Twin-Delayed Deep Deterministic Policy Gradient for altitude control of a flying-wing aircraft with an uncertain aerodynamic model

Völker, W.J.E.; Li, Y.; van Kampen, E.

DOI
10.2514/6.2023-2678

Publication date
2023

Document Version
Final published version

Published in
AIAA SciTech Forum 2023

Citation (APA)
Völker, W. J. E., Li, Y., & van Kampen, E. (2023). Twin-Delayed Deep Deterministic Policy Gradient for altitude control of a flying-wing aircraft with an uncertain aerodynamic model. In *AIAA SciTech Forum 2023* Article AIAA 2023-2678 (AIAA SciTech Forum and Exposition, 2023). https://doi.org/10.2514/6.2023-2678

AIAA SciTech Forum
23-27 January 2023, National Harbor, MD & Online
AIAA SCITECH 2023 Forum

10.2514/6.2023-2678

Twin-Delayed Deep Deterministic Policy Gradient for altitude control of a flying-wing aircraft with an uncertain aerodynamic model

Willem Völker*, Yifei Li† and Erik-Jan van Kampen ‡
Faculty of Aerospace Engineering, Delft University of Technology, Delft, 2629HS, The Netherlands

Recent research on the Flying V - a flying-wing long-range passenger aircraft - shows that its airframe design is 25% more aerodynamically efficient than a conventional tube-and-wing airframe. The Flying V is therefore a promising contribution towards reduction in climate impact of long-haul flights. However, some design aspects of the Flying V still remain to be investigated, one of which is automatic flight control. Due to the unconventional airframe shape of the Flying V, aerodynamic modelling cannot rely on validated aerodynamic-modelling tools and the accuracy of the aerodynamic model is uncertain. Therefore, this contribution investigates how an automatic flight controller that is robust to aerodynamic-model uncertainty can be developed, by utilising Twin-Delayed Deep Deterministic Policy Gradient (TD3) - a recent deep-reinforcement-learning algorithm. The results show that an offline-trained single-loop altitude controller that is fully based on TD3 can track a given altitude-reference signal and is robust to aerodynamic-model uncertainty of more than 25%.

I. Introduction

Aviation is responsible for 3.5% of the human impact on climate change - measured in amount of radiative forcing - or even 5% if the effect of cirrus cloud enhancement is taken into account [1]. As global flight traffic is projected to increase with 4.3% annually [2], while average fuel burn of new commercial jet aircraft decreases by only 1% annually [3], novel solutions are needed.

Airframe designers conventionally decrease fuel burn of next-generation aircraft by increasing aerodynamic efficiency and decreasing structural weight through optimisation of current-generation airframe designs. Aerodynamic-efficiency gains (measured by the increase in lift-to-drag ratio) seem to have reached an asymptote though [4], which may mean that the traditional tube-and-wing airframe is approaching its limits [5]. Therefore, a rethink of the tube-and-wing airframe is needed.

The Flying V is a novel aircraft concept that may enable a step change in aerodynamic efficiency. The V-shaped aircraft concept proposed by Benad [6] of the Berlin University of Technology and Airbus in 2015 promises a lift-to-drag ratio 25% higher than conventional wide-body commercial passenger aircraft [7]. Like other flying wing designs the Flying V realises aerodynamic efficiency gains mainly by having a smaller wetted area for a given payload than tube-and-wing aircraft, as for flying wings providing lift, trimming the aircraft and accommodating the payload are all performed by the same integrated component. Moreover, the Flying V promises to have a lower structural weight than comparable tube-and-wing aircraft [6].

Despite several promising studies on flying wings over the past century, none have made it to market as commercial passenger aircraft. Doubts on the stability and control of flying wings are an important factor hindering acceptance [8]. To ensure stable, controlled and thereby safe flight an automatic flight control system (AFCS) may be a solution. However, the design of an AFCS is difficult for novel aircraft, as no off-the-shelf modelling and design tools are available and thereby accurate simulation models are hard to obtain and control design cannot rely on standard methods and software tools. Furthermore, the flight dynamics of novel aircraft often involve non-linearities, which are particularly hard to model and control. This means that conventional control design - which is usually based on linear control theory - may not provide an adequate solution, as it requires an accurate simulation model for initial gain tuning and its ability to control a plant with non-linear dynamics is limited.

---

*MSc Student, Control and Simulation Research Group, Delft University of Technology, willem.volker@gmail.com

†PhD Candidate, Control and Simulation Research Group, Delft University of Technology, y.li-34@tudelft.nl

‡Assistant Professor, Control and Simulation Research Group, Delft University of Technology, e.vankampen@tudelft.nl, AIAA Member

This research proposes the application of reinforcement learning (RL) - a bio-inspired method based on the principle of learning through interaction with a potentially unknown environment - to automatic flight control of the Flying V. RL methods have several attractive properties that make them promising for AFCS design, especially for novel aircraft. Firstly and most importantly, RL methods are well suited to systems for which no (accurate) model is available. An RL *agent* may learn without any prior knowledge of the plant dynamics, thereby allowing online learning, on the real system. Alternatively an RL agent may learn offline with a simulation model and subsequently generalise the learned behaviour to the real world. If learning takes place in simulation a function approximator such as a deep artificial neural network that is robust to model uncertainties may be integrated in the RL method. And secondly, an RL agent with function approximation may find a solution to a problem that involves non-linear and complex plant dynamics, even if the designer does not know the general shape of a solution a priori.

Fully online learning of more complex control tasks is a subject of ongoing research and recent experiments such as the work by Lee and Kampen [9] show promising results. However, the failure rates reported in state-of-the-art literature on online RL such as [9] indicate that online RL methods are currently not mature enough to meet regulatory requirements, if they were to be used to develop a baseline controller for commercial passenger aircraft. Moreover, validation of a flight controller that learns online is hard. Predicting the way the controller will adapt to unforeseen circumstances is after all inherently difficult.

Occasional failures and unclear validation methods of an AFCS for the Flying V would not contribute to the acceptance of flying wings, which is one of the aims of this research. Therefore, this research will use an offline-learning approach. Offline learning has as its main advantages over online learning that more complex control tasks can be learned and the danger of trying unsafe control signals during the learning phase is not present. As offline learning requires no restrictions on the safety of control signals *tried* during learning and as offline learning allows for a larger amount of samples to be collected than online learning, a wider variety of RL methods is available. A particularly promising family of RL methods that may not be sample-efficient enough for online learning for flight control, but do show state-of-the-art performance on a wide variety of complex problems (such as the game of Go [10]) is the family of deep reinforcement learning (DRL) algorithms.

The amount of literature on DRL for fixed-wing flight control is limited, but the available literature shows promising results. An example of a state-of-the-art DRL algorithm applied to flight control is [11], in which a controller based on proximal policy optimisation (PPO) outperforms a PID-controller. The flight controller is applied to a small, unmanned tailless aircraft. More recently [12] showed that with soft actor-critic (SAC) a successful AFCS can be designed for coupled manoeuvres, such as a 40-degree-bank climbing turn.

An important downside of offline learning is that the adaptive nature of RL is not exploited. Whereas an RL controller that continually learns online (or switches on when a failure is detected) can adapt its behaviour when a control failure occurs or a discrepancy between the simulation model and reality exists, an offline RL controller has to rely purely on behaviour learned in simulation. However, Dally and van Kampen [12] showed that with SAC a robust controller can be designed, which can generalise behaviour learned in simulation to, for example, failure cases and biased, noisy sensors that it has not experienced in simulation.

The contributions of this paper are the following. Firstly, this paper presents an assessment of the robustness of a flight controller based on TD3 to uncertainties in the aerodynamic model of the Flying V. Secondly, this paper provides an indication of the usefulness of TD3 for flight control of an aircraft for which the aerodynamic model is uncertain. Finally, this paper presents the first application of RL to the Flying V. Therefore, the methods presented in this paper may serve as a starting point for further research into RL for flight control of the Flying V.

To investigate whether an AFCS based on TD3 is indeed robust to aerodynamic model uncertainties an altitude controller was developed. Hereby control was fully based on TD3, without inner-loop PID control. Moreover, the controller was structured as a single loop, in order to investigate the ability of TD3 to learn the nonlinear and coupled dynamics associated with altitude control. An additional reason for using a single control loop was to exploit the ability of an RL agent to autonomously learn a task, with minimal input based on domain knowledge from the human control engineer. To the best knowledge of this author this research represents the first application of a model-free DRL algorithm for single-loop altitude control of a passenger aircraft. The offline-trained altitude controller was tested on a flight-simulation model of the Flying V that simulated varying levels of aerodynamic-model uncertainty. The effect of model uncertainty on the altitude-tracking error was then used to evaluate the robustness of an AFCS based on TD3 to aerodynamic-model uncertainty.

