Structural and Multidisciplinary Optimization (2026) 69:51
https://doi.org/10.1007/s00158-025-04244-z

RESEARCH PAPER

An efficient reinforcement learning action strategy for topology optimization: application to muffler design

Kee Seung Oh[1] · Yoon Young Kim[2,3] · Hayoung Chung[4] · Joo Hwan Oh[5]

Received: 24 October 2025 / Revised: 21 December 2025 / Accepted: 28 December 2025

Abstract

Despite the growing interest in applying reinforcement learning (RL) to design optimization, its high computational cost limits its applicability to problems involving expensive function evaluations. In this study, we propose an efficient RL action strategy specifically designed for acoustic topology optimization. The key idea is to assign action values (Q-values) to each element individually and select material-filled elements in descending order of their Q-values until the target volume fraction is met, instead of evaluating Q-values for complete combinations of elements that satisfy the volume constraint. This formulation decouples the learning complexity from the combinatorial explosion of candidate layouts, making the training of the Q-value-estimating neural network more efficient and thus the RL-based approach is more suitable for topology optimization problems requiring fine meshes. As a representative application, we consider the design of a muffler's internal layout to maximize sound transmission loss—a problem where conventional gradient-based methods often fail to achieve near-global optimal solutions. By integrating the proposed method with finite element simulations and a reward function shaped by transmission loss at one or more target frequencies, the RL agent learns policies that directly determine the material distribution for single- or multi-frequency objectives. The resulting muffler designs, based on a two-dimensional finite element model, exhibit near-global optimal performance and outperform those generated by conventional gradient-based methods. The advantages of the proposed approach over standard RL-based topology optimization methods are also clearly demonstrated.

Keywords Reinforcement learning · Element-wise Q-value evaluation · Topology optimization · Muffler design · Noise reduction

1 Introduction

Recently, there has been growing interest in artificial intelligence (AI)-based approaches across various fields, including structural topology optimization (Oh et al. 2019; Sosnovik and Oseledets 2019; Yu et al. 2019; Kollmann et al. 2020; Chandrasekhar and Suresh 2021; Chi et al. 2021; Mukherjee et al. 2021; Nie et al. 2021; Yan et al. 2022). Among various AI-based methodologies, reinforcement learning (RL) has received particular attention for topology optimization because it does not require extensive pre-generated datasets. Hayashi and Ohsaki (2020) introduced a novel framework that combines RL with graph embedding for binary truss topology optimization under stress and displacement constraints, in which the RL agent sequentially removes structural members from a densely connected ground structure. Brown et al. (2022) proposed a deep RL framework to design elementally discretized two-dimensional (2D) topologies by

Responsible Editor: Liwei Wang.

✉ Joo Hwan Oh
ojh86@snu.ac.kr

[1] Institute of Advanced Machines and Design, Seoul National University, 1 Gwanak-Ro, Gwanak-Gu, Seoul 08826, Republic of Korea

[2] Department of Mechanical Systems, Sookmyung Women's University, 100 Cheongpa-Ro 47-Gil, Yongsan-Gu, Seoul 04310, Republic of Korea

[3] IdeAOcean Inc., 1861 Nambusunhwan-Ro, Gwanak-Gu, Seoul 08826, Republic of Korea

[4] Department of Mechanical Engineering, Ulsan National Institute of Science and Technology, UNIST-Gil 50, Eonyang-eup, Ulju-gun, Ulsan 44919, Republic of Korea

[5] Department of Mechanical Engineering, Seoul National University, 1 Gwanak-Ro, Gwanak-Gu, Seoul 08826, Republic of Korea

Published online: 19 February 2026

formulating the structural layout problem as a Markov decision process (MDP). Their method provides a gradient-free and data-independent optimization framework, enabling progressive refinement and leveraging a convolutional neural network-based Q-function to evaluate compliance-minimizing design policies across arbitrary boundaries and loading conditions. Recently, Shin and Yoon (2025) explored acoustic topology optimization with a Double Deep Q-Network (DDQN) agent, highlighting the promise of reinforcement learning (RL) in this field. As a broader perspective on AI-driven inverse design and global search strategies in wave-based (acoustic/phononic) systems, recent review on inverse design of phononic meta-structured materials by Dong et al. (2024) is worth to be referred. These studies complement classical global optimization paradigms (e.g., evolutionary algorithms) and highlight the growing role of data-driven and learning-based approaches in acoustic design.

Despite recent advances, RL-based topology optimization remains computationally demanding, particularly for problems requiring fine finite element meshes, which poses a major barrier to practical deployment. While RL can in principle overcome limitations of gradient-based approaches and automate the design loop, its high computational cost has limited scalability. To address this challenge, we propose a more efficient RL-based topology optimization method that introduces a novel action strategy to markedly reduce the computational complexity of the neural network used to compute action values (Q-values), thereby accelerating training and enabling fine discretizations and multi-frequency objectives.

As a representative design problem, we consider the optimal design of a muffler's internal layout to maximize sound transmission loss at one or more target frequencies as shown in Fig. 1. This problem was selected not only because of our prior experience with acoustic design problems but also because conventional gradient-based methods often struggle to produce near-global optimal solutions. The muffler design problem has been extensively studied in the literature (Munjal 1987; Selamet and Ji 1999; Xu et al. 2004; Denia et al. 2007). Topology optimization of mufflers has also been investigated (Lee and Kim 2009; Lee 2015; Yedeg et al. 2016; Oh and Lee 2017, 2023; Ferrándiz et al. 2020; Lee et al. 2020). While gradient-based methods are efficient for finding local optima, they often have difficulty identifying near-global optimal solutions due to the presence of many local minima (Allaire et al. 1997; Rozvany 2001), and this issue becomes more severe when broadband frequencies are considered (Oh and Lee 2023). Although the gray-element problem has been addressed through filtering techniques (Sigmund 2007), selecting appropriate filters remains challenging and problem-dependent.

Because conventional gradient-based methods often become trapped in local minima for muffler design, RL-based topology optimization offers a promising alternative (Shin and Yoon 2025). However, prior RL approaches remain severely constrained by the aforementioned computational cost. To illustrate the underlying concept of our approach, we first consider a simplified topology optimization problem defined on a design domain discretized into only six finite elements, as shown in Fig. 2a. Without loss of generality, the volume fraction is set to 0.5, meaning that three of the six elements must be filled with material during each RL iteration.

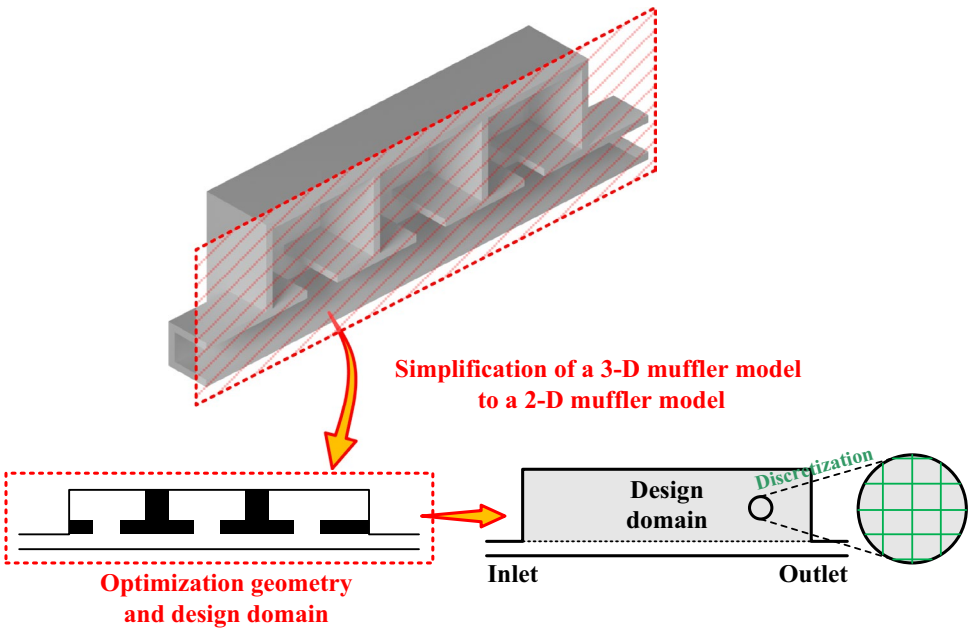

Fig. 1 Simplified two-dimensional muffler model for topological design

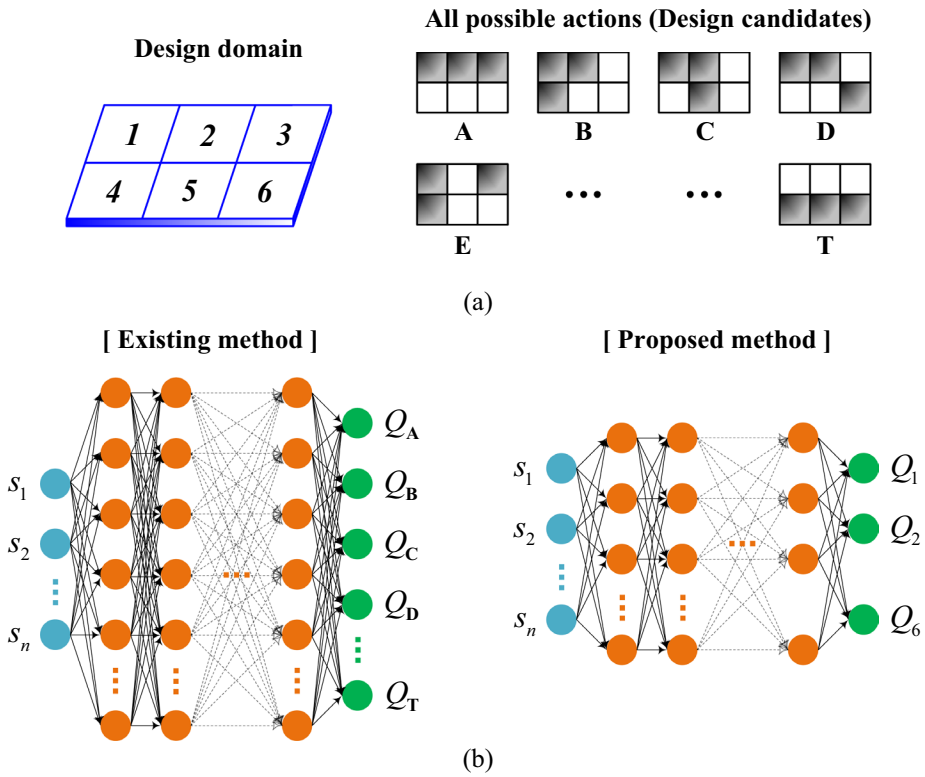

Fig. 2 illustrative example demonstrating the advantages of the proposed method. **a** (left) Design domain discretized into six finite elements and (right) several of the 20 possible actions considered in the conventional RL-based topology optimization under a 50% volume constraint. **b** Neural network architectures for computing Q-values using the (left) conventional and (right) proposed action strategies. In this sample problem, T = 20. The conventional strategy requires 20 output nodes, whereas the proposed strategy requires only 6, corresponding to the number of finite elements

In the existing action strategy (see, e.g., (Hayashi and Ohsaki 2020; Brown et al. 2022)), the Q-values are computed for all 20 possible (some of them are shown in Fig. 2a) actions representing different material distributions that satisfy the volume constraint, and the agent then selects the action with the highest Q-value. Shin and Yoon (2025) employed a Double Deep Q-Network (DDQN) agent that sequentially added or removed individual finite elements during the optimization process. As mesh resolution increases, the number of possible actions grows combinatorially, requiring Q-value estimation for all actions; consequently, applying existing methods to topology optimization with fine meshes (e.g., muffler problems) is practically infeasible.

In contrast, the proposed strategy estimates Q-values for individual elements rather than for every possible material distribution satisfying the volume constraint; only six Q-values, corresponding to the six elements, are computed. The agent then selects the three elements with the highest Q-values for material placement. Since the Q-values are generated by a neural network, the proposed strategy requires a much smaller number of output nodes than the conventional strategy, as illustrated in Fig. 2b. This results in a more compact network and significantly reduces the training cost. The advantage of this approach becomes even more pronounced as the number of finite elements increases, making the proposed method highly scalable and efficient. The detailed algorithm and performance evaluation are presented in the following sections.

As an alternative to the reinforcement learning (RL) approach considered here, one could adopt an end-to-end supervised or deep generative inverse-design model. Such approaches, however, typically require a large offline dataset of high-quality (near-optimal) designs. Constructing this dataset often entails extensive search procedures for a large number of training samples and can therefore impose a computational cost that exceeds what is suggested by simply counting the number of finite element method (FEM) evaluations. In contrast, the proposed RL framework learns online from state transitions generated during the optimization process and does not rely on a pre-collected database of optimal designs.

The remainder of this paper is organized as follows. Section 2 specifies the Markov decision process (MDP) that underpins our RL framework, including the definitions of states, actions, and rewards in the context of topology optimization, as well as the Q-learning procedure and the neural network used to estimate Q-values. Section 3 presents optimized muffler layouts designed to maximize sound transmission loss at target frequencies. Section 4 discusses the results, including comparisons with gradient-based methods, analyses of the optimized layouts, and convergence behavior. Finally, Sect. 5 concludes with a summary of the key findings and directions for future research.

2 Reinforcement learning

2.1 Brief overview of RL-based topology optimization

The fundamental RL method relies on interactions between the agent and the environment through a sequence of observations, actions, and rewards as sketched in Fig. 3. In the RL implementation for topology optimization (TO) (see, Hayashi and Ohsaki 2020; Brown et al. 2022), the action **a** determines the current design layout. For instance, **a** can be expressed using the design variable vector whose elements consist of 0 and 1 representing void and material-filled element states or the list of the element numbers that should be filled with a given material. For a given action, finite element simulation is performed and the state s is defined using the simulation result. In addition, a scalar reward r is evaluated from the simulation result for the selected objective function to be maximized.

To guide the agent to make an optimal action yielding the maximum reward, the agent needs to check the expected reward for each possible action, which is called the action value (Q-value). Then, the agent is trained to predict Q as accurately as possible. After sufficient training, the agent will be able to predict which action (design layout) provides the maximum reward (objective function value) by considering the action with the highest Q-value.

Following the RL-based TO framework in Hayashi and Ohsaki (2020) and Brown et al. (2022), a single with the MDP chain is described in Fig. 3. In a single episode, the agent makes actions and states and rewards are evaluated accordingly. Now, we assume that the agent performs actions for T times where the t-th time step consists of action a_t, state s_t, and reward r_t. In RL, the Q-value is typically defined as the maximum expected value of the cumulative future reward (Sutton and Barto 2018). If the agent is assumed to be the best agent and always make the best action with the maximum reward, the Q-value measured from a t-time attempt is

$$Q^*_{\text{opt}}\Big|_t = \max_{\pi} \mathbb{E}\left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots + \gamma^{T-t} r_T \Big| \pi \right] \tag{1}$$

where Q_{opt} is the optimum Q-value, and γ is the discount factor to facilitate the RL process (generally, $\gamma \leq 1$). Here, the superscript * is introduced to indicate that Q^* is a theoretical value, which is achievable only when the rewards for all possible actions at every time step are known. In Eq. (1), it is assumed that the best agent always makes the best action, but the actual agent cannot. So, we define the following Q-value for each action at th t-th attempt, which will be used to guide the agent to make the best action:

$$Q^*|_t(a) = \mathbb{E}\left[r + \gamma Q^*_{\text{opt}}\Big|_{t+1} \Big| s, a \right]. \tag{2}$$

The Q^*-value defined in Eq. (2) is a function of action a. (Here, state s is also a function of a, but this will be ignored for now.) For each possible action, the corresponding Q-value can be defined. Among them, the action with the largest Q^* is declared as the best action.

