Research Article

Fanming Liu*, Fangming Li, and Xin Jing

INS/gravity gradient aided navigation based on gravitation field particle filter

<https://doi.org/10.1515/phys-2019-0073>

Received May 12, 2019; accepted Sep 02, 2019

Abstract: Swarm intelligence method is an effective way to improve the particle degradation and sample depletion of the traditional particle filter. This paper proposes a particle filter based on the gravitation field algorithm (GF-PF), and the gravitation field algorithm is introduced into the resampling process to improve particle degradation and sample depletion. The gravitation field algorithm simulates the solar nebular disk model, and introduces the virtual central attractive force and virtual rotation repulsion force between particles. The particles are moved rapidly to the high-likelihood region under action of the virtual central attractive force. The virtual rotation repulsion force makes the particles keep a certain distance from each other. These operations improve estimation performance, avoid overlapping of particles and maintain the diversity of particles. The proposed method is applied into INS/gravity gradient aided navigation, by combining the sea experimental data of an inertial navigation system. Compared with the particle swarm optimization particle filter (PSO-PF) and artificial physics optimized particle filter (APO-PF), the GF-PF has higher position estimate accuracy and faster convergence speed with the same experimental conditions.

Keywords: gravitation field, solar nebular disk mode, particle filter, gravity gradient, virtual forces

PACS: 02.30.Yy, 02.70.Uu, 11.10.Lm, 91.50.-r, 91.10.-v

1 Introduction

The inertial navigation system (INS) is the mainly navigation system of underwater vehicle. The positioning error of the INS is accumulated and diverged over time, and other positioning methods are needed to correct systematic positioning errors periodically [1–6]. When GPS, radio location and terrain aided navigation are used to correct the positioning error of INS on underwater vehicle, there are disadvantages that require underwater vehicle needs to transmit or receive signals, which destroy the concealment of the underwater vehicle. Gravity gradient aided navigation has the ability to correct INS positioning error [7]. While gravity gradiometer obtaining the gravity gradient data, the underwater vehicle need not transmit or receive signals. The underwater vehicle can be autonomous positioning even in the special case of GPS or radio location failure.

The principle of INS/gravity gradient aided navigation is to use matching method to achieve the optimal approximation from the INS indicated position to the real position of the underwater vehicle [8, 9]. Matching algorithm is one of the core technologies of gravity/gravity gradient aided navigation system. Numerous scholars have applied TERCOM algorithm [10], ICCP algorithm [11, 12] and SITAN algorithm [13] to gravity/gravity gradient aided navigation, and achieved well results. However, these algorithms have some shortcomings and need to be strengthened for higher positioning accuracy. If the initial positioning error exceeds the allowable range, the matching of ICCP algorithm will fail [14]. It is difficult to establish an accurate model for SITAN algorithm [15]. And the real-time performance and computational complexity of TERCOM algorithm need to be further improved [16]. Particle filter is commonly used in non-linear and non-Gaussian systems such as location tracking and fault diagnosis [17–19]. Traditional particle filter methods have defects such as particle degradation and sample depletion. In recent years, some scholars have combined swarm intelligence method with particle filter to improve these problems. For example, artificial physical optimization particle filter [20], chicken swarm optimization particle filter [21], firefly algorithm optimization particle filter [22], bat algorithm optimization particle filter

*Corresponding Author: **Fanming Liu:** College of Automation, Harbin Engineering University, Harbin, China; Email: hrblfm407@yeah.net

Fangming Li: College of Automation, Harbin Engineering University, Harbin, China; Email: lifangming_2008@163.com

Xin Jing: College of Automation, Harbin Engineering University, Harbin, China; Email: jingxin@hrbeu.edu.cn

ter [23], etc. Swarm intelligence method regards particles as individuals in a biological population. It makes the distribution of particles more reasonable by simulating the movement of biological clusters. It does not involve the direct abandonment of low-weight particles, which can effectively solve the problems of particle degradation and simple exhaustion. Some scholars applied particle filter algorithm to gravity/gravity gradient aided navigation, and obtain good positioning results. LIU [20, 24] proposes an artificial optimization particle filter algorithm, and applied it to gravity / gravity gradient aided navigation system. It gets a positioning accuracy of 133.6m. WANG [25] proposes a PF-based matching algorithm with gravity sample vector, and the CEP is 0.42'. LIU [26] proposes a self-adaptive artificial physics optimized particle filter, and the positioning accuracy is 86.9m.

In this paper, a gravitation field particle filter algorithm (GF-PF) is proposed. Through the interactive virtual force between particles, particles aggregate into the high-likelihood region, the posterior distribution of state is optimized, the sample dilution problem is solved, and the accuracy of the particle filter is improved. The method is applied to INS/gravity gradient aided navigation system. Compared with particle swarm optimization particle filter(PSO-PF) [27] and artificial physics optimized particle filter (APO-PF), the simulation results show that the method proposed in this paper can obtain higher positioning accuracy and faster convergence speed under different test conditions. It can be used to correct the position error of the INS.

