Towards Intelligent Fault-Tolerant Attitude Control of Fixed-Wing Aircraft

Alex B. Zongo and Li Qing(✉)

Tsinghua University, Beijing 100084, China
zongoa10@mails.tsinghua.edu.cn, liqing@tsinghua.edu.cn

Abstract. This study advances flight control systems by integrating deep reinforcement learning to enhance fault tolerance in fixed-wing aircraft. We assess the efficiency of Cross-Entropy Method Reinforcement Learning (CEM-RL) and Proximal Policy Optimization (PPO) algorithms in developing an adaptive stable attitude controller. Our proposed frameworks, focusing on smooth actuator control, showcase improved robustness across standard and fault-induced scenarios. The algorithms demonstrate unique traits in terms of trade-offs between trajectory tracking and control smoothness. Our approach that results in state-of-the-art performance with respect to benchmarks, presents a leap forward in autonomous aviation safety.

Keywords: flight control · fault-tolerance · robustness · reinforcement learning · evolutionary strategies · stability · control smoothness

1 Introduction

With the rapid advancements in Artificial Intelligence (AI), the aviation industry is starting to leverage its benefits and enhance flights controllers' capabilities. Faults manifest in aircraft systems as sensor errors, unexpected phenomena, and system or structural failures [9]. Mitigating these, requires passive or active control strategies [3,5], often implemented via gain schedulers [3] or hardware redundancy. They depend on prior fault knowledge [5], hence limiting generalization to only known fault types [3,7]. Reinforcement Learning (RL), particularly Approximate Dynamic Programming, has shown promise in advanced flight control, for instance, in F-16 jets [2,15]. Deep Reinforcement Learning (DRL) algorithms like Twin Delayed Deep Deterministic (TD3) [18] and Soft Actor-Critic (SAC) [22] demonstrated remarkable fault-tolerance in [7,10] without needing prior model dynamics knowledge. However, on top of RL's limitations, noisy action commands make hardware implementation difficult. A recent focus towards combining RL with Evolutionary Strategies in [4,12,14,20], is leading to innovative promising optimization algorithms for fault-tolerant control [7], despite computational and efficiency challenges. This study builds on the literature, proposing and evaluating frameworks and algorithms for enhanced fault tolerance and

L. Yan et al. (Eds.): ICGNC 2024, LNEE 1353, pp. 1–10, 2025.
https://doi.org/10.1007/978-981-96-2264-1_15

robustness, validated on a high-fidelity Cessna Citation 500 simulation from PH-LAB[1] [8,11].

2 Fundamentals

This section states the problem and introduces the learning framework and algorithms used for this study.

2.1 Reinforcement Learning Problem

In conventional Markov Decision Process-based reinforcement learning, an agent applies an action $a_t \in R^m$ at each time-step and state of a system, and receives feedback in the form of the next state and a reward. The agent's objective is to optimize a policy mapping states to actions, thereby maximizing cumulative rewards. This study focuses on optimizing an aircraft's attitude control, aiming to minimize tracking errors and ensure action smoothness.

Definition 1. *Given a state vector $s(t)$, a control input $u(t)$, and a reference input vector $r(t)$, the optimization problem is defined by Eqs. 1.*

$$u^* = arg\min_u \int_{t_0}^{t_f} L_s(s,u,r) + L_u(u)dt \text{ s.t} \quad |u| \leq u_{max} \text{ and } |\Delta u| \leq \frac{u_{max}}{\Delta t} \quad (1)$$

where L_s minimizes the deviation from the reference trajectory and L_u optimizes for smooth changes between subsequent control inputs.

2.2 Cross-Entropy Method and Proximal Policy Optimization

CEM is an Estimation of Distribution Algorithm that represents a population of policies as a distribution using a covariance matrix. Coupled with TD3 [18], it forms a Deep Neuro-Evolutionary algorithm known as CEM-RL or CEM-TD3 [1] benefiting from TD3's gradient-based policy improvement and CEM's efficiency to refine policy parameters effectively, as shown in Fig. 1a.

PPO [6], known for its successful application in robotics, optimizes a clipped surrogate objective function alongside a value function, balancing the reward maximization while mitigating large policy updates. Figure 1b presents a high-level overview of PPO's learning process.

3 Methodology

This section presents the design of the controller and experiment designs including the training and evaluation strategies.

[1] https://cs.lr.tudelft.nl/citation.

