International Journal of Innovative Computing, Information and Control
Volume **22**, Number **3**, June **2026**

ICIC International ©2026 ISSN 1349-4198
pp. **599–610**

ON-LINE IDENTIFICATION OF NONLINEAR SYSTEMS USING GAUSSIAN PROCESS MODEL AND GENETIC ALGORITHM WITH VARIABLE NUMBER OF GENERATIONS

Tomohiro Hachino* and Daijiro Yamasaki

Graduate School of Science and Engineering
Kagoshima University
1-21-40 Korimoto, Kagoshima 890-0065, Japan
k8979032@kadai.jp
*Corresponding author: hachino@eee.kagoshima-u.ac.jp

Received August 2025; revised November 2025

Abstract. *This paper presents an on-line identification of continuous-time nonlinear systems using a moving-window type Gaussian process (GP) model and genetic algorithm (GA) with variable number of generations. The GP is a Gaussian random function and is constructed by its mean function and covariance function. During the training phase of this model, the hyperparameters of the covariance function and the moving-window length are optimized by the modified GA, whereas the system parameters of the linear terms and the weighting parameters of the mean function are estimated by the recursive least-squares (RLS) method. In order to reduce the computational burden, the number of generations of the GA is adjusted according to the time changes of the system and only the input-output data extracted by the k-means method are utilized in both GA and RLS method. The effectiveness of the proposed method is demonstrated through numerical simulations for a simplified power system. Simulation results show that the computational time of the proposed method is reduced to about 40% of that of the conventional method without deteriorating the identification accuracy.*

Keywords: On-line identification, Nonlinear systems, Gaussian process model, Genetic algorithm, k-means method

1. **Introduction.** Most real systems in the world are nonlinear systems that have various nonlinearities. In order to control and analyze these systems on-line, it is necessary to instantly and accurately detect time changes such as sudden failures and abnormalities during system operation. Therefore, the development of on-line identification method for nonlinear systems is an urgent issue. So far, several on-line identification methods have been proposed, mainly for discrete-time nonlinear systems [1-6]. However, in general, the parameters included in the discrete-time systems do not correspond to real physical quantities, so it is important to consider on-line identification for continuous-time systems. Unfortunately, on-line identification methods for continuous-time nonlinear systems have been seldom studied. In [7,8], radial basis function model-based identification algorithms have been presented by the separable least-squares approach combining the recursive least-squares (RLS) method with the genetic algorithm (GA) [9] or the immune algorithm. These methods require a large number of parameters to be determined and do not provide any information on the reliability or uncertainty of the estimated nonlinear functions. The authors have presented the moving-window type on-line identification based on the Gaussian process (GP) model [10-12]. The GP model was originally proposed by O' Hagan for the regression problem [13]. Since then, the model has been received much attention

DOI: 10.24507/ijicic.22.03.599

for regression, classification and modeling of the dynamic systems [14-18]. The GP model fits naturally into Bayesian paradigm and can express nonlinearity with a small number of parameters. Furthermore, this model can provide not only an estimated nonlinear function but also a measure of its reliability. In [10], the hyperparameters of the covariance function are searched for using the firefly algorithm, while the system parameters of the linear terms and the weighting parameters of the mean function are updated by the RLS method. In [11], an on-line identification method is proposed in which the GP prior model is trained only by particle swarm optimization. To reduce the computational burden of these two methods, a GP-based online identification method using k-means method [19] has been also proposed in [12]. This method aims to reduce the computational burden by making the moving-window length variable and performing identification using a small number of input-output data extracted by the k-means method. However, since the maximum number of generations in the GA is fixed at each time step and all input-output data in the moving window are used in the RLS method, the reduction in the computational burden is still insufficient.

To realize faster on-line identification of continuous-time nonlinear systems, in this paper we propose a GP model identification based on the GA with variable number of generations and the RLS method using only partial training data. In this method, the maximum number of generations of the GA at each time step is adjusted according to the time changes of the target system. In addition, only the input-output data extracted by the k-means method are used in both GA and RLS method. The proposed method improves the efficiency of optimization by GA and reduces the amount of training data, thereby reducing the computational burden required for identification. Moreover, to suppress the vibration of the estimated parameters of the linear terms caused by the reduction of the number of training data in the RLS method, a moving average process is applied to the raw estimated parameters. The proposed identification method can provide information on the reliability of the estimated nonlinear function, and maintain high performance in terms of accuracy and tracking ability while reducing the computational cost.

This paper is organized as follows. In Section 2, the target system to be estimated is given and the problem statement is presented. In Section 3, the identification model is derived using GP prior model. In Section 4, the on-line identification algorithm using the GA with variable number of generations and the RLS method is proposed. In Section 5, the effectiveness of the proposed method is demonstrated through numerical simulations for a simplified power system. Finally, conclusions are stated in Section 6.

2. **Statement of the Problem.** Consider a single-input, single-output, continuous-time nonlinear system:

$$\sum_{\substack{i=0\\ i\neq n_1,n_2,\ldots,n_\alpha}}^{n} a_i p^{n-i} x(t) = f(\boldsymbol{z}(t)) + \sum_{\substack{j=0\\ j\neq m_1,m_2,\ldots,m_\beta}}^{m} b_j p^{m-j} u(t) \quad (a_0 = 1,\ n \geq m)$$

$$\boldsymbol{z}(t) = \left[p^{n-n_1}x(t), p^{n-n_2}x(t), \ldots, p^{n-n_\alpha}x(t), p^{m-m_1}u(t), p^{m-m_2}u(t), \ldots, p^{m-m_\beta}u(t)\right]^{\mathrm{T}}$$

$$y(t) = x(t) + e(t) \tag{1}$$

where p is the differential operator, $u(t)$ and $x(t)$ are the true input and output signals, respectively, and $y(t)$ is the noisy output corrupted by the measurement noise $e(t)$. $f(\cdot)$ is an unknown nonlinear function, which is assumed to be stationary and smooth. n, n_i $(i = 1, 2, \ldots, \alpha)$, m, and m_j $(j = 1, 2, \ldots, \beta)$ are assumed to be known. The goal of this paper is to identify the system parameters $\{a_i\}$ and $\{b_j\}$ of the linear terms and the

nonlinear function $f(\cdot)$ with the confidence measure, in on-line operation from the true input and noisy output data.

