



Article

Asymmetric Airfoil Morphing via Deep Reinforcement Learning

Kelin Lu ¹, Qien Fu ^{1,*}, Rui Cao ^{2,3}, Jicheng Peng ¹ and Qianshuai Wang ⁴¹ School of Automation, Southeast University, Nanjing 210096, China² College of Information Engineering (Artificial Intelligence), Yangzhou University, Yangzhou 225009, China³ College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China⁴ School of Electrical Engineering, North China University of Science and Technology, Tangshan 063210, China

* Correspondence: qienfusc@gmail.com

Abstract: Morphing aircraft are capable of modifying their geometry configurations according to different flight conditions to improve their performance, such as by increasing the lift-to-drag ratio or reducing their fuel consumption. In this article, we focus on the airfoil morphing of wings and propose a novel morphing control method for an asymmetric deformable airfoil based on deep reinforcement learning approaches. Firstly, we develop an asymmetric airfoil shaped using piecewise Bézier curves and modeled by shape memory alloys. Resistive heating is adopted to actuate the shape memory alloys and realize the airfoil morphing. With regard to the hysteresis characteristics exhibited in the phase transformation of shape memory alloys, we construct a second-order Markov decision process for the morphing procedure to formulate a reinforcement learning environment with hysteresis properties explicitly considered. Subsequently, we learn the morphing policy based on deep reinforcement learning techniques where the accurate information of the system model is unavailable. Lastly, we conduct simulations to demonstrate the benefits brought by our learning implementations and validate the morphing performance of the proposed method. The simulation results show that the proposed method provides an average 29.8% performance improvement over traditional methods.



Citation: Lu, K.; Fu, Q.; Cao, R.; Peng, J.; Wang, Q. Asymmetric Airfoil Morphing via Deep Reinforcement Learning. *Biomimetics* **2022**, *7*, 188. <https://doi.org/10.3390/biomimetics7040188>

Academic Editors: Jiang Zhao, Xiangyin Zhang and Chunhe Hu

Received: 29 September 2022

Accepted: 31 October 2022

Published: 3 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: airfoil morphing; shape memory alloys; hysteresis; deep reinforcement learning

1. Introduction

While unmanned aerial vehicles (UAVs) have played a crucial role in various civil and military missions, studies demonstrate that birds usually possess higher flight maneuverability and agility than comparatively-sized aircraft in complex and varying environments [1,2]. One of the critical advantages of birds is that they morph their wings and tails intricately to perform efficient behaviors including perching, hovering and maintaining stability under different flight conditions [3]. Such aerodynamic adaptability has aroused flourishing interest in the design and control of avian-inspired morphing UAVs [4–6]. In this work, we focus on aircraft capable of morphing their wings by modifying the geometric configuration of the airfoil shape, which refers to more specifically camber morphing [7]. Bird wings are usually cambered to generate sufficient lift force at a low angle of attack. The camber does not remain constant through their flight, and many observations show that birds actively control the camber of their proximal wing via remiges and modify the distal wing airfoil shape via their primary feathers [8,9]. These investigations provide insight into the study of airfoil-morphing aircraft. The benefits brought by camber morphing for aircraft include increasing lift, reducing drag and airframe noise mitigation [10]. Applications of camber morphing mechanisms include a lead-edge morphing combined smart droop nose design, which achieves high-lift performance with significantly reduced complexity and mass [11], and a flexible morphing trailing edge design with deformable ribs, which is used to enhance the Fowler flaps and act as a substitution for ailerons for civil transport aircraft [12].

The ideal airfoil is usually generated using shape optimization techniques including gradient-based methods and gradient-free methods [13], which optimize some aerodynamic performance parameters of aircraft, such as the drag coefficient, the lift coefficient and the lift-to-drag ratio [14–16]. Recently, deep reinforcement learning (DRL) approaches such as proximal policy optimization (PPO) [17] have been exploited to learn the airfoil shape directly according to performance metrics computed by computational fluid dynamics (CFD) solvers [18]. In [19], a 3D-printed morphing airfoil model is developed, and the optimal configuration is generated via Q-learning to match the desired pitching moment. In [20], the transfer learning technique is combined with DRL for shape optimization, which formulates a multi-fidelity framework and reduces the computational cost.

Although an optimized airfoil shape can be calculated given a certain flight condition, it is challenging to morph into optimal shapes instantaneously during the flight procedure due to the uncertainty and inconsistency of the environments and aircraft dynamics [2,21]. This galvanizes the utilization of data-driven methods including deep learning and reinforcement learning for morphing control. In [22], the morphing air vehicle was modeled as a smart block in the shape of a rectangular parallelepiped, and the control policy was learned by actor-critic methods. This framework was extended to an ellipsoid-shaped aircraft with Q-learning methods to produce a more efficient policy [23]. The thickness and camber of the airfoil shape were adjusted via Q-learning in [24], where the rewards were related to aerodynamic parameters. The constant-strength source doublet panel method was utilized in [25] to calculate the aerodynamic forces on the morphing air vehicle. In [26], airfoil morphing by vertically moved control points was designed, where the unknown drag and lift coefficients were estimated using neural networks.

Concerning the realization of morphing mechanisms, biologically inspired mechanical joints are often adopted to control sweep-, dihedral- and twist-morphing wings [27–29]. However, the development and actuation for camber-morphing aircraft are more challenging, as they are required to permit smooth transitions in the airfoil shape [2]. Conventional actuators for camber morphing include servo motors and hydraulic actuators [30,31], where the wing is constructed by articulated rigid-linked components [32] or a deformable skin with internal compliant mechanisms [33]. Recent advances in material technologies have led to a proliferation of applications of smart materials, especially shape memory alloys (SMA), for morphing aircraft [34,35]. Shape memory alloys are metallic alloys capable of transforming their crystalline structures between two phases to deform and recover their shapes through heating and cooling, which is suitable for morphing actuators because of their properties, including high-actuation energy densities and large recoverable strains [35]. A morphing wing under subsonic cruise flight conditions was developed in [36] using a flexible skin and a group of SMA actuators, which reduced the fuel consumption. In [37], the authors designed an autonomous morphing helicopter rotor blade where the SMAs equipped on the blade modify the section camber according to ambient temperature. In [38], a super critical airfoil actuated by SMAs was developed, and the transonic aerodynamics are investigated. Nevertheless, to actively control the strain of SMA wires is not straightforward due to the temperature hysteresis exhibited during the phase transformation [39]. This gives rise to increasing investigations of learning-based control for SMAs. Reinforcement learning methods including Q-learning and Sarsa have been applied to adjust the strain of SMAs via resistive heating, where the hysteretic dynamics are modeled as hyperbolic tangent curves [40,41]. However, there is no explicit consideration for the hysteresis properties in the policy-learning procedure of these studies.

In this work, the morphing control for a deformable asymmetric airfoil based on deep reinforcement learning techniques is investigated. The contributions of this work are threefold, as follows. Firstly, the shape of the asymmetric airfoil is designed based on Bézier curves, and the morphing mechanism of such airfoils is modeled via SMA wires. Subsequently, a dynamic system between input voltages and airfoil shapes is developed, which characterizes the hysteresis behaviors in the manner of Markov decision processes. Finally, the morphing policy is constructed based on deep reinforcement learning

approaches without accurate knowledge of system models, which adjusts the airfoils during flight procedures to track the optimal shapes in different flight conditions.

The rest of this paper is organized as follows. In Section 2, the morphing method is developed in three steps. In Section 2.1, the asymmetric airfoil is shaped using piece-wise Bézier curves. In Section 2.2, the morphing airfoil is modeled by SMAs, and a dynamic model is derived. In Section 2.3, a deep reinforcement learning-based morphing policy is developed. In Section 3, simulations are conducted to validate the proposed methods. In Section 4, this work is summarized.

2. Materials and Methods

2.1. Asymmetric Airfoil Shape Modeling

Well-known methods of curve synthesis for airfoil design and optimization include splines (e.g., B-spline and Bézier curves) [42,43], free-form deformation (FFD) [44], and class-shape transformations (CST) [45]. In this work, Bézier curves are selected for their straightforward design procedure and simple calculations. The shape of the asymmetric airfoil is parameterized via N control points, which are connected by Bézier curves [18,20]. To generate an untangled shape, the control points are distributed in an annulus with a predefined inner radius R_1 and outer radius R_2 . Moreover, the annulus is partitioned into N sectors equally, in each of which a control point is placed. These points are sorted with respect to the azimuth and denoted as $p_i \in \mathbb{R}^2$ for $i = 1, \dots, N$ in Cartesian coordinates. Each pair of adjacent points is augmented by two other points and then connected via a cubic Bézier curve. For each control point, p_i , an auxiliary angle, is calculated to determine the tangent to the curve at this point, which is given by

$$\theta_i^* = \alpha_i \theta_{i-1,i} + (1 - \alpha_i) \theta_{i,i+1} \tag{1}$$

where $\theta_{i,i+1}$ is the angle between point p_i and p_{i+1} , and $\alpha_i \in [0, 1]$ is an averaging parameter to modify the local smoothness of the curve. Then, two augmented points for the curve between p_i and p_{i+1} are calculated by

$$\begin{aligned} p_i^* &= p_i + \eta_i \|p_{i+1} - p_i\| \cdot e_i \\ p_i^{**} &= p_{i+1} - \eta_i \|p_{i+1} - p_i\| \cdot e_{i+1} \end{aligned} \tag{2}$$

where $e_i = [\cos(\theta_i^*), \sin(\theta_i^*)]^\top$, and the scale parameter η_i controls the local curvature. The curve connecting p_i and p_{i+1} is given by

$$b(t) = (1 - t)^3 p_i + 3(1 - t)^2 t p_i^* + 3(1 - t) t^2 p_i^{**} + t^3 p_{i+1}, 0 \leq t \leq 1 \tag{3}$$

An example of a valid shape is illustrated in Figure 1. The morphing wings in this work take NACA-2424 [46,47] as the baseline airfoil.