The rest of the paper is organized as follow: In Section 2, the principle of INS/gravity gradient aided navigation is explained; in Section 3, the particle filter based on gravitation field algorithm is described; in Section 4, the particle filter based on gravitation field algorithm model is applied to INS/gravity gradient aided navigation and compared with PSO-PF and APO-PF to verify the performance of the method proposed in this paper under different test conditions. Finally, the Section 5 is a summary of the full paper.

2 INS/Gravity gradient aided navigation

2.1 Principle of INS/gravity gradient aided navigation

The gravity gradient is the full spatial derivative of gravity acceleration. Due to the gravity gradient symmetry and

Figure 1: Structure for INS/gravity aided navigation.

the Poisson equation, there are total five independent tensors of gravity gradient components in different directions, which provide accurate and reliable reference maps for gravity gradient aided navigation [28]. INS/gravity gradient aided navigation is a navigation method that matches the real-time measured gravity gradient data with the pre-stored high-precision gravity gradient maps which can get underwater vehicle position information real-time. The system structure is shown in Figure 1.

During the underwater vehicle navigation, the matching algorithm searches the gravity gradient data from the gravity gradient maps according to the INS indicated position, and matches with the data from gravity gradiometer to get the accurate position of the underwater vehicle. The position is used to revise the positioning error of the INS.

2.2 INS error equation

The INS error equation is usually used as part of integrated navigation model, which is the basis of error estimation and correction of the INS. The INS error equation includes position error equation, velocity error equation and attitude angle error equation. The position error equation is as follow.

$$\begin{cases} \delta\dot{\varphi} = \frac{1}{R_M} \delta V_y. \\ \delta\dot{\lambda} = \frac{V_x}{R_N} \tan \varphi \sec \varphi \delta\varphi + \frac{\sec \varphi}{R_N} \delta V_x. \end{cases} \quad (1)$$

Where, $\delta\varphi$ and $\delta\lambda$ are latitude error and longitude error, respectively. R_M and R_N are meridian curvature radius and

unitary circle curvature radius.

$$\begin{cases} \delta \dot{V}_x = (2\Omega \cos \varphi V_y + \frac{V_x V_y}{R_N} \sec^2 \varphi) \delta \varphi \\ + \frac{V_y}{R_N} \tan \varphi \delta V_x + 2\Omega \sin \varphi V_y \\ + \frac{V_x}{R_N} \tan \varphi \delta V_y + A_y \gamma - g\beta + \Delta A_x. \\ \delta \dot{V}_y = -(2\Omega \cos \varphi V_x + \frac{V_x^2}{R_N} \sec^2 \varphi) \delta \varphi \\ - 2(\Omega \sin \varphi + \frac{V_x}{R_N} \tan \varphi) \delta V_x \\ + g\alpha - A_x \gamma + \Delta A_y. \end{cases} \quad (2)$$

The velocity error is shown in Eq. (2). δV_x and δV_y are east velocity error and north velocity error, respectively. Ω is the earth rotation rate, ΔA_x and ΔA_y are the accelerometer constant bias.

α , β and γ are the angles between the platform system p and the geographic n . The attitude angle error equations are as follows.

$$\begin{cases} \dot{\alpha} = -\frac{\delta V_y}{R_M} + (\Omega \sin \varphi + \frac{V_x}{R_N} \tan \varphi) \beta \\ - (\Omega \cos \varphi + \frac{V_x}{R_N}) \gamma + \varepsilon_x. \\ \dot{\beta} = -\Omega \sin \varphi \delta \varphi + \frac{\delta V_x}{R_N} \\ - (\Omega \sin \varphi + \frac{V_x}{R_N} \tan \varphi) \alpha - \frac{V_y}{R_M} \gamma + \varepsilon_y. \\ \dot{\gamma} = (\Omega \cos \varphi + \frac{V_x}{R_N} \sec^2 \varphi) \delta \varphi + \frac{\tan \varphi}{R_N} \delta V_x \\ + (\Omega \cos \varphi + \frac{V_x}{R_N}) \alpha + \frac{V_y}{R_M} \beta + \varepsilon_z. \end{cases} \quad (3)$$

ε_x , ε_y and ε_z are gyro constant drifts.

