Mathematical Biosciences and Engineering

MBE, 19(12): 12617–12631.
DOI: 10.3934/mbe.2022589
Received: 13 June 2022
Revised: 22 August 2022
Accepted: 22 August 2022
Published: 29 August 2022

http://www.aimspress.com/journal/MBE

Research article

A fault diagnostic approach based on PSO-HMM for underwater thrusters

Zhenzhong Chu[1,2,*], Zhenhao Gu[1], Zhiqiang Li[1], Yunsai Chen[3] and Mingjun Zhang[4]

[1] Logistics Engineering College, Shanghai Maritime University, Shanghai 201306, China
[2] School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China
[3] College of Shipbuilding Engineering, Harbin Engineering University, Harbin 150001, China
[4] College of Mechanical and Electrical Engineering, Harbin Engineering University, Harbin 150001, China

* **Correspondence:** Email: chu_zhenzhong@163.com.

Abstract: In this paper, we describe an approach based on improved Hidden Markov Model (HMM) for fault diagnosis of underwater thrusters in complex marine environments. First, considering the characteristics of thruster data, we design a three-step data preprocessing method. Then, we propose a fault classification method based on HMMs trained by Particle Swarm Optimization (PSO) for better performance than methods based on vanilla HMMs. Lastly, we verify the effectiveness of the proposed approach using thruster samples collected from a fault emulation experimental platform. The experiments show that the PSO-based training method for HMM improves the accuracy of thruster fault diagnosis by 17.5% compared with vanilla HMMs, proving the effectiveness of the method.

Keywords: underwater vehicle; thruster; fault diagnosis; Hidden Markov Model; particle swarm optimization

1. Introduction

The ocean is the frontier of major scientific researches such as the origin of life, the earth's evolution, and climate change. It is also the most important trade channel under the background of economic globalization [1]. Underwater vehicles play an indispensable role as the key equipment for

ocean exploration and development [2–4]. Thrusters are important propulsive devices of an underwater vehicle, working in high-pressure and high-load environments for a long time. Once one or more thrusters of an underwater vehicle fail, the smooth execution of underwater tasks will not be guaranteed. It may even be impossible to recover the underwater vehicle, which will result in a heavy loss. Therefore, it is of urgent need and great significance to carry out research on thruster fault diagnosis to enhance the safety and reliability of underwater thrusters [5–8].

Data-driven fault diagnostic methods have been widely used because of the difficulty in establishing dynamic models of underwater vehicles and their thrusters [9,10]. They can detect faults by directly extracting fault features from observed data, without the mathematical model of the object to be diagnosed. Zhu et al. introduced a deep belief network into multisensory information fusion model to identify uncertain and continuously changing fault patterns of deep-sea human occupied vehicle thruster [11]. Wang et al. improved the performance of support vector domain description model for fault diagnosis of AUV thrusters by describing the distribution forms and rules of the mapping data in a high dimensional feature space and optimizing the parameters of kernel function [12]. Chu et al. proposed a data argumentation method based on random forest regression for fault samples to solve the problem of low classification accuracy caused by extremely imbalanced datasets collected from actual underwater environments [13]. More researches which have been developed or methods which may be applied on underwater thrusters can be obtained in [14–16]. However, the time dependency and cross-correlation in observed data of actual thrusters under different operating conditions are not fully considered because most of the existing data-driven methods only focus on current or voltage at a fixed rotational speed. Additionally, output signals of a thruster contain nonlinear dynamic disturbances caused by operating environments and load changes, etc. It is difficult to rapidly distinguish whether the changes in output signals are caused by thruster faults or disturbances outside, so that the occurrence time of faults cannot be precisely estimated.

The failure process of a thruster generally includes the following stages: normal operation, performance degradation, abnormal output and complete failure. Each stage of the failure process may also contain several unobservable ("hidden") states because of the high dynamicity of the thruster system and its data. HMM is able to establish the relationship between the thruster data and hidden states by extracting intrinsic information from the data. It can model the failure process of the thruster, and is suitable for fault diagnosis of underwater thrusters [17]. Zhang et al. extracted fault feature set of mixed domains from bearing vibration signal by principal component analysis, and trained HMMs on part of the reduced feature set [18]. Soleimani et al. presented a methodology for fault detection, prediction and isolation of automotive exhaust gas aftertreatment system based on the integration of HMM and Bayesian networks [19]. Arpaia et al. proposed a fault detection method exploiting HMMs for fluid machinery without adequate a-priori information about faulty conditions [20]. However, the local optima found by manually setting initial model parameters are not always good enough in the training of traditional HMMs.

In this paper, we explore the HMM for fault diagnosis of underwater thrusters. Overall, our contributions are as follows:

1) We design a three-step data preprocessing method, including smoothing output signals in thruster data by Savitzky-Golay filters, aligning each control signal and its output signal via correlation-based latency estimation, and vector-quantizing the data through *k*-means algorithm.

2) We propose a fault classification method based on HMMs trained by Particle Swarm Optimization (PSO). The HMMs trained by PSO have better parameter estimations and classification performance

than vanilla HMMs.
3) We build a fault emulation experimental platform for fault diagnosis of underwater thrusters and verify the effectiveness of the data preprocessing method and fault classification method using the samples collected from the experiments.

2. Methods

In Section 2, we introduce the theoretical basis of HMM-based thruster fault diagnosis, design the process of preprocessing thruster data, propose the theory of training HMM by PSO, and then describe the framework of the PSO-HMM-based fault diagnostic approach.

2.1. Fault diagnostic method based on HMM

HMM describes a Markov process with hidden states. It is suitable for the modeling of dynamic system and can be used for pattern classification. An HMM can be described as:

$$\lambda = (\pi, A, B), \tag{1}$$

where π is the initial state probability matrix, A is the state transition probability matrix, B denotes a set of emission probability distributions.

In Eq (1), B may be a probability matrix or a set of probability density functions, which depends on whether the observations are discrete (Discrete HMM, DHMM) or continuous (Continuous HMM, CHMM). It is generally assumed that the observations of a CHMM follow multiple Gaussian mixture distributions, that is, each hidden state represents a Gaussian mixture model (GMM).

