Smart Structures and Systems, *Vol. 31*, *No. 3* (*2023*) *247-257*
https://doi.org/10.12989/sss.2023.31.3.247

Optimal sensor placement for structural health monitoring based on deep reinforcement learning

Xianghao Meng [1,2], Haoyu Zhang [1,2], Kailiang Jia [1,2], Hui Li [1,2] and Yong Huang [*1,2]

[1] *Key Lab of Structures Dynamic Behavior and Control of the Ministry of Education, School of Civil Engineering, Harbin Institute of Technology, Harbin, China*
[2] *Key Lab of Smart Prevention and Mitigation of Civil Engineering Disasters of the Ministry of Industry and Information Technology, Harbin Institute of Technology, Harbin, China*

(*Received March 11, 2022, Revised December 2, 2022, Accepted January 6, 2023*)

Abstract. In structural health monitoring of large-scale structures, optimal sensor placement plays an important role because of the high cost of sensors and their supporting instruments, as well as the burden of data transmission and storage. In this study, a vibration sensor placement algorithm based on deep reinforcement learning (DRL) is proposed, which can effectively solve non-convex, high-dimensional, and discrete combinatorial sensor placement optimization problems. An objective function is constructed to estimate the quality of a specific vibration sensor placement scheme according to the modal assurance criterion (MAC). Using this objective function, a DRL-based algorithm is presented to determine the optimal vibration sensor placement scheme. Subsequently, we transform the sensor optimal placement process into a Markov decision process and employ a DRL-based optimization algorithm to maximize the objective function for optimal sensor placement. To illustrate the applicability of the proposed method, two examples are presented: a 10-story braced frame and a sea-crossing bridge model. A comparison study is also performed with a genetic algorithm and particle swarm algorithm. The proposed DRL-based algorithm can effectively solve the discrete combinatorial optimization problem for vibration sensor placements and can produce superior performance compared with the other two existing methods.

Keywords: deep reinforcement learning; discrete combinatorial optimization; modal assurance criterion; sensor placement; structural health monitoring

1. Introduction

Structural health monitoring (SHM) is of great significance to structural safety, maintenance of the infrastructures and has received high attention over the last years (Borlenghi *et al*. 2021, Kaloop *et al*. 2020, Chalioris *et al*. 2020, Lydon *et al*. 2019, Wu *et al*. 2019, Huang *et al*. 2017). In SHM, a rational and effective arrangement of sensors plays an important role in ensuring the safety of the structure (AlSaleh and Clemente 2020). With the continuous development of structural engineering, an increasing number of large-scale complex structures have emerged, such as super high-rise buildings and long-span bridges. In theory, the amount of structural information obtained is proportional to the number of sensors placed (Altunisik *et al*. 2021); thus, the actual service condition of the structure can be more accurately evaluated. However, if the number of sensors installed on the structure is too large, it will lead to an overly high cost of system maintenance and waste. In addition, too many sensors may cause an excessive burden on data transmission and storage (Quqa *et al*. 2021). Therefore, in SHM, it is necessary to determine the optimal locations of the sensor arrangement with a certain number of sensors.

Sensor placement optimization is generally divided into two steps. In the first step, based on the requirement that important data information of the structural response should be effectively collected, we selected the appropriate sensor placement criterion and define the objective function. Subsequently, effective optimization algorithms are employed to determine the best solution for the objective function; that is, the optimal placement of the sensors. Shah and Udwadia (1978) first studied the problem of optimal sensor placement, and time-domain data were used to optimize sensor placement by minimizing the norm of the covariance matrix of the estimation parameters. Kammer (1991) proposed an effective independent method that determines the position of sensors by deleting the sensor position that contributes the least to the modal linear independence to obtain the best modal parameter estimation. Carne and Dohrmann (1994) investigated the utilization of the modal assurance criterion (MAC) for the selection of sensor placement locations, which aims to measure modal shapes as independently as possible. Yin *et al*. (2017) studied the optimal sensor placement of periodic structures ending with bolted joints based on information entropy. Liu *et al*. (2020) developed a sensor placement framework based on the Fisher information matrix to improve damage detection accuracy. Hosseini-Toudeshky and Amjad (2021) studied the sensor placement

*Corresponding author, Professor,
E-mail: huangyong@hit.edu.cn

Copyright © 2023 Techno-Press, Ltd.
http://www.techno-press.org/?journal=sss&subpage=7 ISSN: 1738-1584 (Print), 1738-1991 (Online)

optimization for guided wave-based structural health monitoring by using the detection theory and Bayesian risk framework.

Several optimization algorithms have been applied to obtain a satisfactory sensor placement scheme. Zhao and Nehorai (2007) employ the Gauss-Newton method to optimize the wireless sensor networks in a statistical measurement model. Akbarzadeh *et al*. (2014) proposed an adaptation of the gradient descent method to optimize the position and orientation of the sensors. Recently, with the development of bionics, swarm intelligence optimization algorithms have been proposed. Liu *et al*. (2008) used a genetic algorithm (GA) to optimize the sensor placement of a spatial lattice structure. He *et al*. (2015) improved particle swarm optimization (PSO) to efficiently optimize the sensor placement of a two-dimensional truss structure. Sun and Büyüköztürk (2015) applied an artificial bee colony algorithm to an integer optimization problem for sensor placement. Yi *et al*. (2015) introduced a biological immune mechanism, and developed an improved monkey group algorithm.

Optimal sensor placement is a discrete combinatorial optimization problem. Although different solutions have been proposed (Qin *et al*. 2021, Hou *et al*. 2019), it remains challenging to determine a global optimal solution for large-scale and complex structures (Yang and Lu 2017). Wang *et al*. (2019) applied reinforcement learning (RL) to an optimal sensor placement problem and verified the effectiveness of RL. However, the computer will be out of memory when the dimensions of the sensor placement optimization are high. In recent years, deep learning has been widely used in structural health monitoring (Wang *et al*. 2021, Huang *et al*. 2021, Ding *et al*. 2020, Rosafalco *et al*. 2020) because of its powerful capability to recognize patterns and to build regression and classification models for high-dimensional complex problem or system. In this study, a deep reinforcement learning (DRL)-based sensor optimal placement algorithm was presented. DRL is a combination of deep neural networks and RL with perceptual and decision-making abilities (Mnih *et al*. 2013). The algorithm has been successfully applied in various fields, including machine vision (Oh *et al*. 2015), drone control (Chang *et al*. 2020, Lin *et al*. 2020, Clegg *et al*. 2020), medical treatment (Xu *et al*. 2021, Krishna *et al*. 2021), path planning (Singh *et al*. 2017, Yurtsever *et al*. 2020), and the solving of partial differential equations (Wei *et al*. 2019). In the DRL algorithm, a neural network is employed instead of the original Q-table in RL to fit the mapping relationship between the action and value of the action (Mnih *et al*. 2013), which breaks the limitation that RL cannot be applied to high-dimensional problems (Oh *et al*. 2015). Minh *et al*. (2015) proposed a deep Q network (DQN). During the training procedure, two neural networks were used to improve the fitting accuracy of the functional relationship between the input and the output. The DQN optimization algorithm can realize discrete representation and determine a satisfactory solution through the optimization process of reward feedback, which can suitably solve the problem of high-dimensional sensor optimal placement. We employ DRL method in this study based on the consideration that DRL provides a novel approach to model an optimization problem into a state-action-transition-reward framework without any additive assumption. It is powerful to form a general framework for a bunch of optimization problems including sensor placements, compared with the specifically designed algorithms. DRL methods also have been proved to be effective and efficient in optimization problems with complex state and/or solution spaces, and with complex objective functions. Sensor placement is non-convex, high-dimensional, and discrete combinatorial optimization problem and it is appropriate to employ this method.

