and its pseudocode is shown in Algorithm 1.

Algorithm 1. Parallel Differential Evolution for Asteroid Physical Properties Inversion

Input: f: objective function, *bounds*: boundaries of parameters, *popsize*: population size, F: scaling factor, *atol*: absolute tolerance, *tol*: relative tolerance, *recombination*: crossover probability, *maxiter*: maximum iterations, *workers*: parallel processes
Output: x^*: optimal solution
1: Initialize population $x_1, x_2,\ldots,x_{popsize}$ randomly within *bounds*
2: **Evaluate** fitnesses $f(x_i)$ for all individuals **in parallel** using *workers* processes
3: Set $iter \leftarrow 0$, $Converged \leftarrow False$
4: **while** $iter < maxiter$ **and** not *Converged***do**
5: Calculate *mean*, *stddev* of $f(x_1), f(x_2),\ldots,f(x_{popsize})$
6: **if** $stddev \leqslant atol + tol * |mean|$ **then**
7: $Converged \leftarrow True$
8: **break**
9: **end if**
10: **Evolve** population for one generation:
11: **for** $i \leftarrow 1$ **to** *popsize* **in parallel do**
12: $v_i \leftarrow$ Mutate(x_i) with scaling factor F
13: **end for**
14: **for** $i \leftarrow 1$ **to** *popsize* **in parallel do**
15: $x_i' \leftarrow$ Crossover(v_i) with probability *recombination*
16: **end for**
17: **Selection**:
18: **for** $i \leftarrow 1$ **to** *popsize* **in parallel do**
19: **Evaluate** fitnesses $f(x_i')$
20: **if** $f(x_i') \leqslant f(x_i)$**then**
21: $x_i \leftarrow x_i'$
22: **end if**
23: **end for**
24: $iter++$
25: **end while**
26: **return argmin** $_i f(x_i)$ as x^*

The PDE algorithm works similarly to the standard DE algorithm, but with a crucial modification: multiple evolutionary operations (such as mutation, crossover, and selection)

are performed concurrently using multiple parallel processes. The number of parallel processes, or *workers*, is a parameter that can be adjusted based on the computational resources available.

The PDE algorithm begins by initializing a population of potential solutions within the provided parameter bounds. The fitness of each individual, evaluated by the objective function (Equations (13) or (14)), is then determined in parallel. The algorithm then enters a loop, which continues until either the maximum number of iterations is reached or the population has converged to a satisfactory solution, as determined by the standard deviation of the fitnesses being less than a specified tolerance.

Within the loop, the population is evolved for one generation. Each individual in the population is first mutated and then undergoes crossover, with each operation conducted in parallel across the population. The fitness of each resulting individual is then evaluated, and the selection operation is performed to determine which individuals proceed to the next generation. This process effectively utilizes parallel processing to expedite the evolutionary process.

At the end of the loop, the individual with the lowest fitness (i.e., the best solution) is returned as the optimal solution. This parallel acceleration scheme allows the DE algorithm to efficiently tackle the high-dimensional parameter optimization problem involved in asteroid physical properties inversion.

4. Numerical Experiments and Discussion

This section explores the practical application of the PDE algorithm for inverting the physical properties of asteroids. The exploration begins with an analysis of the performance of the PDE algorithm. Next come the generation and analysis of synthetic lightcurves. This controlled setting, where the real properties of the asteroids are known, allows for a precise evaluation of the algorithm's accuracy and efficiency. Following this, the focus shifts to a real-world case study of asteroid (15) Eunomia, using real observational data to demonstrate the algorithm's capabilities in a practical context. After examining the case study, the application of the PDE algorithm is extended to other real asteroids, further validating its robustness and general applicability. The section concludes with a discussion on a methodology for inverting observational data that accounts for observational uncertainties.

4.1. Performance Analysis of PDE Algorithms

This section presents an analysis of the performance of PDE algorithms on a dual multicore CPU architecture. The focus is on evaluating the computational efficiency of the PDE algorithms under varying degrees of parallelization. The PDE algorithm was implemented using a combination of the C and Python programming languages. C was chosen for its low-level control over hardware interactions, which is critical for performance optimization, while Python was utilized for its robust scientific computing libraries and ease of code integration. The experimental platform employed was a PC server configured with two Hygon C86 7185 CPUs, each with 32 cores and a 2000 MHz clock rate, and 128 GB of RAM. This high-performance setup was selected to subject the PDE algorithms to computationally demanding tasks.

The PDE algorithm was configured with specific parameters for the evaluation:

1. The population size was set at 30, which is standard for ensuring a stable evolutionary process.
2. The convergence criteria were defined by a relative tolerance of 0.000 001 and an absolute tolerance of zero.
3. A dithering scaling factor with a range between 0.5 and 1 was employed. This involved dynamically adjusting the mutation constant for each generation from a uniform distribution within the given range.
4. The crossover probability was set at 0.7 to dictate the mixing of solution traits.

The chosen test case for the performance analysis was asteroid (85) Io, with a data set of 557 points. Workers were distributed in a series of tests with 1, 2, 4, 8, 16, 32, and 64 workers to assess the algorithm's efficiency. This distribution pattern allowed for the observation of performance changes in response to the doubling of the number of processors, indicative of the algorithm's parallel efficiency. For consistency, the maximum number of iterations was capped at 1000 for all tests. This uniformity ensured that each computation was allowed to progress through an equal number of iterations, providing a fair basis for comparison of the execution duration and resource utilization across various worker configurations.

The elapsed time, speedup ratio, and parallelization efficiency are employed as metrics to evaluate the parallel performance of the PDE algorithm. In this paper, the definitions used by Zhang et al. (2023) for the speedup ratio and parallelization efficiency are adopted. The speedup ratio is shown in Equation (18), and the parallelization efficiency is shown in Equation (19).

$$\text{speedup} = \frac{T_1}{T_n}, \tag{18}$$

$$\text{efficiency} = \frac{T_1}{n \cdot T_n}, \tag{19}$$

where n denotes the number of workers and T_n denotes the time required to complete the PDE algorithm with n workers.

The computational efficacy of a PDE algorithm executed on a dual multicore CPU architecture was scrutinized through the systematic increase of worker processes, with particular attention to the metrics of elapsed time, speedup, and computational efficiency, as illustrated in Table 1. When using a single worker, the PDE algorithm is the DE algorithm.