Both types of HMMs can be used to model continuous signals. The signals need to be transformed into discrete observations through vector quantization for modeling with DHMMs, while they can be directly used as observations for modeling with CHMMs. However, the amount of computation of GMM is much larger than that of vector quantization, which will increase the training time and real-time prediction latency. Therefore, DHMMs are used in the proposed fault classification method for lower computational cost and time requirement.

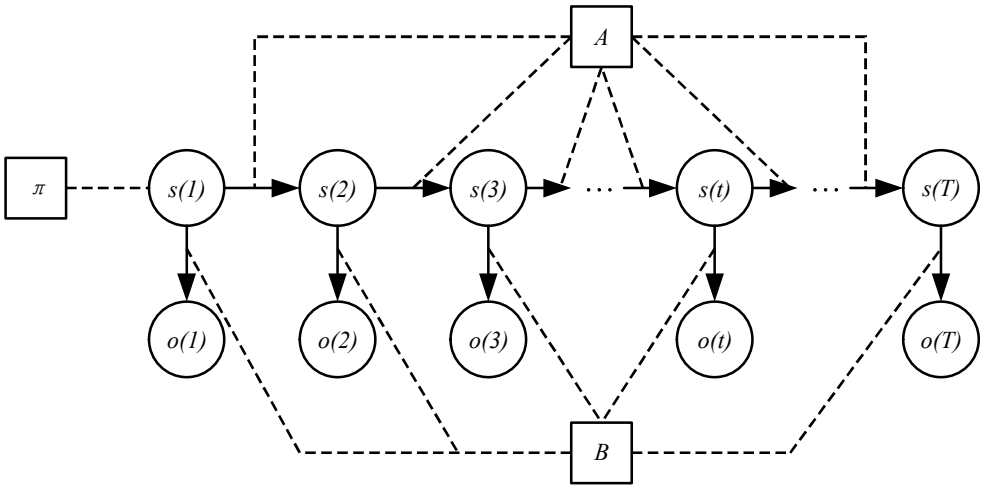

Figure 1. General architecture of an HMM.

Figure 1 shows the general architecture of an HMM. The random variable $s(t)$ denotes the hidden state of thruster at time t. The state of the thruster at initial time depends on the matrix π, and the transition probability from each state to another is determined by the matrix A. The random variable $o(t)$ denotes the observed thruster data at time t. The distribution of the variable $o(t)$, one of the distributions in B, is only governed by the value of the hidden variable $s(t)$.

2.2. Data preprocessing

The preprocessing method of thruster data can be divided into three steps. First, the output signals in thruster data are smoothed by Savitzky-Golay filters to reduce the impact of underwater environmental noises on signal precision. Second, each control signal and its output signal are aligned via latency estimation based on correlation to alleviate dynamic latency between them. Third, the processed data are vector-quantized through *k*-means algorithm to meet the modeling requirement of DHMM.

2.2.1. Signal smoothing

Output signals in actual data of underwater thrusters usually contain a lot of environmental noises. These noises such as sensor errors may seriously affect the accuracy of data-driven diagnostic methods. Therefore, it is necessary to smooth the output signals first. A Savitzky-Golay filter is a convolution filter that smooths data by fitting successive sub-sets of adjacent data points with a low-degree polynomial via the method of linear least squares [21]. The outliers that deviate from the normal trend are pulled back to a certain extent during the fitting process. Suppose that the length of the signal to be smoothed is N, the size W of the sub-set is $2M + 1$, and the degree O of the polynomial is P. The smoothed signal is obtained by:

$$y[n] = \sum_{m=-M}^{M} h[m]x[n-m],\ n = 0,1,2,\ldots,N-1, \tag{2}$$

where $x[n-m]$ is the $(n-m)^{th}$ sampling point, $y[n]$ is the n^{th} smoothed output value, and $h[m]$ is the m^{th} convolution coefficient which is given by:

$$h[m] = h[-m] = \sum_{k=0}^{P} a_k m^k,\ -M \le m \le M,\ P <= 2M, \tag{3}$$

where a_k is the k^{th} polynomial coefficient which is given by:

$$\boldsymbol{a} = (a_0, a_1, \ldots, a_P)^T = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{d}, \tag{4}$$

Where $\boldsymbol{d} = (0,0,\ldots,0,1,0,\ldots,0,0)^T$ is a column vector with the size of ($2M$ + 1, 1), and $\boldsymbol{A^T}$ is a matrix with the size of (P + 1, $2M$ + 1):

$$\boldsymbol{A}^T = \begin{pmatrix} (-M)^0 & \ldots & (-1)^0 & 1 & 1^0 & \ldots & M^0 \\ (-M)^1 & \ldots & (-1)^1 & 0 & 1^1 & \ldots & M^1 \\ (-M)^2 & \ldots & (-1)^2 & 0 & 1^2 & \ldots & M^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ (-M)^P & \ldots & (-1)^P & 0 & 1^P & \ldots & M^P \end{pmatrix}. \tag{5}$$

It can be seen that the values of convolution coefficients depend only on M and P and is independent of the input samples, that is, the same coefficients will be obtained when the size W and

the degree O are fixed. Moreover, both the extent of noise reduction and signal distortion increase as M increases, while they decrease as P increases [22]. Therefore, the size W and the degree O should be carefully chosen to achieve a balance between noise reduction and signal distortion.

2.2.2. Latency estimation

There is a dynamic latency between each control signal and its output signal in thruster data, which will affect the results of on-line data-driven fault diagnosis of underwater thrusters. The output signal in thruster data is composed of rotational speed and current. The latency between the control signal and rotational speed signal is usually large, while the latency between the control signal and current signal is relatively small. We adopt a kind of correlation-based method for latency estimation [23]. The method is described as follows:

1) Assume that the latency between the control signal and rotational speed (or current) signal of a certain sample is t (a non-negative integer) sampling periods.

2) Shift the rotational speed (or current) signal by the assumed number of periods t.