The contribution of the paper is twofold. From the theory aspect, the MAC for quantifying the independence among different modal shapes was selected as the criterion to optimize vibration sensor placement. We then transformed the sensor optimal placement process into a Markov decision process and employed a DRL-based optimization algorithm to maximize the objective function for optimal sensor placement. In the DRL-based optimization algorithm, the sensor placement vector, objective function, and position change of the sensor are assigned as the state, reward, and action, respectively. Moreover, a Double Deep Q-network model was introduced to address the overestimation problem. From the application aspects, a braced frame structural model and a sea-crossing bridge model were employed to verify the effectiveness of the algorithm. The corresponding two optimal sensor placement problems of vibration sensors were solved, and a comparison study with the traditional genetic algorithm and particle swarm optimization algorithm was performed. Our proposed DRL-based optimization algorithm enables to produce a superior sensor configuration, which generates measured modal shapes that have good linear independence from each other. Moreover, the algorithm can directly realize zero or one discrete representation of the result, providing clear support for sensor placement decisions.

The remainder of this paper is organized as follows. In Section 2, a DRL-based optimization algorithm for optimal sensor placement is proposed, including the selection of the objective function, introduction of an improved DRL algorithm, and algorithmic design of the DRL-based sensor placement optimization. We then present two illustrative examples and compare the proposed algorithm with two traditional methods in Section 3, followed by conclusions and future work in Section 4.

2. DRL-Based algorithm for optimal sensor placement

2.1 Objective function

Modal information is vital to understanding the structural dynamic behavior, as the identified structural modal parameters are functions of structural physical parameters; that is, stiffness, mass, and damping. However, in practice, the measured degree of freedom (DOF) is much less than that of the full DOFs of the structure model, which makes it difficult for the obtained modal vectors to guarantee modal orthogonality. In some cases, the important modal information may be lost owing to the small space angle of the modal vectors (Carne and Dohrmann 1994). To study this issue, the MAC matrix is introduced with the element $MAC_{i,j}$ in the i-th row and j-th column, defined as follows

$$MAC_{i,j} = \frac{(\phi_i \phi_j^T)^2}{(\phi_i \phi_i^T)(\phi_j \phi_j^T)} \quad (i,j = 1,2,3\dots N_m) \tag{1}$$

where ϕ_i refers to the i-th modal shape in the modal matrix corresponding to the measured DoFs with sensors placed, and N_m is the total number of investigated modes.

$MAC_{i,j}$ can be considered the cosine value of the angle between the i-th and j-th modal vectors. The smaller the value, the larger the angle between the two vector spaces, and vice versa. Therefore, the non-diagonal element $MAC_{i,j}$ ($i \neq j$) in the MAC matrix can reflect the angle of intersection between the i-th and j-th modal vectors. Generally, when $MAC_{i,j}$ is greater than 0.9, the i-th and j-th modes are similar, and the two modes are indistinguishable; when the value is less than 0.05, the two modes are considered orthogonal to each other, and the two modes can be clearly distinguished (Ewins 1986). Therefore, vibration sensor placement can be realized by minimizing the maximum value of the non-diagonal elements of the MAC matrix. We define the objective function $f(\boldsymbol{\delta})$ of the sensor placement criterion based on the modal confidence theory

$$f(\boldsymbol{\delta}) = 1 - \max_{i \neq j}\{MAC_{i,j}\} \quad (i,j = 1,2,\dots N_m) \tag{2}$$

where $\boldsymbol{\delta} \in \mathbb{R}^{N_d}$ represents the sensor placement vector, which is composed of 1 and 0, implying one and no sensors arranged at a certain position, respectively; N_d refers to the number of candidate sensor placement points, which is usually identical to the number of DoFs of the structural model.

When the maximum value of the objective function $f(\boldsymbol{\delta})$ is obtained, optimal sensor placement is achieved. The optimal solution $\boldsymbol{\delta}^*$ is computed as

$$\boldsymbol{\delta}^* = \underset{\delta}{argmax} f(\boldsymbol{\delta}) \tag{3}$$

2.2 Improved deep reinforcement learning algorithm

2.2.1 Decision and update using Q-learning

Q-learning is a widely used algorithm in RL and was developed to address the nonlinear optimization problem. Particularly, it can be applied for discrete combinatorial optimization and is suitable for the optimization of sensor placement (Wang *et al.* 2019). The Q-learning decision process is a Markov decision process (Kalnoor and Subrahmanyam 2020). The main components of RL include agents, actions, states, and reward (Sutton and Barto 1998). In the sensor placement problem, the current sensor placement configuration, change in the position of the sensor, and objective function can be regarded as the state s_t, action a_t, and reward r_t, respectively. The value of action a_t based on the state s_t can be expressed as $Q(s_t,a_t)$, and the relationship between $Q(s_t,a_t)$ and state s_t and a_t is represented using the Q-table, which is shown in Table 1.

Table 1 *Q*-table (given the evaluation values of the state s_t and action a_t)

State	Action		
	a_1	a_2	…
s_1	$Q(s_1,a_1)$	$Q(s_1,a_2)$	…
s_2	$Q(s_2,a_1)$	$Q(s_2,a_2)$	…
s_3	$Q(s_3,a_1)$	$Q(s_3,a_2)$	…
…	…	…	…

Different from the reward r_t which is a single step reward, the Q-value $Q(s_t,a_t)$ here is intended to represent the expectation of cumulative reward from the current state s_t to the final sensor placement configuration within a finite number of steps, when the current action a_t is selected. This is helpful to avoid local optimization when selecting an action.

Generally, an agent is inclined to choose an action that obtains the best sensor placement solution, that is, the action a_t with the highest $Q(s_t,a_t)$ valve; the new sensor placement s_{t+1} and the corresponding reward r_t are then produced. During the exploration of the agent, $Q(s_t,a_t)$ can be updated according to the following formula

$$\begin{aligned} &Q_{k+1}(s_t,a_t) \leftarrow Q_k(s_t,a_t) \\ &+\alpha\left[r_t + \gamma \underset{a_{t+1}}{max} Q_k(s_{t+1},a_{t+1}) - Q_k(s_t,a_t)\right] \end{aligned} \tag{4}$$

where $\alpha \in (0,1]$ is the learning rate, γ is the discount factor, and k is the execution time of the action.

When the number of updates is sufficiently large, $Q(s_t,a_t)$ converges to a stable value. Thus, the agent can maximize the cumulative reward when at state s_t, and we choose a_t with the highest $Q(s_t,a_t)$ (Sutton and Barto 1998).

When facing a low-dimensional problem, the Q-table of the RL algorithm can sufficiently store $Q(s_t,a_t)$ corresponding to state s_t and action a_t.