The paper is structured as follows. Section II describes the algorithm behind TD3, the modelling and simulation methodology used to simulate the Flying V, and the methodology used to train a TD3 agent offline for flight control of the Flying V. Section III shows the responses and errors corresponding to simulations of the trained TD3 agent for the nominal Flying-V model, as well as the model with simulated aerodynamic-model uncertainty, sensor noise and altered initial conditions. Finally section IV presents the conclusion of the present research.

II. Methodology

This section introduces the methodology used to produce the results in this paper. Firstly, subsection II.A introduces the reinforcement-learning algorithm, TD3, used to develop the altitude controller. Secondly, subsection II.B introduces the controlled system: the Flying V and its flight-simulation model. Thirdly, subsection II.C describes how altitude control for the Flying V was formulated as a reinforcement-learning problem. Fourthly, subsection II.D presents the settings of the hyperparameters and other parameters used to train a TD3 agent for altitude control. Lastly, subsection II.E describes how aerodynamic-model uncertainty, sensor noise, and altered initial conditions were modelled.

A. Twin-Delayed Deep Deterministic Policy Gradient

Twin-Delayed Deep Deterministic Policy Gradient or TD3 is a state-of-the-art model-free DRL algorithm for continuous action spaces introduced by Fujimoto et al. [13] in 2018. As TD3 is based on predecessor algorithm Deep Deterministic Policy Gradient (DDPG), published by Lillicrap et al. [14] in 2016, this section starts with an introduction to the fundamentals of DDPG.

While conventional policy-gradient methods use a stochastic policy to ensure sufficient exploration, Deep Deterministic Policy Gradient (DDPG) improves on these methods by using a deterministic policy $\mu(s, a, \boldsymbol{\theta})$, with s the state, a the action, and θ the parameter vector. DDPG is similar to earlier reinforcement-learning algorithms Q-learning and DQN in the way it approximates the optimal action-value function, through an implementation of the Bellman optimality equation. The parameters that define the action-value function are approximated by networks known as Q-Networks.

Also similarly to DQN, DDPG uses experience replay. A set of previous experiences at each time step $e_t = (s_t, a_t, r_t, s_{t+1})$, with r the reward, is stored in a *replay buffer D*. DDPG applies Q-learning updates to samples of experience (s, a, r, s') which are randomly chosen from the replay buffer.

Furthermore, DDPG borrows the idea of using target networks from DQN. As the target y_{DQN}, given by Eq. (1) (with γ the discount factor), depends on the same parameters $\mathbf{w}$ which are being updated, learning can become unstable. With the use of a separate parameter vector $\mathbf{w}^-$ (a time-delayed version of $\mathbf{w}$) to construct a target network $\hat{Q}$, stability is improved. $\hat{Q}$ is constructed by simply cloning Q every C steps. DDPG updates and averages the target networks once every main network update, instead of every C steps as DQN does.

$$y_{DQN} = r + \gamma \max_{a'} Q(s', a', \mathbf{w}) \tag{1}$$

DDPG adds an additional technique to DQN to enable maximisation over continuous action spaces, which would be too expensive with a normal optimisation algorithm. The optimal action-value function is assumed to be differentiable with respect to its action arguments, such that its gradient can be computed and maximisation can be approximated. For this DDPG uses a *target policy network* μ_{target}, which is constructed in the same way as the target Q-network. The target policy network approximates the maximising action for the target action-value function. The resulting loss function is mimised by stochastic gradient descent.

DDPG can be characterised by its target formulation. The target of DDPG can be represented by

$$y_{DDPG} = r + \gamma Q_{target}\left(s', \mu_{target}\left(s', \boldsymbol{\theta}\right), \mathbf{w}\right)^2 . \tag{2}$$

The policy learning step in DDPG is performed by applying gradient ascent with respect to the policy parameters to solve

$$\max_{\theta} \underset{s\sim D}{\mathrm{E}} \left[Q\left(s, \mu(s, \boldsymbol{\theta})\right)\right] . \tag{3}$$

TD3 aims to improve on DDPG and other actor-critic methods by addressing function approximation errors that cause overestimation of action values and sub-optimal policies. Fujimoto et al. [13] brought about this improvement by adapting DDPG with three techniques, the first of which is *double learning*, introduced by van Hasselt [15, 16]. Double learning works by first producing two independent estimates (i.e., from different samples) of a value and using one estimate to choose the maximising action, while using the other to estimate its value. The value estimate will then be unbiased. Secondly the process is repeated with the role of the two estimates reversed, resulting in a second unbiased estimate. TD3 applies double learning by constructing the target in its loss functions with the smallest of the two action values it learns and names this *clipped double Q-learning*. Thereby underestimation is favoured, which is unlikely to persist during learning, as the policy will not favour actions with low values.

The term *delayed* refers to the second technique, namely that of *delaying policy improvement* until policy evaluation converges. TD3 improves its policy once for every two policy evaluation steps. Delaying the policy together with the use of target networks should reduce variance, a possible cause of overestimation bias.

The third technique is *target policy smoothing*. The smoothing is performed to avoid that incorrectly highly valued estimates are exploited by the agent. By adding noise to the target policy and averaging over mini-batches, variance in the target (caused by function approximation errors) can be smoothed. The reasoning behind this technique is that similar actions should have similar values and outliers are therefore probably incorrect.

Target policy smoothing is applied by injecting clipped noise to the target policy and then clipping the action that results from the target policy with added noise. The target action is then:

$$a'\left(s'\right) = \text{clip}\left(\mu_{\text{target}}\left(s', \boldsymbol{\theta}\right) + \text{clip}(\epsilon, -c, c), a_{\text{Low}}, a_{\text{High}}\right), \quad \epsilon \sim \mathcal{N}(0, \sigma), \tag{4}$$

in which c is the maximum noise value, and ϵ the probability of taking a random action. The action is subsequently clipped to ensure it lies within the valid action range $a_{Low} \leq a \leq a_{High}$.

Clipped double Q-learning is performed by constructing the target y_{TD3} for both action-value functions as

$$y_{TD3} = r + \gamma \min_{i=1,2} Q_{i,\text{target}}\left(s', a'\left(s'\right), \mathbf{w}\right), \tag{5}$$

and then choosing the smallest target value of the two, to which both action-value functions are regressed.

Then, like in DDPG, Eq. (3) is used for policy improvement, although less frequently than in DDPG, as prescribed by the delay technique. For this DDPG simply uses the first of the two action-value function approximations it has learned. Also similar to DDPG, noise in the behaviour policy is added to ensure exploration.

With the improvements to DDPG, which is known to be unstable [17, 18] but already often outperformed competitor algorithms on benchmark tasks when it was introduced, Fujimoto et al. [13] produced a relatively stable state-of-the-art RL algorithm. Benchmarking research by Lazaridis et al. [19], Ball and Roberts [20] shows that comparison of RL algorithms is not straightforward, as performance is highly task-dependent, but it is clear that SAC [21] shows comparable performance to TD3 on benchmark tasks. However, research published by Dong et al. [22] suggests that the stochastic nature of SAC (randomness is maximised during training) might lead to learning more oscillatory behaviour, compared to TD3. Whilst the oscillations might not increase tracking error, they would likely increase actuator wear and decrease passenger comfort in flight. Therefore, to decrease the chance that the RL agent learns a policy that is more oscillatory than desirable for flight control, TD3 was chosen for the present research.

B. Flying-V Model and Simulation

The Flying V is a V-shaped flying wing with a pressurised cabin that extends through both wings, as illustrated in Fig. 1. The cabins are located in the wing's leading edge and the engines at the trailing edge, on top of the wing to reduce noise propagation to the ground. The Flying V design aims to outperform state-of-the-art commercial passenger aircraft designs similar to that of the Airbus A350-900, mainly in terms of lift-to-drag ratio and structural weight. The aircraft has a capacity of 314 passengers in a two-class configuration, a cruise speed of M=0.85, a range of 15,000 km and a wing span of 64.75m, all similar to those of the Airbus A350-900.

The flight control system of the Flying V consists of a pair of inboard elevons δ_{e_i}, a pair of outboard elevons δ_{e_o}, a pair of rudders δ_r and two engines. The present research will use a conventional approach to control allocation and thereby allocate the inboard elevons to pitch control (through symmetrical deflection), the outboard elevons to roll control (through asymmetrical deflection) and the rudders to yaw control (through deflection in the same direction around the aircraft body's vertical axis).

