Underwater Target Tracking Based on Strong Tracking Sparse Grid Quadrature Filter

Chongyang Lv, Fei Yu*, Shu Xiao*, and Qianhui Dong

***Abstract*—Underwater moving object detection/tracking is critical in various applications such as exploration of natural undersea resources, acquiring of accurate scientific data to maintain regular surveillance of missions, navigation and tactical surveillance. In currently, underwater moving target is usually tracked using the traditional non-linear estimators such as Extended Kalman Filter (EKF) and unscented Kalman Filter (UKF). However, if an underwater target moves with delicate maneuver, the accuracy of the filter may decline, even diverge. In this paper, a (STSGQF) is proposed to deal with the problem. The STSGQF is obtained by introducing the Strong Tracking Filter (STF) to the Sparse Grid Quadrature Filter (SGQF). Compared with the Gauss-Hermite Quadrature Filter (GHQF), the sparse grid method is available to reduce the SGQF's computational cost significantly, with slight sacrifice of accuracy its accuracy declines slightly. Meanwhile, the STSGQF has stronger robustness than SGQF against the state change. The effectiveness of STSGQF is demonstrated by the simulation results more robust, better robustness.**

***Index Terms*—Underwater target tracking; Gauss hermite quadrature filter; Sparse grid; Strong tracking filter**

I. Introduction

THE submarine plays a significant role in underwater operation, such as the exploration of marine resource and the demand of military. Detecting, classifying and tracking the underwater target around the submarine were indispensable parts of underwater defense system, which was to ensure the security of submarines [1]. So far, various types of sonar arrays and underwater sensor network had been applied, they were generally mounted on a ship, or deployed prior to the application [1]-[4]. There may be no sufficient flexibility to deal with real-time tracking missions, and the propagation delay of underwater acoustic communication may cause the decline of tracking accuracy [1]. It is important to choose a target tracking algorithm that can track the moving target expeditiously [5]-[8]. Theoretically, the underwater target tracking was a nonlinear estimate process, so the nonlinear filter algorithm is an attractive choice [9]-[11].

The Extended Kalman Filter (EKF) was widely used for state estimation in nonlinear systems [13]-[15]. However, the process of calculating the Jacobian matrix was complex and the nonlinear function must be derivable [14]. Moreover, the EKF may suffer from the problems of performance degrading and diverging in the linearization process [15]. Alcocer et.al proposed the Unscented Kalman Filter (UKF), in which the Unscented Transformation (UT) was adopted to propagate meaning and covariance information [14], [16], [17]. As compared with the EKF, the UKF did not need to calculate the Jacobian matrix and its approximation of posterior distribution can reach 3rd accuracy level [16]. However, the effect of UKF was not good in a high dimensional strong nonlinear system [14]. On the basis of Gauss Hermite Quadrature (GHQ) rule, the Gauss Hermite Quadrature Filter (GHQF) was proposed by Ito et al.[18]-[22]. In theory, the GHQF can reach any order accuracy by selecting the number of the one dimensional quadrature points, but the computational complexity of the GHQF grows exponentially with the system dimension, which was known as "curse of dimensionality" [21], [22]. To solve this problem, the Sparse Grid Quadrature Filter (SGQF) based on the Sparse Grid Quadrature (SGQ) rule was presented [19]-[22]. The SGQ rule is based on a special linear combination of lower level sensor products to extend the one dimensional point set [21]-[23]. Therefore, the number of quadrature points under the SGQ rule was obviously less than that of the GHQ rule [19]. In terms of the point selection strategy, the SGQF was more flexible and it has higher accuracy than UKF [22].

However, if the target state suddenly changes, the accuracy of the proposed filters above (EKF, UKF, GHQF and SGQF) may decline and even diverge due to model errors [14]. The adaptive algorithm is a good strategy to solve the problem. One of them, the Strong Tracking Filter (STF) was proposed by Zhou et al. to overcome the poor robustness[24], in which a fading factor was introduced to provide better state estimation by adjusting the gain matrix in real time [24]-[28], STF had several important merits, including: 1) strong robustness against model uncertainties; 2) better real time state tracking capability, no matter whether the system reaches steady or not; and 3) moderate computational cost[23]-[25]. To improve the accuracy and simplify the computation, a filter called Strong Tracking Unscented Kalman Filter (STUKF) was developed based on the combination of UKF and STF[27]; [28]. The accuracy of SGQF was higher than UKF. In this paper, a novel point based adaptive nonlinear filter algorithm called Strong

This work was supported by the National Natural Science Foundation of China (No. 51679047).

Chongyang. Lv is with the College of Automation, Harbin Engineering University, Harbin 150001, China (e-mail: lvchongyang@hrbeu.edu.cn).

Fei, Yu is with the College of Automation, Harbin Engineering University, Harbin 150001, China

Shu, Xiao is with the College of Automation, Harbin Engineering University, Harbin 150001, China (e-mail: xiaoshu@hrbeu.edu.cn).

Qianhui, Dong is with the College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

* Corresponding author.

Tracking Sparse Grid Quadrature Filter (STSGQF) was proposed. In the STSGQF, the suboptimal fading factor of STF was introduced to time update and measurement update equations of the SGQF to provide better robustness and accuracy.

The rest of the paper was organized as follows: in section II, the GHQF was briefly reviewed and the SGQ rule was introduced by a concrete example. In section III, the STF was reviewed and the STSGQF was derived. Two simulation examples were given in section IV, various kinds of filters were illustrated in weak and strong maneuver conditions, respectively. Conclusions were given in section V.

II. SPARSE GRID QUADRATURE FILTER

The SGQF based on the SGQ rule and the Kalman Filter framework was briefly introduced.

A. Gauss-Hermite Quadrature Filter

Consider a nonlinear system given by

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}) + \boldsymbol{w}_k \tag{1}$$

$$\boldsymbol{z}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_k \tag{2}$$

Where $\boldsymbol{x}_k$ is the state vector and $\boldsymbol{z}_k$ is the measurement vector; $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are independent white Gaussian process noise and measurement noise with covariance $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$, respectively.

The univariate GHQ rule is given by

$$\int f(x)\mathcal{N}(x;0,1)\mathrm{d}x \approx \sum_{i=1}^{s} \omega_i f(\chi_i) \tag{3}$$

where $f(x)$ is the integrand; s is the number of quadrature points; χ_i and ω_i are quadrature points and weights obtained by quadrature rule, respectively.

The univariate points and their weights can be calculated as follows.