For a clear understanding, let us consider time steps having two different possible actions, action A providing 10 rewards and action B providing 5 rewards. Therefore, the reward for every time step is $r_t = 10$ or 5. Assuming $T = 3$ (3 time steps) and $\gamma = 1$ (no discount effect), the action values at the first time step are defined as

$$\begin{aligned} Q^*|_{t=1}(A) &= r_1|_A + Q^*_{\text{opt}}\Big|_{t=2} \\ &= 10 + \max\left(r_2\right) + \max\left(r_3\right) = 30, \end{aligned} \tag{3a}$$

$$\begin{aligned} Q^*|_{t=1}(B) &= r_1|_B + Q^*_{\text{opt}}\Big|_{t=2} \\ &= 5 + \max\left(r_2\right) + \max\left(r_3\right) = 25. \end{aligned} \tag{3b}$$

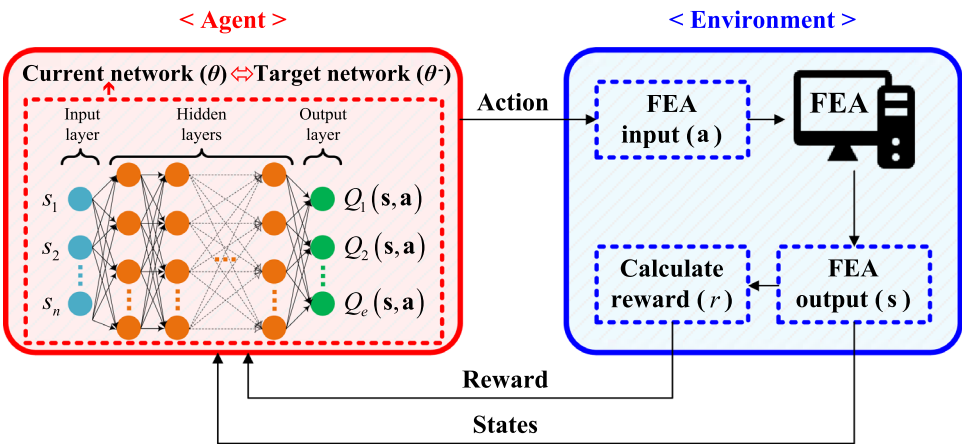

Fig. 3 Markov decision process for the reinforcement learning-based topology optimization process

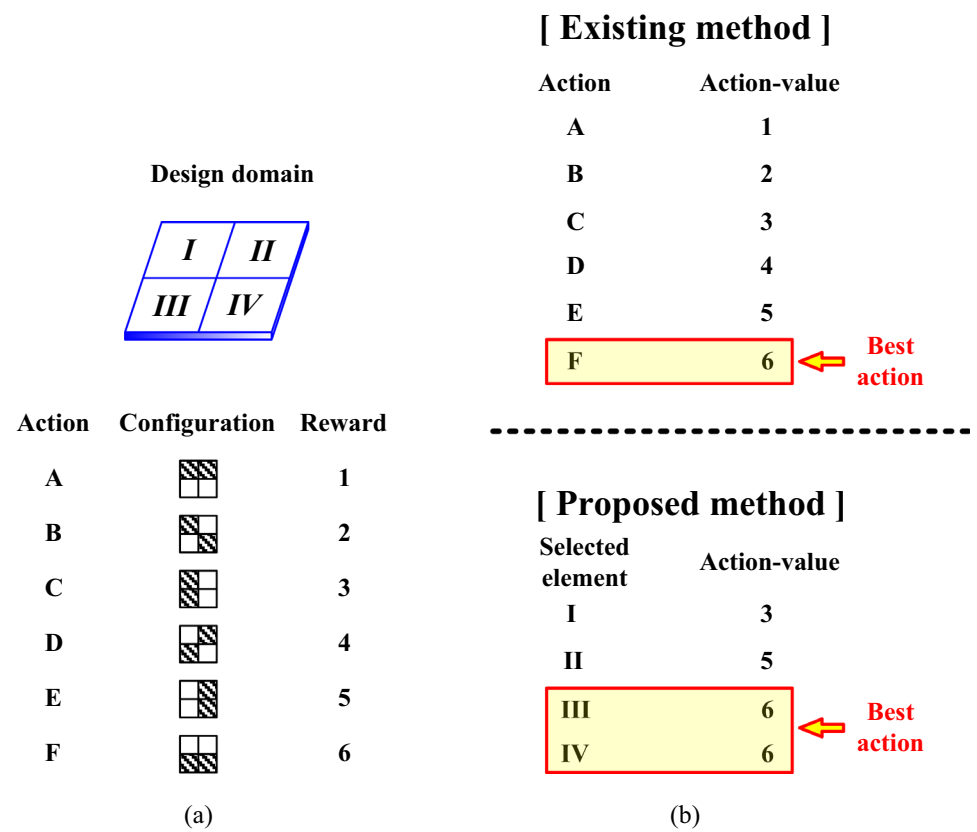

Fig. 4 A simple example to compare the action-wise and element-wise rewards. **a** The table for all possible actions and their rewards. **b** Action value (Q-value) comparison depending on the existing and proposed methods

Equation 3 shows that action A, which has the larger Q^* value, is the best action. Unfortunately, the theoretical Q^*-value is almost impossible to calculate because it requires optimal rewards for all actions in all time steps. Thus, an artificial neural network (ANN) is introduced to predict the action value Q as

$$Q(s,a;\theta) \cong Q^*(s,a), \tag{4}$$

where θ represents the weights in the ANN. Any change in θ adjusts the ANN, resulting in the change in Q so that $Q \sim Q^*$. Thus, there should be proper learning to find the best θ that can predict Q^* as accurately as possible. This is done by minimizing the following loss function with respect to θ_j as (Mnih et al. 2015)

$$L_j^{\mathrm{DQN}}(\theta_j) = \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim U(D)} \left[\left(r + \gamma \max_{a_{t+1}} \hat{Q}\left(s_{t+1}, a_{t+1}; \theta_j^-\right) - Q\left(s_t, a_t; \theta_j\right) \right)^2 \right] \tag{5}$$

in which $\left(s_t, a_t, r_t, s_{t+1}\right) \sim U(D)$ represents uniform random sampling from stored samples in a dataset $D_t = \left\{d_1, d_2, ..., d_t\right\}$, where $d_t = \left(s_t, a_t, r_t, s_{t+1}\right)$ performing experience replay at each time step t. The Q-learning process can be updated based on mini-batches of experience from the stored samples at iteration j denoted in Eq. (5). The hat notation ($\hat{\bullet}$) represents the target replacement for the temporarily fixed network θ^-; the loss function L can be calculated by optimizing with a stochastic optimizer rather than computing the full expectations.

2.2 Proposed RL-based TO

Section 2.1 provides a brief introduction to RL-based TO. In the conventional RL-based TO, Q-values must be calculated for all possible combinations of elements that fill the design domain while satisfying a given volume constraint, as illustrated in Fig. 2 and discussed in the Introduction. The main challenge with this approach lies in its combinatorial nature: as the number of finite elements used to discretize the domain increases, the number of combinations—and thus the number of output nodes in the ANN—grows exponentially. This, in turn, requires a greater number of hidden-layer nodes, leading to significantly higher training costs. Consequently, this conventional action strategy becomes impractical for topology optimization problems involving fine meshes.

To address this issue, we propose a new action strategy aimed at reducing the computational burden. In the conventional strategy, Q-values are assigned to combinations of finite elements that satisfy the volume constraint, and the combination with the highest Q-value is selected as the next action. In contrast, our strategy assigns Q-values individually to each finite element rather than to

their combinations. The material-filled elements are then selected in descending order of their Q-values until the specified number, determined by the target volume fraction, is reached.

To illustrate our approach, consider the example shown in Fig. 4, where the design domain is discretized into four elements with a volume fraction of 0.5. In this case, there are six possible actions. Assume that the reward for each action is evaluated as shown in Fig. 4a, and consider a single time step, $T=1$. Since $T=1$, the Q-values are equivalent to the rewards, and thus the two terms can be used interchangeably. In this situation, the agent selects the action 'F,' which has the highest action value, as illustrated in Fig. 4b. While this decision-making process is straightforward, the computational cost of evaluating Q-values becomes prohibitive as the number of elements increases.

In our proposed strategy, Q-values are not assigned to every possible combination of elements that satisfy the volume constraint. Instead, they are assigned to each element individually. Thus, in this example, only four Q-values are evaluated rather than six. The detailed procedure for assigning Q-values to individual elements will be explained later in this section. Assuming these Q-values are obtained as shown at the bottom of Fig. 4b, the next action is to select and fill two elements—III and IV—in descending order of their Q-values, because the volume constraint is 50% of the four available elements. Although in this small example the number of Q-value evaluations is reduced only from six to four, the computational savings become significant as the number of domain-discretizing finite elements increases.

Before defining the action, state, and reward in detail and presenting the mathematical formulation of the Q-value, it would be better to introduce the overall process of the proposed method. Algorithm 1 summarizes the workflow overview of the proposed reinforcement-learning-based topology optimization framework.

Algorithm 1 Workflow overview of the proposed reinforcement-learning-based topology optimization framework

- Initialize the reinforcement learning agent, including the value networks and the experience replay memory.
- Repeat the following procedure for each design episode.
  - · Reset the environment and build the current state description of the design domain.
  - · Predict an action-value (Q-value) score for every finite element in the design domain.
  - · Choose whether the agent will take explorative or exploitive action based on the predefined probability.
    - — When the agent takes explorative action, choose the predefined number of elements randomly.
    - — When the agent takes exploitive action, choose the predefined number elements by choosing the highest predicted action-value (Q-value) scores.
  - · Construct the design layout by filling solid material to the selected elements while remaining elements are set to be air.
  - · Carry out the acoustic finite element analysis for the design layout and compute the objective value (based on the transmission-loss) over the target frequency range.
  - · Calculate the scalar reward value for the design layout.
  - · Store the transition data in the replay memory.
  - · Update the value network using mini-batches sampled from the replay memory, and periodically synchronize the target network.
  - · Update the exploration probability according to the scheduled decay rule and record the best layout found so far.
- Provide the best-performing design layout across all episodes as the output.

2.2.1 Action by agent and *Q*-values

In the proposed RL-based TO, the action of the agent is defined as choosing a given number E_0 of material-filled elements from the E domain-discretizing finite elements. This action $\mathbf{a}$ is expressed by the following vector:

$$\mathbf{a} = \left\{a_1, a_2, ..., a_e, ..., a_E\right\}^T \tag{6a}$$

$$a_e = \begin{cases} 1 & \text{for the designated element by the agnet} \\ 0 & \text{otherwise} \end{cases} \tag{6b}$$

With a, E_0 elements will be filled with a given material. It should be emphasized that the way the action is defined in the proposed method is fundamentally different from that employed in typical RL approaches that define the action as a sequence of certain choices (e.g., the cardinal direction). Although the selection of a_e is sequential, there is no sequence in filling the selected elements with the given material; only which elements are filled with the material matter. This definition of the action substantially alleviates the computational burden by ensuring that the number of design variables remains within a computationally tractable range. We will show that if the agent is sufficiently trained, then a topological layout with high performance can be obtained.

In the proposed approach, the action is to select E_0 elements in the descending order of the Q-values of all finite elements. To facilitate this selection operation, the 'find' operator is used as

$$\text{find}_{E_0}\vec{Q} := \max\left\{\vec{Q}\middle|\text{for } \forall e \in \mathbb{N} : Q_e \supseteq Q_{e+1} \wedge e \leq E_0 < E\right\}, \tag{7}$$

where $\mathbb{N}$ denotes a positive integer space and $\vec{Q}$ denotes the Q-value stored in descending order. Here, $\text{find}_{E_0}\vec{Q}$ denotes a top-E_0 selection operator: it returns the E_0 largest Q-values among the Q-values of all elements.

As is common in reinforcement learning, the ε-greedy method was used to balance exploration of new design choices and exploitation of the current policy; a random action is chosen with probability ε, while the best-known action is taken with probability $1-\varepsilon$. If the agent follows the exploratory strategy, it selects elements randomly, independent of the Q-value. In contrast, under the greedy strategy, it selects the elements with the highest Q-values. This policy π for multiple-element selection can be expressed as follows:

$$\pi\left\langle \begin{array}{c} \text{Exploration}(\varepsilon) \Rightarrow \mathbf{a} := \text{choose 0 - 1 binary random action considering } E_0 \\ \text{Exploitation}(1-\varepsilon) \Rightarrow \mathbf{a} := \begin{cases} 1 & \text{for } e \text{ - th element where } e = \text{argfind}_{E_0}\vec{Q} \\ 0 & \text{otherwise} \end{cases} \end{array} \right., \tag{8}$$

where

$$\text{argfind}_{E_0}\vec{Q} := \max\left\{e|\text{for } \forall e \in \mathbb{N} : Q_e \supseteq Q_{e+1} \wedge e \leq E_0 < E\right\} \tag{9}$$

in which 'argfind' is a new operation to find the indices for the designated elements based on the Q-values, similar to the 'find' operation previously defined: while 'find' operator returns the E_0 largest Q-values among the Q-values of all elements, $\text{argfind}_{E_0}\vec{Q}$ returns the corresponding element's index values for the E_0 largest Q-values. Here, the value of ε is initially set to be a relatively large value and gradually decreased as the learning progresses. Thus, at the beginning of learning, the agent favors exploration to enhance the effectiveness of training. As learning progresses and the agent has acquired sufficient knowledge, the estimated Q-values become more reliable, and ε is gradually reduced so that the agent's actions are guided primarily by the Q-values. As explained in the Introduction, the proposed single-step action assigns an action value to each element. Compared with prior RL-based TO methods, this scheme yields far fewer action values, which is advantageous for acoustic problems that require very large meshes to resolve wave dynamics. Nonetheless, single-step actions can raise convergence concerns because the optimization is not performed sequentially and thus lacks an explicit guidance mechanism for the evolving design. Conversely, the absence of sequential updates can reduce sensitivity to local minima. The limitations and benefits of the single-step selection strategy are examined in Appendix A.

2.2.2 State from observation of environment

As illustrated in Fig. 3, we define the state as the environment's response to the selected action. Within our RL-based TO formulation, the environment corresponds to a numerical simulation of the target system, and the action encodes the proposed geometry; thus, the state is the resulting simulation output. While no strict rule dictates how to choose the state among available simulation variables, because the state is an input to the action–value estimator, it is preferable—on optimization grounds—to select a quantity directly related to the reward. Accordingly, we define the state as the performance metric of the geometry induced by the action, as

$$s = g(\mathbf{a}), \tag{10}$$

where g denotes the performance function. For the acoustic muffler problem, performance is typically quantified by transmission loss (TL) at a target frequency, computed via standard numerical analysis. Accordingly, we define the state as the TL at the target frequency produced by a given action.

Because acoustic mufflers are typically designed for broad frequency bands, performance is evaluated at multiple frequencies. For a target range $[f_1, f_N]$ with $f_1 < f_2 < \cdots < f_N$, performance at each f_n is obtained by repeating the

numerical analysis. Consequently, the state is defined as a vector that captures multi-frequency performance. Let $g_{f_n}(\mathbf{a})$ denote the performance (e.g., transmission loss) of action $\mathbf{a}$ (the material distribution) at frequency f_n. The MDP state used for optimization is then

$$\mathbf{s} = \{s_1, s_2, ..., s_n, ..., s_N\} = \{g_{f_1}(\mathbf{a}), g_{f_2}(\mathbf{a}), ..., g_{f_n}(\mathbf{a}), ..., g_{f_N}(\mathbf{a})\}, \quad (11)$$

where N denotes the number of frequencies considered, and n is the index for each target frequency.

To avoid confusion, we emphasize that the proposed method employs a non-sequential action. Accordingly, the state at each frequency is defined once for each action. In standard RL settings, actions proceed sequentially, requiring the state to be recalculated at every step. Here, however, a single action is executed—namely, all masses (black elements) are placed simultaneously—so no action history exists and the state for each target frequency is evaluated only once per action.

2.2.3 Reward from observation

As shown in prior RL research (Ng et al. 1999), careful reward design r for a given state s is essential for evaluating actions and learning effective policies. In many benchmark domains—such as Atari—rewards are often simple and sparse (e.g., +1 for a win and -1 for a loss; Mnih et al. 2013, 2015) owing to the clear episodic structure from start to finish. This simplicity enables effective reward-shaping strategies and has been validated across numerous studies (Kulkarni et al. 2016; Silver et al. 2016; Lample and Chaplot 2017). In topology optimization, however, there is no unambiguous notion of 'win' or 'loss.' For example, if one action achieves 30% of the performance of an ideal reference (e.g., a muffler that provides 100% noise reduction across all frequencies) while another achieves 95%, the rewards must differentiate these outcomes. Accordingly, the reward should be graded to promote higher performance, while avoiding excessively large values that can saturate learning signals or induce premature convergence.

Thus, we define the reward as a function of the state as

$$r_n = \left(\frac{s'_n - s_{worst}}{s_{tar} - s_{worst}} \right)^q \quad (12a)$$

$$r = \prod_{n=1}^{N} r_n, \quad (12b)$$

where r_n is the n-th reward corresponding to the n-th target frequency, and s'_n is the state of the n-th target frequency after an action is performed by the agent. The reward shaping consists of an 'objective target s_{tar}' and 'worst case baseline s_{worst}' to set the range of the achievable objective, and a 'penalization parameter q' to make the rewarding not too generous and efficiently teach the agent. The reward provided to the agent must be a scalar; therefore, the per-frequency rewards are aggregated multiplicatively, as in Eq. (12b). In conventional TO, the objective is often a weighted sum of r_n with iteration-dependent weights. Because the present RL-based TO uses a single-step optimization, that weighted-sum scheme is not directly applicable. We thus adopt the multiplicative aggregation in Eq. (12b), although other aggregation forms could also be used.

In Eq. (12a), the quantities s_{tar} and s_{worst} are not rigorously defined in the context of acoustic muffler design. For mathematical test functions with well-specified domain and codomain (e.g., the Rastrigin function), both the worst-case baseline and the target value can be determined explicitly. In practical muffler design, however, such explicit reference values are generally unavailable. Therefore, s_{tar} and s_{worst} for reward shaping are chosen heuristically, informed by the designer's experience and domain knowledge. In the present muffler design problem, s_{worst} is fixed to zero because the transmission loss is nonnegative by definition. Therefore, the primary reward shaping is mainly governed by how to select s_{tar}, which will be investigated later with actual design results.