Table 1
Performance on Multicore CPUs with Different Numbers of Workers (n)

Workers (n)	Elapsed Time (hr)	Speedup	Efficiency
1	3.592	1.000	1.000
2	1.830	1.963	0.981
4	0.957	3.752	0.938
8	0.520	6.904	0.863
16	0.278	12.944	0.809
32	0.159	22.604	0.706
64	0.095	37.983	0.593

Starting with this single worker, which establishes a benchmark duration of 3.592 hr, the PDE algorithm consistently achieves reduced processing times and enhanced speedup with the addition of more workers. Peak efficiency was attained with two workers, reaching 0.981, after which it demonstrated a gradual decline. This pattern emphasizes the principle of diminishing returns as the worker count escalates. Remarkably, 64 workers achieved an extraordinary speedup of 37.983; however, this was accompanied by the lowest efficiency of 0.593, indicating that the drawbacks of increased overhead could outweigh the benefits of extensive parallelism.

Through analysis, the experiments conducted in the subsequent sections of this paper employ a maximum of 20,000 iterations to ensure that the algorithms run until convergence criteria are met, with a relative tolerance of 0.000001 and an absolute tolerance of zero. Additionally, the number of workers is set to 60.

4.2. Generation and Analysis of Synthetic Lightcurves

This section validates the PDE algorithm through the use of a Cellinoid shape as an asteroid model. The positional data for the Earth and asteroid (433) Eros were sourced from the Jet Propulsion Laboratory (JPL) Horizons On-Line Ephemeris System,[6] which is maintained by NASA, for the period 2019 November 1–2021 October 31.

A Cellinoid model-based shape was created using the real parameter values, which include information on the shape, pole, period, and rotational phase angle: $a_1 = 1$, $a_2 = 0.78$, $b_1 = 0.90$, $b_2 = 0.73$, $c_1 = 0.61$, $c_2 = 0.82$, $\lambda = 132°$, $\beta = -42°$, $P = 5.8$ hr, and $\theta = 191°$. Six apparitions were selected for analysis: day 19, day 51, day 212, day 261, day 299, and day 544. The average solar phase angles of these apparitions were 19°, 12°, 24°, 33°, 39°, and 35°, respectively. Subsequently, lightcurve data for each apparition was generated, with each apparition's time span kept below the set period of 5.8 hr.

The study extends to the generation of synthetic lightcurves with the incorporation of controlled noise, aiming to simulate the intensity fluctuations observed in real astronomical data. This technique is essential for validating the analytical methods utilized in the processing of observational lightcurves. Uniform noise, drawn from a distribution within the interval $[-1, 1]$, was introduced to the intensity data. A uniform distribution was selected to ensure that all potential noise values within the range are equally probable. The intensity of the noise was adjusted using a coefficient of 0.02, serving to scale the noise relative to the original data values. The formula for the noised intensity I_{noised} is shown in Equation (20).

$$I_{\text{noised}} = I_{\text{original}} \cdot (1 + N \cdot 0.02) \tag{20}$$

Here, I_{original} denotes the original intensity value, and N represents the noise value from the uniform distribution.

The experiment utilized the lightcurves of these six apparitions with uniform noise as the data set. Initial parameters were randomly generated, and the PDE algorithm was used to derive optimal results. These results yielded optimal parameters for shape, pole, period, and rotational phase angle: $a_1 = 0.93$, $a_2 = 0.73$, $b_1 = 0.46$, $b_2 = 0.89$, $c_1 = 0.69$, $c_2 = 0.83$, $\lambda = 129.^{\circ}00$, $\beta = -43.^{\circ}59$, $P = 5.800004$ hr, and $\theta = 188.^{\circ}61$. Despite a slight deviation in the rotational phase angle and the inverted pole, the inverted and preset periods were highly consistent. Additionally, a good fit using the noisy data is evident, as illustrated in Figure 4.

The PDE algorithm reached convergence after 2242 iterations and the elapsed time is 0.132167 hr. Throughout this process, the values for period, longitude, and latitude oscillated around 5.800004 hr, $129.^{\circ}00$, and $-43.^{\circ}59$, respectively, as shown in Figures 5(b), (c), (d). In the analysis of synthetic lightcurves, the stability of the MSE values plays a crucial role in determining the reliability of the iterative process used for uncertainty estimation. As observed in Figure 5(a), the MSE values stabilize and maintain consistency in the final third of the iterations.

This paper presents a specific method for estimating the uncertainty in the outcomes of the PDE algorithm. The PDE algorithm obtains the optimal solution at each iteration during the iterative process. Recognizing that earlier iterations may yield optimal solutions that deviate from the true solution, this paper considers the optimal solutions from the final third of the iteration process to estimate uncertainty. Solutions from later iterations are more likely to approximate the true solution, and the fluctuations among these later solutions serve as a measure of the uncertainty estimation. This approach resulted in estimated uncertainties for the inverted pole ($128.^{\circ}96 \pm 0.^{\circ}06$, $-43.^{\circ}57 \pm 0.^{\circ}03$) and the inverted period ($5.800004 \pm 0.000\,001$ hr). This uncertainty estimation approach will be adopted in all subsequent experiments presented in this paper.

[6] The web address for accessing the Jet Propulsion Laboratory (JPL) Horizons On-Line Ephemeris System is https://ssd.jpl.nasa.gov/horizons.