The gyro errors and accelerometer errors after calibration are approximately random constants and Gaussian white noises. The gyro drift error is as follow.

$$\varepsilon_i = \varepsilon_{bi} + w_{gi}, \quad i = x, y, z \quad (4)$$

w_{gi} ($i = x, y, z$) is Gaussian white noise with zero mean. The gyro random constant drift error is as follow.

$$\dot{\varepsilon}_{bi} = 0, \quad i = x, y, z \quad (5)$$

The accelerometer bias error is as follow.

$$\Delta A_i = \Delta A_{bi} + w_{ai}, \quad i = x, y \quad (6)$$

w_{ai} ($i = x, y$) is Gaussian white noise with zero mean. The accelerometer random constant error is as follow.

$$\Delta \dot{A}_{bi} = 0, \quad i = x, y \quad (7)$$

2.3 INS/gravity gradient aided navigation filter model

2.3.1 State equation

According to the INS error equations, taking the East-North-Up coordinate as navigation coordinate. The position errors, velocity errors, platform attitude errors, accelerometer bias errors and gyro drift errors are selected as the state vectors of the INS/gravity gradient aided navigation filter model.

$$x(t) = [\delta \varphi \quad \delta \lambda \quad \delta V_x \quad \delta V_y \quad \alpha \quad \beta \quad \gamma \quad \Delta A_x \quad \Delta A_y \quad \varepsilon_x \quad \varepsilon_y \quad \varepsilon_z]^T. \quad (8)$$

The state equation of the INS/gravity gradient aided navigation is as follow.

$$\dot{x}(t) = A(t)x(t) + B(t)w(t). \quad (9)$$

The system state transition matrix A can be derived from Eq. (1) ~ (7).

$$A = \begin{bmatrix} A_{11} & A_{12} & 0_{2 \times 2} & 0_{2 \times 1} & 0_{2 \times 5} \\ A_{21} & A_{22} & A_{23} & A_{24} & I_{5 \times 5} \\ 0_{5 \times 12} \end{bmatrix}. \quad (10)$$

Where, I is unit matrix, $0_{2 \times 2}$ is zero matrix. A_{11} , A_{12} , A_{21} , A_{22} , A_{23} , A_{24} are as shown below.

$$A_{11} = \begin{bmatrix} 0 & 0 \\ \frac{V_x}{R_N} \tan \varphi \sec \varphi & 0 \end{bmatrix}.$$

$$A_{12} = \begin{bmatrix} 0 & \frac{1}{R_M} \\ \frac{\sec \varphi}{R_N} & 0 \end{bmatrix}.$$

$$A_{21} = \begin{bmatrix} 2\Omega \cos \varphi V_y + \frac{V_x V_y}{R_N} \sec^2 \varphi & 0 \\ -2\Omega \cos \varphi V_x - \frac{V_x^2}{R_N} \sec^2 \varphi & 0 \\ 0 & 0 \\ -\Omega \sin \varphi & 0 \\ \Omega \cos \varphi + \frac{V_x}{R_N} \sec^2 \varphi & 0 \end{bmatrix}.$$

$$A_{22} = \begin{bmatrix} \frac{V_y \tan \varphi}{R_N} & 2\Omega \sin \varphi + \frac{V_x}{R_N} \tan \varphi \\ -2\Omega \sin \varphi - \frac{2V_x}{R_N} \tan \varphi & 0 \\ 0 & -\frac{1}{R_M} \\ \frac{1}{R_N} & 0 \\ \frac{\tan \varphi}{R_N} & 0 \end{bmatrix}.$$

$$A_{23} = \begin{bmatrix} 0 & -g & \\ g & 0 & \\ 0 & \Omega \sin \varphi + \frac{V_x}{R_N} \tan \varphi & \\ -\Omega \sin \varphi - \frac{V_x}{R_N} \tan \varphi & 0 & \\ \Omega \cos \varphi + \frac{V_x}{R_N} & \frac{V_y}{R_M} & 0 \end{bmatrix}.$$

$$A_{24} = \begin{bmatrix} A_y & \\ -A_x & \\ -\Omega \cos \varphi - \frac{V_x}{R_N} & \\ -\frac{V_y}{R_M} & \\ 0 & \end{bmatrix}.$$

The system noise $w(t)$ is as follows.

$$w(t) = [0 \quad 0 \quad w_{ax} \quad w_{ay} \quad w_{gx} \quad w_{gy} \quad w_{gz} \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T. \quad (11)$$

Where, w_{ax} , w_{ay} , w_{gx} , w_{gy} , w_{gz} are Gaussian white noise with zero mean, respectively. Corresponding to accelerometer bias errors and gyro random drift errors. System noise matrix is $B = I_{12 \times 12}$.