2.2.2 Deep Q-network and double deep Q-network for deep reinforcement learning

For high-dimensional problems, the use of a Q-table in the RL optimizer is typically faced with the difficulties of insufficient computer storage space and the intractability of the algorithm. To address this problem, a deep neural network is introduced to replace the Q-table, and a DRL algorithm is formed (Zhao *et al.* 2019). To realize this, DQN is employed to approximate the real mapping relationship $Q^*(s,a)$ between state s and action a as the neural network output by continuously updating the neural network parameters $\boldsymbol{\theta}$ as follows

$$Q^*(s,a) \approx Q(s,a;\boldsymbol{\theta}) \tag{5}$$

where $Q(s,a;\boldsymbol{\theta})$ is the neural network function of state s and action a and parameters $\boldsymbol{\theta}$.

Theoretically, continuous training of the network model can accurately fit the mapping between the Q^* value and the

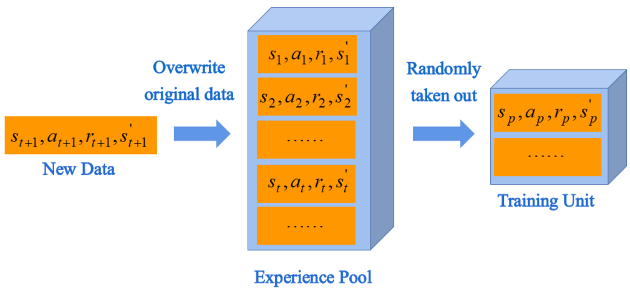


Fig. 1 Replay buffer to access the memory sequence obtained during the training process

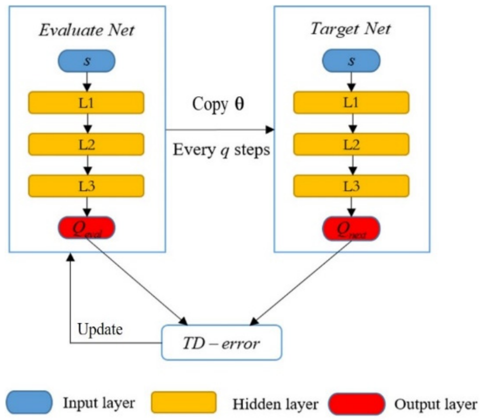


Fig. 2 Double deep neural networks in DRL

state-action pair $\{s,a\}$ in RL. Therefore, DQN provides an effective tool to solve the high-dimensional sensor placement optimization problem. The algorithm mainly comprises two improvements to the Q-learning.

(1) Replay buffer utilization

The neural network training process includes a fixed-capacity experience pool that can store the memory samples. The access process for the memory sample sequence is illustrated in Fig. 1. Each memory sample is denoted by $e_t = \{s_t, a_t, r_t, s_{t+1}\}$ and $t = 1, 2, \ldots, T_{max}$, where T_{max} is the total number of exploration steps. In the exploration process, new memory samples interacting with the environment are stored in the experience pool, and old memory samples are overwritten. During training, a small batch of memory samples is always extracted from the experience pool to update the network parameters via $\boldsymbol{\theta}$. Utilization of the replay buffer reduces the correlation between data sequences (Mnih *et al*. 2015).

(2) Double deep neural networks

In DQN, double deep neural networks are adopted, named evaluate net and target net, and their network parameters are $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$, respectively. At the beginning of the training, the two neural networks employed unified network parameters. When the agent is exploring, the evaluate net interacts with the environment and obtains memory samples, whereas the target net saves the network parameters to provide the training target. To update the neural network parameters $\boldsymbol{\theta}$ of evaluate net, the index of the temporal difference error (*TD-error*) is defined as

$$TD - error = [r_t + \gamma max(Q(s_{t+1}, a_{t+1}; \boldsymbol{\theta}')) - Q(s_t, a_t; \boldsymbol{\theta})]^2 \tag{6}$$

where $r_t + \gamma max(Q(s_{t+1}, a_{t+1}; \boldsymbol{\theta}'))$ is used as the fitting target and γ represents the reward discount factor.

During the training procedure, the evaluate net parameters $\boldsymbol{\theta}$ are updated using *TD-error*; where the target net parameters $\boldsymbol{\theta}'$ are set the same as $\boldsymbol{\theta}$ for each q step. This strategy can improve the stability of the algorithm (Mnih *et al*. 2013). A schematic of the double deep neural networks is shown in Fig. 2.

Note that *TD-error* obtained using Eq. (6) will lead to the problem of overestimation, and double deep Q-network (DDQN) is introduced to address this problem (Van Hasselt *et al*. 2016). DDQN differs from DQN in the method in which *TD-error* is calculated. *TD-error* in DDQN is defined as follows

$$\text{TD-error} = \left[r_t + \gamma Q\left(s_{t+1}, \underset{a_{t+1}}{argmax}\left(Q(s_{t+1}, a_{t+1}; \boldsymbol{\theta})\right), \boldsymbol{\theta}'\right) - Q(s_t, a_t; \boldsymbol{\theta})\right]^2 \tag{7}$$

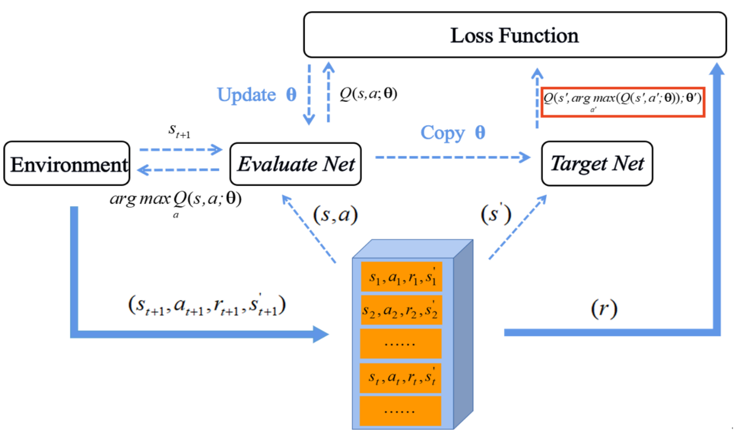


Fig. 3 DDQN optimization algorithm flowchart

In the DDQN, the *TD-error* calculated using Eq. (7) can realize a better mapping between the Q-value and state-action pair$\{s,a\}$. The comparison study in later examples verifies the superiority of the DDQN compared with the DQN. Therefore, the DDQN will be used in the sensor placement optimization, and the algorithm flowchart of the DDQN is demonstrated in Fig. 3.

Sensor placement is a discrete combinatorial optimization problem. When the structure is large-scale and complex, that is, many DoFs exist in the structural model, there is an explosion of alternatives to sensor placements, and the problem becomes NP-hard. Such problems are generally high-dimensional, discontinuous, and non-convex. The DDQN possesses strong generalization ability and can effectively solve high-dimensional problems. Markov decision process can be applied to discrete problems. Moreover, with the help of RL to learn the relationship between different states, the algorithm can solve non-convex and optimization problems. Based on the above characteristics of RL, even in the case of a limited number of explorations, the approximation mapping between the Q values and state-action pair $\{s,a\}$ established by DDQN is sufficient to support the agent in exploring a satisfactory sensor-placement solution.

2.3 DRL-based sensor placement optimization algorithm

To address the problem of optimal sensor placement, the DDQN model is built, including the state space, action space, and reward function.

2.3.1 State space

As mentioned after Eq. (2), binary coding is performed at each position to characterize the discrete sensor placement, where 0 and 1 imply no and one sensor is placed at a certain position. For example, a sensor placement vector $\boldsymbol{\delta} = [\delta_1, \delta_2, \ldots \delta_i \ldots, \delta_{N_d}]^T$, where δ_i $(i = 1,\ 2, \ldots, N_d) \in \{0, 1\}$, and only N_o of them are 1, corresponds to a structural model with N_d DoFs and N_o number of sensors. In this algorithm, all possible sensor placement vectors are adopted as the state space of DDQN.

