



Multi-Disciplinary and Multi-Objective Optimization Method Based on Machine Learning

Jiahua Dai,^{*} Peiqing Liu,[†] Ling Li,[‡] and Qiulin Qu[§]

Beihang University, 100191 Beijing, People's Republic of China

and

Tongzhi Niu[¶]

Huazhong University of Science and Technology, 430074 Wuhan, People's Republic of China

<https://doi.org/10.2514/1.J063213>

The optimization of aircraft is a typical multidisciplinary and multi-objective problem. To solve this problem, the difficulty lies not only in the high cost of discipline performance evaluation but also in the complex coupling relationship between different disciplines. To improve the optimization efficiency, a new optimization method is proposed, including two new algorithms: conditional generative adversarial nets with vector similarity (VS-CGAN) and distributed single-step deep reinforcement learning with transfer learning (TL-DSDRL). For low-cost disciplines, VS-CGAN learns the relationship between variables and objectives through presampling to compress the variable domains. The cosine function is used to evaluate the similarity between the random noise and generated variables to avoid mode collapse. For high-cost disciplines, TL-DSDRL improves optimization efficiency through pretraining. The newly designed reward function and multi-agent cooperation mechanism enhance the multi-objective search ability of reinforcement learning.

Nomenclature

AoA	=	angle of attack
a_t	=	action in step t
C	=	condition of conditional generative adversarial nets
Cd	=	drag coefficient
Cl	=	lift coefficient
c	=	chord length
i	=	episode
K	=	lift-to-drag ratio
M	=	Mach number
Q	=	action value function
Re	=	Reynolds number
r_t	=	reward in step t
s_t	=	state in step t
t	=	step
γ	=	discount factor
δ	=	output of conditional generative adversarial nets
θ	=	flap deflection angle
λ	=	noise of conditional generative adversarial nets
π	=	policy
ϕ	=	spoiler deflection angle
χ_L	=	sweep angle of leading edge
χ_T	=	sweep angle of trailing edge
ω_Q	=	parameters of critic net
ω_π	=	parameters of actor net

I. Introduction

THE design of an aircraft or its components is usually a multidisciplinary design optimization (MDO) and multi-objective

Received 26 May 2023; revision received 7 September 2023; accepted for publication 30 October 2023; published online 19 December 2023. Copyright © 2023 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the eISSN 1533-385X to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

*Ph.D. Candidate, School of Aeronautic Science and Engineering.

[†]Professor, School of Aeronautic Science and Engineering.

[‡]Associate Professor, School of Aeronautic Science and Engineering; lilingase@buaa.edu.cn (Corresponding Author).

[§]Professor, School of Aeronautic Science and Engineering. Associate Fellow AIAA.

[¶]Ph.D. Candidate, State Key Laboratory of Intelligent Manufacturing Equipment and Technology.

task, which is the result of various performance balances [1–3]. The optimization of high-lift devices is a typical multidisciplinary and multi-objective optimization problem [4–8]. It needs to simultaneously consider the requirements of aerodynamics, mechanism, structure, control, and other disciplines for the objectives of high efficiency, high reliability, and light weight. Among those objectives, the contradiction between the best aerodynamic performance and the mechanism's realizability is the most significant. In the past, aerodynamic performance has been mainly concerned with other disciplines matched as much as possible. According to the Federal Airworthiness Regulations (FAR) Part 25 [9], different takeoff and landing stages have different aerodynamic performance requirements. Flaps are designed in several positions to meet these different requirements. The design process is serial that pure aerodynamic optimization is carried out first and complex mechanisms are designed to drive the flaps to reach the optimization positions, resulting in a heavier structure and lower maintainability [10]. The development trend is to adopt simpler mechanisms to pursue comprehensive performance [4,11], as shown in Fig. 1, leading to the problem that the mechanism may not be able to guide the flap to the aerodynamic-optimal positions accurately. For example, the inboard flap of the Boeing 787 is driven by a four-link mechanism [12]. With the reduced number of mechanism parts, this mechanism can only guide the flap to two positions accurately (one is the cruise position). The other positions can only be selected from the mechanism motion track. Therefore, it is necessary to introduce mechanism constraints into the aerodynamic optimization. Franke [13] finds that the lift coefficient of the aerodynamic optimization under the mechanism constraints is 0.1 less than that of the pure aerodynamic optimization. If the traditional serial design method is used, the aerodynamic optimization and the mechanism design will be iterated multiple times until they meet their respective requirements at the same time, which is obviously inefficient.

To make up for the deficiency of the serial process, several MDO methods are proposed [14–16]. Among them, individual discipline feasible (IDF) and multidisciplinary discipline feasible (MDF) are commonly used. IDF separates disciplines by introducing variable copies and checking consistency constraints to meet multidisciplinary feasibility. Compared with IDF, MDF directly analyzes the relationship between design variables and targets, which makes optimization problems simple. From a purely mathematical point of view, the combination of MDF and widely used optimization algorithms (gradient-based method [17] or gradient-free method [18–20]) can solve the optimization problems well. However, MDF requires system analysis for each optimization iteration, and the

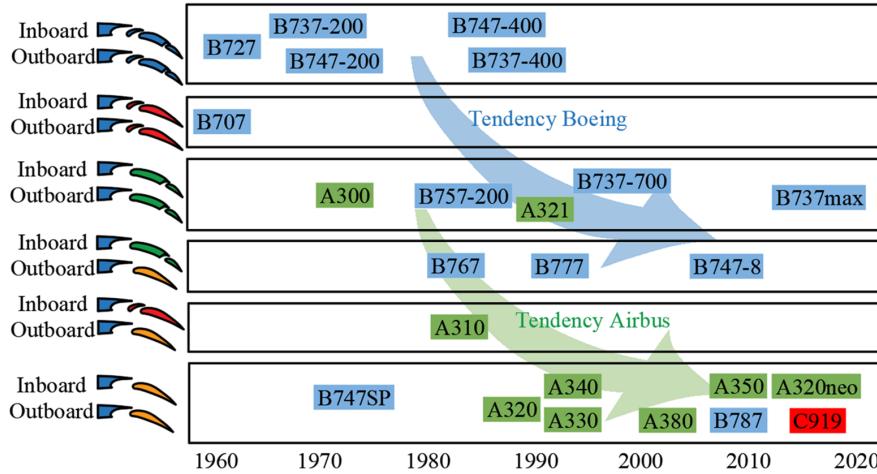


Fig. 1 Development trend of high-lift device mechanisms.

evaluation cost is limited, so it is difficult to conduct. This can be simply divided into two problems. The first one is that the performance evaluation costs of some disciplines are very high, such as solving the Navier–Stokes (N-S) equations to simulate the flowfield. The other is that although the performance evaluation costs of some disciplines are very low, they are closely coupled with the high-cost disciplines, leading to an increase in the overall optimization cost. For these two problems, an idea to improve optimization efficiency by considering the characteristics of different disciplines is proposed with the following two approaches:

1) For the low-cost disciplines, a large number of samples can be evaluated in advance to find the relationship between the variables and the objectives/constraints. This can compress the variable domain to a more appropriate range, reducing the workload of the optimization algorithm.

2) For the high-cost disciplines, some low-precision fast evaluation methods can be used to pretrain the optimization algorithm to obtain a better initial policy. Then, high-precision evaluation methods are applied to search for accurate, optimal solutions.

It should be noted that both the compression of the variable domain and the pretraining of the algorithm are abstract methods. Because the relationships between the optimization objectives and the variables are often implicit and nonlinear, it is difficult to express the relationships analytically. Therefore, the efficiency of the traditional optimization algorithm is difficult to improve with the above two approaches, unless with human intervention. Artificial intelligence, a data-driven technology aiming to learn and imitate human behavior, provides the feasibility.

For the first approach, the use of precollected data to compress the variable domain has become one of the development directions of an advanced optimization algorithm [21]. The generative adversarial network (GAN) completes this process through the game between the generator and the discriminator without extensively annotated training data [22]. GAN has been widely used in shape optimization. Du et al. [23], Chen and Fuge [24], and Li et al. [25] use it to produce B-splines or free-form deformation (FFD) control points of airfoils. Since the generator can map a random noise distribution to a real distribution, this method can reduce abnormal airfoil sampling in the optimization to improve efficiency. Li and Zhang [26–28] extend the application of GAN to 3D aerodynamic optimization by prejudging several sections of the sampled configuration whether they are abnormal to avoid unnecessary aerodynamic performance evaluations. For GAN, the goal of the generator is only to transfer random noise to real data. However, in multidisciplinary optimization, the generated samples should not only meet the performance requirements of a single discipline but also meet the coupling relationship between disciplines. As an improvement, the conditional generative adversarial network (CGAN) [29] generates real data under given conditions. The trained generator of CGAN can quickly generate airfoils with the required stall angle of attack (AoA) or drag characteristics [30]. Wu et al. [31] train a

CGAN model to learn the relationship between missile shape and aerodynamic performance. For both subsonic and supersonic flow conditions, the CGAN method can obtain the same optimization result with less time cost. Despite the benefits of CGAN, the data produced from random noise are not entirely controllable, leading to mode collapse that may result in an overly compressed variable domain. The implementation of Wasserstein GAN [31,32] can partially alleviate these flaws. However, the inherently nonspecific nature of random noise cannot guarantee sufficient diversity within the generated samples. The issue of mode collapse within such optimization methods has not been adequately emphasized, in part due to the rich airfoil database provided by the University of Illinois at Urbana-Champaign (UIUC) [33]. However, for broader multidisciplinary optimization problems, GAN may not be trained on a very diverse database.

Reinforcement learning (RL) combined with transfer learning (TL) can be conducted in the second approach. Yan et al. [34] optimize the shape of a missile, where the low-precision panel method and the high-precision computational fluid dynamics (CFD) method are coupled by TL to improve efficiency and optimal solution performance. Li et al. [35] improve the performance of RL in airfoil optimization by pretraining the policy through imitation learning. Dai et al. [36] propose a 2D-to-3D method for an aerodynamic optimization problem of high-lift device. An agent pretrained through 2D optimization is more efficient in 3D optimization and can find better solutions compared with direct 3D optimization. However, RL is difficult to solve multi-objective optimization problems [21]. For a multi-objective problem, the feedback from performance evaluation will be a vector containing all objectives. For comparison, the reward function used in traditional RL must be a scalar. In addition, the current RL algorithm is generally combined with neural networks, i.e., deep reinforcement learning (DRL). The loss function of the neural network must also be a scalar. To solve multi-objective problems using RL, two types of methods are commonly used, namely, single-policy method and multipolicy method.

The single-policy method aims to learn an optimal policy with subjective preferences. One of the methods is the weighted linear method [36,37]. Obviously, the optimization result will depend heavily on the weights. To obtain the desired multi-objective solution, Qin et al. [38] use the principal component analysis to determine these weights. There is a defect in that the weighted linear method can only be used for the convex optimization problem [39,40], so the weighted Chebyshev method [41] is introduced to solve this problem. It can be used to solve multi-objective optimization problems regardless of the Pareto front shape, and the weight influence will be suppressed [37,42]. However, the weighted Chebyshev function is nonlinear, so the reward is not additive, which is contrary to the update method of the Q values [42,43]. Therefore, although the weighted Chebyshev method has been widely used, there is no theoretical guarantee that the action selected can obtain the maximum expected Q value.

The multipolicy method directly searches for the complete Pareto front. The convex hull value iteration (CHVI) algorithm [44] is one of the algorithms that search for multi-objective solutions with different weights by calculating the union of all convex hulls. Lizotte et al. [45] conduct similar work by fitting value functions with different weights through piecewise functions. CHVI can only be used for optimization problems with known environments. The method of threshold lexicographic ordering (TLO) focuses on one objective and requires other objectives to be greater than the thresholds. It can only solve single-objective optimization problems with multiple constraints [46,47]. Strictly speaking, TLO is not a real multi-objective optimization method. Moffaert and Nowe [48] and Moffaert et al. [49], respectively, introduce the Pareto selection mechanism and the hypervolume indicator into the Q -learning algorithm to obtain two multipolicy methods. However, since the algorithm is developed from Q -learning, the two methods can only solve problems with discrete variables. While most optimization problems have continuous variables, the policy gradient method can overcome the defect.

In this paper, a multidisciplinary and multi-objective optimization method is proposed. It contains two novel algorithms, conditional generative adversarial nets with vector similarity (VS-CGAN) and distributed single-step deep reinforcement learning with transfer learning (TL-DSDRL), which can, respectively, overcome the shortcomings of CGAN and DRL. It attempts to solve an optimization problem from a completely parallel perspective and obtain a solution with better comprehensive performance.

This paper is organized as follows: Section II describes the details of this method. To verify the proposed optimization method, the trailing-edge high-lift device equipped in a wide-body commercial aircraft is optimized. The optimization problem is described in Sec. III, and the results are analyzed and discussed in Sec. IV. Section V is the conclusion.

II. Optimization Approach

This section describes the details of the multidisciplinary and multi-objective optimization method. The overall process is shown in Fig. 2. It mainly includes the following three stages: the green, blue, and purple paths, respectively, represent the operations of stages (1), (2), and (3).

1) *Train VS-CGAN*: Sample over the entire variable domain using Latin hypercube sampling (LHS), and the samples that achieve the basic objectives of low-cost disciplines are labeled as real. A noise vector λ and condition vector C are input into the generator, resulting in fake samples. During the training, the discriminator learns to evaluate the attributes of the samples according to λ and C , while the generator learns to produce samples to deceive the discriminator. The constructed generator will be used in both pretraining and optimization.

2) *Conduct the pretraining of TL-DSDRL*: An agent composed of several actor nets and a single critic net is used to solve the optimization problem cursorily. The optimization objectives are quickly evaluated, approximately by some simplified methods. In an optimization episode, a set of parameters is produced from the agent. Part of the parameters are used as λ , while the others are directly regarded as the optimization variables as well as C . The remaining optimization variables are produced by the generator.