3. **Identification Model by GP Prior Model.** Equation (1) is rewritten as follows:

$$p^n y(t) = f(\boldsymbol{w}(t)) - \sum_{\substack{i=1 \\ i\neq n_1,n_2,\ldots,n_\alpha}}^{n} a_i p^{n-i} y(t) + \sum_{\substack{j=0 \\ j\neq m_1,m_2,\ldots,m_\beta}}^{m} b_j p^{m-j} u(t) + \varepsilon(t)$$

$$\boldsymbol{w}(t) = \left[p^{n-n_1}y(t), p^{n-n_2}y(t), \ldots, p^{n-n_\alpha}y(t), p^{m-m_1}u(t), p^{m-m_2}u(t), \ldots, p^{m-m_\beta}u(t)\right]^{\mathrm{T}} \quad (2)$$

where $\varepsilon(t)$ is an error due to the measurement noise $e(t)$. The delayed state variable filter $F(p)$ is introduced to deal with the higher-order derivative of signals in (2). Multiplying both sides of (2) by $F(p)$ yields

$$p^n y^f(t) = f\left(\boldsymbol{w}^f(t)\right) - \sum_{\substack{i=1 \\ i\neq n_1,n_2,\ldots,n_\alpha}}^{n} a_i p^{n-i} y^f(t) + \sum_{\substack{j=0 \\ j\neq m_1,m_2,\ldots,m_\beta}}^{m} b_j p^{m-j} u^f(t) + \varepsilon^f(t) \quad (3)$$

$$y^f(t) = F(p)y(t), \quad u^f(t) = F(p)u(t), \quad \varepsilon^f(t) = F(p)\varepsilon(t), \quad \boldsymbol{w}^f(t) = F(p)\boldsymbol{w}(t)$$

where $\varepsilon^f(t)$ is assumed to be zero-mean Gaussian noise with variance σ_n^2. Discretizing (3) as $t = t_1, t_2, \ldots, t_N$ yields

$$\boldsymbol{y} = \boldsymbol{f} + \boldsymbol{G}\boldsymbol{\theta}_l + \boldsymbol{\varepsilon} \quad (4)$$

where

$$\begin{aligned}
&\boldsymbol{y} = \left[p^n y^f(t_1), p^n y^f(t_2), \ldots, p^n y^f(t_N)\right]^{\mathrm{T}} \\
&\boldsymbol{f} = \left[f(\boldsymbol{w}^f(t_1)), f(\boldsymbol{w}^f(t_2)), \ldots, f(\boldsymbol{w}^f(t_N))\right]^{\mathrm{T}} \\
&\boldsymbol{G} = \left[\boldsymbol{g}(t_1), \boldsymbol{g}(t_2), \ldots, \boldsymbol{g}(t_N)\right]^{\mathrm{T}} \\
&\boldsymbol{g}(t) = \left[-p^{n-1}y^f(t), \ldots, -p^{n-i}y^f(t), \ldots, -y^f(t), p^m u^f(t), \ldots, p^{m-j}u^f(t), \ldots, u^f(t)\right]^{\mathrm{T}} \\
&\boldsymbol{\theta}_l = \left[a_1, \ldots, a_i, \ldots, a_n, b_0, \ldots, b_j, \ldots, b_m\right]^{\mathrm{T}} \\
&\boldsymbol{\varepsilon} = \left[\varepsilon^f(t_1), \varepsilon^f(t_2), \ldots, \varepsilon^f(t_N)\right]^{\mathrm{T}}
\end{aligned} \quad (5)$$

$\boldsymbol{\theta}_l$ is the unknown parameters vector which is composed of the system parameters of the linear terms.

The function value vector $\boldsymbol{f}$ in (4) can be represented by the GP as follows:

$$\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{m}(\boldsymbol{w}), \boldsymbol{\Sigma}(\boldsymbol{w}, \boldsymbol{w})) \quad (6)$$

where $\boldsymbol{w} = \left[\boldsymbol{w}^f(t_1), \boldsymbol{w}^f(t_2), \ldots, \boldsymbol{w}^f(t_N)\right]$ is the input of the function $\boldsymbol{f}$, $\boldsymbol{m}(\boldsymbol{w})$ is the mean function vector, and $\boldsymbol{\Sigma}(\boldsymbol{w}, \boldsymbol{w})$ is the covariance matrix. In this paper, the mean function vector $\boldsymbol{m}(\boldsymbol{w})$ is given by the linear combination of the input variable as follows:

$$\begin{aligned}
&\boldsymbol{m}(\boldsymbol{w}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\theta}_m \\
&\boldsymbol{\theta}_m = \left[\theta_{m_1}, \theta_{m_2}, \ldots, \theta_{m_{(\alpha+\beta)}}\right]^{\mathrm{T}}
\end{aligned} \quad (7)$$

where $\boldsymbol{\theta}_m$ is the unknown weighting parameter vector of the mean function. The Gaussian kernel is introduced for each element Σ_{pq} of the covariance matrix $\boldsymbol{\Sigma}(\boldsymbol{w}, \boldsymbol{w})$ as follows:

$$\Sigma_{pq} = s\left(\boldsymbol{w}^f(t_p), \boldsymbol{w}^f(t_q)\right) = \sigma_y^2 \exp\left(-\frac{\|\boldsymbol{w}^f(t_p) - \boldsymbol{w}^f(t_q)\|^2}{2\ell^2}\right) \quad (8)$$

where $\|\cdot\|$ is the Euclidean norm.

From (4) and (6), the identification model based on the GP prior is derived as follows:

$$\begin{aligned}
&\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{m}(\boldsymbol{w}) + \boldsymbol{G}\boldsymbol{\theta}_l, \boldsymbol{K}(\boldsymbol{w}, \boldsymbol{w})) \\
&\boldsymbol{K}(\boldsymbol{w}, \boldsymbol{w}) = \boldsymbol{\Sigma}(\boldsymbol{w}, \boldsymbol{w}) + \sigma_n^2 \boldsymbol{I}_N \quad (\boldsymbol{I}_N : N \times N \text{ identity matrix})
\end{aligned} \quad (9)$$

$\boldsymbol{\theta}_c = [\sigma_y, \ell, \sigma_n]^{\mathrm{T}}$ is called hyperparameter vector of the covariance function and is composed of σ_y and ℓ in (8) and σ_n in (9). For simplicity, $\boldsymbol{K}(\boldsymbol{w}, \boldsymbol{w})$ is denoted as $\boldsymbol{K}$ below.

