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

Alex B. Zongo[1] and Li Qing[1]

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

Abstract. This paper presents and evaluates an approach to flight control systems using deep reinforcement learning to enhance fault-tolerance in fixed-wing aircraft. The study explores the Cross-Entropy Methods (CEM) and Proximal Policy Optimization (PPO) algorithms to develop a self-learning attitude controller capable of robust operations and adapting to unexpected failures while maintaining smooth actuator control. The algorithms demonstrate each, unique traits in terms of trade-off between trajectory tracking and control smoothness. A stability analysis shows stable Neural-Network based control. Overall, the trained agents exceeds the state-of-art on normal flight and on more than six fault benchmark scenarios.

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

1 Introduction

Artificial Intelligence (AI) is everywhere. Despite the aviation industry's strong safety record, leveraging AI advances can enhance flight controllers capabilities and further reduce risks. Faults manifest in aircraft systems as sensor errors, unexpected phenomena, system or structural failures [11]. Mitigating these, requires passive or active control strategies [5, 13], often implemented via gain schedulers [5] or hardware redundancy. They depend on prior fault knowledge [13], hence limiting generalization to only known fault types [5, 8]. Reinforcement Learning (RL), particularly Approximate Dynamic Programming, has shown promises in advanced flight control, for instance, in F-16 jets [1, 20]. Deep Reinforcement Learning (DRL) algorithms like Twin Delayed Deep Deterministic (TD3) [7] and Soft Actor Critic (SAC) [9] demonstrated remarkable fault-tolerance in [3, 8] without needing prior model dynamics knowledge. However, on top of RL's limitations, noisy action command, makes hardware implementation difficult. Recent focus towards combining RL with Evolutionary Strategies in [2, 4, 6, 19], is leading to innovative promising optimization algorithms for fault-tolerant control [8], despite computational and efficiency challenges. This study builds on the literature, proposing and evaluating frameworks and algorithms for enhanced fault tolerance and robustness, validated on a high-fidelity Cessna Citation 500 simulation from PH-LAB [1] [10, 14].

2 Fundamentals

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

2.1 Reinforcement Learning Problem

In traditional reinforcement learning with a Markov Decision Process setup, at a each time-step and state of a system, an agent applies an action $a_t \in R^m$ to it which returns 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.*

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

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 co-variance matrix. Coupled with TD3 [7], it forms a Deep Neuro-Evolutionary algorithm known as CEM-RL [16] benefiting from TD3's gradient-based policy improvement and CEM's efficiency to refine policy parameters effectively, as shown in **Fig**. 1a.

PPO [17], 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. The total loss combines the policy's expected advantage and the value function's accuracy, as outlined in Eqs. 2–3 and **Fig.** 1b.

$$L^{CLIP}(\theta) = E_t[\min(\frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)}\widetilde{A}_t, clip(\frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)}, 1-\epsilon, 1+\epsilon)\widetilde{A}_t)] \quad (2)$$

$$L^{VF}(\phi) = E_t[(V_\phi(s_t) - V(s_t))^2] \quad (3)$$

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

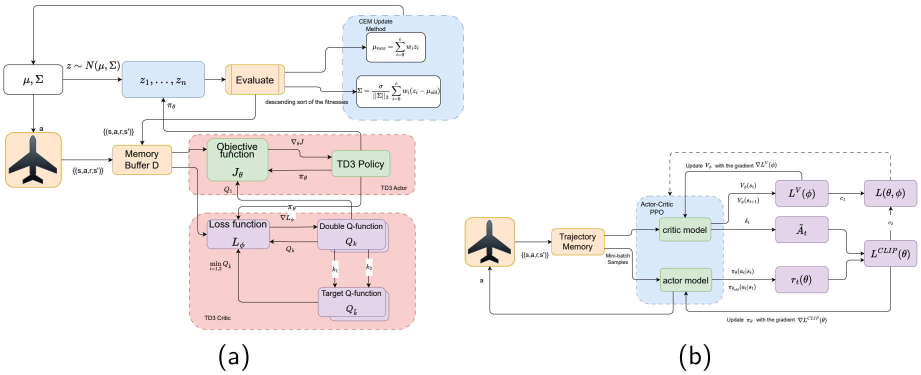


Fig. 1. CEM-TD3 Architecture (a). PPO Learning Architecture (b).