First of all, constructing a symmetric tri-diagonal matrix $\boldsymbol{J}$ with zero diagonal elements and $\boldsymbol{J}_{i,i+1} = \sqrt{i/2}$ $(i = 1, 2, \cdots, s-1)$. Then the j-th quadrature point χ_j is calculated by $\chi_j = \sqrt{2}\lambda_j$, where λ_j is the j-th eigenvector of $\boldsymbol{J}$. The corresponding weight ω_j is calculated by $\omega_j = [(\boldsymbol{v}_j)_1]^2$, where $(\boldsymbol{v}_j)_1$ is the first element of the j-th normalized eigenvector of $\boldsymbol{J}$. Finally, the quadrature points and weights are obtained.

The selection of different number of quadrature points are given by Table I.

The multivariate GHQ rule of n-dimensional vector $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{x}; \boldsymbol{0}, \boldsymbol{I}_n)$ is extended by the above univariate GHQ rule as follows.

$$\int \boldsymbol{f}(\boldsymbol{x})\mathcal{N}(\boldsymbol{x};\boldsymbol{0},\boldsymbol{I}_n)\mathrm{d}\boldsymbol{x} \approx \sum_{i_1=1}^{s}\omega_{i_1} f(\chi_{i_1})\cdots\sum_{i_n=1}^{s}\omega_{i_n} f(\chi_{i_n}) = \sum_{i=1}^{s^n}\omega_i \boldsymbol{f}(\boldsymbol{\chi}_i) \tag{4}$$

where $\boldsymbol{\chi}_i = [\chi_{i_1}, \chi_{i_2}, \cdots, \chi_{i_n}]^{\mathrm{T}}$ and $\omega_i = \prod_{p=1}^{n} \omega_{i_p}$ are the i-th quadrature point and its weight, respectively.

TABLE I
QUADRATURE POINTS AND WEIGHTS

Number of points	Point set G and weight set A	Accuracy level
1	$G_1 = \{0\}$ $A_1 = \{1\}$	1
3	$G_2 = \{-p_1, 0, p_1\} = \{-1.7321, 0, 1.7321\}$ $A_2 = \{\omega_2, \omega_1, \omega_2\}$ $= \{0.1667, 0.6667, 0.1667\}$	2
5	$G_3 = \{-p_3, -p_2, 0, p_2, p_3\}$ $= \{-2.8570, -1.3556, 0, 1.3556, 2.8570\}$ $A_3 = \{\omega_5, \omega_4, \omega_3, \omega_4, \omega_5\}$ $= \{0.0113, 0.2221, 0.5333, 0.2221, 0.0113\}$	3

The multidimensional GHQF algorithm is summarized as follows.

1) Initialization

$$\hat{\boldsymbol{x}}_0 = E[\boldsymbol{x}_0], \quad \boldsymbol{P}_0 = \mathrm{cov}(\boldsymbol{x}_0);$$

where $\hat{\boldsymbol{x}}_0$ is the initial state vector; $\boldsymbol{P}_0$ is the initial state covariance matrix.

2) Prediction

$$\boldsymbol{P}_{k-1} = \boldsymbol{S}_{k-1}\boldsymbol{S}_{k-1}^{\mathrm{T}} \tag{5}$$

$$\boldsymbol{\xi}_{i,k-1} = \boldsymbol{S}_{k-1}\cdot\boldsymbol{\chi}_i + \hat{\boldsymbol{x}}_{k-1} \tag{6}$$

$$\hat{\boldsymbol{x}}_{k|k-1} = \sum_{i=1}^{M}\omega_i \boldsymbol{f}(\boldsymbol{\xi}_{i,k-1}) \tag{7}$$

$$\boldsymbol{P}_{k|k-1} = \sum_{i=1}^{M}\omega_i (\boldsymbol{f}(\boldsymbol{\xi}_{i,k-1}) - \hat{\boldsymbol{x}}_{k|k-1})(\boldsymbol{f}(\boldsymbol{\xi}_{i,k-1}) - \hat{\boldsymbol{x}}_{k|k-1})^{\mathrm{T}} + \boldsymbol{Q}_{k-1} \tag{8}$$

where $\boldsymbol{\xi}_{i,k-1}$ is the transformed point; $M = s^n$ is the number of total points; $\hat{\boldsymbol{x}}_{k|k-1}$ and $\boldsymbol{P}_{k|k-1}$ are the predicted state vector and the predicted state covariance matrix, respectively.

3) Update

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{S}_{k|k-1}\boldsymbol{S}_{k|k-1}^{\mathrm{T}} \tag{9}$$

$$\boldsymbol{\xi}_{i,k|k-1} = \boldsymbol{S}_{k|k-1}\cdot\boldsymbol{\chi}_i + \hat{\boldsymbol{x}}_{k|k-1} \tag{10}$$

$$\hat{\boldsymbol{z}}_{k|k-1} = \sum_{i=1}^{M}\omega_i \boldsymbol{h}(\boldsymbol{\xi}_{i,k|k-1}) \tag{11}$$

$$\boldsymbol{P}_{zz,k|k-1} = \sum_{i=1}^{M}\omega_i (\boldsymbol{h}(\boldsymbol{\xi}_{i,k|k-1}) - \hat{\boldsymbol{z}}_{k|k-1})(\boldsymbol{h}(\boldsymbol{\xi}_{i,k|k-1}) - \hat{\boldsymbol{z}}_{k|k-1})^{\mathrm{T}} + \boldsymbol{R}_k \tag{12}$$

$$\boldsymbol{P}_{xz,k|k-1} = \sum_{i=1}^{M}\omega_i (\boldsymbol{f}(\boldsymbol{\xi}_{i,k-1}) - \hat{\boldsymbol{x}}_{k|k-1})(\boldsymbol{h}(\boldsymbol{\xi}_{i,k|k-1}) - \hat{\boldsymbol{z}}_{k|k-1})^{\mathrm{T}} \tag{13}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{xz,k|k-1}\boldsymbol{P}_{zz,k|k-1}^{-1} \tag{14}$$

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_{k|k-1}) \tag{15}$$

$$\boldsymbol{P}_k = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{P}_{zz,k|k-1}\boldsymbol{K}_k^{\mathrm{T}} \tag{16}$$

where $\boldsymbol{\xi}_{i,\mathrm{k}|k-1}$ is the transformed point; $\boldsymbol{P}_{zz,k|k-1}$ and $\boldsymbol{P}_{xz,k|k-1}$ are innovation covariance matrix and cross covariance matrix, respectively; $\hat{\boldsymbol{z}}_{k|k-1}$ is the predicted measurement; $\boldsymbol{K}_k$ is the gain matrix; $\hat{\boldsymbol{x}}_k$ and $\boldsymbol{P}_k$ are the updates of state vector and state covariance matrix, respectively.