3) Calculate the correlation coefficient $r(U,W)$ (or $r(U,I)$) between the control signal U and aligned rotational speed (or current) signal W (or I) according to Eq (7) (or Eq (8)).

$$r(U,W) = \frac{\sum_{n=0}^{N-1}[(u_n-\bar{u})(w_n-\bar{w})]}{\sqrt{\sum_{n=0}^{N-1}(u_n-\bar{u})^2 \sum_{n=0}^{N-1}(w_n-\bar{w})^2}}, \tag{6}$$

$$r(U,I) = \frac{\sum_{n=0}^{N-1}[(u_n-\bar{u})(i_n-\bar{\imath})]}{\sqrt{\sum_{n=0}^{N-1}(u_n-\bar{u})^2 \sum_{n=0}^{N-1}(i_n-\bar{\imath})^2}}, \tag{7}$$

where $\bar{u}$, $\bar{w}$ and $\bar{\imath}$ is arithmetic mean of control signal, rotation speed signal and current signal, respectively.

1) Repeat the above three steps with different value of t until the maximum correlation coefficient is found.

2) Consider the number of periods t corresponding to the maximum correlation coefficient as the estimate of the latency between the control signal and rotational speed (or current) signal.

Each sample is processed by the above steps to alleviate the latency between the control signal and output signal.

2.2.3. Vector quantization

The data of underwater thrusters belong to continuous signals. They must be vector-quantized to meet the modeling requirement of DHMM. Vector quantization is a kind of data compression technique with high efficiency and large compression ratio. *K*-means clustering is one of the classical vector quantization algorithms.

K-means algorithm is a simple clustering method based on distance with fast convergence speed in most cases. It partitions sampling points into k clusters by measuring distance between sampling point and cluster center. The algorithm is as follows: First, k sampling points are randomly selected from all sampling points as cluster centers. All sampling points are then partitioned into the cluster in which the distance between the sampling point and the cluster center calculated by Eq (9) is the

smallest. After partition, new center of each cluster is calculated and selected. The above two steps are repeated iteratively until a desired level of convergence. Finally, the vector quantized data are generated by representing the sampling points in each cluster via the symbol of the cluster.

$$D = \sum_{i=0}^{N-1} \left\| y_i - u_j \right\|^2, \ j = 0,1,2,\ldots,k-1, \tag{8}$$

where y_i denotes the i^{th} sampling point, u_j denotes the j^{th} cluster center.

2.3. PSO-HMM

There are forward/backward algorithm, Viterbi algorithm and Baum-Welch algorithm for the evaluation task, decoding task and learning task in HMMs, respectively. The learning task in HMMs is a problem of parameter estimation. Assuming that there are Q kinds of hidden states and Z kinds of observations in the HMM, the goal is to find the maximum likelihood estimate of the parameters $\lambda = (\pi, A, B)$ of the HMM given the observed sequences O, which is described as:

$$\lambda = \arg\max_{\lambda} \log P\left(O|\lambda\right) \quad s.t. \quad \begin{cases} \sum_{i=0}^{Q-1} \pi_i = 1 \\ \sum_{j=0}^{Q-1} \alpha_{ij} = 1, \quad 0 \le i \le Q-1, \\ \sum_{i=0}^{Z-1} \beta_{ij} = 1, \quad 0 \le i \le Q-1 \end{cases} \tag{9}$$

Where α_{ij} is one of the elements in the state transition probability matrix A, and β_{ij} is one of the elements in the emission probability matrix B.

Baum-Welch algorithm is usually used to solve the learning task to a certain extent. It is a special case of the expectation-maximization algorithm. Baum-Welch algorithm generally finds a local optimum because of the manual setting of initial model parameters, although it does guarantee the monotonic increasing of the likelihood. In fact, the parameter learning task in HMMs can be regarded as a general optimization problem with constraints. The objective function to be optimized is $log\ P\ (O|\lambda)$. PSO, an optimization algorithm based on swarm cooperation, is able to effectively reduce the gap between the found solution and the global optimal solution. It is easy to program the algorithm and implement parallel processing [24]. Therefore, we introduce PSO into HMM for better parameter estimation.

First, the constraints of PSO are defined by the equality constraints in Eq (9). Second, n sets of parameters of HMM subject to the constraints are randomly initialized as the particles with their own positions and velocities. The fitness value of each particle represents the log probability that the sequences O are observed given the set of parameters. Third, the current global best-known solution of the whole swarm and current local best-known solution of each particle are determined by calculating fitness values. Then, the position and velocity of each particle is updated via Eq (11). The above two steps are repeated iteratively until a certain level of convergence. Lastly, the particle with the maximum fitness value is considered as the optimal solution of the parameter learning task.

$$\begin{cases} v_i^{k+1} = wv_i^k + c_1 r_1 \left(p_{best\ i}^k - x_i^k\right) + c_2 r_2 \left(g_{best}^k - x_i^k\right) \\ x_i^{k+1} = x_i^k + v_i^{k+1} \end{cases}, \tag{10}$$

where x_i^k is the position of the i^{th} particle during the k^{th} iteration, v_i^k is the velocity of the i^{th} particle during the k^{th} iteration, $p_{best\ i}^k$ is the current local best-known position of the i^{th} particle after k iterations, g_{best}^k is the current global best-known position of the whole swarm after k iterations, c_1 denotes the maximum step size for particles to move towards $p_{best\ i}^k$, c_2 denotes the maximum step size for particles to move towards g_{best}^k, r_1 and r_2 are random numbers in the interval [0.0, 1.0], w denotes the ability of particles to retain their original velocities.