3) *Conduct the optimization using TL-DSDRL*: The pretrained agent will get a relatively good initial policy. The optimization process is similar to the pretraining, except that a 3D CFD-accurate method is used to evaluate the objectives.

In this section, the details of the two novel algorithms, VS-CGAN and TL-DSDRL, are described in turn.

A. VS-CGAN for Variable Compression

As mentioned above, it is necessary to compress the variable domain first. However, the relationship between the objectives/constraints and the variables is implicit and nonlinear, and the appropriate range of the variables is difficult to be expressed analytically. For the low-cost disciplines, the data-driven method GAN can be conducted.

The core structure of GAN is a generator and a discriminator. The task of the former is to map an arbitrary noise distribution to a real data distribution and produce new data to deceive the discriminator. The task of the discriminator is to distinguish real data from fake data produced by the generator. The training process of GAN is the confrontation between them. The Jensen–Shannon (JS) divergence is commonly used in GAN to evaluate the difference between the real data distribution and the arbitrary noise distribution. Therefore, the objective of GAN can be expressed as

$$\begin{aligned} \text{Objective of GAN} = & \min_G \max_D \{\mathbb{E}_{\delta \sim P_{\text{real}}(\delta)} [\log D(\delta)] \\ & + \mathbb{E}_{\lambda \sim P_\lambda(\lambda)} [\log(1 - D(G(\lambda)))]\} \end{aligned} \quad (1)$$

where δ belongs to the real distribution $P_{\text{real}}(\delta)$ and λ belongs to the arbitrary noise distribution $P_\lambda(\lambda)$. It should be noted that GAN is an unconditional model, and the objective is to get a general generator. As an improvement, CGAN uses C as an additional parameter to guide the generator to produce the specific data. However, the original CGAN is still not suitable for variable compression because of the following requirements:

1) It is not the first requirement that the produced variable is real, because the objectives will be evaluated and the constraints will be checked in the optimization process, as shown in Fig. 2. By contrast, the variables should be as rich as possible so as not to ignore the solutions of high performance.

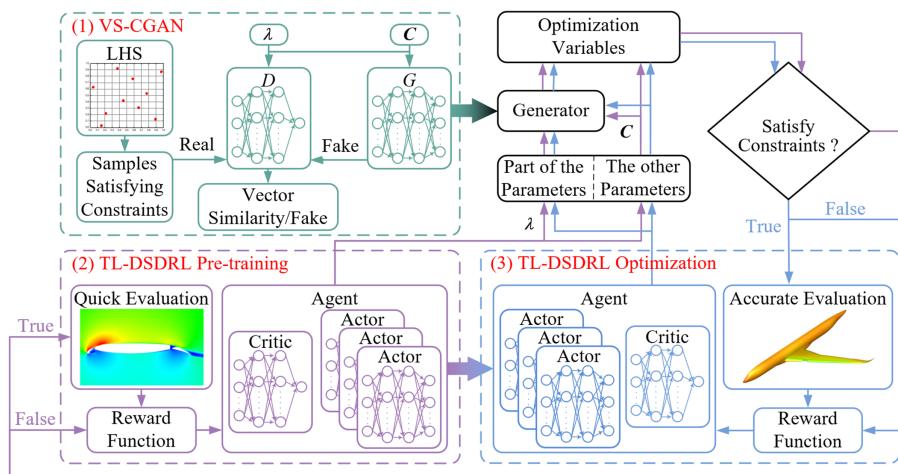


Fig. 2 The process and the framework of the optimization method.

2) The monotonic relationship between the produced variable and the noise should be kept as much as possible. For example, the larger the element in the variable vector, the larger the element in the noise vector. By keeping the monotonic relationship, there is no change in the relationship between the optimization variables and the optimization objectives, thus not increasing the difficulty of the optimization algorithm.

In sum, the noise vector λ still has no physical meaning and no explicable relationship with the data δ produced by the generator. To satisfy the above requirements, a novel algorithm, VS-CGAN, is proposed, and its pseudocode is shown in Algorithm 1. In addition to judging whether δ is real or fake, the discriminator also needs to evaluate the similarity between δ and λ if δ is real. Cosine similarity $VS(\lambda, \delta)$ is used to evaluate the level, as shown in Eq. (2):

$$VS(\lambda, \delta) = \cos(\kappa) = \frac{\lambda \cdot \delta}{\|\lambda\| \|\delta\|} \quad (2)$$

Note that $VS(\lambda, \delta)$ is expected to be 1, which indicates that δ produced by the generator is not only real but also consistent with the direction of the given λ . Different from the original CGAN algorithm, the discriminator's task is more similar to regression than classification. Therefore, the mean square error (MSE) loss function is used instead of the binary cross entry (BCE).

The structure of the VS-CGAN is shown in Fig. 3. Both the generator and the discriminator are fully connected networks with two hidden layers. The sigmoid activate function is added in the

Algorithm 1: VS-CGAN

LHS within the domain of optimization variables
Select all samples meeting the constraints to form a library Ω
Randomly initialize the generator G and the discriminator D

For episode = 1 to M

- Randomly select a group of samples $\{C, \delta\}$ from Ω
- Randomly generate a greedy factor $\epsilon \in [0, 1]$
- If** $\epsilon \leq 0.5$:

 - Let $\lambda = \delta$ and $VS(\lambda, \delta) = 1$

- Else:**

 - Randomly generate the noise vector λ
 - Calculate the similarity $VS(\lambda, \delta)$ as Eq. (2)

End if

Update the discriminator by the loss function $Loss_{D,1}$:
 $Loss_{D,1} = MSE(D(C, \lambda, \delta), VS(\lambda, \delta))$

Update the discriminator by the loss function $Loss_{D,2}$:
 $Loss_{D,2} = MSE(D(C, \lambda, G(C, \lambda)), -1)$

Update the generator by the loss function $Loss_G$:
 $Loss_G = MSE(D(C, \lambda, G(C, \lambda)), 1)$

End for

output layer of the generator to normalize the produced variables between 0 and 1. The tanh is added to the discriminator's output layer to ensure the same value range as the cosine function. The rectified linear unit (ReLU) activation function is added to the other layers to increase the ability of nonlinear fitting. The Adam optimizer [50] within Pytorch is used for all neural network training.

A cheap benchmark is to test VS-CGAN. The simple problem is to compress a two-dimensional square area with a side length of 10 into a radius of 2 with a center of (5, 5), as shown in Eq. (3).

$$\text{objective: } \sqrt{(x-5)^2 + (y-5)^2} < 2, x \in [0, 10], y \in [0, 10] \quad (3)$$

Due to the complex relationship between the optimization variables and the constraints, traditional methods cannot achieve the compression target. Therefore, this paper compares the effectiveness of VS-CGAN and CGAN, and the results are shown in Fig. 4. It can be seen that for the CGAN algorithm, although most of the generated samples (98.7%) are within the red range, there is a significant problem of overcompression, while for VS-CGAN, the generated samples are more diverse.

B. TL-DSDRL for Multi-Objective Optimization

Deep reinforcement learning is a process in which an agent constantly tries in the environment to learn a policy with the maximum reward. DRL is based on the Markov decision process. In a step t , the agent observes the current state s_t and selects action a_t according to a policy $\pi(s_t)$. The state refers to the position of the agent in the environment, which can be understood as the current configuration in optimization problems. Action can be understood as a modification made by the agent, or can be directly regarded as a new configuration. The environment gives a reward r_t and turns to the next state s_{t+1} . In an episode, the total reward G_t obtained from the step t to the end is expressed as

$$G_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \dots$$

$$R_t = r_t + r_{t+1} + r_{t+2} + \dots \quad (4)$$

where γ is a discount factor. The action value function $Q(s_t, a_t)$ is used to evaluate the value of the action expressed as

$$Q(s_t, a_t) = \mathbb{E}(G_t | s = s_t, a = a_t) \quad (5)$$

In the commonly used actor-critic framework, a critic net $Q(s, a | \omega_Q)$ is used to fit the action value function, and an actor net $\pi(s | \omega_\pi)$ modifies its policy according to the gradient provided by the critic net. Note that ω_Q and ω_π are the parameters of the actor and critic nets, respectively. For an optimization problem, finding the optimal solution is the main task. In contrast, the process of changing the initial solution to the optimal solution is not important. Therefore, the single-step deep reinforcement learning (SDRL) is more suitable

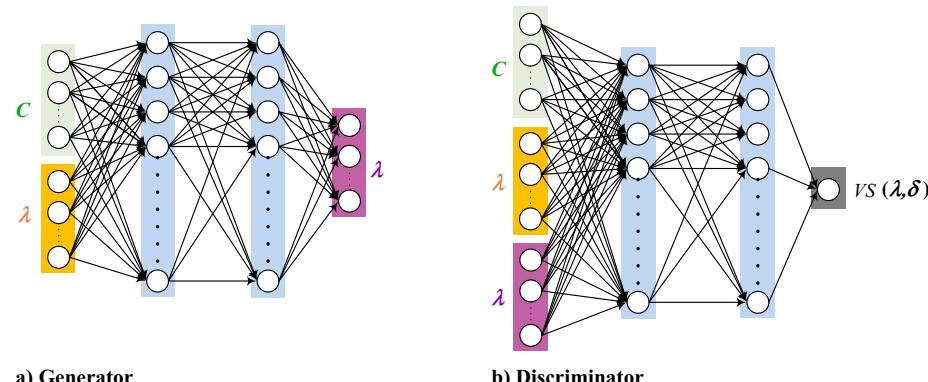


Fig. 3 VS-CGAN structure.

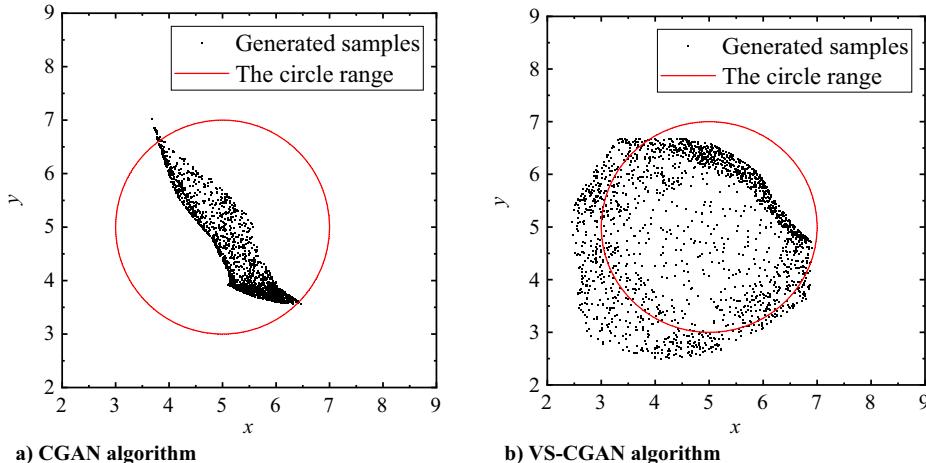


Fig. 4 Comparison of the generated data distribution from CGAN and VS-CGAN.

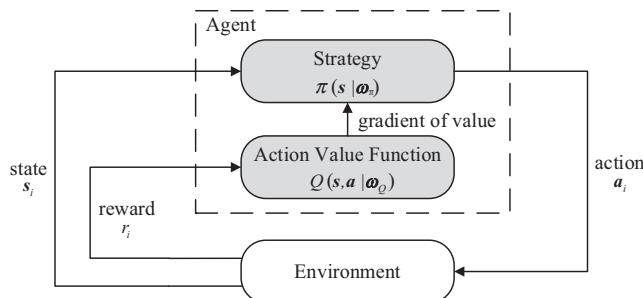


Fig. 5 An episode process of SDRL.

for optimization with $\gamma = 0$ [42]. In the i episode, the process of SDRL is shown in Fig. 5.

A single-objective optimization problem of adjusting the variables $\mathbf{x} \in X$ to maximize an objective $f(\mathbf{x})$ can be easily solved by SDRL, as shown in Algorithm 2. It is a normal distribution sampling, with μ as the mean value and σ^2 as the variance. In an episode of i , the action \mathbf{a}_i of the agent is set as the variable of the new solution, and accordingly the reward r_i is the objective improvement between the new solution and the old one.

However, since the optimal solution of a multi-objective problem is a set of solutions, this problem will become complex. The target of a maximization problem can be defined as

$$\max f(x) = \max(f_1(x), f_2(x), \dots, f_n(x)), \quad x \in X \quad (6)$$

where x is the optimization variable, and f_1-f_n are n objectives. Pareto dominance and Pareto front are often used to evaluate the multi-objective results, and the multi-objective optimization is to find a complete Pareto front.

Algorithm 2: SDRL for single-objective optimization

Randomly initial actor net $\pi(s|\omega_\pi)$ and critic net $Q(s, a|\omega_Q)$

For episode $i = 1$ to M

Randomly initial state s_i

Select action $a_i = \mathcal{N}(\mu, \sigma^2)$, $\mu = \pi(s_i | \omega_\pi)$

Calculate reward $r_i = f(a_i) - f(s_i)$

Let $O(s, a | \omega_O)$ with:

$$Loss_{\text{critic}} \equiv \text{MSE}(r_t)$$

the actor net by gradient descent:

- F -

- 1) Pareto dominance means that, for the two sets of variables $x_1, x_2 \in X$, if $f_i(x_1) > f_i(x_2), \forall i \in \{1, 2, \dots, n\}$, $f(x_1)$ is a dominating solution of $f(x_2)$. If $f_i(x_1) < f_i(x_2), \forall i \in \{1, 2, \dots, n\}$, $f(x_1)$ is a dominated solution of $f(x_2)$. In other cases, $f(x_1)$ is a non-dominated solution.