4. **On-line Identification by the GA.** GP prior model is trained on-line by optimizing the unknown parameter vector $\boldsymbol{\theta} = \left[\boldsymbol{\theta}_m^{\mathrm{T}}, \boldsymbol{\theta}_l^{\mathrm{T}}, \boldsymbol{\theta}_c^{\mathrm{T}}\right]^{\mathrm{T}}$ and the moving-window length W. $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_l$ are estimated by the RLS method, while $\boldsymbol{\theta}_c$ and W are optimized by the GA in which the generation number of each time step is variable.

The flow of the on-line identification is shown below.

step 1: Initialization of time step

Let $k = k_s$ be the starting time step for identification and set $\hat{\boldsymbol{\theta}}_{ml}(k_s) = \left[\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}(k_s), \hat{\boldsymbol{\theta}}_l^{\mathrm{T}}(k_s)\right]^{\mathrm{T}}$ as the initial value of $\boldsymbol{\theta}_{ml} = \left[\boldsymbol{\theta}_m^{\mathrm{T}}, \boldsymbol{\theta}_l^{\mathrm{T}}\right]^{\mathrm{T}}$.

step 2: Initialization of GA

Set the generation counter $g = 1$ and the initial generation number $g_{\max}(k)$ ($G_{\min} \leq g_{\max}(k) \leq G_{\max}$). Generate a random initial population of Q binary strings $\boldsymbol{\Omega}_{[i]}$ ($i = 1, 2, \ldots, Q$) for the hyperparameter vector $\boldsymbol{\theta}_c$ and the moving-window length W.

step 3: Decoding

Decode Q binary strings into real values $\tilde{\boldsymbol{\Omega}}_{[i]} = \left[\boldsymbol{\theta}_{c[i]}^{\mathrm{T}}, W_{[i]}\right]^{\mathrm{T}}$ ($i = 1, 2, \ldots, Q$). The search range of $W_{[i]}$ is $[W_{\min}, W_{\max}]$.

step 4: Collection of input-output data

Collect the datasets $\mathcal{D}_{[i]} = \{[y(k), u(k)], [y(k-1), u(k-1)], \ldots, [y(k - W_{[i]} + 1), u(k - W_{[i]} + 1)]\}$ ($i = 1, 2, \ldots, Q$) which consist of $W_{[i]}$ sets of input-output data.

step 5: Initialization of k-means method

Set the iteration counter $h = 0$. Select $N_{[i]}$ initial centers (centroids) of the clusters $\boldsymbol{C}_{[i]}^h(p)$ ($p = 1, 2, \ldots, N_{[i]}$) from $\mathcal{D}_{[i]}$ randomly, where $N_{[i]} = W_{[i]}/10$.

step 6: Data classification

Calculate the Euclidean distances between $W_{[i]}$ input-output datasets and $N_{[i]}$ centroids $\boldsymbol{C}_{[i]}^h(p)$, and classify the datasets into the clusters with the closest centroids.

step 7: Update of centroids

Let the centers of the datasets belonging to the clusters be the new centroids of the clusters $\boldsymbol{C}_{[i]}^{h+1}(p)$.

step 8: Repetition of k-means method

If the iteration counter $h < h_{\max}$, set the iteration counter $h = h + 1$ and go to step 6.

step 9: Extraction of the partial data

Extract the data closest to the centers of the final clusters $\boldsymbol{C}_{[i]}^{h_{\max}}(p)$ as the partial data $\mathcal{D}_{[i]}^*$ ($i = 1, 2, \ldots, Q$) of $\mathcal{D}_{[i]}$ ($i = 1, 2, \ldots, Q$).

step 10: Construction of the covariance matrix

Construct Q candidates of the covariance matrix $\boldsymbol{K}_{[i]}$ using $\boldsymbol{\theta}_{c[i]}$ and the partial datasets $\mathcal{D}_{[i]}^*$.

step 11: Fitness value evaluation

Calculate the negative log marginal likelihood of the partial datasets $\mathcal{D}_{[i]}^*$:

$$J_{[i]} = \frac{1}{2}\log|\boldsymbol{K}_{[i]}| + \frac{1}{2}\left(\boldsymbol{y}_{[i]} - \boldsymbol{Z}_{[i]}\boldsymbol{\theta}_{ml}(k)\right)^{\mathrm{T}}\boldsymbol{K}_{[i]}^{-1}\left(\boldsymbol{y}_{[i]} - \boldsymbol{Z}_{[i]}\boldsymbol{\theta}_{ml}(k)\right) + \frac{N_{[i]}}{2}\log(2\pi) \tag{10}$$

where $\boldsymbol{Z}_{[i]} = \left[\boldsymbol{w}_{[i]}^{\mathrm{T}} \vdots \boldsymbol{G}_{[i]}\right]$. Evaluate the fitness values $F_{[i]} = C - J_{[i]}$, where C is a positive constant.

step 12: Reproduction

Reproduce each of the individual binary strings with a probability of $F_{[i]}/\sum_{j=1}^{Q} F_{[j]}$.

step 13: Crossover

Select two binary strings randomly and replace each element of the strings with the crossover probability p_c according to a randomly generated mask sequence.

step 14: Mutation

Alter a bit (0 or 1) of the binary string in accordance with the mutation probability p_m.

step 15: Repetition of GA

If the generation counter $g < g_{\max}(k)$, set the generation counter $g = g + 1$ and go to step 3.