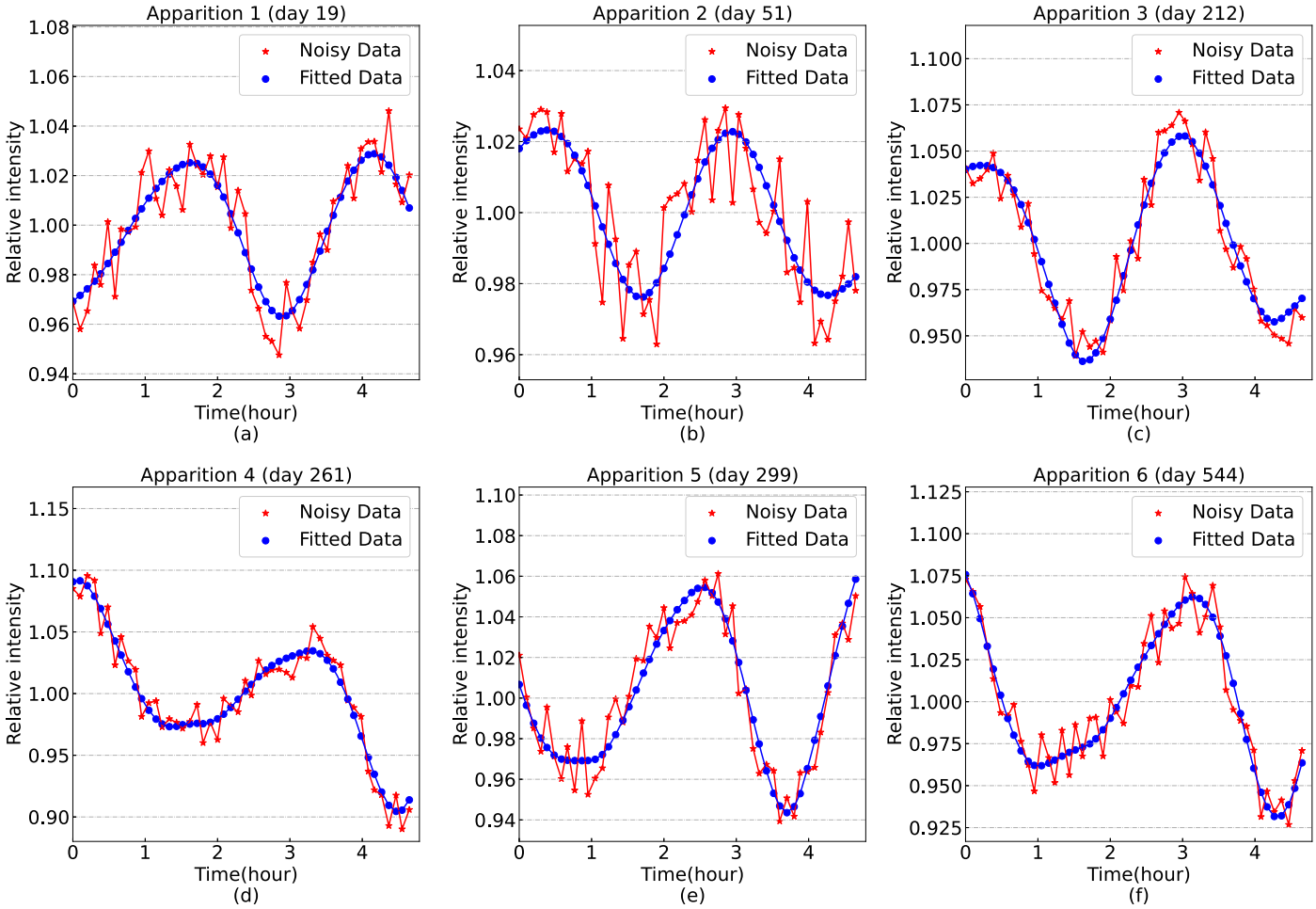

Figure 4. Simulated noisy lightcurves and fitted models for the synthetic lightcurves at six geometries.

4.3. Case Study: Asteroid (15) Eunomia

This section presents a comprehensive analysis of asteroid (15) Eunomia, utilizing data from various observations. According to Lu et al. (2017), the precision of the inverted pole orientation can be enhanced by using observations collected from diverse viewing geometries. Therefore, lightcurves from 12 separate apparitions have been employed in the search for the best-fit parameters, applying the PDE algorithm.

The existing literature provides fundamental parameters for asteroid (15) Eunomia, such as pole orientation and rotation period. Kaasalainen et al. (2002), Nathues et al. (2005), Hanuš et al. (2013) report a rotation period of 6.082753 hr with pole orientation of (3°, − 67°). In contrast, Viikinkoski et al. (2017) report a period of 6.082752 hr and pole orientation of (0°, − 68°), while Vernazza et al. (2021) report a period of 6.082754 hr with pole orientation of (356°, − 70°). Additionally, lightcurves from 12 apparitions have been analyzed, which were collected over the years from 1952 to 2009 (Ďurech et al. 2010). The average solar phase angles for these apparitions ranged from 5°.78 to 20°.93. The total sample size is 554.

The PDE algorithm was applied to the lightcurve data from these 12 apparitions to derive optimal results. The resulting optimal parameters for shape, pole, period, and rotational phase angle were as follows: $a_1 = 0.94$, $a_2 = 0.83$, $b_1 = 0.82$, $b_2 = 0.45$, $c_1 = 0.70$, $c_2 = 0.51$, $\lambda = 358°.23$, $\beta = -64°.07$, $P = 6.082753$ hr, and $\theta = 0°$. A notably good fit was achieved with the observed data, as illustrated in Figure 6. The inverted pole closely aligns with the values sourced from the literature (Kaasalainen et al. 2002; Nathues et al. 2005; Hanuš et al. 2013; Viikinkoski et al. 2017; Vernazza et al. 2021). Consistency was also found between the inverted periods and those sourced from the literature.

The PDE algorithm achieved convergence after 3430 iterations; the elapsed time was 0.322225 hr, during which the values of period, longitude, and latitude oscillated around 6.082753 hr, 358°.23, and −64°.07, respectively, as depicted in Figures 7(b) (c) (d). This paper selects the results from the last third of the iterations for uncertainty estimation, yielding