2.2. Dynamic System of Airfoil Morphing

Denote the polar coordinate of each point p_i as $\{r_i, \theta_i\}_{i=1}^N$. Then the shape of the airfoil is fully determined by the radius r_i , the angle θ_i and the auxiliary angle parameters α_i, η_i . In the flight procedure of the aircraft, we aim to morph the airfoil to maximize desired aerodynamic performances, such as the lift-to-drag ratio C_l/C_d , according to various conditions, including flight position, velocity or attack angle. With the combination of CFD solvers and optimization methods, the preferred airfoil shape at a given flight condition can be determined previously. Such optimized shapes serve as the reference or target airfoil shapes for the morphing task. Then, the problem of how to control the airfoil to achieve optimized shapes during the flight procedure where the flight condition varies is investigated.

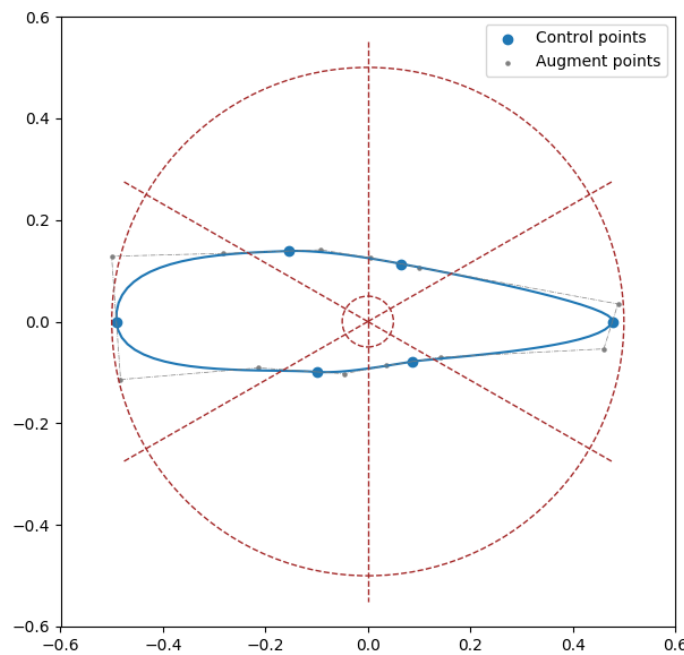


Figure 1. An illustration of a valid airfoil shape with six control points. The dashed circles denote the minimum and maximum radius for each point. The dashed rays split the annulus into N equal sections.

Since there has been a variety of investigations on the position control of DC motors [48], we assume that the polar angle of each control point tracks the reference trajectory well via motors. Additionally, we assume that the auxiliary angle parameters can be adjusted rapidly and accurately. Therefore, in this work, we focus on the optimal morphing in the aspect of modifying the radii of control points.

Smart materials, especially shape memory alloys (SMA), are adopted to realize the airfoil morphing, where the radii of control points are modified via adjusting the length of SMA wires. Firstly, the dynamic model is constructed between the wire temperature and radii. An SMA wire changes its length through the crystal phase transformation between martensite and austenite according to the temperature. The transitions to martensite and austenite have different start and end temperatures, which leads to the hysteresis properties of the strain with respect to the temperature. Instead of common methods such as Preisach model and Krasnosel’skii–Pokrovskii model [49], the SMA hysteresis is characterized using hyperbolic tangent functions for their efficiency in computation and accuracy in curve fitting [41]. The strain is replaced by a radius factor $\gamma_i = (r_i - R_1)/(R_2 - R_1)$ equivalently such that $\gamma_i \in [0, 1]$. For heating and cooling starting with temperatures outside the transformation region, namely that the initial temperature is not between the end temperatures of the phase transformations, the hysteresis properties are modeled by the major hysteresis loops as

$$f_l^{major}(T) = \frac{h_0}{2} \tanh((T - c_{tl})c_b) + w \left(T - \frac{c_{tl} + c_{tr}}{2} \right) + \frac{h_0}{2} + c_s \tag{4}$$

$$f_r^{major}(T) = \frac{h_0}{2} \tanh((T - c_{tr})c_b) + w \left(T - \frac{c_{tl} + c_{tr}}{2} \right) + \frac{h_0}{2} + c_s \tag{5}$$

where T denotes the temperature, and $h_0, c_{tl}, c_{tr}, c_b, w, c_s$ parameterize the shape of the curves. Values of such parameters are chosen to fit the experimental data of SMA [40,41]. The radius factor γ varies according to the f_l^{major} curve when the temperature decreases and f_r^{major} when the temperature increases. Furthermore, switching the temperature direction during the transformation procedure causes a reverse transformation starting

from the current temperature and strain, which is not on the major loop of the reverse transformation. Such transforms are modeled using minor hysteresis loops, which are modeled by hyperbolic tangent curves with similar shapes as the major loops. The function of a rising minor loop is given as

$$f_r^{minor}(T, h) = \frac{h}{2} \tanh((T - c_{tr})c_b) + w \left(T - \frac{c_{tl} + c_{tr}}{2} \right) + h_0 - \frac{h}{2} + c_s \tag{6}$$

where h is selected to ensure the intersection of the consecutive curves at the current point and is given by

$$h = g_r(h_{prev}, T) = \frac{h_{prev} (\tanh((T - c_{tl})c_b) + 1) - 2h_0}{\tanh((T - c_{tr})c_b) - 1} \tag{7}$$

and h_{prev} is the height parameter of the previous curve. Functions for the lowering curves are analogous as

$$f_l^{minor}(T, h) = \frac{h}{2} \tanh((T - c_{tl})c_b) + w \left(T - \frac{c_{tl} + c_{tr}}{2} \right) + \frac{h}{2} + c_s \tag{8}$$

and

$$h = g_l(h_{prev}, T) = \frac{h_{prev} (\tanh((T - c_{tr})c_b) - 1) + 2h_0}{\tanh((T - c_{tl})c_b) + 1} \tag{9}$$

An illustration of the transformation procedure is given in Figure 2.

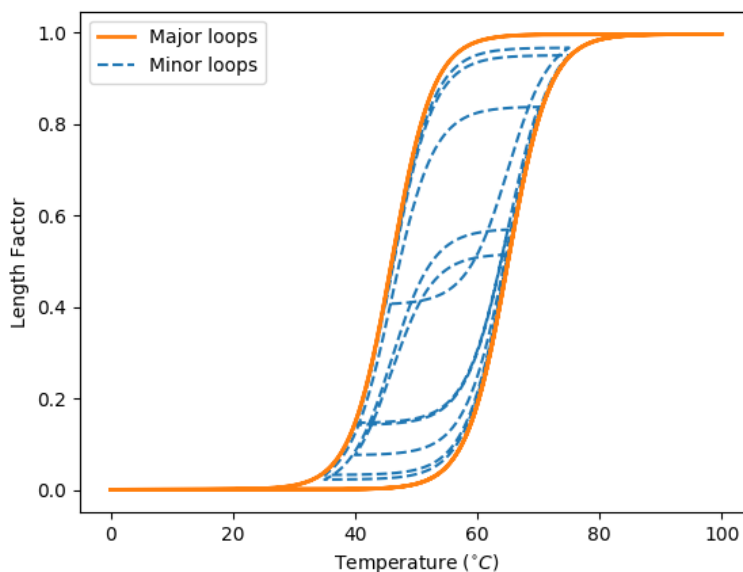


Figure 2. An illustration on the hysteresis loops of SMA wires.

After constructing the temperature-strain model, resist heating is used to actuate the SMA wires [50]. Given the applied voltage v_i , the temperature T follows the heat transfer model [39]

$$m_w c_w \dot{T}_i = \frac{v_i^2}{R_w} - h_w A_w (T_i - T_f) \quad i = 1, \dots, N \tag{10}$$

where m_w is the mass per unit length of the SMA wire, c_w is the specific heat, R_w is the electrical resistance per unit length, h_w is the heat exchange coefficient, A_w is the wire circumferential area, and T_f is the airflow temperature. Combining the temperature-strain relationship and (10), it is shown that the dynamic system between the radius and the input voltage are highly nonlinear due to the hysteresis characteristics. An illustration on the dynamics of SMA wires driven by a sinusoidal voltage input is given in Figure 3.

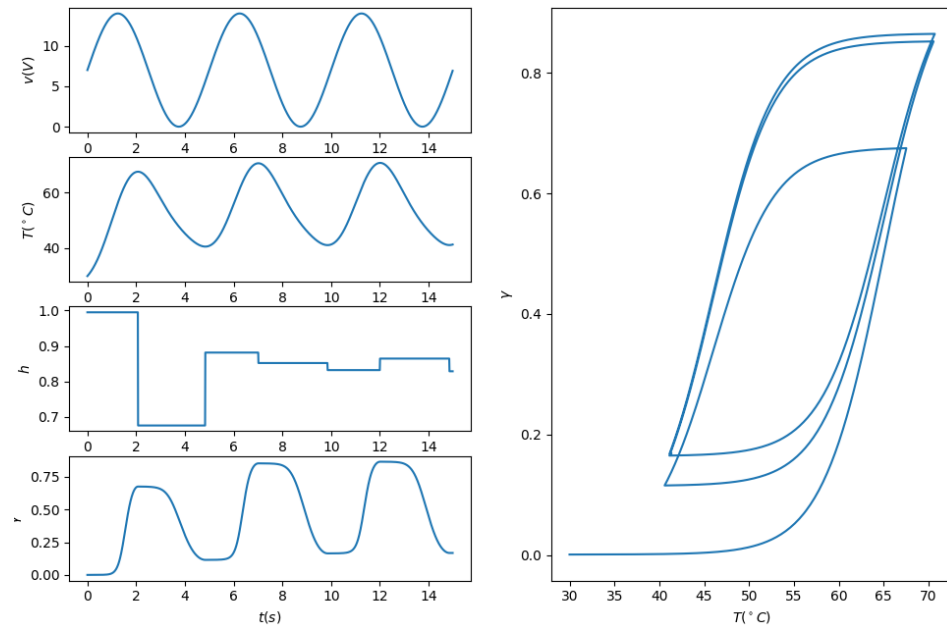


Figure 3. An illustration on the dynamics of SMA wires with sinusoidal voltage input.

In this work, we tackle the morphing problem for the airfoil constructed by the SMA wires whose dynamics are given in (4)–(10). Note that the temperature-strain and voltage-temperature relationships modeled above are not directly accessible to our controller, but serve as the environment from which paths of the states can be sampled. We resort to deep reinforcement learning methods to design the morphing policy.