2.3.2 Action space

During the decision-making process for optimal sensor placement, the agent executes an action in the action space to change the sensor placement state. In our algorithm, the action refers to the change in the position of one sensor, that is, flipping the binary digit of the corresponding entry in the sensor placement vector $\boldsymbol{\delta}$. For example, the sensor placement vector is $[1,1,0,0,1,0\ldots0,1,0,0,1]$, and one action may be $[1,1,0,0,1,0\ldots0,1,0,0,1] \rightarrow [0,1,1,0,1,0\ldots0,1,0,0,1]$ (this action moves the sensor from the first to the third position). It can be inferred that $N_o \times (N_d - N_o)$ actions exist for each state and the total number of actions in the state space is $C_{N_d}^{N_o} \times N_o \times (N_d - N_o)$, which is of significant value for large-scale and complex structures.

Algorithm: DRL-based optimization algorithm

Input: start learning steps t_{start}, memory size *M*, learn rate η, total exploration steps T_{max}, reward discount factor γ, greedy degree ξ, discrete-time parameter $t_{\max}$, and parameter replacement interval q.
Output: optimal sensor placement solution S^* and maximum objection function $f(S^*)$.

1. Initialize experience pool D with capacity *M*, and step number $t \leftarrow 0$;
2. Initialize record maximum reward $r^* \leftarrow 0$, and optimal state s^* arbitrarily;
3. Initialize the *evaluate net* parameters $\boldsymbol{\theta}$ with random values;
4. Initialize the *target net* parameters $\boldsymbol{\theta}'$: $\boldsymbol{\theta}' \leftarrow \boldsymbol{\theta}$;
5. **while** $t \leq T_{max}$ **do**
6. Set $n = 1$, and initialize sensor placement s_1;
7. **While** $n \leq t_{\max}$ **do**
8. Choose action a_t based on greedy degree ξ using the greedy strategy in Mnih *et al*. (2015):
9. Execute action a_t, and produce obtain the new sensor placement state s_{t+1} and reward r_t by Eq. (8);
11. Store the memory sequence $\{s_t, a_t, r_t, s_{t+1}\}$ in experience pool D;
12. **if** $\boldsymbol{r_t > r^*}$
13. $\boldsymbol{r^* \leftarrow r_t, s^* \leftarrow s_{t+1}}$**;**
14. **end**
15. **if** $t \geq t_{start}$
16. Randomly sample the minibatch of memory sequences$\{s_t, a_t, r_t, s_{t+1}\}$from experience pool D;
17. Compute the temporal difference error $TD - error$ using Eq. (7)
18. Train *evaluate net* and update $\boldsymbol{\theta}$ using $TD - error$ by RMSProp optimizer;
19. **end**
20. **if** $t\,\mathrm{Mod}\,q = 0$
    Update parameters $\boldsymbol{\theta}'$: $\boldsymbol{\theta}' \leftarrow \boldsymbol{\theta}$;
21. **end**
22. $s_t \leftarrow s_{t+1}, t \leftarrow t + 1, n \leftarrow n + 1$;
23. **end**
24. **end**
25. Obtain the optimal sensor placement solution $S^* \leftarrow s^*$.

2.3.3 Reward function

After performing an action, the sensor placement solution must be evaluated, and the corresponding evaluation value is regarded as the reward. In our algorithm, the MAC-based objective function $f(\boldsymbol{\delta})$ is used as the reward function, which is given by

$$r_t = f(\boldsymbol{\delta}_t) \tag{8}$$

where $\boldsymbol{\delta}_t$ is the sensor placement vector for the next state, s_{t+1}, which is obtained by the selected action a_t.

2.3.4 The pseudocodes of the algorithm

Aiming at the characteristics of the sensor optimal placement problem, we propose a sensor optimal placement algorithm based on the DRL. The pseudocode of the algorithm is shown on the previous page.

In the algorithm, the parameters are first initialized in Lines 1-5. In Line 6, the initialization state of s_1 is performed with evenly distributed sensors. Subsequently, the process of exploration is implemented in Lines 7-11, in which the simple memory sequence related to the sensor placement is obtained and stored in the memory pool; Next, the current optimal sensor placement solution and the corresponding reward value are recorded in Lines 12-14. Finally, Lines 15-19 implement the parameter update process.

3. Illustrative examples

In this section, two illustrative examples are presented to verify the effectiveness of the proposed DRL-based optimization algorithm, in which a braced-framed and a bridge structural model are employed.

3.1 Example 1: three-dimensional ten-story braced frame

In this example, a finite element model (FEM) of a ten-story three-dimensional braced frame is first established. Four columns are placed at four corners on each floor, and the mass of each floor is 10 metric tons. The plan for each floor is a square with a side length of 4 m. The interstory stiffnesses of each column are 10 MN/m and 15 MN/m in the x and y directions, respectively. Moreover, the face of each floor is reinforced by a brace with a stiffness of 20 MN/m. Therefore, the interstory stiffnesses are 80 and 100 MN/m in the x and y directions, respectively. Consequently, the first eight natural frequencies of the structural model are 2.02 Hz, 2.13 Hz, 3.69 Hz, 6.02 Hz, 6.34 Hz, 9.88 Hz, 10.40 Hz, 10.98 Hz, respectively. According to the structural information, the modal shapes can be readily computed (Yuen *et al*. 2006).

In Fig. 4, the structural model is demonstrated, where O_l is the center of the plan of each floor. The torsion of the structure can be calculated by measuring three positions on each floor (Yuen *et al*. 2006). Therefore, 30 candidate measuring points exist on this ten-story structure.

For this structure, the first eight modal shapes are employed to calculate the objective function $f(\boldsymbol{\delta})$ in Eq. (2). Subsequently, the DRL-based optimization algorithm is

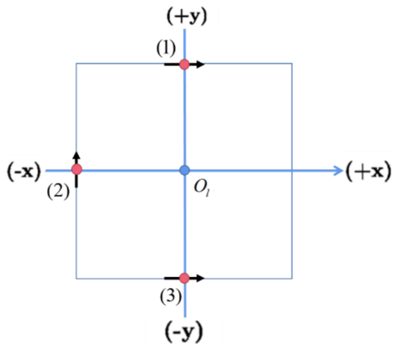


Fig. 4 Floor plan of the frame structure

used to optimize the placement of the vibration sensors of this frame structure by maximizing the objective function $f(\boldsymbol{\delta})$.

3.1.1 Optimization of the vibration sensor configuration

In this application, the number of vibration sensors to be arranged is assumed to be 10. In this case, the number of possible vibration sensor arrangement solutions can be computed as $C_{30}^{10} = 3.0045 \times 10^7$. According to the design of the action and state spaces in Subsection 2.3, the number of state spaces of the DRL is equal to the total number of vibration sensor placement solutions, and the number of actions of the DRL is $C_{30}^{10} \times 10 \times (30 - 10) = 6.0090 \times 10^9$. The introduction of a neural network to replace the Q-table in RL is necessary to address such numerous actions.