- 2) Pareto front is a set of solutions \mathbf{f} satisfying that $\exists j, \mathbf{f}(\mathbf{x}_j) \in \mathbf{f}$, dominates $\forall \mathbf{f}(\mathbf{x}_j) \in \Gamma$.

To solve a multi-objective problem using DRL, a simple idea is to set the reward function as Eq. (6). It directly takes the Pareto dominance as the evaluation standard.

$$r_i = \begin{cases} 1, & \text{nondominated solution} \\ 2, & \text{dominating solution} \\ -1, & \text{dominated solution} \end{cases} \quad (7)$$

However, it is inappropriate due to the following reasons:

1) The Markov property, one of the foundations of RL, is difficult to be satisfied. It requires that the reward only depends on the current state s_i and action a_i . The Pareto dominance is related to the previous samples. For example, a sample may be a dominating solution in the early stage of optimization, but may become a dominated solution later.

2) The reward function does not reflect level information. There is also a comparison between two solutions with the same Pareto dominance. The reward function should still include the level like $r_i = f(\mathbf{a}_i) - f(s_i)$ in Algorithm 2.

3) The reward function cannot fully express the requirement of finding a complete Pareto front. If the optimization algorithm only samples in part of domain, it can also obtain an infinite number of nondominated solutions. According to Eq. (7), the algorithm also continues to receive positive rewards. The requirement similar to the Crowding-distance of NSGA-II [51] should be included.

The reward function of DRL is generally a scalar, and the unique Pareto-compliant scalar indicator is the hypervolume [49]. It refers to the size of the region dominated by the solutions on the Pareto front. For a double-objective (f_1 and f_2) optimization problem, the hypervolume can be expressed as the gray area in Fig. 6, where J_1 , J_2 , and J_3 represent the solutions on the Pareto front, and J_{ref} represents a reference point.

Based on the hypervolume scalar indicator, the reward function is set as Eq. (7), where $HV()$ is an operator of calculating the hypervolume of the elements in it, $f(s_i)$ are the sets of objectives of each sampled solutions, and $f(a_i)$ are the sets of objectives of the new solution. If the new solution is a nondominated solution, the reward function evaluates the contribution of the solution to the hypervolume growth. If the new solution is a dominating solution, a positive reward much larger than that of a nondominated solution is expected. If the new solution is dominated, a negative reward will be fed back. Here,

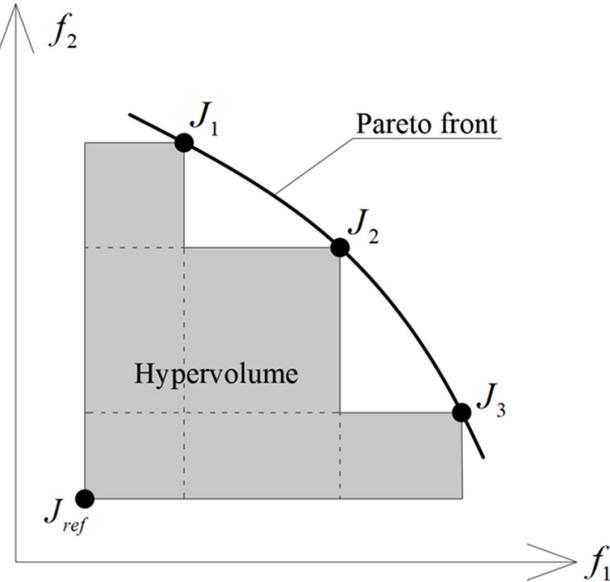


Fig. 6 Schematic diagram of a double-objective hypervolume.

the method from Fonseca et al. [52] is used to quickly calculate the hypervolume.

$$r_i = \begin{cases} HV(f(s_i) \cup f(a_i)) - HV(f(s_i)), \text{nondominated solution} \\ HV(f(a_i)), \text{dominating solution} \\ HV(f(a_i)) - HV(f(s_i)), \text{dominated solution} \end{cases} \quad (8)$$

The search ability of a single actor in the solution domain is limited. Under the influence of randomness, multiple actors can find different areas of Pareto front, i.e., a more complete Pareto front. Therefore, a novel optimization method DSDRL is proposed, and its pseudocode is shown in Algorithm 3. It uses k actors to search in the

Algorithm 3: DSDRL for multi-objective optimization

Random sampling k times

$$s_{1,1} = [x_1, x_2, \dots, x_k]$$

Randomly initialize critic net $Q(s, a|\omega_Q)$

Randomly initialize actor nets $[\pi_1(s|\omega_{\pi_1}), \pi_2(s|\omega_{\pi_2}), \dots, \pi_k(s|\omega_{\pi_k})]$

Initialize a replay buffer

For episode $i = 1$ to M

For actor nets $j = 1$ to k

 Select action $a_{i,j} = \mathcal{N}(\mu, \sigma^2), \mu = \pi_j(s_{i,j}|\omega_{\pi_j})$

 Calculate the objectives of the new solution $f(a_{i,j})$

 Calculate reward $r_{i,j}$ as Eq. (7)

 Store the experience $[s_{i,j}, a_{i,j}, r_{i,j}]$ into the replay buffer

 Randomly select n experiences $[s, a, r]$ from the replay buffer

 Update the critic net $Q(s, a|\omega_Q)$ with:

$$\text{Loss}_{\text{critic}} = \frac{1}{n} \sum \text{MSE}(r, Q(s, a|\omega_Q))$$

 Update the actor net j by gradient descent:

$$\nabla \omega_{\pi_j} = \frac{1}{n} \sum \nabla_a Q(s, a|\omega_Q)|_{s=s_{i,j}, a=\pi_j(s_{i,j})} \cdot \nabla_{\omega_{\pi_j}} \pi_j(s|\omega_{\pi_j})|_{s=s_{i,j}}$$

If $j < k$:

 Replace the j element:

$$s_{i,j+1} = \left[\dots \underbrace{a_{i,j}}_{\text{the } j \text{ element}} \dots \right]$$

Else

$$s_{i+1,1} = s_{i,k}$$

End if

End for

End for

variable domain, and the goal of each actor is to find the sample that can maximize the increase of the hypervolume. In each episode, the initial state is $s_{i,1} = [x_1, x_2, \dots, x_k]$. The search result of the j actor will replace the corresponding position in the state to ensure that the dimension remains unchanged. Since the goals of all the actors are the same, the single critic can also share the search information of each actor to improve the efficiency. Similar to the deep deterministic policy gradient (DDPG) algorithm [53], replay buffer and mini-batch stochastic gradient descent are used to train the actors and the critic. The Adam optimizer [50] is used to train the neural networks.

The Fonseca function [54] is selected to verify the multi-objective search ability of the proposed DSDRL algorithm. As a typical test benchmark, this problem has two independent variables and two objectives, as shown in Eq. (9). The test results are shown in Fig. 7. Compared to the RL method for merging into a single objective [55], the Pareto front obtained by DSDRL is closer to that of NSGA-II, which is more complete.

$$\text{objectives: } \begin{cases} \text{Minimize } f_1(x, y) = 1 - e^{-(x-1)^2 - (y+1)^2} \\ \text{Minimize } f_2(x, y) = 1 - e^{-(x+1)^2 - (y-1)^2} \end{cases}, \quad x \in (-2, 2), y \in (-2, 2) \quad (9)$$

In almost all disciplines, there are many low-cost methods that can be used to evaluate the respective objectives approximately. In the traditional design process, these low-cost methods are first applied. Then designers can propose better schemes according to their experience based on these results. The designers can be replaced by TL. Relevant studies [34–36] have proved that the neural networks after TL can conduct the similar work better than those of training from random initialization. TL-DSDRL will pretrain the agent in the low-cost methods and have an initial policy, improving the optimization efficiency and finding more solutions closer to the Pareto front.

III. Optimization Problem and Settings

The proposed optimization method in this paper is tested in the aerodynamics-mechanism optimization of the trailing-edge high-lift device. The high-lift device is equipped in a wide-body commercial aircraft, as shown in Fig. 8. The origin of the Cartesian coordinate system is located in the nose with the X -axis in the chordal direction and the Z -axis in the spanwise direction. The wing is equipped with two flaps (the inboard one and the outboard one) with downward deflectable spoilers. The inboard and outboard flaps are driven by different linkage mechanisms, as shown in Fig. 9. The outboard mechanism proposed by Mark [56] and Andreani [57] allows the flap moving in the spanwise direction. The spoilers can deflect downward to improve the aerodynamic performance by controlling the width of the gap [58–60]. The optimization problem is to change the positions of the spoilers and the flaps with the spanwise constraints and the mechanism constraints to improve the takeoff and landing aerodynamic performances of the baseline configuration. The baseline configuration is set as follows. For takeoff, the inboard flap and the outboard flap deflection angles are, respectively, 20 and 22.5° , while the inboard spoilers and the outboard spoiler's deflection angles are 6 and 6.6° . For landing, the inboard flap and the outboard flap deflection angles are, respectively, 40 and 45° , while the inboard spoilers and the outboard spoiler's deflection angles are 10 and 11° .

A. Optimization Objectives, Variables, and Constraints

According to the FAR [9] and related studies [10,60], the objectives include the following aspects, as shown in Eq. (10). The takeoff configuration needs a large lift coefficient C_l_{takeoff} to reduce the taxiing distance and a large lift-to-drag ratio K_{takeoff} to climb quickly, while the landing configuration needs a large lift coefficient C_l_{land} to reduce the approach speed. A typical incoming flow condition of $M = 0.2$ and $Re = 3.1 \times 10^7$ is selected for both takeoff and landing phases. The AoA of the takeoff configuration can vary in the range of $6^\circ \sim 9^\circ$. Taking the takeoff α as an optimization variable is to find

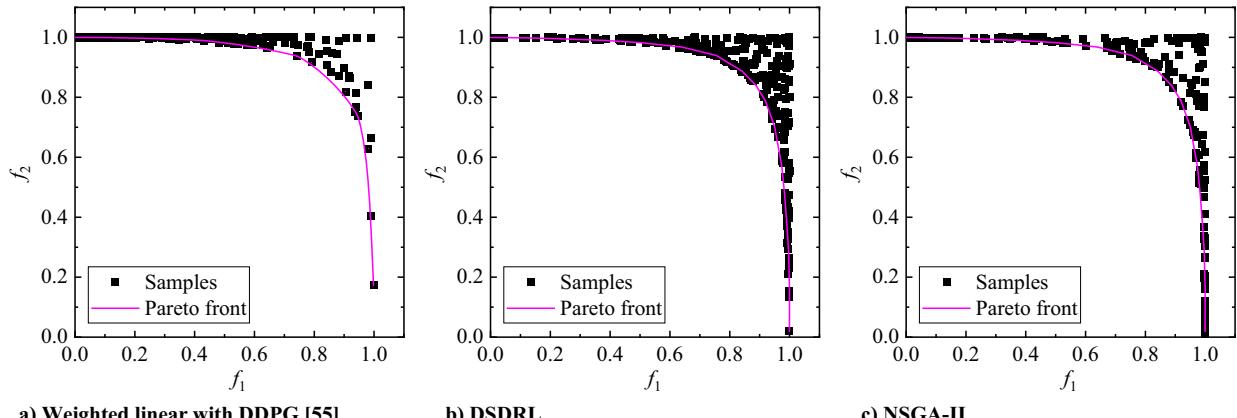


Fig. 7 Comparison of Pareto fronts obtained by different algorithms.

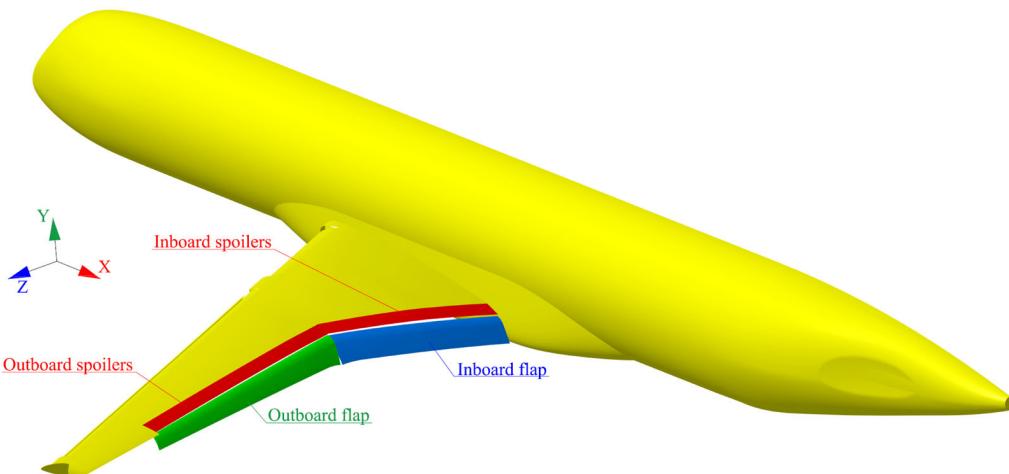


Fig. 8 Optimization model of the wide-body commercial aircraft.