2.4. Framework of the approach

Figure 2 shows the framework of the thruster fault diagnostic approach built in this study. The whole process of the fault diagnosis is as follows:

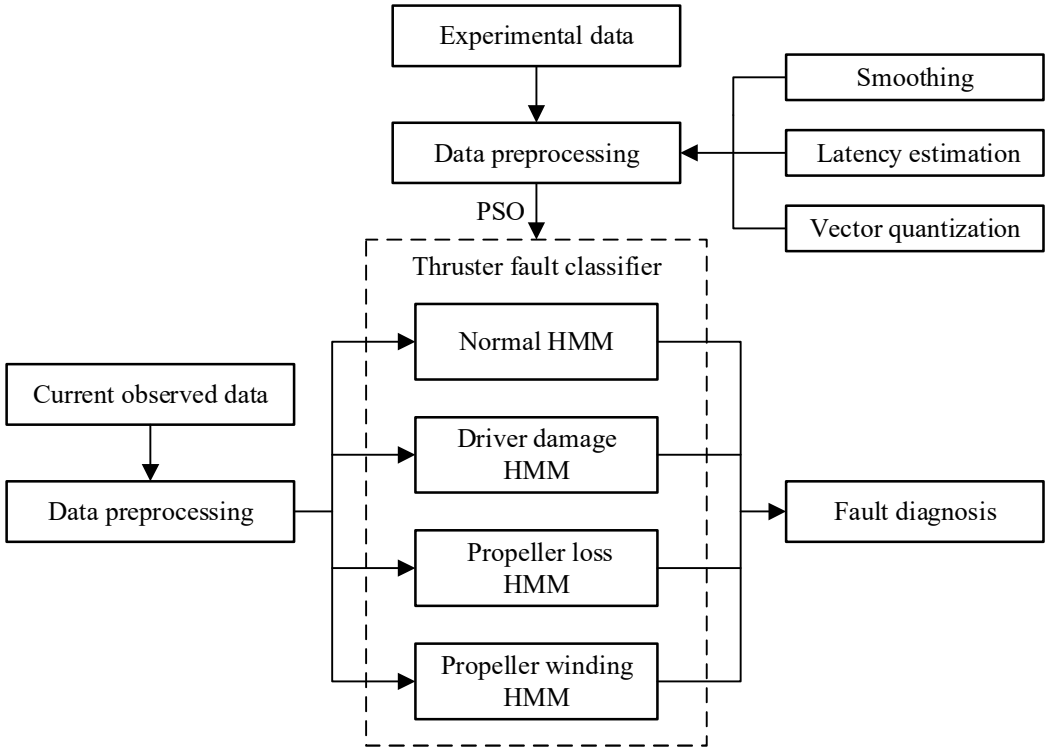

Figure 2. Framework of the PSO-HMM-based fault diagnostic approach.

1) A fault emulation experimental platform is built. The experimental samples of the thruster under normal state and three kind of fault states: driver damage, propeller loss and propeller winding, are obtained from fault emulation experiments.
2) The samples are preprocessed by three main steps. First, they are smoothed by Savitzky-Golay filters and segmented into relatively short samples through sliding window. Then, the samples are aligned via latency estimation. Lastly, they are scaled by min-max normalization and vector quantized by *k*-means algorithm.
3) Four HMMs are trained by PSO on the preprocessed samples of normal state, driver damage state,

propeller loss state and propeller winding state, respectively. These four HMMs comprise the whole thruster fault classifier.

4) The current observed thruster data are input into the thruster fault classifier after the same preprocessing. The output probabilities of the data under the four HMMs are calculated by forward algorithm. The state corresponding to the HMM with the maximum output probability is taken as the current state of the thruster.

The thruster fault diagnosis is implemented through the above process.

3. Experiments

In Section 3, we introduce the details of thruster fault emulation experiments, show the effect of the designed data preprocessing method, and analyze the results of the PSO-HMM-based fault diagnosis. We also discuss the advantages of the proposed approach by comparing with some other methods.

3.1. Fault emulation experiments of underwater thruster

In order to test the classification performance of the PSO-HMM-based thruster fault diagnostic approach, we choose a thruster commonly used in the industry as the experimental object. Table 1 shows the specifications of the thruster.

Table 1. Specifications of the thruster.

Parameters	Values
Product model	T561–150
Rated voltage	150 VDC
Rated speed	2600 rpm
Rated power	1100 W
Control signal	-5–5 V

Figure 3 shows the structure of the experimental platform. The platform is used to collect the data of the thruster under normal state and three kind of fault states. During one experiment, the thruster is arranged on a fixed support which is in a pool. The PC starts sending control signal to the thruster through the STM32 after the system is powered on. The thruster starts working according to control signal. Meanwhile, output signal consisting of rotational speed signal and current signal is collected by the STM32 and current sensor, and stored by the PC. We collected the experimental samples of the four kinds of thruster states in order by emulating normal operation and different faults over the thruster.

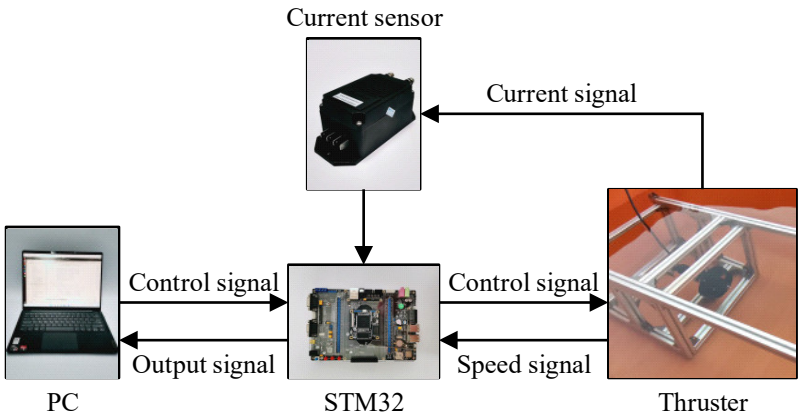

Figure 3. Structure of the experimental system.

3.2. Results of data preprocessing

As described in Section 2.2, we firstly performed smoothing over the output signals in the samples via Savitzky-Golay filters. The degree O was set to 3. The size W was set to 9 and 19 for rotational speed signal and current signal, respectively. Take the sampling points 1–2000 of the thruster under normal state as an example, the results of smoothing the rotational speed signal and current signal are shown in Figure 4. We observed that the noise in the smoothed signal is significantly less than that in the original signal and the signal distortion is not very serious. It shows the effectiveness of this method.