3 Methodology

This section presents the design of the controller and experiment setup which includes the training and evaluation strategies.

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 [10, 14]. The model is trimmed for specific flight conditions summarized in table 3.1. The complete state of the aircraft, denoted by $x \in \mathbb{R}^{12}$, is defined in Eq. 4.

$$x = [p, q, r, V_{tas}, \alpha, \beta, \theta, \phi, \psi, H, X_e, Y_e]^T \tag{4}$$

where X_e, Y_e represent the longitudinal and lateral displacements relative to the trim point.

Name	Description
Nominal	Trim condition: H=2,000m and $V_{tas} = 90m/s$
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	Aileron deflection clipped at ±1°
Saturated Elevator	Elevator deflection clipped at ±2.5°
Partial Loss of Elevator	Elevator effectiveness coefficient multiplied by 0.3 gain
Jammed Rudder	Rudder stuck at 15°
High Dynamic Pressure	Trim conditions: H=2,000m and $V_{tas} = 150m/s$
Low Dynamic Pressure	Trim conditions: H=10,000m, $V_{tas} = 90m/s$
Wind and Sensor Noise	models identified from flight tests [8] and isolated from [15]

Table 1. Evaluation Cases and Trim Conditions from [8]

The controller commands the aircraft's primary control surfaces, i.e, the elevator, ailerons and rudder, confined within physical limitations [12] and mapped by Eqs. 6-7. An inner auto-throttle handles thrust [14], while trim tab and flap deflections are held at zero. In **Eqs. 5** the observed states are derived from the complete state at 100 Hz and augmented with the tracking error.

$$s := [\varDelta\theta, \varDelta\phi, 0-\beta, p, q, r] \tag{5}$$

$$a_t := [\delta_e, \delta_a, \delta_r]^T \in [-1,1]^3 \tag{6}$$

$$u := u_{min} + (a_t + 1)\frac{u_{max} - u_{min}}{2} \tag{7}$$

The environment returns a reward signal that minimizes the tracking error and prevents abrupt changes to the control inputs by means of keeping the body rates low and also using a smoothness metric S_m introduced in [8]. See **Fig.** 2c.

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. 8.*

$$R = -\frac{w_1}{3}||\dot{x}||_1 - \frac{w_2}{3}||clip(c_r \cdot \delta X, -1, 1)||_1 - \frac{2w_3}{\varDelta T}(T_{max} - T) + S_m \tag{8}$$

3.2 Experiment Setup

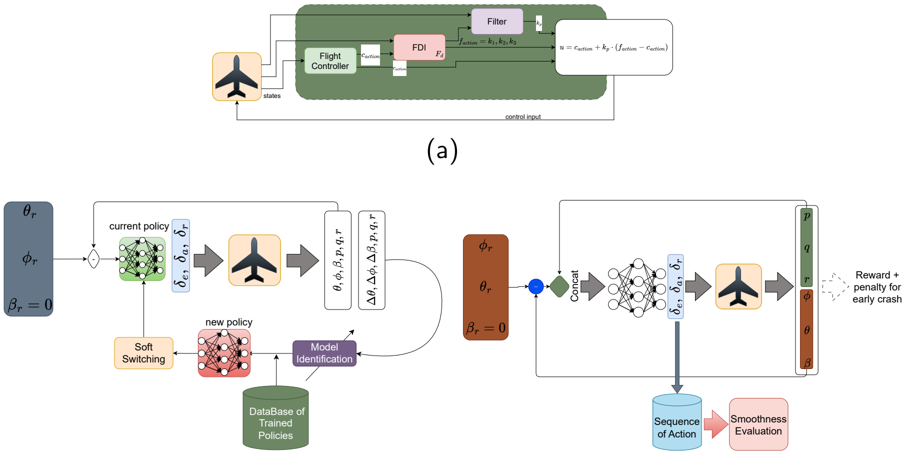
Both algorithms are trained offline on the normal plant dynamics for 2000 steps per episode. PPO is trained for more than 10^6 time steps requiring 2 hours, while CEMTD3 for 100 generations requiring almost 8h with population of up to 50. All the computations are done on a 12 intel(R) i7-5930K 3.5GHz CPU cores with NVIDIA GeForce GTX TITAN X graphics computer. An online evaluation framework inspired from [8] is designed for CEM-TD3 while both a Neural Network (NN) based Fault Detection and Identification unit and a NN-based filter are developed for the PPO controller. Note that prior training, a hyperparameters sweeping was conducted to extract appropriate values for the algorithms. Figures 2a and 2b describe, on a higher level, the process of evaluation and adaption for the designed control system.