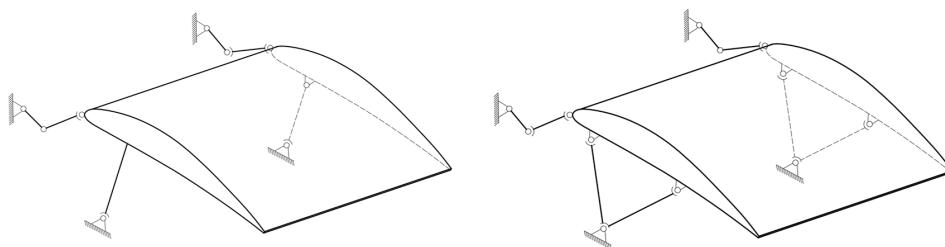


Fig. 9 Mechanisms of the inboard and outboard flaps.

more multi-objective solutions. The AoA of the landing configuration is 8°.

$$\text{Objectives : } \begin{cases} \text{Maximize } Cl_{\text{takeoff}}|_{\text{AoA}=6^\circ \sim 9^\circ} \\ \text{Maximize } K_{\text{takeoff}}|_{\text{AoA}=6^\circ \sim 9^\circ} \\ \text{Maximize } Cl_{\text{land}}|_{\text{AoA}=8^\circ} \end{cases} \quad (10)$$

The other optimization variables include the takeoff and landing positions of the two flaps (inboard flap and outboard flap) and the corresponding deflection angles of the spoilers. The deployment process of the flap can be regarded as the spatial motion of a rigid body. According to the principle of kinematics, the motion can be decomposed into rotation around an axis and translation along the axis. Thus, for the cruise position, an axis contains four independent variables, and a rotation and a displacement can be used to

represent the takeoff or landing positions of a flap. However, subject to the constraints, the spatial locations of the two flaps are limited, and the actual number of optimization variables is reduced. Due to the small sweep angle χ_T of the inboard trailing edge, the motion of the inboard flap is designed to rotate around a fixed axis; that is, the axes describing the takeoff and landing positions of the inboard flap are same and the inboard flap do not displace axially. Thus, for the inboard flap, the variables include an axis with four independent variables $X_{\text{inboard},\text{inner}}$, $Y_{\text{inboard},\text{inner}}$, $X_{\text{inboard},\text{outer}}$, and $Y_{\text{inboard},\text{outer}}$; a takeoff deflection $\theta_{\text{takeoff,inboard}}$; and a landing deflection $\theta_{\text{land,inboard}}$. The inboard flap has a total of six variables.

The takeoff and landing positions of the outboard flap can be selected in a wider range due to the use of a more complex mechanism. The axes describing its takeoff and landing positions are different. However, in order to satisfy the spanwise continuity, the axial displacement of the outboard flap is not treated as a variable but as a

following parameter. Thus for the outboard flap, the variables include two axes with eight independent variables $X_{\text{takeoff}}^{\text{outboard,inner}}$, $Y_{\text{takeoff}}^{\text{outboard,inner}}$, $X_{\text{takeoff}}^{\text{outboard,outer}}$, $Y_{\text{takeoff}}^{\text{outboard,outer}}$, $X_{\text{land}}^{\text{outboard,inner}}$, $Y_{\text{land}}^{\text{outboard,inner}}$, $X_{\text{land}}^{\text{outboard,outer}}$, and $Y_{\text{land}}^{\text{outboard,outer}}$; a takeoff deflection $\theta_{\text{takeoff}}^{\text{outboard}}$; and a landing deflection $\theta_{\text{land}}^{\text{outboard}}$. The outboard flap has a total of 10 variables.

The motion of the spoilers is simple. In this paper, the takeoff and landing deflections of the inboard spoilers are used as variables, represented by $\phi_{\text{inboard}}^{\text{takeoff}}$ and $\phi_{\text{inboard}}^{\text{land}}$. The outboard spoilers must follow to meet spanwise continuity. With the adding of the takeoff AoA, the total number of variables is 19.

Since the variables of the inboard and outboard flaps are independent, the two flaps may be misaligned at the kink position, as shown in Fig. 10. Too large clearance will produce unnecessary flow phenomena

$$\text{Spanwise Constraints: } \begin{cases} \Delta X < 4.5\%c \\ \Delta Y < 4.5\%c \\ \Delta \beta_{XZ} < 7^\circ \\ \Delta \beta_{XY} < 2^\circ \end{cases} \quad (11)$$

In addition, another type of constraints is that the parameters of the solved mechanism must be reasonable. Due to the form of the mechanism, these constraints are only for the outboard flap. Specifically, the outboard flap has four mechanism constraints [61], as shown in Eq. (12):

$$\text{Mechanism Constraints: } \begin{cases} \text{Distance from the } D^i(D^o) \text{ to the lower surface of the flap} > 100 \\ \text{Distance from the } N^i(N^o) \text{ to the lower surface of the flap} > 100 \\ \alpha^i(\alpha^o) > 75^\circ \\ \beta^i(\beta^o) < 15^\circ \end{cases} \quad (12)$$

non and increase the noise [10]. Four parameters are used to describe the level of the misalignment, which are defined in the following:

1) ΔX represents the difference between the leading-edge X coordinates of the two flaps.

2) ΔY represents the difference between the maximum Y coordinates of the two flaps.

3) $\Delta \beta_{XZ}$ represents the angle of the two flaps' chord projected to the XZ plane.

4) $\Delta \beta_{XY}$ represents the angle of the two flaps' chord projected to the XY plane.

The four constraints are represented by the local chord length c of the kink section, and all the spanwise constraints are shown in Eq. (11):

where D and N are hinges attached to the outboard flap, with superscripts i for inside and o for outside, as shown in Fig. 11. Note that α is defined as the angle between the red plane determined by DNE and the green plane tangent to the lower surface of the flap, as shown in Fig. 11a. Note that β is defined as the angle between the red plane determined by DNE and the blue plane parallel to the symmetrical plane of the fuselage, as shown in Fig. 11b.

B. Application of VS-CGAN and TL-DSDRL

First, VS-CGAN is used to compress the variable domains aiming to satisfy the constraints proposed in Eqs. (11) and (12). In this process, there is no aerodynamic performance evaluation, so that a large number of samples can be collected.

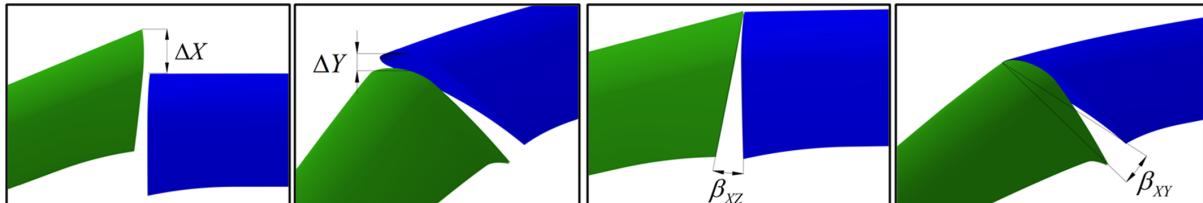


Fig. 10 Schematic diagram of the four parameters.

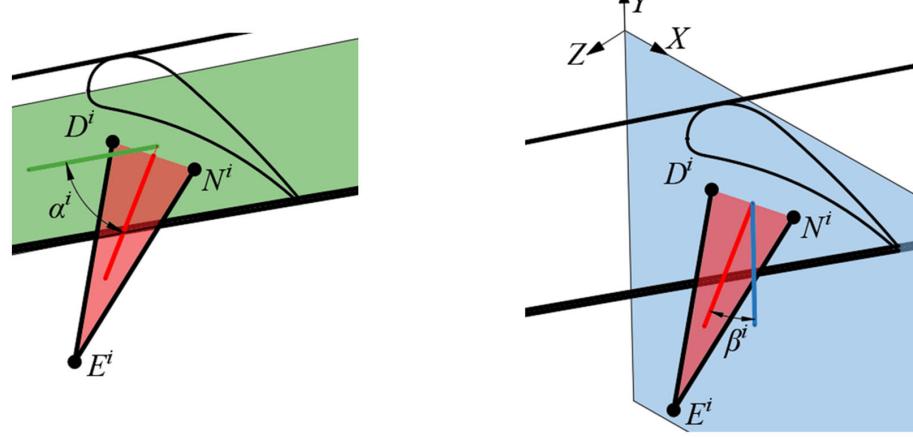


Fig. 11 Schematic diagram of the mechanism constraints [61].

Two generators are used to produce the takeoff and landing variables of the outboard flap, respectively. Take the takeoff configuration as an example. The inboard optimization variables are regarded as the condition vector \mathbf{C} , as shown in Eq. (13). The variables of the outboard flap are regarded as the data $\boldsymbol{\delta}$, as shown in Eq. (14), which is generated by the generator. Each element in \mathbf{C} and $\boldsymbol{\delta}$ is normalized between 0 and 1. The dimension of a noise vector λ in Eq. (15) is the same as that of $\boldsymbol{\delta}$.

$$\mathbf{C} = \text{Normalize}\{[X_{\text{inboard,inner}}, Y_{\text{inboard,inner}}, X_{\text{inboard,outer}}, Y_{\text{inboard,outer}}, \theta_{\text{inboard}}^{\text{takeoff}}, \phi_{\text{inboard}}^{\text{takeoff}}]\} = [C_1, C_2, \dots, C_6] \quad \forall C_i \in [0, 1] \quad (13)$$

$$\boldsymbol{\delta} = \text{Normalize}\{[X_{\text{outboard,inner}}^{\text{takeoff}}, Y_{\text{outboard,inner}}^{\text{takeoff}}, X_{\text{outboard,outer}}^{\text{takeoff}}, Y_{\text{outboard,outer}}^{\text{takeoff}}, \theta_{\text{outboard}}^{\text{takeoff}}]\} = [\delta_1, \delta_2, \dots, \delta_5] \quad \forall \delta_i \in [0, 1] \quad (14)$$

$$\lambda = [\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5] \quad \forall \lambda_i \in [0, 1] \quad (15)$$

The structures of each layer in the generator and the discriminator are shown in Table 1. The more complex the network structure, the better the fitting ability. But at the same time it is easier to overfit. The generator and the discriminator adopt a typical double-hidden layer and fully connected structure. For the learning rate, a smaller value will slow the network convergence, but a larger one will fail to converge to an optimal parameter. For GAN algorithms, the learning rates of generator and discriminator should be similar, so that the convergence processes are synchronized. Based on similar studies [23,24,31], the learning rates for both are set at 0.0005. For more complex problems, a more complex network structure with a smaller learning rate may be required. A total of 40,000 episodes are executed.

Then, TL-DSDRL is conducted to solve the optimization problem. The structure of the agent is shown in Table 2. The learning rate of TL-DSDRL should be related to the absolute value of the reward. A larger absolute value of reward should be applied to a smaller learning rate. This paper uses a learning rate of 0.003 to match the reward $\sim \mathcal{O}(1)$. The action is selected by sampling in the normal distribution. The distribution takes the output of the actor net as an average with the variance decaying from 0.4 to 0.15. This will increase the random search in the early stage to find more complete Pareto front and focus more on local search in the later stage.

The proposed TL-DSDRL uses the states of the actors to approximately replace the nondominated solution set based on all historical samples, and then the actors are guided to find other nondominated

solutions. Therefore, too few actor nets will result in incomplete Pareto solution set. However, there is no parameter transfer between each actor net. Too many actor nets will lead to the too many parameters need to be trained, which will lead to the decline of the optimization efficiency. In the example in this paper, there are 3 optimization objectives, and 12 actor nets are selected based on those. The critic net is to fit the action value function. Because of the

powerful nonlinear fitting ability of neural network, a single critic net is selected.

In general, the aerodynamic performance change of 3D can be approximated by the aerodynamic performance change of several sections, so that the 2D optimization can be applied to pretrain the agent. For a swept wing with infinite spanwise, the incoming flow can be divided into two parts at the leading edge, one is vertical to the leading edge and the other is parallel. The vertical airflow mainly contributes to the aerodynamic force [62,63]. The lift coefficient $Cl_{\text{Section } i}$ and the drag coefficient $Cd_{\text{Section } i}$ of the section i can be calculated from $Cl_{\text{Section } i}^n$ and $Cd_{\text{Section } i}^n$ of the normalized airfoil as

$$Cl_{\text{Section } i} = Cl_{\text{Section } i}^n \cos^2 \chi_L \quad (16)$$

$$Cd_{\text{Section } i} = Cd_{\text{Section } i}^n \cos^3 \chi_L \quad (17)$$

where χ_L is the sweep angle of the leading edge. To speed up the pretraining, the number of selected parts should be minimized on the basis that the 2D aerodynamic performance has the same trend with 3D. A single section is inaccurate to evaluate the trend of the 3D aerodynamic performance, because different 3D configurations may have the similar 2D section, as shown in Fig. 12.

Two sections can more accurately predict the 3D position change of the flap. Therefore, for two independent flaps, four takeoff configuration sections and four landing configuration sections need to be evaluated. In this optimization problem, the spanwise continuity of the inboard and outboard flaps at kink must be required, so they can share a same section. Figure 13 shows the three selected sections, in which Secs. I and III are located in the middle spanwise of the two flaps, and Sec. II is close to the kink. The aerodynamic performance linear scalarization of the three sections is used to approximately evaluate the 3D aerodynamic performance, as shown in Eq. (18), where $c_{\text{Section } i}$ is the local chord length of the cruise configuration section.

$$Cl = \frac{\sum_{i=1}^3 c_{\text{Section } i} \cdot Cl_{\text{Section } i}}{\sum_{i=1}^3 c_{\text{Section } i}}, \quad Cd = \frac{\sum_{i=1}^3 c_{\text{Section } i} \cdot Cd_{\text{Section } i}}{\sum_{i=1}^3 c_{\text{Section } i}}, \\ K = Cl/Cd \quad (18)$$

Table 1 Structure of VS-CGAN networks

Layer name	Neurons	Activate function
<i>Generator</i>		
Hidden layer 1	30	ReLU
Hidden layer 2	30	ReLU
Output layer	5	Sigmoid
<i>Discriminator</i>		
Hidden layer 1	45	ReLU
Hidden layer 2	45	ReLU
Output layer	1	Tanh

Table 2 Structure of TL-DSDRL networks

Layer name	Neurons	Activate function
<i>Actor</i>		
Hidden layer 1	256	ReLU
Hidden layer 2	256	ReLU
Output layer	19	Tanh
<i>Critic</i>		
Hidden layer 1	256	ReLU
Hidden layer 2	256	ReLU
Output layer	1	None

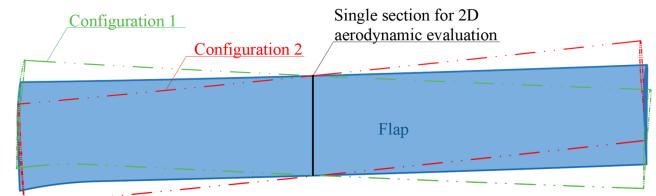


Fig. 12 Comparison of 2D section change of different 3D configurations.

