


ITC 1/51 Information Technology and Control Vol. 51 / No. 1 / 2022 pp. 86-103 DOI 10.5755/j01.itc.51.1.29776	An Adaptive Fuzzy Neural Network Based On Progressive Gaussian Approximate Filter with Variable Step Size	
	Received 2021/09/09	Accepted after revision 2022/12/03
	 http://dx.doi.org/10.5755/j01.itc.51.1.29776	

HOW TO CITE: Zhu, G. (2022). An Adaptive Fuzzy Neural Network Based On Progressive Gaussian Approximate Filter with Variable Step Size. *Information Technology and Control*, 51(1), 86-103. <https://doi.org/10.5755/j01.itc.51.1.29776>

An Adaptive Fuzzy Neural Network Based On Progressive Gaussian Approximate Filter with Variable Step Size

Guorui Zhu

School of Mechanical and Electrical Engineering, Southwest Petroleum University, Chengdu, 610500, China;
e-mail: zhgr1995@163.com

Corresponding author: zhgr1995@163.com

The nonlinear filtering problem is a hot spot in robot navigation research. As an advanced method that can effectively improve the robustness and accuracy of the system, the progressive Gaussian approximate filter with variable step size (PGAFVS) still has some shortcomings, how to resolve the nonlinear filtering problem in the application of tightly coupled integration under the premise of the prior uncertainty and further promote robustness high measurement accuracy, which becomes the purpose of this paper. This paper formulates the processing of trajectory tracking measurement noise problem as a Kalman filtering procedure and the measurement noise covariance matrix in controller, is jointly estimated based on the progressive Gaussian approximate filter (PGAF), after that, PGAFVS can be deduced. Then we proposed an adaptive fuzzy and backpropagation neural network controller based on PGAFVS (AFNPGA-VS) that can improve the application of tightly coupled integration under the premise of the prior uncertainty and further promote robustness high measurement accuracy. The simulation results show that the proposed algorithm outperforms the state-of-the-art methods.

KEYWORDS: Nonlinear filter, progressive measurement update, neural network.

1. Introduction

The Kalman filter (KF) uses the attenuation function model as a matrix, and the power of the matrix represents recursion. Finally, a class of logarithmic difference equations can be derived, and the optimal solution of the difference equation can be predicted by covariance. The KF can fuse information in an uncertain environment, at the same time, it also keeps the ability of high information accuracy. In the field of filtering algorithm optimization of navigation systems, the popular research direction is based on the inertial navigation algorithm to expand algorithm function. Crassidis [5] proposed a sigma-point filter formulation that was shown for the GPS (global positioning system)/INS (inertial navigation systems) applications, it can be used in the sigma-point filter using quaternion kinematics without a unit quaternion estimate. The filter improved the performance of the sigma-point filter by reducing the size of the covariance matrix. According to the difference of physical quantities involved, they are divided into loose combinations and compact combinations. However, in urban tunnels, surrounding buildings and environment sealing can cause satellite signals to lockout, thus unable to locate GPS, nor can obtain GPS speed and position, so that complex environments such as a dense wall or wild mountain environments in another city cause multiple path reception of GPS, which will make GPS positioning accuracy worse [21]. Another method based on SINS (strapdown inertial navigation system)/GPS pseudo distance and pseudo distance rate compact combination has good practical value and higher precision because satellites work less than four ones. Compared with loose combination tightly combination model is more complex and requires higher real-time requirement and synchronization requirements for real systems. Xu et al. [33] proposed the ARIMA (auto-regressive-integrated-moving-average) model compared with other positioning methods, methodology cost lower and sensors have higher accuracy, and the accurate localization methodology using MEMS-INS (micro-electro-mechanical system based inertial navigation system)/GPS/in-vehicle sensors integration can be used in uncertain noise of MEMS-INS and can improve the accuracy of predict INS errors in the case of the GPS outages, but the filter not applicable in the outdoor environment,

especially in the intersection environment. However, the GPS positioning system is affected by occlusion interference multipath reflection and so on which leads to lower positioning accuracy or positioning failure thus unable to guarantee automatic driving function requirement [27].

It is well known that the Kalman filter is widely used in linear systems state-space model. However nonlinear models are playing a decisive role in navigation tracking and positioning. Extended Kalman filter (EKF) is a nonlinear linearization filtering method based on KF. Cubature Kalman filter (CKF) has been widely used in GPS navigation processing. Although EKF exists dependent on first order linearization of system models to propagate state mean and covariance, linear processes and system accuracy decrease thus resulting in performance degradation and failure of running results [10]. However, researchers have made a lot of efforts to expand the scope of application of this technology [8-13]. Especially adaptive algorithm combined with EKF has been used to improve the accuracy of the system model and solve divergence problems caused by a single use of EKF. Another method to enhance the robustness of the filter is to improve noise robustness by using the variational Bayes method. Variational Bayes method is proposed firstly by experts in the machine learning domain and used for offline estimation. Now it is widely recognized as a method of estimating noise parameters and states simultaneously. Sarkka and Nummenmaa [22] proposed an adaptive Kalman filter based on adaptive Kalman filter which is based on the combination posterior distribution of state and noise parameters separately into a separable variational approximation. Using the recursive algorithm, Kalman filter is used to estimate state at each step, and sufficient statistic of noise variance is estimated by fixed-point iteration. Wang et al. [32] proposed a theory that introduces VB into multisensor information fusion and proposes augmented and sequential adaptive fusion algorithms. All noise distributions in the above researches are conducted according to Gaussian distribution, so the noise processing effect is improved less obviously. To solve this problem, Nurminen et al. [18] proposed a theory about nonlinear filter and smoother for linear discrete-time state-space model based

skew-t-distributed, they removed some posterior independence approximations from the earlier versions of the algorithms to avoid significant underestimation of the posterior covariance matrix to reduce the error caused by the variational approximation. And they used the expectation propagation (EP) algorithm removal of independence approximations of the mean and covariance matrix. Moreover, the application of skewed distribution is not limited to positioning based on radio signals. In biological, partial distribution is used as a modeling tool for Multivariate skew cell classification data [26]. In meteorology, skewed distributions may also be applied to remote sensing studies on tropical cyclone patterns relating to topographic impacts [25], skewness distribution used for modeling asymmetric data [16].

The fuzzy algorithm with EKF into robot positioning system which is suitable for local adaptive neural fuzzy extended Kalman filtering algorithm. Adjusting estimation EKF algorithm process noise covariance matrix and measuring noise covariance matrix etc. elements are measured at each sampling time [1]. The aim is to reduce the mismatch between theoretical values and actual covariance between new theories and values. Sasiadek et al. [23] proposed the fuzzy systems with adaptive Kalman filter. The adaptive filter process navigation data fusion design for fuzzy logic adaptive systems (FLAS) to measure noise covariance matrix. Akhlaghi et al. [2] proposed an adaptive filtering approach to adaptively estimate Q and R based on innovation and residual EKF, the proposed method is more robust against the initial errors in Q and R. Zakari et al. [34] proposed a novel interval type-2 fuzzy fractional-order super twisting algorithm to deal with classes of fully-actuated and under-actuated nonlinear systems in presence of uncertainty. Petkovic et al. [19] proposes an adaptive neuro-fuzzy inference (ANFIS) algorithm by an artificial neural network (ANN) and the fuzzy inference system (FIS) improving precipitation estimation accuracy.

To optimize the parameter estimation and empirical robustness analysis, researchers proposed and improved the adaptive filters. Nourmohammadi and Keighobadi [17] proposed the adaptive integration scheme is proposed, which is robust to the performance of the attitude-heading reference system (AHRS) and can be used for the AHRS and the SINS according to vehicle maneuvering conditions. In or-

der to solve this shortcoming, a novel fuzzy adaptive extended Kalman filter exploiting the Student's t distribution (FASTEKF) for a robot path tracking is proposed [12]. In FASTEKF, the distributions of process and measurement noise are processed by Student's t distribution. With the adaptive fuzzy controller, the adaptive factors are designed to adjust the covariance matrices of the process and measurement noises simultaneously. Li et al. [14] proposed an improved modified gain extended Kalman filter (MGEKF) based on backpropagation neural network (BPNN), termed BPNN-MGEKF algorithm, the algorithm solved the problem that the error measurement value is adopted by the MGEKF algorithm in practical application, which leads to the error in the result. Zhang et al. [35] proposed the accelerated BP (GAD-BP) neural network model. In this model, the data are loaded and the genetic algorithm is used to train the network parameters to obtain the global optimal solution. And the LM (Levenberg-Marquardt) algorithm is used for further numerical optimization, they used the AdaGrad method to dynamically adjust the learning rate at the same time.

In this paper, we propose an adaptive fuzzy and neural network controller based on PGAF with variable step size (AFNPGAF-VS) for resolving and researching the nonlinear filtering problem with prior uncertainty but high measurement accuracy, and the proposed method can overcome the limitation that the PGAF has poor estimation accuracy. And the method can also effectively improve the processing effect of the nonlinear filter. We summarize the contributions of this task as follows:

- 1 The proposed algorithm can be applied to the research field of an inspection robot for the first time.
- 2 This paper formulates the processing of trajectory tracking measurement noise problem as a Kalman filtering procedure, which can recursively optimize the noise processing.
- 3) To perform Kalman filtering for treating noise, therefore, this paper builds a BPNN and an adaptive fuzzy controller. The BPNN aims at obtaining and optimizing the posterior error covariance matrix based on the prior results. The adaptive fuzzy controller aims at deal with the measurement noise preferably.

The remainder of this paper is arranged as follows. Section 2 introduces some basic theories including the BPNN, the fuzzy adaptive system and the PGAF.

In Section 3, this paper introduces the AFNPGAF-VS algorithm and its running process. Simulation comparisons and analyses of the CKF, PGAF-VS, AF-PGAF-VS (an adaptive fuzzy controller based on PGAF-VS), and AFNPGAF-VS algorithms are shown in Section 4; Conclusions are given in Section 5.