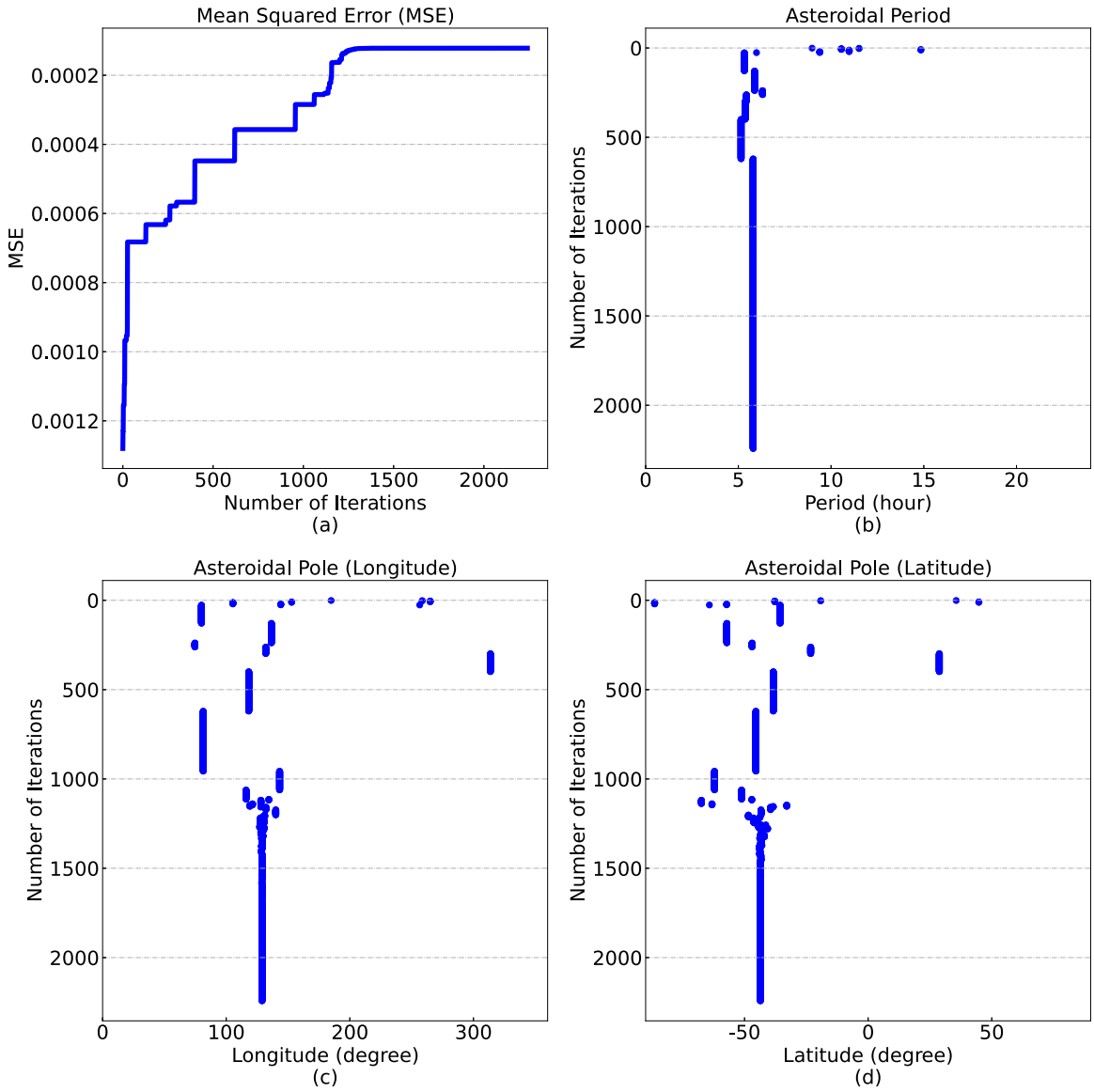

Figure 5. Convergence analysis of PDE results for the synthetic lightcurves. (a) Mean squared error (MSE) at each iteration. (b) Optimal period identified at each iteration, with final optimal period of 5.800004 hr. (c) Optimal longitude identified at each iteration, with final optimal longitude of 129.°00. (d) Optimal latitude identified at each iteration, with final optimal latitude of −43.°59.

estimated uncertainties for the inverted pole (358.°22 ± 0.°05, −64.°06 ± 0.°02) and the inverted period (6.082753 ± 0.000001 hr).

4.4. Application to Other Real Asteroids

The PDE algorithm was applied to the lightcurve data of 16 different asteroids, with the results presented in Table 2. The table includes the inverted pole orientations and rotational periods, as determined both by the DAMIT database values (Ďurech et al. 2010) and by the PDE algorithm.

The algorithm was executed on a PC server equipped with two Hygon C86 7185 CPUs and 128 GB of RAM. Table 2 demonstrates the efficiency of the PDE algorithm, which computed the results for each asteroid in less than an hour, all while staying within the maximum limit of 20,000 iterations.

The PDE results for the pole orientations (longitude and latitude) and rotational periods are presented with their associated uncertainties. The PDE algorithm was able to accurately derive the rotational periods of the asteroids from the lightcurve data, and the latitude of the pole orientation was also close to the reference values (Ďurech et al. 2010). A significant