Suppose the filter period is T , and the discrete state equations as follows.

$$X_{k+1} = A_k X_k + B_k W_k. \quad (12)$$

Where,

$$A_k = I + TA(t). \quad (13)$$

$$B_k = T(I + \frac{T}{2}A(t))B(t). \quad (14)$$

2.3.2 Observe equation

Taking the gravity gradient data as system observation, and the gravity gradient measurement data has a corresponding relationship with the underwater vehicle position, the observation equation of INS/gravity gradient aided navigation system is as follow.

$$y_{jk} = \Gamma_{jk}(\varphi, \lambda) + v. \quad (15)$$

Where, y_{jk} is gravity gradient measurement data, which are 5×1 dimensions, and the subscript $y, k = x, y, z$ represents the spatial direction; (φ, λ) represents the real position of the underwater vehicle; $\Gamma_{jk}(\varphi, \lambda)$ represents the gravity gradient data at the real position; v is the observing Gaussian white noise.

Eq. (15) is difficult to express with a precise analytic function, which lead to the inability of the observation equation to be modeled. Since the particle filter estimates the probability distribution of the states by discrete randomly sampled particles and their weights, it is not necessary to obtain the specific form of the observation equation, and the processing can be processed by obtaining the correspondence between them. During the navigation of the underwater vehicle, the real position (φ, λ) is not available, and it can be replaced by the INS indicated position $(\hat{\varphi}, \hat{\lambda})$ and the position errors $(\delta\varphi, \delta\lambda)$. The observation equation can be expressed as follow.

$$y_{jk} = \Gamma_{jk}(\hat{\varphi} + \delta\varphi, \hat{\lambda} + \delta\lambda) + v. \quad (16)$$

3 Particle filter based on gravitation field algorithm

3.1 Particle filter

Particle filter is an approximate Bayesian filter method based on sequential Monte Carlo. The method approximates the current probability distribution by using a random sequence composed of finite particles and their weight, and continuously updates and recurs according to the particle filter algorithm.

The state equation and the observation equation of the nonlinear system are as follow.

$$\begin{cases} x_k = f(x_{k-1}, w_k). \\ y_k = h(x_k, v_k). \end{cases} \quad (17)$$

Where, x_k is the state vector; y_k is the observation vector; w_k is the process noise; v_k is the observation noise. At the k time, particle filter obtains a new particle set by predictive sampling.

$$P_k = \{(x_k^i, w_k^i) | i = 1, \dots, N\}. \quad (18)$$

Using the Eq. (19) to approximate the posterior probability density at k time.

$$P(x_k | y_{1:k}) \approx \sum_{i=1}^N \omega_k^i \delta(x_k - x_k^i). \quad (19)$$

Where, N is the numbers of particles, $\delta(\cdot)$ is the Dirichlet function, x_k^i and ω_k^i are the i th particle and its normalized weight, respectively. $y_{1:k}$ is all observations before the k time. According to the principle of sequential importance sampling, the update importance weight equation is as follow.

$$\omega_k^i \propto \omega_{k-1}^i \frac{p(y_k | x_k^i) \cdot p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, y_k)}. \quad (20)$$

State estimation equation.

$$x_k = \sum_{i=1}^N \omega_k^i x_k^i. \quad (21)$$

At the $k = 0$ time, particle filter initializes the sample set, and determines the prior probability of the particle state and initial value. At the next moment, particles update the state according to the system state transition equation, and calculate the weight of all particles according to the observations. Then, the output of posterior probability is obtained.

3.2 Gravitation field algorithm

The gravitation field algorithm(GFA) proposed in this paper is a new heuristic search algorithm which simulates the solar system nebula model [29–31]. The GFA mainly includes three steps, calculation the attractive force from the central dust, calculation the rotation repulsive force and position update. The virtual force model used in the GFA does not have fixed form. The GFA is described as follows.

3.2.1 The attractive force of central dust

In order to move the dust to the high-likelihood region, it is prescribed that all the dust is attracted by the central dust x_{best} , and the rest dust do not generate attractive force at the k time. The virtual mechanical model is as follow

$$P = K_a d_i (x_{best} - x_i). \quad (22)$$

Where, P represents the virtual attractive force of x_{best} on x_i ; K_a is the attractive force coefficient used to adjust the attractive force strength; d_i is the Euclidean distance between x_i and x_{best} .

3.2.2 The rotation repulsive force

While all the dust is attracted by the central dust, all the dust generates rotation repulsive force to the dust within the perceived range, so that the optimized objects are sparse and avoid excessive concentration. The virtual mechanical model is as follows [32].

$$F_{ij} = K_r \left(\frac{1}{d_{ij}^2} - \frac{1}{d_{th}^2} \right) \frac{(x_j - x_i)}{d_{ij}}. \quad (23)$$

Where, F_{ij} is the virtual rotation repulsive force of x_j on x_i , d_{th} is the perceived radius, K_r is the rotation repulsive

force coefficient used to adjust the rotation repulsive force strength, and d_{ij} is the Euclidean distance between x_i and x_j .