2. Basic Theories

In this section, in order to better illustrate the theories used by the proposed algorithm, we introduce some basic theories including the backpropagation neural network technology, the fuzzy adaptive system and the progressive Gaussian approximate filter.

2.1. Fuzzy Logic Adaptive System

Fuzzy theory was first born in the 1960s. In the beginning, concepts such as fuzzy set and membership function were used to solve problems such as mathematical modeling of information asymmetry. Later, it has been widely applied in the range of artificial intelligence, fuzzy decision making, and adaptive control theory. In many cases, if you are using the fuzzy mathematics and fuzzy logic concepts, general can solve the related issues of fuzzy control, using fuzzy logic can be solved in artificial intelligence or related issues in the field of uncertainty information, etc., so the application of fuzzy theory category is very wide [20, 15, 4, 31].

The steps of fuzzy modeling can be completed by using fuzzy rule description, which is related to the establishment of fuzzy set, the determination of fuzzy logic, and the application of fuzzy reasoning. Fuzzy rules are divided into two categories. One is the Mam-

dani fuzzy rule, which emphasizes the nonlinear description of local linear model interpolation and is commonly used in the field of fuzzy control of nonlinear systems. The other is to divide the fuzzy space into several fuzzy subspaces, which can not only be used in fuzzy control but also widely used in the Takagi-Sugeno (T-S) fuzzy rule in mathematical modeling. A typical fuzzy control system must have three links: fuzzy, fuzzy reasoning and fuzzy, fuzzy is to obtain the numerical, and the scope of the process to become a blur after fuzzy reasoning, a fuzzy value reasoning based on a knowledge base and modeling, the fuzzy value after the final output of a process, the blur is will get processed reverse reasoning, fuzzy values for a specific value clearly.

To simplify the complexity of system design, especially for nonlinear, time-varying, incomplete model systems, and in the process of establishing function model this paper needs to use control laws to describe the relationship between system variables, so we choose to use T-S fuzzy controller.

If a typical T-S fuzzy model is described by the mathematical formulas, it can be written as follows:

Model rule i : IF Input x_1 is F_1^i and Input x_2 is F_2^i and Input x_n is F_n^i

THEN Output:

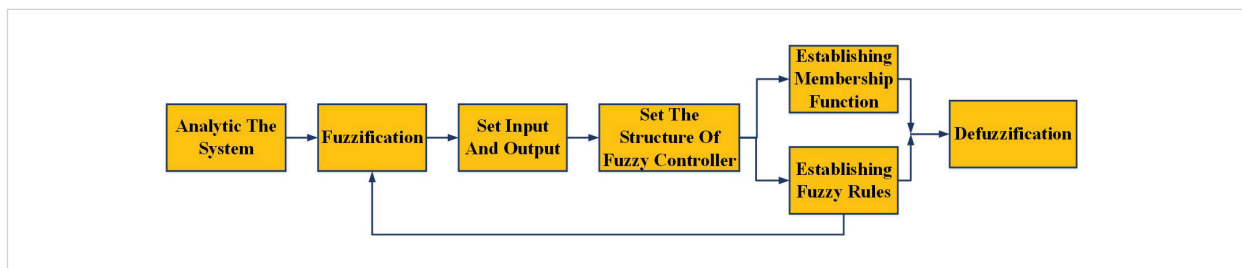
$$y_k = f_k(x_1, x_2, \dots, x_n) = C_{k0} + C_{k1}x_1 + \dots + C_{kn}x_n \quad (1)$$

where the membership function of the input variable vector can be expressed as F , $C_{ki}(i = 0 \dots n)$ are constants in the k -th rule. the output level in the zero-order model is a constant:

For the first-order model, the fuzzy rule can be expressed in the form:

Figure 1

A fuzzy system architecture



Model Rule i: IF Input x_1 is F_1^1 and Input x_2 is F_2^1 ;
THEN Output:

$$y_k = C_{10} + C_{11}x_1 + C_{12}x_2 \tag{2}$$

so this paper can use the weighted average method of defuzzification to get the output value. The weighted average defuzzification method can be written as follows:

$$y = \sum_{k=1}^M w_k y_k, \tag{3}$$

where w_k is the weights, it can be written as follows,

$$w_k = \frac{\prod_{i=1}^n \mu_{F_i^k}^k(x_i)}{\sum_{j=1}^M \left[\prod_{i=1}^n \mu_{F_i^j}^j(x_i) \right]}, \tag{4}$$

with $\sum_{i=1}^M w_i = 1$ and the μ represent the membership function.

2.2. Back Propagation Neural Network

BP neural network is generally based on a large number of measured data, using multiple simulation and learning, all the predicted values and the output values are constantly error analysis, so as to obtain the output of the closest to the ideal state after a large number of data learning. This often involves the theoretical knowledge of variable weights, mapping relations, mathematical expression, etc., these processes or processing links are based on the 'black box model', which is similar to fuzzy processing within a certain range of numerical inference. Therefore, according to the error positive feedback correction of reverse transmission, the error is continuously controlled within the adaptive threshold, and satisfactory results can be obtained. Under artificial conditions, the neural network cannot rely on a thin database to complete, so the limitation is that a large number of data must be mastered and collected to complete the operating conditions of BP neural network modeling [24, 6-7].

The complete BP neural network consists of the following three parts: input layer, output layer and hidden layer. There are many neurons in the input layer and output layer, which are responsible for data pro-

cessing. The specific number of neurons depends on the data that the algorithm needs to calculate. The training content and samples used in this algorithm will be mentioned in the following simulation. The purpose of the hidden layer is to process the data once (many times), the purpose is to strengthen the training effect, but multi-layer processing will make time too long, thus affecting the efficient operation of the mechanism. In order to achieve the overall optimal training effect, the selection of the number of hidden layers is also important. Only when the optimal number is reached can the optimal training samples be obtained efficiently and quickly. The hidden layer selected in this paper is finally selected as one layer so that the data processing speed is the fastest and the data training result is the most $y_k = C_{10} + C_{11}x_1 + C_{12}x_2$ ideal.

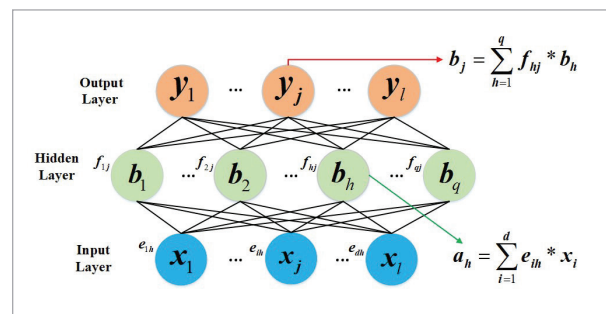
A basic BP neural network model will include two links. One is the reverse transfer, that is, to correct the error between the expected value and the predicted value, so as to obtain positive feedback. The other is forward transfer, which complies with each link to complete the operation to get the expected value.

BP network consists of input layer, hidden layer and output layer. Given the training set $D = \{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}$, where $x_n \in R^d, y_n \in R^1$, the input example is composed of d attributes, output 1-dimensional real-valued variables. Now, this paper looks at how to obtain the output value, and how to adjust the weights and thresholds from the output value.

The signal flow diagram of the BP algorithm is shown in the following Figure 2.

In this section, this paper sets the e_{ih} as the weight of the input layer to the hidden layer and sets the f_{jh} as

Figure 2
BPNN



the threshold of the h neurons in the hidden layer. The weights from the hidden layer to the output layer are set as f_{hj} , and the threshold of the j neurons in the output layer is represented by θ_j . In the following Figure 2, there are d input neurons, q hidden neurons, q hidden neuron thresholds, and one output neuron, so there is one output neuron threshold, respectively.

The activation function of the hidden layer and the output layer is Sigmoid function, it can be assumed that the training output of the neural network is $y_k = (y_1^{k'}, y_2^{k'}, \dots, y_l^{k'})$ and output to one-dimensional vector, where $y_i^{k'} = f(b_i - \theta_i)$, and then the least square method of prediction error is summarized as follows,

Input layer to hidden layer:

$$A_h = \sum_{i=1}^d e_{ih} x_i \quad (5)$$

The activation function of hidden layer:

$$B_h = f(a_h - f_h) \quad (6)$$

Hiding layer to output layer:

$$B_j = \sum_{h=1}^q f_{hj} b_h \quad (7)$$

Activation function of output layer:

$$Y_j^k = f(b_j - \theta_j) \quad (8)$$

2.3. The Gaussian Approximate Filter (GAF)

The process equation and the measurement equation are respectively expressed as [29],

$$x_k = f(x_{k-1}, u_k, w_k) \quad (9)$$

$$z_k = h(x_k, v_k), \quad (10)$$

where x_k and z_k are the state vector and the measurement vector, respectively, the k -th time step is k , the process noise is w_k , the measurement noise vector is v_k , $f(\cdot)$ and $h(\cdot)$ are the nonlinear process and measurement function, respectively. The control input vector is u_k , it can be written as $u_k = [\hat{v}_k, \hat{\omega}_k, \hat{\theta}_k]^T$. The control input vector measured is composed of a

strap-down SINS or a compass. The x_0 is initial state, it is a Gaussian random vector with mean $\hat{x}_{0|0}$ and covariance matrix $P_{0|0}$, and it is independent from w_k and v_k .

Nonlinear filtering is designed to estimate unknown system state x_k and noise measurement vector $z_{1:k}$, where $z_{1:k} = z_{jj=1}^k$ means the noise measurements from time index 1 to k . It is important to obtain the posterior PDF of system state x_k , so it can be achieved by two steps: time update and measurement update. The one-step predictive PDF $p(x_k|z_{1:k-1})$ of system state during the time update stage can be written as,

$$p(x_k|z_{1:k}) = \int p(x_k|x_{k-1})p(x_k|z_{1:k-1})dx_{k-1} \quad (11)$$

where $p(x_k|z_{1:k-1})$ is the posterior PDF at last time index $k-1$, and $p(x_k|x_{k-1})$ is the state transition PDF.