4 Results and Discussion

This section presents and discusses the results of training, evaluation and stability analysis of the proposed methods in terms of learning curves, fault-tolerance and robustness.

4.1 Learning Curves

Figure 3 illustrates the average performance score of the generated population with respect to the number of generations for CEMTD3 and the episodic reward


states
Flight Controller
c_{action}
FDI
F_d
Filter
k_p
$f_{action} = k_1, k_2, k_3$
$u = c_{action} + k_p \cdot (f_{action} - c_{action})$
control input
θ_r
ϕ_r
$\beta_r = 0$
current policy
$\delta_e, \delta_a, \delta_r$
$\theta, \phi, \beta, p, q, r$
$\Delta\theta, \Delta\phi, \Delta\beta, p, q, r$
new policy
Soft Switching
Model Identification
DataBase of Trained Policies
Concat
p
q
r
ϕ
θ
β
Reward + penalty for early crash
Sequence of Action
Smoothness Evaluation

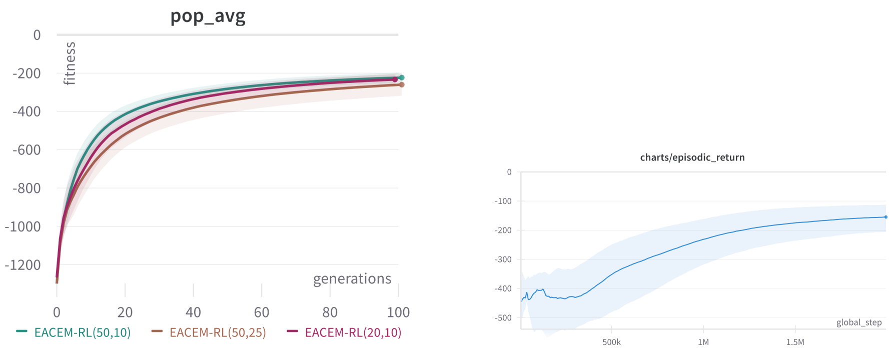
(a)

(b)

(c)

Fig. 2. (a) PPO **(b)** CEMTD3's agents High-Level Adaptation Mechanisms. **(c)** Control Interface

over the time steps for PPO. Indeed each new generation (CEMTD3) outperforms its predecessor. The returns are averaged and smoothed across three different seeds and converge towards values comparable to the results reported in related studies [3, 8, 18, 21]. Also, the smoothness of the actions improves during the training.



(a)

(b)

Fig. 3. (a) CEMTD3's Population Average Performance Score. **(b)** PPO's Episodic reward

4.2 Fault-Tolerance and Stability

About nine (9) fault and disturbance cases outlined in table 3.1 are used to assess the agent's tolerance and adaptation. Figure 4 presents comparative results of the proposed algorithm alongside with related works, considering a similar reference trajectory. Specifically, CEMTD3 shows state-of-the-art action smoothness in all cases as depicted in fig. 5 exceeding the benchmark results. PPO dominates in terms of tracking error. However, the jammed rudder environment remains notably challenging. In addition, a stability analysis was conducted by linearizing the combined controller-plants to check for the eigen-values and time-series responses. Again, the systems are stable. Further analyses reveal that the adaptation mechanisms in figs. 2 are comparatively efficient in being robust and adaptive just as the standalone bare-bone trained controllers. Indeed, with CEMTD3 in fig. 2b, given a database of pre-trained agents which can be updated, the system switches to appropriate policies based on a system identification model by predictively evaluating in parallel the strategies over some time horizon. Switching between control policies is done via Polyack update mechanism as used in TD3 [7]. In the PPO adaptation framework, the FDI was pre-trained to detect failure and to return corrective parameters which are later used by a filter to smooth out the control inputs.