2.3. Reinforcement Learning based Morphing Control

Reinforcement learning (RL) methods are capable of learning a control policy from interactions between the given agent and environment, with no requirement on the knowledge of system models [51]. The learning procedures are based on Markov decision processes (MDPs), which are given by 4-tuples $\{S, A, R, P\}$, where S and A are the state space and action space containing all the states and actions, respectively, R is the reward function giving $r_k = R(s_k, a_k, s_{k+1})$ as rewards, and P is the transition function giving $P(s_{k+1}|s_k, a_k)$ as state transition probabilities. In this section, the airfoil morphing problem is solved in the RL framework, where we aim to find the optimal policy maximizing the expected total rewards.

Before choosing the states and actions for the morphing problem, the morphing system is investigated further to construct an MDP from it. Firstly, the voltage-temperature dynamics (10) of SMA wires is discretized via Euler methods as

$$T_{i,k} = T_{i,k-1} + \Delta t \cdot \sigma(T_{i,k-1}, v_{i,k-1}) \tag{11}$$

where $\Delta t > 0$ is the discretizing time step, and

$$\sigma(T, v) = \frac{v^2}{m_w c_w R_w} - \frac{h_w A_w}{m_w c_w} (T - T_f) \tag{12}$$

With regard to the temperature-strain dynamics, since the minor loops converge with the major loops outside the SMA’s transformation region, we assume that the major loops determine the initial states of the wires, and minor loops dominate the dynamics during the

morphing procedure. We denote the function (8) as $f_l(T, h)$ and (6) as $f_r(T, h)$ for simplicity, and introduce the signum function

$$\text{sgn}(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{13}$$

Then, $\text{sgn}(T_k - T_{k-1})$ is used to discriminate the status of raising or lowering the temperature at time k . According to (11) and the fact that $\Delta t > 0$, we describe the temperature direction at time k by

$$\chi_{i,k} \triangleq \text{sgn}(T_k - T_{k-1}) = \text{sgn}(\sigma(T_{i,k-1}, v_{i,k-1})) \tag{14}$$

Note that the strain is dependent on the height parameter of the current loop. Recall from (7) and (9) that the value of the height parameter changes when the direction of temperature switches. Then, the time-varying parameter is determined by

$$h_{i,k} = (1 - \chi_{i,k-1:k}) \cdot (\chi_{i,k} g_r(T_{i,k}, h_{i,k-1}) + (1 - \chi_{i,k}) g_l(T_{i,k}, h_{i,k-1})) + \chi_{i,k-1:k} h_{i,k-1} \tag{15}$$

where

$$\chi_{i,k-1:k} \triangleq \text{sgn}(\sigma(T_{i,k-2}, v_{i,k-2})\sigma(T_{i,k-1}, v_{i,k-1})) \tag{16}$$

detects the reversal of temperature direction. Subsequently, the radius factor given the temperature and height parameter is calculated by choosing the rising or lowering loop according to the current temperature direction as

$$\gamma_{i,k} = \chi_{i,k} f_r(T_{i,k}, h_{i,k}) + (1 - \chi_{i,k}) f_l(T_{i,k}, h_{i,k}) \tag{17}$$

Summarizing the relationships (11), (15) and (17), it seems appropriate to select T and h as states and v as actions to construct a RL environment for airfoil morphing. The radius factor can be treated as an observation since it is dependent on only the current voltage, temperature and height parameter. Since the length and temperature of SMA wires can be measured directly via sensors equipped on the airfoils, these values are assumed to be available. However, the height parameter is not a realistic physical characteristic but just a coefficient fitting the hyperbolic tangent curves to the actual SMA properties, which makes the measurement on h not available. Actually, we aim to adjust the position of control points to achieve a reference airfoil shape, such that we want to learn a policy on modifying the lengths (i.e., the radius factors). We observe from the loop functions (6) and (8) that, given the temperature T , the function of γ with respect to h is bijective. Therefore, denoting the relationship (17) as $\gamma_{i,k} = f(h_{i,k}, T_{i,k}, T_{i,k-1}, v_{i,k-1})$, we can obtain an inverse function as

$$h_{i,k} = f^{-1}(\gamma_{i,k}, T_{i,k}, T_{i,k-1}, v_{i,k-1}) \tag{18}$$

Combining (15) and (18), the dynamics of radius factors are described as

$$\begin{aligned} \gamma_{i,k} = f \left((1 - \chi_{i,k-1:k}) \cdot \left[\chi_{i,k} g_r \left(T_{i,k}, f^{-1}(\gamma_{i,k-1}, T_{i,k-1}, T_{i,k-2}, v_{i,k-2}) \right) \right. \right. \\ \left. \left. + (1 - \chi_{i,k}) g_l \left(T_{i,k}, f^{-1}(\gamma_{i,k-1}, T_{i,k-1}, T_{i,k-2}, v_{i,k-2}) \right) \right] \right. \\ \left. + \chi_{i,k-1:k} f^{-1}(\gamma_{i,k-1}, T_{i,k-1}, T_{i,k-2}, v_{i,k-2}), T_{i,k}, T_{i,k-1}, v_{i,k-1} \right) \end{aligned} \tag{19}$$

Note that $\chi_{i,k-1:k}$ is dependent on the temperature and voltage at time $k - 2$. This makes the transition function of γ a second-order difference equation.

Proposition 1. Given a second-order MDP with states $s' \in \mathbb{R}^{n_{s'}}$ and actions $a' \in \mathbb{R}^{n_{a'}}$ satisfying

$$p(s'_k | s'_{k-1}, a'_{k-1}, s'_{k-2}, a'_{k-2}, \dots, s_0, a_0) = p(s'_k | s'_{k-1}, a'_{k-1}, s'_{k-2}, a'_{k-2}) \quad k \geq 2 \quad (20)$$

select $s_k = [s_k^\top, s_{k-1}^\top]^\top$ and $a_k = [a_k^\top, a_{k-1}^\top]^\top$. Then, an MDP can be constructed by states s and actions a (with an additionally defined state transition function and reward function). Moreover, if the second-order MDP with s' and a' has a deterministic state transition function

$$s'_k = f_s(s'_{k-1}, a'_{k-1}, s'_{k-2}, a'_{k-2}) \quad (21)$$

then the MDP constructed by s and a satisfies the transition function

$$s_k = \left[f_s^\top \left([\mathbf{I}_{n_{s'}}, 0] s_{k-1}, [\mathbf{I}_{n_{a'}}, 0] a_{k-1}, [0, \mathbf{I}_{n_{s'}}] s_{k-1}, [0, \mathbf{I}_{n_{a'}}] a_{k-1} \right), ([0, \mathbf{I}_{n_{s'}}] s_{k-1})^\top \right]^\top \quad (22)$$

This proposition can be derived directly via the properties of MDPs. According to Equations (11), (19) and Proposition 1, it is reasonable to choose the radius factors $\{\gamma_{i,k-1}, \gamma_{i,k}\}$ and temperatures $\{T_{i,k-1}, T_{i,k}\}$ as states with input voltages $\{v_{i,k-1}, v_{i,k}\}$ as actions. We restrict the states by $\gamma \in [0, 1]$ and the actions by $v \in [0, V_{\max}]$.

The reference radius factors of the optimized airfoil shape under certain condition c_k are denoted as $\gamma_{i,k}^{\text{ref}}(c_k)$. When determining the reward, we expect the airfoil to morph to reference shapes accurately and rapidly. Therefore, a sparse reward function comprised of two components at time k is designed by

$$r'_k = R(s_k, a_k, c_k) = \sum_{i=1}^N (r_{k,i}^{\text{pos}} + r_{k,i}^{\text{vol}}) \quad (23)$$

where the position reward conveying the requirement of morphing accuracy is given by

$$r_{k,i}^{\text{pos}} = \begin{cases} r_p, & |\gamma_{i,k} - \gamma_{i,k}^{\text{ref}}(c_k)| \leq e_{\text{thr}} \\ 0, & |\gamma_{i,k} - \gamma_{i,k}^{\text{ref}}(c_k)| > e_{\text{thr}} \end{cases} \quad (24)$$

and the voltage reward aiming to increase the morphing speed is given by

$$r_{k,i}^{\text{vol}} = \begin{cases} r_v, & |\gamma_{i,k} - \gamma_{i,k}^{\text{ref}}(c_k)| > e_{\text{thr}} \quad \text{and} \quad \text{sgn}(v_{i,k} - v_{i,k-1}) = \text{sgn}(\gamma_{i,k}^{\text{ref}}(c_k) - \gamma_{i,k}) \\ 0, & \text{otherwise} \end{cases} \quad (25)$$

where $r_p > r_v > 0$ and $e_{\text{thr}} > 0$ are tunable hyperparameters. When the position error is small, the voltage reward is eliminated to mitigate the oscillation. The choice of adjacent voltages as actions permits the calculation of the voltage reward in the MDP framework. Furthermore, the total return to be maximized is given by

$$R(\tau) = \sum_{k=0}^K r'_k \quad (26)$$

where $\tau = (s_0, a_0, c_0, s_1, a_1, c_1 \dots)$ denotes the sequence of states, actions and conditions.

After establishing the MDP, we proceed to tackle the morphing task based on deep reinforcement learning techniques. Our learning method is designed based on the soft actor-critic (SAC) algorithm [52], which is an off-policy reinforcement learning method compatible with continuous state and action spaces. In the actor-critic framework, the agent learns to interact with the environment and obtain maximum rewards via training two types of neural networks iteratively. The first one is named a critic network and accepts current states and actions as input to approximate the action-value function, which serves as an evaluation of the current policy. The second one is denoted as an actor network and generates actions according to the system states and optional external inputs. After the

training is converged, the policy is determined by the actor network and conducted for online executions, which in this work is the morphing task. In SAC, a stochastic policy is learned with additional entropy regularization in the rewards, which improves the ability of exploration and achieves faster convergence for a variety of control problems. According to the MDP of the morphing procedure, we choose the state s_k to be $s_k = [s'_{1,k-1}, s'_{1,k}, \dots, s'_{N,k-1}, s'_{N,k}]$ and the action as $a_k = [a'_{1,k-1}, a'_{1,k}, \dots, a'_{N,k-1}, a'_{N,k}]$, where $s'_{i,k} = [\gamma_{i,k}, T_{i,k}]$ and $a'_{i,k} = v_{i,k}$. Note that since the reference airfoil shapes guide the morphing, the flight condition should be incorporated for the generation and evaluation of the actions. We denote the distribution of the stochastic policy as $\pi(a|s, c)$.