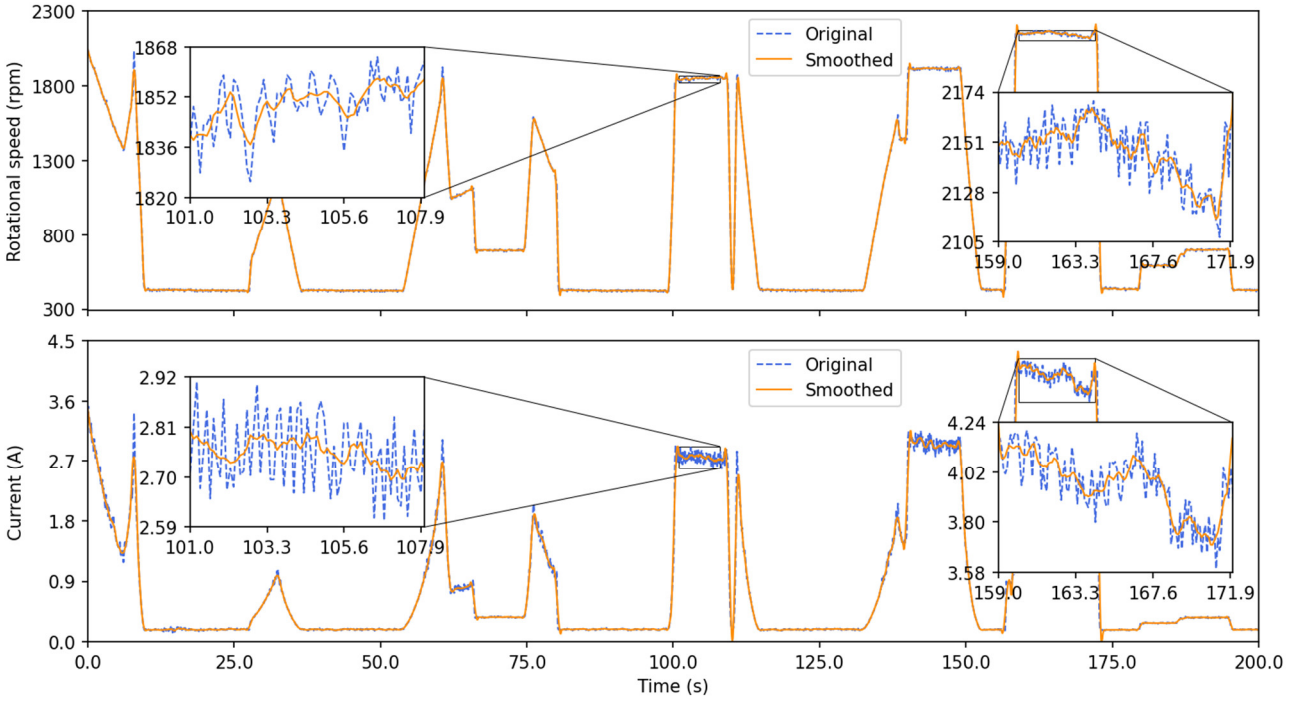

Figure 4. Results of smoothing the output signal.

After smoothing the signals, we segmented the samples into relatively short samples through sliding window. Each short sample consists of 320 three-dimensional sampling points. The three dimensions denote control signal, rotational speed signal, and current signal. Thus, the size of each sample is 320 × 3. There are 100 samples for each thruster state, i.e., a total of 400 samples. We randomly selected 40 samples of each thruster state as test set and used all remaining samples for training data, i.e., a total of 240 training samples and 160 test samples.

Then, we alleviated the time delay between each control signal and its output signal via the correlation-based latency estimation. Take the sampling points 1–320 of normal state as an example, the correlation coefficients between the control signal and rotational speed (or current) signal with assumed latency from 0 to 15 sampling periods are shown in Figure 5. It can be seen that the number of periods t corresponding to the maximum correlation coefficient between the control signal and rotational speed (or current) signal is 2 (or 1). Thus, we shifted the rotational speed signal and current signal by 2 and 1 periods, respectively, to align the data.

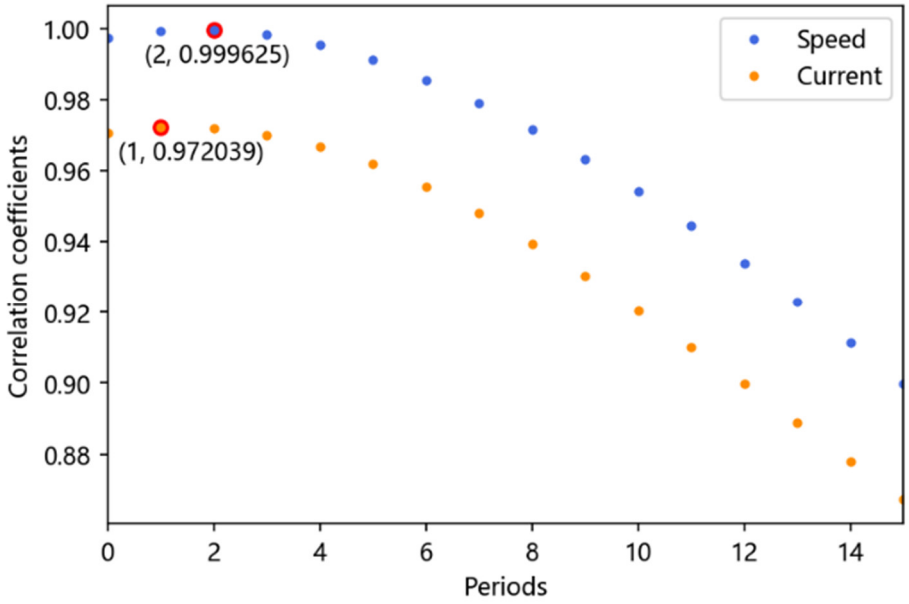

Figure 5. Correlation coefficients with different assumed latencies.