B. Sparse grid quadrature rule

As can be seen in the Eq. (4), s^n quadrature points were needed to solve the multivariate numerical integration. The computational cost grows exponentially as dimension increases, hence the "curse of dimension"[21]. The sparse-grid method utilized a linear combination of low-level sensor products to approximate the multivariate integral [18].

The multivariate numerical integral based on sparse-grid method was shown as follows.

$$\begin{aligned} I &= \int f(\boldsymbol{x})\mathcal{N}(\boldsymbol{x};\boldsymbol{0},\boldsymbol{I}_n)\mathrm{d}\boldsymbol{x} \approx I_{n,L}(f) \\ &= \sum_{q=L-n}^{L-1}(-1)^{L-1-n}C_{n-1}^{L-1-n}\sum_{\Phi\in N_q^n}(I_{i_1}\otimes\cdots\otimes I_{i_n})(f) \end{aligned} \tag{17}$$

where $\boldsymbol{x} = [x_1, x_2, \cdots, x_n]^{\mathrm{T}} \in \mathbb{R}^n$; $I_{n,L}(\boldsymbol{f})$ is an approximation to the n-dimensional integral with accuracy level L, which means that $I_{n,L}(\boldsymbol{f})$ is accurate to all the polynomials of the form $x_1^{i_1}x_2^{i_2}\cdots x_n^{i_n}$, and satisfies the condition of $\sum_{j=1}^{n} i_j \le 2L-1$. The binomial coefficient is marked as C_{n-1}^{L-1-n}; The sensor product is marked as $\otimes$. $I_{i_j}(\boldsymbol{f})$ is the j-th univariate integral with the accuracy level of $i_j \in \Phi$, and $\Phi \triangleq (i_1, i_2, \cdots, i_n)$ is an accuracy level sequence. Accuracy level i_j means $I_{i_j}(\boldsymbol{f})$ is accurate up to the $(2i_j - 1)$-th order of the univariate polynomial. N_q^n is the set of accuracy level sequences, it is defined as

$$\begin{cases} N_q^n = \left\{\Phi : \sum_{j=1}^{n} i_j = n+q\right\}, & q \ge 0 \\ N_q^n = \varnothing, & q < 0 \end{cases} \tag{18}$$

where q is a parameter of nonnegative integer and satisfies the condition of $L-n \le q \le L-1$; $\varnothing$ is the empty set.

The point set of the univariate integral $I_{i_j}(\boldsymbol{f})$ is marked as X_{i_j}. The univariate quadrature points in X_{i_j} are selected through the univariate GHQ rule mentioned in section A. $x_s = [x_{s_1}, \cdots, x_{s_n}]^{\mathrm{T}}$ is the s-th quadrature point of the n-dimensional vector and $x_{s_j} \in X_{i_j}$; ω_{s_j} is the corresponding weight in $I_{i_j}(\boldsymbol{f})$ associated with the point x_{s_j}. The set of n-dimensional sparse-grid quadrature points is given by

$$X_{n,L} = \bigcup_{q=L-n}^{L-1}\bigcup_{\Phi\in N_q^n}(X_{i_1}\otimes\cdots\otimes X_{i_n}) \tag{19}$$

where $\bigcup$ is denoted as the union of the quadrature point sets. Each element in N_q^n can build a special sensor product X_{i_j} where $i_j \in \Phi \in N_q^n$.

The SGQ rule can be regarded as that an n-dimensional integral is expanded by the univariate integral through a particular strategy of point selection. The explicit form of the Eq. (17) can be written as

$$\begin{aligned} I_{n,L}(f) = \sum_{q=L-n}^{L-1}\sum_{\Phi\in N_q^n}\sum_{x_{s_1}\in X_{i_1}}\cdots\sum_{x_{s_n}\in X_{i_n}} f(x_{s_1},\cdots,x_{s_n}) \\ \times\left\{(-1)^{L-1-n}C_{n-1}^{L-1-n}\prod_{j=1}^{n}\omega_{s_j}\right\} \end{aligned} \tag{20}$$

The selection strategy of the SGQ points and the corresponding weights is shown as follows.

1) The dimension n and accuracy level L of the multivariate integral are determined at first;

2) Parameter q and the set of accuracy level sequences N_q^n can be calculated by Eq. (18);

3) $\Phi \triangleq (i_1, i_2, \cdots, i_n)$ is obtained, then the SGQ points like x_s are obtained;

4) Finally, calculating the corresponding weight ω_{s_j}. If the point is new, add it to the set of SGQ points χ, and calculate the corresponding weight of χ_s as follows

$$W_s = (-1)^{L-1-q}C_{n-1}^{L-1-q}\prod_{j=1}^{n}\omega_{s_j} \tag{21}$$

If the point already exists, make it stay and update its weight by recursion addition.

To explain the SGQ rule specifically, the process of quadrature points with accuracy level $L=3$ is analyzed. The value of q can be 0, 1 or 2 and the dimension is $n \ge 3$. The univariate quadrature points and weights of different accuracy are shown in TABLE 1.

When $q=0$, $\sum_{j=1}^{n} i_j = n$, so

$$N_0^n = \left\{ \underbrace{(1,1,1,\cdots,1)(1,1,1,\cdots,1)\cdots(1,1,1,\cdots,1)}_{n} \right\}$$

that is to say, all of the n points are the origin, the number of the SGQ points is $X_{q=0} = 1$. By the Eq. (21), the weight is

$$W = (-1)^{3-1-0} \times \mathrm{C}_{n-1}^{3-1-0} \times (\underbrace{1 \times 1 \times \cdots \times 1}_{n}) = \frac{(n-1)(n-2)}{2}$$