For the design of the DRL-based optimization algorithm, the hyperparameters are selected as follows: start learning steps $t_{start} = 3000$, memory size M=2000, total training steps N = 80000, learning rate $\eta = 0.001$ and $\eta = 0.0001$ for the first 20000 steps and the next 60000 steps, the reward discount factor $\gamma = 0.5$, greedy degree $\xi = 0.97$, and $\xi = 1.0$ for the first 20000 steps and the last 60000 steps, respectively, discrete-time parameter $t_{\max} = 1$, network parameter replacement interval q = 1000, and $batch\ size = 128$.

Using the selected hyperparameters of the algorithm, two deep neural networks, evaluate net and target net, are built, which have the same 3-layer network structure with 64 neurons in the hidden layer, and the numbers of input and output nodes are equal to 30 and 200, respectively. The

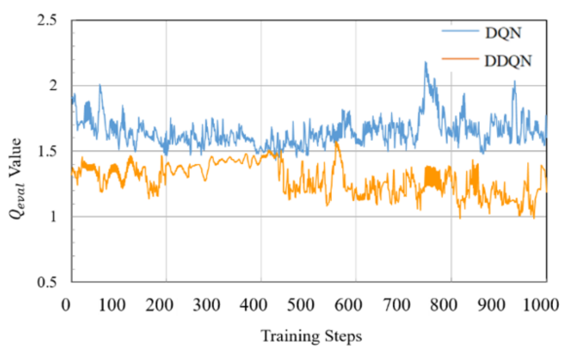


Fig. 5 Q_{eval} of DQN and DDQN in the training process sigmoid function is selected as the activation function, and the RMSProp optimizer is employed to train the neural network.

Initially, we run the DQN optimization algorithm; however, it is difficult to find a satisfactory solution for sensor placement because of the overestimation problem (Van Hasselt *et al*. 2016), which makes Q_{eval} larger than Q^*. The Q_{eval} of the two algorithms is shown in Fig. 5, and it can be observed that the Q_{eval} value of the DDQN is lower than that of the DQN. Therefore, the overestimation problem in DQN can be solved using DDQN for optimal sensor placement.

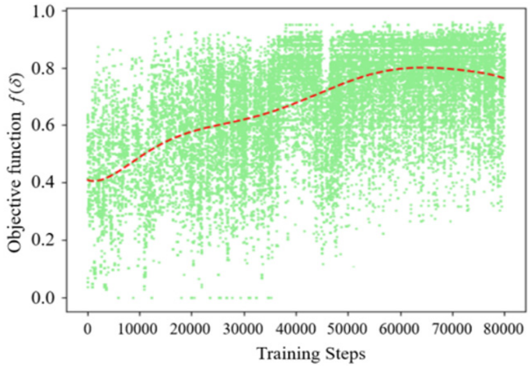
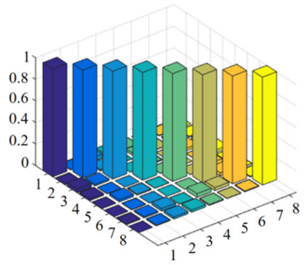
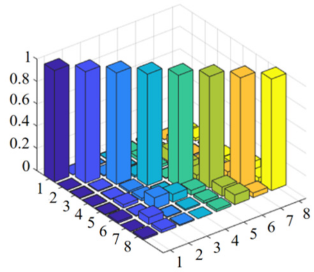


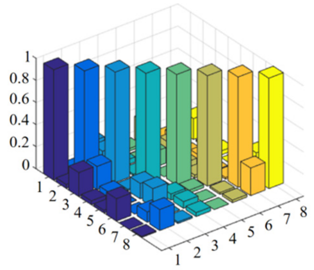
Fig. 6 Visualization of the optimization process



(a) DRL



(b) PSO



(c) GA

Fig. 7 Histogram of MAC matrix corresponding to the optimization results of three algorithms (braced frame structure)

In Fig. 6, the optimization process of the DRL-based optimization algorithm is demonstrated, where the green dots and red dashed curve indicate the objective function $f(\boldsymbol{\delta})$ in different states and the trend generated from least-squares fitting, respectively. As observed, the proposed DRL-based optimization algorithm has a relatively small coefficient of variation of the objective function value in the later stage of training, and the mean value of the objective function $f(\boldsymbol{\delta})$ can reach a value of 0.8. After 80000 steps of exploration, we finally obtain the optimized objective function value, $f(\boldsymbol{\delta}^*) = 0.9649379$. The corresponding optimized sensor placement vector is $\boldsymbol{\delta}^* =$ [0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0]$^{\mathrm{T}}$, that is, the optimal sensor indices are 5, 7, 12, 15, 17, 19, 24, 26, 27 and 28.

In Fig. 7(a), the histogram of the MAC matrix elements calculated from the optimal sensor placement vector $\boldsymbol{\delta}^*$ is shown. It is obvious that all non-diagonal elements of the MAC matrix are minor (the maximum value is 0.0350621, which is less than 0.05). Therefore, it can be concluded that the modal shapes identified by these optimally placed sensors have good linear independence among the various modal shapes (Carne and Dohrmann 1994).

To further verify the performance of the proposed algorithm, a comparison study with the traditional genetic algorithm (GA) and particle swarm optimization (PSO) algorithm is performed. In this study, the number of vibration sensors is 10 and the objective function $f(\boldsymbol{\delta})$ is set as the fitness function of the two algorithms. In the GA, the number of genomes is $N_d = 30$, and linear conditions are used to realize that the number of sensors is selected as 10. The bound of each variable is set as [0,1]. The mating probability and the mutation probability are selected as 0.8 and 0.09, respectively. In the particle swarm algorithm, the parameters of population dimension, population numbers, and iteration numbers are taken as 30, 40, and 100, respectively. In addition, the factor and the inertia factor are taken as 0.2 and 0.8, respectively. In Figs. 7(b) and (c), the MAC matrices obtained by the optimized sensor placement results are shown for the two algorithms. The non-diagonal elements of the MAC matrix obtained using the two optimization algorithms are obviously larger than the DRL-based algorithm among the three, meaning the independence of the measured modal information among the different modes obtained by the sensor placement solutions of the two algorithms is not high. Table 2 shows

Table 2 The optimization results of GA, PSO, and DRL-based algorithms when the number of vibration sensors is 10

Algorithms	Maximum value of off-diagonal elements	Objective function values $f(\boldsymbol{\delta})$
DRL	0.035063	0.964937
GA	0.249993	0.750007
PSO	0.102067	0.897933

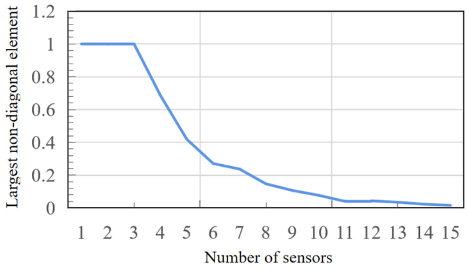


Fig. 8 Curve of $\max_{i \neq j}\{MAC_{i,j}\},\ i,j = 1,2\ldots8$ versus the number of measuring sensors (frame structure)

the sensor optimization results of the three algorithms, and it can be observed that the objective function optimized using the DRL-based optimization algorithm is the largest, which is consistent with the results in Fig. 7.