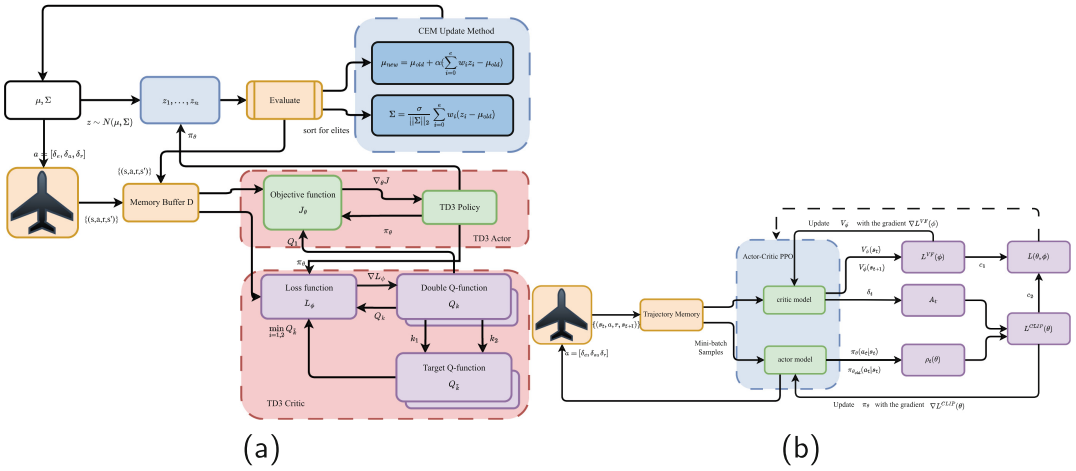


Fig. 1. CEM-TD3 Architecture **(a)**. PPO Learning Architecture **(b)**.

3.1 Aircraft Model and Interface

The aircraft model is a high-fidelity 6-Degrees of Freedom non-linear dynamics borrowed from PH-LAB by TU Delft [8,11].

The model is trimmed for specific flight regimes summarized in Table 1, with complete state of the aircraft, denoted by $x \in \mathbb{R}^{12}$, and defined in Eq. 2.

$$x = [p, q, r, V_{tas}, \alpha, \beta, \theta, \phi, \psi, H, X_e, Y_e]^T \tag{2}$$

where X_e, Y_e represent the longitudinal and lateral displacements relative to the trim point. p, q and r are the pitch, roll and yaw rate. θ, ϕ, ψ are the pitch, roll and heading angles. α, β are the angle of attack and side-slip angles. V_{tas} is the true airspeed, and H the altitude.

Table 1. Evaluation Cases and Trim Conditions from [7]

Name	Description
Nominal	Trimmed at H=2,000 m and $V_{tas} = 90\,m/s$
Iced Wings	α_{max} is reduced by 30% and the C_D increased with 0.6
Aft Shifted CG	The center of gravity is shifted aft by 0.25 m
Saturated Aileron	Deflection clipped at $\pm 1°$
Saturated Elevator	Deflection clipped at $\pm 2.5°$
Partial Loss of Elevator	A 70% reduction in effectiveness coefficient
Jammed Rudder	Rudder stuck at 15°
High Dynamic Pressure	Trimmed at H=2,000 m and $V_{tas} = 150\,m/s$
Low Dynamic Pressure	Trimmed at H=10,000 m, $V_{tas} = 90\,m/s$
Wind and Sensor Noise	Identified from flight tests [7] and isolated from [13]

The controller commands the aircraft's elevator δ_e, ailerons δ_a and rudder δ_r surfaces, confined within physical limitations [16] and mapped by Eqs. 4–5. An inner auto-throttle handles thrust [11], while trim tab and flap deflections are held at zero. In Eq. 3, the observed states are derived from the complete state at 100 Hz (p, q, r) and augmented with the pitch, roll and side-slip tracking error.

$$s := [\Delta\theta, \Delta\phi, 0 - \beta, p, q, r] \tag{3}$$

$$a_t := [\delta_e, \delta_a, \delta_r]^T \in [-1, 1]^3 \tag{4}$$

$$u := u_{min} + (a_t + 1)\frac{u_{max} - u_{min}}{2} \tag{5}$$

where a_t is the controller's actions. u_{min}, u_{max} represent the minimum and maximum actuator's angular deflection used to map a_t into the control input u. The environment returns a reward signal that minimizes the tracking error and prevents abrupt changes to the control inputs by keeping the body rates low and also using a smoothness metric S_m introduced in [7].