step 16: Determination of the GP prior model

Determine the vector $\hat{\boldsymbol{\Omega}}(k) = \left[\hat{\boldsymbol{\theta}}_c^{\mathrm{T}}(k), \hat{W}(k)\right]^{\mathrm{T}} = \left[\hat{\sigma}_y(k), \hat{\ell}(k), \hat{\sigma}_n(k), \hat{W}(k)\right]^{\mathrm{T}}$ and the corresponding partial data $\mathcal{D}^*(k)$ from the string with the best fitness value over all the past generations. Construct the suboptimal prior mean function and the prior covariance function:

$$\begin{aligned} &\hat{m}\left(\boldsymbol{w}^f(t)\right) = \left(\boldsymbol{w}^f(t)\right)^{\mathrm{T}} \hat{\boldsymbol{\theta}}_m(k) \\ &\begin{cases} \hat{s}\left(\boldsymbol{w}^f(t_p), \boldsymbol{w}^f(t_q)\right) = \hat{\sigma}_y^2(k) \exp\left(-\dfrac{\left\|\boldsymbol{w}^f(t_p) - \boldsymbol{w}^f(t_q)\right\|^2}{2\hat{\ell}^2(k)}\right) \\ \hat{k}\left(\boldsymbol{w}^f(t_p), \boldsymbol{w}^f(t_q)\right) = \hat{s}\left(\boldsymbol{w}^f(t_p), \boldsymbol{w}^f(t_q)\right) + \hat{\sigma}_n^2(k)\delta_{pq} \end{cases} \end{aligned} \tag{11}$$

where $\hat{s}\left(\boldsymbol{w}^f(t_p), \boldsymbol{w}^f(t_q)\right)$ is an element of the estimated covariance matrix $\hat{\boldsymbol{\Sigma}}(k)$, $\hat{k}\big(\boldsymbol{w}^f(t_p), \boldsymbol{w}^f(t_q)\big)$ is an element of the estimated covariance matrix $\hat{\boldsymbol{K}}(k)$, and δ_{pq} is the Kronecker delta, which is 1 if $p = q$ and 0 otherwise.

step 17: Estimation of the nonlinear function

Estimate the predictive mean function:

$$\hat{f}\left(\boldsymbol{w}_*^f(t)\right) = \hat{m}\left(\boldsymbol{w}_*^f(t)\right) + \hat{\boldsymbol{\Sigma}}\left(\boldsymbol{w}_*^f(t), \boldsymbol{w}\right)\left\{\hat{\boldsymbol{K}}(k)\right\}^{-1}\left(\boldsymbol{y} - \boldsymbol{w}^{\mathrm{T}}\hat{\boldsymbol{\theta}}_m(k) - \boldsymbol{G}\hat{\boldsymbol{\theta}}_l(k)\right) \tag{12}$$

and the predictive variance

$$\hat{\sigma}_*^2(t) = \hat{s}\left(\boldsymbol{w}_*^f(t), \boldsymbol{w}_*^f(t)\right) - \hat{\boldsymbol{\Sigma}}\left(\boldsymbol{w}_*^f(t), \boldsymbol{w}\right)\left\{\hat{\boldsymbol{K}}(k)\right\}^{-1}\hat{\boldsymbol{\Sigma}}\left(\boldsymbol{w}, \boldsymbol{w}_*^f(t)\right) \tag{13}$$

where $\boldsymbol{w}_*^f(t)$ is a new input of GP. $\boldsymbol{y}$, $\boldsymbol{w}$, and $\boldsymbol{G}$ are constructed according to the optimal value $\hat{N}(k) = \hat{W}(k)/10$. $\hat{f}\left(\boldsymbol{w}_*^f(t)\right)$ is the estimated nonlinear function and $\hat{\sigma}_*^2(t)$ is utilized as the confidence region of the estimated nonlinear function at the time step k.

step 18: Update of time step

Set the time step $k = k + 1$.

step 19: Estimation of $\boldsymbol{\theta}_{ml}$

Estimate $\hat{\boldsymbol{\theta}}_{ml}(k)$ with RLS method [10] using only the partial data $\mathcal{D}^*(k-1)$.

step 20: Smoothing of $\boldsymbol{\theta}_{ml}$

Obtain $\bar{\hat{\boldsymbol{\theta}}}_{ml}(k)$ by applying the moving average process to $\hat{\boldsymbol{\theta}}_{ml}(k)$, and use this as the final estimates of $\boldsymbol{\theta}_{ml}(k)$.

$$\bar{\hat{\boldsymbol{\theta}}}_{ml}(k) = \frac{1}{M}\sum_{i=k-M+1}^{k}\hat{\boldsymbol{\theta}}_{ml}(i) \tag{14}$$

step 21: Update of generation number

Determine the next generation number $g_{\max}(k)$ using the moving-window length $\hat{W}(k-1)$ as follows:

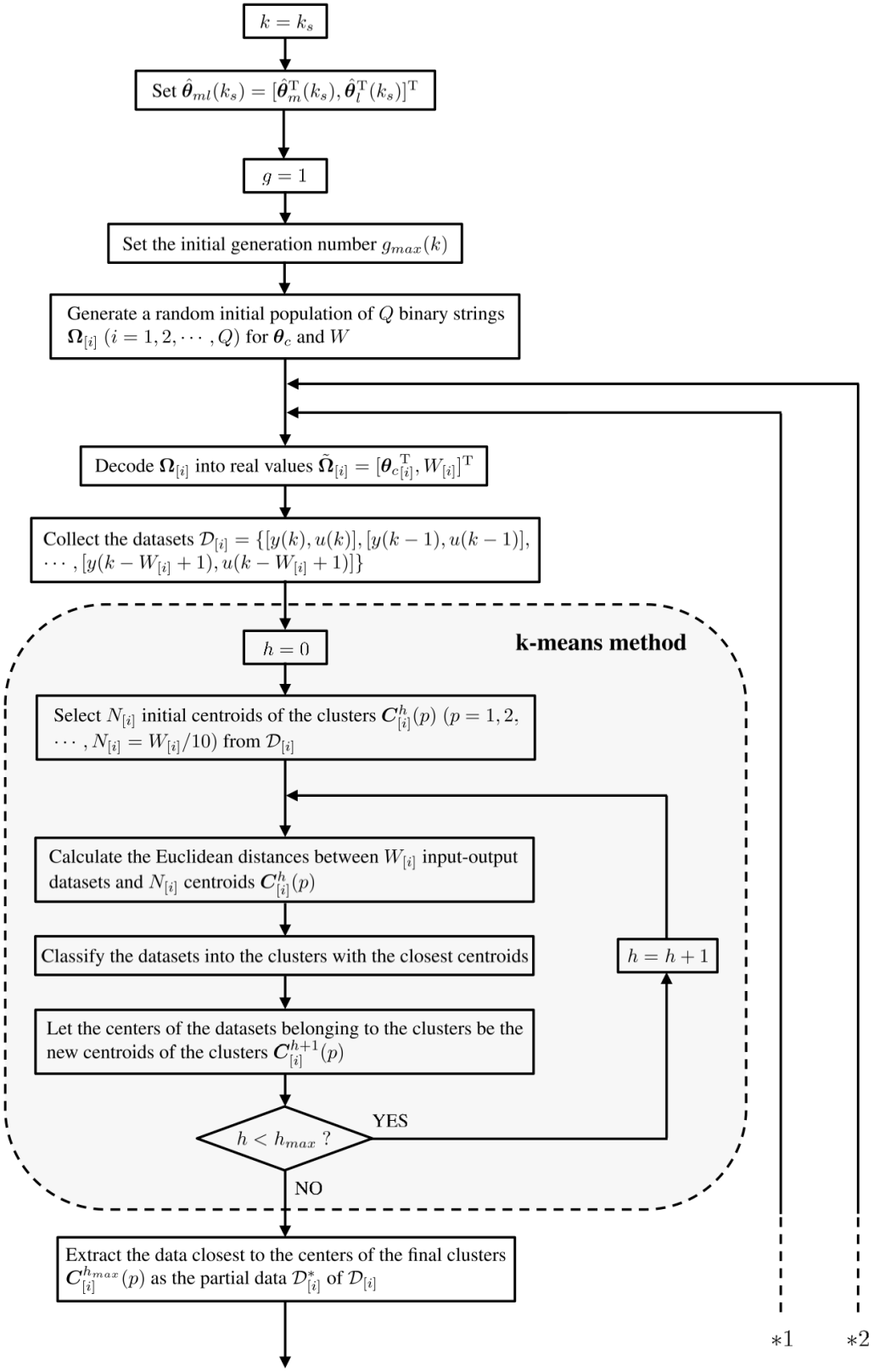
$$g_{\max}(k) = \mu\hat{W}(k-1) + \nu$$

$$\mu = (G_{\min} - G_{\max})/(W_{\max} - W_{\min}), \quad \nu = G_{\max} - \mu W_{\min} \tag{15}$$

step 22: Repetition

Set $g = 1$ and go to step 3.

The flowchart of the proposed method is shown in Figure 1.



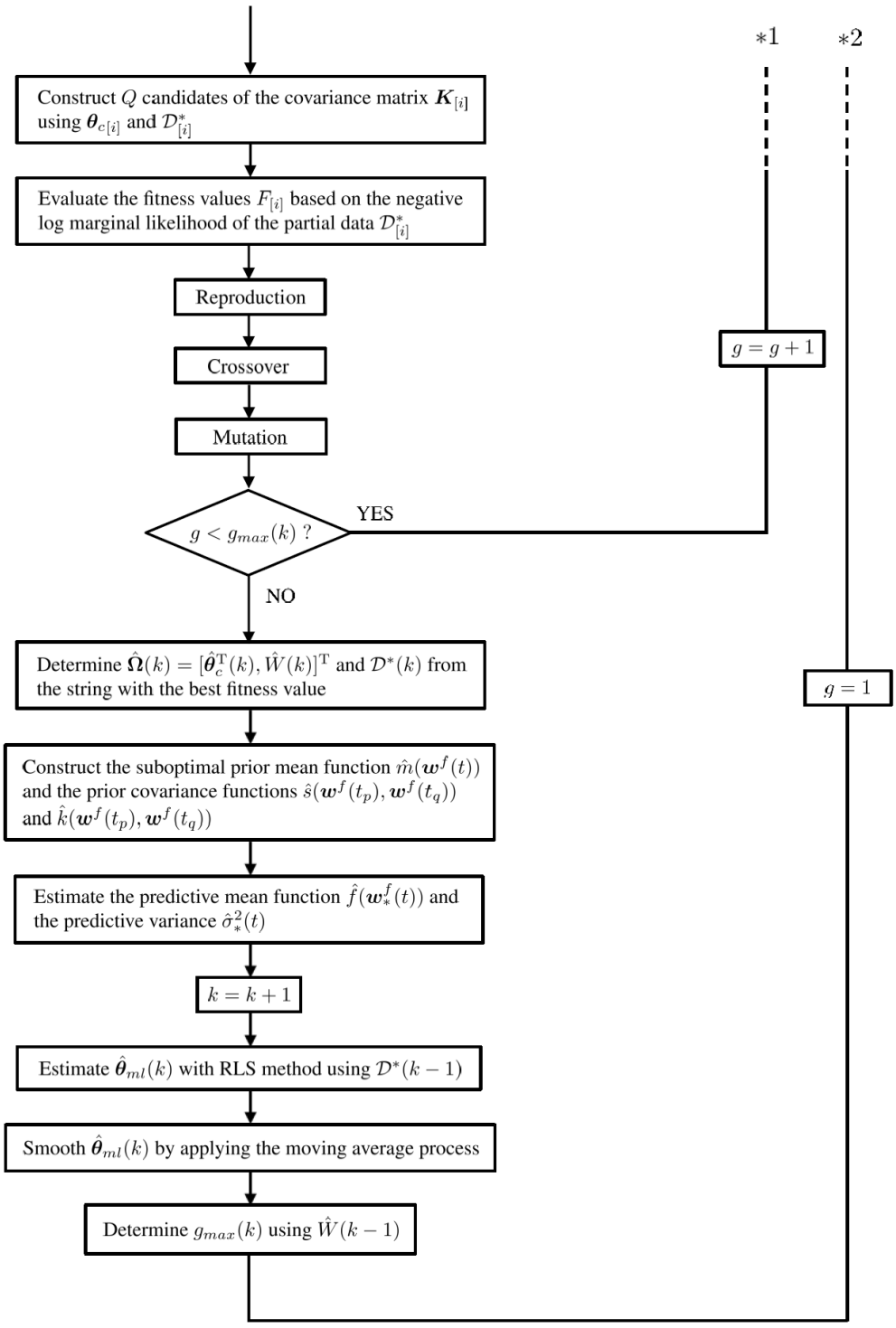


FIGURE 1. Flowchart of the proposed method

5. **Numerical Simulations.** Consider the following simplified power system represented by the oscillation equation [20]:

$$\begin{cases} \ddot{x}(t) + a_1 \dot{x}(t) = f(\boldsymbol{z}(t)) \\ f(\boldsymbol{z}(t)) = -\dfrac{P_e}{\tilde{M}} + \dfrac{P_{in}}{\tilde{M}} = -\dfrac{P_{em}}{\tilde{M}}(1 + u(t)) \sin x(t) + \dfrac{P_{in}}{\tilde{M}} \\ \boldsymbol{z}(t) = [x(t), u(t)]^{\mathrm{T}}, \quad y(t) = x(t) + e(t) \end{cases} \tag{16}$$

where $u(t) = \Delta E_{fd}(t)$: increment of excitation voltage, $x(t) = \delta(t)$: phase angle, $\tilde{M}$: inertia coefficient, $\tilde{D}$: damping coefficient, P_e: generator output power, P_{in}: turbine output power, and P_{em}: maximum output power. $a_1 = \tilde{D}/\tilde{M}$ is the system parameter of the linear term. The measurement noise $e(t)$ is zero-mean white Gaussian noise with a standard deviation $\sigma_e = 0.0036$ (noise-to-signal ratio: 1.2%). The target system (16) changes stepwise as shown in Table 1.

TABLE 1. System parameter a_1 and nonlinear function $f(\boldsymbol{z}(t))$

t[s]	k	a_1	$f(\boldsymbol{z}(t))$
$[0, 5]$	$[0, 500]$	3.0	$-18.3(1 + u(t)) \sin x(t) + 13.3$
$(5, 20]$	$(500, 2000]$	1.0	$-16.7(1 + u(t)) \sin x(t) + 13.3$
$(20, 35]$	$(2000, 3500]$	2.0	$-15.0(1 + u(t)) \sin x(t) + 13.3$

The sampling period is taken to be $T = 0.01$ [s], and the third-order Butterworth filter with the cutoff frequency $\omega_c = 10$ [rad/s] is used as a delayed state variable filter. The initial value for the RLS method is set as $\boldsymbol{\theta}_{ml}(k_s) = \mathbf{0}$. The setting parameters of the GA and the k-means method are chosen as follows:

GA
- population size: $Q = 6$
- crossover probability: $p_c = 0.9$
- mutation probability: $p_m = 0.1$
- search range of the moving-window length: $[W_{\min}, W_{\max}] = [100, 800]$
- range of the generation number: $[G_{\min}, G_{\max}] = [1, 9]$

k-means method
- maximum iteration number: $h_{\max} = 3$

The computational burden is greatly affected by the population size Q and the range of the generation number $[G_{\min}, G_{\max}]$ of the GA, and the maximum iteration number $h_{\max}$ of the k-means method. Since it is desirable to reduce the computational burden as much as possible for on-line identification, these setting parameters are taken to be sufficiently small so as not to deteriorate the identification accuracy.

Figure 2 shows the true system parameter a_1 and the estimated system parameter $\hat{a}_1$ of the linear term. This figure indicates that the proposed method can track the time-varying system parameter well and can significantly suppress the oscillation of the estimated parameters caused by using small amounts of training data.

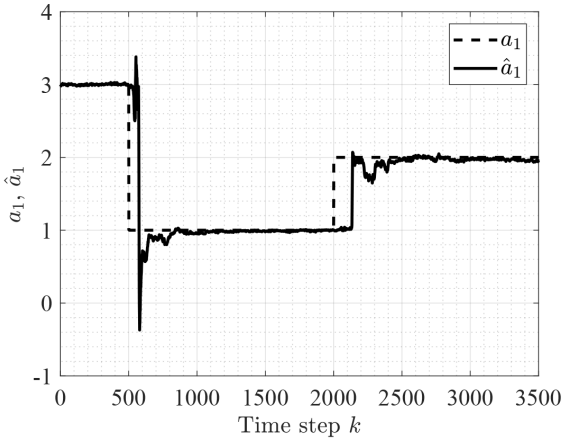
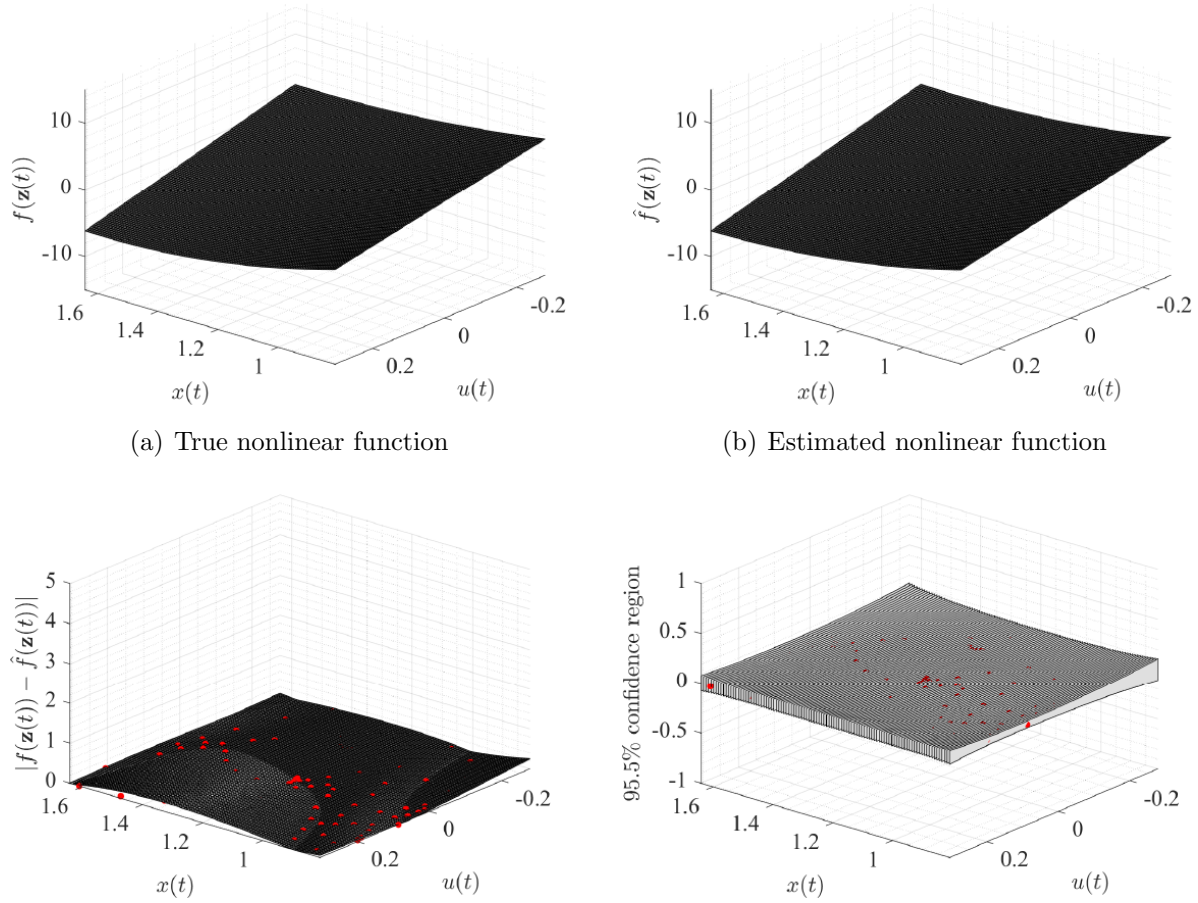