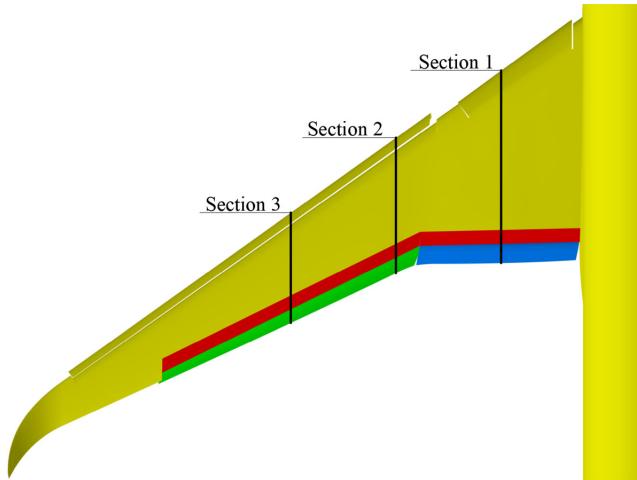


Fig. 13 Schematic diagram of three sections selected.

C. CFD Validation

After TL, the accurate aerodynamic performance evaluation of 3D CFD is essential for the optimization, so that the validation of the CFD method is conducted in this subsection. Firstly, the aerodynamic performance of the CRM-HL model published at the 4th AIAA CFD High Lift Prediction Workshop (HLPW-4) [64] is verified. The inflow conditions are $M = 0.2$ and $Re = 5.49 \times 10^6$. The commercial software ANSYS ICEM (ICEM is one module of ANSYS software) is used to generate the structure grid with the total of 155 million nodes, as shown in Fig. 14. A cuboid domain based on the mean aerodynamic chord (MAC) is used for simulation. The downstream side of the domain is 200 times greater than the MAC, while the other sides are 150 times. The opening boundary condition with specified orthogonal velocities is used for the far field, and the symmetry plane of the model is set as a symmetry boundary. The no-slip boundary condition is used for the walls, and the grid height of the first layer near the walls meets $y^+ = 1$. The Reynolds-averaged Navier-Stokes (RANS) equations with shear-stress transport $k-\omega$

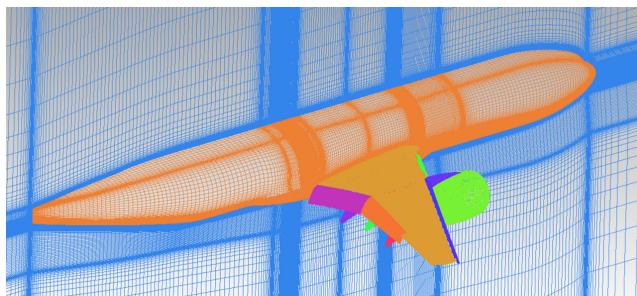
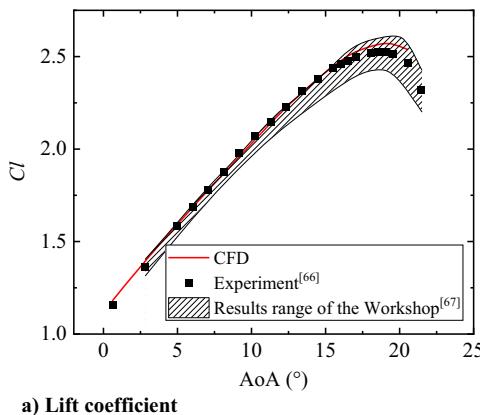


Fig. 14 Grid of the CRM-HL model.



a) Lift coefficient

turbulence model are solved to predict the aerodynamic performance by ANSYS CFX software. The solver is based on the element vertex finite volume method and employs a pseudotransient relaxation method to predict the steady flow. A high-resolution upwind advection scheme [65] is adopted.

The simulated aerodynamic performance is compared with the experimental data [66] and the RANS results published from the workshop [67] in Fig. 15. In the range of medium AoA ($AoA = 5\text{--}15^\circ$), the simulated lift coefficient is in good agreement with the experimental one. In the other range, the predicted lift coefficient is larger, which may result from the inaccuracy of the RANS method predicting the separation. For the high-lift configuration, the flap gap obscured by the main element at a small AoA may result in separation, while the separation is also easy to occur at a large AoA due to the negative pressure gradient. Similarly, the drag coefficient agrees well with the experimental results at small AoA and medium AoA and deviates at large AoA. However, all the simulated results are completely located in the gray-shaded area; i.e., the results envelope from the workshop, verifying the accuracy of the numerical method used in this paper.

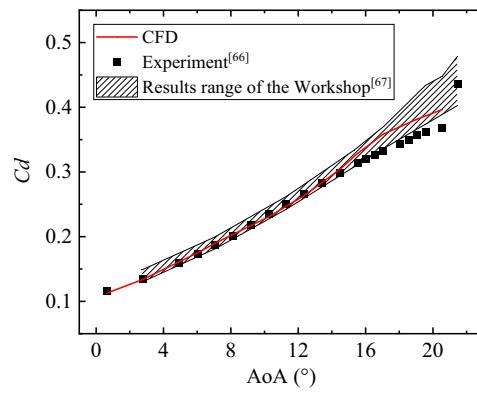
Then the grid independence is conducted. For the takeoff and landing configurations, three grids with different numbers are, respectively, generated as the coarse, medium, and fine presented in Table 3. The simulated results are shown in Fig. 16. For the takeoff configuration, the results from three grids are similar, except that the lift coefficient predicted by the fine grid is larger in the stall region. For the landing configuration, the coarse grid predicts a relatively larger lift at a small AoA, and the stall occurs ahead, while the other two grids match well with each other. In this study, the optimization point of AoA is in the range of $6^\circ \text{--} 9^\circ$. Thus, the coarse grid is used for the optimization to reduce cost, and the medium grid is used for the aerodynamic performance analysis. The adaptive mesh is constructed by ICEM scripts [68].

IV. Results and Discussion

The proposed multidisciplinary and multi-objective optimization method is used to solve the above optimization problem. In this section, the compression effect of VS-CGAN on the variable domains is discussed. Subsequently, the optimization efficiency of TL-DSDRL, which is pretrained by 2D optimization, is compared with the direct

Table 3 Summary of the grids

Type	Number of nodes	y^+	Near-wall growth rate
Coarse	9,608,026 (takeoff configuration)	2.0	1.22
	9,956,073 (landing configuration)		
Medium	28,756,845 (takeoff configuration)	1.5	1.15
	27,880,192 (landing configuration)		
Fine	82,176,895 (takeoff configuration)	1.0	1.08
	83,563,806 (landing configuration)		



b) Drag coefficient

Fig. 15 Comparison of the simulated aerodynamic performance with the published one.

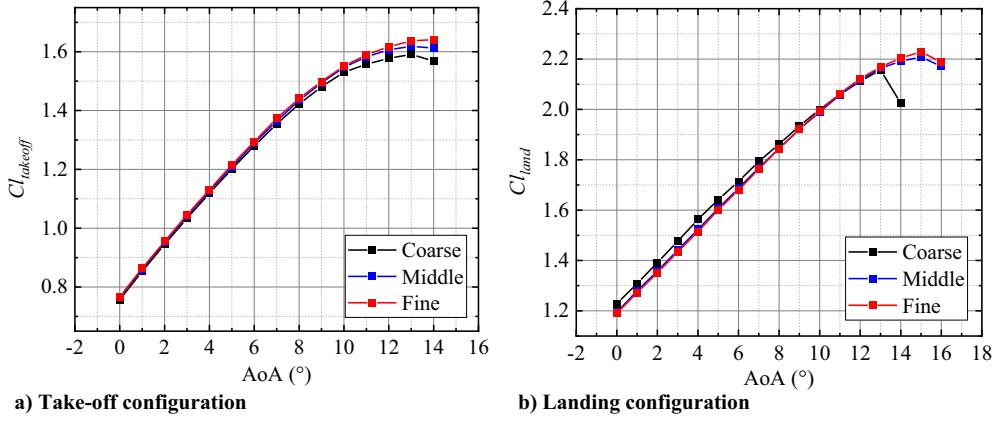


Fig. 16 Comparison of the lift coefficient from different grids.

optimization. Finally, the aerodynamic performance of the optimized configuration is analyzed, and the feasibility of the guiding mechanism is verified.

First, the compression effect of VS-CGAN on the variable domains is tested. The movements of the inboard and the outboard flaps are complex spatial motions, and it is difficult to evaluate whether they meet the spanwise continuity at the kink position from a single variable. However, the relationship between the deflection angles is relatively obvious; i.e., the deflection angle of the outboard flap should maintain a positive correlation with that of the inboard flap. Therefore, the consistency of the deflection angles of the inboard and outboard flaps is essential. Consider the takeoff configuration as an example; LHS is carried out in the whole optimization variable domain. If the variables of the outboard flap are directly sampled, as shown in Fig. 17a, only a very small number of samples (8.3%) meet the

spanwise continuity. This means that in the optimization process, the optimization algorithm will get a lot of negative feedback, which will lead to divergence. As a comparison, the noise vector λ is sampled, and the actual parameters of the outboard flap are generated by VS-CGAN. The proportion of the samples meeting the constraints has increased to 43.4%, as shown in Fig. 17b, suggesting that VS-CGAN has learned the policy to meet the spanwise continuity. The deflection angle of the outboard flap is slightly larger than that of the inboard one, mainly because the outboard flap has a larger sweep angle and the deflection angle of its projection on the XY plane is smaller. Figure 18 shows the comparison of the landing samples between direct sampling and the VS-CGAN method. The proportion of samples meeting the constraints has increased from 5.5 to 28.5%. Compared with the takeoff samples, this value is lower because the flap spanwise continuity of the landing configuration is more difficult to satisfy.

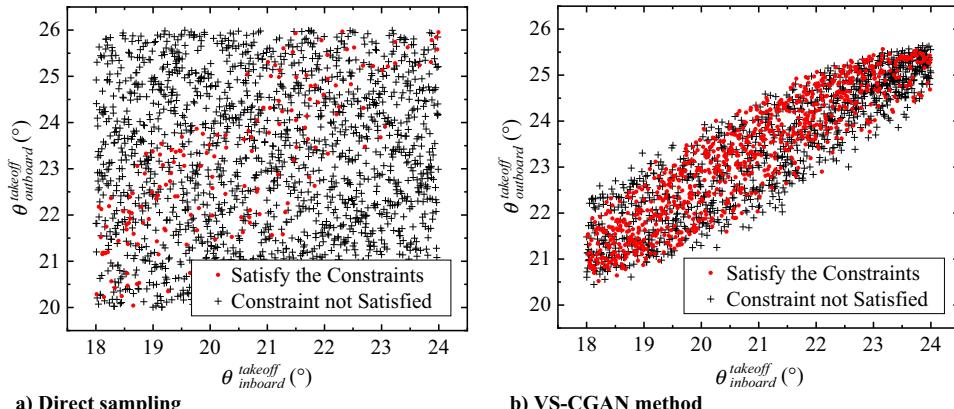


Fig. 17 Comparison of takeoff samples between direct sampling and VS-CGAN method.

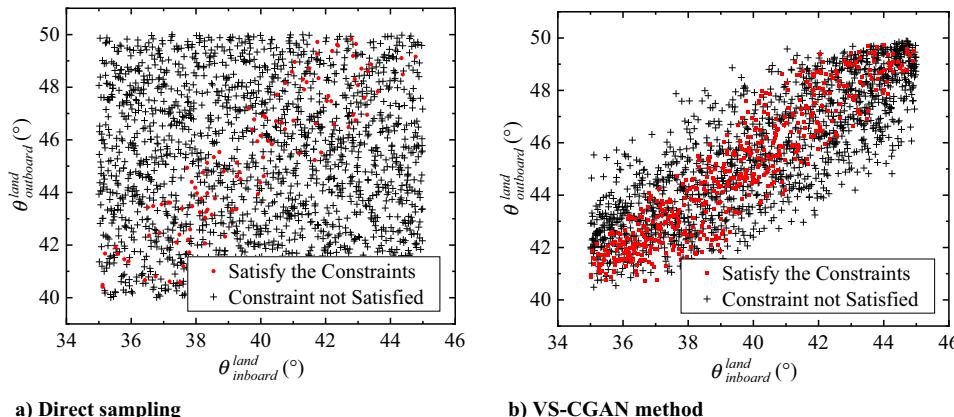
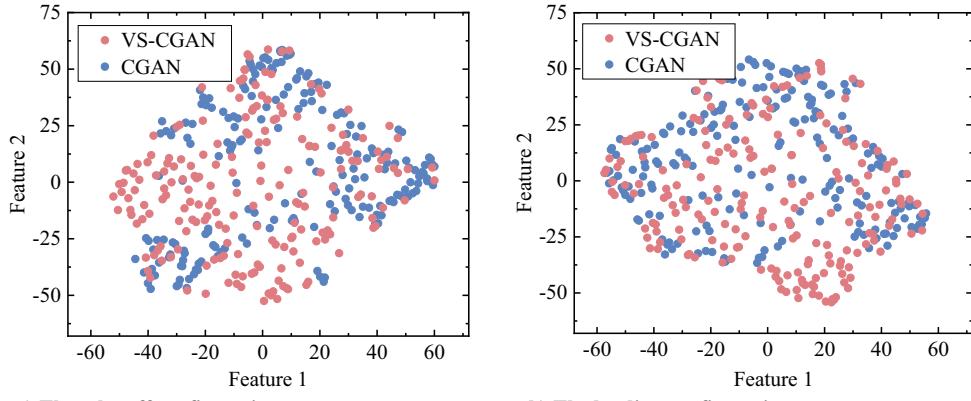


Fig. 18 Comparison of landing samples between direct sampling and VS-CGAN method.



a) The take-off configuration

b) The landing configuration

Fig. 19 Comparison of the samples between CGAN and VS-CGAN.