The posterior PDF $p(x_k|z_{1:k-1})$ of system state during the measurement update stage can be written as,

$$P(x_k|z_{1:k-1}) = \frac{p(x_k, z_k|z_{1:k-1})}{p(z_k|z_{1:k-1})}, \quad (12)$$

where the joint PDF of the x_k and the z_k is formulated as follow,

$$p(x_k, z_k|z_{1:k-1}) = p(z_k|x_k)p(x_k|z_{1:k-1}) \quad (13)$$

where $p(z_k|x_k)$ is the measurement likelihood PDF, and its function is formulated as follows,

$$p(z_k|z_{1:k}) = \int p(x_k, z_k|z_{k-1})dx_k = \int p(z_k|x_k)p(x_k|z_{1:k-1})dx_k \quad (14)$$

Based on the assumption that the joint PDF of x_k and z_k is approximated as Gaussian, (11)-(14) can be formulated as follows,

$$\begin{aligned} & p(x_k, z_k|z_{k-1}) \\ & \approx \hat{p}(x_k, z_k|z_{k-1}), \\ & = N \left(\begin{bmatrix} x_k \\ z_k \end{bmatrix}; \begin{bmatrix} \hat{x}_{k|k-1} \\ \hat{z}_{k|k-1} \end{bmatrix}, \begin{bmatrix} P_{k|k-1} & P_{k|k-1}^{xz} \\ (P_{k|k-1}^{xz})^T & P_{k|k-1}^{zz} \end{bmatrix} \right) \end{aligned} \quad (15)$$

where $\hat{x}_{k|k-1}$ and $P_{k|k-1}$ are the first two moments with respect to $p(x_k|z_{1:k-1})$. $\hat{z}_{k|k-1}$ and $P_{k|k-1}^{zz}$ are the first two moments with respect to $p(z_k|z_{1:k-1})$, and $P_{k|k-1}^{xz}$ is the cross covariance matrix between x_k and z_k .

The posterior PDF $p(x_k | z_{1:k})$ can be approximated as Gaussian distribution.

Under the framework of PGAF, the posterior PDF can be written as follows,

$$p(x_k | z_{1:k}) \propto p(x_k | z_{1:k-1}) \prod_{\gamma=1}^N [P(z_k | x_k^\gamma)]^{\lambda_\gamma}, \quad (16)$$

where $z_{1:k} = \{z_j\}_{j=1}^k$, $\sum_{\gamma=1}^n \lambda_\gamma = 1, \lambda_\gamma \in (0,1)$, is the progressive step size. x_k^λ is the state vector at the $\tilde{\epsilon}$ -th recursion, and N is the number of product units in the progressive process.

The intermediate one-step predicted PDF and likelihood PDF can be written as Gaussian,

$$P(x_k^\gamma | z_{1:k}^{\gamma-1}) = N(x_k^\gamma; \hat{x}_{k|k}^{\gamma-1}, P_{k|k}^{\gamma-1}) \quad (17)$$

$$P(z_k | x_k^\gamma) = N(z_k; h_k(x_k^\gamma), R_k^\gamma), \quad (18)$$

where $N(\cdot; \mu, \Sigma)$ is a Gaussian PDF with mean μ and covariance matrix Σ . $\hat{x}_{k|k}^\gamma$ is the estimate of x_k^γ , and $z_{1:k}^\gamma$ is the gradually absorbed measurements sequence. $P_{k|k}^\gamma$ is the estimated error covariance matrix at the γ -th recursion. And the j -th diagonal element of R_k^γ can be written as

$$R_k^\gamma = \text{diag}[(\sigma_{k,1}^\gamma)^2, \dots, (\sigma_{k,j}^\gamma)^2]. \quad (19)$$

The state x_k^γ together with the step size λ_γ and the inaccurate measurement noise covariance matrix R_k^γ are, respectively, assumed to satisfy Gaussian, uniform and Inverse Wishart (IW) priors:

$$p(x_k^\gamma, \lambda_\gamma, R_k^\gamma | z_{1:k}^{\gamma-1}) \approx N(x_k^\gamma; \hat{x}_{k|k}^{\gamma-1}, P_{k|k}^{\gamma-1}) U(\lambda_\gamma; 0, \eta_\gamma) \cdot \text{IW}(R_k^\gamma; u_{k|k-1}^\gamma, U_{k|k-1}^\gamma). \quad (20)$$

However, in many cases, the progressive step size is selected according to engineering experience, which is usually replaced by constant $\lambda_\gamma = \frac{1}{N}$. This leads to the limitation of selecting the optimal step size under the existing PGAF framework, which hinders the further development of relevant theories. In addition, combined with the multiple approximation calculation of Gaussian weighted integration, it is bound

to produce a large integral approximation error. The approximation error accumulated by multi-step iteration is used as an additional pseudo-measurement noise for modeling and subsequent calculation. This process is often accompanied by the generation of the integral approximation error, medium time-varying actual environment and other non-modeling errors that will eventually lead to the decrease of the accuracy of MNMCM. In the process of searching the data, it is found that the measured data of the inspection robot with a tightly coupled integrated navigation system show that the error sources of the measurement noise of pseudo range and pseudo range rate are the adverse effects of multipath reflection and heavy weather. The VS-PGAF can solve these problems effectively.

2.4. The PGAF with Variable Step Size

The approximate posterior PDF can be written as [3],

$$p(x_k^\gamma, \lambda_\gamma, R_k^\gamma | z_{1:k}) \approx q(x_k^\gamma) q(\lambda_\gamma) q(R_k^\gamma). \quad (21)$$

The optimal solution satisfies can be written as,

$$\log q(\theta) = E_{(\Lambda, \theta)} [\log p(\Lambda, z_{1:k})] + c_\theta. \quad (22)$$

a Measurement update

Before achieving the ultimate threshold ϵ , the step size λ, γ , which will be estimated in the measurement update.

The PDF $p(\Lambda, z_{1:k})$ can be written as follows,

$$p(\Lambda, z_{1:k}) = c(\gamma) N\left(z_k; h_k(x_k^\gamma), \frac{R_k^\gamma}{\lambda_\gamma}\right) N(x_k^\gamma; \hat{x}_{k|k}^{\gamma-1}, \hat{P}_{k|k}^{\gamma-1}) \cdot U(\lambda_\gamma; 0, \eta_\gamma) \text{IW}(R_k^\gamma; u_{k|k-1}^{\gamma-1}, U_{k|k-1}^{\gamma-1}) \quad (23)$$

where $c(\gamma)$ is the normalization constant. And

$$\begin{aligned} \log p(\Lambda, z_{1:k}) = & 0.5 m \log \lambda_\gamma - 0.5 (x_k^\gamma - \hat{x}_{k|k}^{\gamma-1})^T (P_{k|k}^{\gamma-1})^{-1} (x_k^\gamma - \hat{x}_{k|k}^{\gamma-1}) \\ & - 0.5 (u_{k|k-1}^\gamma + m + 2) \log |R_k^\gamma|, \quad (24) \\ & - 0.5 \lambda_\gamma (z_k - h_k(x_k^\gamma))^T (R_k^\gamma)^{-1} (z_k - h_k(x_k^\gamma)) \\ & - 0.5 \text{tr}(U_{k|k-1}^\gamma (R_k^\gamma)^{-1}) + c_\Lambda \end{aligned}$$

where c_Λ is the constant with respect to Λ . Let $\theta = \lambda_\gamma$,

and utilizing (24) in (22), this paper obtains

$$\begin{aligned} \log q^{i+1}(\lambda_\gamma) &= 0.5m \log \lambda_\gamma \\ &\quad - 0.5\lambda_\gamma \text{tr}(\mathbf{D}_k^{\gamma(i)} \mathbf{E}^i (\mathbf{R}_k^\gamma)^{-1}) + \mathbf{c}_\lambda, \end{aligned} \quad (25)$$

where the approximate PDF of $q(\cdot)$ at the $i+1$ -th iteration can be written as $q_{i+1}(\cdot)$, and $\mathbf{D}_k^{\gamma(i)}$ can be written as follows,

$$\mathbf{D}_k^{\gamma(i)} = \mathbf{E}^i [(z_k - \mathbf{h}_k(x_k^\gamma))(\mathbf{h}_k(x_k^\gamma))^T] \quad (26)$$

Thus $q^{i+1}(\lambda_\gamma)$ can be updated as

$$q^{i+1}(\lambda_\gamma) = G(\lambda_\gamma; \alpha^{i+1}, \beta^{i+1}), \quad (27)$$

where $G(\cdot; \alpha, \beta)$ is the Gamma PDF based on shape parameter α and rate parameter β . So α^{i+1} and β^{i+1} can be given by

$$\alpha^{i+1} = 0.5m + 1, \beta^{i+1} = 0.5 \text{tr}(\mathbf{D}_k^{\gamma(i)} \mathbf{E}^i (\mathbf{R}_k^\gamma)^{-1}). \quad (28)$$

Let $\theta = \mathbf{R}_k^\gamma$, and utilizing (24) in (22), this paper obtains

$$\begin{aligned} \log q^{i+1}(\mathbf{r}_k^\gamma) &= 0.5(u_{k|k-1}^\gamma + m + 2) \log |\mathbf{r}_k^\gamma| \\ &\quad - 0.5 \text{tr} \left[\lambda_\gamma \text{tr} \left(\mathbf{D}_k^{\gamma(i)} \mathbf{E}^{i+1} \lambda_\gamma + \mathbf{U}_{k|k-1}^\gamma \right) (\mathbf{r}_k^\gamma)^{-1} \right] + \mathbf{c}_r. \end{aligned} \quad (29)$$