The flight simulation model used for this research was based on stability-and-control derivatives obtained from the vortex-lattice method applied to a numerical model of the Flying V [25]. From the stability-and-control derivatives the force and moment coefficients C_X, C_Y, C_Z, C_L, C_M, and C_N were computed according to Eq. (6), in which X is interchangeable with a force along, or moment around, a different axis. Hereby X, Y and Z are the forces along, and L, M and N are the moments around the body x-, y- and z-axes.

$$\begin{aligned} C_X = C_X(\alpha) + C_X(\alpha,\beta) + C_X(\alpha,p) + C_X(\alpha,q) + C_X(\alpha,r) + C_X(\alpha,\delta_{r_{left}}) + C_X(\alpha,\delta_{e_{o,left}}) + \\ C_X(\alpha,\delta_{e_{i,left}}) + C_X(\alpha,\delta_{r_{right}}) + C_X(\alpha,\delta_{e_{o,right}}) + C_X(\alpha,\delta_{e_{i,right}}) \end{aligned} \tag{6}$$

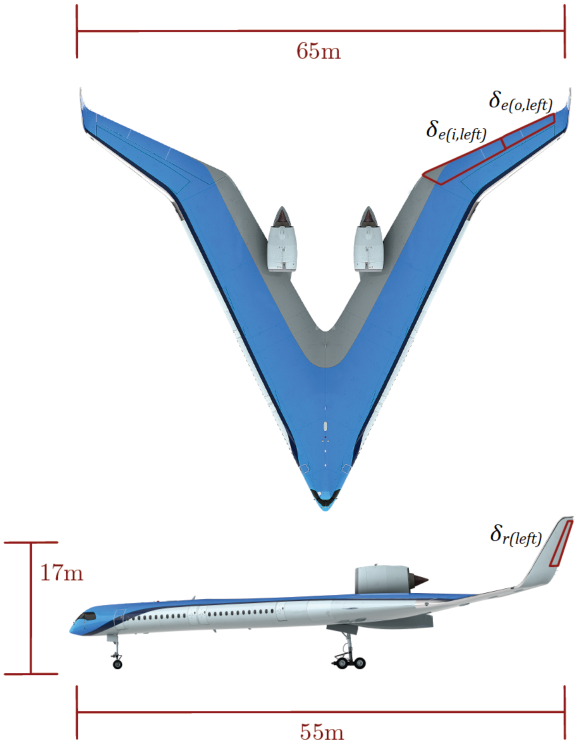


Fig. 1 Illustration of the control-surface layout and outer dimensions of the Flying V [23, 24].

The computed force and moment coefficients, the current state and control input and the geometrical properties of the Flying V were subsequently used as inputs for the equations of motion, from which the 12 states in Table 1 could be obtained, describing the position, velocity, Euler angles and angular rates of the Flying V. The control input consisted of the deflection of the three pairs of control surfaces and the thrust force, as summarised in Table 2, whereby rudder deflection is positive to the left, and elevon deflection is positive down.

Table 1 States in the flight-simulation model of the Flying V

State	Definition	Unit
X_e	position along earth x-axis	m
Y_e	position along earth y-axis	m
Z_e	position along earth z-axis	m
U_b	speed along body x-axis	m/s
V_b	speed along body y-axis	m/s
W_b	speed along body z-axis	m/s
p	rotational rate around body x-axis	rad/s
q	rotational rate around body y-axis	rad/s
r	rotational rate around body z-axis	rad/s
ϕ	rotational angle around body x-axis	rad
θ	rotational angle around body y-axis	rad
ψ	rotational angle around body z-axis	rad
α	angle of body x-axis w.r.t. airflow vector (around body y-axis)	rad
β	angle of body x-axis w.r.t. airflow vector (around body z-axis)	rad

Table 2 Control inputs to the flight-simulation model of the Flying V

Control input	Definition	Unit
$\delta_{r_{left}}$	left rudder angle	deg
$\delta_{e_{o,left}}$	left outboard-elevon angle	deg
$\delta_{e_{i,left}}$	left inboard-elevon angle	deg
$\delta_{e_{i,right}}$	right inboard-elevon angle	deg
$\delta_{e_{o,right}}$	right outboard-elevon angle	deg
$\delta_{r_{right}}$	right rudder angle	deg
T_1	left-engine thrust force	N
T_2	right-engine thrust force	N

For this research actuator dynamics were modelled as a first-order system with a time constant of 0.05 s, which approximates the dynamics of a fast elevator actuator. The actuator time constant and rate limit values were based on examples from [26]. Control surface deflections were limited at angles of ± 30 deg, as proposed by Cappuyns [25].

C. Altitude Control as a Reinforcement-Learning Problem

The task of the TD3 agent in this research was to track a given altitude reference signal h_{ref}. The reward was defined as Eq. (7), meaning that the agent measures performance through the absolute value of the altitude error $h_{error} = h - h_{ref}$, where h is the measured altitude and h_{ref} the given reference altitude. The reward space was limited by clamping the altitude error value if it was larger than 50 m.

$$r = -|h_{error}|, \quad h_{error} \leq 50 \text{ m} \tag{7}$$

To learn and subsequently perform an altitude reference tracking task the agent could manipulate the Flying V's pitch rate by deflecting the inboard elevons symmetrically. The agent effectively controlled the elevon-deflection *change* $\Delta\delta_{e_i}$, which was restricted to the interval $(-0.6, 0.6)$ deg. The restriction of $\Delta\delta_{e_i}$ effectively imposed a rate limit of 60 deg/s, as the sampling rate was set to 100 Hz. Control of $\Delta\delta_{e_i}$ by the agent was implemented through Eq. (9) (inspired by the work of Dally and van Kampen [12]) in which $\Delta\delta_{e_i min} = -0.6$ and $\Delta\delta_{e_i max} = 0.6$.

$$\delta_{e_i t} = \delta_{e_i t-1} + \Delta\delta_{e_i t}, \quad -0.6 \text{ deg} \leq \Delta\delta_{e_i} \leq 0.6 \text{ deg} \tag{8}$$

$$\Delta\delta_{e_i} = \Delta\delta_{e_i min} + (a+1)\frac{\Delta\delta_{e_i max} - \Delta\delta_{e_i min}}{2}, \quad a \in (0, 1) \tag{9}$$

To learn a relation between aircraft states, the agent's actions and performance with respect to the reward, the agent was given four observations. The first observation available to the agent was the altitude error h_{error}, necessary to distinguish between positive and negative altitude errors, as the reward was defined in terms of the absolute value of h_{error}. The second observation was the measured pitch angle θ, which directly influences change in altitude. The third observation was the pitch rate q, which in turn directly influences pitch angle. Pitch rate is itself directly influenced by the inboard elevon deflection angle δ_{e_i}. The fourth observation was the inboard elevon angle δ_{e_i}, which was added because action a was formulated in terms of a change with respect to an otherwise-unknown current deflection angle. All observations were normalised through before being given to the agent, with the ranges given in Table 3. The observation s is summarised in Eq. (10). The agent-environment interaction resulting from the reward, action and observation specified in this section is summarised by the block diagram shown in Fig.2.

$$s = [h_{error}, \theta, q, \delta_{e_i}]^T \tag{10}$$

D. Training

Training of the TD3 agent was performed in episodes of 20 seconds each, with a sampling rate of 100 Hz. The reference signal during training was given by Eq. (11), where h_0 is the initial altitude and g_{climb} is the climb gradient.

Table 3 Ranges for normalisation of the observation

Observation	Range	Unit
h_{error}	(-300,300)	m
θ	(-20,20)	deg
q	(-20,20)	deg
δ_{e_i}	(-30,30)	deg

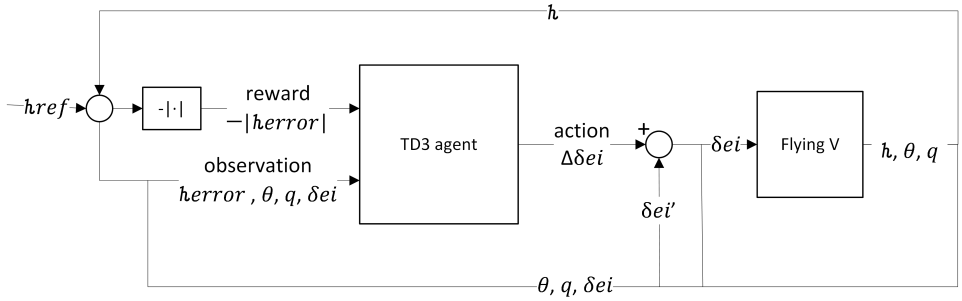


Fig. 2 Agent-environment interaction used to train the TD3 agent for altitude control.

g_{climb} was randomly initialised at the start of each episode and was drawn from a uniform distribution in the interval $[-3000, 3000]$ ft/min. Thereby the climb gradients during training approximately corresponded to the climb gradients prescribed for an Airbus A350-900 [27], the reference aircraft for the Flying V.

$$h_{ref} = \begin{cases} h_0, & \text{if } t < 2 \\ h_0 + g_{climb}(t-2), & \text{otherwise} \end{cases} \tag{11}$$

For this research the Flying V was simulated in the cruise condition, with a neutral centre of gravity location and a weight equal to the maximum takeoff weight. The Flying V was initialised with initial conditions at or near the trimmed state at 10 km at the start of each episode, resulting in the initial conditions summarised in Table 4, whereby random initialisation was based on a uniform distribution. The outboard elevons and rudders were kept constant at zero degrees. The thrust force was kept constant at the trim value for this flight condition, equal to 17479 N per engine. All states that are not included in Table 4 were initialised as zero.