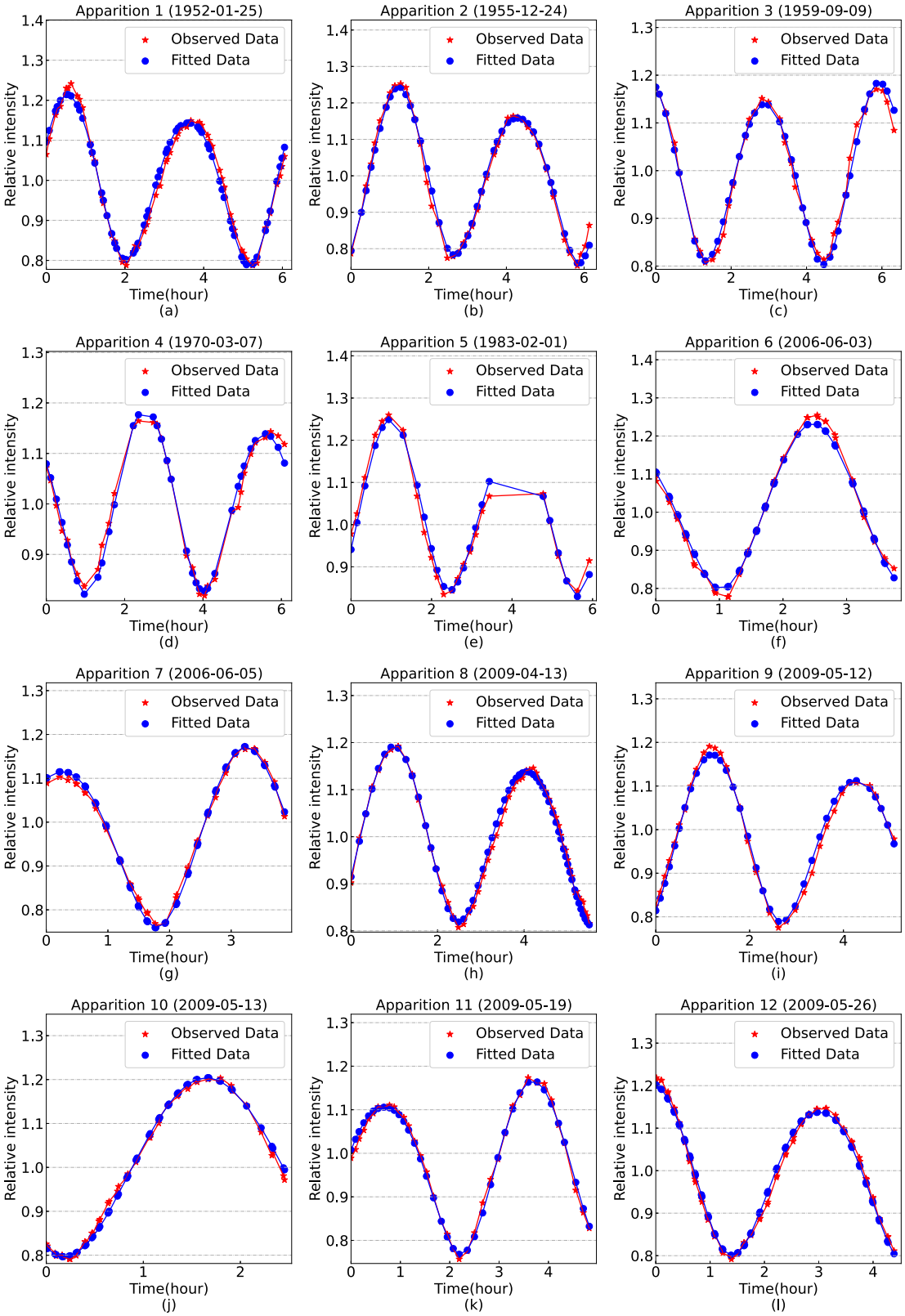

Figure 6. Observed photometric lightcurves and fitted models for asteroid (15) Eunomia at 12 apparitions on 1952 January 25, 1955 December 24, 1959 September 9, 1970 March 7, 1983 February 1, 2006 June 3, 2006 June 5, 2009 April 13, 2009 May 12, 2009 May 13, 2009 May 19, and 2009 May 26.

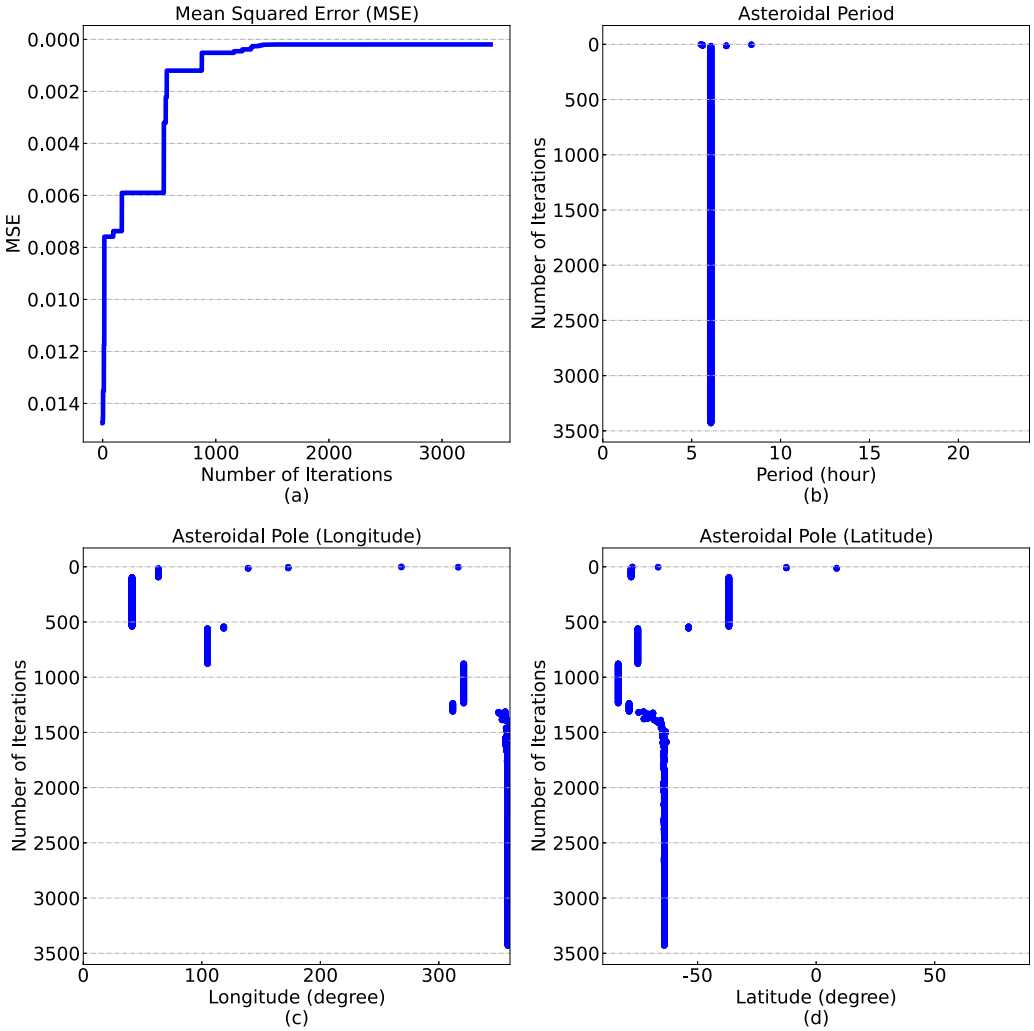

Figure 7. Convergence analysis of PDE results for the asteroid (15) Eunomia. (a) Mean squared error (MSE) at each iteration. (b) Optimal period identified at each iteration, with final optimal period of 6.082753 hr. (c) Optimal longitude identified at each iteration, with final optimal longitude of 358.°23. (d) Optimal latitude identified at each iteration, with final optimal latitude of $-64.^{\circ}07$.

deviation in longitude was observed for eight asteroids compared to the reference values (Ďurech et al. 2010), which can be attributed to the geometric flexibility of the Cellinoid shape model. As suggested by Lu et al. (2017), this deviation in longitude can be explained by adjusting the rotational phase angle and the octant positions in the model.