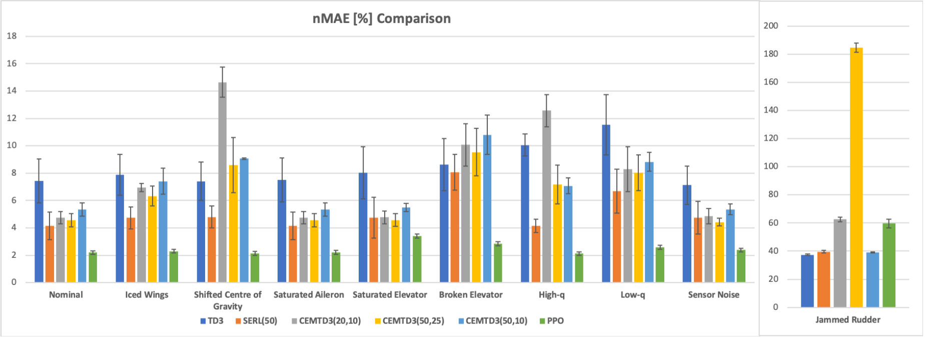


Fig. 4. Comparison of nMAE between CEMTD3(20,10), CEMTD3(50,25), CEMTD3(50,10), PPO with the literature TD3 and SERL(50) on evaluation cases. (Best performing agents). **Note**: CEMTD3(Population Size, Elite Size)

Moreover, by comparing the inputs between a normal flight and a partial loss of the elevator in Figure 6, a more accentuate deflection of the elevator signal is noticed which indicates an involved strategy to address the failure, since in this case the tail is 70% less efficient. In general, 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 involved control nature must be considered against potential trade-offs like hardware limits and passenger comfort.