Thus $q^{i+1}(\mathbf{r}_k^\gamma)$ can be updated as

$$Q^{i+1}(\mathbf{r}_k^\gamma) = \text{IG}(\mathbf{r}_k^\gamma; \mathbf{u}_k^{\gamma(i+1)}, \mathbf{U}_k^{\gamma(i+1)}), \quad (30)$$

where the dof parameter $\mathbf{u}_k^{\gamma(i+1)}$ and inverse scale matrix $\mathbf{U}_k^{\gamma(i+1)}$ are given by

$$\mathbf{u}_k^{\gamma(i+1)} = \mathbf{u}_{k|k-1}^\gamma + 0.5 \quad (31)$$

$$\mathbf{U}_k^{\gamma(i+1)} = \mathbf{U}_{k|k-1}^\gamma + 0.5 \mathbf{E}^{i+1} \lambda_\gamma \mathbf{D}_k^{\gamma(i)}. \quad (32)$$

Let $\theta = \mathbf{x}_k^\gamma$, and utilizing (22) in (24), this paper obtains,

$$\begin{aligned} \log q^{i+1}(\mathbf{x}_k^\gamma) &= \\ &\quad - 0.5 (\mathbf{x}_k^\gamma - \hat{\mathbf{x}}_{k|k}^{\gamma-1})^T (\mathbf{P}_{k|k}^{\gamma-1})^{-1} (\mathbf{x}_k^\gamma - \hat{\mathbf{x}}_{k|k}^{\gamma-1}) \\ &\quad - 0.5 (z_k - \mathbf{h}_k(\mathbf{x}_k^\gamma))^T (\mathbf{R}_k^\gamma)^{-1} \cdot (z_k - \mathbf{h}_k(\mathbf{x}_k^\gamma)) + \mathbf{c}_x, \end{aligned} \quad (33)$$

where

$$\tilde{\mathbf{R}}_k^\gamma = \frac{\mathbf{E}^{i+1} [\mathbf{R}_k^\gamma]}{\mathbf{E}^{i+1} [\lambda_\gamma]}. \quad (34)$$

And then the $\mathbf{E}^{i+1} [\mathbf{R}_k^\gamma]$ can be written as [10],

$$\mathbf{E}^{i+1} [\mathbf{R}_k^\gamma] = \text{diag} \left(\frac{\mathbf{U}_k^{\gamma(i+1)}}{\mathbf{u}_k^{\gamma(i+1)}} \right), \quad (35)$$

which can be regarded as the posterior PDF of the state in the standard GA filter, and using (33)-(35), the PDF $q^{i+1}(\mathbf{x}_k^\gamma)$ can be processed as Gaussian, it can also be updated by the modified likelihood PDF $\mathbf{N}[z_k; \mathbf{h}_k(\mathbf{x}_k^\gamma), \mathbf{R}_k^\gamma]$ at same time.

$$Q^{i+1}(\mathbf{x}_\gamma) = \mathbf{N}(\mathbf{x}_k^\gamma; \hat{\mathbf{x}}_{k|k}^{i+1}, \mathbf{P}_{k|k}^{i+1}), \quad (36)$$

where $\hat{\mathbf{x}}_{k|k}^{\gamma(i+1)}$ is the state estimation and $\mathbf{P}_{k|k}^{\gamma(i+1)}$ is estimated error covariance matrix at the γ -th recursion and the $i+1$ -th fixed-point iteration. The preset threshold is ε , the progressive loop will be stopped when the rest of the progressive increment is less than it. After N -th recursions,

$$\begin{aligned} q^{i+1}(\mathbf{x}_\gamma) &= \mathbf{N}(\mathbf{x}_k^\gamma; \hat{\mathbf{x}}_{k|k}^{i+1}, \mathbf{P}_{k|k}^{i+1}) \\ &= \frac{\mathbf{N}(z_k; \mathbf{h}_k(\mathbf{x}_k^\gamma), \hat{\mathbf{R}}_k^\gamma) \mathbf{N}(\mathbf{x}_k^\gamma; \hat{\mathbf{x}}_{k|k}^{\gamma-1}, \mathbf{P}_{k|k}^{\gamma-1})}{\int \mathbf{N}(z_k; \mathbf{h}_k(\mathbf{x}_k^\gamma), \hat{\mathbf{R}}_k^\gamma) \mathbf{N}(\mathbf{x}_k^\gamma; \hat{\mathbf{x}}_{k|k}^{\gamma-1}, \mathbf{P}_{k|k}^{\gamma-1}) d\mathbf{x}_k^\gamma} \end{aligned} \quad (37)$$

where $\hat{\mathbf{x}}_{k|k}^{\gamma(i+1)}$ and $\mathbf{P}_{k|k}^{\gamma(i+1)}$ denote the first two moments with respect $q^{i+1}(\mathbf{x}_\gamma)$ and can be written as,

$$\mathbf{K}_k^{\gamma(i+1)} = \mathbf{P}_{k|k}^\gamma (\mathbf{P}_{k|k}^{\gamma(i+1)})^{-1} \quad (38)$$

$$\mathbf{P}_{k|k}^{\text{zzy}(i+1)} = \frac{\mathbf{z}_k (\mathbf{z}_k)^T}{2n} - \hat{\mathbf{z}}_{k|k}^\gamma (\hat{\mathbf{z}}_{k|k}^\gamma)^T \quad (39)$$

$$\mathbf{P}_{k|k}^{\text{xzy}(i+1)} = \frac{\mathbf{x}_k (\mathbf{z}_k)^T}{2n} - \hat{\mathbf{x}}_{k|k}^\gamma (\hat{\mathbf{z}}_{k|k}^\gamma)^T \quad (40)$$

$$\hat{\mathbf{x}}_{k|k}^{\gamma(i+1)} = \hat{\mathbf{x}}_{k|k}^\gamma \mathbf{K}_k^{\gamma(i+1)} (z_k - \hat{\mathbf{z}}_{k|k}^\gamma) \quad (41)$$

$$\mathbf{P}_{k|k}^{\gamma(i+1)} = \mathbf{P}_{k|k}^\gamma - \mathbf{K}_k^{\gamma+1} \mathbf{P}_{k|k}^{\text{zzy}(i+1)} (\mathbf{K}_k^{\gamma+1})^T, \quad (42)$$

where $\mathbf{K}_k^{\gamma(i+1)}$ is the progressive filtering gain.

b Time update

In the existing PGAF framework, the predicted state estimation $\hat{x}_{k|k-1}$ and predicted error covariance matrix $P_{k|k-1}$ during the time update stage can be formulated as follows,

$$\begin{aligned}\hat{x}_{k|k-1} &= \int x_k p(x_k | z_{1:k-1}) dx_k \\ &\approx \int f_{k-1}(x_{k-1}) \\ &\quad N(x_{k-1}; \hat{x}_{k-1|k-1}, P_{k-1|k-1}) dx_{k-1}\end{aligned}\quad (43)$$

$$\begin{aligned}P_{k|k-1} &= \int x_k x_k^T p(x_k | z_{1:k-1}) dx_k - \hat{x}_{k-1|k-1} (\hat{x}_{k-1|k-1})^T \\ &\approx \int f_{k-1}(x_{k-1}) f_{k-1}^T(x_{k-1}) N(x_{k-1}; \hat{x}_{k-1|k-1}, P_{k-1|k-1}) dx_{k-1} \\ &\quad + Q_{k-1} - \hat{x}_{k-1|k-1} (\hat{x}_{k-1|k-1})^T\end{aligned}\quad (44)$$

The predicted PDF $p(R_k | z_{1:k-1})$ can be formulated as

$$p(R_k | z_{1:k-1}) = \int p(R_k | R_{k-1}) p(R_{k-1} | z_{1:k-1}) dR_{k-1} \quad (45)$$

where $p(R_{k-1} | z_{1:k-1})$ is the posterior PDF of MNMCM at $k-1$ time, and $p(R_k | R_{k-1})$ is the transition PDF. In practical applications, the transition model $p(R_k | R_{k-1})$ is hard to determine, thus in this paper, considering that the MNMCM is mildly time-varying in many practical applications, a discount factor $\rho \in (0, 1)$ is utilized to illustrate the time-fluctuation level. The prior parameters based on the previous posterior PDF and the discount factor ρ can be formulated as

$$\alpha_{k|k-1,j} = \rho \alpha_{k-1,j} \quad (46)$$

$$\beta_{k|k-1,j} = \rho \beta_{k-1,j} \quad (47)$$

The proposed VS-PGAF is summarized as Table 1, where the preset maximum number of VB recursions can be expressed as N_{VB} , the ultimate number of fixed-point iterations is M .

3. The Model Decomposition and Algorithm

In order to further optimize the measurement noise and improve the calculation accuracy of P matrix, this part will introduce how to use the adaptive fuzzy

controller and the BPNN technologies based on the PGAF algorithm framework.

3.1. Optimization of Measurement Noise

This single point can be seen as a fuzzy set that has been clarified in advance. Because this algorithm can improve the efficiency of clarity and greatly simplify the amount of computation in the usual Mamdani system. The selection of fuzzy control rules and membership function of fuzzy variables is the key to the design of fuzzy controller. Considering that its essence is parameter optimization, genetic algorithm is introduced into fuzzy control. Genetic algorithm is a global search algorithm, which is very suitable for the optimal design of fuzzy controller.

The conclusion of fuzzy reasoning mainly depends on the relationship between input and output and the synthesis algorithm between fuzzy sets. The fuzzy adaptive system of robot measurement noise adopts the fuzzy controller structure of double input and a single output. The two inputs of the system have 7 fuzzy subsets, so 49 fuzzy rules can be formed, and the inference rules can be expressed as,

Table 1

Fuzzy rule table

I-b \ I-a	NL	NS	Z	PS	PL
NL	7	6	6	5	4
NS	6	5	5	4	3
Z	6	5	4	3	2
PS	5	4	3	3	2
PL	4	3	2	2	1

In general, when the covariance becomes larger and deviates from the zero mean, the filter will tend to be unstable. In order to avoid filter divergence and improve robustness, the fuzzy logic system is used to select the appropriate adaptive factor (ζ), and the posterior error covariance matrix is optimized by adjusting the proportion of measurement noise.

3.2. Optimization of Covariance Matrix

Firstly, the data are imported into the input layer, and then the training set and test set randomly generated

by the toolbox are allowed. To ensure the reliability and operation efficiency of the training set, a total of 10 samples need to be collected in the training set, and 7 groups of samples need to be collected in the test set. Secondly, the divided training set and test set are normalized. Finally, the training parameters are set, BPNN is run, and the simulated data are denormalized and then output.