When $q=1$, $\sum_{j=1}^{n} i_j = n+1$,

$$N_1^n = \left\{ \underbrace{(2,1,1,\cdots,1)(1,2,1,\cdots,1)\cdots(1,1,1,\cdots,2)}_{n} \right\}$$

for the sequence $(2,1,1,\cdots,1)$, there are 3 n-dimensional SGQ points $[p_1,0,\cdots,0]^{\mathrm{T}}$, $[-p_1,0,\cdots,0]^{\mathrm{T}}$ and $[0,0,\cdots,0]^{\mathrm{T}}$. The origin has been chosen, so the number of the SGQ points is $X_{q=1} = 2n$. The weights of the points are

$$W = \begin{cases} \underbrace{(-1)^{3-1-1} \times \mathrm{C}_{n-1}^{3-1-1} \times n \times (\omega_1 \times \underbrace{1 \times 1 \times \cdots \times 1}_{n-1}) = -n(n-1)\omega_1}_{[0,0,\cdots,0]^{\mathrm{T}}} \\ \underbrace{(-1)^{3-1-1} \times \mathrm{C}_{n-1}^{3-1-1} \times (\omega_2 \times \underbrace{1 \times 1 \times \cdots \times 1}_{n-1}) = -(n-1)\omega_2}_{\text{the type of } [\pm p_1,0,\cdots,0]^{\mathrm{T}}} \end{cases}$$

When $q=2$, $\sum_{j=1}^{n} i_j = n+2$,

$$N_2^n = \left\{ \begin{matrix} \underbrace{(3,1,1,\cdots,1)(1,3,1,\cdots,1)\cdots(1,1,1,\cdots,3)}_{n} \\ \underbrace{(2,2,1,\cdots,1)(2,1,2,\cdots,1)\cdots(1,1,1,\cdots,2)}_{C_n^2} \end{matrix} \right\}$$

Following the previous analysis, if only one of the accuracy level sequence is 3, the number of points is $4n$; if two accuracy levels are selected for 2 simultaneously, the number is $C_n^2 \times 2 \times 2 = 2n(n-1)$, so $X_{q=2} = 2n^2 + 2n$. The weights of the points are

$$W = \begin{cases} \underbrace{\begin{matrix} (-1)^{3-1-2} \times \mathrm{C}_{n-1}^{3-1-2} \times n \times (\omega_3 \times \underbrace{1 \times 1 \times \cdots \times 1}_{n-1}) \\ +(-1)^{3-1-2} \times \mathrm{C}_{n-1}^{3-1-2} \times \mathrm{C}_n^2 \times (\omega_1 \times \omega_1 \times \underbrace{1 \times \cdots \times 1}_{n-2}) \\ = n\omega_3 + \dfrac{n(n-1)}{2}{\omega_1}^2 \end{matrix}}_{[0,0,\cdots,0]^{\mathrm{T}}} \\ \underbrace{(-1)^{3-1-2} \times \mathrm{C}_{n-1}^{3-1-2} \times \mathrm{C}_{n-1}^{1} \times (\omega_1 \times \omega_2 \times \underbrace{1 \times \cdots \times 1}_{n-2}) = (n-1)\omega_1\omega_2}_{\text{the type of } [\pm p_1,0,\cdots,0]^{\mathrm{T}}} \\ \underbrace{(-1)^{3-1-2} \times \mathrm{C}_{n-1}^{3-1-2} \times (\omega_2 \times \omega_2 \times \underbrace{1 \times \cdots \times 1}_{n-2}) = {\omega_2}^2}_{\text{the type of } [\pm p_1,\pm p_1,0,\cdots,0]^{\mathrm{T}}} \\ \underbrace{(-1)^{3-1-2} \times \mathrm{C}_{n-1}^{3-1-2} \times (\omega_4 \times \underbrace{1 \times 1 \times \cdots \times 1}_{n-1}) = \omega_4}_{\text{the type of } [\pm p_2,0,\cdots,0]^{\mathrm{T}}} \\ \underbrace{(-1)^{3-1-2} \times \mathrm{C}_{n-1}^{3-1-2} \times (\omega_5 \times \underbrace{1 \times 1 \times \cdots \times 1}_{n-1}) = \omega_5}_{\text{the type of } [\pm p_3,0,\cdots,0]^{\mathrm{T}}} \end{cases}$$

As a result, the number of the SGQ points with $L=3$ is

$$X_{n,3} = 2n^2 + 4n + 1, \quad n \ge 2 \tag{22}$$

and the corresponding weights are

$$W_3 = \begin{cases} (1)\ [0,0,\cdots,0]^{\mathrm{T}} \\ \quad \dfrac{(n-1)(n-2)}{2} - n(n-1)\omega_1 + n\omega_3 + \dfrac{n(n-1)}{2}{\omega_1}^2 \\ (2)\ \text{the type of } [\pm p_1,0,\cdots,0]^{\mathrm{T}} \\ \quad -(n-1)\omega_2 + (n-1)\omega_1\omega_2 \\ (3)\ \text{the type of } [\pm p_1,\pm p_1,0,\cdots,0]^{\mathrm{T}} \\ \quad {\omega_2}^2 \\ (4)\ \text{the type of } [\pm p_2,0,\cdots,0]^{\mathrm{T}} \\ \quad \omega_4 \\ (5)\ \text{the type of } [\pm p_3,0,\cdots,0]^{\mathrm{T}} \\ \quad \omega_5 \end{cases} \tag{23}$$

Similarly, the number of the SGQ points with $L=2$ is

$$X_{n,2} = 2n + 1, \quad n \ge 2 \tag{24}$$

The weights W_2 are

$$W_2 = \begin{cases} (1)\ [0,0,\cdots,0]^{\mathrm{T}} \\ \quad -(n-1) + n\omega_1 \\ (2)\ \text{the type of } [\pm p_1,0,\cdots,0]^{\mathrm{T}} \\ \quad \omega_2 \end{cases} \tag{25}$$

For example, the process of 2-dimensional point selection was shown intuitively in Fig. 1.

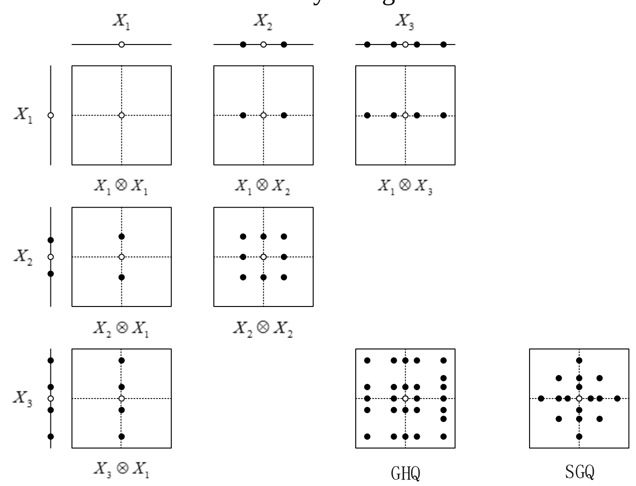

Fig.1 Sparse grid points and traditional points

The quadrature points χ and the weights W obtained by SGQ rule were substituted in GHQF, then the SGQF was derived.