Table 4 Initial conditions

State	Value or range	Unit
M	0.85	-
h_0	10	km
U_b	253.9 ± 1	m/s
W_b	18.2 ± 0.5	m/s
θ	4.1 ± 1	deg
q	0 ± 1	deg/s

Hyperparameters were tuned manually through trial and error. Most hyperparameter values were copied from Fujimoto et al. [13], but some hyperparameters required tuning. Especially the exploration noise required a relatively large increase compared to the value originally used by Fujimoto et al.. The same type of artificial neural network was used to represent the actor and the critic. The networks' first layers consisted of 4 input neurons (one for each observation), followed by two fully connected hidden layers with 128 neurons each. The hidden layers were followed by a ReLu layer, a fully connected output layer, and finally a tanh function. Hyperparameter settings are summarised in Table 5.

Table 5 Hyperparameter settings (adapted from [13])

Parameter	Value
hidden units per layer	128
hidden layers	2
exploration noise	0.4
smoothing noise	0.2
smoothing noise limits	± 0.5
learning rate	1e-3
discount rate	0.99
batch size	100
experience buffer length	1e6
target smoothing factor	0.005
policy update frequency	2
target update frequency	2
optimiser	Adam

The episode-reward curve for training a TD3 agent for altitude control typically did not converge to a steady episode-reward value. Therefore, the learned policy was saved every time the average episode reward reached a value of at least -20,000. Hereby the episode reward was averaged over the last 5 episodes. After a successful training run all saved policies were compared, based on their mean-absolute error when following a predefined test-reference signal. Hereby the test-reference signal included climb and descent sections as well as horizontal sections. The climb and descent gradients were equal to ± 1500 ft/min, comparable to the standard gradients of the Airbus A350-900 at high altitude [27]). The policy that resulted in the lowest mean-absolute-reference-tracking error for the test-reference signal was used to produce the results presented hereafter.

E. Simulation of Altered Conditions for Robustness Testing

1. Aerodynamic-Model Uncertainty

To simulate the discrepancy that may exist between the aerodynamic model of the Flying V and the real-world Flying V, uncertainty factors were applied to the stability-and-control derivatives on which the flight simulation model for this research was based. Each of the 11 stability-and-control derivatives, as named in Eq. (6), was multiplied by an uncertainty factor ν_i, whereby the left and right surfaces of a control surface pair were given the same factor. This process resulted in an altered aerodynamic coefficient $\hat{C}_X(\dots,\dots)$, as summarised in Eq. (12), where X can be substituted for a different force or moment coefficient.

$$\hat{C}_X(\dots,\dots) = C_X(\dots,\dots)\cdot\nu_i, \quad \nu_i \sim \mathcal{N}(1,\sigma) \quad \text{and} \quad \sigma \in [0.0625, 0.125, 0.25] \tag{12}$$

Hereby the means of the stability-and-control derivatives of the Flying V were assumed to be at the values obtained from the vortex-lattice method applied to a numerical model of the Flying V. As reported by van Overeem and van Kampen [28] the maximum error in the aerodynamic model of the Flying V used for the present research is estimated to be 25%. To simulate an uncertainty with a maximum of 25% the uncertainty factors were drawn from a normal distribution with a standard deviation of 0.125, such that 95% of the uncertainty-factor samples would fall in the aerodynamic-error range $[-25\%, 25\%]$. After the main set of simulation runs with a standard deviation of 0.125, two other sets of simulations were run, of which one had a standard deviation of 0.0625, to simulate the scenario that the actual maximum aerodynamic-model uncertainty is less than 25%, and the other had a standard deviation of 0.25, to simulate the scenario that the actual maximum aerodynamic-model uncertainty is more than 25%. For each of the three standard deviations a set of 100 simulations of 100 s each was run. During each simulation of 100 s the altitude-reference signal remained horizontal for $0 \leq t < 10$ s, descended at a rate of 1500 ft/min for $10 \leq t < 30$ s, was again horizontal for $30 \leq t < 50$ s, climbed at a rate of 1500 ft/min for $50 \leq t < 70$ s and was horizontal for

$70 \leq t < 100$ s. This reference signal will hereafter be used as the standard reference signal for testing the altitude controller developed for this research and will be referred to as reference signal *RS*.

2. Sensor Noise

As the Flying V is still under development it is hard to precisely estimate the noise that will be present in the real-world aircraft. However, by using measurements of the sensor noise in other aircraft as a reference, a first assessment of the robustness of the altitude controller developed in this research can be made. To assess the robustness of the altitude controller, sensor noise was not simulated during training, but only added to the simulated Flying V during tests for robustness to sensor noise.

Research by Grondman et al. [29] into sensor noise in the Cessna Citation PH-LAB aircraft was used as a reference for simulating sensor noise in the present research, because of the well-documented noise bias and standard deviation values for several types of sensors. Noise was simulated for the present research by adding a noise sample at each time step, from normal distributions with the biases and standard deviations shown in Table 6.

Table 6 Sensor noise [29]

Sensor	Bias	Standard deviation	Unit
altitude	8.0e−3	6.7e−2	m
pitch angle	4.0e−3	3.2e−5	rad
pitch rate	3.0e−5	6.3e−4	rad/s

3. Altered Initial Conditions

Altered initial conditions were simulated by randomly initialising states at the start of each of 100 simulation runs of 100 s in duration each. Hereby the trimmed initial condition I_{trim} was altered through Eq. (13), resulting in altered initial condition I_{alt}. ξ is a scaling factor drawn from a uniform distribution.

$$I_{alt} = I_{trim} + \xi \cdot I_{max}, \quad \xi \sim \mathcal{U}(-1, 1) \tag{13}$$

III. Results and Discussion

This section presents results obtained from simulated tests with the altitude controller developed for this research, accompanied with a discussion of these results. Firstly, subsection III.A shows and discusses the altitude-tracking performance of the controller under nominal conditions. Secondly, subsection III.B shows and discusses the robustness of the controller to aerodynamic-model error. Thirdly, subsection III.C shows and discusses the robustness of the controller to sensor noise, in combination with alternative reference-signal shapes. Fourthly, subsection III.D shows and discusses the robustness of the controller to altered initial conditions. Lastly, subsection III.E shows and discusses the stability during training and the sampling efficiency of TD3, as observed in this research.

A. Altitude-Tracking Performance under Nominal Conditions

In [30] the Federal Aviation Authority (FAA) specifies that for aircraft to be authorised to fly in Reduced Vertical Separation Minimum airspace (between 8.8 and 12.5 km altitude), an automatic altitude control system should be

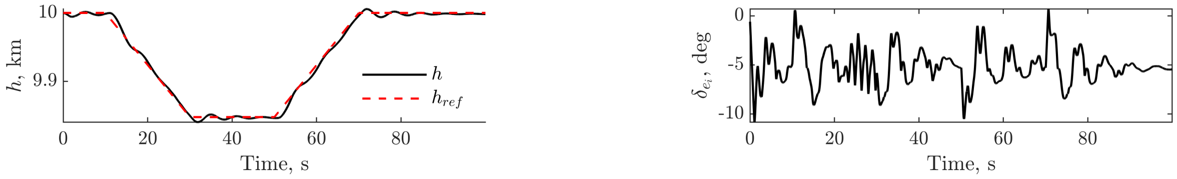
> ". . . capable of controlling aircraft height within a tolerance band of ±65 ft (±20 m) about the acquired altitude when the aircraft is operated in straight and level flight under nonturbulent, nongust conditions.".

Therefore, a maximum-absolute-altitude-tracking error of 20 m, under nominal flight conditions, is used to determine whether the altitude controller in this research is successful. Figure 3 shows the simulated response of an offline-trained TD3 agent to a given altitude-reference signal with climb and descent gradients of 1500 ft/min, whereby the Flying V started from trimmed initial conditions and thrust was kept constant during the whole manoeuvre. Figure 3a shows that the agent is able to track the altitude-reference signal in both horizontal, descent, and climb phases. The mean-absolute-altitude-tracking error for the reference signal shown in Fig. 3a was 3.0 m, and the maximum-absolute-altitude-tracking error was 11.6 m. Therefore, even as the reference signal included climb and descent phases, rather than level flight as specified by the FAA [30], the test shown in Fig. 3 indicates that the altitude controller is successful under nominal conditions.