Summarize the steps as follows,

Set $I_i = P_{k|k}^{\gamma(i+1)}$, input layer to hidden layer:

$$A_h = \sum_{i=1}^d v_{ih} I_i. \tag{48}$$

The activation function of hidden layer:

$$B_h = f(A_h - f_h). \tag{49}$$

Hiding layer to output layer:

$$B_j = \sum_{h=1}^q f_{hj} B_h. \tag{50}$$

Activation function of output layer:

$$O_j^k = f(B_j - \theta_j). \tag{51}$$

The error:

$$E_k = \frac{1}{2} \sum_{j=1}^l (O_j^k - O_j^k)^2 \tag{52}$$

this paper set $P_{k|k}^{\gamma(i+1)} = O_j$, ϵ and when $E_k < \epsilon$, the calculation results are not credible, and the neural network is recalculated.

3.3. The Algorithm Derivation

Robustness and accuracy of the algorithm is shown below. In order to reduce the influence of noise on the AF-NPGAF-VS, this paper proposed that the measurement noise covariance matrix by the adaptive fuzzy controller and estimated error covariance matrix can be optimized by BPNN algorithm, it can be expressed as,

$$P_{k|k}^{\gamma(i+1)} = \frac{z_k (z_k)^T}{2n} - \hat{z}_{k|k}^{\gamma} (\hat{z}_{k|k}^{\gamma})^T + R_f \tag{53}$$

$$K_k = P_{k|k}^{\gamma} (P_{k|k}^{\gamma(i+1)})^{-1} \tag{54}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k}^{\gamma} + K_k (z_k - \hat{z}_{k|k}^{\gamma}) \tag{55}$$

$$P_{k|k} = P_{k|k}^{\gamma} - K_k P_{k|k}^{\gamma(i+1)} K_k^T, \tag{56}$$

where $R_f = \zeta \cdot R$, and $P_{k|k}$ was processed by (35)-(39).

The condition for executing a fixed point iteration loop is

$$\text{Norm}(\Xi^{(i+1)}, \Xi^{(i)}) > \vartheta, \tag{57}$$

where the threshold value ϑ is set as 10^{-6} , and the $\text{Norm}(\cdot, \cdot)$ represents the 2-norm distance.

In addition, the corresponding prior outputs is the initial values of progressive measurement update, it can be expressed as follows,

$$\hat{x}_{k|k}^0 = \hat{x}_{k|k-1} \tag{58}$$

$$\hat{P}_{k|k}^0 = \hat{P}_{k|k-1}, \tag{59}$$

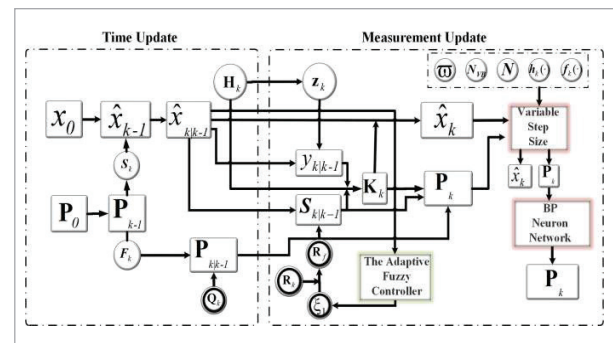
meanwhile this paper has

$$\alpha_{k|k,j}^0 = \alpha_{k|k-1,j} + 0.5 \tag{60}$$

$$\beta_{k|k,j}^0 = \beta_{k|k-1,j}. \tag{61}$$

The proposed robust filter is summarized in Algorithm 1. And the Figure 3 provides the structure dia-

Figure 3
The Algorithm Flow Chart



gram of the proposed algorithm. The chart consists of two parts: the "Time Update" part represents the time update stage of algorithm; the "Measurement Update" part represents the measurement update stage of algorithm, it including the adaptive fuzzy controller system, the PGAF with variable step size and the BPNN controller.

Algorithm 1 An Adaptive Fuzzy Neural Network Based On Progressive Gaussian Approximat FilterWith Variable Step Size.

Input: $P_{k-1|k-1}$, $\hat{x}_{k-1|k-1}$, R_k , z_k , N , N_{VB} , $f_{k-1}(\cdot)$, $h_{k-1}(\cdot)$, \mathcal{O} .

Output: $\hat{x}_{k|k}$, $P_{k|k}$.

Compute $\hat{x}_{k|k-1}$ and $P_{k|k-1}$ by the time update of the standard GAF.

Compute the $u_{k|k-1}^{\gamma}$ and $U_{k|k-1}^{\gamma}$ as (38),

Measurement update:

Initialize the progressive loop: $\hat{x}_{k|k} \rightarrow \hat{x}_{k|k-1}^0, P_{k|k} \rightarrow P_{k|k-1}^0, R_0 \rightarrow R_k^0$

For $\gamma = 1 : N$

Initialize the fix-point iterations loop: $\hat{x}_{k|k}^{\gamma-1} \rightarrow \hat{x}_{k|k-1}^{\gamma-1(0)}, P_{k|k}^{\gamma-1} \rightarrow P_{k|k-1}^{\gamma-1(0)}, R_0^{\gamma-1} \rightarrow E^0[R_k^{\gamma}]$

While $\text{Norm}(\Lambda(i), \Lambda(i-1)) > \delta$ and $i \leq N_{VB}$

Compute $D^{\gamma(i)}$ by (21) with $\hat{x}_{k|k}^{\gamma-1(i-1)}$ and $P_{k|k}^{\gamma-1(i-1)}, \alpha_{i+1}$ and β_{i+1} by (23)

Compute $E^{i+1}\lambda^{\gamma}$ by the definition of the integral, and compute $u_k^{\gamma(i+1)}, U_k^{\gamma(i+1)}$ by (26), (27)

Compute $E^{i+1}[R_k^{\gamma}]$ and \tilde{R}_k^{γ} by (28) and (29)

Compute $\hat{z}_{k|k}^{\gamma}, P_{k|k}^{zz\gamma(i+1)}$ and $P_{k|k}^{zz\gamma}$ by Gaussian weight integration rule

Compute by(44)-(47): $\tilde{P}_{k|k}^{zz\gamma(i+1)} = \frac{z_k(z_k)^T}{2n} - \frac{z_k^{\gamma}(z_k^{\gamma})^T}{2n} +$

$R_f, R_f = \zeta \cdot R, \tilde{K}_k^{\gamma(i+1)} = P_{k|k}^{\gamma} (\tilde{P}_{k|k}^{zz\gamma(i+1)})^{-1},$

$\hat{x}_{k|k}^{\gamma(i+1)} = \hat{x}_{k|k}^{\gamma} \tilde{K}_k^{\gamma(i-1)} (z_k - z_k^{\gamma}, \tilde{P}_{k|k}^{\gamma(i+1)}) = P_{k|k}^{\gamma} -$

$\tilde{K}_k^{\gamma+1} \tilde{P}_{k|k}^{zz\gamma(i+1)} (\tilde{K}_k^{\gamma+1})^T$. End While

$\hat{x}_{k|k}^{\gamma(i+1)} \rightarrow \hat{x}_{k|k}^{\gamma}, \tilde{P}_{k|k}^{\gamma(i+1)} \rightarrow P_{k|k}^{\gamma}, E^{M+1}[R_k^{\gamma}] \rightarrow R_k^{\gamma}$.

When the step size $< \epsilon$, the progressive loop stop.

End For

Update information $\hat{x}_{k|k}^{\gamma} \rightarrow \hat{x}_{k|k}, P_{k|k}^{\gamma} \rightarrow P_{k|k}$

4. Simulation Results

In this simulation, in order to compare with the existing similar filtering algorithms [3], we chose the existing PGAF-VS, the PGAF-VS with the adaptive fuzzy controller (AFPGAF-VS) and the proposed PGAF that with both the adaptive fuzzy controller and the

BPNN are taken into comparison. All the filters are configured with identical parameters.

In addition, this paper considers that the standard CKF is one of the most typical GAF. Thus, this paper also put CKF into the comparison. The existing AFP-GAF-VS are taken into comparison to verify the superiority of the AFNPGAF-VS. The existing PGAF-VS is set as $N = 20$. For AFP-GAF-VS and AFNPGAF-VS, this paper set the discount factor $m=1-e^{-4}$, and $\alpha_{0j}=0$, $\beta_{0j}=1$ with $j=1 \dots m$, respectively.

To compare the performances of the proposed filters with existing filters, the root-mean-square errors (RMSEs) of the position, velocity, and attitude are chosen as the performance metrics. The RMSE in position is defined as [9],

$$\text{RMSE}_{\text{pos}} = \sqrt{\frac{1}{M} \sum_{s=1}^M ((\lambda_k^s - \hat{\lambda}_k^s)^2 + (\rho_k^s - \hat{\rho}_k^s)^2)}, \quad (62)$$

where the true and estimated positions at the s -th Monte Carlo run can be written as (λ_k^s, ρ_k^s) and $(\hat{\lambda}_k^s, \hat{\rho}_k^s)$. The RMSE value can be used as an important index to describe the estimation accuracy of algorithm. Similar to the RMSE in position, this paper can also formulate the RMSE in velocity and attitude.

In this case, it is seen from Figures 4-6 and Table 2 that the proposed filter has smaller RMSEs of the state variable than the other filters over most time throughout the simulation process. The proposed filter has considerably improved filtering accuracy as compared to the CKF, PGAF-VS and the AFP-GAF-VS.