Lastly, we performed min-max normalization and vector quantization over the samples. The range of each feature was scaled into the interval [0.0, 1.0]. The number of clusters k was set to 8. Figure 6 shows the 3D view of the sampling points and clustering result. Each color in Figure 6(A) represents a kind of thruster state, while each color in Figure 6(B) represents a cluster. All the sampling points are partitioned into eight clusters. We then generated the vector quantized samples by representing the sampling points in each cluster via the symbol of the cluster. For example, a normalized sampling point (Control: 0.385, Speed: 0.391, Current: 0.126) which belongs to Cluster 4 is transformed into a scalar 4, while a normalized sampling point (Control: 0.869, Speed: 0.631, Current: 0.668) which belongs to Cluster 6 is transformed into a scalar 6.

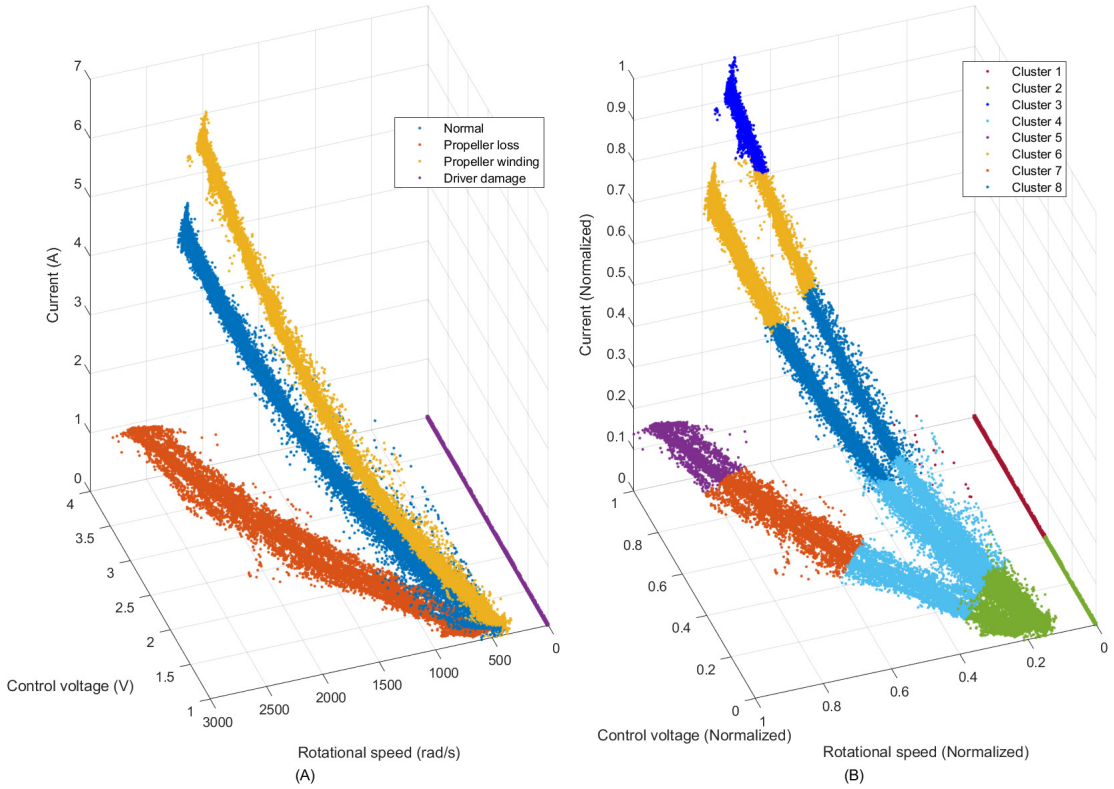

Figure 6. 3D view of the sampling points and result of *k*-means clustering.

3.3. Results of fault diagnosis

We trained four HMMs, including HMM of normal state, HMM of driver damage state, HMM of propeller loss state, and HMM of propeller winding state, on the preprocessed training samples of the four thruster states by the proposed PSO-HMM algorithm. The number of the states in HMMs was set to 3. The parameters of PSO are shown in Table 2. We set the values of c_1 and c_2 by experience and tuned the size of the swarm and w by grid search. The explored grid is the cross-product of the size values ranging in [100, 200, 300, …, 1000] and *w* values in [0.3, 0.4, 0.5, …, 1.5]. We chose the values displayed in Table 2, considering both performance and computational cost. The iterative processes of the training are shown in Figure 7. All the training processes reach the convergence in about 50 steps.

Table 2. Parameters of PSO.

Parameters	Value
Maximum iterations	100
Size of the swarm	500
c_1	2
c_2	2
w	0.6
x_{min}, x_{max}	0, 1
v_{min}, v_{max}	-0.05, 0.05

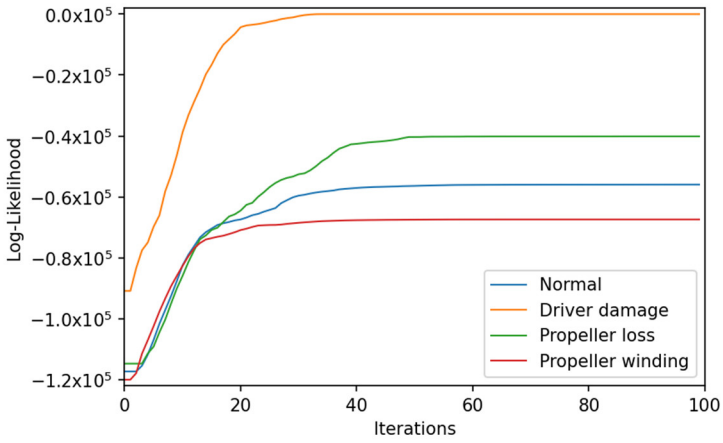

Figure 7. Training processes of the four PSO-HMMs.

We input the test samples into the four HMMs for evaluating the performance of the PSO-HMM-based fault classifier. Figure 8 and Table 3 show the results of the fault diagnosis. Note that we transformed negative infinity into -1×10^4 to plot all the log probability values into the figure. It can be seen that the samples of driver damage fault and propeller loss fault are completely distinguished from the others, while some of the samples of propeller winding fault are diagnosed as normal samples. We conjecture that the confusion is mainly because the propeller winding fault emulated in Section 3.1 is not serious and the fault features are not obvious when the control voltage is not high enough. We also observed that a few of the normal samples are misclassified. The reason may be the loss of information due to the high compression level of the vector quantization.