Definition 2. *Given* $\dot{x} := [p, q, r]^T$, $\delta X = [\theta_r - \theta, \phi_r - \phi, 0 - \beta]$, $c_r = \frac{6}{\pi}[1, 1, 4]$, *a scaling factor, and* $w_{1,2,3} | \sum w_i = 1$ *weight coefficients, the reward function is defined by Eq. 6.*

$$R = -\frac{w_1}{3}||\dot{x}||_1 - \frac{w_2}{3}||clip(c_r \cdot \delta X, -1, 1)||_1 - \frac{2w_3}{\Delta T}(T_{max} - T) + S_m \tag{6}$$

Note that S_m introduced in [7] is adapted to fit in the instantaneous reward signal. It measures the smoothness of the actions taken so far via a weighted sum of frequency amplitudes [7] through a Fast Fourier Transform.

3.2 Experiment Configuration

Both algorithms are trained offline on the normal plant dynamics for 2000 steps per episode with cosine smoothed reference signals uniformly sampled ($\theta \in [-25°, 25°]$, $\phi \in [-45°45°]$, $\beta_r = 0$) [10,19,21]. All computations are done on 12 Intel(R) i7-5930K 3.5GHz CPU cores with NVIDIA GeForce GTX TITAN X graphics computer. An online evaluation framework inspired by [7] is designed. Before training, appropriate hyperparameters were determined via an hyperparameters sweep. The process of evaluation and adaption for the designed control system is described on a higher level in Fig. 2. A Gaussian Process used and trained as Model Identification function.

Definition 3. *Given a set of data* $\mathcal{D} = \{(s_k, s_{k+1}) : k = 1, \ldots, N - 1\}$ *drawn from an unknown function* $s_{k+1} = f(s_k)$, *the objective is to find* f *in order to be used to predict new observed values given* s_k^*. *[17].*

$$f(s) \sim \mathcal{GP}(m(s), k(s, s')) \tag{7}$$

$$s_{k+1} = f(s_k) + \epsilon_k, \ \epsilon_k \sim \mathcal{N}(0, \sigma_n^2) \tag{8}$$

where f *is a Gaussian Process with mean* $m = \mathbb{E}[f(s)]$ *and covariance function or kernel* $k(s, s') = Cov[f(s), f(s')]$ *which is usually a square exponential covariance function [17].*

With fully observed state and control input s_t, *and* u_t, f is trained to predict $s_{t+1} = f(s_t, u_t)$. This allows the policies in the database to be evaluated predictively and in parallel, over a time horizon, here $t_h = 10s$ every 2 s. Hence, the new policy is chosen if performance (tracking error over t_h) exceeds to a certain degree the current policy (by 25%), which is softly replaced via the Polyack update mechanism also used in TD3 [18].

Stability of the trained controllers is performed via a linear model that is trained on experiences collected from a simulated agent-environment (for various flight conditions) to approximated the A and B matrices in Eq. 9 via mean squared error loss function. The system takes as input a reference signal and outputs the state (pitch, roll and side-slip angle) of the aircraft.

$$g(X) = X' = A \cdot X + B \cdot u \tag{9}$$

where $u = [\theta_r, \phi_r, 0]^T$ while $X = [\theta, \phi, \beta]$. Hence A and B are 3-by-3 matrices.

Definition 4. *Given experience data from the simulation* $ref = [\theta_r, \phi_r, 0]$ *and* $y = [\theta_t, \phi_t, \beta_t]$, $g(ref) = \hat{y} = [\theta', \phi', \beta']$, *a gradient descent algorithm is used to update* A *and* B. $\eta = 0.001$ *is chosen as learning rate.*

$$L(X, u) = ||y - \hat{y}||_2^2 \tag{10}$$

$$A \leftarrow A - \eta \nabla_X L \tag{11}$$

$$B \leftarrow B - \eta \nabla_u L \tag{12}$$

The eigenvalues of A are then used to indicate the stability of the overall system.