Figure 4
RMSEs of the target position

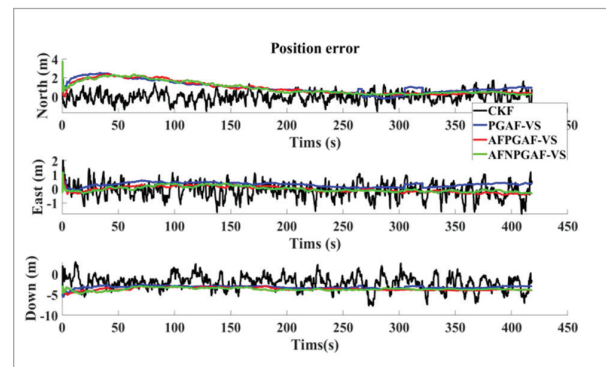


Figure 5
RMSEs of the target velocity

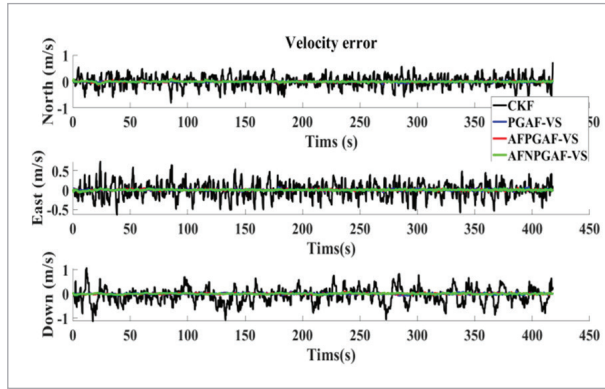
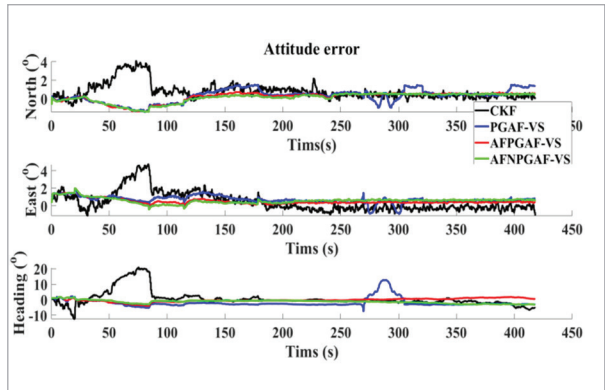


Figure 6
RMSEs of the target attitude



This section will list the average value of WCR using the CKF, PGAF-VS, AFPGAF-VS, the proposed filter, the results are as follows in Figures 4-6. As can be seen, the comparison results of three different RMSEs (include position, velocity, attitude) between the three algorithms indicated that the proposed filter showed a relatively ideal level, with obvious stability and accuracy than other filters, so the proposed filter performed slightly better than PGAF-VS and CKF in these three indices.

By using the normalized estimation error square NEES to estimate the variance index, the robust of the filtering algorithm can be quantified. In addition, the evaluation method can also ulteriorly reflect the integrity of the algorithm [34]. The estimated value \hat{x}_k^+ and the error covariance matrix P_k^+ at k-th time

step can be calculated, then the normalized estimation error squared (NEES) can be written as [28],

$$\begin{aligned} \varepsilon_k &= (x_k - x_k^+)^T \cdot (P_k^+)^{-1} \cdot (x_k - x_k^+) \\ &= x_k^{+T} \cdot (P_k^+)^{-1} \cdot x_k^+ = \|x_k^+\|_{(P_k^+)^{-1}}^2 \end{aligned} \quad (63)$$

Table 2
Average value of RMSEs

Filters	CKF	PGAF-VS	AFPGAF-VS	AFNPGAF-VS
North Position	-0.997	1.199	0.930	0.925
East Position	-0.123	0.299	-0.047	-0.029
Down Position	-3.061	-3.180	-3.046	-2.680
North Velocity	-0.005	-0.009	-0.006	-0.006
East Velocity	-0.008	0	-0.001	-0.001
Down Velocity	-0.081	-0.004	-0.002	-0.001
North Attitude	0.014	0.002	-0.003	0.002
East Attitude	0.009	0.013	0.010	0.007
Down Attitude	0.016	-0.034	-0.015	-0.012

Table 3
NEES values of the filters

Filters	CKF	PGAF-VS	AFP-GAF-VS	AFNPGAF-VS
	6.467	6.869	2.935	2.333

Table 3 shows the NEES values of the proposed filter and existing filter. From the results, this paper can conclude that the NEES value of the proposed filter is the smallest, so the reliability and integrity of the proposed filter are better than the existing filters. This result further explains that the proposed filter has better estimation credibility, this paper also needs to see that the computational complexity of the proposed filter is slightly higher than that of the existing filters.

The stability of filters operation effect can be expressed by the averaged absolute value of biases (AAVB), the AAVB is defined as follows:

$$AAVB = \frac{1}{T} \sum_{k=1}^T \left| \frac{1}{M} \sum_{s=1}^M ((\lambda_k^s - \hat{\lambda}_k^s)^2) \right|. \quad (64)$$

This paper can see from the Figures 7-9 that the proposed filter has smaller AAVBs of position, velocity and attitude but slightly higher computational complexity than CKF and PGAF-VS. Moreover, the proposed filter has smaller bias and better estimation accuracy than the AFPGAF-VS.

Figure 7

AAVBs of the target position

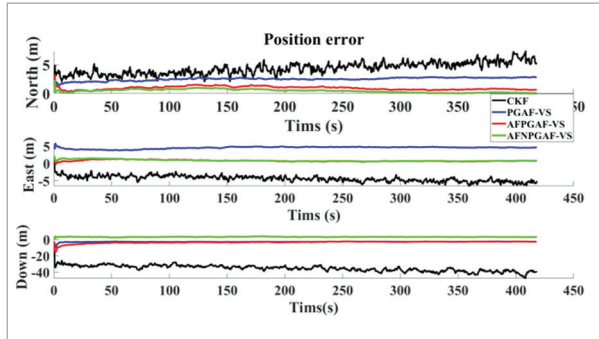


Figure 8

AAVBs of the target velocity

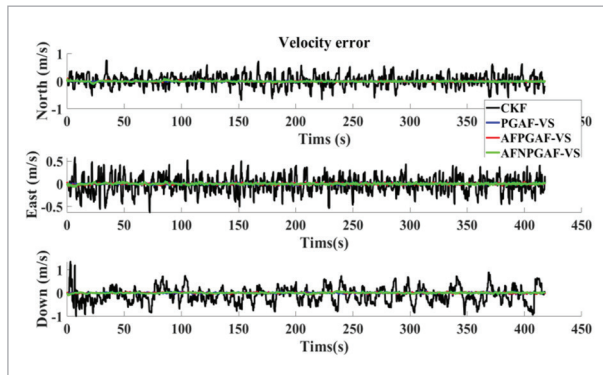
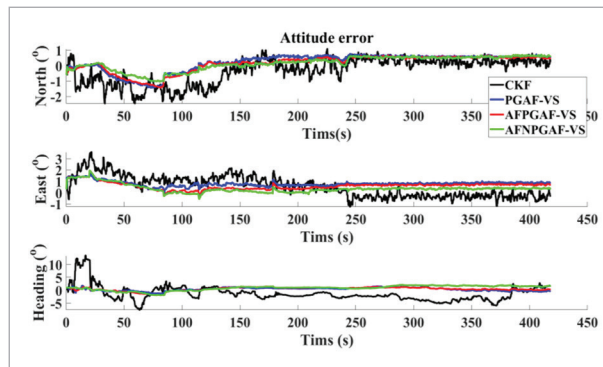


Figure 9

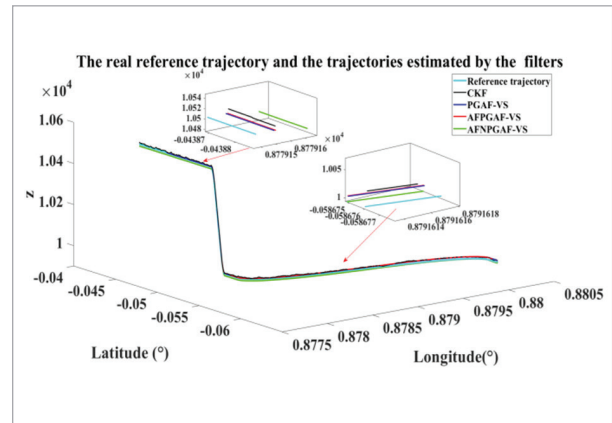
AAVBs of the target attitude



The real reference trajectory and the trajectories estimated by different filters are compared in Figure 10. It is seen from Figure 10 that the estimated trajectories from the proposed filter are closer to the true trajectory as compared with existing filters most time, particularly after longitude and latitude ($0.05^\circ, 0.87^\circ$), the accuracy of the PGAF-VS and AFPGAF-VS are better than the standard CKF, but worse than the proposed filter, which is caused by the process and measurement outliers.

Figure 10

The real trajectory and the trajectories estimated



5. Field Performance Test

To further test the performance of the proposed filter in robot navigation system, this section will use the navigation filter results of the robot in the actual motion to validate the simulation part.