Unlike the predetermined toy example shown in Fig. 4 which was crafted purely for explanatory purposes, the scalar reward r in our formulation is employed to update the Q-value. Because Q is defined on a per-element basis, the reward should be distributed only to those elements selected by the current action. Let E denote the number of discretized elements, and let $\mathbf{a} \in \{0, 1\}^E$ be the selection vector ($a_e = 1$ if element e is chosen). The resulting per-element reward vector of size E is defined as

$$\mathbf{r} = r\mathbf{a} \quad (13a)$$

$$r_e = r a_e \ (e = 1, ..., E). \quad (13b)$$

Using the reward defined in Eq. (13), the Q-values representing the qualities of each element are updated by the Q-learning method, as described in the following subsection.

2.2.4 Loss function with double deep Q-network

As explained, the Q-value is defined for each element. From the definition of the Q-value, its optimal can be written with the populated reward $\mathbf{r}$ in Eq. (13) as

$$Q^*(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\mathbf{s}'} \left[\mathbf{r} + \gamma \operatorname{find}_{\mathbf{a}'} Q^*(\mathbf{s}', \mathbf{a}') \middle| \mathbf{s}, \mathbf{a} \right], \quad (14)$$

where the prime symbol $(\cdot')$ denotes the next time step. Considering previously explained Eq. (4), the function

approximator (ANN) should be introduced, and the loss function should be re-defined. We used the modified DDQN method (Van Hasselt et al. 2016) for TO considering the preliminary defined MDP in Eqs. (6–13) as follows:

$$L_j(\theta_j)=\mathbb{E}_{(\mathbf{s},\mathbf{a},\mathbf{r},\mathbf{s}')\sim U(D)}\left[\left(\mathbf{r}+\gamma \operatorname*{find}_{\mathbf{a}'} E_0\hat{\vec{Q}}\left(\mathbf{s}',\operatorname*{argfind}_{\mathbf{a}} E_0\vec{Q}(\mathbf{s}',\mathbf{a};\theta_j);\theta_j^-\right)-\vec{Q}(\mathbf{s},\mathbf{a};\theta_j)\right)^{\circ 2}\right], \tag{15}$$

where the circle (∘) in the superscript denotes the Hadamard product (Horn and Johnson 2012). In addition, the termination of RL from an optimization perspective should be set if

$$s_{\text{tar}} \le s_m^{\text{avg}}(\mathbf{a}_m) \tag{16a}$$

$$\mathbf{a}_m=\begin{cases}1 & \text{for } e\text{ - th node or element by argfind}_{E_0}\vec{Q}(\mathbf{s}_{\text{tar}};\theta_m)\\ 0 & \text{otherwise}\end{cases}, \tag{16b}$$

where s_m^{avg} represents the average state values at the m-th episode calculated by $s_m^{\text{avg}}(\mathbf{a}_m)=\frac{1}{N}\sum_{n=1}^{N} s_n(\mathbf{a}_m)$. $\mathbf{s}_{\text{tar}}$ in Eq. (16b) is obtained by populating s_{tar} with the corresponding number of states (N) so that the vectorized objective targets can be used as the test set to determine the termination. The entire Q-learning process used in this study is presented in Algorithm 2 as a pseudoalgorithm.

Algorithm 2 Reinforcement learning (RL)-based topology optimization (TO)

1: **For** $m = 1, M$ **do** (m: episode)

2: **For** $t = 1, T$ **do**

3: With probability ε select a random $\mathbf{a}_t$, otherwise select

$$\mathbf{a}_t=\begin{cases}1 & \text{for } e\text{-th node or element where } e=\text{argfind}_{E_0}\vec{Q}\\ 0 & \text{otherwise}\end{cases}$$

4: Execute action $\mathbf{a}_t$ for function or functional evaluation and observe reward $\mathbf{r}_t$

5: Set $\mathbf{s}_{t+1}=\mathbf{s}_t,\mathbf{a}_t$

6: Store transition $(\mathbf{s}_t,\mathbf{a}_t,\mathbf{r}_t,\mathbf{s}_{t+1})$ in D

7: Sample random minibatch of transition $(\mathbf{s}_j,\mathbf{a}_j,\mathbf{r}_j,\mathbf{s}_{j+1})$ from D

8: Set $\mathbf{y}_j=\begin{cases}\mathbf{r}_j & \text{if episode terminates at step } j+1\\ \mathbf{r}_j+\gamma \operatorname*{find}_{\mathbf{a}'} E_0\hat{\vec{Q}}\left(\mathbf{s}'_{j+1},\operatorname*{argfind}_{\mathbf{a}} E_0\vec{Q}(\mathbf{s}'_{j+1},\mathbf{a}_j;\theta);\theta^-\right) & \text{for otherwise}\end{cases}$

9: Perform a gradient descent step on $\left(\mathbf{y}_j-\vec{Q}(\mathbf{s}_j,\mathbf{a}_j;\theta)\right)^{\circ 2}$ w.r.t. the network parameter θ

10: Every C steps reset $\hat{\vec{Q}}=\vec{Q}$

11: **End for**

12: If $s_{\text{tar}} \le s_m^{\text{avg}}(\mathbf{a}_m)$ terminates the entire learning process where

$$\mathbf{a}_m=\begin{cases}1 & \text{for } e\text{-th node or element by argfind}_{E_0}\vec{Q}(\mathbf{s}_{\text{tar}};\theta_m)\\ 0 & \text{otherwise}\end{cases}$$

13: **End for**

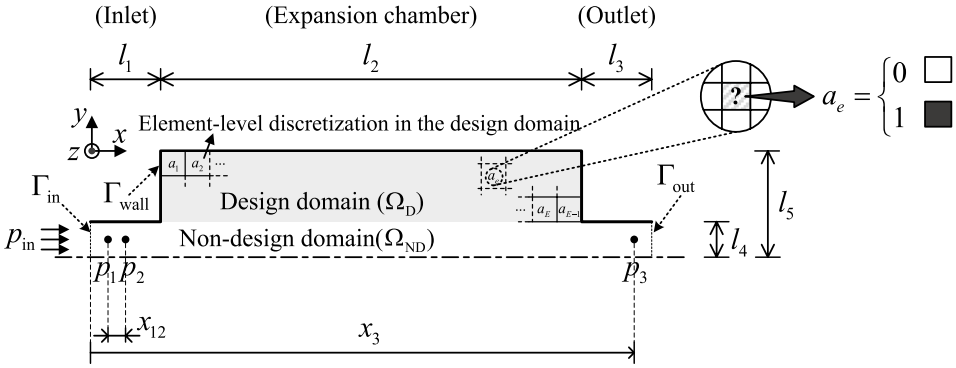

Fig. 5 The problem definition of the acoustic TO of a muffler's internal layout maximizing its TL

In summary, the proposed reinforcement learning-based topology optimization proceeds as follows: (i) According to the policy, the agent places E_0 black finite elements on the design domain either randomly (exploration) or based on the action value Q (exploitation). (ii) A finite element analysis (FEA) is performed to evaluate performance, and a reward is computed. (iii) The reward is used to update the loss in Eq. (15). These steps are repeated episodically. Thus, the classical topology optimization loop can be replaced by a reinforcement learning formulation. Moreover, because the action value is defined per element, the required ANN can be substantially smaller.

3 Acoustic TO problems

Using the proposed method described in Sect. 2, the topology optimization of a muffler's internal layout was carried out. Figure 5 illustrates the problem setting, where the goal is to find a muffler's interior configuration maximizing the transmission loss (TL). When a non-zero mean airflow exists in the muffler, a multiphysics analysis is required to evaluate the TL. In the present study, TL is therefore computed using acoustic analysis under quiescent air conditions in order to simplify the calculation. It is noted, however, that from the perspective of RL-based optimization, the proposed method can be directly extended to cases involving a non-zero mean flow. The design domain (gray region) was discretized into 300 elements. The material properties of the rigid body (regarded as the material filling the finite elements) and the air (filling the void) are given by

$$\rho_e(a_e) = \begin{cases} \rho_{\text{air}} = 1.21\ \text{kg}/\text{m}^3 & \text{for } a_e = 0 \\ \rho_{\text{rigid}} = 10^7 \rho_{\text{air}} & \text{for } a_e = 1, \end{cases} \tag{17}$$

$$B_e(a_e) = \begin{cases} B_{\text{air}} = \rho_{\text{air}} c_{\text{air}}^2 & \text{for } a_e = 0 \ (\text{where } c_{\text{air}} = 343\text{m/s}) \\ B_{\text{rigid}} = 10^9 B_{\text{air}} & \text{for } a_e = 1\,. \end{cases} \tag{18}$$

The selected number (300) of the domain-discretizing finite elements is found be sufficient to precisely capture the involved wave phenomena in the inside layer and to correctly calculate the TL value. The same number of the finite elements is used for other problems considered in this study and all numerical calculations including the TL calculation are performed using the finite element analysis. (More details can be found in Appendix B.)

As the boundary conditions for the problem shown in Fig. 5, the pressure was set as $p_{\text{in}} = 1$ Pa on boundary Γ_{in}, and $\mathbf{u}_{\text{wall}} = \{0, 0\}^T \text{m/s}$ was imposed on boundary Γ_{wall}. For all internal layouts appearing during the topology optimization, the TL was calculated by using the pressures at three points (1, 2, and 3) (Wu and Wan 1996):

$$p_{\text{in}} = \left(p_1(f_n), p_2(f_n)\right) = p_1(f_n) - p_2(f_n) e^{-\text{i}kx_{12}} \tag{19a}$$

$$p_{\text{out}} = \left(p_3(f_n)\right) = p_3(f_n)\left(1 - e^{-2\text{i}kx_{12}}\right) \tag{19b}$$

$$TL(p_1, p_2, p_3; f_n) = 10 \log_{10} \frac{\left|p_{\text{in}}\left(p_1(f_n), p_2(f_n)\right)\right|^2}{\left|p_{\text{out}}\left(p_3(f_n)\right)\right|^2}, \tag{19c}$$

where $\text{i} = \sqrt{-1}$ and the variables p_1, p_2, and p_3 denote the acoustic pressures calculated at the three selected points shown in Fig. 5. Symbol x_{12} represents the distance between two points in the inlet region. The actuating frequency and wave number are denoted by f_n and k, respectively, where $k = 2\pi f_n / c_{\text{air}}$.

Using the analysis method described above, TO was performed using the proposed RL method. As explained, the action $\mathbf{a}$ selects elements to be filled with the rigid material while the unselected ones remain to be filled with air. For each action $\mathbf{a}$, the state is defined by the transmission loss (TL) as

$$\bar{g}_{f_n}^{TL}(\mathbf{a}; f_n) = TL\left(\bar{p}_1(\mathbf{a}; f_n), \bar{p}_2(\mathbf{a}; f_n), \bar{p}_3(\mathbf{a}; f_n)\right) \tag{20a}$$

$$\mathbf{s} = \left\{s_1, s_2, ..., s_n, ..., s_N\right\} = \left\{\overline{g}_{f_1}^{TL}(\mathbf{a};f_n), \overline{g}_{f_2}^{TL}(\mathbf{a};f_n), ..., \overline{g}_{f_n}^{TL}(\mathbf{a};f_n), ..., \overline{g}_{f_N}^{TL}(\mathbf{a};f_n)\right\}. \tag{20b}$$

Note that in the current muffler design problem, the TL values typically vary between 0 and 50. If the state is defined differently, the normalization of TL would be necessary for stable training.

With the state defined in Eq. (20a), the reward can also be calculated with the properly defined s_{tar} and s_{worst}. Because there is no known specific target TL value at the optimum point, the target state (s_{tar}) in Eq. (12) may be arbitrarily selected so that the averaged TL value can be expected to exceed s_{tar} in the frequency range of interest (Recall Eq. (16a)) (In theory, s_{tar} can be found by considering all possible designs, but this is impractical.) The worst state (s_{worst}) is obviously '0' because TL cannot have negative values due to the definition of Eq. (19).

To determine the optimum action for the present RL-based TO of the muffler's internal layout, we implemented an ANN architecture and stochastic optimization for the loss function in Eq. (15) using TensorFlow, and used Adam (Kingma and Ba 2014). We used a fully connected ANN architecture consisting of input, hidden, and output layers, as shown in Fig. 6. The activation functions in the ANN architecture are the rectified linear unit (ReLU). Python (version 3.9.7) was used for all simulations, including reinforcement learning, optimization algorithms, and analysis. TensorFlow version 2.5.0 was used to build the ANN architecture (a.k.a. 'model' in TensorFlow) and Adam was used to update the weights in the ANN. The hardware configuration used in this study is as follows. The CPU is the AMD Ryzen 9 5900X 12-Core Processor (3.70 GHz), the GPU is the Nvidia GeForce GTX 1650, and the RAM capacity is 32 GB.

A summary of the MDP for the muffler design problem is presented as follows. Using the pre-evaluated Q-value defined for each element, the action is defined. Here, the agent performs the action based on the ε-greedy method, as shown in Eq. (8). The action is to select the elements to be filled with the solid material by the agent. According to Eqs. (6–9), the action is represented by E_0 components with a value of 1. For each action, the states are evaluated with the TL values at the target frequencies, and the reward can be obtained from the current states with the arbitrarily selected parameters s_{tar} and s_{worst} from Eq. (12). All variables $\mathbf{a}$, $\mathbf{s}$, and $\mathbf{r}$ are used to further train the agent by updating the ANN as shown in Eqs. (14) and (15), and the updated ANN provides the updated Q-value which will be used to determine the action at the next step. The training process will be repeated until the convergence criterion in Eq. (16) is satisfied. After the training is finished, the agent can provide the optimal action (corresponding to the optimal design). The remaining parameters for the muffler design problem are presented in Tables 1 and 2.

3.1 Maximization of TL at a single target frequency (Case I)

Using the aforementioned procedure, we performed the TL maximization at several single frequencies for the volume ratio of 12% (equivalently, $E_0 = 25$). The three target frequencies are considered: $f = 400$ Hz (Case I-1), $f = 700$ Hz (Case I-2), and $f = 1000$ Hz (Case I-3). Note that for the

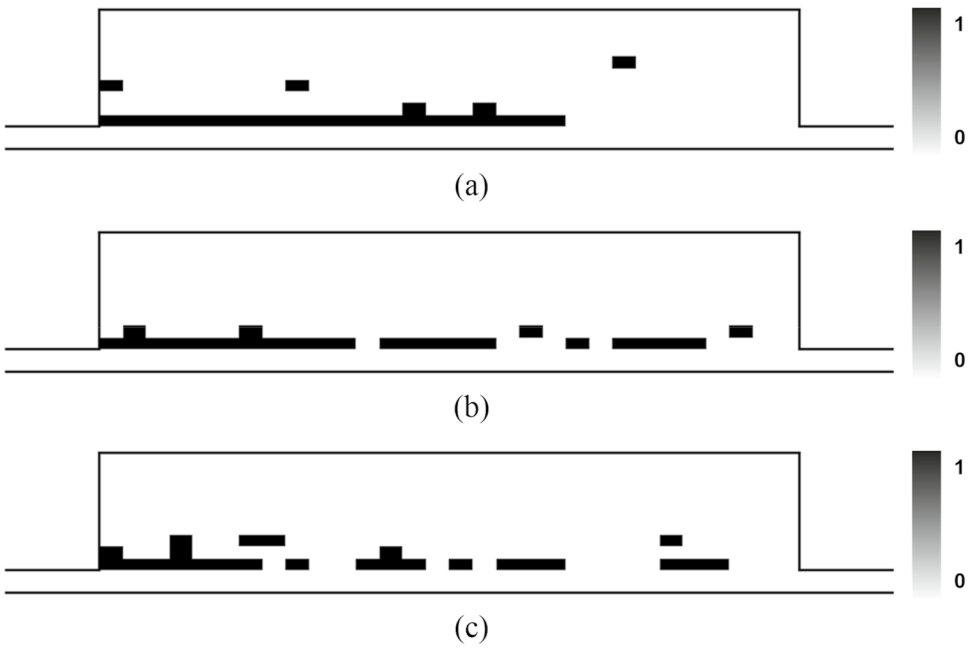

Fig. 6 Optimized layouts by the proposed RL method for single-frequency TL maximization. **a** Case I-1 ($f = 400$ Hz).**b** Case I-2 ($f = 700$ Hz). **c** Case I-3 ($f = 1000$ Hz)

mesh and volume constraint considered here ($E = 300$ and $E_0 = 25$), almost 10^{37} Q-values should be covered as the output if the ANN used in the earlier studies (Hayashi and Ohsaki 2020; Brown et al. 2022) is employed. Training an ANN with about 10^{37} output nodes corresponding to the actions is practically impossible.

We set s_{tar} as $s_{\text{tar}} = 40$, based on the previous research (Lee and Kim 2009; Lee 2015; Oh and Lee 2017; Lee et al. 2020) on noise attenuation in real automotive mufflers. As shown in Eq. (16a), s_{tar} governs the termination of the training process. From a viewpoint of a gradient-based TO, Eq. (16a) may be viewed as a convergence criterion. The convergence condition in the present RL-based method is based on the current value, not on the change of the value. Note that we intentionally choose a high value of s_{tar} in the present study to make the termination condition not limit the overall optimization process and also to make the considered optimization problem very challenging.