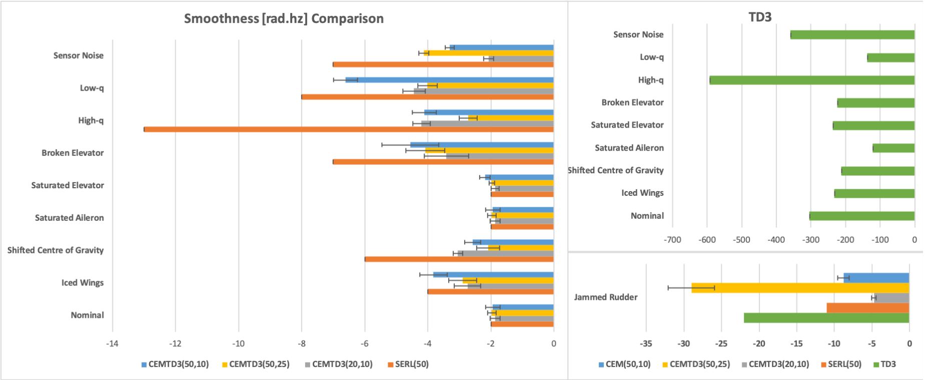
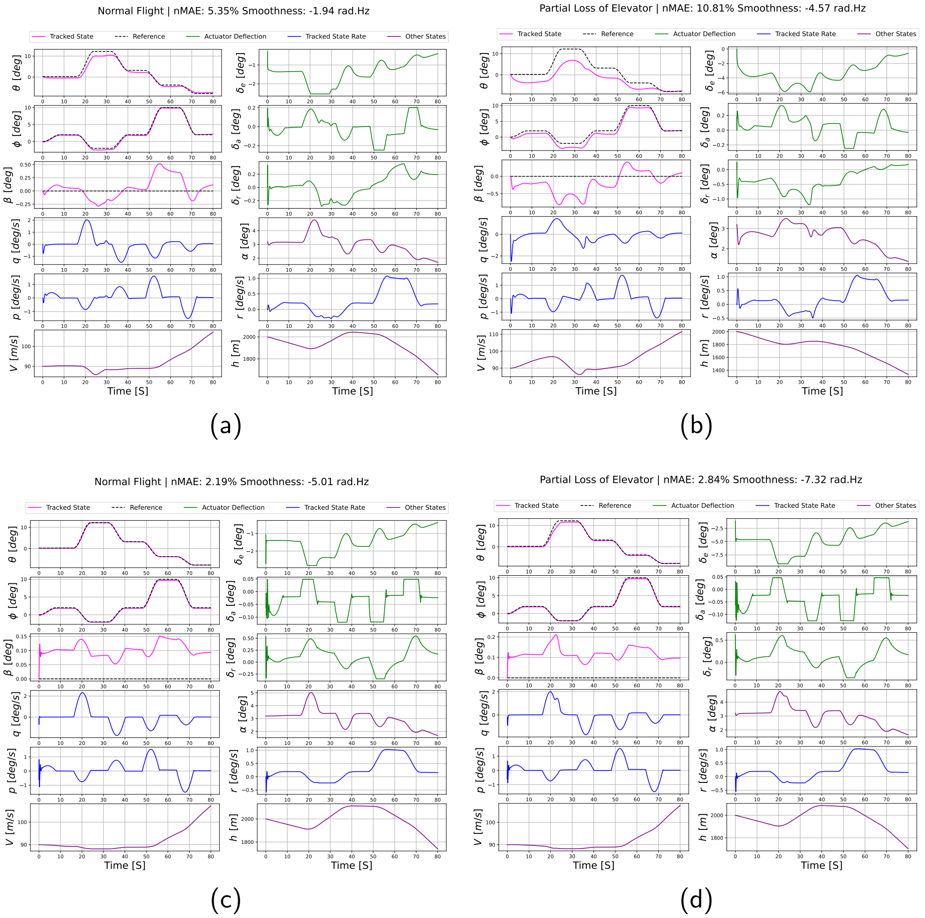


Fig. 5. Comparison of Action Policy Smoothness between CEMTD3(20,10), CEMTD3(50,25), CEMTD3(50,10) with TD3 and SERL(50) on evaluation scenarios.



(a)

(b)

(c)

(d)

Fig. 6. Evaluation Time-Traces of CEMTD3 (a) on Normal Flight , (b) With Partial Loss of Elevator, and PPO (c) on Normal Flight, (d) With Partial Loss of the Elevator.

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 CEMTD3's performance, particularly its balance between tracking accuracy and control smoothness, and PPO's robust adaptation across various operational scenarios significantly surpass existing benchmarks.

The growing complexity of autonomous systems requires sophisticated and adaptable controllers. In fact 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 more flexible hardware applicability. Overall, these results represents a step towards integrating AI into fault-tolerant and safety-critical systems.

Future improvements can focus on expanding the range of fault scenarios including complex concurrent fault fault conditions. Moreover, enhancing the online adaptation frameworks with advanced efficient model estimation techniques could offer valuable insights. Finally, validating the results requires real 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 science and technology innovation 2030-"new generation artificial intelligence" major project (2018AAA0101605).

Bibliography

[1] Chu Q, Y Z, EJ VK (2015) 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

[2] Cully A, Clune J, Tarapore D, Mouret JB (2015) Robots that can adapt like animals. Nature 521(7553):503–7, DOI 10.1038/nature14422, URL https://www.ncbi.nlm.nih.gov/pubmed/26017452

[3] Dally K, Kampen EJV (2022) Soft actor-critic deep reinforcement learning for fault tolerant flight control. In: AIAA SCITECH 2022 Forum, American Institute of Aeronautics and Astronautics, DOI 10.2514/6.2022-2078, URL https://doi.org/10.2514%2F6.2022-2078