The richness of the data generated by VS-CGAN warrants further study. Data dimensionality reduction is performed on 200 sets of LHS sampling results using the t -distributed stochastic neighbor embedding (t-SNE) algorithm [69]. Figure 19 shows the comparison of takeoff and landing samples between CGAN and VS-CGAN. Compared to CGAN, VS-CGAN algorithm covers a wider space, indicating richer data content. The trained generator of VS-CGAN is applied in the optimization process.

It can be seen that VS-CGAN can convert a large number of constraints in MDO into a compression of the optimization variable domain, which greatly reduces the difficulty of the subsequent optimization algorithm and the possibility of divergence of the

optimization algorithm due to a large number of negative feedbacks. However, VS-CGAN can only be used for strictly restricted constraints. In aircraft design, different constraints may have different levels, and VS-CGAN cannot be used in cases where some constraints cannot be satisfied.

Secondly, the optimization process of TL-DSDRL is compared with the direct optimization of DSDRL. The 2D optimization is performed in 600 episodes as pretraining in TL-DSDRL, while the 3D optimization is performed in 600 episodes in TL-DSDRL and DSDRL, respectively. The variation process of the hypervolume is shown in Fig. 20. The reference point of the hypervolume is set at (0,0,0), and the three coordinates correspond to the three normalized optimization objectives. TL-DSDRL can obtain a larger hypervolume in the whole 600 episodes, i.e., a more complete Pareto front. Moreover, after 36 episodes, TL-DSDRL can obtain the final hypervolume of DSDRL. The 2D pretraining is performed using a personal computer equipped with a 12-core CPU at 3.70 GHz and consumes about 15 minutes per episode. As a comparison, the 3D optimization is carried out by a high-performance cluster equipped with a 512-core CPU at 2.25 GHz and consumes about 60 min. per episode. The results confirm that the TL of 2D optimization can save about 94% of the 3D computational cost with a similar performance in this problem.

To analyze the reason for the increase in optimization efficiency, the Pareto fronts generated by DSDRL and TL-DSDRL within 90 episodes are compared, as shown in Fig. 21. The takeoff performance of the samples is shown in Fig. 21a, and it can be seen that TL does not contribute to a significant improvement while the landing performance of the samples improves well, as shown in Fig. 21b. The reason is that the Pareto front, which only considers the takeoff performance, is simple. More precisely, the takeoff lift coefficient and the takeoff lift-to-drag ratio are negatively correlated. This leads to a high probability that a new sample, regardless of its landing

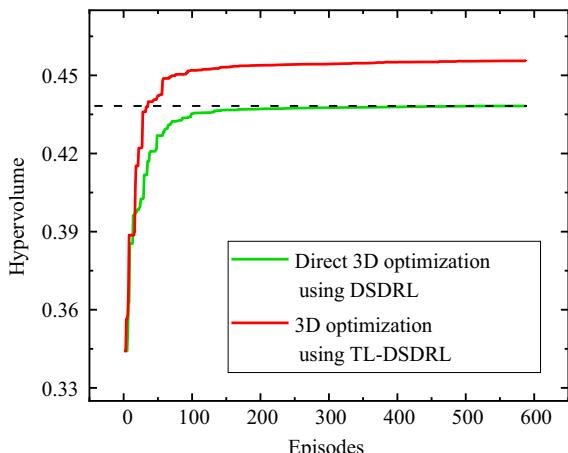
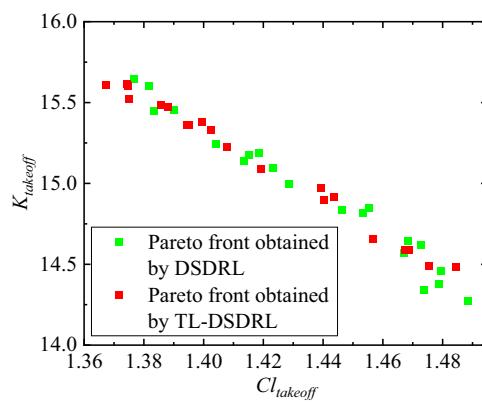
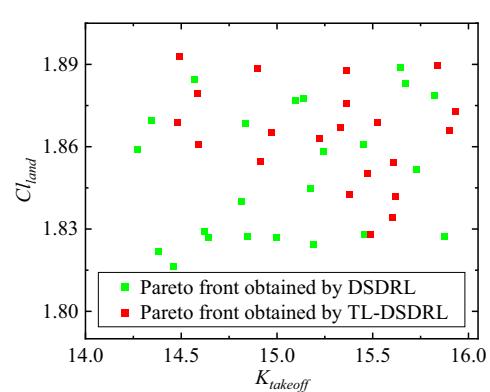


Fig. 20 Optimization processes of DSDRL and TL-DSDRL.



a) The distribution of the take-off lift-to-drag ratio and the take-off lift coefficient



b) The distribution of the landing lift coefficient and the take-off lift-to-drag ratio

Fig. 21 Comparison of the Pareto fronts obtained by DSDRL and TL-DSDRL within 90 episodes.

performance, will be a nondominated solution, i.e., at least one of the objectives is better than the existing samples, and get a positive reward. Thus, the optimization of landing performance is difficult in such problems, which is also a challenge for other multi-objective algorithms based on Pareto front with limited cost. By combining the low-order and high-order data in TL, TL-DSDRL achieves more efficient optimization. However, the process of TL is unexplained. Perhaps the prior knowledge of the physics embedded in the lower-order data and the higher-order data may achieve better results.

Thirdly, the sample distribution of TL-DSDRL optimization is analyzed. Figure 22 shows the distribution of the takeoff lift-to-drag ratio and the takeoff lift coefficient. The improvement in the takeoff configuration is relatively not large, especially considering the potential numerical errors, but this does not prevent the algorithm from finding a set composed of nondominant solutions. The solution set is helpful for engineers to select suitable configurations. The Pareto front shows an approximately straight line, indicating a simple trend between them at the design point. In contrast, the relationship between the landing performance and the takeoff performance is not clear, as shown in Figs. 23 and 24 namely, under the same takeoff performance, the landing performance always reaches the boundary. As analyzed above, it mainly results from the obvious negative relationship between the takeoff lift coefficient and the takeoff lift-to-drag ratio. From a mathematical point of view, more episodes may be required to obtain a small-gap Pareto front of the landing performance. Unfortunately, it is difficult to track the convergence due to the limited cost of the engineering design. A sample that has high performance of the three objectives is selected as the “optimized configuration.” Compared with the baseline configuration,

the takeoff lift coefficient of the optimized configuration is increased by 0.53%, the takeoff lift-to-drag ratio is increased by 1.03%, and the landing lift coefficient is increased by 1.95%.

Finally, the performance of the optimized configuration is presented and compared with the baseline configuration. The takeoff lift coefficient and the takeoff lift-to-drag ratio of the baseline configuration and the optimized configuration are shown in Fig. 25. The takeoff lift coefficient of the optimized configuration is obviously improved in the whole range of AoA, and the takeoff lift-to-drag ratio is also increased when the takeoff lift coefficient is larger than 1.4 for the requirement of the takeoff performance. The comparison of the streamline and the pressure coefficient distribution at 8° is shown in Fig. 26. The lift improvement mainly comes from the spoilers. The increase in the spoiler deflection angle leads to an increase in main wing circulation and negative pressure on the upper surface. The other part of the lift improvement comes from the enhancement of the flap suction peaks. It is noteworthy that there are two suction peaks. The first one is caused by the leading edge of the flap, and the second one is caused by the gap. Because the flap is driven by the hinge mechanism, the Fowler motion of its takeoff configuration is far less than that of the traditional Fowler flap, which leads to the two suction peaks. After optimization, the flap deflection angle increases, resulting in the enhancement of the first suction peak, while the change in gap width increases the second suction peak. Figure 27 shows that the stall of the takeoff configuration is due to the flow separation at the kink position. More specifically, the leading-edge high-lift device cannot be installed here because of the engine pylon. The optimized configuration does not restrain the separation zone, so that the stall AoA does not increase, while the increase of the circulations at other spanwise positions leads to an increase in the maximum lift coefficient.

The comparison of the landing lift coefficient between the baseline configuration and the optimized configuration is shown in Fig. 28. At small and medium AoA, the landing lift coefficient of the optimized configuration is increased by about 0.03. However, the stall AoA and the maximum lift coefficient do not change significantly. The surface pressure coefficient distributions and streamlines at $\text{AoA} = 8^\circ$ and $\text{AoA} = 15^\circ$ are shown in Figs. 29 and 30 for further analysis. The lift improvement is mainly contributed by the increase in the downward deflection angle of the spoilers, which enhances the negative pressure at the spoiler shaft and increases the camber of the main element. This effect gradually increases from the wing root to the wing tip. At stall, there is an obvious flow separation region at the location where the engine pylon will be installed because there is no leading-edge high-lift device locally. The optimization does not improve the stall performance because the trailing-edge high-lift device basically does not affect the leading edge. Even so, the maximum lift coefficient of the optimized landing configuration exceeds 2.2, which is a typical value for a commercial aircraft [70].

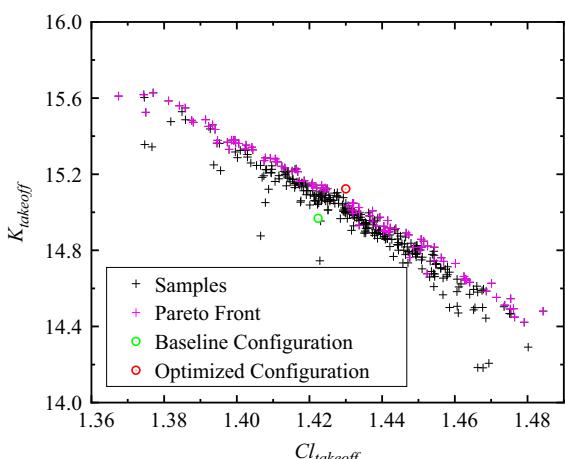


Fig. 22 The distribution of the takeoff lift-to-drag ratio and the takeoff lift coefficient.

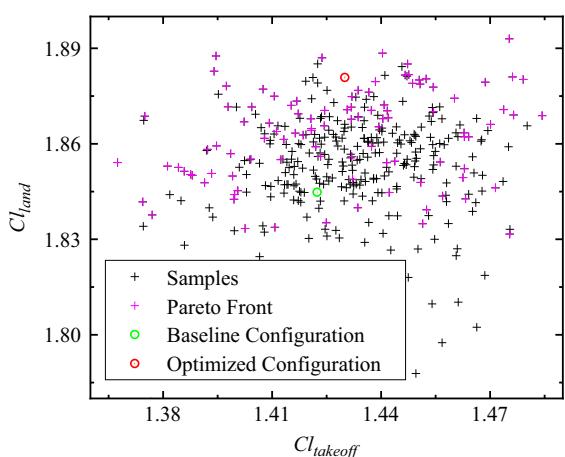


Fig. 23 The distribution of the landing lift coefficient and the takeoff lift coefficient.

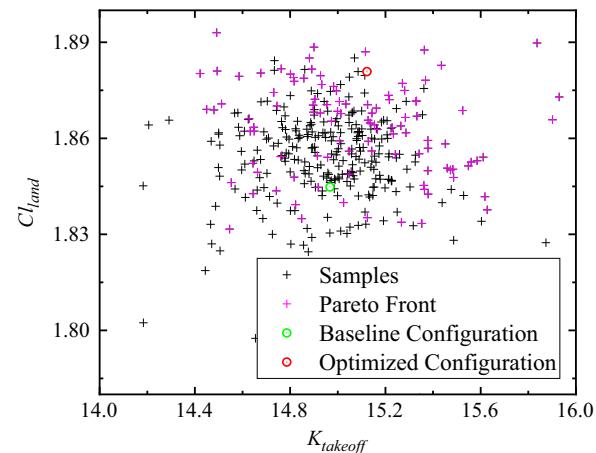
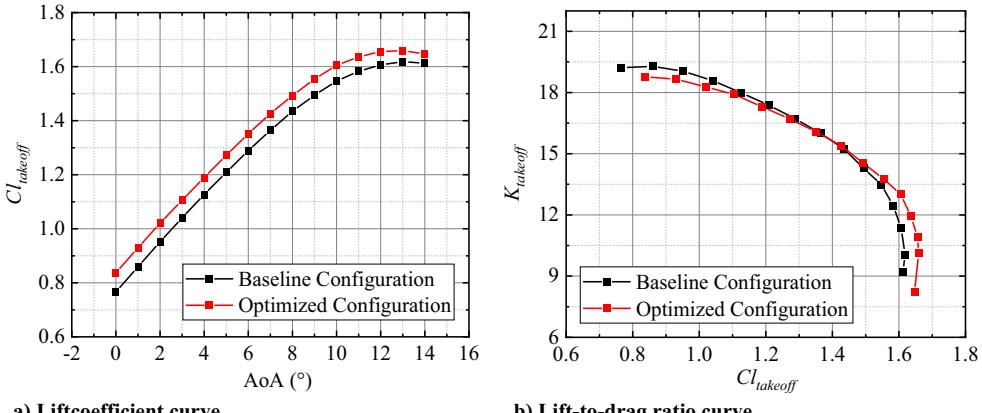


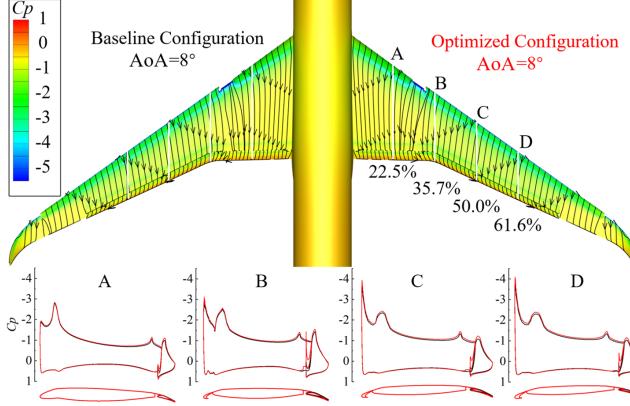
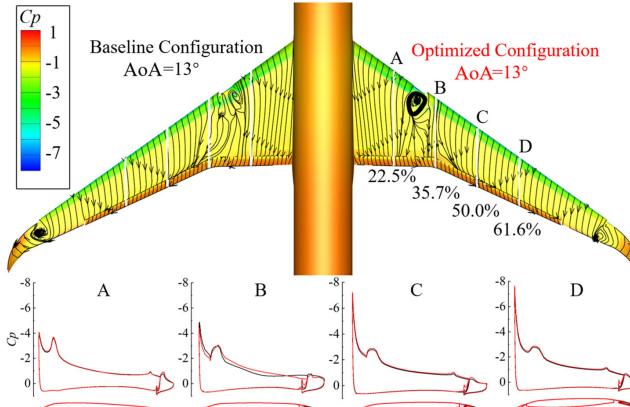
Fig. 24 The distribution of the landing lift coefficient and the takeoff lift-to-drag ratio.



a) Liftcoefficient curve

b) Lift-to-drag ratio curve

Fig. 25 Takeoff aerodynamic performance of the baseline configuration and the optimized configuration.