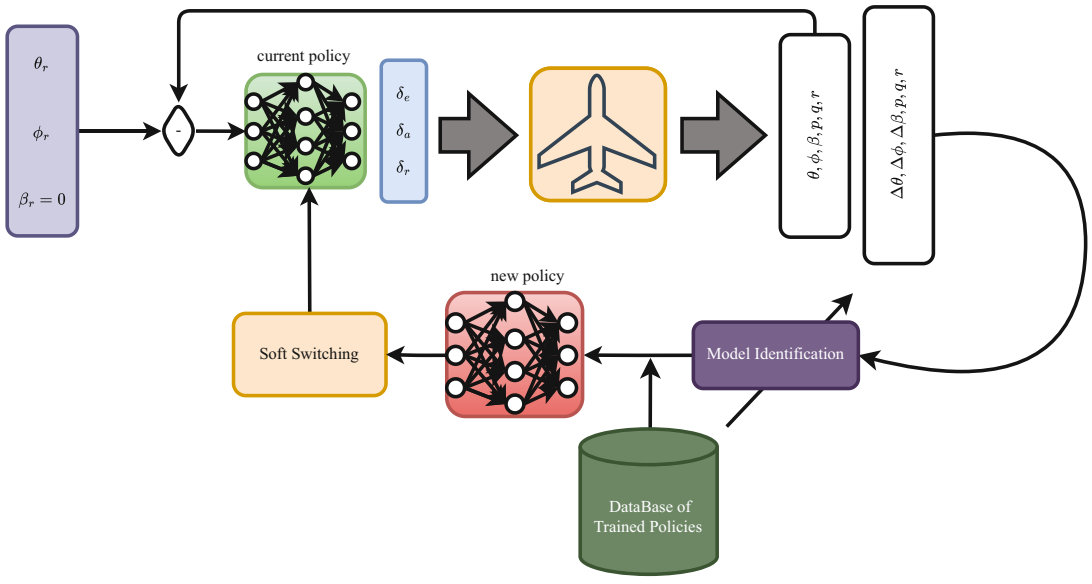


Fig. 2. High-Level Adaptation Mechanism

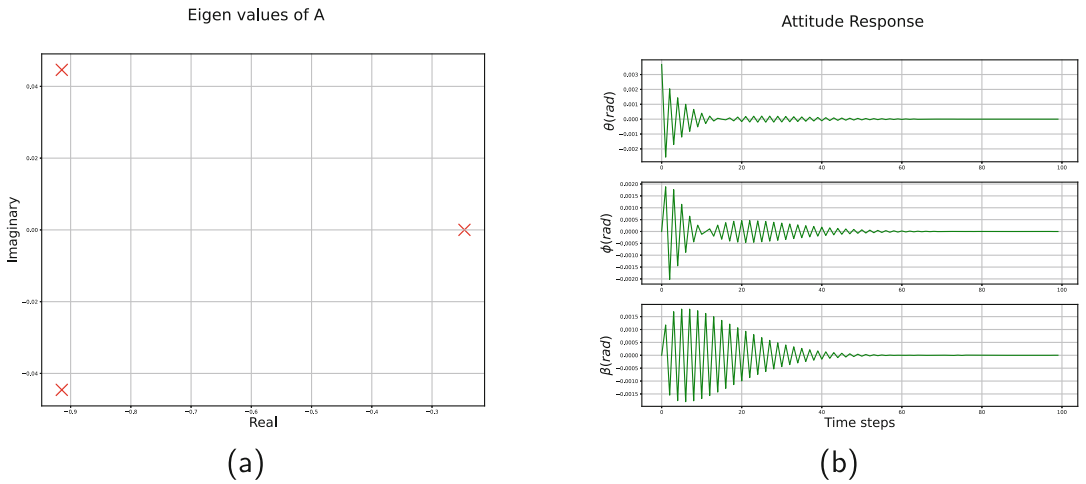


Fig. 3. Stability Analysis (a) Eigen-values (b) Time series attitude response on Partial Loss of Elevator Model with PPO controller.