According to the demand, we finally use the complex environment surrounding a college teaching building as an experimental field and chose to build a suitable inspection robot, the hardware circuit modules of the robot control system in this experiment are as follows [30],

- 1 Wireless communication module
- 2 Rocker module
- 3 Motor drive module
- 4 Main controller circuit
- 5 External connection module
- 6 Power module

Figure 11
The inspection robot

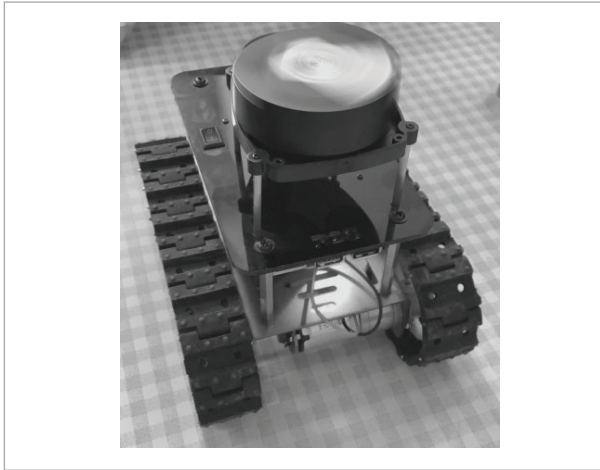
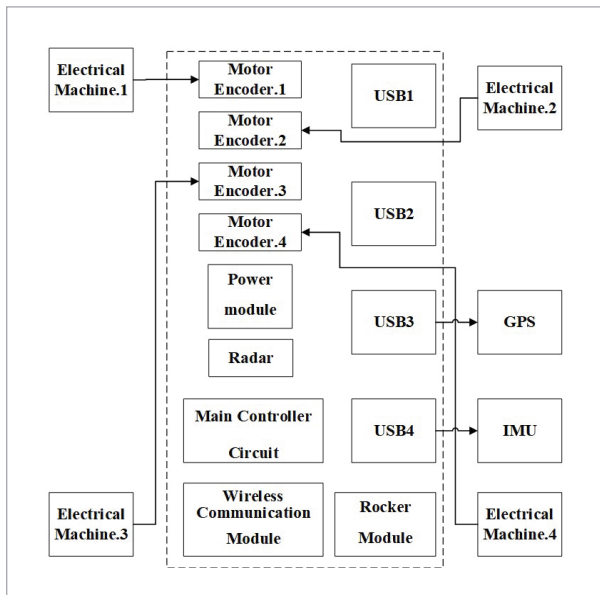


Figure 13
Robot operation road map



Figure 12
The structure design of robot



The robot and its structure used in the experiment are as shown in Figures 11-12, experimental field is as shown in Figures 13.

The parameters of gyroscope and accelerometer parameters in IMU used in the inspection robot are as shown in Table 4, the actual outputs of accelerometer and gyroscope are shown in Figures 14-15, the basic parameters of the robot are as shown in Table 5,

Figure 14
Actual output of accelerometer

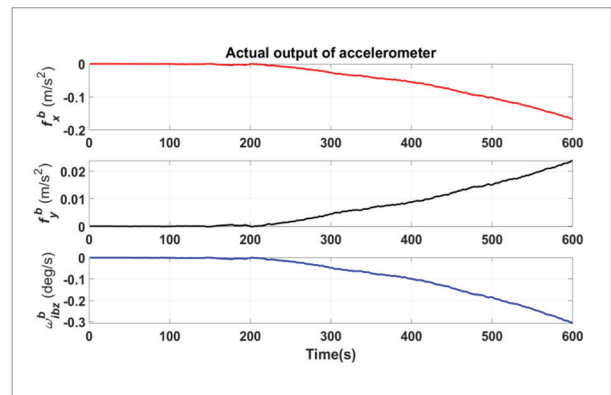


Figure 15
Actual output of gyroscope

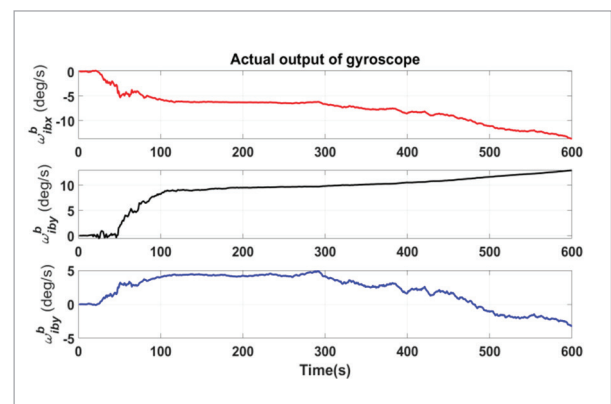


Table 4

Gyro and accelerometer parameters

	Gyroscope	Accelerometer
Enter the data range	$\pm 600^\circ/\text{s}$	$\pm 10\text{g}$
Zero offset	$0.01^\circ/\text{h}$	$1\text{mg}'$
Scaling factor error	3ppm	10ppm
Random walk	$0.03^\circ/\text{h}^{1/2}$	$10\mu\text{g}/\text{Hz}^{1/2}$
Output data type	angle increment	velocity gain

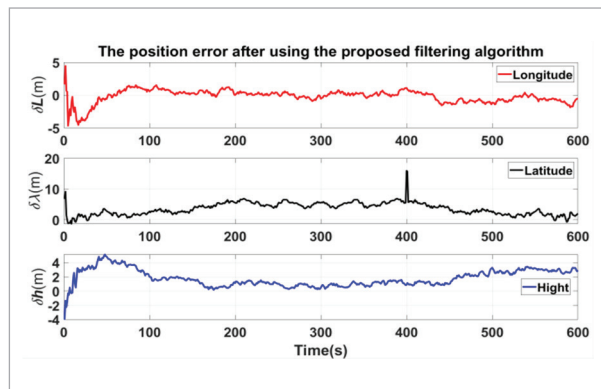
The navigation running time length is 600s, actual trajectory of the robot is shown in Figure 13. The actual robot road map given by the map software can be seen that the robot trajectory is surrounded by buildings and jungle, and the satellite navigation observation conditions are not good. In this case, the advantages of combined navigation can be well-reflected.

The measured data of the combined navigation filtering algorithm before and after the improved filtering algorithm was uniformly recorded and compiled into Tables 6-7, which contains the error mean/variance statistical results of the robot during the measured process, controlling the data acquisition time at 600s, acquisition interval at 5s.

Figures 16-18 show the different error changes after using the filtering algorithm. It can be clearly seen from the figures that after using the proposed filtering algorithm, the filtering effect of the measured data is significantly improved. At the beginning of the

Figure 16

Position difference before and after optimization



three-dimensional velocity error in altitude, northward and eastward, there is a slight jump but it converges quickly, and it maintains a stable state after 300s. In terms of position error, the longitude and height errors are controlled within 5m, all three errors show a good filtering effect in general.

Figure 17

Velocity difference before and after optimization

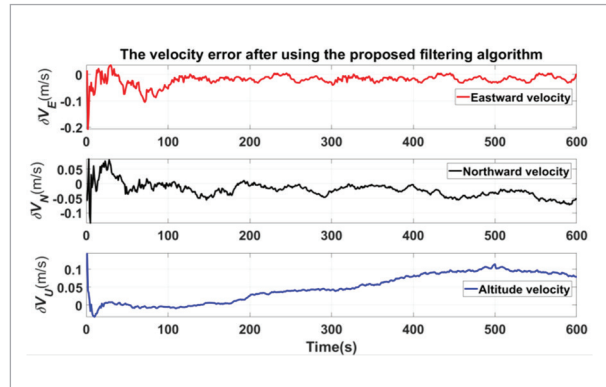
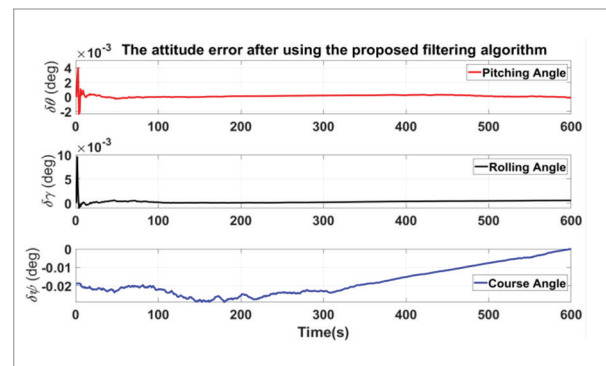


Figure 18

Attitude difference before and after optimization



After sorting the data collected by the robot, the experimental results before and after the improved algorithm were generated. In the statistics of the positioning error mean, that when the simulation time is 600s, the positioning error value near zero is very small, so the noise in the gyroscope and the accelerometer is zero mean Gaussian white noise, the improved algorithm and the improved combined navigation filter algorithm are the position estimation during the positioning. The positioning error variance of the three

Table 5

Basic parameters of the robot

GPS Model	Robot Model	Bottom-floor Drive Plate	Robot Size
BDS+GPS+ GLONASS	UFX-554072	STM32F 103RCT6	329mm×242mm× 440mm(3.1kg)
Operating System	Programming language	Battery capacity	Radar Model
Ubuntu mate 18.04	Python & C++	8000mA	Rplider A1
IMU Model	Straight-Line Speed (Max)	Drive Model Type	Electrical Machinery Type
Nine-axis gyro sensor	1.2m/s	Differential ID governor drive	The DC has brush motor

Table 6

Position error mean and error mean square

Position error mean(m)	After processing	Before processing
The X-axis	-0.0114	-0.0121
The Y-axis	-0.0226	-0.0422
The Z-axis	-0.0237	-0.0531
Position error mean square(m ²)	After processing	Before processing
The X-axis	16.8126	19.0581
The Y-axis	42.9512	51.8275
The Z-axis	33.2949	37.0410

Table 7

Velocity error mean and error mean square

Velocity error mean(m/s)	After processing	Before processing
The X-axis	-0.0193	-0.0281
The Y-axis	-0.0295	-0.0395
The Z-axis	-0.0437	-0.0594
Velocity error mean square(m/s ²)	After processing	Before processing
The X-axis	0.4169	0.5214
The Y-axis	1.3294	1.5015
The Z-axis	0.9325	1.0125

axes shows through the comparison of the proposed algorithm, the result of the combined navigation filtering algorithm measures the measured noise and optimizes the posterior error covariance matrix using the BPNN to control the error in the ideal range. It is shown that the improved filtering algorithm has improved significantly.

6. Conclusion

In this paper, we proposed an adaptive fuzzy and neural network controller base on PGAF with variable step size could be used in the prior uncertainty and higher requirements for measurement accuracy, looking at the results of the experiment as a whole, the proposed algorithm improves the estimation accuracy and robustness obviously. As compared with the existing PGAF-VS, under the premise that less gradual progress is required, the proposed filter can achieve better estimation accuracy. But considering that this paper has some limitations, for example, the non-Gaussian noise cannot be considered in this paper's research range, so the next research goal will be to solve the problem about nonlinear filtering optimization under different noise distributions.

Acknowledgement

This paper has not received any funding.