Fig. 26 Comparison of surface pressure and streamline of the takeoff configurations at AoA = 8 $^{\circ}$.Fig. 27 Comparison of surface pressure and streamline of the takeoff configurations at AoA = 13 $^{\circ}$.

The mechanism design results are modeled in CATIA (is a commercial software). The Digital Mock Up module is used to simulate the deployment process of the flap under the guidance of the mechanism, as shown in Fig. 31, where the black lines represent the rods of the mechanism. The results show that the designed mechanism can guide the flap to the optimized position accurately.

V. Conclusions

In this paper, a multidisciplinary and multi-objective optimization method is proposed. Specifically, depending on the cost of performance evaluation across disciplines, the method includes two novel algorithms to improve optimization efficiency. For low-cost

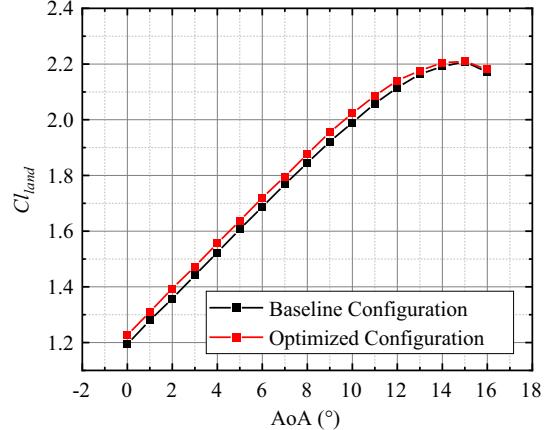
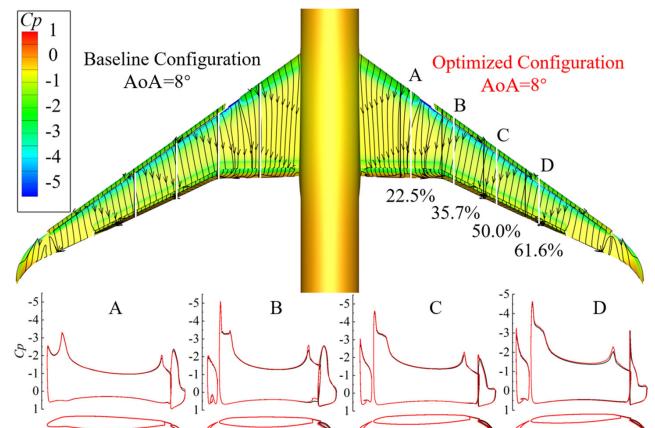


Fig. 28 Comparison of landing lift coefficient between baseline configuration and optimized configuration.

Fig. 29 Comparison of surface pressure and streamline of the landing configurations at AoA = 8 $^{\circ}$.

disciplines, VS-CGAN uses a large number of samples in advance to learn the relationship between variables and objectives, thus providing a more appropriate variable domain. The cosine function is used to describe the similarity between the random noise and the generated variables, and the physical properties of the random noise are given to avoid mode collapse. While for high-cost disciplines, TL-DSDRL uses the training experience of agents in low-precision prediction models to improve the optimization efficiency in high-precision prediction models through TL. The newly designed reward function and multi-agent cooperation mechanism improve the multi-objective search ability of RL.

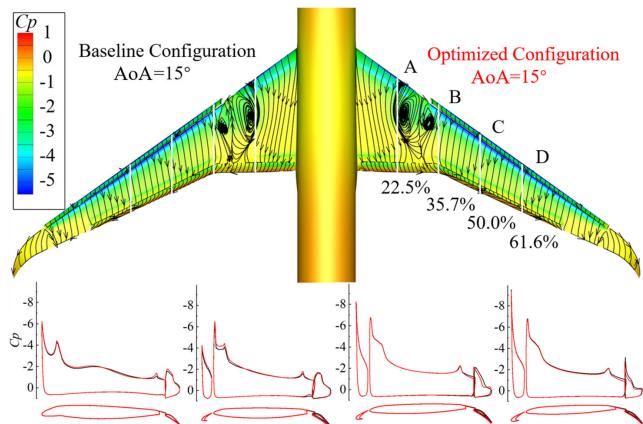


Fig. 30 Comparison of surface pressure and streamline of the landing configurations at $\text{AoA} = 15^\circ$.

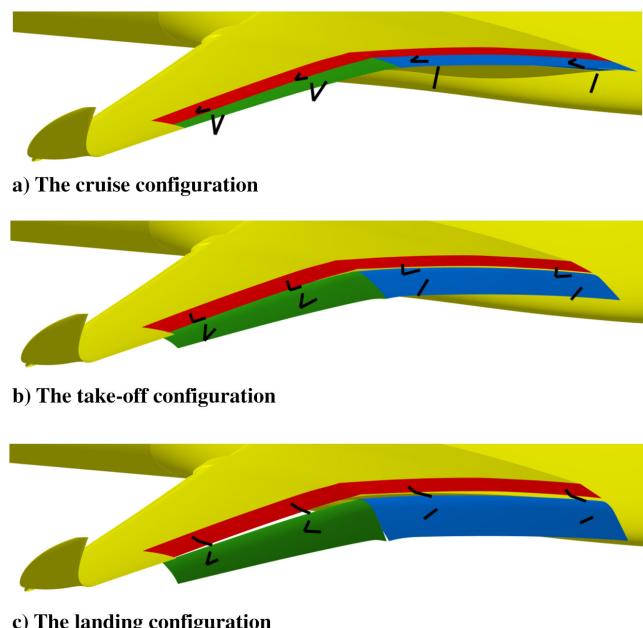


Fig. 31 Kinematics simulation of the designed mechanism.

Considering the aerodynamics-mechanism optimization problem of a wide-body airliner's high-lift device as an example, the proposed optimization method is tested. VS-CGAN can learn the synchronous motion law of the inboard and outboard flaps, and the proportion of samples meeting the requirements of mechanism and position has been greatly increased. Taking the hypervolume as the indicator, the agent after TL can obtain the same Pareto front at the evaluation cost of 6% or increase the hypervolume by 4% at the same cost.

Acknowledgment

This work is financially supported by the National Natural Science Foundation of China (Nos. 12272024 and 12002021).

References

- [1] Antoine, N. E., and Kroo, I. M., "Framework for Aircraft Conceptual Design and Environmental Performance Studies," *AIAA Journal*, Vol. 43, No. 10, 2005, pp. 2100–2109. <https://doi.org/10.2514/1.13017>
- [2] Bravo-Mosquera, P. D., Catalano, F. M., and Zingg, D. W., "Unconventional Aircraft for Civil Aviation: A Review of Concepts and Design Methodologies," *Progress in Aerospace Sciences*, Vol. 131, May 2022, Paper 100813. <https://doi.org/10.1016/j.paerosci.2022.100813>
- [3] Benouaali, A., and Kachel, S., "Multidisciplinary Design Optimization of Aircraft Wing Using Commercial Software Integration," *Aerospace Science and Technology*, Vol. 92, Sept. 2019, pp. 766–776. <https://doi.org/10.1016/j.ast.2019.06.040>
- [4] Reckzeh, D., "Aerodynamic Design of Airbus High-Lift Wings in a Multidisciplinary Environment," *European Congress on Computational Methods in Applied Science and Engineering*, Univ. of Jyvaskyla, Jyvaskyla, 2004, pp. 1–19.
- [5] Tian, Y., Quan, J. C., Liu, P. Q., Li, D. D., and Kong, C. H., "Mechanism/Structure/Aerodynamic Multidisciplinary Optimization of Flexible High-Lift Devices for Transport Aircraft," *Aerospace Science and Technology*, Vol. 93, Oct. 2019, Paper 104813. <https://doi.org/10.1016/j.ast.2018.09.045>
- [6] Liu, P. Q., Li, D. D., Qu, Q. L., and Kong, C. H., "Two-Dimensional New-Type High-Lift Systems with Link/Straight Track Mechanism Coupling Downward Deflection of Spoiler," *Journal of Aircraft*, Vol. 56, No. 4, 2019, pp. 1524–1533. <https://doi.org/10.2514/1.C035145>
- [7] Dobrzynski, W., "Almost 40 Years of Airframe Noise Research: What Did We Achieve?" *Journal of Aircraft*, Vol. 47, No. 2, 2010, pp. 353–367. <https://doi.org/10.2514/1.44457>
- [8] Casalino, D., Diozzi, F., Sannino, R., and Paonessa, A., "Aircraft Noise Reduction Technologies: A Bibliographic Review," *Aerospace Science and Technology*, Vol. 12, No. 1, 2008, pp. 1–17. <https://doi.org/10.1016/j.ast.2007.10.004>
- [9] PART 25: Airworthiness Standards: Transport Category Airplanes, Federal Aviation Administration, Washington, D.C., 2002.
- [10] Rudolph, P. K. C., "High-Lift Systems on Commercial Subsonic Airliners," NASA CR-4746, 1996.
- [11] Reckzeh, D., "Aerodynamic Design of the High-Lift-Wing for a Megaliner Aircraft," *Aerospace Science and Technology*, Vol. 7, No. 2, 2003, pp. 107–119. [https://doi.org/10.1016/S1270-9638\(02\)00002-0](https://doi.org/10.1016/S1270-9638(02)00002-0)
- [12] Yan, H., Yao, X. F., Lu, J. G., and Liu, J., "Transmission Efficiency of the Motion Mechanism in High-Lift Devices," *Journal of Aircraft*, Vol. 54, No. 4, 2020, pp. 1–12. <https://doi.org/10.2514/1.C035927>
- [13] Franke, D. M., "Aerodynamic Optimization of a High-Lift System with Kinematic Constraints," Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Vol. 121, Springer, Berlin, 2013, pp. 9–16. https://doi.org/10.1007/978-3-642-35680-3_2
- [14] Haftka, R. T., "Optimization of Flexible Wing Structures Subject to Strength and Induced Drag Constraints," *AIAA Journal*, Vol. 15, No. 8, 1977, pp. 1106–1177. <https://doi.org/10.2514/3.7400>
- [15] Martins, J. R. R. A., and Lambe, A. B., "Multidisciplinary Design Optimization: A Survey of Architectures," *AIAA Journal*, Vol. 51, No. 9, 2013, pp. 2049–2075. <https://doi.org/10.2514/1.J051895>
- [16] Cramer, E. J., Dennis, J. E., Jr., Frank, P. D., Lewis, R. M., and Shubin, G. R., "Problem Formulation for Multidisciplinary Optimization," *SIAM Journal on Optimization*, Vol. 4, No. 4, 1994, pp. 754–776. <https://doi.org/10.1137/0804044>
- [17] Martins, J. R. R. A., and Hwang, J. T., "Review and Unification of Methods for Computing Derivatives of Multidisciplinary Computational Models," *AIAA Journal*, Vol. 51, No. 11, 2013, pp. 2582–2599. <https://doi.org/10.2514/1.J052184>
- [18] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, 2002, pp. 182–197.
- [19] Eberhart, R., and Kennedy, J., "A New Optimizer Using Particle Swarm Theory," *Proceedings of The 6th International Symposium on Micro Machine and Human Science*, Inst. of Electrical and Electronics Engineers, New York, 1995, pp. 39–43.
- [20] Steinbrunn, M., Moerkotte, G., and Kemper, A., "Heuristic and Randomized Optimization for the Join Ordering Problem," *International Journal on Very Large Data Bases*, Vol. 6, No. 3, 1997, pp. 191–208. <https://doi.org/10.1007/s007780050040>
- [21] Li, J. C., Du, X. S., and Martins, J. R. R. A., "Machine Learning in Aerodynamic Shape Optimization," *Progress in Aerospace Sciences*, Vol. 134, Oct. 2022, Paper 100849. <https://doi.org/10.1016/j.paerosci.2022.100849>
- [22] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., "Generative Adversarial Networks," *Communications of the ACM*, Vol. 63, No. 11, 2020, pp. 139–144. <https://doi.org/10.1145/3422622>
- [23] Du, X. S., He, P., and Martins, J. R. R. A., "Rapid Airfoil Design Optimization via Neural Networks-Based Parameterization and Surrogate Modeling," *Aerospace Science and Technology*, Vol. 113, June