[4] Dianati M, Song IS, Treiber M (2002) An introduction to genetic algorithms and evolution strategies. URL https://www.semanticscholar.org/paper/An-Introduction-to-Genetic-Algorithms-and-Evolution-Dianati-Song/79eabba1c148c7ac3c33e50895bec4d41a5fed2b

[5] Edwards C, Lombaerts T, Smaili H (2010) Fault tolerant flight control a benchmark challenge. Lecture notes in control and information sciences 399:1–560

[6] Evgenia P, Jan C, Bart J (2021) A systematic literature review of the successors of "neuroevolution of augmenting topologies. Evolutionary Computation 29(1):1–73, DOI 10.1162/evco_a_00282, URL https://doi.org/10.1162/evco_a_00282

[7] Fujimoto S, Hoof HV, Merger D (2018) Addressing function approximation error in actor-critic methods. https://arxivorg/abs/180209477 DOI https://doi.org/10.48550/arXiv.1802.09477

[8] Gavra V (2022) Evolutionary reinforcement learning: A hybrid approach for safety-informed intelligent fault-tolerant flight control. Thesis, TU Delft, URL http://repository.tudelft.nl/

[9] Haarnoja T, Z A, Abbeel P, Levine S (2018) 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, URL %Uhttps://proceedings.mlr.press/v80/haarnoja18b.html

[10] van den Hoek M A, de Visser C C, D MP (2018) Identification of a Cessna Citation II Model Based on Flight Test Data, book section Chapter 14, pp 259–277. DOI 10.1007/978-3-319-65283-2_14

[11] Isermann R, Ballé P (1997) Trends in the application of model-based fault detection and diagnosis of technical processes. Control Engineering Practice 5(5):709–719, DOI https://doi.org/10.1016/S0967-0661(97)00053-1, URL https://www.sciencedirect.com/science/article/pii/S0967066197000531

[12] Konatala R, Kampen EJV, Looye G (2021) 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, URL https://doi.org/10.2514/6.2021-0883

[13] Lavretsky E, Wise KA (2013) Robust and Adaptive Control, 1st edn. Advanced Textbooks in Control and Signal Processing, DOI https://doi.org/10.1007/978-1-4471-4396-3

[14] Linden Vd (1998) Dasmat-delft university aircraft simulation model and analysis tool: A matlab/simulink environment for flight dynamics and control analysis,. URL http://resolver.tudelft.nl/uuid:25767235-c751-437e-8f57-0433be609cc1

[15] Moorhouse D, Woodcock R (1982) 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

[16] Pourchot A, Sigaud O (2019) Cem-rl: Combining evolutionary and gradient-based methods for policy search. DOI https://doi.org/10.48550/arXiv.1810.01222, 1810.01222

[17] Schulman FJ, Wolski P, Dhariwal AR, Klimov O (2017) Proximal policy optimization algorithms. https://arxivorg/abs/170706347 DOI https://doi.org/10.48550/arXiv.1707.06347

[18] Seres P, Erik-Jan VK, Liu C (2022) Distributional reinforcement learning for flight control: A risk-sensitive approach to aircraft attitude control using distributional rl. Master's thesis, TU Delft, URL http://resolver.tudelft.nl/uuid:6cd3efd1-b755-4b04-8b9b-93f9dabb6108

[19] Stanley KO, Clune J, Lehman J, Miikkulainen R (2019) Designing neural networks through neuroevolution. Nature Machine Intelligence 1(1):24–35, DOI 10.1038/s42256-018-0006-z

[20] Sun B, E-J VK (2019) Incremental model-based global dual heuristic programming for flight control. IFAC-PapersOnLine 52(29):7–12, DOI https://doi.org/10.1016/j.ifacol.2019.12.613, URL https://www.sciencedirect.com/science/article/pii/S2405896319325558

[21] Teirlinck C, Erik-Jan VK (2022) Reinforcement learning for flight control: Hybrid offline-online learning for robust and adaptive fault-tolerance. Master's thesis, TU Delft, URL http://resolver.tudelft.nl/uuid:dae2fdae-50a5-4941-a49f-41c25bea8a85