These results indicate that the PDE algorithm can be effectively applied to determine the pole orientations and rotational periods of other real asteroids. This opens up new possibilities for future research in asteroid science and can contribute to a more comprehensive understanding of the physical properties of asteroids.

4.5. Discussion

The results obtained in this study underscore the effectiveness and reliability of the PDE algorithm in determining the pole orientations and rotational periods of asteroids from lightcurve data. The consistency between the inverted rotational periods from PDE and the values listed in the DAMIT database (Ďurech et al. 2010) provides an indication of the algorithm's accuracy and, by extension, its potential utility in asteroid science.

A noteworthy aspect of the PDE algorithm's performance is its computational efficiency. The ability to compute results for each asteroid in less than an hour, while staying within the maximum limit of 20,000 iterations, suggests that the algorithm could be applicable to large-scale data sets. This opens up new possibilities for the analysis of extensive asteroid lightcurve data, which could, in turn, lead to more robust findings in asteroid science.

The results of this research accentuate the effectiveness and reliability of the PDE algorithm for determining the pole

Table 2
Comparison of Asteroid Pole Orientations and Rotational Periods from the DAMIT Database (Ďurech et al. 2010) vs. the PDE Algorithm Results

Asteroid	LCs	Size	DAMIT (λ, β, P)	PDE Result (λ, β, P)	PDE Result with Uncertainty (λ, β, P)	Elapsed Time	Iterations
(19) Fortuna	14	534	(98°, 57°, 7.443224 hr) (97°, 69°, 7.443222 hr) (103°, 60°, 7.443224 hr)	(109.°36, 81.°22, 7.443222 hr)	(109.°37 ± 0.°46, 81.°24 ± 0.°13, 7.443222 ± 0.000001 hr)	0.499642 hr	5557
(21) Lutetia	13	622	(54°, − 7°, 8.168269 hr) (52°, − 6°, 8.168271 hr)	(54.°56, −6.°08, 8.168 271 hr)	(54.°63 ± 0.°67, −6.°21 ± 0.°78, 8.168271 ± 0.000001 hr)	0.334544 hr	3221
(29) Amphitrite	10	542	(322°, − 28°, 5.390119 hr) (324°, − 26°, 5.390119 hr) (323°, − 29°, 5.39012 hr)	(131.°42, −26.°99, 5.390127 hr)	(130.°39 ± 2.°62, −27.°88 ± 2.°12, 5.390127 ± 0.000001 hr)	0.447484 hr	4785
(43) Ariadne	11	521	(253°, − 15°, 5.76199 hr) (251°, − 10°, 5.761987 hr)	(252.°44, −15.°15, 5.760 681 hr)	(252.°43 ± 0.°01, −15.°14 ± 0.°03, 5.760681 ± 0.000001 hr)	0.219741 hr	2509
(44) Nysa	16	661	(101°, 51°, 6.421417 hr)	(103.°09, 51.°92, 6.421 416 hr)	(103.°09 ± 0.°06, 51.°91 ± 0.°03, 6.421416 ± 0.000001 hr)	0.332210 hr	3070
(45) Eugenia	16	654	(124°, − 33°, 5.699152 hr) (127°, − 35°, 5.699152 hr)	(124.°64, −32.°17, 5.699 150 hr)	(124.°56 ± 0.°11, −32.°22 ± 0.°09, 5.699150 ± 0.000001 hr)	0.378442 hr	3512
(55) Pandora	8	499	(223°, 18°, 4.804043 hr)	(32.°57, 21.°79, 4.804 802 hr)	(32.°61 ± 0.°90, 21.°78 ± 0.°55, 4.804802 ± 0.000001 hr)	0.423068 hr	4907
(66) Maja	16	638	(49°, − 70°, 9.7357 hr) (225°, − 68°, 9.73572 hr)	(314.°12, −64.°98, 9.735 703 hr)	(313.°88 ± 1.°15, −65.°10 ± 0.°41, 9.735703 ± 0.000001 hr)	0.571718 hr	5419
(69) Hesperia	18	588	(250°, 17°, 5.65534 hr) (71°, − 2°, 5.65534 hr)	(267.°93, −1.°99, 5.655332 hr)	(267.°28 ± 1.°47, −1.°36 ± 1.°88, 5.655332 ± 0.000001 hr)	0.554159 hr	5551
(73) Klytia	12	474	(44°, 83°, 8.28307 hr) (266°, 68°, 8.28307 hr)	(65.°78, 81.°87, 8.283 067 hr)	(65.°73 ± 0.°07, 81.°93 ± 0.°10 8.283067 ± 0.000001 hr)	0.390488 hr	4630
(85) Io	12	557	(95°, − 65°, 6.874783 hr) (92°, − 68°, 6.874784 hr)	(296.°12, −74.°16, 6.874 796 hr)	(295.°64 ± 1.°84, −74.°19 ± 0.°59, 6.874796 ± 0.000001 hr)	0.809104 hr	8588
(130) Elektra	14	579	(64°, − 88°, 5.224664 hr) (64°, − 90°, 5.224663 hr)	(164.°53, −88.°08, 5.224 663 hr)	(164.°80 ± 1.°37, −88.°08 ± 0.°02, 5.224663 ± 0.000001 hr)	0.461312 hr	4770
(166) Rhodope	7	279	(173°, − 3°, 4.714799 hr) (345°, − 22°, 4.714793 hr)	(185.°08, −3.°04, 4.715 909 hr)	(184.°95 ± 0.°59, −2.°13 ± 3.°05, 4.715904 ± 0.000029 hr)	0.290588 hr	5266
(281) Lucretia	8	446	(128°, − 49°, 4.349711 hr) (309°, − 61°, 4.349711 hr)	(26.°36, −72.°34, 4.349 684 hr)	(34.°50 ± 8.°17, −76.°20 ± 3.°81, 4.349684 ± 0.000001 hr)	0.482110 hr	6059
(311) Claudia	23	386	(30°, 40°, 7.53138 hr) (214°, 43°, 7.53138 hr)	(215.°59, 42.°37, 7.531390 hr)	(215.°56 ± 0.°13, 42.°34 ± 0.°24, 7.531390 ± 0.000001 hr)	0.289181 hr	4021
(355) Gabriella	9	435	(341°, 83°, 4.828994 hr) (159°, 88°, 4.828994 hr)	(243.°35, 86.°52, 4.828994 hr)	(243.°26 ± 0.°08, 86.°51 ± 0.°01, 4.828994 ± 0.000001 hr)	0.349521 hr	4494