FIGURE 2. System parameter of the linear term



(a) True nonlinear function

(b) Estimated nonlinear function

(c) Absolute error between true and estimated nonlinear functions

(d) 95.5% confidence region around estimated nonlinear function

FIGURE 3. Nonlinear function

The proposed method can estimate the nonlinear function and its confidence measure at all time steps. As an example, the true nonlinear function $f(\boldsymbol{z}(t))$, the estimated nonlinear function $\hat{f}(\boldsymbol{z}(t))$, the absolute error between $f(\boldsymbol{z}(t))$ and $\hat{f}(\boldsymbol{z}(t))$, and the double standard deviation confidence interval (95.5% confidence region) around $\hat{f}(\boldsymbol{z}(t))$ at the final time step $k = 3500$ are depicted in Figure 3. The red dots in these figures are the input-output data used for identification, which are extracted by the k-means method. Clearly the estimated nonlinear function $\hat{f}(\boldsymbol{z}(t))$ is very close to the true nonlinear function $f(\boldsymbol{z}(t))$, and the confidence region of the estimated nonlinear function is very small on the data region.

The moving-window length $W(k)$ optimized by the GA at each time step is shown in Figure 4, and the generation number $g_{\max}(k)$ of the GA at each time step is depicted in Figure 5. From Figure 4, we can confirm that the moving-window length $W(k)$ decreases rapidly when the target system changes, and then gradually increases and reaches around the maximum value $W_{\max}$. On the other hand, as shown in Figure 5, the generation number $g_{\max}(k)$ of the GA increases rapidly when the target system changes, and then gradually decreases and reaches around the minimum value $G_{\min}$. Therefore, the proposed method can automatically adjust the moving-window length $W(k)$ and the generation number $g_{\max}(k)$ of the GA depending on the degree of time-varying characteristics of the target system.

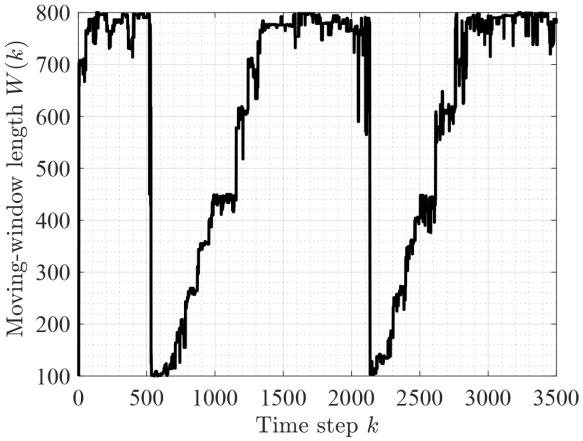


FIGURE 4. Moving-window length $W(k)$

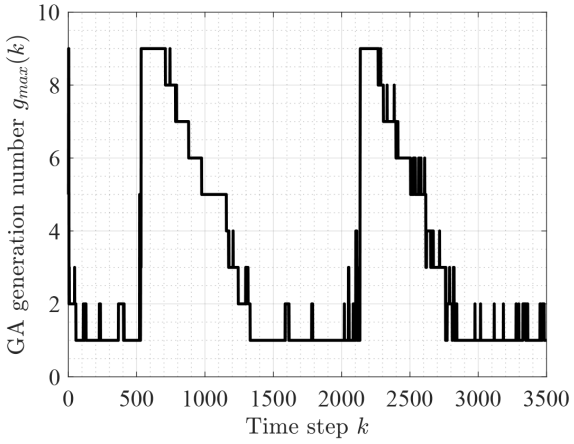


FIGURE 5. Generation number of GA

TABLE 2. The mean squares errors between $x(t)$ and $\hat{x}(t)$

	$k = 500$	$k = 2000$	$k = 3500$
Proposed method	1.15×10^{-6}	2.91×10^{-5}	1.53×10^{-5}
Conventional GP-based method	7.87×10^{-7}	9.72×10^{-6}	9.99×10^{-6}

Table 2 shows the mean squares errors (MSE) between the true output $x(t)$ and the estimated output $\hat{x}(t)$ by the proposed method and the conventional GP-based method [12] at $k = 500$, 2000 and 3500. The MSE values are calculated by

$$\frac{1}{N}\sum_{k=1}^{N}(x(kT) - \hat{x}(kT))^2 \tag{17}$$

where $N = 1500$ data of both $x(t)$ and $\hat{x}(t)$ are generated for validation. Although the proposed method has a slightly larger MSE than the conventional GP-based method, it can be seen that the proposed method can estimate the mathematical model of the target system with sufficient accuracy.