The measured altitude h in Fig. 3a oscillated around the reference altitude h_{ref} during the limited simulation time shown. However, the response for $t \geq 70$ s shows that the oscillations died out over time. Oscillations were also visible in the angle of attack (Fig. 3e). As angle of attack is related to load factor, the oscillations may cause passenger discomfort if the policy that was simulated to produce Fig. 3 were applied in the real-world aircraft. Moreover, as Fig. 3b shows, oscillations were also present in the inboard-elevon-deflection angle during the whole manoeuvre, which may increase actuator wear. Passenger comfort and actuator wear were not part of the scope of the current research however. Therefore the reward function (Eq. (7) did not include a penalty for oscillations.

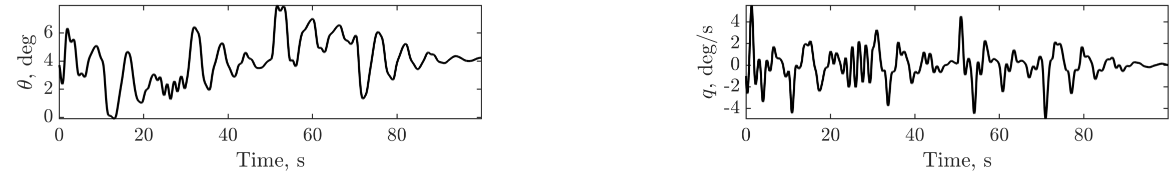
One source of the oscillations is likely the short-period eigenmode of the Flying V. At the flight condition simulated in Fig. 3 the short-period frequency is approximately 0.34 Hz, as determined by simulating an impulse response of the Flying V at 10 km altitude, starting from a trimmed condition, and subsequently measuring the frequency of the short-period oscillation in the pitch rate[*]. The pitch-rate signal shown in Fig. 3d has a spike in the frequency content at a frequency of 0.34 Hz[†], which may be related to the short period. However, a larger spike in the frequency content of the pitch-rate signal shown in Fig. 3d, compared to the spike at 0.34 Hz, is at a frequency of 0.14 Hz, which is not close to the short-period frequency nor the phugoid frequency of 0.0084 Hz. Therefore, the oscillations with a frequency of 0.14 Hz are likely caused mainly by the oscillating inboard-elevon deflections resulting from the policy learned by the TD3 agent.

Oscillations in the policy learned by the agent were already greatly reduced by formulating the action as a change in elevon angle. Earlier attempts to allow the agent to manipulate the elevon-deflection angle directly consistently resulted in a policy with excessively high-gain control. Adding a penalty term for high elevon-deflection rates in the reward function was not found to offer a solution, as the added penalty term caused learning to be unsuccessful. Thereby it is hypothesised that this added penalty term makes the reinforcement-learning problem too complex for the TD3 agent.



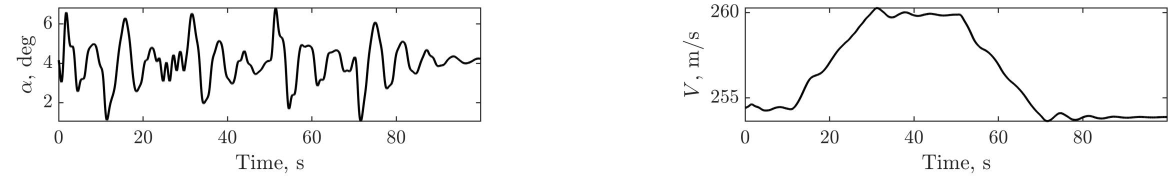
(a) Altitude

(b) Inboard-elevon-deflection angle



(c) Pitch angle

(d) Pitch rate



(e) Angle of attack

(f) True airspeed

Fig. 3 Nominal response of the TD3 altitude-control agent to an altitude-reference signal with climb and descent gradients of 1500 ft/min.

---

[*] The eigenmode frequencies were obtained from the non-linear simulation model. Therefore, the mentioned eigenmode frequencies are an approximation of the values that would have been obtained by first linearising and reducing the simulation model.

[†] Spikes in the frequency content were found from the periodogram of the pitch-rate signal.

B. Robustness to Aerodynamic-Model Error

Figure 4 shows the mean-absolute-altitude-reference-tracking errors for three levels of aerodynamic uncertainty. The middle boxplot in Fig. 4 corresponds to simulations of the previously estimated aerodynamic-model-uncertainty range of the Flying-V model used for this research. The right boxplot shows the scenario that the actual uncertainty range is higher than the previously estimated range. As Fig. 4 shows, the mean, third quartile, and highest outlier increase with aerodynamic uncertainty. The outlier that can be seen in Fig. 4 at a mean-absolute error of 3.6 m, for a standard deviation of $\sigma = 0.25$, represents the worst-case scenario that came forth from the simulations that were run for this research.

As van Overeem and van Kampen [28] adopted a similar methodology for simulating aerodynamic-model uncertainty to the present research, with also a maximum standard deviation of 0.25, the results presented in this paper indicate that the TD3 controller presented here is robust to at least the same level of aerodynamic-model uncertainty as the INDI controller presented by [28]. However, differences between the research of van Overeem and van Kampen and the present research in the flight control task and the flight conditions simulated limits the possibilities for a direct comparison.

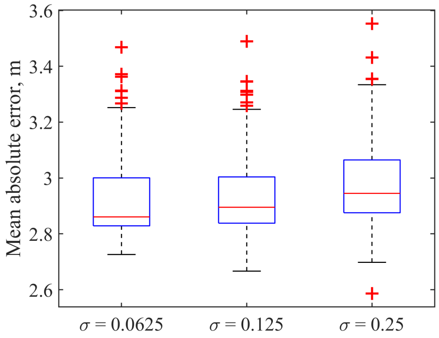


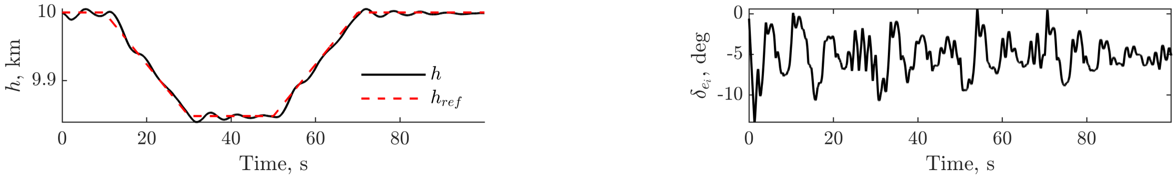
Fig. 4 Altitude-reference-signal-tracking error for three levels of aerodynamic-model uncertainty. The middle boxplot corresponds to a simulation of the uncertainty range estimated in previous research. The left and right boxplots show the scenarios that the uncertainty range is lower or higher than was estimated.

To assess whether flight control is safe in the worst-case scenario in terms of aerodynamic uncertainty, the response corresponding to the simulation with a mean-absolute error of 3.6 m was plotted, shown in Fig.5. The signal shown in Fig. 5a shows that the measured altitude h remains close to the reference altitude h_{ref}. The maximum-absolute-altitude-tracking error was 14.4 m, which is smaller than 20 m and thereby complies with the requirement set in subsection III.A. The transient response dies out and the steady-state response has a small error, as can be most clearly observed at $t > 70$ s in Fig. 5a. Moreover, Fig. 5e shows that the angle of attack does not come near the stall angle of 18 degrees [25], nor the angle of attack at which the Flying V has a pitch-break tendency, equal to 20 degrees [31].

C. Robustness to Sensor Noise and Alternative Reference-Signal Shapes

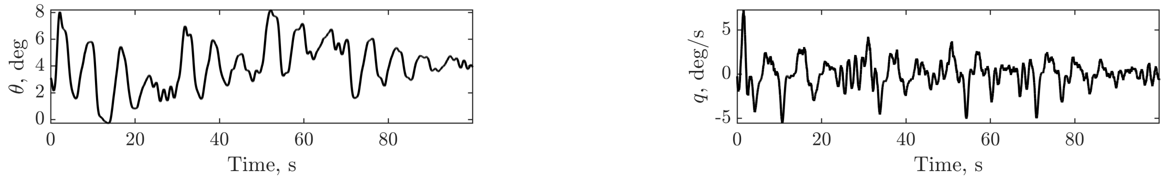
As mentioned in subsection III.A the nominal mean-absolute-reference tracking error was 3.0 m. Figure 6 was simulated for the same reference signal and therefore shows that sensor noise (which the TD3 agent did not encounter during training) results in an increase of approximately 0.02 m in the mean-absolute reference tracking error. The figure also shows that the worst outlier for the 100 simulations that were run for this research was at an increase of 0.06 m in the mean-absolute-reference tracking error. Therefore, these simulations show that the altitude controller developed for the present research is robust to the simulated sensor noise.