Note. "LCs" represents the number of lightcurves. "Size" indicates the size of the sample. "(λ, β)" denotes the longitude and latitude of poles in the ecliptic frame, respectively. Longitude varies from 0° to 360°, while latitude ranges from −90° to 90°. "P" is the derived rotational period, measured in hours. "DAMIT" represents data obtained from the source cited as Ďurech et al. (2010). "PDE Result" represents the optimal solution obtained from the last iteration of the PDE algorithm. "PDE Result with uncertainty" indicates that this paper selects the results from the final third of the iterations for uncertainty estimation by the PDE algorithm. "Elapsed Time" denotes the total time taken by the PDE algorithm to reach the solution, measured in hours. "Iterations" represents the number of iterations performed by the PDE algorithm to reach the solution. The maximum limit for iterations is set at 20,000. As suggested by Lu et al. (2017), this deviation in longitude can be explained by the geometric flexibility of the Cellinoid shape model, which can be adjusted by altering the rotational phase angle and the octant positions.

Table 3
Comparison of Asteroid Pole Orientations and Rotational Periods from the Hipparcos Dataset (Cellino et al. 2019) vs. the PDE Algorithm Results

Asteroid	Size	DAMIT (λ, β, P)	PDE Result (λ, β, P)	PDE Result with Uncertainty (λ, β, P)	Elapsed Time	Iterations
(6) Hebe	91	(340°, 42°, 7.274471 hr) (342°, 50°, 7.274467 hr) (342°, 51°, 7.274467 hr)	(316.°17, 20.°46, 7.274408 hr)	(315.°97 ± 1.°19, 20.°64 ± 1.°38, 7.274419 ± 0.000001 hr)	0.409761 hr	14,011
(15) Eunomia	83	(3°, − 67°, 6.082753 hr) (0°, − 68°, 6.082752 hr) (356°, − 70°, 6.082754 hr)	(360.°00, −71.°21, 6.082 794 hr)	(355.°86 ± 4.°20, −70.°65 ± 0.°82, 6.082783 ± 0.000024 hr)	0.234453 hr	8127
(39) Laetitia	112	(323°, 32°, 5.138238 hr) (323°, 33°, 5.138238 hr)	(316.°33, 10.°61, 5.138238 hr)	(316.°23 ± 0.°37, 10.°51 ± 0.°57, 5.138238 ± 0.000001 hr)	0.182914 hr	5661

Note. "Size" indicates the size of the sample. "(λ, β)" denotes the longitude and latitude of poles in the ecliptic frame, respectively. Longitude varies from 0° to 360°, while latitude ranges from −90° to 90°. "P" is the derived rotational period, measured in hours. "DAMIT" represents data obtained from the source cited as Ďurech et al. (2010). "PDE Result" represents the optimal solution obtained from the last iteration of the PDE algorithm. "PDE Result with uncertainty" indicates that this paper selects the results from the final third of the iterations for uncertainty estimation by the PDE algorithm. "Elapsed Time" denotes the total time taken by the PDE algorithm to reach the solution, measured in hours. "Iterations" represents the number of iterations performed by the PDE algorithm to reach the solution. The maximum limit for iterations is set at 20,000.

orientations and rotational periods of asteroids from lightcurve data. Nonetheless, the DAMIT database (Ďurech et al. 2010) does not provide observational uncertainties. The section addresses a methodology for inverting observational data that includes observational uncertainties. This involves enhancing Equations (13) and (14) by incorporating the observational uncertainty parameter σ_i, as delineated below:

$$\mathrm{MSE} = \frac{1}{M}\sum_{i=1}^{M}\left(\frac{L_i - \tilde{L}_i}{\sigma_i}\right)^2 \tag{21}$$

$$\mathrm{MSE} = \frac{1}{M}\sum_{i=1}^{M}\left[\frac{1}{\sigma_i}\left(\frac{L_i}{\langle L_i\rangle} - \frac{\tilde{L}i}{\langle \tilde{L}i\rangle}\right)\right]^2 \tag{22}$$

This paper employs sparse photometric data from the Hipparcos data set (Cellino et al. 2019), which includes observational uncertainties. Asteroids (6) Hebe, (15) Eunomia, and (39) Laetitia are selected for experimentation, with the results presented in Table 3.

As indicated by Table 3, even though sparse photometric data from the Hipparcos data set (Cellino et al. 2019) were used, accurate rotational periods were still obtained. However, only the pole orientation of asteroid (15) Eunomia achieved relatively good results, whereas the other two asteroids showed deviations from the reference values (Ďurech et al. 2010). This is in agreement with the conclusion drawn by Lu et al. (2017), which states that more observations collected in various viewing geometries can improve the accuracy of the inverted pole orientation. In other words, obtaining accurate results for pole orientations requires more varied viewing geometries. The issue of deviated longitudes for the eight asteroids in Table 2 can also be addressed by increasing the number of viewing geometries.

As with any algorithm, there is always room for improvement. Future research could focus on enhancing the PDE algorithm's efficiency and accuracy. For instance, exploring methods for optimizing the algorithm's parameters or implementing machine learning approaches could potentially improve its performance. Furthermore, the algorithm could be tested on a wider range of asteroid types and sizes to assess its versatility and adaptability.

5. Conclusion

This study demonstrates a new approach for efficiently determining the rotational periods and pole orientations of asteroids from observed lightcurve data. The proposed Parallel Differential Evolution (PDE) algorithm leverages a Cellinoid shape model and parallel computing to accelerate the inversion process.

During performance testing of the PDE algorithm, it was discovered that the PDE is more efficient than the DE algorithm, achieving an extraordinary speedup of 37.983 with 64 workers. Additionally, the results confirm the effectiveness of the PDE algorithm and the Cellinoid model on both simulated and real asteroid data. For simulated asteroids, the PDE method recovers the real rotational periods and pole positions with high precision, even when uniform noise is introduced. The Cellinoid shape model provides a robust approximation for the irregular shapes of asteroids, enabling efficient computation. Further analysis of asteroid lightcurves has substantiated the reliability of the PDE algorithm. The derived rotational periods show excellent concordance with the reference values from the DAMIT database (Ďurech et al. 2010), despite the simplicity of the Cellinoid model. With adequate viewing geometries, the derived pole orientations generally match the reference values from the DAMIT database (Ďurech et al. 2010). Furthermore, the PDE approach demonstrates satisfactory computational efficiency, converging

to solutions within 20,000 iterations and under one hour for all test cases. The statistical approach adopted in this work to estimate the uncertainties is a simple one, which can be further improved in future research.