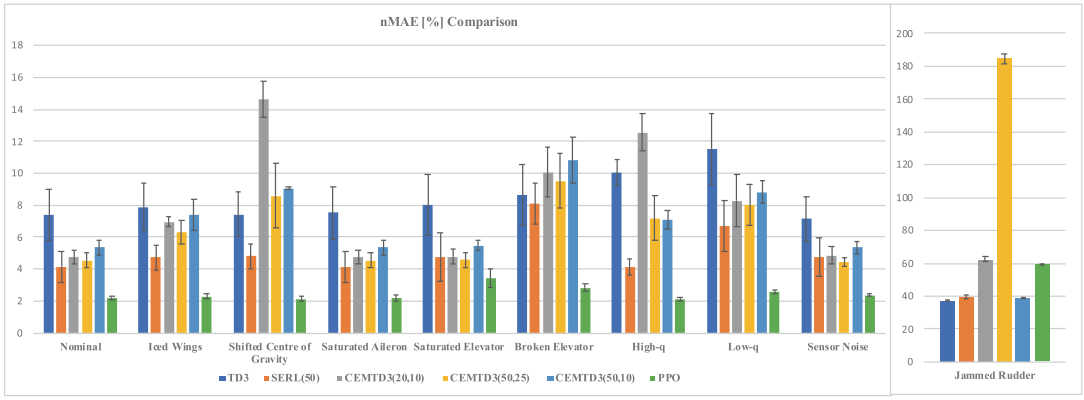
4 Results and Discussion

Various fault and disturbance cases outlined in Table 1 were used to assess the agent's tolerance and adaptation. Figure 4a presents comparative results of the proposed framework and algorithms alongside related works, considering a similar reference trajectory. CEM-TD3 shows state-of-the-art action smoothness in all cases as depicted in Fig. 4b, exceeding the benchmark results while PPO dominates in terms of tracking error. However, the jammed rudder environment remains notably challenging.

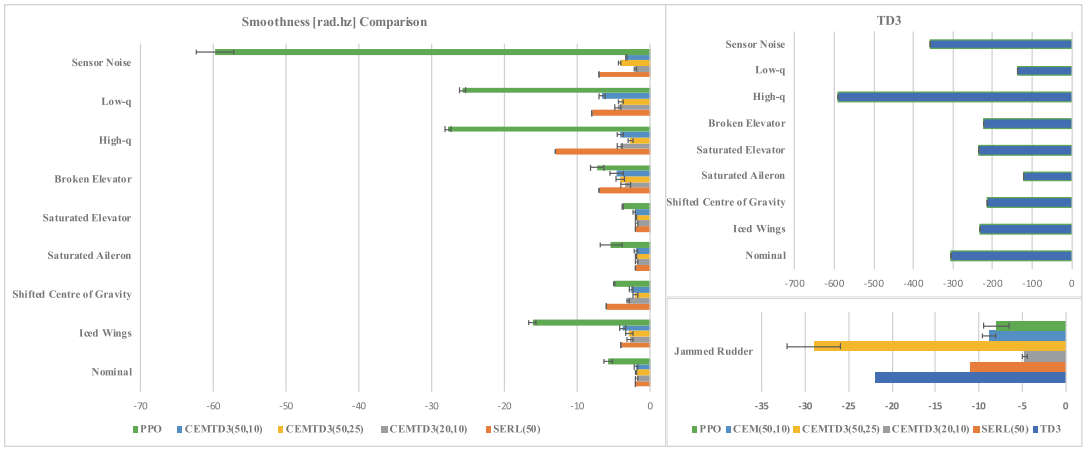
In addition, a stability analysis, conducted by linearizing the combined controller and plant model dynamics, indicates that the systems are stable, even in the presence of faults as shown in Fig. 3. The sharpness in the attitude response could be explained by the linearized dynamics. Nonetheless, such feature is mitigated by the very low amplitude of the oscillations.

Further analyses reveal that the adaptation mechanism in Fig. 2 is comparatively efficient in being robust and adaptive like the standalone trained controllers.

Moreover, by comparing the inputs between a normal flight and a partial loss of the elevator in Fig. 5, a more accentuated deflection of the elevator signal indicates a more pronounced strategy to mitigate the failure, since, in this case the tail is 70% less efficient. Overall, CEM-TD3 generally exhibits less aggressive control and higher nMAE values across all scenarios, indicating a more conservative control strategy but with less precision in following the desired trajectory. PPO appears, on the other hand, to be the most robust system, able to maintain trajectory (nMAE $\leq 2.7\%$) in almost all tested conditions, but its more aggressive control nature must be considered against potential trade-offs like hardware applicability and passenger comfort.