In Fig. 8, the MAC criterion-based objective function $f(\boldsymbol{\delta})$ is optimized by the DRL-based optimization algorithm, and the results with different numbers of vibration sensors are shown, where the abscissa and ordinate denote the number of sensors and index of $\max_{i \neq j}\{MAC_{i,j}\},\ i,j = 1,2\ldots8$, respectively. It can be observed that with an increase in the number of sensors, $\max_{i \neq j}\{MAC_{i,j}\},\ i,j = 1,2\ldots8$ decreases continuously. However, when the number of sensors reaches 10, the decreasing trend stabilizes, implying that any newly added sensors contribute little to reducing the independence of the measured modal information among different modes.

In addition, our proposed DRL-based optimization algorithm can directly realize a binary representation in which a single variable $\delta \in \{0,1\}$ exists, such that $\delta = 1$ and 0 represent a sensor that should or should not be placed at a certain location. This is suitable for sensor placement

(a) (b)

Fig. 9 (a) Haicang Bridge; (b) The corresponding FEM

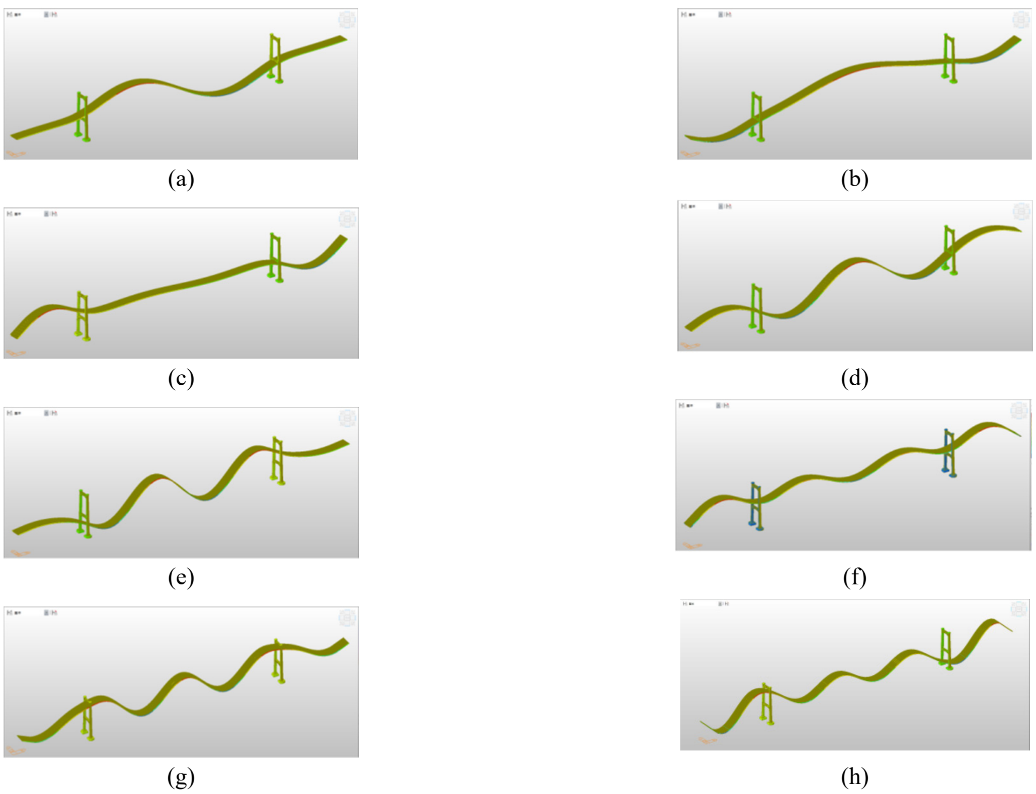
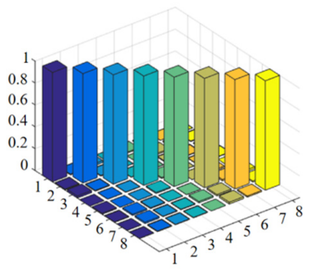
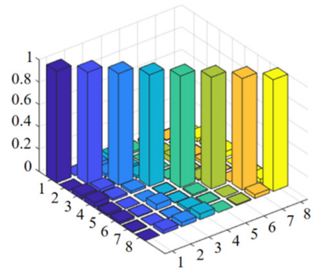


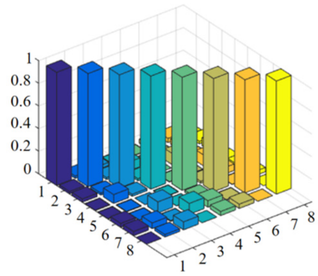
Fig. 10 First eight vertical modal shapes of the main girder and (a)-(h) are the vertical modes of orders 1-8, respectively



(a) DRL



(b) PSO



(c) GA

Fig. 11 Histogram of the MAC matrix corresponding to the optimization results of three algorithms (Haicang Bridge model)

decisions. Moreover, our proposed algorithm can overcome the problems of weak local optimization and premature convergence, which usually occur in GA and PSO algorithms.

3.2 Illustrative Example 2: a sea-crossing bridge model

In this example, the proposed algorithm is applied to a more complicated bridge structural model to verify its effectiveness. The prototype of this model is the Haicang Bridge in China, as shown in Fig. 9(a). Midas Civil software is employed to construct this structural model, as shown in Fig. 9(b), which contains 702 elements and 705 nodes.

The vertical modes of the bridge play a dominant role in the dynamic behavior of the bridge structure, and the main structural component of the vertical vibration is the main girder. The main girder of this bridge is simulated using girder elements in the FEM model with 87 nodes; thus, the girder is discretized into 87 DoFs, which are candidate locations to place vibration sensors. The first eight vertical modes $\{\phi_i\}_{i=1}^{8}$ of the main girder are calculated using Midas Civil software, as shown in Figs. 10(a)-(h).

Table 3 Optimization results of three algorithms for vibration sensor placement of Haicang Bridge

Algorithms	Maximum value of off-diagonal elements	Objective function values
DRL	0.018831	0.981169
GA	0.073907	0.926093
PSO	0.050186	0.949814

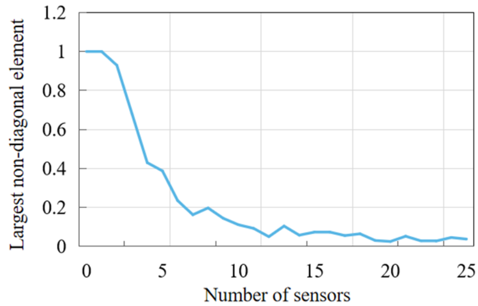


Fig. 12 Variation curve of the largest non-diagonal element of MAC matrix versus the number of measuring sensors (Haicang Bridge model)

Based on the first eight modes and candidate sensor placement scheme $\boldsymbol{\delta}$, the objective function $f(\boldsymbol{\delta})$ to evaluate the sensor placement solution can be calculated. For the 20 sensors in 87 candidate locations, $C_{87}^{20} = 2.375 \times 10^{19}$ possible sensor placement solutions exist, which is much larger than that in the previous example.

The architecture of DDQN in this example is the same as before, except that the number of input and output nodes are N_d = 87 and N_o = 20, respectively. The selection of the hyperparameters is also the same, except for the start learning steps t_{start} = 2500, memory size M = 2500, and network parameter replacement interval q = 800.