3.2.3 Position update

The resultant force of the individual x_i is defined as the vector sum of all the virtual forces.

$$F_i = \sum F_{ij} + P. \quad j = 1, 2, \dots, n, j \neq i \quad (24)$$

Where, n is the amount of dust. x_i complete one iteration and update position by Eq. (25).

$$x'_i = x_i + F_i. \quad (25)$$

Where, x'_i is the updated position. Its value is limited by the minimum displacement and maximum displacement, namely $x'_i \in [L_{min}, L_{max}]$. The GFA terminates after the optimization condition or maximum iterations is satisfied.

3.3 Particle filter based on gravitation field algorithm

Traditional particle filter resampling method avoids particles impoverishment by duplicating large weight particles and deleting smaller weight particles. After iterations, it brings the problem of particle diversity dilution. To alleviate the above problems, the gravitation field algorithm is introduced into the resampling process of particle filter. The virtual attractive force makes the particles rapidly concentrate to the optimal particle. At the same time, the virtual rotation repulsive force makes the optimization particles avoid overlapping or crowding, guarantees the diversity of particles and the distribution of the posterior probability density. Therefore, the GFA has strong global optimization ability and faster convergence speed. After prediction, GF-PF treats candidate particle set as dust in the nebula model. Through the virtual force between the dust, a new suggested distribution is generated, and then the weight of the new particle set is updated and resampled. The computational steps of the GFA are as follows, and the flow chart is shown in Figure 2.

Step 1. Initialization. Sampling $x_0^i |_{i=1}^N$ from $p(x_0)$, the weight ω_0^i is set to $\frac{1}{N}$. Set the perception radius d_{th} , virtual force coefficient K_a and K_r , maximal iteration number EndMax.

Step 2. Prediction. Sample new particle set from $x_k^i \sim q(x_k^i | x_{k-1}^i, y_k)$ and calculate the weights of particles.

Step 3. Optimize particle distribution.

Calculate the Euclidean distance $dist(s_i, s_j)$ between particles.

The particle with the largest weight is regarded as the center dust, and the virtual attractive force from center dust to the surrounding dust is calculated by Eq. (22).

Eq. (23) is used to calculate the virtual rotation repulsive force of dust on the other dust within the perceived range.

Eq. (24) is used to calculate the resultant force F_i of particles.

The position of particles is updated by Eq. (25).

Step 4. Get new particle set, and complete iterative. Recalculate new particle's weight and normalize.**Step 5.** Resampling. If the number of valid particle N_{eff} is less than the set threshold N_{th} , resample the particle filter algorithm and return to $\{\tilde{x}_k^t, \tilde{\omega}_k^t\}_{i=1}^N$.**Step 6.** State Estimation.

$$\hat{x}_k^i = \sum_{i=1}^N \tilde{\omega}_k^i \tilde{x}_k^i$$

4 Experiment

4.1 Track and gravity gradient maps used in the test

The INS output period is 1s, data recording is 7 hours, gyro constant drift errors are $\varepsilon_x = \varepsilon_y = \varepsilon_z = 0.01^\circ/h$, accelerometer bias errors are $\Delta A_x = \Delta A_y = 100\mu g$, platform initial error angles are $\alpha = 0.1^\circ$, $\beta = 0.1^\circ$ and $\gamma = 0.5^\circ$, respectively. Due to the errors of gyros and accelerometers, the position errors of INS increases with time, as showed in Figure 3.

GPS and INS tracks are shown in Figure 4. The Pentagon is the starting position of GPS record. The underwater vehicle first sails southeast. After 2 hours, the underwater vehicle turns to the northwest.

Gravity gradient maps are obtained by forward modeling of free air gravity anomaly data. The resolution is $30'' \times 30''$, including five independent tensors, such as Γ_{xx} , Γ_{xy} , Γ_{xz} , Γ_{yy} and Γ_{yz} . Data characteristics of gravity gradient maps are shown in Table 1.

During the simulation, the real-time gravity gradient data is simulated by superimposing Gaussian white noise on the gravity gradient data at the actual position of the underwater vehicle (provided by GPS). Data that is not on the grid is obtained by bilinear interpolation.

Table 1: Data statistics of gravity gradient maps

	Min(E)	Max(E)	Mean(E)	Std(E)
Γ_{xx}	-41.6567	47.5624	-0.2495	4.4941
Γ_{xy}	-30.1590	33.1530	-0.0698	3.0714
Γ_{xz}	-69.2312	72.2177	0.2195	5.9581
Γ_{yy}	-54.4439	60.6621	-0.6259	4.7291
Γ_{yz}	-92.4407	66.4486	-0.7001	5.6174

Figure 2: Flow chart of particle filter based on gravitation field algorithm.

4.2 Application and analysis of GF-PF in INS/gravity gradient aided navigation

The GF-PF is applied to INS/gravity gradient aided navigation and compared with particle swarm optimization particle filter and artificial physics optimized particle filter [20] to verify the performance.

Figure 3: Position errors of inertial navigation system.