- 2021, Paper 106701.
<https://doi.org/10.1016/j.ast.2021.106701>
- [24] Chen, W., and Fuge, M., "Synthesizing Designs with Interpart Dependencies Using Hierarchical Generative Adversarial Networks," *Journal of Mechanical Design*, Vol. 141, No. 11, 2019, Paper 111403.
<https://doi.org/10.1115/DETC2018-85339>
- [25] Li, J. C., Zhang, M. Q., Tay, C. M. J., Liu, N. Y., Cui, Y. D., Chew, S. C., and Khoo, B. C., "Low-Reynolds-Number Airfoil Design Optimization Using Deep-Learning-Based Tailored Airfoil Modes," *Aerospace Science and Technology*, Vol. 121, Feb. 2022, Paper 107309.
<https://doi.org/10.1016/j.ast.2021.107309>
- [26] Li, J. C., and Zhang, M. Q., "On Deep-Learning-Based Geometric Filtering in Aerodynamic Shape Optimization," *Aerospace Science and Technology*, Vol. 112, May 2021, Paper 106603.
<https://doi.org/10.1016/j.ast.2021.106603>
- [27] Li, J. C., and Zhang, M. Q., "Efficient Aerodynamic Shape Optimization with Deep-Learning-Based Geometric Filtering," *AIAA Journal*, Vol. 58, No. 10, 2020, pp. 4243–4259.
<https://doi.org/10.2514/1.J059254>
- [28] Li, J. C., and Zhang, M. Q., "Adjoint-Free Aerodynamic Shape Optimization of the Common Research Model Wing," *AIAA Journal*, Vol. 59, No. 6, 2021, pp. 1990–2000.
<https://doi.org/10.2514/1.J059921>
- [29] Mirza, M., and Osindero, S., "Conditional Generative Adversarial Nets," arXiv (preprint), Nov. 2014, pp. 2672–2680.
<https://doi.org/10.48550/arXiv.1411.1784>
- [30] Yilmaz, E., and German, B. J., "Conditional Generative Adversarial Network Framework for Airfoil Inverse Design," *AIAA Aviation Forum*, AIAA Paper 2020-3185, 2020.
<https://doi.org/10.2514/6.2020-3185>
- [31] Wu, P., Yuan, W. Y., Ji, L. L., Zhou, L., Zhou, Z., Feng, W. B., and Guo, Y. K., "Missile Aerodynamic Shape Optimization Design Using Deep Neural Networks," *Aerospace Science and Technology*, Vol. 126, July 2022, Paper 107640.
<https://doi.org/10.1016/j.ast.2022.107640>
- [32] Arjovsky, M., Chintala, S., and Bottou, L., "Wasserstein GAN," arXiv (preprint), 2017.
<https://doi.org/10.48550/arXiv.1701.07875>
- [33] Pant, P., Joseph, J., and Omkar, S. N., "A Novel Framework for Multi-Objective Optimization of Airfoils Using Invasive Weed Optimization," *AIAA Aviation Forum*, AIAA Paper 2020-3118, 2020.
<https://doi.org/10.2514/6.2020-3118>
- [34] Yan, X. H., Zhu, J. H., Kuang, M. C., and Wang, X. Y., "Aerodynamic Shape Optimization Using a Novel Optimizer Based on Machine Learning Techniques," *Aerospace Science and Technology*, Vol. 86, March 2019, pp. 826–835.
<https://doi.org/10.1016/j.ast.2019.02.003>
- [35] Li, R. Z., Zhang, Y. F., and Chen, H. X., "Learning the Aerodynamic Design of Supercritical Airfoils Through Deep Reinforcement Learning," *AIAA Journal*, Vol. 59, No. 10, 2021, pp. 3988–4001.
<https://doi.org/10.2514/1.J060189>
- [36] Dai, J. H., Liu, P. Q., Qu, Q. L., Li, L., and Niu, T. Z., "Aerodynamic Optimization of High-Lift Devices Using a 2D-To-3D Optimization Method Based on Deep Reinforcement Learning and Transfer Learning," *Aerospace Science and Technology*, Vol. 121, Feb. 2022, Paper 107348.
<https://doi.org/10.1016/j.ast.2022.107348>
- [37] Moffaert, K. V., Drugan, M. M., and Nowe, A., "Scalarized Multi-Objective Reinforcement Learning: Novel Design Techniques," *IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*, Inst. of Electrical and Electronics Engineers, New York, 2013, pp. 191–199.
<https://doi.org/10.1109/ADPRL.2013.6615007>
- [38] Qin, S., Wang, S. Y., Wang, L. Y., Wang, C., Sun, G., and Zhong, Y. J., "Multi-Objective Optimization of Cascade Blade Profile Based on Reinforcement Learning," *Applied Science*, Vol. 11, No. 1, 2020, p. 106.
<https://doi.org/10.3390/app11010106>
- [39] Das, I., and Dennis, J. E., "A Closer Look at Drawbacks of Minimizing Weighted Sums of Objectives for Pareto Set Generation in Multicriteria Optimization Problems," *Structural and Multidisciplinary Optimization*, Vol. 14, Aug. 1997, pp. 63–69.
<https://doi.org/10.1007/BF01197559>
- [40] Issabekov, R., and Vamplew, P., "An Empirical Comparison of Two Common Multiobjective Reinforcement Learning Algorithms," *Advances in Artificial Intelligence*, Vol. 7691, Springer, Berlin, 2012, pp. 626–636.
https://doi.org/10.1007/978-3-642-35101-3_53
- [41] Gass, S., and Saaty, T., "The Computational Algorithm for the Parametric Objective Function," *Naval Research Logistics Quarterly*, Vol. 2, Nos. 1–2, 1955, p. 39.
<https://doi.org/10.1002/nav.3800020106>
- [42] Kim, S., Kim, I., and You, D., "Multi-Condition Multi-Objective Optimization Using Deep Reinforcement Learning," *Journal of Computational Physics*, Vol. 462, No. 1, 2022, Paper 111263.
<https://doi.org/10.1016/j.jcp.2022.111263>
- [43] Vamplew, P., Foale, C., and Dazeley, R., "The Impact of Environmental Stochasticity on Value-Based Multiobjective Reinforcement Learning," *Neural Computing and Applications*, Vol. 34, Feb. 2020, pp. 1783–1799.
<https://doi.org/10.1007/s00521-021-05859-1>
- [44] Barrett, L., and Narayanan, S., "Learning All Optimal Policies with Multiple Criteria," *Proceedings of the 25th International Conference on Machine Learning*, Assoc. for Computing Machinery, New York, 2008, pp. 41–47.
<https://doi.org/10.1145/1390156.1390162>
- [45] Lizotte, D. J., Bowling, M., and Murphy, S. A., "Efficient Reinforcement Learning with Multiple Reward Functions for Randomized Controlled Trial Analysis," *Journal of Machine Learning Research*, Vol. 13, 2012, pp. 3253–3295.
<https://doi.org/10.5555/3104322.3104411>
- [46] Gabor, Z., Kalmar, Z., and Szepesvari, C., "Multi-Criteria Reinforcement Learning," *Proceedings of the 15th International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., San Francisco, 1998, pp. 197–205.
- [47] Geibel, P., "Reinforcement Learning for MDPs with Constraints," *Proceedings of the 17th European Conference on Machine Learning*, Springer, Berlin, 2006, pp. 646–653.
https://doi.org/10.1007/11871842_63
- [48] Moffaert, K. V., and Nowe, A., "Multi-Objective Reinforcement Learning Using Sets of Pareto Dominating Policies," *Journal of Machine Learning Research*, Vol. 15, No. 1, 2014, pp. 3663–3692.
- [49] Moffaert, K. V., Drugan, M. M., and Nowe, A., "Hypervolume-Based Multi-Objective Reinforcement Learning," *Evolutionary Multi-Criterion Optimization*, Springer, Berlin, 2013, pp. 352–366.
https://doi.org/10.1007/978-3-642-37140-0_28
- [50] Kingma, D. P., and Ba, J., "Adam: A Method for Stochastic Optimization," arXiv (preprint), Jan. 2017.
<https://doi.org/10.48550/arXiv.1412.6980>
- [51] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, 2002, pp. 182–197.
<https://doi.org/10.1109/4235.996017>
- [52] Fonseca, C. M., Paquete, L., and Lopez-Ibanez, M., "An Improved Dimension-Sweep Algorithm for the Hypervolume Indicator," *IEEE Congress on Evolutionary Computation*, Inst. of Electrical and Electronics Engineers, New York, 2006, pp. 1157–1163.
<https://doi.org/10.1109/CEC.2006.1688440>
- [53] Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D., "Continuous Control with Deep Reinforcement Learning," arXiv (preprint), July 2019.
<https://doi.org/10.48550/arXiv.1509.02971>
- [54] Fonseca, C. M., and Fleming, P. J., "An Overview of Evolutionary Algorithms in Multiobjective Optimization," *Evolutionary Computation*, Vol. 3, No. 1, 1995, pp. 1–16.
<https://doi.org/10.1162/evco.1995.3.1.1>
- [55] Dai, J. H., Liu, P. Q., and Li, L., "Multi-Objective Aerodynamic Optimization of 2d High-Lift Device Based on Distributed Deep Reinforcement Learning," *33rd Congress of the International Council of the Aeronautical Sciences*, ICAS, Stockholm, 2022, pp. 1–13.
- [56] Mark, H., U.S. Patent Application for "Wing of an Aircraft or Spacecraft, Comprising a Mobile Flow Body," Docket No. PCT/EP2010/054338, filed 14 Oct. 2010.
- [57] Andreani, L., U.S. Patent Application for "Lateral Coupling Device for Holding and Guiding at Least One Aerodynamic Body in Relation to the Main Wing of an Aircraft, Wing and Aircraft Comprising the Same," Docket No. PCT/EP2009/002964, filed 23 April 2009.
- [58] Ji, Q., Zhang, Y. F., Chen, H. X., and Ye, J. K., "Aerodynamic Optimization of a High-Lift System with Adaptive Dropped Hinge Flap," *Chinese Journal of Aeronautics*, Vol. 35, No. 11, 2022, pp. 191–208.
<https://doi.org/10.1016/j.cja.2022.03.012>
- [59] Scholz, P., Mahmood, S. S., Casper, M., Stefan, W., Skoogh, D., and Adden, S., "Design of Active Flow Control at a Dropped Spoiler Configuration," *31st AIAA Applied Aerodynamics Conference*, AIAA Paper 2013-2518, 2013.
<https://doi.org/10.2514/6.2013-2518>
- [60] Wang, W. H., Liu, P. Q., Tian, Y., and Qu, Q. L., "Numerical Study of the Aerodynamic Characteristics of High-Lift Droop Nose with the Deflection of Fowler Flap and Spoiler," *Aerospace Science and Technology*,

- Vol. 48, Jan. 2016, pp. 75–85.
<https://doi.org/10.1016/j.ast.2015.10.024>
- [61] Dai, J. H., Liu, P. Q., Chen, Y. C., Wen, X. L., Li, L., and Dong, M., “Design and Optimization of a Hinged Mechanism Guiding Flap to Three Positions,” *Journal of Aircraft*, Vol. 60, No. 3, 2023, pp. 883–897.
<https://doi.org/10.2514/1.C036902>
- [62] Bourgault-Cote, S., Ghasemi, S., Mosahebi, A., and Laurendeau, E., “Extension of a Two-Dimensional Navier-Stokes Solver for Infinite Swept Flow,” *AIAA Journal*, Vol. 55, No. 2, 2017, pp. 662–667.
<https://doi.org/10.2514/1.J055139>
- [63] Wild, J., Brezillon, J., Mertins, R., Quagliarella, D., Germain, E., Amoignon, O., and Moens, F., “Realistic High-Lift Design of Transport Aircraft by Applying Numerical Optimization,” *European Computational Fluid Dynamics*, TU Delft, Delft, The Netherlands, 2006, pp. 1–16.
- [64] Lacy, D., and Clark, A. M., “Definition of Initial Landing and Takeoff Reference Configurations for the High Lift Common Research Model (CRM-HL),” *AIAA Aviation Forum*, AIAA Paper 2020-2771, 2020.
<https://doi.org/10.2514/6.2020-2771>
- [65] Barth, T. J., and Jespersen, D. C., “The Design and Application of Upwind Schemes on Unstructured Meshes,” *27th Aerospace Sciences Meeting*, AIAA Paper 1989-0366, 1989.
<https://doi.org/10.2514/6.1989-366>
- [66] Evans, A., Lacy, D., Smith, I., and Rivers, M., “Test Summary of the NASA Semi-Span High-Lift Common Research Model at the QinetiQ 5-Metre Low-Speed Wing Tunnel,” *AIAA Aviation Forum*, AIAA Paper 2020-2770, 2020.
<https://doi.org/10.2514/6.2020-2770>
- [67] Ollivier-Gooch, C. F., and Coder, J. G., “Lessons Learned by the Fixed-Grid RANS TFG for HLPW-4/GMGW-3,” *AIAA Aviation Forum*, AIAA Paper 2022-3211, 2022.
<https://doi.org/10.2514/6.2022-3211>
- [68] ANSYS ICEM CFD Programmer’s Guide Release 18.2, ANSYS, Inc., Canonsburg, PA, 2017.
- [69] Maaten, L. V. D., and Hinton, G., “Visualizing Data Using t-SNE,” *Journal of Machine Learning Research*, Vol. 9, No. 11, 2008, pp. 2579–2605.
- [70] Ting, E., Chaparro, D., Nguyen, N., and Fujiwara, G. E. C., “Optimization of Variable-Camber Continuous Trailing-Edge Flap Configuration for Drag Reduction,” *Journal of Aircraft*, Vol. 55, No. 6, 2018, pp. 2217–2239.
<https://doi.org/10.2514/1.C034810>

S. Brunton
Associate Editor