To verify the effectiveness of the algorithm for this high-dimensional sensor optimization problem, the GA and PSO algorithms are also introduced for comparison, and the results are presented in Fig. 11 and Table 3. The objective function value of our proposed algorithm is also the largest among the three algorithms. Correspondingly, $\max_{i \neq j}\{MAC_{i,j}\},\ i,j = 1,2\ldots8$ is smaller than that of the other two algorithms. That is, the position optimized by the proposed algorithm has better linear independence among different modal shapes. In conclusion, the DRL-based optimization algorithm produces the best optimization performance for sensor placement. In addition, it is observed that the non-diagonal elements of the three MAC matrices are relatively smaller than those in Subsection 3.1, presumably because of the more refined FEM of the Haicang Bridge compared to that in the previous example.

In Fig. 12, the largest nondiagonal element of the MAC matrix versus the number of sensors is shown. Similar to the results in Fig. 8, $\max_{i \neq j}\{MAC_{i,j}\},\ i,j = 1,2\ldots8$ decreased with an increase in the number of sensors. It can be observed that the changing trend of the maximum values stabilized when the number of sensors is 20; that is, the independence of the measured modal information among different modes is sufficiently high when the sensor number is 20.

4. Conclusions

An effective vibration sensor placement optimization algorithm based on deep reinforcement learning (DRL) is proposed in this study. The modal assurance criterion (MAC) for quantifying the independence among different modal shapes was selected as the criterion to optimize vibration sensor placement. We then transformed the sensor optimal placement process into a Markov decision process and employed a DRL-based optimization algorithm to maximize the objective function for optimal sensor placement. In the DRL-based optimization algorithm, the sensor placement vector, objective function, and position change of the sensor are considered as the state, reward, and action, respectively. A Double Deep Q-network was introduced to address the overestimation problem. Using the limited exploration steps of agents for DRL, the proposed algorithm can efficiently address high-dimensional, non-convex, and discrete combinatorial sensor placement optimization problem. In addition, this algorithm directly provided a binary representation of sensor placement results, which is perfectly suitable for sensor placement decisions.

To verify the effectiveness of the algorithm, a braced frame structural model and a sea-crossing bridge model were employed. We also performed a comparison study with the traditional genetic algorithm and particle swarm optimization algorithm. Our proposed DRL-based optimization algorithm obtained a superior sensor configuration, which obtained measured modal shapes that have good linear independence from each other. In addition, we observed that with an increase in the number of vibration sensors, the independence of different modal shapes corresponding to the optimal sensor placement increased gradually. However, the independence stabilized when the number of vibration sensors reached a certain value.

In future studies, it would be useful to explore effective DRL algorithms for multi-type sensor placements to increase the applicability of sensor placement methods.

Acknowledgments

The study was supported by National Key Research and Development Program of China under Grant No. 2021YFF0501003 and National Science Foundation of China under Grant Nos. U2139209 and 52078174.

References

Akbarzadeh, V., Lévesque, J.C., Gagné, C. and Parizeau, M. (2014), "Efficient sensor placement optimization using gradient descent and probabilistic coverage", *Sensors*, **14**, 15525-15552. https://doi.org/10.3390/s140815525

AlSaleh, R.J. and Clemente, F. (2020), "Combining GPS and accelerometers' records to capture torsional response of cylindrical tower", *Smart Struct. Syst., Int. J.*, **25**(1), 111-122. https://doi.org/10.12989/sss.2020.25.1.111

Altunisik, A.C., Sevim, B., Sunca, F. and Okur, F.Y. (2021), "Optimal sensor placements for system identification of concrete arch dams", *Adv. Concrete Constr., Int. J.*, **11**(5), 397-407. https://doi.org/10.12989/acc.2021.11.5.397

Borlenghi, P., Gentile, C. and Saisi, A. (2021), "Detecting and localizing anomalies on masonry towers from low-cost vibration monitoring", *Smart Struct. Syst., Int. J.*, **27**(2), 319-333. https://doi.org/10.12989/sss.2021.27.2.319

Carne, T.G. and Dohrmann, C.R. (1994), A Modal Test Design Strategy for Model Correlation (No. SAND-94-2702C; CONF-950240-4), Sandia National Labs., Albuquerque, NM, USA.

Chalioris, C.E., Voutetaki, M.E. and Liolios, A.A. (2020), "Structural health monitoring of seismically vulnerable RC frames under lateral cyclic loading", *Earthq. Struct., Int. J.*, **19**(1), 29-44. https://doi.org/10.12989/eas.2020.19.1.029

Chang, C.C., Tsai, J., Lu, P.C. and Lai, C.A. (2020), "Accuracy Improvement of Autonomous Straight Take-off, Flying Forward, and Landing of a Drone with Deep Reinforcement Learning", *Int. J. Computat. Intell. Syst.*, **13**(1), 914-919. https://doi.org/10.2991/ijcis.d.200615.002

Clegg, A., Erickson, Z., Grady, P., Turk, G., Kemp, C.C. and Liu, C.K. (2020), "Learning to collaborate from simulation for robot-assisted dressing", *IEEE Robot. Automat. Lett.*, **5**(2), 2746-2753. https://doi.org/10.1109/LRA.2020.2972852

Ding, Z., Li, J. and Hao, H. (2020), "Structural damage identification by sparse deep belief network using uncertain and limited data", *Struct. Control Health Monitor.*, **27**(5), e2522. https://doi.org/10.1002/stc.2522

Ewins, D.J. (1986), "Modal Testing: Theory and Practice", *J. Vib. Acoust. Stress Reliabil. Des.*, **108**(1), 109-110. https://doi.org/10.1115/1.3269294

He, C., Xing, J., Li, J., Yang, Q., Wang, R. and Zhang, X. (2015), "A new optimal sensor placement strategy based on modified modal assurance criterion and improved adaptive genetic algorithm for structural health monitoring", *Mathe. Probl. Eng.*, **11**, 1-10. https://doi.org/10.1155/2015/626342

Hosseini-Toudeshky, H. and Amjad, F.A. (2021), "Sensor placement optimization for guided wave-based structural health monitoring", *Struct. Monitor. Maint., Int. J.*, **8**(2), 125-150. https://doi.org/10.12989/SMM.2021.8.2.125

Hou, R.R., Xia, Y. and Zhou, X.Q. (2019), "Genetic algorithm based optimal sensor placement for L_1-regularized damage detection", *Struct. Control Health Monitor.*, **26**(1), e2274. https://doi.org/10.1002/stc.2274

Huang, Y., Beck, J.L. and Li, H. (2017), "Bayesian system identification based on hierarchical sparse Bayesian learning and Gibbs sampling with application to structural damage assessment", *Comput. Meth. Appl. Mech. Eng.*, **318** (2017), 382-411. https://doi.org/10.1016/j.cma.2017.01.030

Huang, Y., Zhang, H., Li, H. and Wu, S. (2021), "Recovering compressed images for automatic crack segmentation using generative models", *Mech. Syst. Signal Process.*, **146**. https://doi.org/107061. 10.1016/j.ymssp.2020.107061

Kalnoor, G. and Subrahmanyam, G. (2020), "A review on applications of Markov decision process model and energy efficiency in wireless sensor networks", *Procedia Comput. Sci.*, **167**, 2308-2317. https://doi.org/10.1016/j.procs.2020.03.283

Kaloop, M.R., Elsharawy, M., Abdelwahed, B., Hu, J.W. and Kim, D. (2020), "Performance assessment of bridges using short-period structural health monitoring system: Sungsu bridge case study", *Smart Struct. Syst., Int. J.*, **26**(5), 667-680. https://doi.org/10.12989/sss.2020.26.5.667

Kammer, D.C. (1991), "Sensor placement for on-orbit model identification and correlation of large space structures", *J. Guid.*