Figure 4: GPS and INS tracks.

Simulation conditions: INS running alone for 2 hours, and then introduce INS/gravity gradient aided navigation. The initial position errors of GF-PF, PSO-PF and APO-PF is 2000m, the population of particles is 100, the maximum number of iterations is 20, and the filter period is 10s. The gyro constant drift errors and accelerometer bias errors are taken as the system noises and the observation noise is $\sigma_v^2 = 1E$. GF-PF parameters setting as: perceived radius is $d_{th} = 20m$, attractive force coefficient is $K_a = 10$, rotation repulsive force coefficient is $K_r = 1$, maximum displacement is $L_{max} = 30m$, minimum displacement is $L_{min} = L_{max}$. The initial parameters of the PSO-PF are set as follows [33]: inertial weight is 0.99, acceleration constant is $c_1 = c_2 = 0.1$. The initial parameters of the APO-PF are set as follows [20]: attractive force coefficients are $K_{a1} = 10$, $K_{a2} = 1$, repulsive force coefficient is $K_b = 0.9$, perceived radius is $r_s = 150m$, threshold value is $D_{th} = 10m$.

10 independent simulation experiments of gravity gradient aided navigation are carried out using PSO-PF, APO-

PF and GF-PF respectively. Figure 5 shows the tracks of the real track (GPS), INS track, PSO-PF matching track, GF-PF matching track and APO-PF matching track after INS running alone for 2 hours.

The matching tracks of GF-PF, PSO-PF and APO-PF all track the real track well. PSO-PF and GF-PF have faster convergence speed, and APO-PF has the slowest convergence speed.

The root mean square error (RMSE) of latitude error and longitude error at time is as follow.

$$RMSE_t = \left(\frac{1}{10} \sum_{m=1}^{10} \|(\hat{\phi}, \hat{\lambda})_t^{GFPF, (m)} - (\phi, \lambda)_t^{GPS, (m)}\|^2 \right)^{\frac{1}{2}}. \quad (26)$$

Where, $(\hat{\phi}, \hat{\lambda})_t^{GFPF, (m)}$ is the underwater vehicle longitude error and latitude error of the GF-PF at the t time obtained in the m th experiment, and $(\phi, \lambda)_t^{GPS, (m)}$ is the real latitude and longitude of the underwater visual at the t time.

After 10 independent simulations using GF-PF, PSO-PF and APO-PF respectively, the RMSE of the longitude and latitude errors is shown in Figure 6.

When the INS/gravity gradient aided navigation start working, the INS longitude error is about 414m, and the latitude error is about 1318m. At about 4th hour, the latitude error of the APO-PF has a large error. The latitude and longitude errors of PSO-PF are relatively large. Between the 3rd hour and the 7th hour, the RMSE of the PSO-PF algorithm is 214.23m, the RMSE of the APO-PF is 128.56m, and the RMSE of the GF-PF is 48.15m. The convergence speed of GF-PF and PSO-PF is almost same, which is better than APO-PF. Compared with the PSO-PF and APO-PF, the performance of GF-PF is more stable, and the positioning accuracy is higher.

4.3 Compare the effects of observation noise on the performance of each algorithm

Increase the observation noise to $\sigma_v^2 = 5E$, and other conditions and track remain same. The performance of each algorithm is compared. After 10 independent simulations using GF-PF, PSO-PF and APO-PF respectively, the RMSE of the longitude and latitude errors is shown in Figure 7.

At about 4th hour, the latitude error of the APO-PF has a large error. PSO-PF has the largest latitude error fluctuation. Between the 6th hour and the 7th hour, the longitude error and latitude error of the PSO-PF and APO-PF shows a large error. The latitude error of the GF-PF has a greatly fluctuates, but the longitude error maintains a high positioning accuracy. Between the 4th hour and the 7th hour, the

(a) PSO-PF

(b) APO-PF

(c) GF-PF

Figure 5: PSO-PF, APO-PF and GF-PF matching tracks.

Figure 6: Position error curves of PSO-PF, APO-PF and GF-PF.

Figure 7: Position error curves of PSO-PF, APO-PF and GF-PF (map resolution $30'' \times 30''$, gradient measure accuracy $5E$).

latitude error of GF-PF is higher than APO-PF in part time. Between the 3rd hour and the 7th hour, the RMSE of the PSO-PF algorithm is 1303.01m, the RMSE of the APO-PF is 482.66m, and the RMSE of the GF-PF is 376.13m. Compared with the PSO-PF and APO-PF, the performance of GF-PF is more stable, and the positioning accuracy is higher. After increasing the observation noise to $\sigma_v^2 = 5E$, the convergence speed of GF-PF is slower than that of PSO-PF, but faster than APO-PF.