References

1. Abdelnour, G., Ch, S., Chiu, S. Applying Fuzzy Logic to the Kalman Filter Divergence Problem. In Proceedings of IEEE Systems Man and Cybernetics Conference - SMC, 1993, 1, 630-635. <https://doi.org/10.1109/ICSMC.1993.384814>
2. Akhlaghi, S., Zhou, N., Huang, Z. Adaptive Adjustment of Noise Covariance in Kalman Filter for Dynamic State Estimation. In 2017 IEEE Power Energy Society General Meeting, 2017, 1-5. <https://doi.org/10.1109/PES-GM.2017.8273755>
3. Bai, M., Huang, Y., Zhang, Y., Jia, G. A Novel Progressive Gaussian Approximate Filter for Tightly Coupled Gnss/Ins Integration. IEEE Transactions on Instrumentation and Measurement, 2020, 69(6), 3493-3505. <https://doi.org/10.1109/TIM.2019.2932155>
4. Chen, M., Chen, K. Research on Novel Normal Fuzzy Kalman Filter for Speech Enhancement in Noisy Environment. In 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA), 2021, 88-92. <https://doi.org/10.1109/ICPECA51329.2021.9362606>
5. Crassidis, J. L. Sigma-point Kalman Filtering for Integrated GPS and Inertial Navigation. IEEE Transactions on Aerospace and Electronic Systems, 2006, 42(2), 750-756. <https://doi.org/10.1109/TAES.2006.1642588>
6. Fu, Z., Luo, Z. A Vehicle Driving State Estimation Algorithm Based on Kelman Neural Network and Unscented Kalman Filter. In 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2021, 415-419. <https://doi.org/10.1109/IAEAC50856.2021.9390729>
7. Guang, H., Ji, L. Bayesian State Estimation in Sensorimotor Systems with Particle Filtering. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2020, 28(7), 1528-1538. <https://doi.org/10.1109/TNSRE.2020.2996963>
8. Huang, Y., Zhang, Y. A New Process Uncertainty Robust Student's t Based Kalman Filter for Sins/Gps Integration. IEEE Access, 2017, 5, 14391-14404. <https://doi.org/10.1109/ACCESS.2017.2726519>
9. Huang, Y., Zhang, Y. Design of High-Degree Student's t-Based Cubature Filters. Circuits, Systems, and Signal Processing, 2018, 37(6), 2206-2225. <https://doi.org/10.1007/s00034-017-0662-y>
10. Huang, Y., Zhang, Y., Wu, Z., Li, N., Chambers, J. A Novel Adaptive Kalman Filter with Inaccurate Process and Measurement Noise Covariance Matrices. IEEE Transactions on Automatic Control, 2018, 63(2), 594-601. <https://doi.org/10.1109/TAC.2017.2730480>
11. Huang, Y., Zhang, Y., Xu, B., Wu, Z., Chambers, J. A. A New Adaptive Extended Kalman Filter for Cooperative Localization. IEEE Transactions on Aerospace and Electronic Systems, 2018, 54(1), 353-368. <https://doi.org/10.1109/TAES.2017.2756763>
12. Lai, X., Zhu, G. R., Chambers, J. A. A Fuzzy Adaptive Extended Kalman Filter Exploiting the Students t. Distribution for Mobile Robot Tracking, 2021, 32(10), 105017. <https://doi.org/10.1088/1361-6501/ac0ca9>
13. Lai, X., Zhu, G. R., Chambers, J. A. Robust Student's t-based Trajectory Tracking for Inspection Wall-climbing Robot. In 2020 International Conference on Robots Intelligent System (ICRIS), 2020, 36-39. <https://doi.org/10.1109/ICRIS52159.2020.00017>
14. Li, S., Ma, W., Liu, J., Chen, H. A Kalman Gain Modify Algorithm Based on BP Neural Network. In 2016 16th International Symposium on Communications and Information Technologies (ISCIT), 2016, 453-456. <https://doi.org/10.1109/ISCIT.2016.7751672>
15. Matia, F., Jimenez, V., Alvarado, B. P., Haber, R. The fuzzy Kalman Filter: Improving Its Implementation by Reformulating Uncertainty Representation. Fuzzy Sets and Systems, 2021, 402, 78-104. <https://doi.org/10.1016/j.fss.2019.10.015>
16. Murray, P. M., Browne, R. P., McNicholas, P. D. A Mixture of SDB Skew-t Factor Analyzers. Econometrics and Statistics, 2017, 3, 160-168. <https://doi.org/10.1016/j.ecosta.2017.05.001>
17. Nourmohammadi, H., Keighobadi, J. Fuzzy Adaptive Integration Scheme for Low-Cost Sins/Gps Navigation System. Mechanical Systems and Signal Processing, 2018, 99, 434-449. <https://doi.org/10.1016/j.ymssp.2017.06.030>
18. Nurminen, H., Ardeshiri, T., Piche, R., Gustafsson, F. Skew- Filter and Smoother with Improved Covariance Matrix Approximation. IEEE Transactions on Signal Processing, 2018, 66(21), 5618-5633. <https://doi.org/10.1109/TSP.2018.2865434>
19. Petkovic, D., Gocic, M., Shamshirb, S. Adaptive Neuro-Fuzzy Computing Technique for Precipitation Estimation. Facta Universitatis. Series: Mechanical Engineering, 2016, 14(2), 209-218. <https://doi.org/10.22190/FUME1602209P>

20. Pires, D. S., Serra, G. L. O. Methodology for Modeling Fuzzy Kalman Filters of Minimum Realization from Evolving Clustering of Experimental Data. *ISA Transactions*, 2020, 105, 1-23. <https://doi.org/10.1016/j.isatra.2020.05.034>
21. Qi, H., Moore, J. B. Direct Kalman Filtering Approach for Gps/Ins Integration. *IEEE Transactions on Aerospace and Electronic Systems*, 2002, 38(2), 687-693. <https://doi.org/10.1109/TIM.2018.2805231>
22. Sarkka, S., Nummenmaa, A. Recursive Noise Adaptive Kalman Filtering by Variational Bayesian Approximations. *IEEE Transactions on Automatic Control*, 2009, 54(3), 596-600. <https://doi.org/10.1109/TAC.2008.2008348>
23. Sasiadek, J. Z., Wang, Q., Zeremba, M. B. Fuzzy Adaptive Kalman Filtering For Ins/Gps Data Fusion. In *Proceedings of the 2000 IEEE International Symposium on Intelligent Control*. Held jointly with the 8th IEEE Mediterranean Conference on Control and Automation (Cat. No.00CH37147), 2000, 181-186. <https://doi.org/10.1109/ISIC.2000.882920>
24. Sen, G. D., Gunel, G. O., Guzelkaya, M. Extended Kalman Filter Based Modified Elman-Jordan Neural Network for Control and Identification of Nonlinear Systems. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, 2020, 1-6. <https://doi.org/10.1109/ASYU50717.2020.9259812>
25. Tagle, F., Castruccio, S., Crippa, P., Genton, M. Assessing Potential Wind Energy Resources in Saudi Arabia with a Skew-t Distribution. 2019, 02, 159-178. <https://arxiv.org/abs/1703.04312v2>
26. Tan, F., San, L. H., Abdullah, K. Remote Sensing Study Between Relationship of the Cloud and Rainfall Patterns of the Tropical Cyclone on Orographic Effects. In *2011 IEEE International Conference on Imaging Systems and Techniques*, 2011, 39-44. <https://doi.org/10.1109/IST.2011.5962223>
27. Tseng, C. h., Lin, S. F., Jwo, D. J. Fuzzy Adaptive Cubature Kalman Filter for Integrated Navigation Systems. *Sensors*, 2016, 16(8), 1167. <https://doi.org/10.3390/s16081167>
28. Tseng, C. H., Lin, S. F., Jwo, D. J. Robust Huber-based Cubature Kalman Filter for GPS Navigation Processing. *Journal of Navigation*, 2016, 70(3), 1-20. <https://doi.org/10.1017/S0373463316000692>
29. Tzikas, D. G., Likas, A. C., Galatsanos, N. P. The Variational Approximation for Bayesian Inference. *IEEE Signal Processing Magazine*, 2008, 25(6), 131-146. <https://doi.org/10.1109/MSP.2008.929620>
30. Wang, J., Zhang, T., Xu, X., Li, Y. A Variational Bayesian Based Strong Tracking Interpolatory Cubature Kalman Filter for Maneuvering Target Tracking. *IEEE Access*, 2018, 6, 52544-52560. <https://doi.org/10.1109/ACCESS.2018.2869020>
31. Wang, Y., Sun, S., Tian, Y., Sun, J., Xu, L. Image Reconstruction Based on Fuzzy Adaptive Kalman Filter in Electrical Capacitance Tomography. *IEEE Transactions on Instrumentation and Measurement*, 2021, 70, 1-10. <https://doi.org/10.1109/TIM.2021.3099563>
32. Wang, Z., Zhou, W., Xue, L., Ling, H., Chen, T. An EKF Algorithm Based on a Skew-t Measurement Noise Model. In *2017 36th Chinese Control Conference (CCC)*, 2017, 5105-5109. <https://doi.org/10.23919/ChiCC.2017.8028161>
33. Xu, Q., Li, X., Chan, C. Enhancing Localization Accuracy of Mems-Ins/Gps/In-Vehicle Sensors Integration During GPS Outages. *IEEE Transactions on Instrumentation and Measurement*, 2018, 67(8), 1966-1978. <https://doi.org/10.1109/TAES.2002.1008998>
34. Zakeri, E., Moezi, S. A., Eghtesad, M. Optimal Interval Type-2 Fuzzy Fractional Order Super Twisting Algorithm: A Second Order Sliding Mode Controller for Fully-Actuated and Under-Actuated Nonlinear Systems. *ISA Transactions*, 2019, 85, 13-32. <https://doi.org/10.1016/j.isatra.2018.10.013>
35. Zhang, D., Hu, G., Lu, J., Yin, X., Ren, Q. Short-term Load Forecasting Based on Gad-BP Neural Network. In *2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 2019, 1525-1529. <https://doi.org/10.1109/IMCEC46724.2019.8984012>