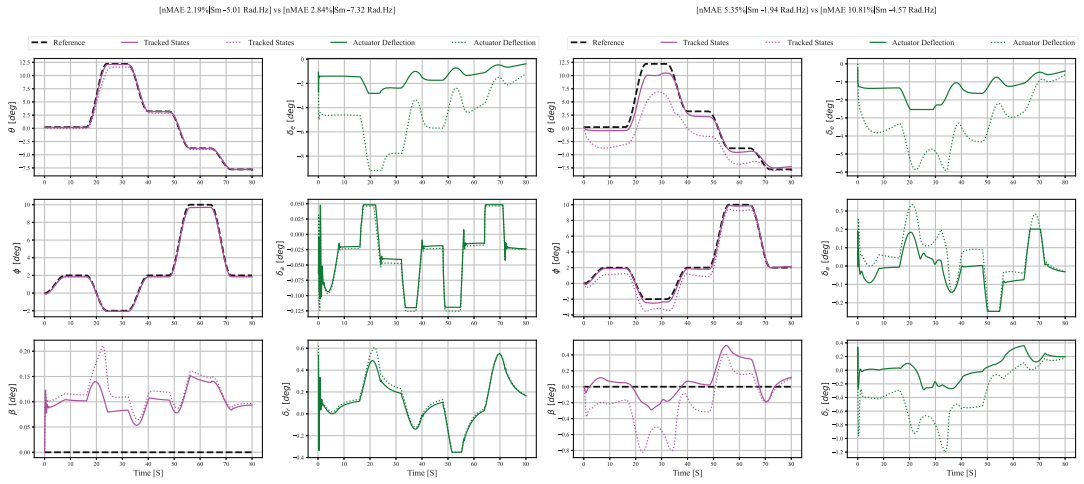


(a)



(b)

Fig. 4. Comparison between CEM-TD3 and PPO with the literature TD3 and SERL(50) on evaluation cases. (Best performing agents) (a) normalized Mean Absolute tracking Error (nMAE) (b) Action Policy Smoothness.



(a) (b)

Fig. 5. A Normal Flight (straight line [-]) versus Flight with Partial Loss of Elevator (dotted line [·]) (a) PPO agent (b) CEM-TD3 Agent.

5 Summary and Conclusions

This work combines two bio-inspired frameworks, Deep Reinforcement Learning (TD3) and Evolutionary Strategy (CEM), resulting in CEM-TD3 alongside PPO to train controllers on a non-linear fixed-wing aircraft model, aiming at optimal and smooth attitude control. This study advances the field by improving action policy smoothness and fault tolerance. The CEM-TD3's trade-off between tracking accuracy and control smoothness, and PPO's robust adaptation across various operational scenarios significantly matches existing benchmarks.

The growing complexity of autonomous systems requires sophisticated and adaptable controllers. A model capable of handling unforeseen faults is invaluable, given the unpredictability of potential scenarios. The improved action policy smoothness contributes to system efficiency, particularly in energy consumption, making hardware applicability more flexible. Overall, these results represent a step towards integrating AI into fault-tolerant and safety-critical systems.

Future improvements can focus on expanding the range of fault scenarios to complex concurrent fault conditions. Moreover, enhancing the online adaptation frameworks with efficient model estimation techniques could offer valuable insights. Finally, validating the results requires practical flight tests. Advances in the explainability of NN-based controllers shall increase trustworthiness and reliability.

Acknowledgement. This research is supported by the Tsinghua University Initiative Scientific Research Program (20234616001), the National Natural Science Foundation of China (No. 61771281), and the Science and Technology Innovation 2030-"new generation artificial intelligence" major project (2018AAA0101605).

References

1. Aloïs, P., Olivier, S.: CEM-RL: Combining evolutionary and gradient-based methods for policy search (2019). https://doi.org/10.48550/arXiv.1810.01222, 1810.01222
2. Bo, S., Eric-Jan, V.K.: Incremental model-based global dual heuristic programming for flight control. IFAC-PapersOnLine **52**(29), 7–12 (2019). https://doi.org/10.1016/j.ifacol.2019.12.613
3. Christopher, E., Thomas, L., Hafid, S.: Fault tolerant flight control a benchmark challenge. Lecture Notes Control Inform. Sci. **399**, 1–560 (2010)
4. Cully, A., Clune, J., Tarapore, D., Mouret, J.B.: Robots that can adapt like animals. Nature **521**(7553), 503–7 (2015). https://doi.org/10.1038/nature14422, https://www.ncbi.nlm.nih.gov/pubmed/26017452
5. Eugene, L., Kevin, A.W.: Robust and Adaptive Control, 1st edn. Advanced Textbooks in Control and Signal Processing (2013). https://doi.org/10.1007/978-1-4471-4396-3
6. Filip, J.S., Prafulla, W., Alec, R.D., Oleg, K.: Proximal policy optimization algorithms (2017). https://doi.org/10.48550/arXiv.1707.06347