With regard to the construction of the action-value function, instead of directly applying (23), we augment the reward with policy entropy as

$$r_k = R(s_k, c_k, a_k) = \sum_{i=1}^N \left(r'_{k,i} + \beta H_{\pi}(s_{i,k}, c_k) \right) \tag{27}$$

where

$$H_{\pi}(s_{i,k}, c_k) = \mathbb{E}_{a_{i,k} \sim \pi(\cdot|s_{i,k}, c_k)} [-\log \pi(a_{i,k}|s_{i,k}, c_k)] \tag{28}$$

is the entropy representing the randomness of the policy, and $\beta > 0$ is the trade-off coefficient. According to (26), we define a finite-horizon undiscounted return to be maximized. Nevertheless, a discount factor is applied when evaluating the value functions to focus on recent rewards, since the future reference shapes are not accessible at the current time step. Then, the action-value function with the regularized reward function is introduced as

$$Q^{\pi}(s, a, c, k) = \mathbb{E}_{\tau \sim \pi} \left[- \sum_{t=k}^K \sum_{i=1}^N \tilde{\zeta}^t r'_{k,i} + \sum_{t=k+1}^K \sum_{i=1}^N \tilde{\zeta}^t \beta H_{\pi}(s_{i,t}, c_t) \mid s_k = s, a_k = a, c_k = c \right] \tag{29}$$

Afterwards, the Bellman equation for the action-value function is given as

$$Q^{\pi}(s_k, a_k, c_k, k) = r_k + \mathbb{E}_{s_{k+1} \sim P, a_{k+1} \sim \pi} [\tilde{\zeta} (Q^{\pi}(s_{k+1}, a_{k+1}, c_{k+1}, k+1) - \beta \log \pi(a_{k+1}|s_{k+1}, c_{k+1}))] \tag{30}$$

and following the schedule of SAC [52], we approximate the expectation by

$$Q^{\pi}(s_k, a_k, c_k, k) \approx r_k + \tilde{\zeta} (Q^{\pi}(s_{k+1}, \tilde{a}_{k+1}, c_{k+1}, k+1) - \beta \log \pi(\tilde{a}_{k+1}|s_{k+1}, c_{k+1})) \tag{31}$$

where $\{s_k, a_k, c_k, s_{k+1}, c_{k+1}\}$ are sampled from replay buffers, and the next action \tilde{a}_{k+1} is sampled from the current policy $\pi(\cdot|s_{k+1}, c_{k+1})$.

For the learning of the action-value functions, the double-Q trick is applied to avoid overestimation [53]. Two critic networks for Q functions are implemented as $Q_{\phi_1}(s, a, c, k)$ and $Q_{\phi_2}(s, a, c, k)$, where ϕ_1 and ϕ_2 are parameters. Additionally, for the stabilization of the training procedure, the target networks $Q_{\phi'_1}$ and $Q_{\phi'_2}$, which are copies of Q_{ϕ_1} and Q_{ϕ_2} , are used and updated by polyak averaging after each time we update the main critic networks as

$$\phi' \leftarrow \rho \phi' + (1 - \rho) \phi \tag{32}$$

where $\rho \in (0, 1)$ is the update hyperparameter. Summarizing all these settings, the loss for critic networks is given by the mean squared Bellman error function as

$$L(\phi) = \sum_{b \in B} (Q_{\phi}(s_k, a_k, c_k, k) - y(s_{k+1}, a_{k+1}, c_{k+1}, k+1)) \tag{33}$$

where B is the sampled batch with elements $b = \{s_k, a_k, c_k, r_k, k, s_{k+1}, c_{k+1}, k+1\}$ of the replay buffer, and where

$$y(s_{k+1}, a_{k+1}, c_{k+1}, k+1) = \tilde{\zeta} \left(Q_{\phi'}(s_{k+1}, \tilde{a}_{k+1}, c_{k+1}, k+1) - \beta \log \pi(\tilde{a}_{k+1}|s_{k+1}, c_{k+1}) \right) \tag{34}$$

is calculated using target networks. Stochastic gradient descent is applied to update ϕ_1 and ϕ_2 with respect to the loss functions $L(\phi_1)$ and $L(\phi_2)$.

Subsequently, we aim to find the policy that maximizes the expected action-value function with respect to the actions. Denote the parameters of actor network as θ . Since the action at time step k is composed of the input voltages at $k - 1$ and k , the actor network should be designed as $\pi_\theta(s_k, c_k, a_{k-1})$, where an identity layer is applied to propagate the previous voltages. However, the distribution of new input voltages is still only dependent on s_k and c_k . Therefore, we use $\pi_\theta(a_k|s_k, c_k)$ to denote the density of action a_k . Then, the value function to be optimized is given as

$$V^{\pi_\theta}(s_k, c_k, k) = \mathbb{E}_{a_k \sim \pi_\theta} [Q^{\pi_\theta}(s_k, a_k, c_k, k) - \beta \log(\pi_\theta(a_k|s_k, c_k))] \tag{35}$$

The reparameterization trick is adopted here for the sake of the efficient computation of gradients [54]. We introduce a standard normal distributed variable $\zeta \sim p(\zeta) = \mathcal{N}(0, \mathbf{I}_{n_a})$ and calculate the input voltage according to a deterministic squashing function as

$$v_k = \frac{V_{\max}}{2} (1 + \tanh(\mu_\theta(s_k, c_k) + \sigma_\theta(s_k, c_k) \circ \zeta_k)) \tag{36}$$

where $\mu_\theta(s, c)$ and $\sigma_\theta(s, c)$ are parameterized neural networks and \circ denotes element-wise multiplication. Then, the expectation over $a_k \sim \pi$ can be converted to the expectation over the normal variable whose distribution is irrelevant to the states and net parameters. Additionally, the squashed Gaussian policy constrains the input voltages in $[0, V_{\max}]$. According to (36), the action given s_k, c_k and ζ_k is written as $a_k = a_\theta(s_k, c_k, \zeta_k)$. Note that this is an invertible map between a_k and ζ_k . Therefore, we can compute the log-probabilities in closed form according to the change of the variable formula [55] as

$$\begin{aligned} \log(\pi_\theta(a_k|s_k, c_k)) &= \log(p(\zeta_k)) - \log |\det J_{a_\theta}(\zeta_k)| \\ &= \log(p(a_\theta^{-1}(s_k, c_k, a_k))) + \log |\det J_{a_\theta^{-1}}(a_k)| \end{aligned} \tag{37}$$

where a_θ^{-1} is the inverse function of a_θ given s_k and c_k , and J denotes the Jacobian matrix.

With Q^{π_θ} approximated by the minimum of the two critic networks, the loss function for the actor network is obtained as

$$L(\theta) = \sum_{b \in B} \left(\min_{i=1,2} Q_{\phi_i}(s_k, a_\theta(s_k, c_k, \zeta_k), c_k, k) - \beta \log(\pi_\theta(a_\theta(s_k, c_k, \zeta_k)|s_k, c_k)) \right) \tag{38}$$

Then, we conduct the training by updating the actor and critic networks iteratively. The agent interacts with various randomly generated time-varying reference shape sequences to acquire training data, which facilitates the exploration and enables the policy to handle different morphing scenarios. When the training converges, we can use the actor network to calculate the required voltages and morph the airfoil to the reference shapes. Finally, the overall flowchart of the proposed morphing mechanism is given in Figure 4.

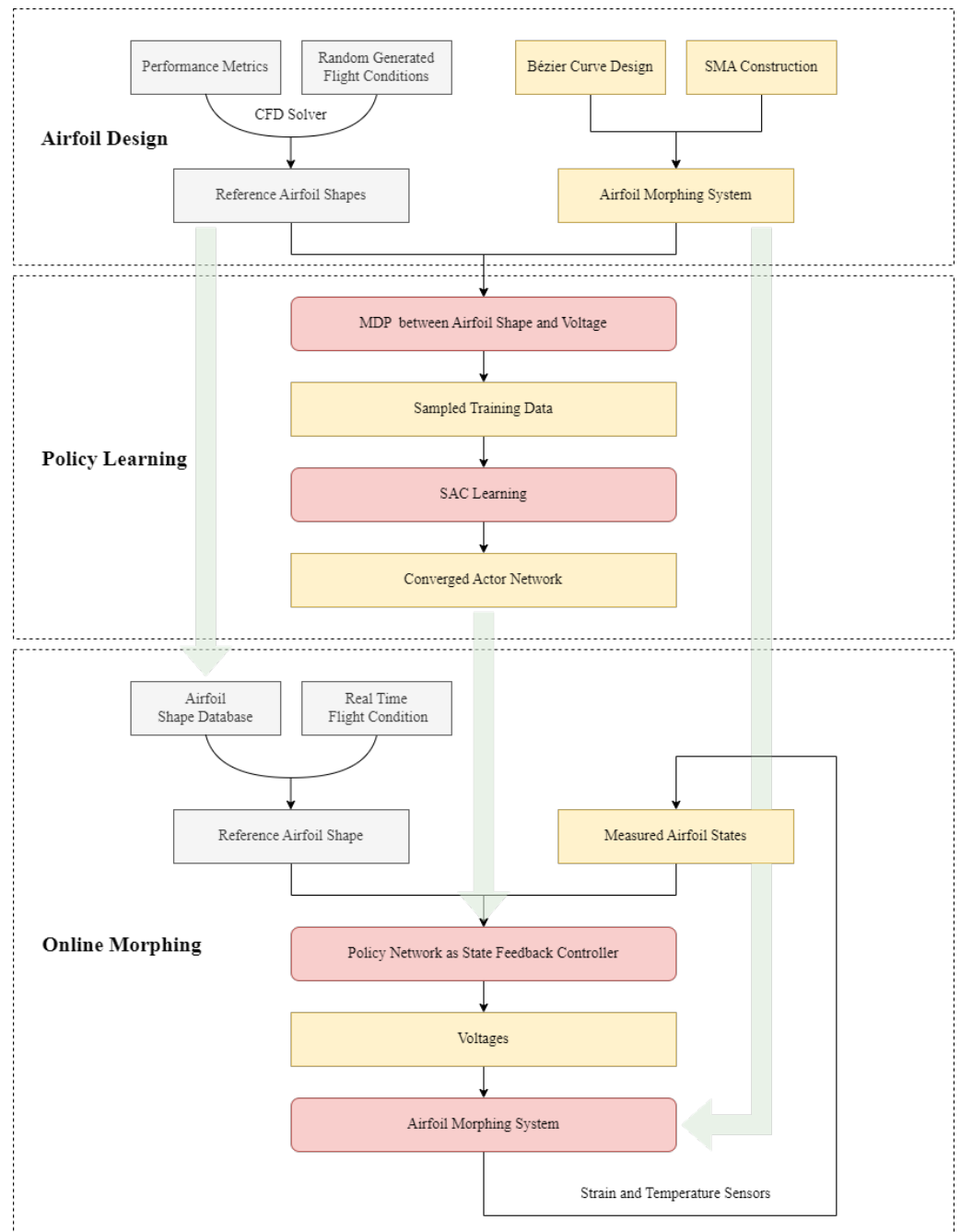


Figure 4. Overall flowchart of the proposed DRL-based airfoil morphing framework.