Figures 6 and 7 illustrate the optimized internal layouts of the muffler and their corresponding TL curves, respectively. In the present planar muffler model, solid elements are assumed to be attached to a background plate (not shown); in addition to the main partitions, some RL-optimized designs contain isolated rigid elements. Note that those elements are not floating; they are connected to the background wall as in Fig. 1 so that they behave like rigid obstacles. Although such features may appear non-intuitive, they can still affect TL because they behave as rigid obstacles that locally perturb the acoustic field, thereby inducing additional scattering and reflection to modify interference patterns within the chamber. Therefore, even isolated solid elements contribute meaningfully to the transmission loss. As shown in Fig. 7, compared with the nominal layouts without solid material, the TL values of the optimized layouts increased substantially, demonstrating the effectiveness of the proposed approach. Specifically, the TL values at the target frequency reached 53.05 dB for Case I-1, 41.43 dB for Case I-2, and 46.34 dB for Case I-3.

Table 2 Specific dimensions used to solve muffler design problem for the muffler design problem

Symbol	Value
l_1	0.04 m
l_2	0.30 m
l_3	0.04 m
l_4	0.02 m
l_5	0.12 m
x_1	0.01 m
x_{12}	0.01 m
x_3	0.34 m

3.2 Maximization of TL at multiple target frequencies (Case II)

Here, we consider multi-frequency cases to for broadband TL maximization. The frequency range of interest is between 400 and 1400 Hz, and it will be considered at the following 11 discrete frequencies to facilitate analysis: $f_1 = 400$ Hz, $f_2 = 500$ Hz, $f_3 = 600$ Hz, $f_4 = 700$ Hz, $f_5 = 800$ Hz, $f_6 = 900$ Hz, $f_7 = 1000$ Hz, $f_8 = 1100$ Hz, $f_9 = 1200$ Hz, $f_{10} = 1300$ Hz, $f_{11} = 1400$ Hz. Therefore, there are 11 states in Eq. (20b) in this case and 11 neurons in the input layers in Fig. 3. To see the effects of the volume constraints, we consider 3 cases: $E_0 = 30$ for Case II-1, $E_0 = 35$ for Case II-2, and $E_0 = 40$ for Case II-3. (As in Case I, E is set to be 300.) The target state was set to $s_{\text{tar}} = 20$.

Table 1 Hyperparameters used to perform reinforcement learning for the muffler design problem

Hyperparameter	Description	Value
Discount factor (γ)	Discount factor used in Q-learning update process	0.99
Learning rate (α)	Learning rate used by Adam	0.005 (for Cases I-1, I-2, I-3) 0.000005 (for Cases II-1, II-2, II-3)
Initial exploration	Initial value of ε in ε-greedy exploration	1
Final exploration	Final value of ε in ε-greedy exploration	0
Penalization parameter (q)	Penalization parameter to determine curvature of reward function in Eq. (6)	4
Number of hidden layers	Number of hidden layers in ANN considering the fully connected	15
Number of neurons in hidden layer	Number of neurons in hidden layer	300
Minibatch size	Split dataset to train model using stochastic algorithm (Adam)	32
Iterations	Number of iterations to complete single episode	20
Replace target (C)	Frequency with which target network is updated	32
Maximum episodes	Maximum number of episodes to factitiously stop learning if entire process does not converge	200

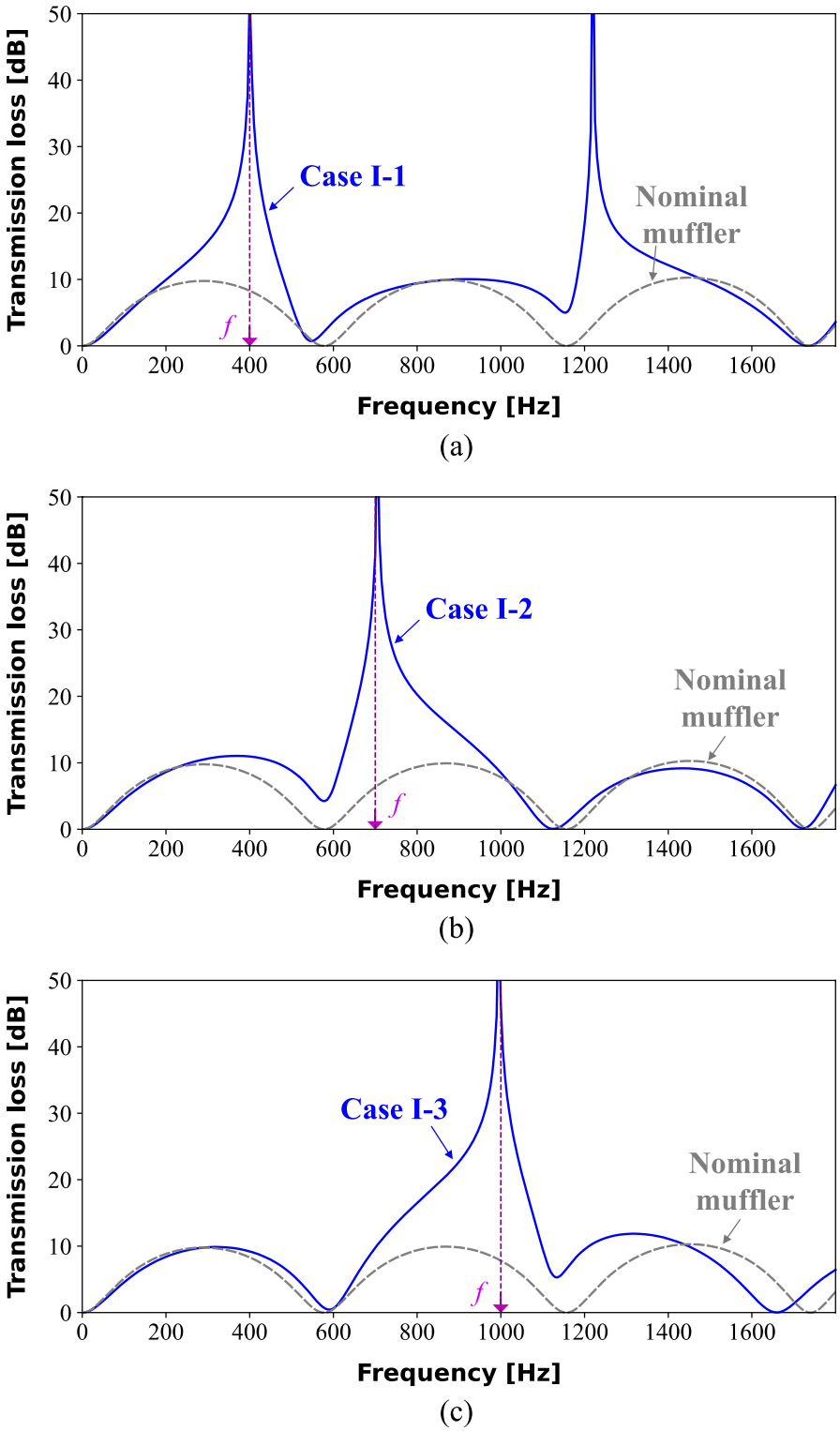

Fig. 7 Comparison of TL curves obtained from the optimized layouts by the proposed RL method and the nominal layout for **a** Case I-1 ($f = 400$ Hz), **b** Case I-2 ($f = 700$ Hz), and **c** Case I-3 ($f = 1000$ Hz)

The optimized internal layouts of the muffler for Case II are shown in Fig. 8. The TL curves obtained from these optimized layouts are compared with those of the nominal layout without solid elements in Fig. 9. The TL values of the optimized mufflers at multiple target frequencies, along with their averages, are summarized in Table 3. Both the TL curves and the average TL values at the target frequencies show notable increases, demonstrating the feasibility of the proposed RL-based TO.

4 Discussions

In this section, we first demonstrate that the topologically optimized results obtained by the proposed RL method outperform those produced by gradient-based methods, and we provide supporting explanations for why the present layouts achieve higher TL values. We then discuss several noteworthy aspects of the proposed RL strategy from a numerical standpoint, including convergence.

4.1 Performance comparison with gradient-based methods

To compare the performance of the proposed method with existing gradient-based approaches, the widely used density-based TO method was employed to solve the TL maximization problem (Case II) for broadband frequencies. The method of moving asymptotes (MMA) (Svanberg 1987) was adopted as the gradient-based optimizer. Further details of the gradient-based topology optimization procedure are provided in Appendix C for completeness.

Figure 10 presents the optimized layouts obtained by the gradient-based method using the formulation described in Appendix C, with the initial layouts taken as empty domains without any material. For clarity, the results obtained by the gradient-based method are denoted with the additional label "Grad." For example, Case II-Grad-1 refers to the result produced by the gradient-based method for Case II-1 defined in Sect. 3.

As shown by the TL results in Fig. 11, the optimized TL values obtained with the proposed RL method are consistently superior to those produced by the gradient-based method for all problems in Case II. The superior performance of the RL-based approach arises from its ability to balance exploitation and exploration in the search for optimal solutions, unlike gradient-based approaches that are more susceptible to being trapped in local extrema.

Because the obtained solutions (i.e., the optimized layouts) are highly dependent on the initial layouts—particularly for gradient-based optimization methods (Nocedal and Wright 1999; Bendsøe and Sigmund 2003)—we

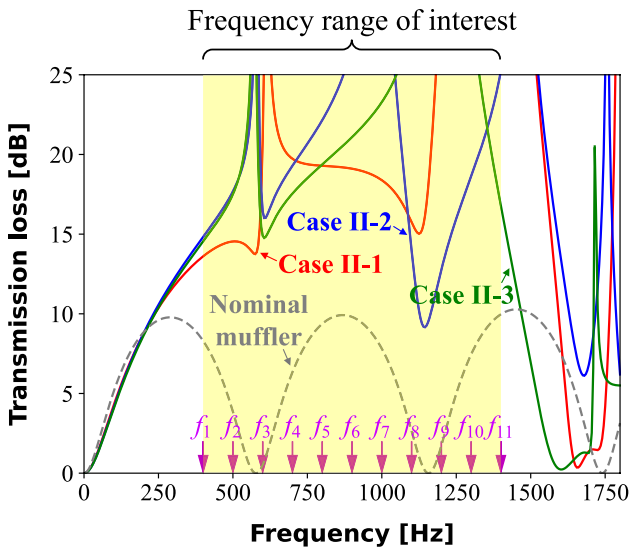

Fig. 9 Comparison of TL curves obtained from the optimized layouts and the nominal layout for **a** Case II-1 ($E_0 = 30$), **b** Case II-2 ($E_0 = 35$), and **c** Case II-3 ($E_0 = 40$)

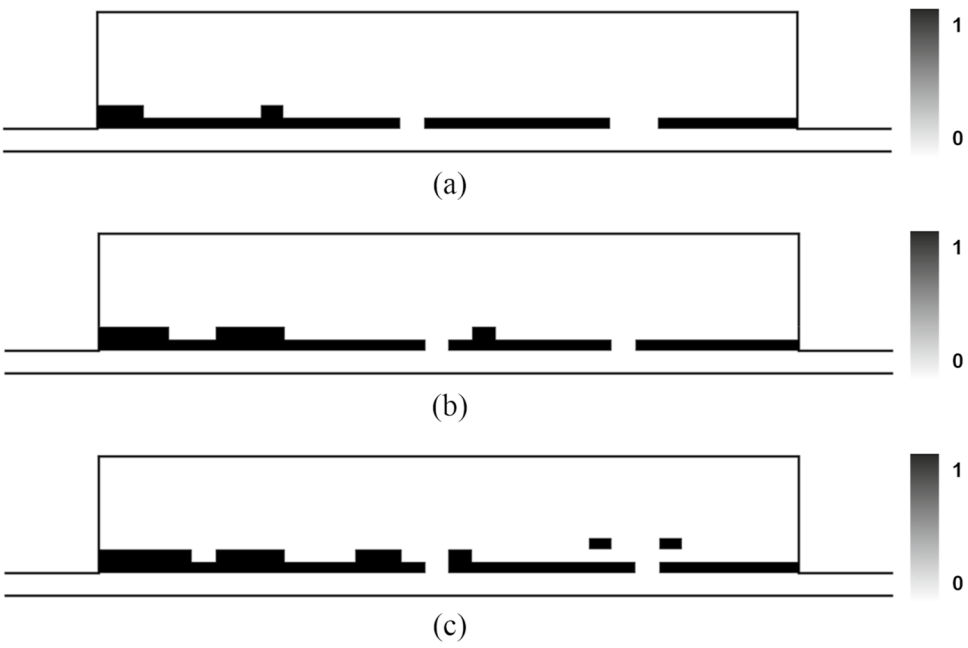

Fig. 8 Optimized layouts by the proposed RL method for broadband TL maximization (400–1400 Hz). **a** Case II-1 ($E_0 = 30$), **b** Case II-2 ($E_0 = 35$), and **c** Case II-3 ($E_0 = 40$) with $E = 300$ for problems

Table 3 TL values of optimized muffler at target frequencies and their average (Cases II-1, II-2, and II-3)

Target frequency	Case II-1	Case II-2	Case II-3
	TL values for optimized muffler		
$f_1 = 400$ Hz	13.61 dB	14.83 dB	14.53 dB
$f_2 = 500$ Hz	14.54 dB	17.68 dB	17.50 dB
$f_3 = 600$ Hz	20.93 dB	16.42 dB	14.90 dB
$f_4 = 700$ Hz	19.73 dB	19.30 dB	17.55 dB
$f_5 = 800$ Hz	19.29 dB	22.15 dB	19.37 dB
$f_6 = 900$ Hz	19.11 dB	26.74 dB	21.09 dB
$f_7 = 1000$ Hz	18.39 dB	42.07 dB	23.28 dB
$f_8 = 1100$ Hz	15.75 dB	13.98 dB	29.03 dB
$f_9 = 1200$ Hz	32.06 dB	12.78 dB	37.50 dB
$f_{10} = 1300$ Hz	31.66 dB	18.61 dB	29.14 dB
$f_{11} = 1400$ Hz	33.45 dB	25.26 dB	16.76 dB
	Average TL values at target frequencies		
	21.68 dB	20.89 dB	21.86 dB

performed each acoustic topology optimization 20 times with randomly selected initial layouts for a fair comparison between the gradient-based and RL-based methods. Figure 12 presents the average TL values over the target frequency band, where the x-axis denotes the trial number from 1 to 20. (Here, the trial numbers are rearranged such that lower TL values correspond to lower trial indices for clearer visualization.)

While the same target value, $s_{tar} = 20$ dB was used for both methods, only the proposed RL-based method yielded optimized layouts achieving TL values above the target. This is expected, since the RL agent aims directly at reaching the target value, thereby producing layouts with TL values clustered near the target. By contrast, the gradient-based method produced optimized layouts with TL values below the target. This is largely due to the additional stopping criterion commonly employed in gradient-based optimization: if the change in TL between two consecutive iterations falls below a threshold (here, 10^{-6}), the iteration terminates. Although one could, after extensive trial and error, fine-tune this criterion to obtain layouts exceeding the target TL, we did not pursue such an adjustment because it lies outside the scope of this paper.

The key point illustrated in Fig. 12 is that the proposed RL-based method offers a viable alternative to conventional approaches, as it not only tends to deliver better near-global solutions but can also reduce the overall problem-solving time compared to the gradient-based method. In the following subsection, we further investigate the physical mechanisms underlying the improved TL performance of the RL-optimized layouts.