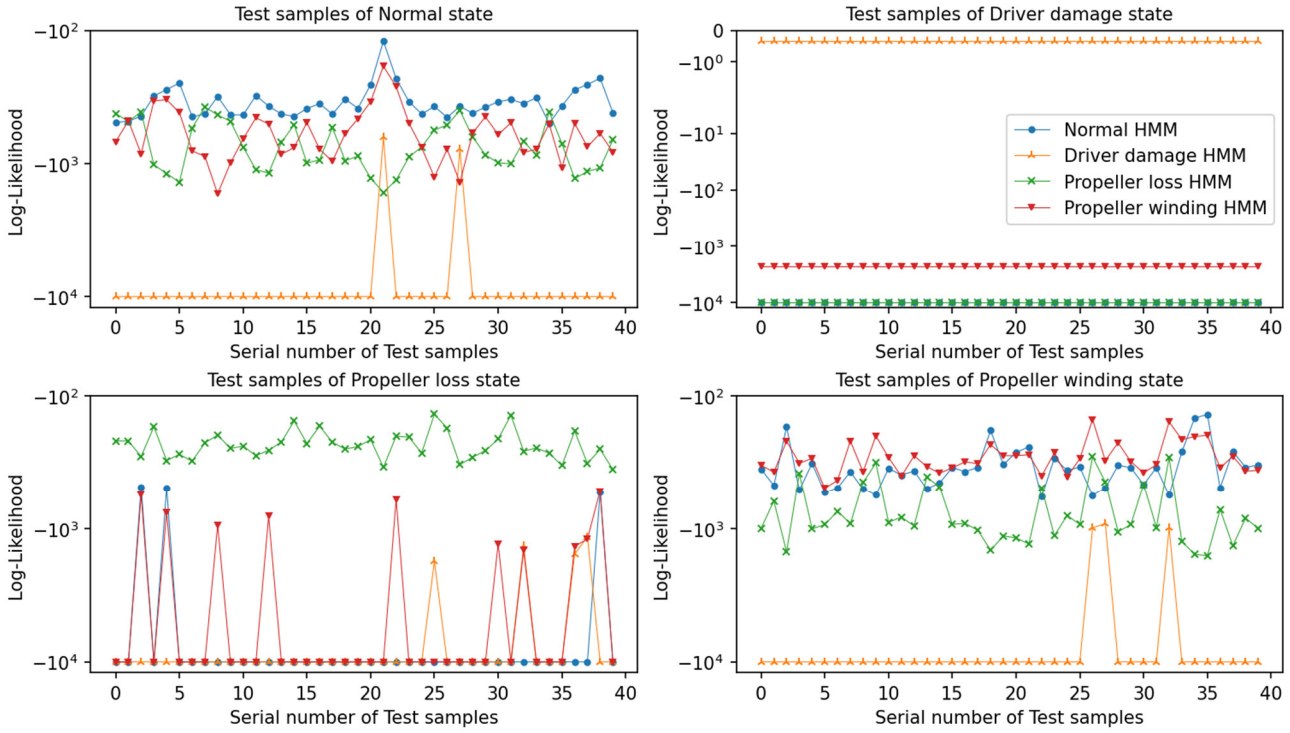

Figure 8. Results of the samples of each thruster state.

Table 3. Confusion matrix and accuracy of the fault classification.

Real \ Pred	Normal	Driver damage	Propeller loss	Propeller winding	Accuracy
Normal	35	0	4	1	87.5%
Driver damage	0	40	0	0	100%
Propeller loss	0	0	40	0	100%
Propeller winding	12	0	0	28	70%

Table 4. Comparison of the accuracy of different fault classifiers.

Class of test samples	Quantity	Accuracy		
		tsfresh-SVM	vanilla HMM	PSO-HMM
Normal	40	95%	45%	87.5%
Driver damage	40	100%	95%	100%
Propeller loss	40	100%	100%	100%
Propeller winding	40	37.5%	47.5%	70%
All	160	83.125%	71.875%	89.375%

In order to prove the advantages of the proposed approach, we trained a tsfresh-SVM (part of the RFR-SVM that is proposed in [13]) and a fault classifier consisting of four vanilla HMMs for comparison. The results are shown in Table 4. It can be seen that the accuracy of the PSO-HMM is higher than that of the vanilla HMM on each thruster state except propeller loss state, showing the effectiveness of training parameters of HMM via PSO. Although the accuracy of the tsfresh-SVM on normal state is little higher than that of the PSO-HMM, the performance of the PSO-HMM on propeller winding state is much better than that of the tsfresh-SVM. Moreover, the total accuracy of the PSO-HMM on the test samples is 89.375%, further illustrating the effectiveness of the proposed diagnostic approach for driver damage fault, propeller loss fault, and propeller winding fault.

4. Conclusions

In this paper, we provided a fault diagnostic approach based on PSO-HMM for underwater thrusters. Four kinds of thruster states, including normal state, driver damage state, propeller loss state and propeller winding state, can be effectively classified using the approach. The experiments show that the PSO-based parameter learning method for HMM improves the accuracy of thruster fault diagnosis by 17.5% compared with the classical method, proving the effectiveness of the method. Furthermore, the proposed data preprocessing method can be a reference for fault diagnosis of underwater thrusters in future because of its usefulness to thruster data for noise reduction and latency alleviation. In the next step, we will pay more attention on increasing the accuracy of classification on a highly imbalanced thruster dataset and the authenticity of fault emulation experiments so that we can finally apply fault classification on real thruster datasets.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 51839004, No. 52171313, and No. 51909039), the High tech ship innovation project (CY04N20) and the Key

Laboratory Fund Project 6142215200305.