To assess how the shape of the altitude-reference signal affects tracking error and safety of the control policy, the altitude controller developed for this research was simulated for two alternative reference-signal shapes, with simulated sensor noise. The same policy as was simulated to produce the results presented in previous sections of this paper was used, so the agent had not encountered these reference signals nor experienced the sensor noise during training. Figure 7 shows the response to a sinusoidal signal with a frequency of 0.01 Hz and an amplitude of 500 m, thereby simulating average climb and descent gradients of approximately 4000 ft/min. Figures 7b-7e show that oscillations in the control signal and the Flying V's attitude subside faster than for the reference signal shown in Fig. 5, which



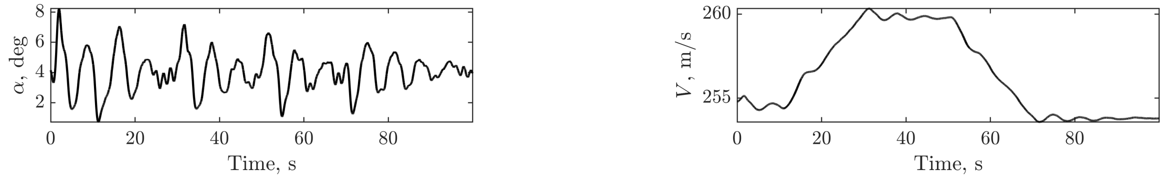
(a) Altitude

(b) Inboard-elevon-deflection angle



(c) Pitch angle

(d) Pitch rate



(e) Angle of attack

(f) True airspeed

Fig. 5 Worst response out of 100 runs in terms of altitude-tracking error for the TD3 altitude-control agent on a model of the Flying V with a simulated maximum aerodynamic-model error of 50%.

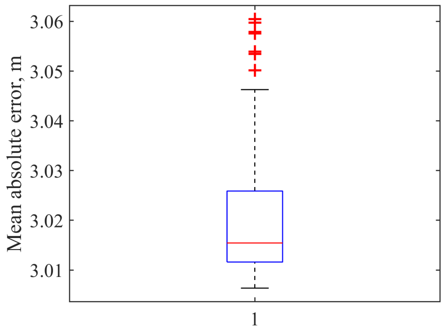
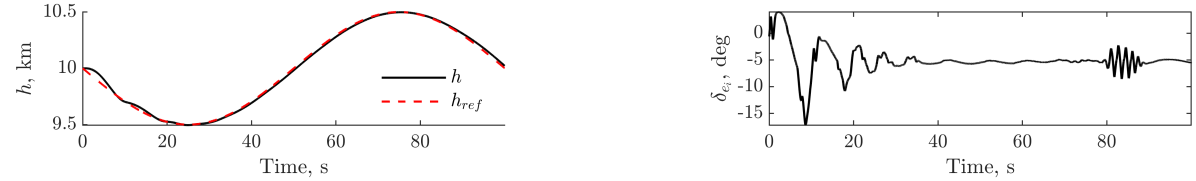


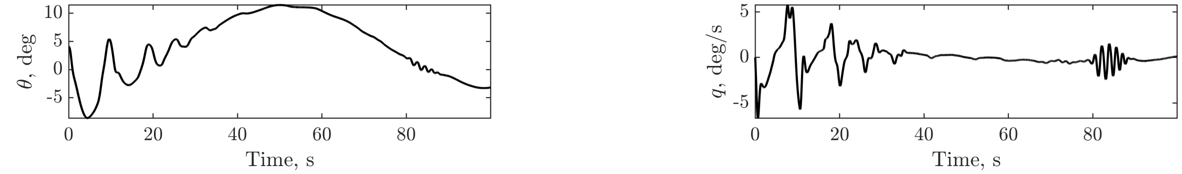
Fig. 6 Robustness to sensor noise and bias measured in terms of the altitude-reference-signal-tracking error. The error for the same reference signal with ideal sensors was 3.0 m.

may be explained by the more gradual change in flight path angle corresponding to the sinusoidal altitude-reference signal. The oscillations that start after around 80 s may be explained by the relatively large decrease in airspeed (larger than for the nominal signal, see Fig. 3f), which causes aerodynamic damping and control effectiveness to decrease. The mean-absolute-reference-tracking error for this sinusoidal signal was 13.3 m. The increase in error with respect to standard reference signal *RS* is reasonable because of the higher gradients of the sinusoidal signal. The maximum-absolute-reference-tracking for the sinusoidal signal was 50 m, but this maximum error occurred at the start of the simulation (after 3.6 s), when a sudden change in climb gradient of -4000 ft/min was commanded. After allowing a settling time of 15 s to adapt to the initial sudden change in climb gradient, the maximum-absolute-reference-tracking was 25 m.

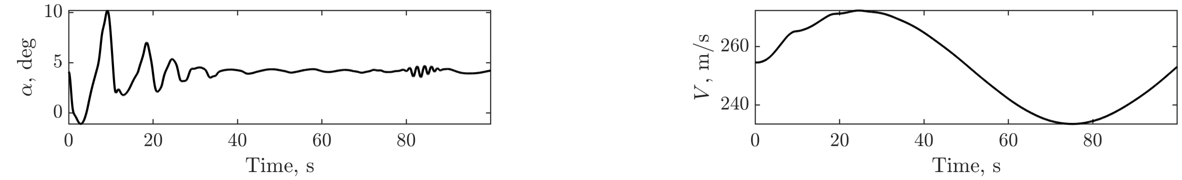
Figure 8 shows the response to a triangular altitude-reference signal with a gradient of 1500 ft/min and simulated sensor noise, for which the mean-absolute-reference-tracking error was 5.9 m and the maximum-absolute-reference-tracking error was 22.2 m. The triangular signal has more extreme changes in climb gradient than a realistic reference signal that would be encountered in the real world, but it provides insight into the safety of the controller in case



(a) Altitude **(b) Inboard-elevon-deflection angle**



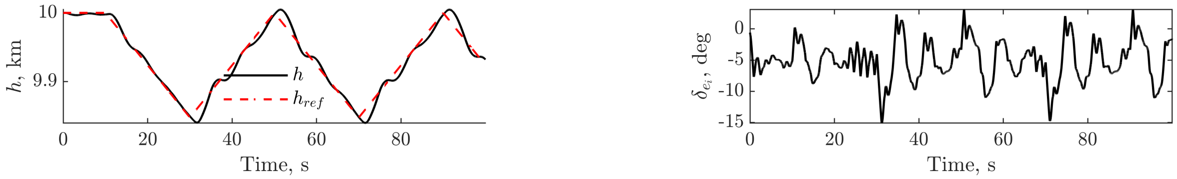
(c) Pitch angle **(d) Pitch rate**



(e) Angle of attack **(f) True airspeed**

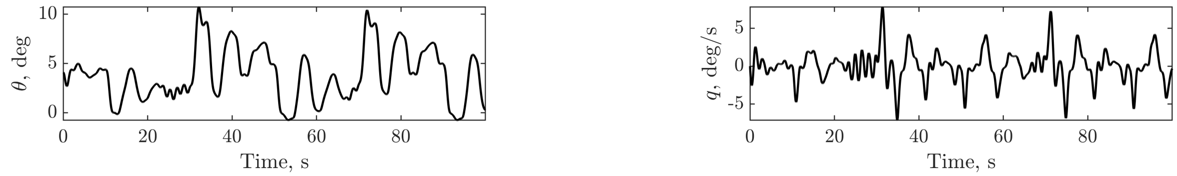
Fig. 7 Robustness to sensor noise and reference-signal shape, shown through the response to a sinusoidal reference signal.

sudden changes in climb gradient are commanded by the pilot. Although oscillations are visible in all signals shown in Fig. 8, the angle of attack shown in Fig. 8e does not come near the stall or pitch-break angle and the airspeed shown in Fig. 8f remains near the nominal value of 254 m/s, so the controller keeps the aircraft in a safe state. Furthermore, as the maximum-absolute-reference-tracking error for a sinusoidal signal exceeds the requirements stated in subsection III.A for level flight (i.e., a horizontal reference signal) by only 2.5%, and the error for a triangular signal exceeds the requirements by only 1%, the controller is considered robust to alternative reference signals with sensor noise.



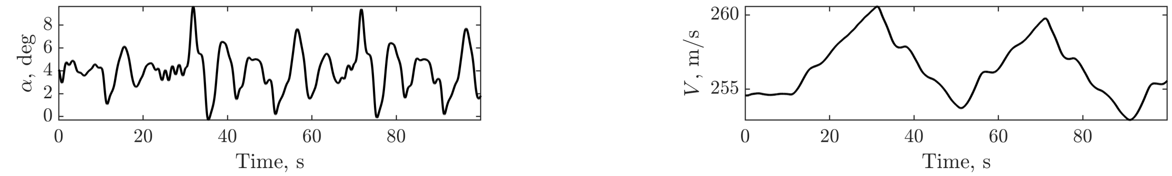
(a) Altitude

(b) Inboard-elevon-deflection angle



(c) Pitch angle

(d) Pitch rate



(e) Angle of attack

(f) True airspeed

Fig. 8 Robustness to sensor noise and reference-signal shape, shown through the response to a triangular reference signal with climb gradients of 1500 ft/min.