According to the literature [18], the n-variate integral $I_{n,L}$ in the Eq. (17) was exact for n-variate polynomials of the total order up to $2L-1$. Then the SGQF with $L=3$ was superior to the EKF and the UKF, furthermore, its computation was far less than the GHQF.

III. STRONG TRACKING SPARSE GRID QUADRATURE FILTER

The STF was introduced to improve the robustness of the SGQF. The STF showed significant reduction of system uncertainty to the system uncertainty like unpredictable disturbances created by external condition and model uncertainties [23].

A. *Strong Tracking Filter*

The STF can timely adjust the gain matrix with a fading factor to deal with sudden changes. So the predicted covariance matrix $\boldsymbol{P}_{k|k-1}$ should be changed as

$$\boldsymbol{P}_{k|k-1} = \lambda_k \boldsymbol{F}_{k|k-1}\boldsymbol{P}_{k-1}\boldsymbol{F}_{k|k-1}^{\mathrm{T}} + \boldsymbol{Q}_{k-1} \tag{26}$$

where λ_k was the fading factor. Its suboptimal solution was

$$\lambda_k = \begin{cases} \lambda_0 \,, \lambda_0 \geq 1 \\ 1 \quad, \lambda_0 < 1 \end{cases} \tag{27}$$

where

$$\lambda_0 = \frac{\mathrm{tr}[\boldsymbol{N}_k]}{\mathrm{tr}[\boldsymbol{M}_k]} \tag{28}$$

$$\begin{cases} \boldsymbol{N}_k = \boldsymbol{V}_k - \boldsymbol{H}_k \boldsymbol{Q}_{k-1}\boldsymbol{H}_k^{\mathrm{T}} - \boldsymbol{R}_k \\ \boldsymbol{M}_k = \boldsymbol{H}_k \boldsymbol{F}_{k|k-1}\boldsymbol{P}_{k-1}\boldsymbol{F}_{k|k-1}^{\mathrm{T}}\boldsymbol{H}_k^{\mathrm{T}} \end{cases} \tag{29}$$

and

$$\begin{cases} \boldsymbol{F}_{k|k-1} = \dfrac{\partial \boldsymbol{f}_{k-1}(\boldsymbol{x}_{k-1})}{\partial \hat{\boldsymbol{x}}_{k-1}} \\ \boldsymbol{H}_k = \dfrac{\partial \boldsymbol{h}_k(\boldsymbol{x}_k)}{\partial \hat{\boldsymbol{x}}_{k|k-1}} \end{cases} \tag{30}$$

where $\mathrm{tr}(\bullet)$ was the trace of matrix; V_k was the covariance matrix of the residual sequence, it was unknown and can be estimated by

$$\boldsymbol{V}_k = \begin{cases} \boldsymbol{\varepsilon}_1 \boldsymbol{\varepsilon}_1^{\mathrm{T}} & k = 1 \\ \dfrac{\rho \boldsymbol{V}_{k-1} + \boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^{\mathrm{T}}}{1+\rho} & k > 1 \end{cases} \tag{31}$$

where $0 < \rho \leq 1$ was the forgetting factor, usually, $\rho = 0.95$; ε_k was the residual sequence as follows

$$\boldsymbol{\varepsilon}_k = \boldsymbol{z}_k - \hat{\boldsymbol{z}}_{k|k-1} \tag{32}$$

The revised predicted covariance matrix took the place of the original matrix, then we can get the STF.

B. *Strong Tracking Sparse Grid Quadrature Filter*

Before the fading factor was introduced, the predicted covariance matrix was marked as $\boldsymbol{P}_{k|k-1}^{-}$, the innovation covariance matrix was marked as $\boldsymbol{P}_{zz,k|k-1}^{-}$, the cross covariance matrix was marked as $\boldsymbol{P}_{xz,k|k-1}^{-}$, and the predicted measurement was marked as $\hat{\boldsymbol{z}}_{k|k-1}^{-}$ .Process noise and measurement noise were independent, so

$$\begin{aligned} \boldsymbol{P}_{zz,k|k-1}^{-} &= \mathrm{E}[(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_{k|k-1})(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_{k|k-1})^{\mathrm{T}}] \\ &= \mathrm{E}\{[\boldsymbol{H}_k(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k-1}) + \boldsymbol{w}_k - \boldsymbol{v}_k]\cdot[\boldsymbol{H}_k(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k-1}) + \boldsymbol{w}_k - \boldsymbol{v}_k]^{\mathrm{T}}\} \\ &= \boldsymbol{H}_k \mathrm{E}[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k-1})(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k-1})^{\mathrm{T}}]\boldsymbol{H}_k^{\mathrm{T}} + \mathrm{E}[(\boldsymbol{w}_k - \boldsymbol{v}_k)(\boldsymbol{w}_k - \boldsymbol{v}_k)^{\mathrm{T}}] \\ &= \boldsymbol{H}_k \boldsymbol{P}_{k|k-1}^{-}\boldsymbol{H}_k^{\mathrm{T}} + \boldsymbol{R}_k \end{aligned} \tag{33}$$

and

$$\boldsymbol{P}_{xz,k|k-1}^{-} = \boldsymbol{P}_{k|k-1}^{-}\boldsymbol{H}_k^{\mathrm{T}} \tag{34}$$

then

$$\boldsymbol{H}_k = [\boldsymbol{P}_{xz,k|k-1}^{-}]^{\mathrm{T}}(\boldsymbol{P}_{k|k-1}^{-})^{-1} \tag{35}$$

The Eq. (33), (34) and (35) were put into the Eq. (29), the equivalent expressions of $\boldsymbol{N}_k$ and $\boldsymbol{M}_k$ were given by

$$\begin{cases} \boldsymbol{N}_k = \boldsymbol{V}_k - \boldsymbol{R}_k - [\boldsymbol{P}_{xz,k|k-1}^{-}]^{\mathrm{T}}(\boldsymbol{P}_{k|k-1}^{-})^{-1}\cdot\boldsymbol{Q}_{k-1}(\boldsymbol{P}_{k|k-1}^{-})^{-1}\boldsymbol{P}_{xz,k|k-1}^{-} \\ \boldsymbol{M}_k = \boldsymbol{P}_{zz,k|k-1}^{-} - \boldsymbol{V}_k + \boldsymbol{N}_k \end{cases} \tag{36}$$

and the fading factor λ_k can be calculated by Eq. (27)~(36).