7. Gavra, V.: Evolutionary reinforcement learning: A hybrid approach for safety-informed intelligent fault-tolerant flight control. Thesis, TU Delft (2022). http://repository.tudelft.nl/
8. van den Hoek, M.A., de Visser, C.C., Pool, D.M.: Identification of a Cessna Citation II Model Based on Flight Test Data. book section Chapter 14, pp. 259–277 (2018). https://doi.org/10.1007/978-3-319-65283-2_14
9. Isermann, R., Ballé, P.: Trends in the application of model-based fault detection and diagnosis of technical processes. Control. Eng. Pract. **5**(5), 709–7 (1997). https://doi.org/10.1016/S0967-0661(97)00053-1
10. Killian, D., Erik-Jan, V.K.: Soft actor-critic deep reinforcement learning for fault tolerant flight control. In: AIAA SCITECH 2022 Forum, American Institute of Aeronautics and Astronautics (2022). https://doi.org/10.2514/6.2022-2078, https://doi.org/10.2514/2F6.2022-2078
11. Linden, V.D.: DASMAT-Delft university aircraft simulation model and analysis tool: A Matlab/Simulink environment for flight dynamics and control analysis (1998). http://resolver.tudelft.nl/uuid:25767235-c751-437e-8f57-0433be609cc1
12. Mehrdad, D., In Soo, S., Mark, T.: An introduction to genetic algorithms and evolution strategies (2002). https://www.semanticscholar.org/paper/An-Introduction-to-Genetic-Algorithms-and-Evolution-Dianati-Song/79eabba1c148c7ac3c33e50895bec4d41a5fed2b
13. Moorhouse, D., Woodcock, R.: Background information and user guide for mil-f-8785c, military specification-flying qualities of piloted airplanes. Tech. rep, Air Force Wright Aeronautical Labs Wright-Patterson AFB OH (1982)
14. Papavasileiou, E., Cornelis, J., Jansen, B.: A systematic literature review of the successors of "neuroevolution of augmenting topologies. Evol. Comput. **29**(1), 1–7 (2021). https://doi.org/10.1162/evco_a_00282
15. QiPing, C., Zhou, Y., Eric-Jan, V.K.: Incremental approximate dynamic programming for nonlinear flight control design. In: Proceedings of the 3rd CEAS EuroGNC: Specialist Conference on Guidance, Navigation and Control, Toulouse, France, 13-15 April 2015 (2015)
16. Ramesh, K., Erik-Jan, V.K., Gertjan, L.: Reinforcement learning based online adaptive flight control for the cessna citation II(PH-LAB) aircraft. In: AIAA SCITECH 2022 Forum, American Institute of Aeronautics and Astronautics (2021). https://doi.org/10.2514/6.2021-0883
17. Rasmussen, C.E., Williams, C.K.I.: Gaussian processes for machine learning. Adaptive computation and machine learning, MIT Press (2006). https://www.worldcat.org/oclc/61285753
18. Scott, F., Herke, V.H., David, M.: Addressing function approximation error in actor-critic methods (2018). https://doi.org/10.48550/arXiv.1802.09477
19. Seres, P., Erik-Jan, V.K., Liu, C.: Distributional reinforcement learning for flight control: A risk-sensitive approach to aircraft attitude control using distributional RL. Master's thesis, TU Delft (2022). http://resolver.tudelft.nl/uuid:6cd3efd1-b755-4b04-8b9b-93f9dabb6108
20. Stanley, K.O., Clune, J., Lehman, J., Miikkulainen, R.: Designing neural networks through neuroevolution. Nat. Mach. Intell. **1**(1), 24–35 (2019). https://doi.org/10.1038/s42256-018-0006-z

21. Teirlinck, C., Erik-Jan, V.K.: Reinforcement learning for flight control: Hybrid offline-online learning for robust and adaptive fault-tolerance. Master's thesis, TU Delft (2022). http://resolver.tudelft.nl/uuid:dae2fdae-50a5-4941-a49f-41c25bea8a85
22. Tuomas, H., Aurick, Z., Pieter, A., Sergey, L.: Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In: Proceedings of the 35th International Conference on Machine Learning, vol. 80, pp. 1861–1870 (2018). https://proceedings.mlr.press/v80/haarnoja18b.html