4.4 Compare the effects of gravity gradient maps resolution on the performance of each algorithm

Gravity gradient maps with resolution $1' \times 1'$ are obtained by interval point extraction, and other conditions and

Figure 8: Position error curves of PSO-PF, APO-PF and GF-PF (map resolution $1' \times 1'$, gradient measure accuracy $1E$).

track remain same. The performance of each algorithm is compared. After 10 independent simulations using GF-PF, PSO-PF and APO-PF respectively, the RMSE of the longitude and latitude errors is shown in Figure 8.

Between the 6th hour and the 7th hour, the latitude error of the PSO-PF and APO-PF shows a large error. The latitude error of the APO-PF maintains a stable effect, but the positioning error is higher than the GF-PF algorithm. Between the 4th hour and the 5th hour, the longitude error of the PSO-PF and GF-PF shows a large error. At about 6th hour, the longitude error of APO-PF has a large error. At about 7th hour, the longitude error of PSO-PF has a large error. Between the 3rd hour and the 7th hour, the RMSE of the PSO-PF algorithm is 462.27m, the RMSE of the APO-PF is 325.08m, and the RMSE of the GF-PF is 140.70m. Compared with the PSO-PF and APO-PF, the performance of GF-PF is more stable, the convergence speed is faster, and the positioning accuracy is higher.

5 Conclusion

Particle filter can avoid the linearization problem in INS / gravity gradient aided navigation. The intelligent optimization algorithm can effectively solve the particle degradation and sample depletion problems of the traditional particle filter algorithm. In this paper, the gravitation field algorithm is introduced into the resampling of the particle filter. The virtual forces are used to optimize the particle distribution, maintain the particle diversity, and form a uniform coverage capability for the posterior probability density. Applied the GF-PF to INS/gravity gradient aided navigation and compares it with PSO-PF and APO-PF. The

results show that the positioning accuracy of GF-PF is higher than that of PSO-PF and APO-PF. The convergence speed of GF-PF and PSO-PF is almost same, which is better than APO-PF. Under different test conditions of observation noise and gravity gradient map resolutions, compared with PSO-PF and APO-PF, GF-PF still maintains the highest positioning accuracy and faster convergence speed. The performance of GF-PF is less affected by observation noise and gravity gradient map resolutions than that of PSO-PF and APO-PF.

Acknowledgement: This paper is supported by Natural Science Foundation of China (No. 6163308).

References

- [1] Yao J.Q., The method of gravity gradient aided positioning based on STUPF, *Navigation Positioning & Timing*, 2018, 5, 60-63.
- [2] Qin W.T., Guo J.F., Cui N.G. and Wang X.G., INS/GGI navigation in arctic area based on high degree cubature kalman filter, *Tactical Missile Technology*, 2018, 6, 85-94.
- [3] Tchier F., Inc M., Kilic B. and Akgül A., On soliton structures of generalized resonance equation with time dependent coefficients, *Optik*, 2017, 128, 218-223.
- [4] Boutarfa B., Akgül A. and Inc M., New approach for the Fornberg-Whitham type equations, *Journal of Computational and Applied Mathematics*, 2017, 312, 13-26.
- [5] Akgül A., New reproducing kernel functions, *Mathematical Problems in Engineering*, 2015, 1-11.
- [6] Akgül A. and Kilicman A., Solving delay differential equations by an accurate method with interpolation, *Abstract and Applied Analysis*, 2015, 1-8.
- [7] Han Y.R., Wang B., Deng Z.H. and Fu M.Y., A combined matching algorithm for underwater gravity aided navigation, *IEEE/ASME Transactions on Mechatronics*, 2018, 23, 233-241.
- [8] Wu L., Wang H.B., Chai H., Zhang L., Hsu H. and Wang Y., Analysis and selection of global marine gravity/gravity gradient aided navigation matching areas, *The 9th China Satellite Navigation Conference*, Harbin, China, 2018, 887-895.
- [9] Inc M., Akgül A. and Kilicman A., Numerical solutions of the second-order one-dimensional telegraph equation based on reproducing kernel Hilbert space method, *Abstract and Applied Analysis*, 2013, 1-14.
- [10] Han Y.R., Wang B., Deng Z.H. and Fu M.Y., An improved TERCOM based algorithm for gravity aided navigation, *IEEE Sensors Journal*, 2016, 16, 2537-2544.
- [11] Wang K.D., Zhu T.Q., Qin Y.J., Jiang R., and Li Y., Matching error of the iterative closest contour point algorithm for terrain-aided navigation, *Aerospace Science and Technology*, 2018, 73, 210-222.
- [12] Akgül A., Hashemi M. S., Inc M. and Raheem S. A., Constructing two powerful methods to solve the Thomas-Fermi equation, *Nonlinear Dynamics*, 2017, 87, 1435-1444.
- [13] Wei E.D., Dong C.J., Yang Y.L. and Tang S.Q., A robust solution of integrated SITAN with TERCOM algorithm: weight-reducing iter-