The process of the STSGQF based on the SGQF was given as follows.

1) $\boldsymbol{P}_{k|k-1}^{-}$, $\boldsymbol{P}_{zz,k|k-1}^{-}$, $\boldsymbol{P}_{xz,k|k-1}^{-}$ and $\hat{\boldsymbol{z}}_{k|k-1}^{-}$ can be calculated by Eq. (5)~(13);

2) The fading factor λ_k is calculated by Eq. (27)~(36);

3) The fading factor λ_k is introduced to the original predicted covariance matrix $\boldsymbol{P}_{k|k-1}^{-}$, and it can be changed as

$$\boldsymbol{P}_{k|k-1} = \lambda_k \sum_{i=1}^{M} \omega_i (\boldsymbol{f}(\xi_{i,k-1}) - \hat{x}_{k|k-1})(\boldsymbol{f}(\xi_{i,k-1}) - \hat{x}_{k|k-1})^{\mathrm{T}} + \boldsymbol{Q}_{k-1} \tag{37}$$

4) According to the Eq. (9)~(13), the new innovation covariance matrix $\boldsymbol{P}_{zz,k|k-1}$, the new cross covariance matrix $\boldsymbol{P}_{xz,k|k-1}$ and the new predicted measurement $\hat{\boldsymbol{z}}_{k|k-1}$ with fading factor are obtained;

5) Update the filter with $\boldsymbol{P}_{zz,k|k-1}$, $\boldsymbol{P}_{xz,k|k-1}$ and $\hat{\boldsymbol{z}}_{k|k-1}$ by the Eq. (14)~(16).

IV. RESULTS AND DISCUSSION

For both weak and strong maneuver, we compared the performances of different filters under the two conditions.

A. *Weak maneuver*

In Cartesian coordinate, we assumed that the underwater target is in motion with constant velocity of $(10,\ 5,\ 0)\,m/s$ at the initial position $(200,\ 100,\ -100)\,m$ in the directions of x, y and z; the period T is 0.01s, and tracking the target 300 steps; 100 Monte-Carlo simulations were carried out.

When choosing CA model, the state vector is

$$\boldsymbol{x}_k = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} & \ddot{x} & \ddot{y} & \ddot{z} \end{bmatrix}^{\mathrm{T}}$$

State transition matrix is

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F_1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{F_1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{F_1} \end{bmatrix}$$

$$\boldsymbol{F}_1=\begin{bmatrix}1 & T & T^2/2\\ 0 & 1 & T\\ 0 & 0 & 1\end{bmatrix}$$

Initial state covariance matrix is

$$\boldsymbol{P}=\mathrm{diag}([x_0^{\ 2}\quad y_0^{\ 2}\quad z_0^{\ 2}\quad \dot{x}_0^{\ 2}\quad \dot{y}_0^{\ 2}\quad \dot{z}_0^{\ 2}\quad \ddot{x}_0^{\ 2}\quad \ddot{y}_0^{\ 2}\quad \ddot{z}_0^{\ 2}])$$

Process noise covariance matrix is

$$\boldsymbol{Q}=\begin{bmatrix}\boldsymbol{Q_1} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \boldsymbol{Q_1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \boldsymbol{Q_1}\end{bmatrix}$$

$$\boldsymbol{Q}_1=\begin{bmatrix}T^5/20 & T^4/8 & T^3/6\\ T^4/8 & T^3/3 & T^2/2\\ T^3/6 & T^2/2 & T\end{bmatrix}$$

Measurement noise covariance matrix is

$$\boldsymbol{R}=\mathrm{diag}([\rho\quad \theta\quad \theta])$$

Where ρ was the error of radial distance, and θ was the pitch and azimuth's errors.

Five filters were compared, GHQFs with $L=2$ and $L=3$, SGQFs with $L=2$ and $L=3$, and UKF with $\kappa=0$ were marked as GHQF2, GHQF3, SGQF2, SGQF3 and UKF(0), respectively. The Root Mean Square Error (RMSE) of position marked as RMSE-s and the RMSE of velocity marked as RMSE-v in Fig. 2 and Fig. 3.

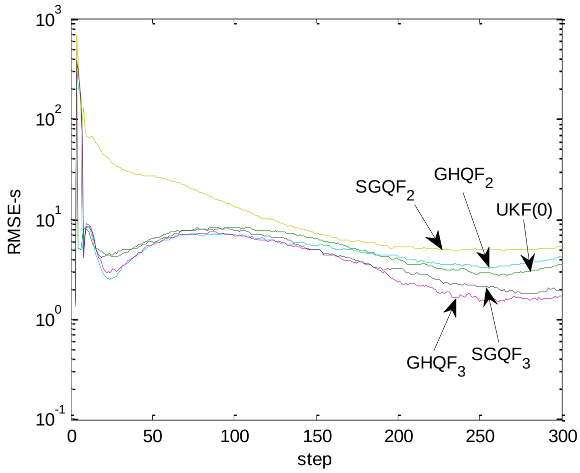

Fig. 2 Position Root Mean Square Error

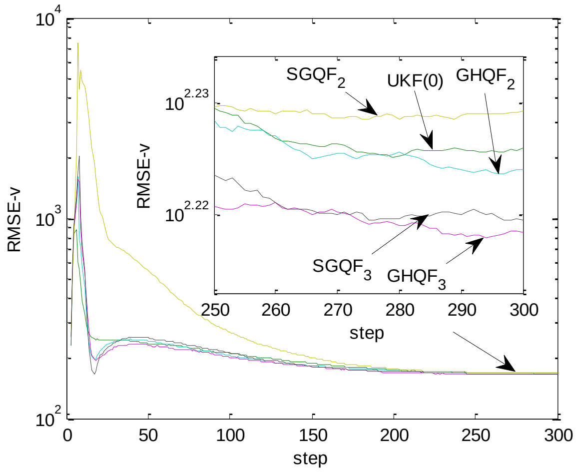

Fig. 3 Velocity Root Mean Square Error

According to the result in Fig. 2, it is clear that the GHQF3 performed the best and the tracking accuracy of the SGQF2 was the worst. SGQFs performed worse than GHQFs with the same accuracy level. However, the SGQF3 was superior to the GHQF2 and the UKF (0). In Fig. 3, the relationship of five filters' RMSE-v was similar to RMSE-s.