4.2 Analysis of optimized layouts by the proposed RL method

In this section, we investigate the physical mechanisms responsible for the higher TL values obtained by the layouts optimized with the proposed RL method compared to those produced by the gradient-based method. To this end, we focus on the distinct differences between the configurations in Fig. 8 (RL method) and Fig. 10 (gradient-based method). The optimized layouts in Fig. 8 feature no explicitly formed vertical partitions; instead, they include horizontal partitions

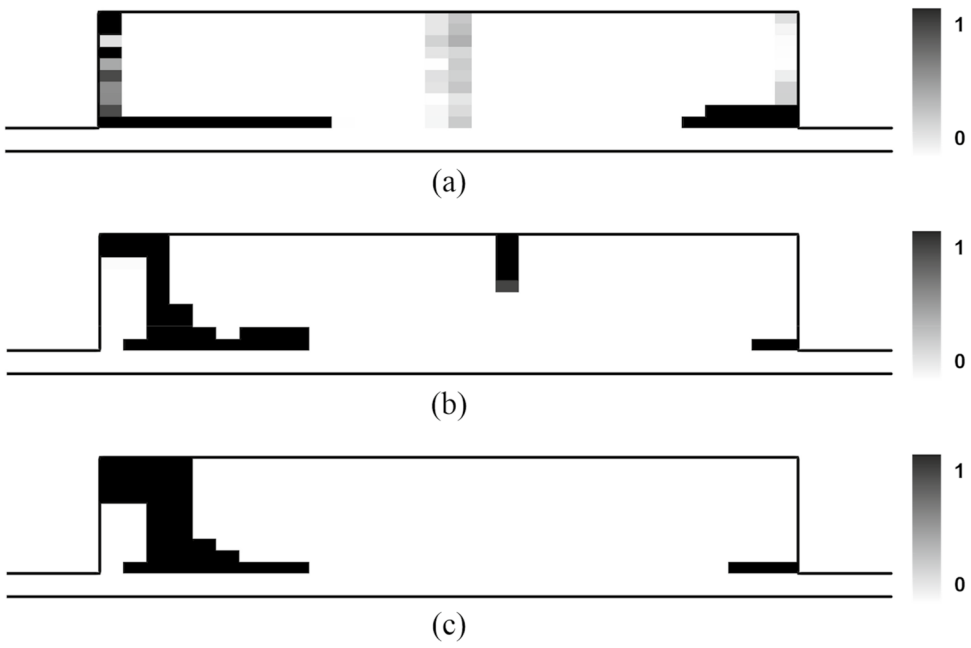

Fig. 10 Optimized layout obtained the gradient-based density method for broadband TL maximization (400–1400 Hz). **a** Case II-Grad-1 ($E_0 = 30$), **b** Case II-Grad-2 ($E_0 = 35$), and **c** Case II-Grad-3 ($E_0 = 40$) with $E = 300$

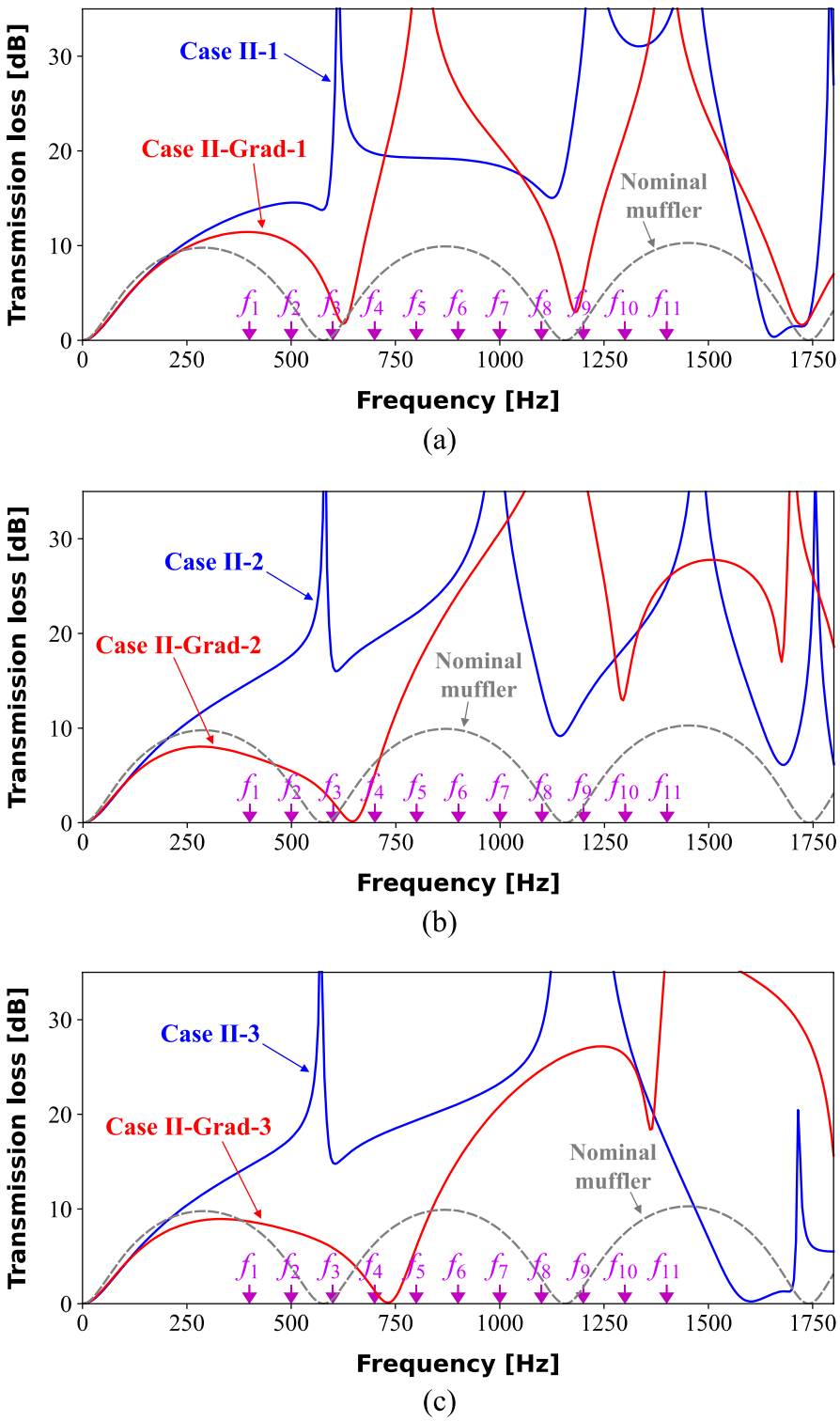

Fig. 11 Comparison of TL curves by the optimized layouts obtained by the proposed RL method and the gradient-based density method, along with those of the nominal layouts for **a** Case II-1 ($E_0 = 30$), **b** Case II-2 ($E_0 = 35$), and **c** Case II-3 ($E_0 = 40$). 'Grad'-labeled curves denote the gradient-based density method, whereas others without the label denote the proposed RL-based method

connected to the inlets and outlets, which are partially perforated. In addition, some isolated material-filled elements are observed. By contrast, the optimized layouts in Fig. 10 (gradient-based method) contain vertical partitions as well as unperforated horizontal partitions. Similar configurations have also been reported in earlier gradient-based studies (Lee and Kim 2009; Lee 2015; Oh and Lee 2017, 2023).

To investigate the mechanism behind the improved transmission loss achieved by the layouts optimized with the proposed RL method, we consider a representative case using the layout obtained for Case II-3 and examine the acoustic

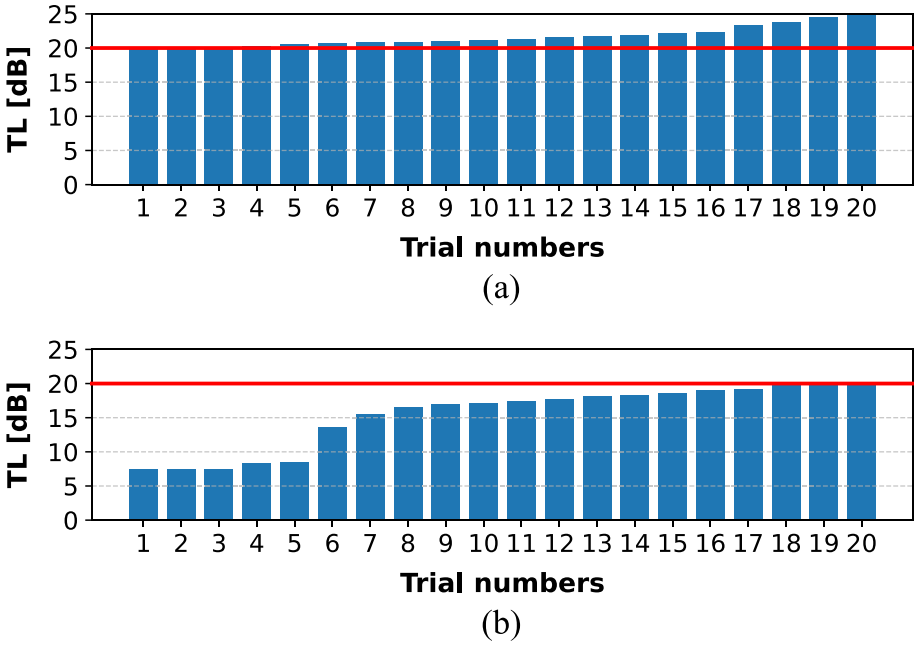

Fig. 12 Bar plots of 20 repeated trials. The x-axis denotes the rearranged trial numbers, and the y-axis denotes the average transmission loss (TL) over the target frequency band for **a** Case II-2, and **b** Case II-Grad-2

pressure distributions at the target frequencies shown in Fig. 13. In particular, we focus on the pressure distribution at $f_8 = 1100$ Hz, where the region above the horizontal partition behaves as a Helmholtz resonator (Cai et al. 2017), forming a standing wave. Consequently, the TL of the optimized layout obtained by the proposed RL method at this frequency is much higher than that of the layout obtained by the gradient-based method, as demonstrated in Fig. 11.

We also examined the effects of isolated rigid elements A and B marked in Fig. 14a on the TL value when they are either removed or dislocated. The resulting modified layouts are shown in Fig. 14b–g, and their TL curves are compared with that of the original RL-based optimized layout for Case II-3 in Fig. 15. Even small modifications significantly affect the TL curves, since they alter the formation of the pressure distribution. In particular, the modifications denoted as Mod-2, Mod-3, Mod-4, and Mod-5 reduce the TL value at $f_8 = 1100$ Hz, while contributing to increased TL at other frequencies. Although the detailed TL mechanisms of the RL-optimized layouts remain complex, these results highlight the significance of the optimized configurations obtained by the proposed method, which would otherwise be difficult to identify.

The broadband TL improvement produced by the RL-optimized layouts can be interpreted as the combined outcome of (i) strong impedance discontinuities introduced by the solid wall and (ii) local field perturbations caused by isolated solid elements. The main solid walls promote reflection and standing-wave formation by segmenting the chamber into acoustically coupled sub-volumes, whereas isolated elements act as compact scatterers that generate additional back-scattering, local resonance, and/or phase shifts. Through these mechanisms, isolated elements effectively "fine-tune" the interference and resonance conditions, which can shift, split, or broaden TL peaks across frequencies. This interpretation is consistent with the sensitivity results in Figs. 14, 15, where removing or relocating elements A and B leads to noticeable changes in the TL curve, including peak reduction near specific target frequencies and compensating increases at others.

4.3 Convergence and computational aspects of the proposed RL strategy

In reinforcement learning, if the overall setting is appropriate, the plots of Q-values and rewards with respect to the episodes typically exhibit an increasing trend (Henderson et al. 2018). Using the optimization results, the Q-value behavior across different episodes in Case II was examined. Figure 16 presents the sum of the Q-values in the muffler design domain together with the rewards for Cases II-1, II-2, and II-3. Although many episodes were required due to the complexity of the physics involved, the results show an overall increasing trend. Figure 17 further illustrates the Q-value distributions and the actions chosen by the agent for selected episodes of Cases II-1, II-2, and II-3, providing insight into the RL-based optimization process. It can be observed that the elements with high Q-values are located near the horizontal partition, which agrees with established principles in acoustic muffler design. In this study, the magnitude of the predicted Q-value for an element can be interpreted as the agent's estimation on how strongly selecting the element will increase the TL-based objective value (i.e., the expected

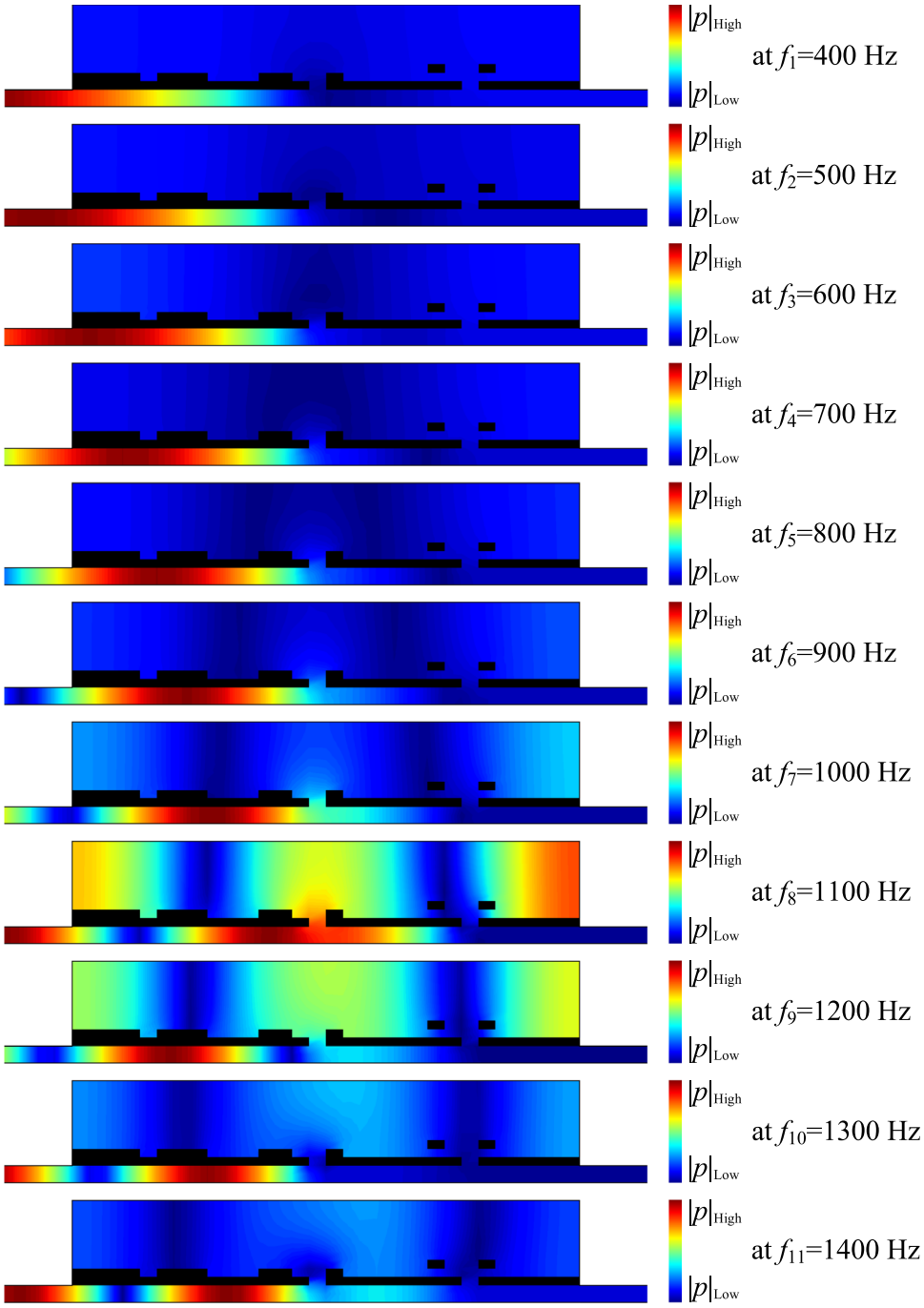

Fig. 13 Magnitude of the acoustic pressure (p) inside the muffler with the optimized layout by the proposed RL method for Case II-3. Rigid material regions are shown in black

cumulative reward under the learned policy), rather than as a direct physics-based sensitivity.

To verify the consistency and robustness of Q-learning under repeated runs with identical conditions, Case II-2 was executed 20 times. The results, shown in Fig. 18, indicate that the proposed method consistently attempts to maximize the Q-values to optimize the muffler, although each optimization trajectory differs slightly. In early episodes, the agent's actions did not appear meaningful from the perspective of acoustic engineers, since effective muffler design theoretically requires establishing both vertical and horizontal partitions to suppress noise at target frequencies. However, in later episodes, the agent identified physically meaningful configurations and converged to optimized layouts that achieve the desired objective, similar to the learning behavior demonstrated in the Pong game (Mnih et al. 2015). For each trial, the curves in Fig. 18 exhibit an almost monotonic increasing pattern, confirming that the proposed method is not highly unstable. Note that the required number of episodes varied between 10 and 60, reflecting both the

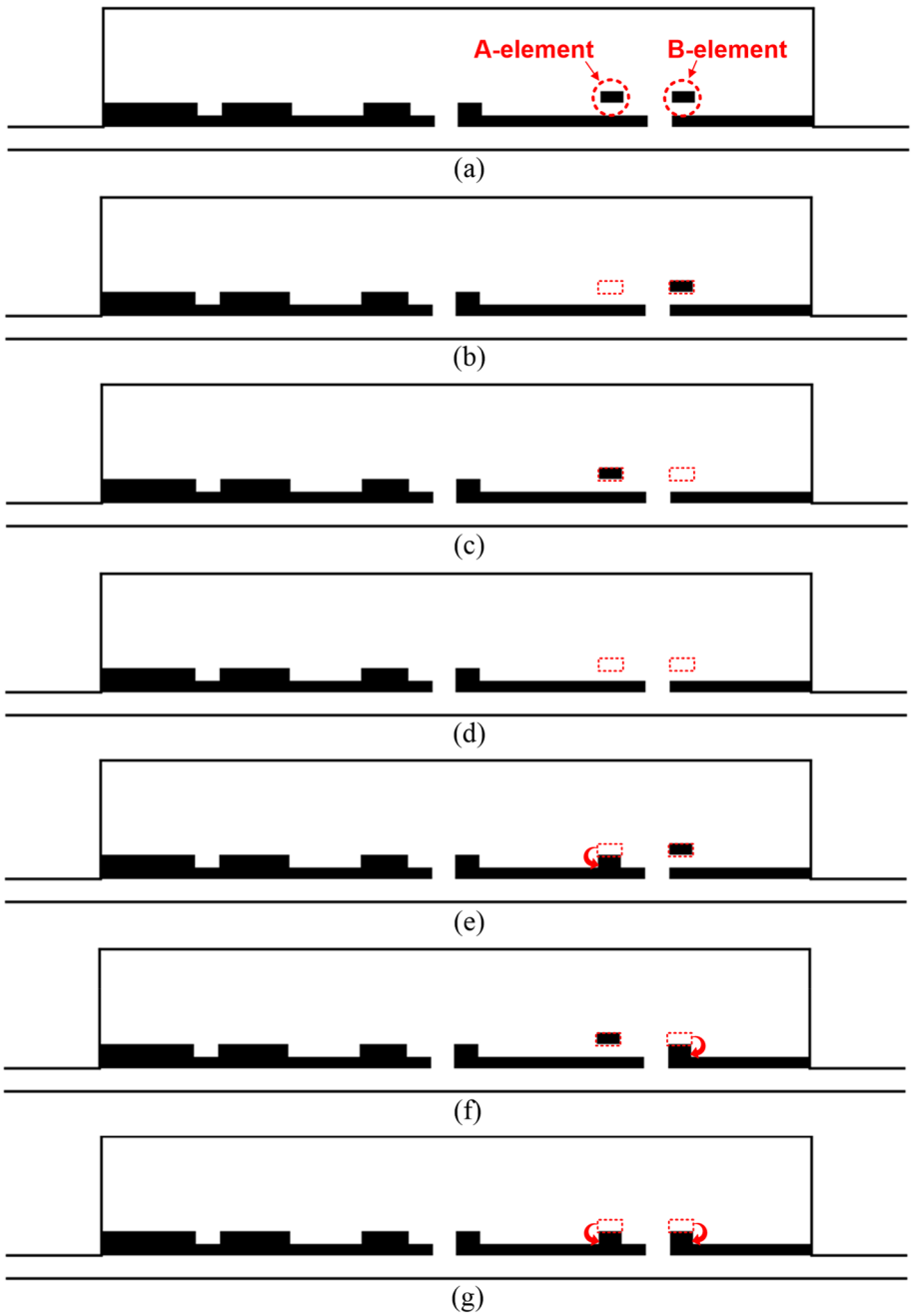

Fig. 14 **a** Original layout optimized by the proposed RL method for Case II-3, and **b–g** its modifications involving isolated elements A and B: **b** Case II-3-Mod-1, **c** Case II-3-Mod-2, **d** Case II-3-Mod-3, **e** Case II-3-Mod-4, **f** Case II-3-Mod-5, and **g** Case II-3-Mod-6

stochastic nature of RL and the presence of local minima in the optimization problem. Nevertheless, the proposed method consistently provided superior solutions compared to conventional topology optimization, as will be further discussed in the next section.