3. Results

In this section, a simulation is conducted to validate the proposed morphing method. The simulation is arranged in two stages. Firstly, we implement our method with random generated reference shapes and perform ablation studies to examine the superiorities brought by different parts of our algorithm. Subsequently, we apply the proposed method to track optimized airfoil shapes in different flight conditions and show the morphing procedures. The values of parameters in our simulation are given in Table 1 [40,41].

Table 1. Values of system parameters used in simulations.

Parameter	Value	Parameter	Value
m_w	1.14×10^{-4}	A_w	4.72×10^{-4}
c_w	837.4	R_w	50.8
T_f	20	h_w	120
H	0.995	c_b	0.147
c_w	1.25×10^{-5}	c_s	0.001
c_{tl}	46	c_{tr}	65

3.1. Tracking Random Shapes

In this stage, the superiorities of our method are illustrated in a variety of perspectives, including the state/action selection, reward configuration and entropy regularization. Without loss of generality, piece-wise constant trajectories of the radius factor for one control point are generated to represent the reference shapes. Then, our method is compared with three different settings of RL algorithms. The proposed method is denoted as RLM-SAC.

- Second-order state/action versus first-order state/action

In Section 2, second-order MDP is adopted to model the hysteresis characteristics of the morphing system. Therefore, we chose the states and actions as combinations of that in current step and previous step. We compared the performance with RL algorithms where the policy is generated according to only current states, and the value function was also evaluated with only current states and actions as inputs that are applied in existing investigations on controlling SMA wires. We refer to this as RLM-FO.

- Sparse reward versus squared error reward

We designed a sparse reward taking value in $\{0, 1\}$, which is different from traditional RL-based morphing research. We compared that with the square error rewards, which is given by

$$r'_k = R(s_k, c_k) = - \sum_{i=1}^N |\gamma_{i,k} - \gamma_{i,k}^{\text{ref}}(c_k)|^2 \quad (39)$$

which is named RLM-SER.

- SAC versus DQN

The entropy regularization improves the capability of exploration in our algorithm. A modified deep Q learning method was implemented as a comparison, where only the entropy loss was removed, and both the double-Q setting and reparameterization trick remained. We denote this as RLM-DQN.

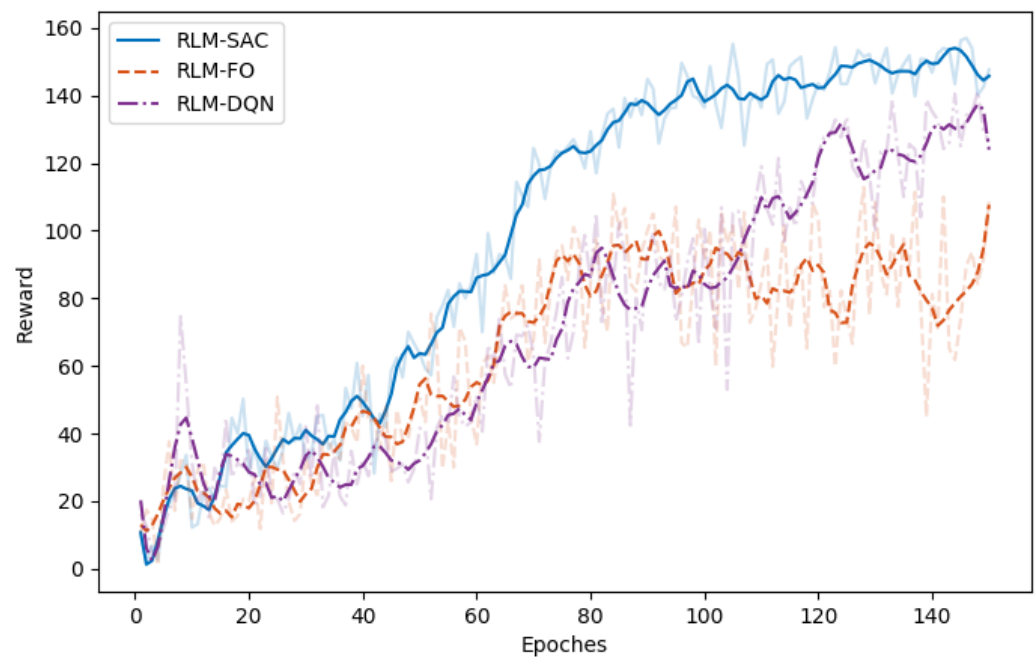
All RL realizations were trained through 150 epochs, in each of which 5 episodes with 40 s of time and 200 time steps were executed. The critic network was constructed using multilayer perceptrons (MLP) of 3 hidden layers and 128 units per layer. The actor network adopted similar structures, where additional fully connected layers were attached to produce the mean and standard variations of the policy. The training was started with actions uniformly sampled from the valid action space bounded by $[0, V_{\max}]$ for 5000 steps to explore the state space sufficiently. Then, the networks were updated every step with a batch size of 200. A fixed learning rate was set as 0.002, and other hyperparameters used in RL training are shown in Table 2. The actions were generated from the stochastic policy in training phase but produced in a deterministic way according to the mean value of the actor network in the testing phase.

Table 2. Values of hyperparameters used in RL.

Parameter	Value	Parameter	Value
ζ	0.98	α	0.2
ρ	0.995	V_{\max}	15
r_p	1	r_v	0.02
e_{thr}	0.1		

The results are shown in Figures 5–8. In Figures 5 and 6, we illustrate the rewards acquired in the training and test trajectories, respectively, during the training procedure. A Savitzky–Golay filter [56] was adopted to smooth the data such that the values and trends of the rewards are illustrated more clearly. Note that the reward of RLM-SER was not included because of a different reward setting. After the training was finished, the algorithms were executed on 100 random generated test reference trajectories. We present the root-mean-squared error (RMSE) of the radius factor through the flight time in Figure 7. Some of the trajectories and the corresponding results produced by the four RL realizations are depicted in Figure 8 for an intuitive comparison.

From the results, we can see the benefits of each important component in our algorithm. Firstly, RL using square error reward totally fails to produce effective actions in our environment, which is shown from both the RMSE and example trajectories. RLM-FO acquires inferior performance compared with RLM-SAC, especially in the temperature-switching procedure, which can be validated by the middle sections of the example trajectories. This is a result of the fact that the hysteresis cannot be characterized well by first-order states and actions. Lastly, RLM-DQN obtains better performance than RLM-FO and RLM-SRE, and it achieves a similar reward to RLM-SAC at the end of training. However, from the illustrations of rewards in Figures 5 and 6, it is shown that RLM-DQN converges much slower than RLM-SAC. This is due to the improvement of exploration capability provided by entropy regularization.

**Figure 5.** Reward of different RL realizations over the training trajectories. The data are smoothed via Savitzky–Golay filter with window size 15 and order 5.

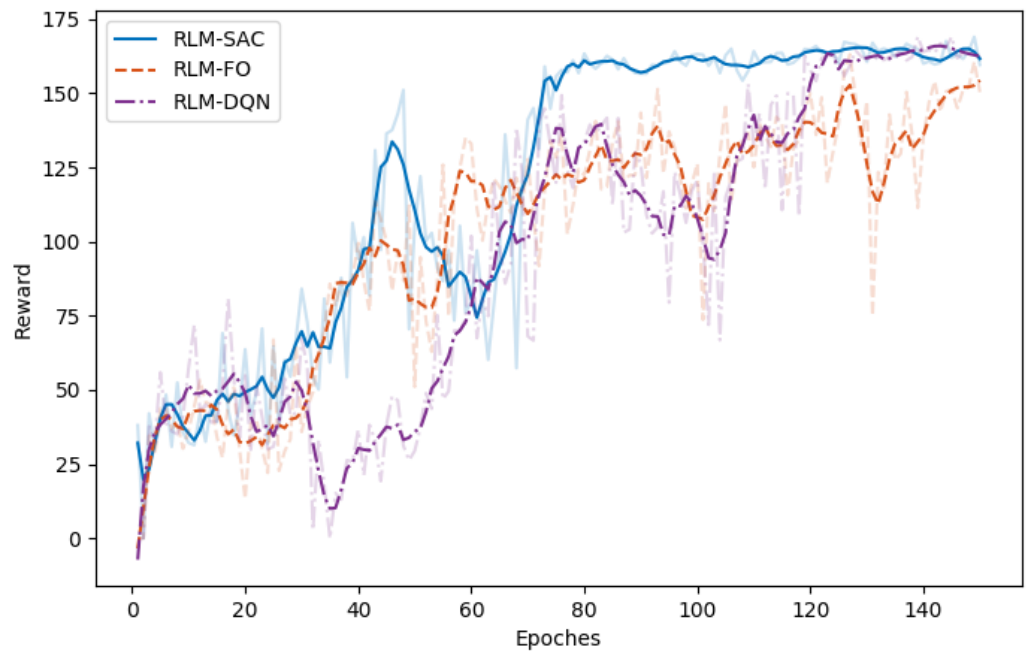


Figure 6. Reward of different RL realizations over the test trajectories. The data are smoothed via Savitzky–Golay filter with window size 15 and order 5.