The performance of algorithm was evaluated not only by considering its result but also its efficiency. The performance of filters was presented in Table Ⅱ.

TABLE Ⅱ
FILTERS' RUNNING TIME

Filter	Time/s
$GHQF_2$	1371.311
$GHQF_3$	28295.863
$SGQF_2$	38.515
$SGQF_3$	197.390
UKF(0)	25.498

Result in Table Ⅱ showed that the GHQFs run much longer than the SGQFs with the same accuracy level, especially the GHQF3 has run nearly 8 hours. Although the GHQF3 performs the best, it is the worst in terms of efficiency. The SGQF2 shortens about 97% running time contrasted with the GHQF2, only need more than half minute, yet its effect is not expected. The SGQF3 runs for more than 3 minutes, and shortens about 99% time contrasted with the GHQF3, furthermore, its tracking accuracy is slightly lower than the GHQF3 and better than the other 3 filters. The UKF (0) spends less than half minute on processing, but its accuracy is not high enough. Considering comprehensively, the SGQF3 is the best among the 5 filters.

B. *Strong maneuver*

In Cartesian coordinate, we assumed that the underwater target is in motion with velocity of (10, -5, -1)m/s at the initial position (200, 100, -100)m in the directions of x, y and z, respectively; tracking the target 500 steps. Other parameters and initial variables were presented in section A. Changes of acceleration of the target were shown in Table III.

TABLE III
CHANGE OF ACCELERATIONS

step *i*	x-acceleration /m/s2	y-acceleration / m/s2	z-acceleration / m/s2
0-100	0	0	0
100-200	-10	20	-5
200-300	0	0	0
300-400	-10	-20	15
400-500	0	0	0

In this test, the performances of the STF, the STUKF, the STSGQF3 and the SGQF3 were compared. Trajectory of the target was presented in Fig. 4. The RMSE-s and the RMSE-v are given by Fig. 5 and Fig. 6.

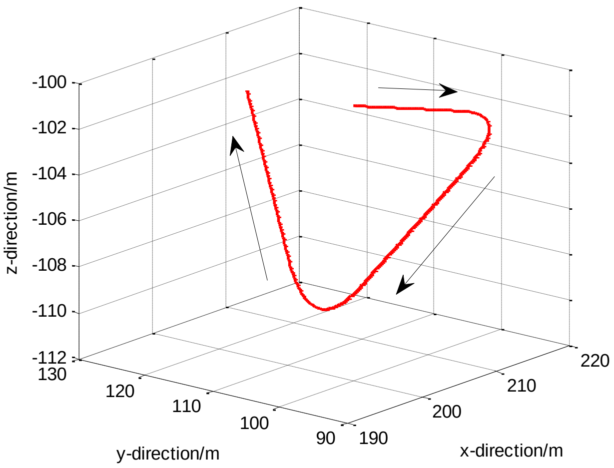

Fig.4 Target motion trajectory

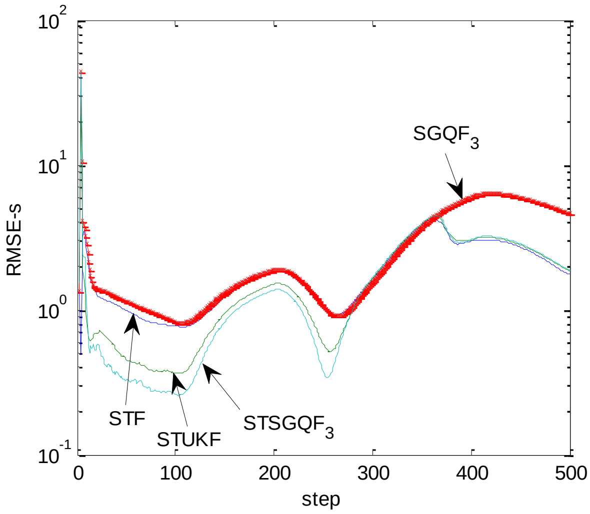

Fig. 5 Position Root Mean Square Error

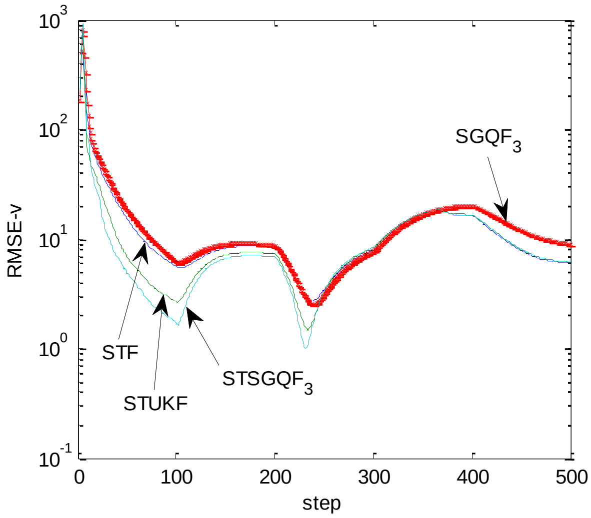

Fig. 6 Velocity Root Mean Square Error

In order to verify the effectiveness which reducing the computation load of STSGQF, the Monte Carlo method was used. We sampled 100,000 random numbers between 0 and 1. These values were combined with STF, STUKF, STSGQF and SGQF, respectively. The average run time of STF, STUKF, $STSGQF_3$ and $SGQF_3$ were 272.356s, 380.377s, 583.798s and 710.254s respectively as seen in Table Ⅳ.

It can be seen from Fig. 4 that the underwater target makes sharp turns. The results of tracking were shown in Fig. 5 and Fig. 6, the accuracy of strong tracking algorithms was higher than the SGQF3, especially when the state changes, and they converged more quickly than the SGQF3. Among the strong tracking algorithms, the STSGQF3 was the best overall. Considering the filters' running time, the computational cost of the STF was the least, the STSGQF3 cost about 10 minutes. Compared with the SGQF3, the running time of the STSGQF3 increased, but the STSGQF3 was improved against the system uncertainties with increasing the running time.

TABLE Ⅳ
FILTERS' RUNNING TIME

Filter	Time/s
STF	272.356
STUKF	380.377
$STSGQF_3$	583.798
$SGQF_3$	710.254

V. CONCLUSIONS

In this paper, the Strong Tracking Sparse Grid Quadrature Filter (STSGQF) is proposed. The STF is introduced to timely adjust the gain matrix, and the SGQ rule is used to select quadrature points flexibly and moderately. Simulation examples of weak and strong maneuver are provided for evaluating the performances of SGQF and STSGQF, respectively. We conclude that the SGQF can reduce the computational cost significantly compared with GHQF. After the STF is integrated into SGQF, the robustness of STSGQF is significantly improved despite the increase of running time.