- ation technique for underwater vehicle's gravity-aided inertial navigation system, *Navigation*, 2017, 64, 111-122.
- [14] Xiao J., Qi X.H. and Duan X.S., Research status of magnetic matching algorithm and its improvement strategies, *Electronics Optics & Control*, 2018, 25, 55-59, 73.
 - [15] Zheng H., Wang Y., Wang H.B. and Wu L., Simulation re- search of earth's gravity and geomagnetism potential field aided under- water navigation, *Geomatics & Information Science of Wuhan University*, 2012, 37, 1198-1202.
 - [16] Zhao L., Gao N., Huang B.Q., Wang Q.Y. and Zhou J.H., A novel terrain-aided navigation algorithm combined with the TERCOM algorithm and particle filter, *IEEE J. Sensors*, 2015, 15, 1124–1131.
 - [17] Seitz J., Vaupel T. and Thielecke J., A particle filter for Wi-Fi az- imuth and position tracking with pedestrian dead reckoning, 2013 Workshop on Sensor Data Fusion: Trends, Solutions, Appli- cations(SDF), Bonn, Germany, 2013, 1-6.
 - [18] Akgül A., Inc M., Kilicman A. and Baleanu D., A new ap- proach for one-dimensional sine-Gordon equation, *Advances in Difference Equations*, 2016, 8, 1-20.
 - [19] Hashemi M. S., Inc M., Kilic B. and Akgül A., On solitons and invariant solutions of the Magneto-electro-elastic circular rod, *Waves in Random and Complex Media*, 2016, 26(3), 259- 271.
 - [20] Liu F.M., Qian D. and Liu C.H., An artificel physics optimized particle filter, *Control and Decision*, 2012, 27, 1145- 1149, 1156.
 - [21] Zhang J.C., Kang F.J., Liang H.T. and Xu H., Research on chicken swarm optimization based particle filter, *Journal of System Sim- ulation*, 2017, 29, 295-300, 308.
 - [22] Gao M.L., Li L.L., Sun X.M., Yin L.J. and Li H.T., Firefly algorithm (FA) based particle filter method for visual tracking, *Optik*, 2015, 126, 1705-1711.
 - [23] Chen Z.M., Tian M.C., Wu P.L., Bo Y.M., Gu F.F., and Yue C., Intelli- gent particle filter based on bat algorithm, *Acta Phys. Sin.*, 2017, 66, 41-50.
 - [24] Liu F.M., Wei X., Qian D. and Li F.M., The application of optimiza- tion particle filter in gravity aided positioning, *Journal of Harbin Institute of Technology*, 2012, 44, 145-148.
 - [25] Wang B., Yu L., Deng Z.H. and Fu M.Y., A particle filter- based matching algorithm with gravity sample vector for underwater gravity aided navigation, *IEEE/ASME TRANSAC- TIONS ON MECHA- TRONICS*, 2016, 21, 1399-1406.
 - [26] Liu F.M. and Li F.M., Application of self-adaptive artificial physics optimized particle filter in INS/gravity gradient aided navigation, *Proceedings of 2018 IEEE International Conference on Mecha- tronics and Automation*, Changchun, China, 2018, 1070-1075.
 - [27] Gao G.D., Lin M. and XU L., New PSO particle filter method based on likelihood-adjustment, *Journal of computer applica- tions*, 2017, 37, 980-985.
 - [28] Wang Z.Y. and Sun F., Rao-Blackwellised Particle Filter and It's Ap- plication in Navigation, *Journal of Projectiles, Rockets, Missiles and Guidance*, 2013, 33, 153-159.
 - [29] Liu F., Li F., Lin N. and Jing X., Gravity aided positioning based on real-time ICCP with optimized matching sequence length, *IEEE Access*, 2019, 7, 97440-97456.
 - [30] Akgül A., Khan Y., Akgul E. K., Baleanu D. and Qurashi M. M. A., Solutions of nonlinear systems by reproducing kernel method, *The Journal of Nonlinear Sciences and Applications*, 2017, 10, 4408-4417.
 - [31] Inc M., Akgül A. and Kilicman A., A novel method for solving KdV equation based on reproducing kernel Hilbert space method, *Abstract and Applied Analysis*, 2013, 1-12.
 - [32] Ding C.Y. and Peng J., A hooping sensor deployment scheme based on virtual force, *Proceedings of 2015 IEEE Conference on Robotics and Biomimetics*, Zhuhai ,China, 2015, 988-993.
 - [33] Bonyadi M. and Michalewicz Z., Impacts of coefficients on move- ment patterns in the particle swarm optimization algorithm, *IEEE Transactions on Evolutionary Computation*, 2017, 21, 378-390.