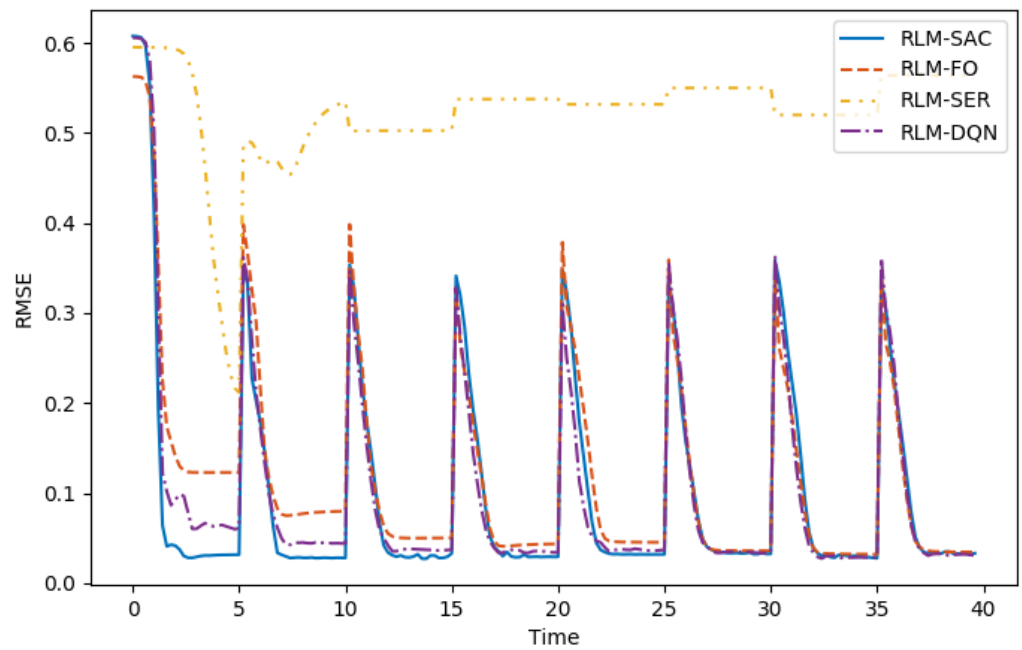


Figure 7. RMSE of radius factors generated by different RL realizations. The flight conditions in test trajectories change every 5 s, and at this time, the airfoil is expected to morph to a new reference shape.

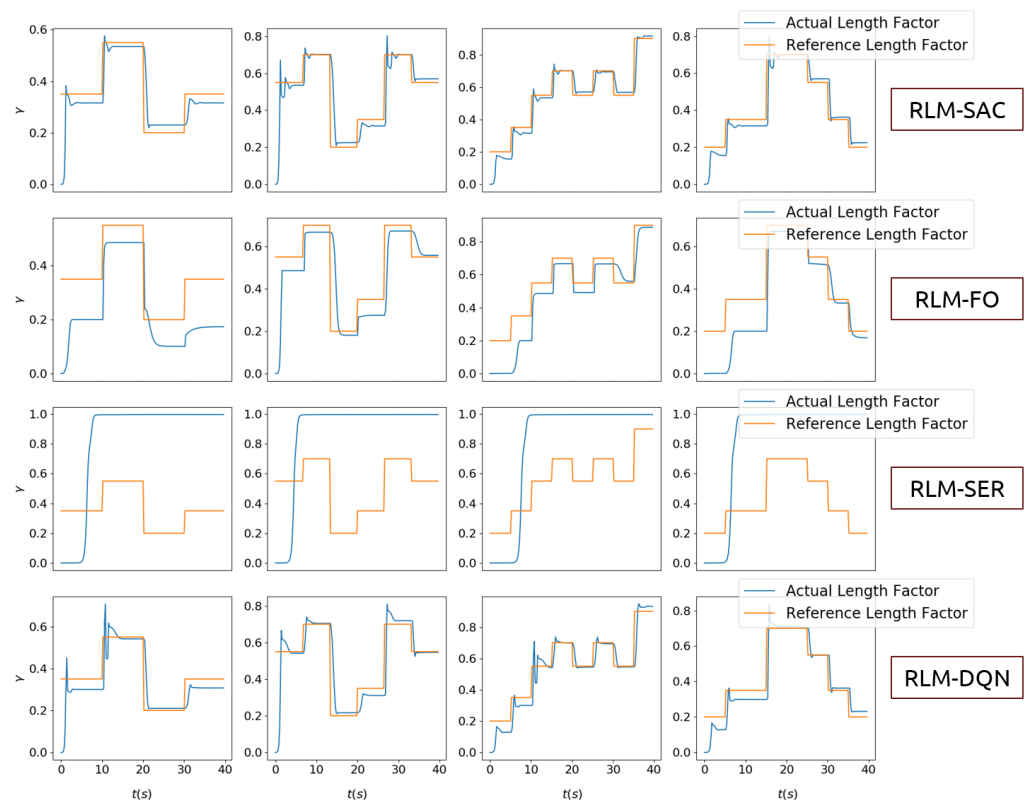


Figure 8. Illustration of some reference trajectories and the performance of each RL method.

3.2. Morphing Procedure Simulation

In this stage, the trained actor network is applied to morph an airfoil controlled by four points in a given flight procedure with varying flight conditions. Since the focus of this work is morphing control, in each condition, the optimal shape is assumed to be solved in anticipation by shape optimization techniques and determined by the radii and angles of control points [14,18,20]. The average coefficient and scale coefficient are fixed as $\alpha = [0.12, 0.4, 0.4, 0.12]$ and $\eta = [0.5, 0.5, 0.5, 0.5]$. The trajectory of flight conditions and the corresponding parameters of optimal shapes are shown in Figure 9.

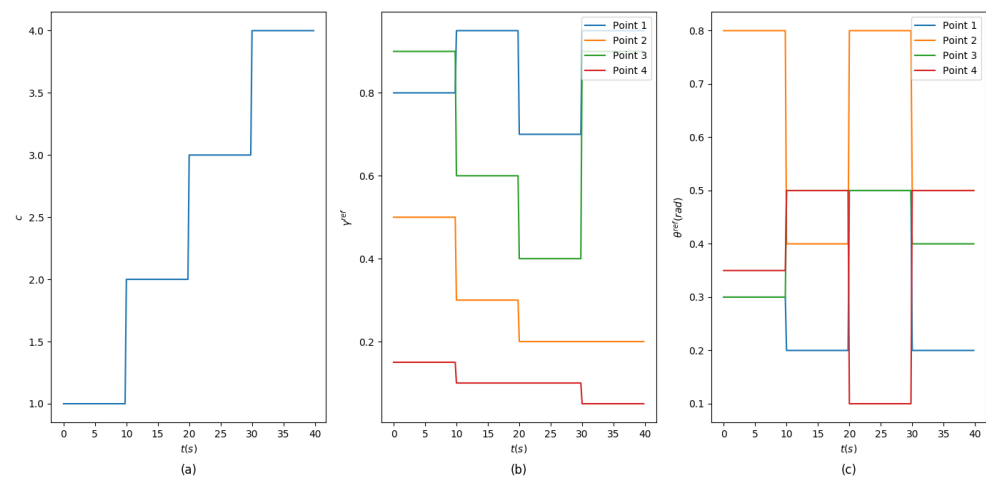


Figure 9. Trajectory of flight conditions. (a) Condition indexes. (b) Trajectories of optimal radius factor for each control point. (c) Trajectories of optimal angle for each control point.

Since the position control technique of motors is relatively sufficiently developed, we assumed that the angles of the points are controlled by DC motors with accurately known

linear dynamic models, which can drive the points to desired angles rapidly with subtle errors. Then we generate input voltages to heat the SMA wires and adjust the radius factors. The voltages, temperatures and radius factors of all points are summarized in Figure 10. It is shown that with constrained voltages, the wires can track the reference lengths well.

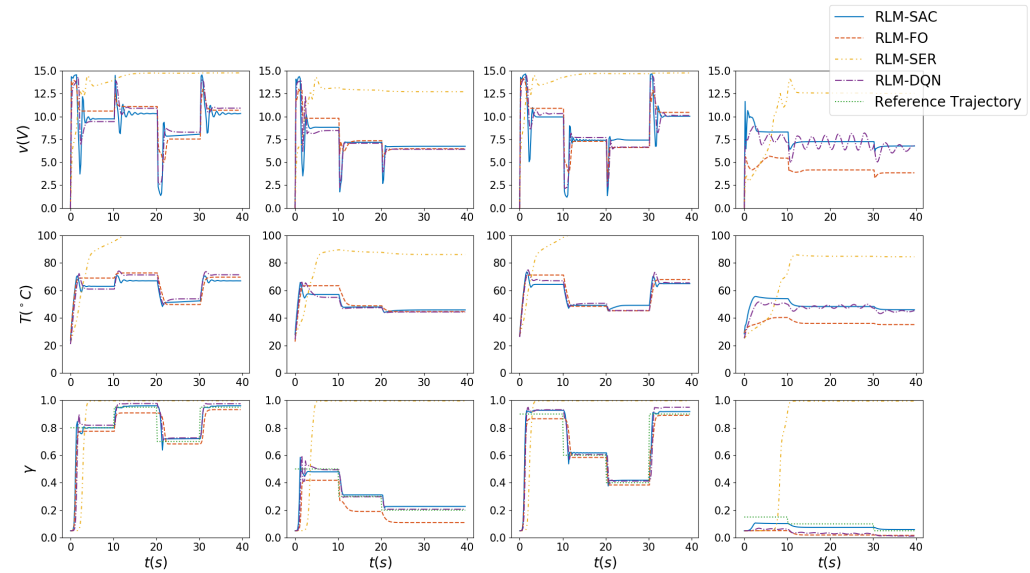


Figure 10. Voltages, temperatures and radius factors of control points. Each column represents the values of a point.

Furthermore, we illustrate the morphing procedures in Figure 11. We can see intuitively that with the proposed RLM-SAC method, the airfoil is capable of morphing into the optimized shape within about 3 s after encountering a new flight environment, which validates both the morphing accuracy and morphing speed.