Additional investigations were conducted on the sensitivity of the proposed method to optimization parameters. First, the effect of the iteration number j was considered. We compare the results with $j=20$ (the iteration number used to obtain all the results shown above) and $j=10$ for Case II-2, shown in Fig. 19. The results show no significant difference between the two cases, and satisfactory optimization could still be achieved with fewer iterations. However, the optimization with 10 iterations exhibited less stability, with the sum of Q-values showing more zig-zag patterns.

Next, we investigate the effect of the penalization parameter q used to define the reward in Eq. (12a). Specifically, the results obtained with $q=4$ (the value used in all previous cases) are compared with those for $q=5$ in Case II-2. Figure 20 presents the corresponding optimized layouts and TL values for $q=5$. In this case, the RL-based topology optimization was terminated at the maximum number of episodes (200), rather than by the convergence criterion. This indicates that the change of q altered the

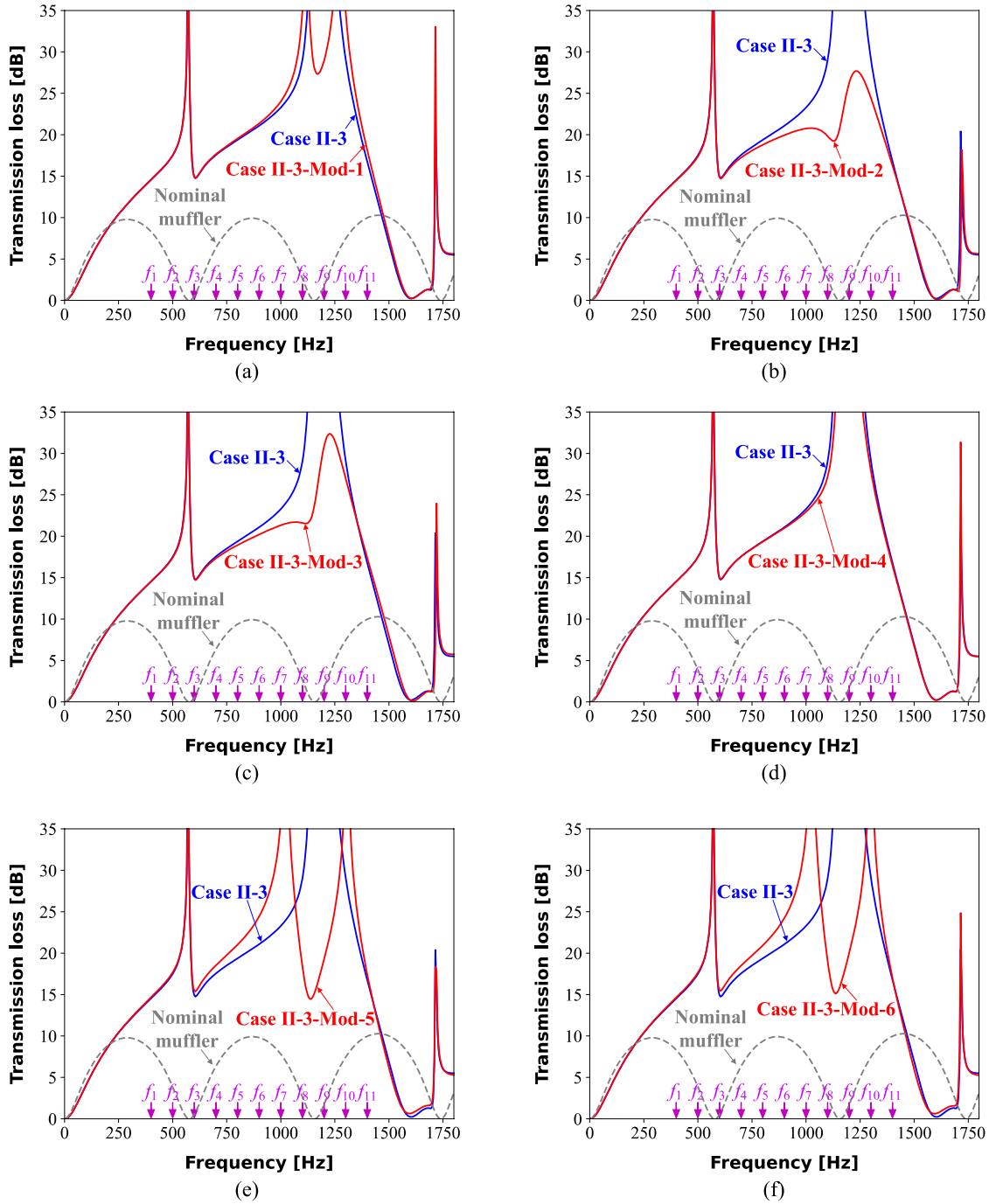

Fig. 15 Comparison of TL curves by the modified layouts described in Fig. 14 with that by the original layout for Case II-3: **a** Case II-3-Mod-1, **b** II-3-Mod-2, **c** Case II-3-Mod-3, **d** Case II-3-Mod-4, **e** Case II-3-Mod-5, and **f** Case II-3-Mod-6

overall convergence. If q is increased, the reward tends to be more extreme; large difference becomes larger and small difference becomes smaller. Accordingly, as shown in Fig. 20b, a larger q value led to a rapid increase in TL at the beginning of the optimization; however, the TL remained nearly unchanged when the design layout change is governed by small reward values. These results indicate that careful consideration is required when selecting the penalization parameter q.

Finally, to check the effect of s_{tar} on the overall optimization process, we repeated the RL-based TO of Case II-3 but the s_{tar} value is changed from 20 to 40. The final design layout and design history are plotted in Fig. 21. Here, it can be seen that the performance of the final design

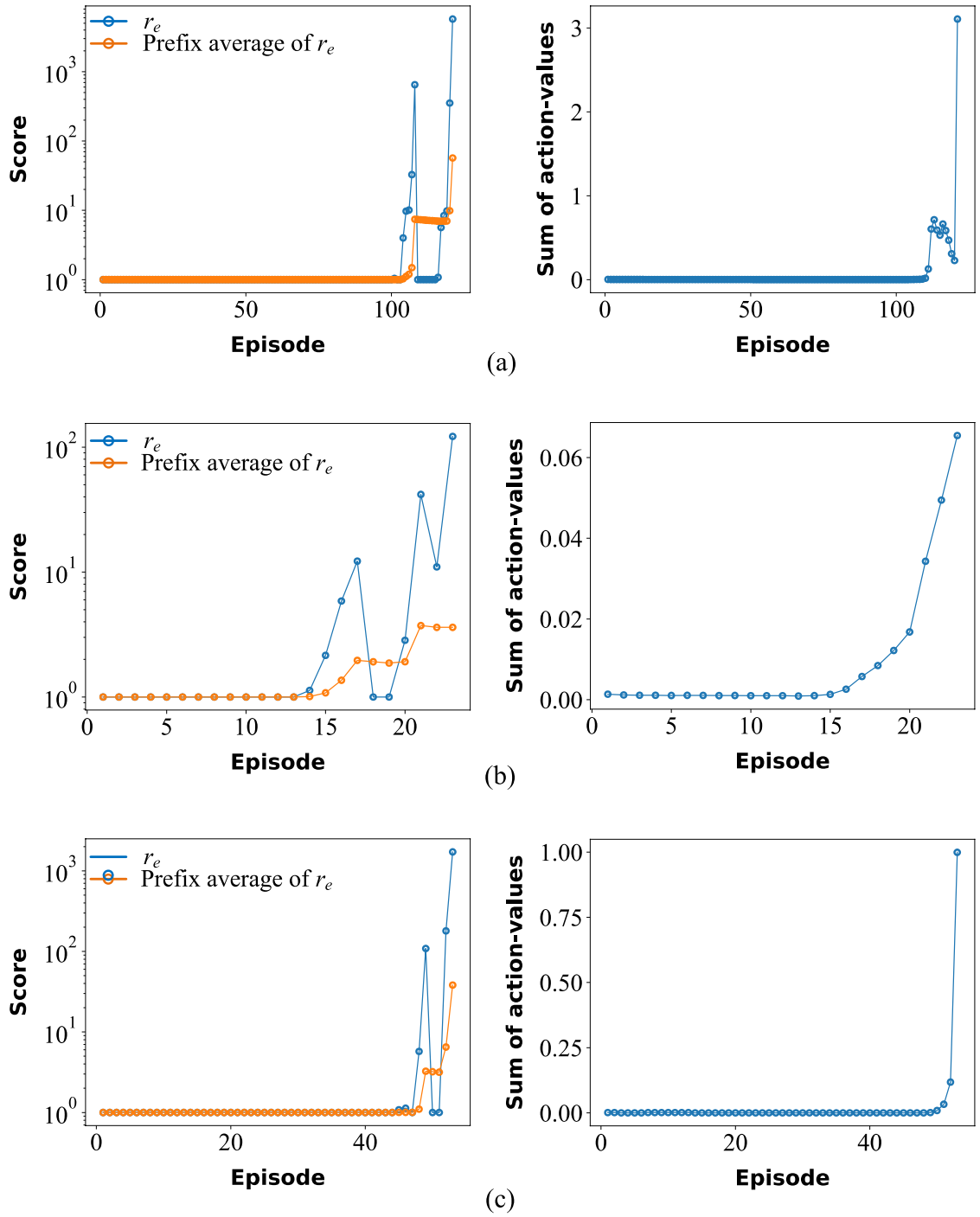

Fig. 16 Evolution of the reward and the sum of the action values (Q-values) for **a** Case II-1 ($E_0 = 30$), **b** Case II-2 $E_0 = 35$, and **c** II-3 $E_0 = 40$. The symbol r_e denotes reward in Eq. (13b) (The prefix average reward means the average of all rewards observed from the beginning up to the current index)

layout is not good, and the design history exhibits largely oscillatory behavior. This is mainly due to the change of the reward shaping, resulting in a smaller increase in the reward. Accordingly, the learning process became slowed and unstable. These observations indicate that s_{tar} should be chosen with care—ideally based on a realistic performance upper bound for the given configuration (e.g., preliminary runs or engineering estimates)—to balance learning stability and optimization aggressiveness.

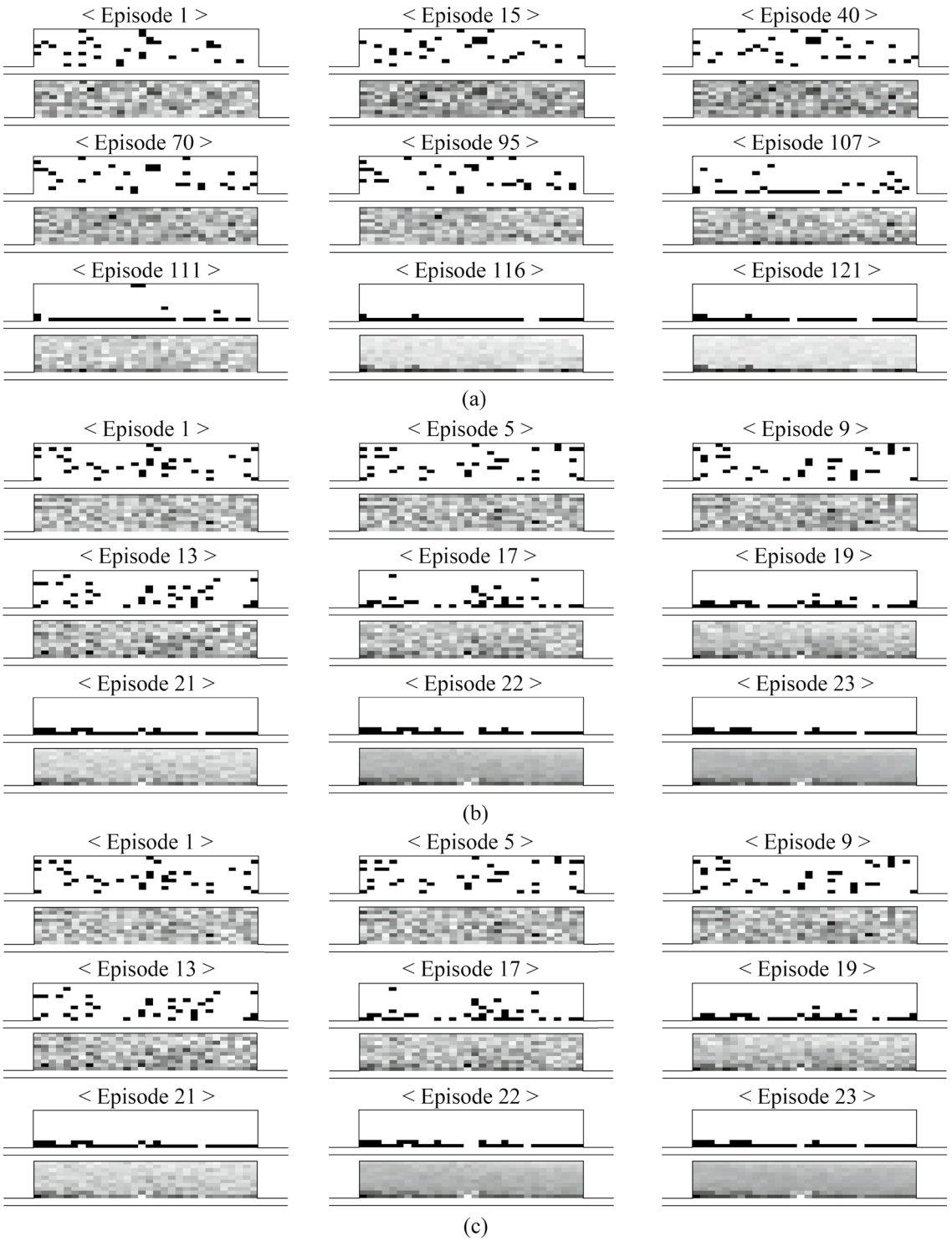

Fig. 17 Evolution of actions (top) and Q-values (bottom) in selected episodes: **a** Case II-1, **b** Case II-2, and **c** Case II-3. Q-values are plotted in grayscale

5 Conclusions

This study introduced an efficient reinforcement learning (RL) action strategy for topology optimization and demonstrated its effectiveness through the design of acoustic mufflers. The central idea of the proposed approach is to assign Q-values to individual finite elements rather than to all possible combinations of elements, thereby eliminating

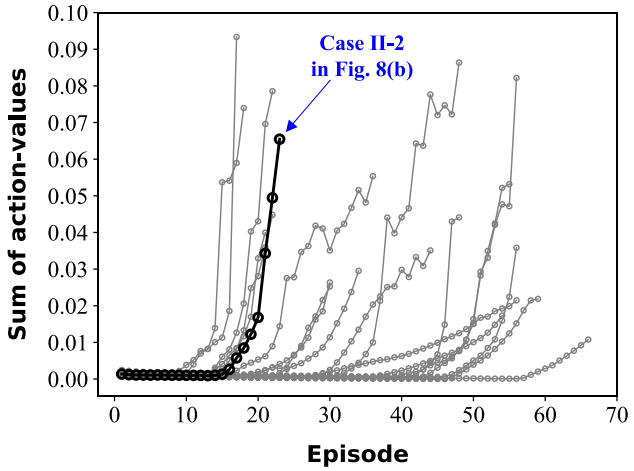

Fig. 18 History of the sum of action values (Q-values) for Case II-2 over 20 trials under identical conditions