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. C. Knight, S. McGarry, J. Hayward, P. Osman, S. Behrens, A review of ocean energy converters, with an Australian focus, *AIMS Energy*, **2** (2014), 295–320. https://doi.org/10.3934/energy.2014.3.295
2. C. Wu, Y. Dai, L. Shan, Z. Zhu, Z. Wu, Data-driven trajectory tracking control for autonomous underwater vehicle based on iterative extended state observer, *Math. Biosci. Eng.*, **19** (2022), 3036–3055. https://doi.org/10.3934/mbe.2022140
3. Z. Chu, F. Wang, T. Lei, C. Luo, Path planning based on deep reinforcement learning for autonomous underwater vehicles under ocean current disturbance, *IEEE Trans. Intell. Veh.*, **2022** (2022). https://doi.org/10.1109/TIV.2022.3153352
4. X. Li, Y. Song, J. Guo, C. Feng, G. Li, T. Yan, et al., Sensor fault diagnosis of autonomous underwater vehicle based on extreme learning machine, in *2017 IEEE Underwater Technology (UT)*, (2017), 1–5. https://doi.org/10.1109/UT.2017.7890303
5. Y. Chen, Z. Chu, K. Liu, L. Yang, D. Zhu, Research progress on thruster fault diagnosis technology for deep-sea underwater vehicle, *J. Propul. Technol.*, **41** (2020), 2465–2474. https://doi.org/10.13675/j.cnki.tjjs.200274
6. S. Nascimento, M. Valdenegro-Toro, Modeling and soft-fault diagnosis of underwater thrusters with recurrent neural networks, *IFAC-PapersOnLine*, **51** (2018), 80–85. https://doi.org/10.1016/j.ifacol.2018.09.473
7. Z. Chu, F. Meng, D. Zhu, C. Luo, Fault reconstruction using a terminal sliding mode observer for a class of second-order MIMO uncertain nonlinear systems, *ISA Trans.*, **97** (2020), 67–75. https://doi.org/10.1016/j.isatra.2019.07.024
8. A. Shumsky, A. Zhirabok, C. Hajiyev, Observer based fault diagnosis in thrusters of autonomous underwater vehicle, in *2010 Conference on Control and Fault-Tolerant Systems (SysTol)*, (2010), 11–16. https://doi.org/10.1109/SYSTOL.2010.5676076
9. M. Kordestani, M. Saif, M. E. Orchard, R. Razavi-Far, K. Khorasani, Failure prognosis and applications—A survey of recent literature, *IEEE Trans. Reliab.*, **70** (2021), 728–748. https://doi.org/10.1109/TR.2019.2930195
10. K. Zhong, M. Han, B. Han, Data-driven based fault prognosis for industrial systems: a concise overview, *IEEE/CAA J. Autom. Sin.*, **7** (2020), 330–345. https://doi.org/10.1109/JAS.2019.1911804
11. D. Zhu, X. Cheng, L. Yang, Y. Chen, S. X. Yang, Information fusion fault diagnosis method for deep-sea human occupied vehicle thruster based on deep belief network, *IEEE Trans. Cybern.*, **52** (2022), 9414–9427. https://doi.org/10.1109/TCYB.2021.3055770
12. Y. Wang, W. Zhang, F. Di, W. Gong, An AUV thruster fault diagnosis method based on the improved SVDD, in *2018 IEEE 8th International Conference on Underwater System Technology: Theory and Applications (USYS)*, (2018), 1–5. https://doi.org/10.1109/USYS.2018.8778887

13. Z. Chu, Z. Li, Z. Gu, Y. Chen, M. Zhang, A fault diagnosis method for underwater thruster based on RFR-SVM, *Proc. Inst. Mech. Eng., Part M: J. Eng. Marit. Environ.*, **2022** (2022). https://doi.org/10.1177/14750902221095423
14. J. He, Y. Li, J. Cao, Y. Li, Y. Jiang, L. An, An improved particle filter propeller fault prediction method based on grey prediction for underwater vehicles, *Trans. Inst. Meas. Control*, **42** (2020), 1946–1959. https://doi.org/10.1177/0142331219901202
15. V. Filaretov, A. Zuev, A. Zhirabok, Development of fault detection and identification system for thrusters of underwater robots, in *2019 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, (2019), 1–6. https://doi.org/10.1109/FarEastCon.2019.8934386
16. H. R. Karimi, Y. Lu, Guidance and control methodologies for marine vehicles: a survey, *Control Eng. Pract.*, **111** (2021), 104785. https://doi.org/10.1016/j.conengprac.2021.104785
17. L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, *Proc. IEEE*, **77** (1989), 257–286. https://doi.org/10.1109/5.18626
18. Q. Xu, Z. Liu, H. Zhao, Method of turnout fault diagnosis based on hidden Markov model, *J. China Railw. Soc.*, **40** (2018), 98–106. https://doi.org/10.3969/j.issn.1001-8360.2018.08.013
19. M. Soleimani, F. Campean, D. Neagu, Integration of Hidden Markov Modelling and Bayesian Network for fault detection and prediction of complex engineered systems, *Reliab. Eng. Syst. Saf.*, **215** (2021), 107808. https://doi.org/10.1016/j.ress.2021.107808
20. P. Arpaia, U. Cesaro, M. Chadli, H. Coppier, L. De Vito, A. Esposito, et al., Fault detection on fluid machinery using Hidden Markov Models, *Measurement*, **151** (2020), 107126. https://doi.org/10.1016/j.measurement.2019.107126
21. B. A. Fernandes, G. D. Colletta, L. H. C. Ferreira, O. O. Dutra, Utilization of Savitzky-Golay filter for power line interference cancellation in an embedded electrocardiographic monitoring platform, in *2017 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, (2017), 227–232. https://doi.org/10.1109/MeMeA.2017.7985880
22. R. W. Schafer, What Is a Savitzky-Golay Filter? [Lecture Notes], *IEEE Signal Process. Mag.*, **28** (2011), 111–117. https://doi.org/10.1109/MSP.2011.941097
23. G. Xu, X. Wang, Y. Zhao, Adaptive fault diagnosis for thruster system of underwater vehicles, *Ship Sci. Technol.*, **42** (2020), 95–100. https://doi.org/10.3404/j.issn.1672-7649.2020.06.019
24. J. Kennedy, R. Eberhart, Particle swarm optimization, in *Proceedings of ICNN'95 - International Conference on Neural Networks*, **4** (1995), 1942–1948. https://doi.org/10.1109/ICNN.1995.488968