Quantitative comparisons on the length factor differences and shape differences are given in Table 3. The average length factor differences are calculated according to the reference length factor and actual length factor as

$$e_{\text{length}} = \frac{1}{KN} \sum_{k=0}^K \sum_{i=1}^N \frac{|\gamma_{i,k} - \gamma_{i,k}^{\text{ref}}|}{\gamma_{i,k}^{\text{ref}}} \tag{40}$$

The difference between the actual and reference shapes are evaluated using distances between the control points. An average shape difference over all time is calculated as

$$e_{\text{shape-avg}} = \frac{1}{KN} \sum_{k=0}^K \sum_{i=1}^N \|p_{i,k} - p_{i,k}^{\text{ref}}\|_2 \tag{41}$$

where $\|p_i - p_i^{\text{ref}}\|_2$ is the L_2 distance between the two Cartesian coordinates $p_{i,k}$ and $p_{i,k}^{\text{ref}}$. Additionally, the steady shape error is evaluated by

$$e_{\text{shape-end}} = \frac{1}{4N} \sum_{t=1}^4 \sum_{i=1}^N \|p_{i,k_t} - p_{i,k_t}^{\text{ref}}\|_2 \tag{42}$$

where k_t denotes the end time step of each flight condition. It is shown that our method acquires the best performance on all metrics. The proposed RLM-SAC method provides an average 29.8% performance improvement over the second-best RLM-DQN method. The length differences and average shape differences, which are averaged over all time steps, demonstrate that our method can morph the airfoil into desired shapes more rapidly, while the steady shape difference validates the morphing accuracy. Lastly, in this work, we give

the reference airfoil shape directly and focus on the morphing performance of the proposed method. It will be interesting and meaningful to combine the shape optimization task with the morphing control problem in the future.

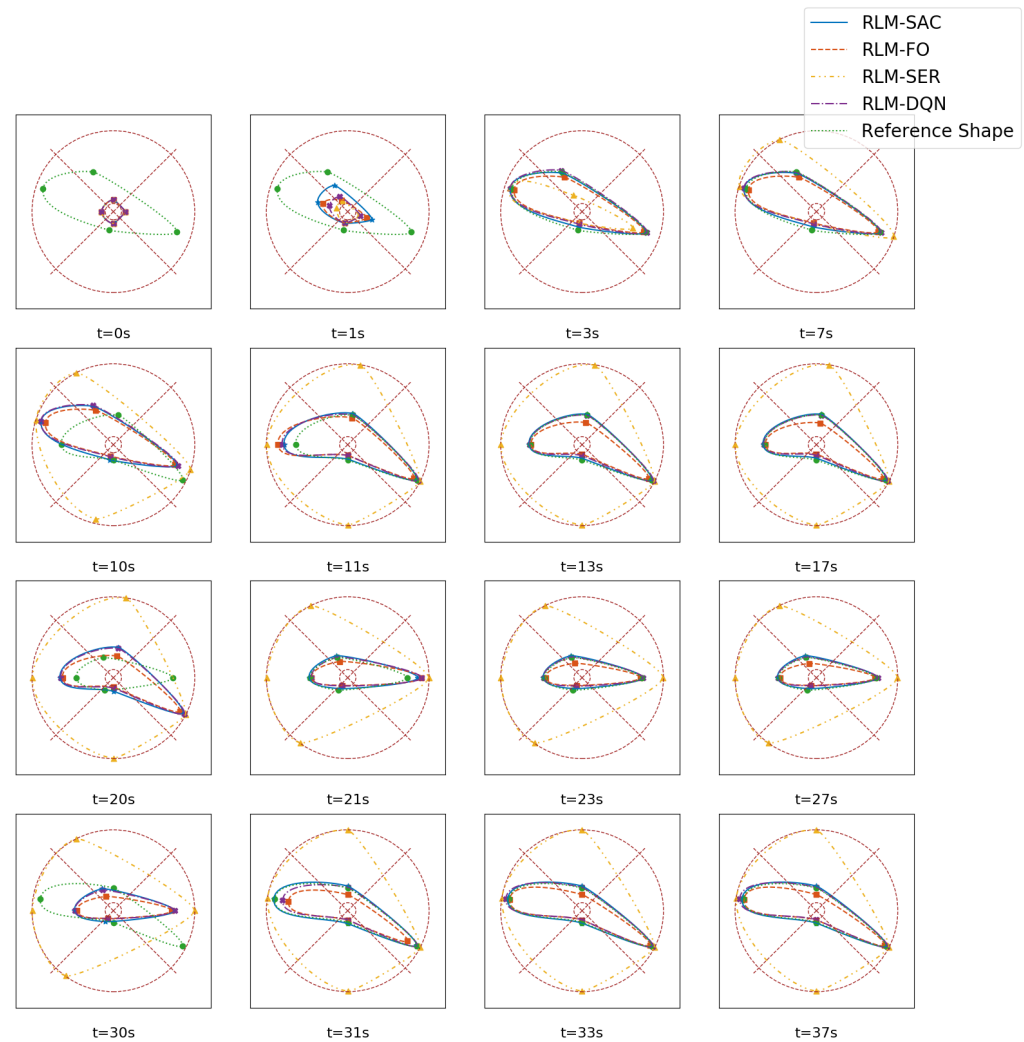


Figure 11. Illustration of the morphing procedure. Each row represents a morphing stage.

Table 3. Quantitative comparisons of different morphing methods.

	RLM-SAC	RLM-FO	RLM-SER	RLM-DQN
e_{length}	14.04%	31.44%	323.6%	23.24%
$e_{\text{shape-avg}}$	0.0533	0.0645	0.2202	0.0601
$e_{\text{shape-end}}$	0.0019	0.0055	0.0494	0.0031

4. Conclusions

In this work, a novel deep reinforcement learning-based morphing control method is proposed for an asymmetric morphing airfoil. The airfoil is designed via Bézier curves and is capable of morphing from a baseline shape to an asymmetric shape. The morphing mechanism is modeled via SMA wires, which adjust shape parameters, especially the radii of the control points. To actuate the SMA wires, resistive heating is performed, but the hysteresis characteristics between the SMA strain and temperature make the dynamic system nonlinear and non-Markovian, which brings difficulties to the design of the control algorithm and the RL framework. Therefore, hyperbolic tangent curves are adopted to model the strain-temperature relationship and derive a second-order MDP describing the system, which is then transformed into a valid MDP and provides guidance for the selection

of states and actions. Based on the constructed MDP, we modify the SAC algorithm and develop an RL scheme where input voltages are generated to morph the airfoil instantaneously according to reference-optimized shapes. Lastly, ablation studies on random generated reference trajectories are conducted to demonstrate the benefits brought by different components of our RL implementations, and we perform simulations of morphing procedures to validate that our method is able to morph the airfoil into the optimized shapes rapidly and accurately. Future works include incorporating the aerodynamic performance optimization directly into the morphing control and exploiting learning-based morphing policies for more complicated bio-inspired morphing aircraft.

Author Contributions: Conceptualization, K.L.; methodology, K.L. and Q.F.; software, Q.F.; validation, K.L., Q.F. and R.C.; formal analysis, K.L. and Q.F.; investigation, K.L., Q.F. and R.C.; writing—original draft preparation, Q.F.; writing—review and editing, K.L., R.C., J.P. and Q.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant number No. 61903084, 61973075, 62073075).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CFD	Computational Fluid Dynamics
MDP	Markov Decision Process
NACA	National Advisory Committee for Aeronautics
RL	Reinforcement Learning
RMSE	Root-Mean-Squared Error
SAC	Soft Actor-Critic
SMA	Shape Memory Alloy
UAV	Unmanned Aerial Vehicle

References

1. Floreano, D.; Wood, R.J. Science, technology and the future of small autonomous drones. *Nature* **2015**, *521*, 460–466. [[CrossRef](#)] [[PubMed](#)]
2. Harvey, C.; Gamble, L.L.; Bolander, C.R.; Hunsaker, D.F.; Joo, J.J.; Inman, D.J. A review of avian-inspired morphing for UAV flight control. *Prog. Aerosp. Sci.* **2022**, *132*, 100825. [[CrossRef](#)]
3. Gerdes, J.W.; Gupta, S.K.; Wilkerson, S.A. A review of bird-inspired flapping wing miniature air vehicle designs. *J. Mech. Robot.* **2012**, *4*, 021003. [[CrossRef](#)]
4. Ajanic, E.; Feroskhan, M.; Mintchev, S.; Noca, F.; Floreano, D. Bioinspired wing and tail morphing extends drone flight capabilities. *Sci. Robot.* **2020**, *5*, eabc2897. [[CrossRef](#)]
5. Harvey, C.; Baliga, V.; Goates, C.; Hunsaker, D.; Inman, D. Gull-inspired joint-driven wing morphing allows adaptive longitudinal flight control. *J. R. Soc. Interface* **2021**, *18*, 20210132. [[CrossRef](#)]
6. Derrouaoui, S.H.; Bouzid, Y.; Guiatni, M.; Dib, I. A comprehensive review on reconfigurable drones: Classification, characteristics, design and control technologies. *Unmanned Syst.* **2022**, *10*, 3–29. [[CrossRef](#)]
7. Barbarino, S.; Bilgen, O.; Ajaj, R.M.; Friswell, M.I.; Inman, D.J. A review of morphing aircraft. *J. Intell. Mater. Syst. Struct.* **2011**, *22*, 823–877. [[CrossRef](#)]
8. Carruthers, A.C.; Walker, S.M.; Thomas, A.L.; Taylor, G.K. Aerodynamics of aerofoil sections measured on a free-flying bird. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2010**, *224*, 855–864. [[CrossRef](#)]
9. Liu, T.; Kuykendoll, K.; Rhew, R.; Jones, S. Avian wing geometry and kinematics. *AIAA J.* **2006**, *44*, 954–963. [[CrossRef](#)]
10. Li, D.; Zhao, S.; Da Ronch, A.; Xiang, J.; Drofelnik, J.; Li, Y.; Zhang, L.; Wu, Y.; Kintscher, M.; Monner, H.P.; et al. A review of modelling and analysis of morphing wings. *Prog. Aerosp. Sci.* **2018**, *100*, 46–62. [[CrossRef](#)]