Table 3 shows the computational time per one step for the proposed method and the conventional GP-based method. The proposed method reduces the computational time of the GA to less than half of the conventional GP-based method by making the generation numbers of the GA variable, and reduces the computational time of the RLS method

TABLE 3. Computational time

	GA	RLS	Total
Proposed method	2.83×10^{-2} [s]	2.33×10^{-4} [s]	2.88×10^{-2} [s]
Conventional GP-based method	6.59×10^{-2} [s]	7.29×10^{-3} [s]	7.42×10^{-2} [s]

(CPU: Intel(R) Core(TM) i7-13700K 3.40GHz)

to about 3% of that of the conventional GP-based method by using only the training data extracted by the k-means method. The total computational time of the proposed method is reduced to about 40% of that of the conventional GP-based method. Therefore, the proposed method can improve the computational burden without deteriorating the identification accuracy.

6. **Conclusions.** In this paper, a new on-line GP model identification of continuous-time nonlinear systems has been presented. The hyperparameters of the covariance function and the moving-window length are determined by the GA, while the system parameters of the linear terms and the weighting parameters of the mean function are estimated by the RLS method. To reduce the computational burden, the maximum number of generations in the GA is adjusted according to the time changes of the target system. Moreover, only the input-output data extracted by the k-means method are utilized in both GA and RLS method. Simulation results show that the proposed method can significantly reduce the computational burden while maintaining high identification accuracy and tracking performance. The development of GP-based identification method for continuously time-varying nonlinear systems is one of the future works.

REFERENCES

[1] S. Chen, S. A. Billings and P. M. Grant, Recursive hybrid algorithm for non-linear system identification using radial basis function networks, *International Journal of Control*, vol.55, no.5, pp.1051-1070, 1992.

[2] Y. Lu, N. Sundararajan and P. Saratchandran, Identification of time-varying nonlinear systems using minimal radial basis function neural networks, *IEE-Part D, Control Theory and Applications*, vol.144, no.2, pp.202-208, 1997.

[3] Y. Wang, Y. Liu, J. Chen, S. Tang and M. Deng, Online identification of Hammerstein systems with B-spline networks, *International Journal of Adaptive Control and Signal Processing*, vol.38, no.6, pp.2074-2092, 2024.

[4] H. Xu, L. Xu and S. Shen, Online identification methods for a class of Hammerstein nonlinear systems using the adaptive particle filtering, *Chaos, Solitons and Fractals*, vol.186, 115181, pp.1-13, 2024.

[5] J. Wang, W. Xiong, F. Ding, Y. Zhou and E. Yang, Parameter estimation method for separable fractional-order Hammerstein nonlinear systems based on the on-line measurements, *Applied Mathematics and Computation*, vol.488, 129102, pp.1-17, 2025.

[6] T. Kösters, C. Illg and O. Nelles, Adaptive nonlinear system identification of separation unit for raw materials, *Control Engineering Practice*, vol.162, 106370, pp.1-16, 2025.

[7] T. Hachino and H. Takata, On-line identification method of continuous-time nonlinear systems using radial basis function network model adjusted by genetic algorithm, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol.E87-A, no.9, pp.2372-2378, 2004.

[8] T. Hachino and H. Takata, On-line identification of continuous-time nonlinear systems using radial basis function networks and immune algorithm, *Proc. of International Conference on Control and Automation*, pp.587-592, 2005.

[9] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Massachusetts, 1989.

[10] K. Sugiyama and T. Hachino, On-line identification of continuous-time nonlinear systems using Gaussian process models, *Proc. of 2018 Joint 10th International Conference on Soft Computing and Intelligent Systems and 19th International Symposium on Advanced Intelligent Systems*, pp.1219-1224, 2018.
[11] T. Hachino, On-line Gaussian process model identification of nonlinear systems using particle swarm optimization, *International Journal of Innovative Computing, Information and Control*, vol.17, no.6, pp.2191-2200, 2021.
[12] D. Yamasaki and T. Hachino, On-line identification of time-varying nonlinear systems based on Gaussian process model using k-means method, *Proc. of the International Council on Electrical Engineering Conference 2024*, O-025, pp.1-6, 2024.
[13] A. O'Hagan, Curve fitting and optimal design for prediction (with discussion), *Journal of the Royal Statistical Society B*, vol.40, pp.1-42, 1978.
[14] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, 2006.
[15] J. Kocijan, A. Girard, B. Banko and R. Murray-Smith, Dynamic systems identification with Gaussian processes, *Mathematical and Computer Modelling of Dynamical Systems*, vol.11, no.4, pp.411-424, 2005.
[16] M. F. Huber, Recursive Gaussian process, on-line regression and learning, *Pattern Recognition Letters*, vol.45, pp.85-91, 2014.
[17] K. Seo and M. Yamakita, Nonlinear time-varying system identification with recursive Gaussian process, *Proc. of 2017 American Control Conference*, pp.825-830, 2017.
[18] S. Tang, K. Fujimoto and I. Maruta, Learning dynamic systems using Gaussian process regression with analytic ordinary differential equations as prior information, *IEICE Transactions on Information and Systems*, vol.E104-D, no.9, pp.1440-1449, 2021.
[19] J. MacQueen, Some methods for classification and analysis of multivariate observations, *Proc. of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp.281-297, 1967.
[20] H. Takata, An automatic choosing control for nonlinear systems, *Proc. of the 35th IEEE Conference on Decision and Control*, pp.3453-3458, 1996.

Author Biography

Tomohiro Hachino received the B.E., M.E. and Dr. Eng. degrees in Electrical Engineering from Kyushu Institute of Technology, Japan, in 1991, 1993, and 1996, respectively.

Dr. Hachino is currently a professor at the Graduate School of Science and Engineering, Kagoshima University, Japan. His research interests include nonlinear control and identification, signal processing, and evolutionary computation.

Daijiro Yamasaki received the B.E. and M.E degrees in Electrical and Electronics Engineering from Kagoshima University, Japan, in 2023 and 2025, respectively. His research interests include nonlinear system identification and evolutionary computation.