Control Dyn., **14**(2), 251-259.
Krishna, A., Bartake, K., Niu, C., Wang, G., Lai, Y., Jia, X. and Mueller, K. (2021), "Image synthesis for data augmentation in medical ct using deep reinforcement learning", arXiv preprint arXiv:2103.10493.
Lin, K., Gong, L., Li, X., Sun, T., Chen, B., Liu, C., Zhang, Z., Pu, J. and Zhang, J. (2020), "Exploration-efficient deep reinforcement learning with demonstration guidance for robot control", arXiv preprint arXiv:2002.12089.
Liu, W., Gao, W.C., Sun, Y. and Xu, M.J. (2008), "Optimal sensor placement for spatial lattice structure based on genetic algorithms", *J. Sound Vib*., **317**, 75-189. https://doi.org/10.1016/j.jsv.2008.03.026
Liu, C.Y., Teng, J. and Zhen, P. (2020), "Optimal sensor placement for bridge damage detection using deflection influence line", *Smart Struct. Syst., Int. J*., **25**(2), 169-181. https://doi.org/10.12989/sss.2020.25.2.169
Lydon, D., Taylor, S.E., Lydon, M., del Rincon, J.M. and Hester, D. (2019), "Development and testing of a composite system for bridge health monitoring utilising computer vision and deep learning", *Smart Struct. Syst., Int. J*., **24**(6), 723-732. https://doi.org/10.12989/sss.2019.24.6.723
Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. and Riedmiller, M. (2013), "Playing atari with deep reinforcement learning", arXiv preprint, arXiv:1312.5602
Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G. and Petersen, S. (2015), "Human-level control through deep reinforcement learning", *Nature*, **518**(7540), 529-533. https://doi.org/10.1038/nature14236
Oh, J., Guo, X., Lee, H., Lewis, R.L. and Singh, S. (2015), "Action-conditional video prediction using deep networks in Atari games", *Adv. Neural Inform. Process. Syst*., **28**.
Qin, X., Zhan, P., Yu, C., Zhang, Q. and Sun, Y. (2021), "Health monitoring sensor placement optimization based on initial sensor layout using improved partheno-genetic algorithm", *Adv. Struct. Eng*., **24**(2), 252-265. https://doi.org/10.1177/1369433220947198
Quqa, S., Giordano, P.F., Limongelli, M.P., Landi, L. and Diotallevi, P.P. (2021), "Clump interpolation error for the identification of damage using decentralized sensor networks", *Smart Struct. Syst., Int. J*., **27**(2), 351-363. https://doi.org/10.12989/sss.2021.27.2.351
Rosafalco, L., Manzoni, A., Mariani, S. and Corigliano, A. (2020), "Fully convolutional networks for structural health monitoring through multivariate time series classification", *Adv. Model. Simul. Eng. Sci*., **7**(1), 1-31. https://doi.org/10.1186/s40323-020-00174-1
Shah, P. and Udwadia, F.E. (1978), "A methodology for optimal sensor locations for identification of dynamic systems", *J. Appl. Mech. Transact. ASME*, **45**(1), 188-196. https://doi.org/10.1115/1.3424225
Singh, Y., Sharma, S., Sutton, R. and Hatton, D. (2017), "Path planning of an autonomous surface vehicle based on artificial potential fields in a real time marine environment", *Proceedings of the 16 International Conference on Computer and IT Applications in the Maritime Industries*, May.
Sun, H. and Büyüköztürk, O. (2015), "Optimal sensor placement in structural health monitoring using discrete optimization", *Smart Mater. Struct*., **24**(12), 125. 10.1088/0964-1726/24/12/125034
Sutton, R.S. and Barto, A.G. (1998), "Reinforcement Learning: An Introduction", *IEEE Transact. Neural Networks*.
Van Hasselt, H., Guez, A. and Silver, D. (2016), "Deep reinforcement learning with double q-learning", *Proceedings of the AAAI Conference on Artificial Intelligence*.
Wang, Z., Li, H.X. and Chen, C.L. (2019), "Reinforcement learning-based optimal sensor placement for spatiotemporal modeling", *IEEE Transact. Cybernet*., **50**(6), 2861-2871. https://doi.org/10.1109/TCYB.2019.2901897
Wang, R., Li, J., An, S., Hao, H., Liu, W. and Li, L. (2021), "Densely connected convolutional networks for vibration based structural damage identification", *Eng. Struct*., **245**, 112871. https://doi.org/10.1016/j.engstruct.2021.112871
Wei, S.Y., Jin, X.W. and Li, H. (2019), "General solutions for nonlinear differential equations: a rule-based self-learning approach using deep reinforcement learning", *Computat. Mech*., **64**(5), 1361-1374. https://doi.org/10.1007/s00466-019-01715-1
Wu, W.H., Jhou, J.W., Chen, C.C. and Lai, G. (2019), "A novel recursive stochastic subspace identification algorithm with its application in long-term structural health monitoring of office buildings", *Smart Struct. Syst., Int. J*., **24**(4), 459-474. https://doi.org/10.12989/sss.2019.24.4.459
Xu, C., Zhang, D., Chong, J., Chen, B. and Li, S. (2021), "Synthesis of gadolinium-enhanced liver tumors on nonenhanced liver MR images using pixel-level graph reinforcement learning", *Med. Image Anal*., **69**, p. 101976. https://doi.org/10.1016/j.media.2021.101976
Yang, C. and Lu, Z.X. (2017), "An interval effective independence method for optimal sensor placement based on non-probabilistic approach", *Inst. Solid Mech*., **60**(2), 186-198. https://doi.org/10.1007/s11431-016-0526-9
Yi, T.H., Li, H.N. and Zhang, X.D. (2015), "Health monitoring sensor placement optimization for Canton Tower using immune monkey algorithm", *Struct. Control Health Monitor*., **22**(1), 123-138. https://doi.org/10.1002/stc.1664
Yin, T., Yuen, K.V., Lam, H.F. and Zhu, H. (2017), "Entropy-based optimal sensor placement for model identification of periodic structures endowed with bolted joints", *Comput.-Aided Civil Infrastr. Eng*., **32**(12), 1007-1024. https://doi.org/10.1111/mice.12309
Yuen, K.V., Beck, J.L. and Katafygiotis, L.S. (2006), "Efficient model updating and health monitoring methodology using incomplete modal data without mode matching", *Struct. Control Health Monitor*., **13**, 91-107. https://doi.org/10.1002/stc.144
Yurtsever, E., Capito, L., Redmill, K. and Ozgune, U. (2020), "Information-driven distributed maximum likelihood estimation based on Gauss-Newton method in wireless sensor networks", In: *2020 IEEE Intelligent Vehicles Symposium* (*IV*), pp. 1311-1316. https://doi.org/10.1109/IV47402.2020.9304735
Zhao, T. and Nehorai, A. (2007), "Information-driven distributed maximum likelihood estimation based on Gauss-Newton method in wireless sensor networks", *IEEE Transact. Signal Process*., **55**(9), 4669-4682. https://doi.org/10.1109/TSP.2007.896267
Zhao, T., Wang, P. and Li, S. (2019), "Traffic signal control with deep reinforcement learning", *Proceedings of 2019 International Conference on Intelligent Computing, Automation and Systems* (*ICICAS*), Chongqing, China, December. https://doi.org/10.1109/ICICAS48597.2019.00164

BS