Finally, in the strong maneuver case, we prove that the STSGQF can take tracking quickly and efficiently.

REFERENCES

[1] G. Isbitiren and O. B. Akan, "Three-Dimensional Underwater Target Tracking With Acoustic Sensor Networks," *IEEE Trans. on Vehicular Technology*, vol. 60, no. 8, pp. 3897–3906, 2011.

[2] S. K. Rao, *Application Of Statistical Estimators For Underwater Target Tracking*. CRC Press Inc., 2006, pp. 1040–1044.

[3] D. P. Eickstedt, M. R. Benjamin, H. Schmidt and J . J. Leonard, "Adaptive Tracking of Underwater Targets with Autonomous Sensor Networks," *Journal of Underwater Acoustics*, 2006, pp. 465–495.

[4] E. Dalberg, A. Lauberts, R. K. Lennartsson, M. J. Levonen and L. Persson, "Underwater target tracking by means of acoustic and electromagnetic data fusion," presented at the 2006 International Conference on Information Fusion, Florence, pp. 1-7.

[5] C. H. Yu and J. W. Choi, "Interacting multiple model filter-based distributed target tracking algorithm in underwater wireless sensor networks", *International Journal of Control, Automation and Systems*, vol. 12, no. 3, pp. 618-627, 2014.

[6] A. Sebastian, G. Gerardo, L. Andre and A. Rozenfel, "Evaluation of an efficient approach for target tracking from acoustic imagery for the perception system of an autonomous underwater vehicle," *International Journal Of Advanced Robotic Systems*, vol. 11, no. 1, pp. 1, 2014.

[7] T. Li, D. Zhao, Z. Huang, C. Liu, and S. Su, "A wavelet-based grey particle filter for self-estimating the trajectory of manoeuvring autonomous underwater vehicle," *Transactions of the Institute of Measurement & Control*, vol. 36, no. 3, pp. 321-335, 2014.

[8] T. D. Zhang, L. Wan, B. Wang and W. J. Zeng, "Underwater object tracking based on improved particle filter," *Journal of Shanghai Jiaotong University*, vol. 6, pp. 934-948, 2012.

[9] R H Zhan and J Zhang, *Nonlinear Filtering Theory with Target Tracking Application*, Beijing: National Defense Industry Press, 2013.

[10] L. U. Chuan-Guo, X . X. Feng and Z. Di, "Pure bearing tracking based on improved cubature Kalman filter," *Systems Engineering & Electronics*, vol. 34, no. 1, pp. 28-33, 2012.

[11] F. Liccardo, S. Strano and M. Terzo, "Real-time nonlinear optimal control of a hydraulic actuator", *Engineering Letters*, vol. 21, no.4, pp. 241-246, 2013.

[12] A. Nattapol, "A bayesian filtering approach with time-frequency representation for corrupted dual tone multi frequency identification", *Engineering Letters*, vol. 24, no.4, pp. 370-377, 2016.

[13] A. J. Hang, *Bayesian Estimation and Tracking A practical Guide*, X. Wang, X. Yu, Translation. Beijing: National Defense Industry Press, 2014.

[14] L. Zhao, *Nonlinear System Filtering Theory*, Beijing: National Defense Industry Press, 2012.

[15] A. Alcocer, P. Oliveira and A. Pascoal, "Study and implementation of an EKF GIB-based underwater positioning system," *Control Engineering Practice*, vol. 15, no. 6, pp. 689-701, 2007.

[16] X. X. Wang, Q. Pan, H. Huang and A. Gao, "Overview of deterministic sampling filtering algorithms for nonlinear system," *Control & Decision*, vol. 27, no. 6, pp. 801-812, 2012.

[17] Q. Dong, Q. Sun and F. Yu, "A new combination rule in evidence theory," *Engineering Letters,* vol. 24, no. 3, pp. 284-289, 2016.

[18] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910-927, 2000.

[19] B. Jia, M. Xin and Y. Cheng, "Sparse-grid quadrature nonlinear filtering," *Automatica*, vol. 48, no. 2, pp. 327-341, 2012.

[20] C. Y. Ran, X. H. Cheng and H. P. Wang, "Application of sparse-grid Gauss filter in SINS initial alignment," *Journal of Chinese Inertial Technology*, vol. 21 no. 5, pp. 591-597, 2013.

[21] W. U. Zong-Wei, M. L. Yao, M. A. Hong-Guang, W. M. Jia and F. H. Tian, "Sparse-Grid Square-Root Quadrature Nonlinear Filter," *Acta Electronica Sinica*, vol. 40, no. 7, pp. 1298-1303, 2012.

[22] X. U. Song, X. X. Sun, S. G. Liu, L. Xi and C. Ming, "Dimension-wise Adaptive Spare Grid Quadrature Nonlinear Filter," *Acta Automatica Sinica*, vol. 40, no. 6, pp. 1249-1264, 2014.

[23] F. Srairi, L. Saidi, F. Djeffal and M. Meguellati, "Modeling, control and optimization of a new swimming microrobot design," *Engineering Letters*, vol. 24, no. 1, pp. 106-112, 2016.

[24] D. H. Zhou, Y. G. Xi and Z. J. Zhang, "A suboptimal multiple fading extended Kalman filter," *Acta Automatica Sinica*, vol. 17, no. 6, pp. 689-695, 1991.

[25] B. Ye and Y. Xu, "Comparison Between a Strong Tracking Filter and Kalman Filter for Target Tracking," *Journal of Air Force Radar Academy*, vol. 22, no. 2, pp. 17-19, 2002.

[26] X. X. Wang, L. Zhao and H. X. Xue, "Strong tracking CDKF and application for integrated navigation," *Control & Decision*, vol. 25, no. 12, pp. 1837-1842, 2010.

[27] L. Yang, B. Li, X. Li and L. Ge, "A Novel Particle Filter Algorithm Based on Strong Tracking UKF," *Journal Of Investigative Medicine*, vol. 62, no. 8, pp. 28-29, 2014.

[28] Z. Li, P. Pan, D. Gao and D. Zhao, "An Improved Unscented Kalman Filter Based on STF for Nonlinear Systems". in *Proc. 2nd International Congress on Image And Signal Processing*, Tianjin, China, 2009, pp. 4070-4074.