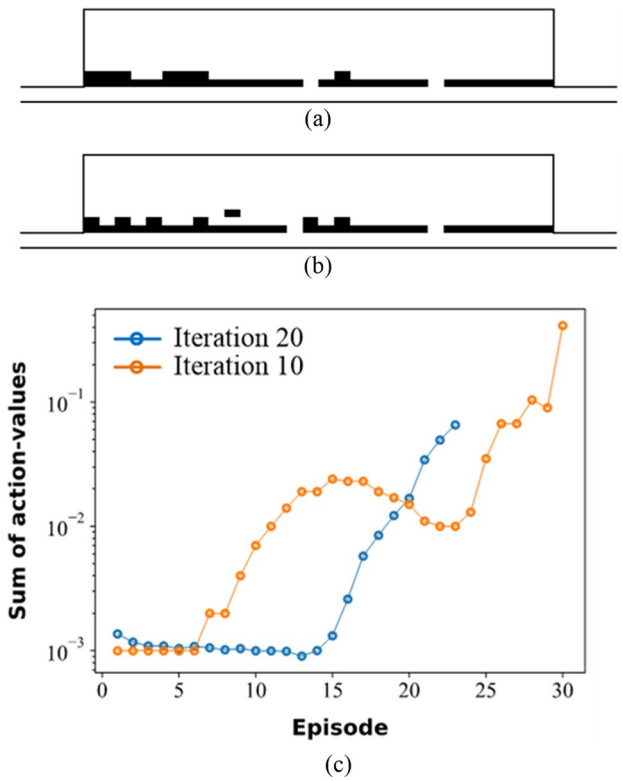

Fig. 19 Effects of the iteration number (a hyperparameter) on the optimized results for Case II-2: **a** optimized layout with 20 iterations, **b** optimized layout with 10 iterations, and **c** comparison of the sum of action values (Q-values)

the combinatorial explosion that has limited the scalability of previous RL-based topology optimization frameworks. This reformulation drastically reduces the number of output nodes required in the neural network, alleviates

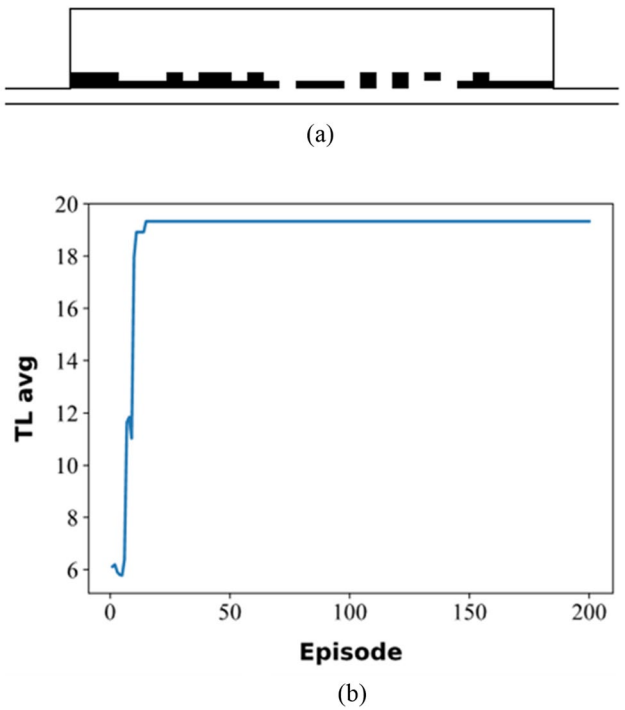

Fig. 20 Optimized results obtained with $q=5$ (penalization parameter in reward definition) to be compared with the baseline value $q=4$ used in all previous cases: **a** optimized layout and **b** evolution of the averaged transmission loss over episodes

computational costs, and enables the use of fine meshes in problems governed by expensive finite element analyses. Taken together, these advances establish a practical and scalable RL-based topology optimization paradigm well suited to high-resolution acoustic applications.

Through single- and multi-frequency muffler design problems, the proposed method consistently produced layouts with substantially higher transmission loss than those obtained by conventional gradient-based approaches. In particular, the RL-optimized layouts achieved near-global optimal performance across all cases considered, while gradient-based methods often converged to suboptimal local minima. The physical analyses revealed that the RL-based layouts leveraged acoustic mechanisms such as Helmholtz resonance and wave interference, yielding enhanced broadband performance. Sensitivity studies further demonstrated that the method is robust across multiple trials and exhibits stable convergence trends, although optimization parameters such as the iteration number and penalization factor play an important role in shaping the learning trajectory.

Overall, the proposed RL action strategy not only improves computational efficiency but also enhances solution quality, establishing reinforcement learning as a viable alternative to gradient-based topology optimization in complex acoustic design problems. Beyond muffler optimization,

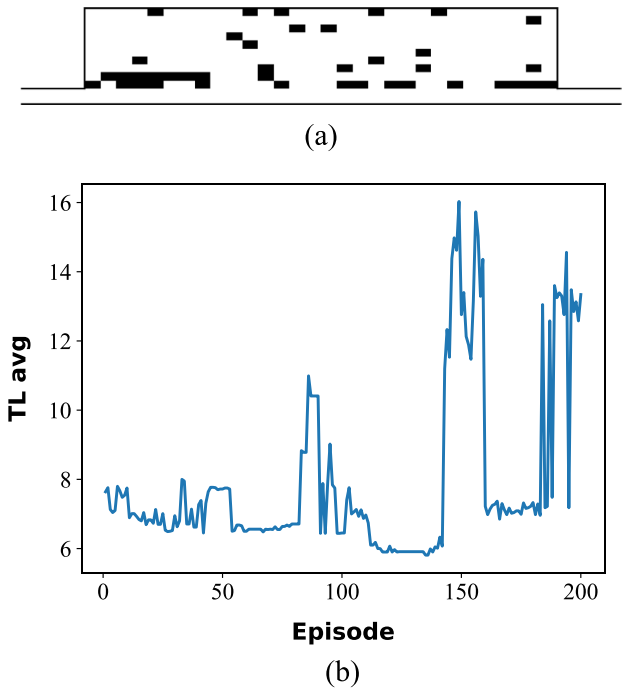

Fig. 21 Optimized results obtained with $s_{tar}=40$ (objective target value) to be compared with the baseline value $s_{tar}=20$ used in Case II-3: **a** optimized layout and **b** evolution of the averaged transmission loss over episodes

the framework shows strong potential for broader applications involving highly nonlinear system responses with many local optima. By combining scalability, robustness, and near-global search capability, the proposed approach could pave the way for reinforcement learning to become a transformative tool in large-scale engineering design optimization.

While the present study is limited to two-dimensional (2D) models, three-dimensional (3D) layouts can be generated by extruding the optimized designs in the direction perpendicular to the considered 2D plane. In principle, the proposed method can also be extended to fully 3D models; however, its direct application introduces additional challenges that remain to be addressed. For example, ensuring the continuity of an optimal layout must be considered alongside the significantly increased computational burden. Furthermore, fabrication-related issues unique to 3D structures should be incorporated into the optimization process. Accordingly, defining appropriate reward functions and addressing these related issues constitute an interesting and challenging direction for future research.

Appendix A

Comparison of the results from the single-step and multi-step action

As described in the manuscript, the proposed RL-based TO employs a single-step action, where the agent selects the entire layout (i.e., a set of material-filled elements) in one decision. One may alternatively consider a multi-step sequential action, in which the agent places elements one-by-one and repeatedly updates the policy/value function to decide the next element. In fact, this sequential decision-making protocol is conceptually aligned with early RL-based TO schemes (Hayashi and Ohsaki-like). However, as explained in the main paper, the method with Hayashi and Ohsaki defines action as an array of element index to be filled with material, which means that there are too many possible actions (in the current mesh set, there are almost 1037 actions) so that building ANN for those actions are impossible.

Accordingly, to compare the proposed method with the previous method or multi-step sequential approach, we newly implemented the sequential multi-step action protocol within our muffler problem, which is equivalent to the method suggested by Hayashi and Ohsaki, under the same mesh resolution and reported the computational cost. In the sequential multi-step protocol, elements are selected one-by-one with repeated policy updating/retraining.In Fig. 22, we compare

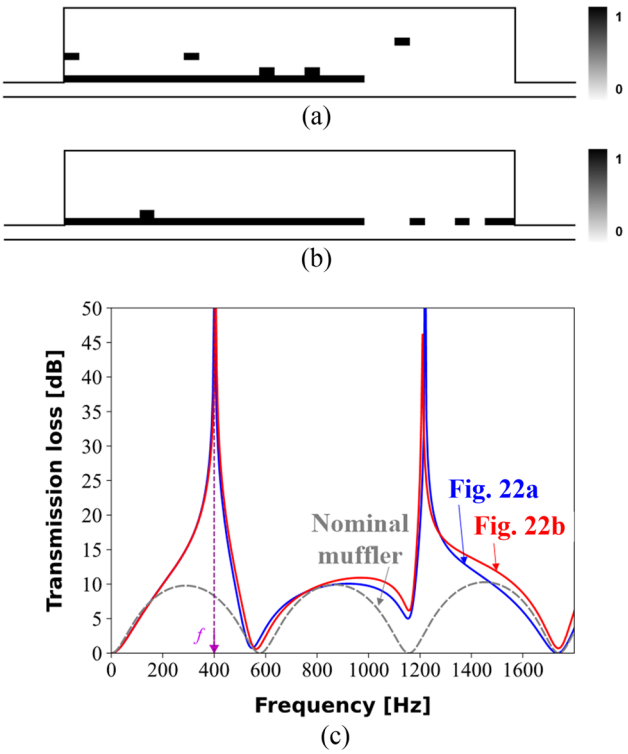

Fig. 22 Comparison of the results from the existing and proposed methods for RL-based TO: **a** Existing method, **b** Proposed method, and **c** TL curves for the both methods

the optimization outcomes of the two action formulations for the same single-objective test case (same mesh resolution and optimization settings). The final configurations obtained by the multi-step (Fig. 22a) and single-step (Fig. 22b) actions are very similar, and the resulting TL curves in Fig. 22c are also comparable. Importantly, the multi-step scheme incurs additional overhead due to repeated policy updating across multiple placement steps. In our implementation, the average wall-clock time per episode was 88.06 s for the multi-step scheme, whereas it was 68.71 s for the proposed single-step scheme, indicating that the proposed method is 21.97% faster (19.35 s reduction per episode). This comparison supports the use of the single-step action as a computationally preferable formulation while maintaining similar acoustic performance.

Appendix B

Finite element method formulation for acoustic analysis

This section explains the acoustic finite element analysis (FEA) used in the proposed optimization method. FEA was conducted to obtain the acoustic pressure at the nodes in a discretized domain for the muffler geometry, as shown in Fig. 5. To achieve this, the following governing equation, also known as the Helmholtz equation, adopting plane wave propagation for linear acoustics, should be solved in the frequency (f) domain (Kinsler et al 2000).

$$\nabla \cdot \left(\frac{1}{\rho}\nabla p\right) + \frac{\omega^2}{B}p = 0, \tag{21}$$

where p denotes the acoustic pressure, and is the solution of the governing equation. ρ, B, and ω in Eq. (21) represent air density, bulk modulus, and angular frequency, respectively. In addition, the bulk modulus and angular frequency can be represented by $B = \rho c^2$ and $\omega = 2\pi f$ (π is the mathematical constant here), respectively, where c is the speed of sound. Dirichlet, Neumann, and Robin boundary conditions were applied to the inlet (Γ_{in}), wall (Γ_{wall}), and outlet (Γ_{out}) boundaries of the muffler, represented as

$$p = p_{\text{in}} \quad \text{on } \Gamma_{\text{in}} \tag{22}$$

$$\nabla p \cdot \mathbf{n} = \left(-\mathrm{i}\omega\rho\mathbf{u}_{\text{wall}}\right) \cdot \mathbf{n} \quad \text{on } \Gamma_{\text{wall}} \tag{23}$$

$$p/(\nabla p \cdot \mathbf{n}) = \mathrm{i}c/\omega \quad \text{on } \Gamma_{\text{out}}, \tag{24}$$

where $\mathrm{i} = \sqrt{-1}$ is the imaginary unit, and $\mathbf{u}_{\text{wall}}$ is the particle velocity at the wall boundary, which can be obtained from the linearized Euler equation:

$$\mathbf{u} = -\nabla p/\mathrm{i}\omega\rho. \tag{25}$$

The governing equations (Eq. (21) and the boundary conditions in Eqs. (22–24) can be altered to the FEM form according to the conventional Galerkin method (Reddy 2019) as

$$\left[\mathbf{K} - \omega^2\mathbf{M}\right]\mathbf{P} = \mathbf{F}, \tag{26}$$

where

$$\mathbf{K} = \mathop{\mathrm{A}}_{e=1}^{E} \mathbf{k}_e \tag{27}$$

$$\mathbf{M} = \mathop{\mathrm{A}}_{e=1}^{E} \mathbf{m}_e \tag{28}$$

$$\mathbf{F} = \mathop{\mathrm{A}}_{e=1}^{E} \mathbf{f}_e \tag{29}$$

$$\mathbf{k}_e = \int_{\Omega_e} \frac{1}{\rho_e}\nabla\mathbf{N}_e^T\nabla\mathbf{N}_e d\Omega_e \tag{30}$$

$$\mathbf{m}_e = \int_{\Omega_e} \frac{1}{B_e}\mathbf{N}_e^T\mathbf{N}_e d\Omega_e + \frac{1}{\mathrm{j}\omega\rho_e c_e}\int_{\Gamma^e_{\text{out}}} \mathbf{N}_e^T\mathbf{N}_e d\Gamma^e_{\text{out}} \tag{31}$$

$$\mathbf{f}_e = -\mathrm{j}\omega\int_{\Gamma^e_{\text{in}}} \nabla\mathbf{N}_e d\Gamma^e_{\text{in}}, \tag{32}$$

where $\mathbf{K}$, $\mathbf{M}$, and $\mathbf{F}$ represent the global stiffness matrix, global mass matrix, and global load vector, respectively. $\mathbf{P}$ is a solution vector containing all nodal values in the domain. The subscript or superscript 'e' denotes the element-level equations; $\mathbf{k}_e$, $\mathbf{m}_e$, and $\mathbf{f}_e$ are the element-level local stiffness and mass matrices, and load vector, respectively. In Eqs. (30, 31, and 32), $\mathbf{N}_e$ represents the shape function, which could be any type of smooth function.

Appendix C

Conventional acoustic TO method

The optimization formulation for conventional acoustic TO is expressed as

$$\min_{0\le a_e\le 1} \sum_{n=1}^{N}\left(\frac{1}{2}w_n\bar{L}_n\right) = \min_{0\le a_e\le 1} \sum_{n=1}^{N}\left(\frac{1}{2}w_n\left|TL(\mathbf{a};f_n) - s_{\text{tar}}\right|^2\right) \tag{33}$$

$$\text{s.t.} \quad \sum_{e=1}^{E} a_e \Big/ E \le V_e\left(=\frac{E_0}{E}\right), \tag{34}$$

where w_n and $\overline{L}_n$ represent the scaling weighting factor and each sub-objective function for the conventional multi-objective TO method, respectively (Marler and Arora 2010). V_e is the volume ratio of the allowed number of black elements. Although a binarized action with 0 or 1 was introduced in the main article, let us assume that **a** is a design variable vector with elements that continuously vary from 0 to 1 in Appendix C. As shown in Eq. (35), the scaling weighted-sum method (Lee and Kikuchi 2010) is used in the conventional method to optimize each sub-objective function as fairly as possible. These values can be obtained as

$$w_n = \overline{L}_n^{\text{old}} \Bigg/ \left(\sum_{n=1}^{N} \overline{L}_n^{\text{old}} \right), \tag{35}$$

where $\overline{L}_n^{\text{old}}$ is the objective value at the previous iteration step during optimization.

The continuously varying design variables can be parameterized by the following interpolation functions considering the rational approximation of material properties (RAMP) method (Stolpe and Svanberg 2001) to assign the material properties to the corresponding e-th finite element in the design domain.

$$\rho_e(a_e) = \left(1/\rho_{\text{air}} + a_e\left(1/\rho_{\text{rigid}} - 1/\rho_{\text{air}}\right)\right)^{-1} \tag{36}$$

$$B_e(a_e) = \left(1/B_{\text{air}} + a_e\left(1/B_{\text{rigid}} - 1/B_{\text{air}}\right)\right)^{-1}. \tag{37}$$

The material interpolation functions for acoustic TO have been validated in several studies (Lee and Kim 2009; Lee 2015; Yoon et al. 2020; Chen et al. 2021). In addition, as most acoustic TO applications do not have the checkerboard problem as reported in the literature (Kook et al. 2013; Yang and Du 2013; Ferrándiz and Denia 2020), filter schemes were not introduced in this study.