11. Vasista, S.; Riemenschneider, J.; Monner, H.P. Design and testing of a compliant mechanism-based demonstrator for a droop-nose morphing device. In Proceedings of the 23rd AIAA/AHS Adaptive Structures Conference, Kissimmee, FL, USA, 5–9 January 2015; p. 1049.
12. Monner, H.P. Realization of an optimized wing camber by using formvariable flap structures. *Aerosp. Sci. Technol.* **2001**, *5*, 445–455. [[CrossRef](#)]
13. Skinner, S.N.; Zare-Behtash, H. State-of-the-art in aerodynamic shape optimisation methods. *Appl. Soft Comput.* **2018**, *62*, 933–962. [[CrossRef](#)]
14. Wang, Y.; Shimada, K.; Farimani, A.B. Airfoil gan: Encoding and synthesizing airfoils for aerodynamic-aware shape optimization. *arXiv* **2021**, arXiv:2101.04757.
15. Achour, G.; Sung, W.J.; Pinon-Fischer, O.J.; Mavris, D.N. Development of a conditional generative adversarial network for airfoil shape optimization. In Proceedings of the AIAA Scitech 2020 Forum, Orlando, FL, USA, 6–10 January 2020; p. 2261.
16. He, X.; Li, J.; Mader, C.A.; Yildirim, A.; Martins, J.R. Robust aerodynamic shape optimization—From a circle to an airfoil. *Aerosp. Sci. Technol.* **2019**, *87*, 48–61. [[CrossRef](#)]
17. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv* **2017**, arXiv:1707.06347.
18. Viquerat, J.; Rabault, J.; Kuhnle, A.; Ghraieb, H.; Larcher, A.; Hachem, E. Direct shape optimization through deep reinforcement learning. *J. Comput. Phys.* **2021**, *428*, 110080. [[CrossRef](#)]
19. Syed, A.A.; Khamvilai, T.; Kim, Y.; Vamvoudakis, K.G. Experimental Design and Control of a Smart Morphing Wing System using a Q-learning Framework. In Proceedings of the 2021 IEEE Conference on Control Technology and Applications (CCTA), San Diego, CA, USA, 9–11 August 2021; pp. 354–359.
20. Bholra, S.; Pawar, S.; Balaprakash, P.; Maulik, R. Multi-fidelity reinforcement learning framework for shape optimization. *arXiv* **2022**, arXiv:2202.11170.
21. Liu, J.; Shan, J.; Hu, Y.; Rong, J. Optimal switching control for Morphing aircraft with Aerodynamic Uncertainty. In Proceedings of the 2020 IEEE 16th International Conference on Control & Automation (ICCA), Sapporo, Japan, 6–9 July 2020; pp. 1167–1172.
22. Valasek, J.; Tandale, M.D.; Rong, J. A reinforcement learning-adaptive control architecture for morphing. *J. Aerosp. Comput. Inf. Commun.* **2005**, *2*, 174–195. [[CrossRef](#)]
23. Valasek, J.; Doebbler, J.; Tandale, M.D.; Meade, A.J. Improved adaptive-reinforcement learning control for morphing unmanned air vehicles. *IEEE Trans. Syst. Man, Cybern. Part B* **2008**, *38*, 1014–1020. [[CrossRef](#)]
24. Lampton, A.; Niksch, A.; Valasek, J. Reinforcement learning of a morphing airfoil-policy and discrete learning analysis. *J. Aerosp. Comput. Inf. Commun.* **2010**, *7*, 241–260. [[CrossRef](#)]
25. Niksch, A.; Valasek, J.; Carlson, L.; Strganac, T. Morphing Aircraft Dynamical Model: Longitudinal Shape Changes. In Proceedings of the AIAA Atmospheric Flight Mechanics Conference and Exhibit, Honolulu, HI, USA, 18–21 August 2008; p. 6567.
26. Júnior, J.M.M.; Halila, G.L.; Kim, Y.; Khamvilai, T.; Vamvoudakis, K.G. Intelligent data-driven aerodynamic analysis and optimization of morphing configurations. *Aerosp. Sci. Technol.* **2022**, *121*, 107388. [[CrossRef](#)]
27. Paranjape, A.A.; Chung, S.J.; Selig, M.S. Flight mechanics of a tailless articulated wing aircraft. *Bioinspir. Biomimetics* **2011**, *6*, 026005. [[CrossRef](#)] [[PubMed](#)]
28. Chang, E.; Matloff, L.Y.; Stowers, A.K.; Lentink, D. Soft biohybrid morphing wings with feathers underactuated by wrist and finger motion. *Sci. Robot.* **2020**, *5*, eaay1246. [[CrossRef](#)] [[PubMed](#)]
29. Di Luca, M.; Mintchev, S.; Heitz, G.; Noca, F.; Floreano, D. Bioinspired morphing wings for extended flight envelope and roll control of small drones. *Interface Focus* **2017**, *7*, 20160092. [[CrossRef](#)]
30. Hetrick, J.; Osborn, R.; Kota, S.; Flick, P.; Paul, D. Flight testing of mission adaptive compliant wing. In Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Palm Springs, CA, USA, 4–7 May 2007; p. 1709.
31. Gilbert, W.W. Mission adaptive wing system for tactical aircraft. *J. Aircr.* **1981**, *18*, 597–602. [[CrossRef](#)]
32. Alulema, V.H.; Valencia, E.A.; Pillajo, D.; Jacome, M.; Lopez, J.; Ayala, B. Degree of deformation and power consumption of compliant and rigid-linked mechanisms for variable-camber morphing wing UAVs. In Proceedings of the AIAA Propulsion and Energy 2020 Forum, Online, 24–28 August 2020; p. 3958.
33. Vasista, S.; Riemenschneider, J.; Van de Kamp, B.; Monner, H.P.; Cheung, R.C.; Wales, C.; Cooper, J.E. Evaluation of a compliant droop-nose morphing wing tip via experimental tests. *J. Aircr.* **2017**, *54*, 519–534. [[CrossRef](#)]
34. Barbarino, S.; Flores, E.S.; Ajaj, R.M.; Dayyani, I.; Friswell, M.I. A review on shape memory alloys with applications to morphing aircraft. *Smart Mater. Struct.* **2014**, *23*, 063001. [[CrossRef](#)]
35. Sun, J.; Guan, Q.; Liu, Y.; Leng, J. Morphing aircraft based on smart materials and structures: A state-of-the-art review. *J. Intell. Mater. Syst. Struct.* **2016**, *27*, 2289–2312. [[CrossRef](#)]
36. Brailovski, V.; Terriault, P.; Georges, T.; Coutu, D. SMA actuators for morphing wings. *Phys. Procedia* **2010**, *10*, 197–203. [[CrossRef](#)]
37. DiPalma, M.; Gandhi, F. Autonomous camber morphing of a helicopter rotor blade with temperature change using integrated shape memory alloys. *J. Intell. Mater. Syst. Struct.* **2021**, *32*, 499–515. [[CrossRef](#)]
38. Lv, B.; Wang, Y.; Lei, P. Effects of Trailing Edge Deflections Driven by Shape Memory Alloy Actuators on the Transonic Aerodynamic Characteristics of a Super Critical Airfoil. *Actuators* **2021**, *10*, 160. [[CrossRef](#)]

39. Elahinia, M.H.; Ashrafiuon, H. Nonlinear control of a shape memory alloy actuated manipulator. *J. Vib. Acoust.* **2002**, *124*, 566–575. [[CrossRef](#)]
40. Kirkpatrick, K.; Valasek, J. Active length control of shape memory alloy wires using reinforcement learning. *J. Intell. Mater. Syst. Struct.* **2011**, *22*, 1595–1604. [[CrossRef](#)]
41. Kirkpatrick, K.; Valasek, J.; Haag, C. Characterization and control of hysteretic dynamics using online reinforcement learning. *J. Aerosp. Inf. Syst.* **2013**, *10*, 297–305. [[CrossRef](#)]
42. Chen, W.; Fuge, M. B\`ezierGAN: Automatic Generation of Smooth Curves from Interpretable Low-Dimensional Parameters. *arXiv* **2018**, arXiv:1808.08871.
43. Lepine, J.; Guibault, F.; Trepanier, J.Y.; Pepin, F. Optimized nonuniform rational B-spline geometrical representation for aerodynamic design of wings. *AIAA J.* **2001**, *39*, 2033–2041. [[CrossRef](#)]
44. Yasong, Q.; Junqiang, B.; Nan, L.; Chen, W. Global aerodynamic design optimization based on data dimensionality reduction. *Chin. J. Aeronaut.* **2018**, *31*, 643–659.
45. Grey, Z.J.; Constantine, P.G. Active subspaces of airfoil shape parameterizations. *AIAA J.* **2018**, *56*, 2003–2017. [[CrossRef](#)]
46. Abbott, I.H.; Von Doenhoff, A.E.; Stivers, L., Jr. *Summary of Airfoil Data*; No. NACA-TR-824; National Advisory Committee for Aeronautics, Langley Memorial Aeronautical Laboratory: Langley Field, VA, USA, 1945. Available online: <https://ntrs.nasa.gov/citations/19930090976> (accessed on 6 September 2013).
47. Silisteanu, P.D.; Botez, R.M. Two-dimensional airfoil design for low speed airfoils. In Proceedings of the AIAA Atmospheric Flight Mechanics Conference, Monterey, CA, USA, 5–8 August 2012.
48. Thomas, N.; Poongodi, D.P. Position control of DC motor using genetic algorithm based PID controller. In Proceedings of the World Congress on Engineering, London, UK, 1–3 July 2009; Volume 2, pp. 1–3.
49. Hassani, V.; Tjahjowidodo, T.; Do, T.N. A survey on hysteresis modeling, identification and control. *Mech. Syst. Signal Process.* **2014**, *49*, 209–233. [[CrossRef](#)]
50. Ma, J.; Huang, H.; Huang, J. Characteristics analysis and testing of SMA spring actuator. *Adv. Mater. Sci. Eng.* **2013**, *2013*, 823594:1-823594:7 [[CrossRef](#)]
51. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 2018.
52. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Proceedings of the International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 1861–1870.
53. Van Hasselt, H.; Guez, A.; Silver, D. Deep reinforcement learning with double q-learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30.
54. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. *arXiv* **2013**, arXiv:1312.6114.
55. Papamakarios, G.; Nalisnick, E.T.; Rezende, D.J.; Mohamed, S.; Lakshminarayanan, B. Normalizing Flows for Probabilistic Modeling and Inference. *J. Mach. Learn. Res.* **2021**, *22*, 1–64.
56. Savitzky, A.; Golay, M.J. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.* **1964**, *36*, 1627–1639. [[CrossRef](#)]