Random initialisation of the flight condition during training, as specified in subsection II.D, was found to have a large positive effect on robustness to alternative reference-signal shapes and altered initial conditions (see subsection III.D) during testing. However, randomly initialising the flight condition was found to only work for small ranges near the trimmed condition, as the problem became too hard for the agent to find a successful policy if the initial flight condition differed too much from the trimmed condition. Therefore, it is recommended to experiment with the range in initial conditions implemented during training if a similar future research project is undertaken, to find a range that is not too large to hinder training, nor too small to diminish the effect on robustness of the resulting policy.

D. Robustness to Altered Initial Conditions

In the real world the Flying V may not always be in a trimmed state when the altitude controller is engaged, due to atmospheric disturbances or pilot inputs that temporarily force the Flying V to deviate from the trimmed state. Therefore the robustness of the altitude controller to initial conditions that deviate from the trimmed state was tested, hereafter referred to as *altered initial conditions*. Hereby the states that directly affect longitudinal motion - U_b, W_b, q and θ - and the altitude h were altered. A maximum deviation I_{max} (see Eq. (13)) was chosen for each altered state, as shown in Table 7. Figure 9 shows that for some combinations of altered initial conditions the mean-absolute-reference-tracking error is higher than the nominal error of 3.0 m for the same reference signal. However, the increase in error is limited to a maximum of 1.3 m. Moreover, Fig. 9 shows that, for reference signal *RS*, the mean-absolute-reference-tracking error for some combinations of initial conditions is lower than for the nominal initial conditions.

Table 7 Limits of altered initial conditions

Initial condition	I_{max}	Unit
U_b	10	m/s
W_b	1	m/s
θ	2	deg
q	2	deg/s
h	100	m

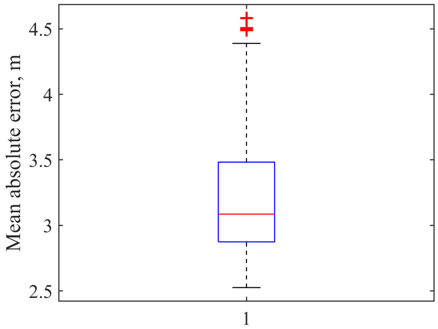
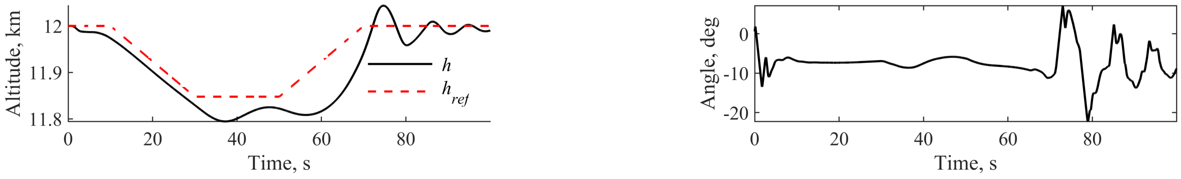
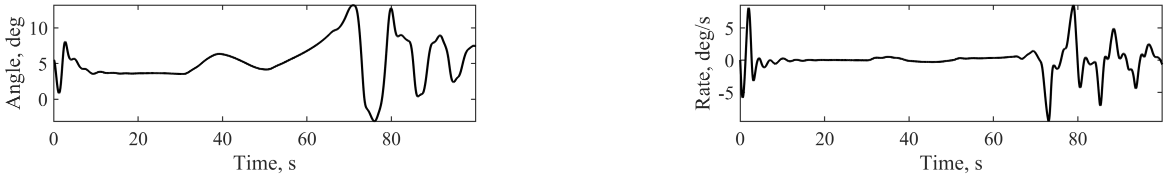


Fig. 9 Robustness to random combinations of initial horizontal and vertical airspeeds, pitch angles, pitch rates and altitudes that differ from the nominal values, measured in terms of the altitude-reference-tracking error in response to reference signal *RS*.

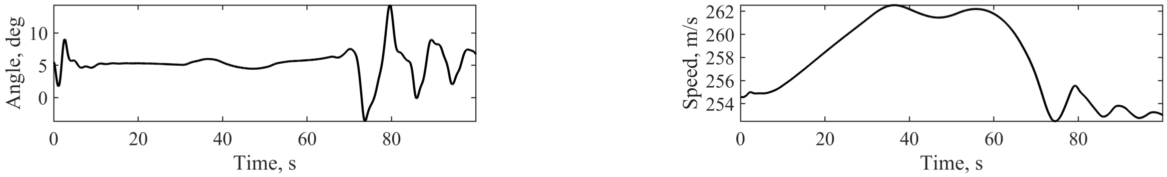
To assess how well the altitude controller can generalise behaviour to initial conditions that differ more extremely from the conditions that the TD3 agent was trained for (namely trimmed cruise flight at 10 km altitude), additional simulations were run for more extremely altered initial altitudes. Simulation starting at 8 km altitude for reference signal *RS* resulted in a mean-absolute-reference-tracking error of 2.8 m, which is 0.3 m smaller than the error at the nominal altitude of 10 km. Simulation at 12 km altitude, on the other hand, resulted in a mean-absolute-reference-tracking error of 27.7 m, which is 24.7 m larger than the error at the nominal altitude of 10 km. The smaller error at 8 km altitude may be explained by the higher air density at lower altitude, resulting in more control effectiveness and more damping of eigenmodes, while the larger error at 12 km altitude may similarly be explained by less control effectiveness and less damping of the eigenmodes. As Fig. 10a shows, the agent is not able to accurately follow the commanded reference signal. However, as Fig. 10e shows, the controller does keep the Flying V in a safe flight regime, as the angle of attack does not come near the stall angle of 18 deg. Moreover, airspeed barely falls to values lower than the nominal airspeed, as Fig. 10f shows.



(a) Altitude **(b) Inboard-elevon-deflection angle**



(c) Pitch angle **(d) Pitch rate**



(e) Angle of attack **(f) True airspeed**

Fig. 10 Response of the altitude controller when initialised at an altitude of 12 km, which is 2 km higher than the altitude it was trained for.

E. Training Stability and Sampling Efficiency of TD3

Figure 11 shows the moving average of the episode reward of the training run that produced the TD3 agent used to produce the results presented in this paper, averaged over 50 episodes of 20 s each. The figure shows that the TD3 agent improved its policy relatively consistently during the first approximately 150 episodes, but did not converge to that policy. Only after approximately 1300 episodes did the policy temporarily converge to a policy that more consistently resulted in relatively high episode rewards. Three aspects visible from the reward curve shown in Fig. 11 are typical for the reward curves of the several flight control problems tested for the research presented in this paper.

Firstly, fast initial improvement in the policy did not immediately lead to finding a successful policy, but was usually followed by divergence to unsuccessful policies. Secondly, once a successful policy was found, the agent eventually diverged from this policy. Thirdly, sample efficiency was low, leading to long training times. Finding the policy used to produce the results for this research (at 1466 episodes) took approximately 18 hours on an Intel Xeon CPU E5-1620 v3 @ 3.50GHz.

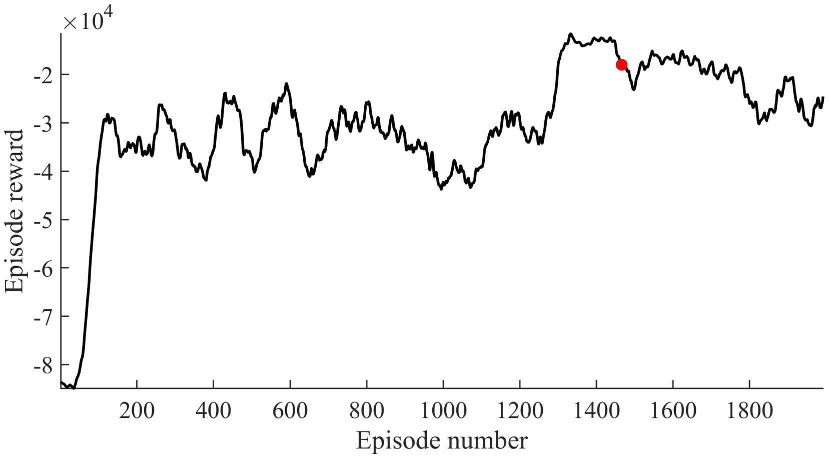


Fig. 11 Average reward during training of a TD3 agent for altitude control of the Flying V. The policy obtained at 1466 episodes (marked in red) was used to produce the results presented in this paper.