Sensitivity analysis is mandatory for a gradient-based optimization algorithm and can be performed by taking the derivative of the objective function with respect to the design variables. As the objective function in Eq. (33) has only one functional term $TL(\mathbf{a};f_n)$, its derivative should be applied to TL (Tortorelli and Michaleris 1994). To clearly reveal the independent variables and parameters of TL, we rewrite Eq. (19) as

$$\overline{p}_1 = \overline{p}_1(\mathbf{a};f_n),\ \ \overline{p}_2 = \overline{p}_2(\mathbf{a};f_n),\ \ \overline{p}_3 = \overline{p}_3(\mathbf{a};f_n) \tag{38}$$

$$\overline{p}_{\text{in}}(\overline{p}_1,\overline{p}_2) = \overline{p}_1 - \overline{p}_2 e^{-\mathrm{j}kx_{12}} \tag{39}$$

$$\overline{p}_{\text{out}}(\overline{p}_3) = \overline{p}_3\left(1 - e^{-2\mathrm{j}kx_{12}}\right) \tag{40}$$

$$TL\left(\overline{p}_1(\mathbf{a};f_n),\overline{p}_2(\mathbf{a};f_n),\overline{p}_3(\mathbf{a};f_n)\right) = 10\log_{10}\frac{\left|\overline{p}_{\text{in}}\right|^2}{\left|\overline{p}_{\text{out}}\right|^2}. \tag{41}$$

According to (Lee and Kim 2009), the sensitivity analysis for TL can be evaluated as

$$\frac{\partial TL(\overline{p}_1,\overline{p}_2,\overline{p}_3)}{\partial a_e} = \frac{10}{\ln 10}\left(\frac{1}{\left|\overline{p}_{\text{in}}\right|^2}\frac{\partial\left|\overline{p}_{\text{in}}\right|^2}{\partial a_e} - \frac{1}{\left|\overline{p}_{\text{out}}\right|^2}\frac{\partial\left|\overline{p}_{\text{out}}\right|^2}{\partial a_e}\right) \tag{42}$$

$$\begin{aligned}\left|\overline{p}_{\text{in}}\right|^2 = {} & \frac{1}{\gamma}\left(\text{Re}(\overline{p}_1) - \text{Re}(\overline{p}_2)\cos\left(kx_{12}\right) - \text{Im}(\overline{p}_2)\sin\left(kx_{12}\right)\right)^2 \\ & + \frac{1}{\gamma}\left(\text{Im}(\overline{p}_1) - \text{Im}(\overline{p}_2)\cos\left(kx_{12}\right) + \text{Re}(\overline{p}_2)\sin\left(kx_{12}\right)\right)^2,\end{aligned} \tag{43}$$

$$\left|\overline{p}_{\text{out}}\right|^2 = \left(\text{Re}(\overline{p}_3)\right)^2 + \left(\text{Im}(\overline{p}_3)\right)^2 \tag{44}$$

$$\gamma = \left(1 - \cos\left(2kx_{12}\right)\right)^2 + \left(\sin\left(2kx_{12}\right)\right)^2 \tag{45}$$

$$\begin{aligned}\frac{\partial\left|\overline{p}_{\text{in}}\right|^2}{\partial a_e} = {} & \frac{2}{\gamma}\left(\text{Re}(\overline{p}_1) - \text{Re}(\overline{p}_2)\cos\left(kx_{12}\right) - \text{Im}(\overline{p}_2)\sin\left(kx_{12}\right)\right) \\ & \cdot\left(\text{Re}\left(\frac{\partial\overline{p}_1}{\partial a_e}\right) - \text{Re}\left(\frac{\partial\overline{p}_2}{\partial a_e}\right)\cos\left(kx_{12}\right) - \text{Im}\left(\frac{\partial\overline{p}_2}{\partial a_e}\right)\sin\left(kx_{12}\right)\right) \\ & + \frac{2}{\gamma}\left(\text{Im}(\overline{p}_1) - \text{Im}(\overline{p}_2)\cos\left(kx_{12}\right) + \text{Re}(\overline{p}_2)\sin\left(kx_{12}\right)\right) \\ & \cdot\left(\text{Im}\left(\frac{\partial\overline{p}_1}{\partial a_e}\right) - \text{Im}\left(\frac{\partial\overline{p}_2}{\partial a_e}\right)\cos\left(kx_{12}\right) + \text{Re}\left(\frac{\partial\overline{p}_2}{\partial a_e}\right)\sin\left(kx_{12}\right)\right),\end{aligned} \tag{46}$$

$$\frac{\partial\left|\overline{p}_{\text{out}}\right|^2}{\partial a_e} = 2\left(\text{Re}(\overline{p}_3)\right)\text{Re}\left(\frac{\partial\overline{p}_3}{\partial a_e}\right) + 2\left(\text{Im}(\overline{p}_3)\right)\text{Im}\left(\frac{\partial\overline{p}_3}{\partial a_e}\right). \tag{47}$$

To obtain the derivatives of Eqs. (42–47), the derivative of the nodal vectors for the acoustic pressure (**P**) with respect to the e-th design variable (a_e) must be calculated. One should first take the derivative of the system-level matrix equation (Eq. 26) using the chain rule as follows:

$$\frac{\partial\mathbf{P}}{\partial a_e} = \left[\mathbf{K} - \omega^2\mathbf{M}\right]^{-1}\left[\frac{\partial\mathbf{K}}{\partial a_e} - \omega^2\frac{\partial\mathbf{M}}{\partial a_e}\right]\mathbf{P}. \tag{48}$$

To determine the sensitivity of Eq. (48), two matrix derivative terms with respect to the design variables $\partial\mathbf{K}/\partial a_e$ and $\partial\mathbf{M}/\partial a_e$ should be preliminarily calculated. Considering the design variable assignment for the e-th element, the derivative only affects the corresponding element; thus, the expressions can be simplified as

$$\frac{\partial\mathbf{K}}{\partial a_e} = \frac{\partial\mathbf{k}_e}{\partial a_e} \tag{49}$$

$$\frac{\partial \mathbf{M}}{\partial a_e} = \frac{\partial \mathbf{m}_e}{\partial a_e}. \tag{50}$$

Through material interpolation using Eqs. (36) and (37), the element stiffness and mass matrices can be rewritten as

$$\mathbf{k}_e = \int_{\Omega_e} \frac{1}{\rho_e(a_e)} \nabla \mathbf{N}_e^T \nabla \mathbf{N}_e d\Omega_e \tag{51}$$

$$\mathbf{m}_e = \int_{\Omega_e} \frac{1}{B_e(a_e)} \mathbf{N}_e^T \mathbf{N}_e d\Omega_e, \tag{52}$$

where Ω_e denotes the e-th element-level domain. The two derivatives $\partial \mathbf{k}_e / \partial a_e$ and $\partial \mathbf{m}_e / \partial a_e$ in Eqs. (49) and (50) can be easily obtained using the simple chain rule as follows:

$$\frac{\partial \mathbf{k}_e}{\partial a_e} = \frac{\partial \mathbf{k}_e}{\partial \rho_e} \frac{\partial \rho_e}{\partial a_e} = -\frac{1}{\rho_e} \frac{\partial \rho_e}{\partial a_e} \mathbf{k}_e \tag{53}$$

$$\frac{\partial \mathbf{k}_e}{\partial \rho_e} = \int_{\Omega_e} -\frac{1}{\rho_e^2} \nabla \mathbf{N}_e^T \nabla \mathbf{N}_e d\Omega_e = -\frac{1}{\rho_e} \int_{\Omega_e} \frac{1}{\rho_e} \nabla \mathbf{N}_e^T \nabla \mathbf{N}_e d\Omega_e = -\frac{1}{\rho_e} \mathbf{k}_e \tag{54}$$

$$\frac{\partial \mathbf{m}_e}{\partial a_e} = \frac{\partial \mathbf{m}_e}{\partial B_e} \frac{\partial B_e}{\partial a_e} = -\frac{1}{B_e} \frac{\partial B_e}{\partial a_e} \mathbf{m}_e \tag{55}$$

$$\frac{\partial \mathbf{m}_e}{\partial B_e} = \int_{\Omega_e} -\frac{1}{B_e^2} \mathbf{N}_e^T \mathbf{N}_e d\Omega_e = -\frac{1}{B_e} \int_{\Omega_e} \frac{1}{B_e} \mathbf{N}_e^T \mathbf{N}_e d\Omega_e = -\frac{1}{B_e} \mathbf{m}_e. \tag{56}$$

Acknowledgments The authors acknowledge that Prof. Hayoung Chung has contributed as a co-corresponding author to this paper, while the contact information could not be provided due to Structural and Multidisciplinary Optimization's policy.

Author contributions Kee Seung Oh: Conceptualization, Methodology, Software, Investigation, and Writing–original draft. Yoon Young Kim: Supervision and Writing—review and editing. Hayoung Chung: Funding acquisition, Supervision, and Writing—review and editing. Joo Hwan Oh: Funding acquisition, Supervision, and Writing—review and editing.

Funding Open Access funding enabled and organized by Seoul National University. This work was supported by the InnoCORE program of the Ministry of Science and ICT (N10250154) and by the National Research Foundation of Korea (NRF) grants (Nos. RS-2023-00240918, RS-2024-00343120, RS-2024-00406514, RS-2025-02216282, RS-2025-23525252) funded by the Korean Government.

Data Availability The datasets and code generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

Replication of results The presented results were mainly obtained using our Python codes and may be provided on reasonable request.

References

Allaire G, Bonnetier E, Francfort G, Jouve F (1997) Shape optimization by the homogenization method. Numer Math 76(1):27–68

Bendsøe MP, Sigmund O (2003) Topology optimization: theory, methods, and applications. Springer, Berlin

Brown NK, Garland AP, Fadel GM, Li G (2022) Deep reinforcement learning for engineering design through topology optimization of elementally discretized design domains. Mater des 218:110672

Cai C, Mak CM, Shi X (2017) An extended neck versus a spiral neck of the Helmholtz resonator. Appl Acoust 115:74–80

Chandrasekhar A, Suresh K (2021) TOuNN: topology optimization using neural networks. Struct Multidiscip Optim 63(3):1135–1149

Chen Y, MengcF HX (2021) Creating acoustic topological insulators through topology optimization. Mech Syst Signal Process 146:107054

Chi H, Zhang Y, Tang TLE, Mirabella L, Dalloro L, Song L, Paulino GH (2021) Universal machine learning for topology optimization. Comput Methods Appl Mech Eng 375:112739

Denia FD, Selamet A, Fuenmayor FJ, Kirby R (2007) Acoustic attenuation performance of perforated dissipative mufflers with empty inlet/outlet extensions. J Sound Vib 302(4–5):1000–1017

Dong HW, Shen C, Liu Z, Zhao SD, Ren Z, Liu CX, He X, Cummer SA, Wang YS, Fang D, Cheng L (2024) Inverse design of phononic meta-structured materials. Mater Today 80:824–855

Ferrándiz B, Denia FD, Martínez-Casas J, Nadal E, Ródenas JJ (2020) Topology and shape optimization of dissipative and hybrid mufflers. Struct Multidiscip Optim 62:269–284

Van Hasselt H, Guez A, Silver D (2016) Deep reinforcement learning with double Q-learning, In: Proceedings of the AAAI conference on artificial intelligence

Hayashi K, Ohsaki M (2020) Reinforcement learning and graph embedding for binary truss topology optimization under stress and displacement constraints. Front Built Environ 6:59

Henderson P, Islam R, Bachman P, Pineau J, Precup D, Meger D (2018) Deep reinforcement learning that matters, In: Proceedings of the AAAI conference on artificial intelligence

Horn RA, Johnson CR (2012) Matrix analysis. Cambridge University Press, New York

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kinsler LE, Frey AR, Coppens AB, Sanders JV (2000) Fundamentals of acoustics. John Wiley & Sons, New York

Kollmann HT, Abueidda DW, Koric S, Guleryuz E, Sobh NA (2020) Deep learning for topology optimization of 2D metamaterials. Mater des 196:109098

Kook J, Jensen JS, Wang S (2013) Acoustical topology optimization of Zwicker's loudness with Padé approximation. Comput Methods Appl Mech Eng 255:40–66
Kulkarni TD, Saeedi A, Gautam S, Gershman SJ (2016) Deep successor reinforcement learning. arXiv preprint arXiv:1606.02396.
Lample G, Chaplot DS (2017) Playing FPS games with deep reinforcement learning, In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 31.
Lee JW (2015) Optimal topology of reactive muffler achieving target transmission loss values: design and experiment. Appl Acoust 88:104–113
Lee J, Kikuchi N (2010) Structural topology optimization of electrical machinery to maximize stiffness with body force distribution. IEEE Trans Magn 46(10):3790–3794
Lee JW, Kim YY (2009) Topology optimization of muffler internal partitions for improving acoustical attenuation performance. Int J Numer Methods Eng 80(4):455–477
Lee JK, Oh KS, Lee JW (2020) Methods for evaluating in-duct noise attenuation performance in a muffler design problem. J Sound Vib 464:114982
Marler RT, Arora JS (2010) The weighted sum method for multi-objective optimization: new insights. Struct Multidiscip Optim 41(6):853–862
Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, Riedmiller M, Fidjeland AK, Ostrovski G et al (2015) Human-level control through deep reinforcement learning. Nature 518(7540):529–533
Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D, Riedmiller M (2013) Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602.
Mukherjee S, Lu D, Raghavan B, Breitkopf P, Dutta S, Xiao M, Zhang W (2021) Accelerating large-scale topology optimization: state-of-the-art and challenges. Arch Comput Methods Eng 28(7):4549–4571
Munjal ML (1987) Acoustics of ducts and mufflers with application to exhaust and ventilation system design. John Wiley & Sons, New York
Ng AY, Harada D, Russell S (1999) Policy invariance under reward transformations: Theory and application to reward shaping, In: International Conference on Machine Learning
Nie Z, Lin T, Jiang H, Kara LB (2021) TopologyGAN: topology optimization using generative adversarial networks based on physical fields over the initial domain. J Mech des 143(3):031715
Nocedal J, Wright SJ (1999) Numerical optimization. Springer, New York
Oh KS, Lee JW (2017) Topology optimization for enhancing the acoustical and thermal characteristics of acoustic devices simultaneously. J Sound Vib 401:54–75
Oh KS, Lee JW (2023) Auxiliary algorithm to approach a near-global optimum of a multi-objective function in acoustical topology optimization. Eng Appl Artif Intell 117:105488
Oh S, Jung Y, Kim S, Lee I, Kang N (2019) Deep generative design: integration of topology optimization and generative models. J Mech des 141(11):111405
Reddy JN (2019) Introduction to the finite element method. McGraw-Hill, New York
Rozvany GI (2001) Aims, scope, methods, history and unified terminology of computer-aided topology optimization in structural mechanics. Struct Multidiscip Optim 21(2):90–108
Selamet A, Ji ZL (1999) Acoustic attenuation performance of circular expansion chambers with extended inlet/outlet. J Sound Vib 223(2):197–212
Shin M, Yoon GH (2025) A numerical study of reinforcement learning for acoustic topology optimization. Struct Multidiscip Optim 68(9):1–23
Sigmund O (2007) Morphology-based black and white filters for topology optimization. Struct Multidiscip Optim 33(4):401–424
Silver D, Huang A, Maddison CJ, Guez A, Sifre L, Van Den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M et al (2016) Mastering the game of Go with deep neural networks and tree search. Nature 529(7587):484–489
Sosnovik I, Oseledets I (2019) Neural networks for topology optimization. Russ J Numer Anal Math Model 34(4):215–223
Stolpe M, Svanberg K (2001) An alternative interpolation scheme for minimum compliance topology optimization. Struct Multidiscip Optim 22(2):116–124
Sutton RS, Barto AG (2018) Reinforcement learning: An introduction. MIT Press, Massachusetts
Svanberg K (1987) The method of moving asymptotes—a new method for structural optimization. Int J Numer Methods Eng 24(2):359–373
Tortorelli DA, Michaleris P (1994) Design sensitivity analysis: overview and review. Inverse Probl Eng 1(1):71–105
Wu TW, Wan GC (1996) Muffler performance studies using a direct mixed-body boundary element method and a three-point method for evaluating transmission loss. J Vib Acoust 118:479–484
Xu MB, Selamet A, Lee IJ (2004) Huff NT, Sound attenuation in dissipative expansion chambers. J Sound Vib 272(3–5):1125–1133
Yan J, Zhang Q, Xu Q, Fan Z, Li H, Sun W, Wang G (2022) Deep learning driven real time topology optimisation based on initial stress learning. Adv Eng Inform 51:101472
Yang R, Du J (2013) Microstructural topology optimization with respect to sound power radiation. Struct Multidiscip Optim 47(2):191–206
Yedeg EL, Wadbro E, Berggren M (2016) Interior layout topology optimization of a reactive muffler. Struct Multidiscip Optim 53:645–656
Yoon WU, Park JH, Lee JS, Kim YY (2020) Topology optimization design for total sound absorption in porous media. Comput Methods Appl Mech Eng 360:112723
Yu Y, Hur T, Jung J, Jang IG (2019) Deep learning for determining a near-optimal topological design without any iteration. Struct Multidiscip Optim 59(3):787–799