This work demonstrates a promising new technique for gaining insight into the rotational dynamics of asteroids from photometric observations. By accelerating the inversion process, the PDE algorithm and Cellinoid model could enable analysis of large data sets and more detailed asteroid shape modeling. Future research can build on these methods to construct a more comprehensive understanding of asteroid properties and dynamics. Improvements such as parameter optimization, incorporation of additional shape models, and machine learning integration could further enhance the capabilities.

Overall, this study presents an important advancement in an area of significance to planetary science and solar system dynamics. The results highlight the potential for synergistic approaches that combine robust optimization algorithms, parallel computing, and simplified shape modeling to reveal new insights from asteroid lightcurve data. Further development of these techniques promises to accelerate our understanding of these small bodies and their interactions within our solar system.

Acknowledgments

The authors would like to thank the anonymous reviewers for the helpful comments. This work was supported by the Characteristic innovation project of Guangdong Provincial Department of Education (No. 2023KTSCX195), Scientific Computing Research Innovation Team of Guangdong Province (No. 2021KCXTD052), Guangdong Key Construction Discipline Research Capacity Enhancement Project (No. 2022ZD JS049), Technology Planning Project of Shaoguan (No. 230330108034184) and Science and Technology Development Fund, Macau SAR (No. 0096/2022/A).

References

Bottke, W. F., Jr, Durda, D. D., Nesvorný, D., et al. 2005, Icarus, 175, 111
Cellino, A., Hestroffer, D., Lu, X.-P., Muinonen, K., & Tanga, P. 2019, A&A, 631, A67
Cellino, A., Muinonen, K., Hestroffer, D., & Carbognani, A. 2015, PSS, 118, 221
Cellino, A., Pannunzio, R., Zappala, V., Farinella, P., & Paolicchi, P. 1985, A&A, 144, 355
Cellino, A., Zappalà, V., Di Martino, M., Farinella, P., & Paolicchi, P. 1987, Icarus, 70, 546
Cellino, A., Zappalà, V., & Farinella, P. 1989, Icarus, 78, 298
Ďurech, J., Kaasalainen, M., Herald, D., et al. 2011, Icarus, 214, 652
Ďurech, J., Sidorin, V., & Kaasalainen, M. 2010, A&A, 513, A46
Hanuš, J., Marchis, F., & Ďurech, J. 2013, Icarus, 226, 1045
Hapke, B. 1984, Icarus, 59, 41
Ji, J., Tan, D., Bao, C., et al. 2023, RAA, 23, 125015
Kaasalainen, M. 2001, A&A, 376, 302
Kaasalainen, M., & Lamberg, L. 2006, IP, 22, 749
Kaasalainen, M., Lamberg, L., Lumme, K., & Bowell, E. 1992a, A&A, 259, 318
Kaasalainen, M., Lamberg, L., & Lumme, K. 1992b, A&A, 259, 333
Kaasalainen, M., Lu, X., & Vänttinen, A.-V. 2012, A&A, 539, A96
Kaasalainen, M., & Torppa, J. 2001, Icarus, 153, 24
Kaasalainen, M., Torppa, J., & Muinonen, K. 2001, Icarus, 153, 37
Kaasalainen, M., Torppa, J., & Piironen, J. 2002, Icarus, 159, 369
Kaasalainen, S., Kaasalainen, M., & Piironen, J. 2005, A&A, 440, 1177
Kaasalainen, S., Piironen, J., Kaasalainen, M., et al. 2003, Icarus, 161, 34
Karttunen, H. 1989, A&A, 208, 314
Karttunen, H., & Bowell, E. 1989, A&A, 208, 320
Li, X.-R., Zhang, Y.-X., Zhao, H.-B., Zhang, X., & Lu, X.-P. 2023, RAA, 23, 055007
Lu, X., Zhao, H., & You, Z. 2014, EMP, 112, 73
Lu, X.-P., Cellino, A., Hestroffer, D., & Ip, W.-H. 2016, Icarus, 267, 24
Lu, X.-P., Huang, X.-J., Ip, W.-H., & Hsia, C.-H. 2018, PSS, 153, 1
Lu, X.-P., & Ip, W.-H. 2015, PSS, 108, 31
Lu, X.-P., Ip, W.-H., Huang, X.-J., & Zhao, H.-B. 2017, PSS, 135, 74
Lu, X.-P., & Jewitt, D. 2019, AJ, 158, 220
Lumme, K., & Bowell, E. 1981a, AJ, 86, 1694
Lumme, K., & Bowell, E. 1981b, AJ, 86, 1705
Lumme, K., Karttunen, H., & Bowell, E. 1990, A&A, 229, 228
Martikainen, J., Muinonen, K., Penttilä, A., Cellino, A., & Wang, X.-B. 2021, A&A, 649, A98
Muinonen, K., & Lumme, K. 2015, A&A, 584, A23
Muinonen, K., Torppa, J., Wang, X.-B., Cellino, A., & Penttilä, A. 2020, A&A, 642, A138
Muinonen, K., Uvarova, E., Martikainen, J., et al. 2022, FASS, 9, 821125
Muinonen, K., Wilkman, O., Cellino, A., Wang, X., & Wang, Y. 2015, PSS, 118, 227
Nathues, A., Mottola, S., Kaasalainen, M., & Neukum, G. 2005, Icarus, 175, 452
Russell, H. N. 1906, ApJ, 24, 1
Surdej, J., & Surdej, A. 1978, A&A, 66, 31
Tian, J., Zhao, H.-B., & Li, B. 2022, RAA, 22, 125004
Vernazza, P., Ferrais, M., Jorda, L., et al. 2021, A&A, 654, A56
Viikinkoski, M., Hanuš, J., Kaasalainen, M., Marchis, F., & Ďurech, J. 2017, A&A, 607, A117
Zhang, Y.-X., Lu, X.-P., Guo, T., & Li, X.-R. 2023, ASR, 71, 3479