IV. Conclusion

The research presented in this paper shows that a single-loop controller based on TD3 can learn altitude control of a non-linear simulation model of the Flying V in an offline setting, and satisfies the set requirement of a maximum-absolute-altitude-tracking error of 20 m. Hereby the controller only observes the altitude-tracking error, the pitch angle, the pitch rate, and the elevon-deflection angle. The results also show that the controller is robust to aerodynamic-model error, sensor noise, various shapes of the altitude-reference signal, and unfavourable initial flight conditions. Therefore, the research presented in this paper suggests that deep-reinforcement learning and in particular TD3 has the potential to be used for creating robust flight controllers. However, several questions remain open to investigation.

To build on this research we recommend to investigate how robust a controller with the structure proposed in this research is to faults and atmospheric disturbances that were not simulated for this research. Furthermore, to increase the applicability of the controller to a wider variety of flight regimes we recommend to investigate the addition of airspeed and height in the observation and training at various altitudes and airspeeds. To prevent the problem from becoming too inconsistent during training and too complex for the TD3 agent to find a successful policy, we suggest the use of a learning curriculum. The learning curriculum may include progressively more difficult initial conditions, atmospheric disturbances and a varying aerodynamic model to increase robustness of the learned policy. Lastly, the use of TD3 or similar RL algorithms for lateral-directional control or a combination of lateral-directional and longitudinal control may be investigated.

By continuing research into reinforcement learning for flight control, with the use of more sample-efficient reinforcement-learning algorithms, the development of new methods to explain control policies, and the creation of standardised practices to develop reinforcement-learning-based flight controllers, researchers may bring reinforcement learning for flight control to a level at which it can be used in industry. In that way, safer, more autonomous flight of passenger aircraft with novel airframe designs such as the Flying V may one day become reality.

References

[1] Lee, D. S., Pitari, G., Grewe, V., Gierens, K., Penner, J. E., Petzold, A., Prather, M. J., Schumann, U., Bais, A., Berntsen, T., Iachetti, D., Lim, L. L., and Sausen, R., "Transport impacts on atmosphere and climate: Aviation," *Atmospheric Environment*, Vol. 44, No. 37, 2010, pp. 4678–4734. https://doi.org/10.1016/J.ATMOSENV.2009.06.005.

[2] Airbus, "Cities, airports & aircraft," Tech. rep., Airbus S.A.S., 2019. URL https://www.airbus.com/aircraft/market/global-market-forecast.html.

[3] Zheng, X. S., and Rutherford, D., "Fuel burn of new commercial jet aircraft: 1960 to 2019," Tech. rep., The International Council on Clean Transportation, 9 2020.

[4] Martinez-Val, R., Palacin, J. F., and Perez, E., "The evolution of jet airliners explained through the range equation," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 222, No. 6, 2008, pp. 915–919. https://doi.org/10.1243/09544100JAERO338.

[5] Martinez-Val, R., “Flying wings. An new paradigm for civil aviation?” *Acta Polytecnica*, Vol. 47, No. 1, 2007. URL http://ctn.cvut.cz/ap/.

[6] Benad, J., “The Flying V - A new Aircraft Configuration for Commercial Passenger Transport,” Tech. rep., Airbus Operations GmbH and Berlin University of Technology, 2015. https://doi.org/10.25967/370094.

[7] Faggiano, F., Vos, R., Baan, M., and Van Dijk, R., “Aerodynamic design of a flying V aircraft,” *17th AIAA Aviation Technology, Integration, and Operations Conference, 2017*, American Institute of Aeronautics and Astronautics Inc, AIAA, 2017. https://doi.org/10.2514/6.2017-3589.

[8] Wood, R. M., and Bauer, S. X., “Flying wings / flying fuselages,” *39th Aerospace Sciences Meeting and Exhibit*, American Institute of Aeronautics and Astronautics Inc., 2001. https://doi.org/10.2514/6.2001-311.

[9] Lee, J., and Kampen, E.-J., “Online reinforcement learning for fixed-wing aircraft longitudinal control,” *AIAA Scitech 2021 Forum*, 2021, pp. 1–20.

[10] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Graepel, T., and Hassabis, D., “Mastering the game of Go without human knowledge,” *Nature*, Vol. 550, No. 7676, 2017, pp. 354–359. https://doi.org/10.1038/nature24270.

[11] Bøhn, E., Coates, E. M., Moe, S., and Johansen, T. A., “Deep Reinforcement Learning Attitude Control of Fixed-Wing UAVs Using Proximal Policy Optimization,” *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2019, pp. 523–533. https://doi.org/10.1109/ICUAS.2019.8798254, URL http://arxiv.org/abs/1911.05478http://dx.doi.org/10.1109/ICUAS.2019.8798254.

[12] Dally, K., and van Kampen, E., “Soft Actor-Critic Deep Reinforcement Learning for Fault-Tolerant Flight Control,” *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, American Institute of Aeronautics and Astronautics Inc, AIAA, 2022. https://doi.org/10.2514/6.2022-2078.

[13] Fujimoto, S., van Hoof, H., and Meger, D., “Addressing Function Approximation Error in Actor-Critic Methods,” *35th International Conference on Machine Learning*, Stockholm, 2018. URL http://arxiv.org/abs/1802.09477.

[14] Lillicrap, T., Hunt, J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D., “Continuous control with deep reinforcement learning,” *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, 2016.

[15] van Hasselt, H., “Double Q-learning,” *Advances in neural information processing systems*, Vol. 23, 2010, pp. 2613–2621.

[16] van Hasselt, H., “van Hasselt, Hado Philip. Insights in reinforcement rearning: formal analysis and empirical evaluation of temporal-difference learning algorithms,” Ph.D. thesis, Utrecht University, 2011.

[17] Duan, Y., Chen, X., Houthooft, R., Schulman, J., and Abbeel, P., “Benchmarking Deep Reinforcement Learning for Continuous Control,” *33rd International Conference on Machine Learning*, New York, 2016. URL http://arxiv.org/abs/1604.06778.

[18] Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D., “Deep reinforcement learning that matters,” *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, AAAI press, 2018, pp. 3207–3214.

[19] Lazaridis, A., Fachantidis, A., and Vlahavas, I., “Deep Reinforcement Learning: A State-of-the-Art Walkthrough,” *Journal of Artificial Intelligence Research*, Vol. 69, 2020, pp. 1421–1471.

[20] Ball, P., and Roberts, S., “OffCon3: What is state-of-the-art anyway?” *arXiv*, 2021. URL http://arxiv.org/abs/2101.11331.

[21] Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S., “Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor,” *arXiv*, 2018. URL http://arxiv.org/abs/1801.01290.

[22] Dong, Y., Shi, Z., Chen, K., and Yao, Z., “Self-learned suppression of roll oscillations based on model-free reinforcement learning,” *Aerospace Science and Technology*, Vol. 116, 2021. https://doi.org/10.1016/j.ast.2021.106850.

[23] Delft University of Technology, “Flying-V,” , 2019. URL https://www.tudelft.nl/lr/flying-v.

[24] Van Overeem, S., Wang, X., and van Kampen, E.-J., “Modelling and Handling Quality Assessment of the Flying-V Aircraft,” *AIAA Scitech 2022 Forum*, Delft University of Technology, 2022.

[25] Cappuyns, T., “Handling qualities of a Flying V configuration,” Tech. rep., TU Delft Aerospace Engineering, 2019.

[26] Stevens, B., Lewis, F., and Johnson, E., *Aircraft control and simulation: Dynamics, controls design, and autonomous systems: Third edition*, 2015. https://doi.org/10.1002/9781119174882.

[27] Skybrary, "AIRBUS A350-900 | SKYbrary Aviation Safety," , 2012. URL https://skybrary.aero/aircraft/a359.

[28] van Overeem, S., and van Kampen, E., "Modelling, Control, and Handling Quality Analysis of the Flying-V," Tech. rep., Delft University of Technology, Delft, 2022. URL http://resolver.tudelft.nl/uuid:7fd04eec-41d4-4967-b246-89fdfac2446e.

[29] Grondman, F., Looye, G. H., Kuchar, R. O., Chu, Q. P., and van Kampen, E. J., "Design and flight testing of incremental nonlinear dynamic inversion based control laws for a passenger aircraft," *AIAA Guidance, Navigation, and Control Conference, 2018*, Vol. 0, American Institute of Aeronautics and Astronautics Inc, AIAA, 2018. https://doi.org/10.2514/6.2018-0385.

[30] Carty, R. C., "Advisory Circular Subject: Authorization of Aircraft and Operators for Flight in Reduced Vertical Separation Minimum (RVSM) Airspace," Tech. rep., Federal Aviation Administration, 2019.

[31] Palermo, M., and Vos, R., "Experimental aerodynamic analysis of a 4.6%-scale flying-v subsonic transport," *AIAA Scitech 2020 Forum*, Vol. 1 PartF, American Institute of Aeronautics and Astronautics Inc, AIAA, 2020. https://doi.org/10.2514/6.2020-2228.