CEAS-GNC-2022-077

CEAS EuroGNC 2022
"Conference on Guidance, Navigation and Control"
3-5 May 2022 @ Technische Universität Berlin, Germany

Integrated Updraft Localization and Exploitation: End-to-End Type Reinforcement Learning Approach

Stefan Notter Research Associate, Institute of Flight Mechanics and Controls, University of Stuttgart, 70569, Stuttgart, Germany. stefan.notter@ifr.uni-stuttgart.de

Gregor Müller Graduate Student, Institute of Flight Mechanics and Controls, University of Stuttgart, 70569, Stuttgart, Germany. gregor.mueller94@gmail.com

Walter Fichter Professor, Institute of Flight Mechanics and Controls, University of Stuttgart, 70569, Stuttgart, Germany. walter.fichter@ifr.uni-stuttgart.de

ABSTRACT

Autonomous soaring constitutes an appealing academic sample problem for investigating machine learning methods within the scope of aerospace guidance, navigation, and control. The stochastic nature of small-scale meteorological phenomena renders the task of localizing and exploiting thermal updrafts suited for applying a reinforcement learning approach. Within this work, we present a training setup for learning an integrated control strategy for autonomous localization and exploitation of thermal updrafts. In particular, we propose a deep artificial neural network featuring a *Long Short-Term Memory* to represent the policy. Instead of just implementing a static control law, the recurrent structure facilitates observability and enables mapping the hard-to-model dynamics of thermal updrafts. The end-to-end type control policy integrates an estimator for updraft localization, including a latent state-transition model. We show in simulation, that the trained agent autonomously localizes and exploits stochastic, non-stationary thermal updrafts. The unaltered reinforcement learning setup can be deployed to further improve the control policy through real-world interactions.

Keywords: CEAS EuroGNC; Autonomous Soaring; Intelligent Systems; Artificial Intelligence, Reinforcement Learning; Long Short-Term Memory; End-to-End Learning, Integrated Filtering and Control

1 Introduction

By exploiting atmospheric energy, soaring aircraft can cover large distances without consuming fossil fuel or even electric power. While large transport aircraft have long been taking advantage of strong upper winds, especially unpiloted aircraft can profit from reducing their energy demand by exploiting thermal updrafts in the lower atmosphere. The majority of these aerial vehicles use electric propulsion systems. Thus, range and endurance are limited. Consequently, a significant amount of work has gone into developing guidance and control strategies to exploit thermal updrafts in the last decades [1–4].

To automatically exploit a thermal, one has to locate an updraft first. We classify previously published approaches for the problem of mapping thermal updrafts into model-free methods and methods that employ a thermal updraft observation model. While model-free methods allow for thoroughly mapping thermals as complex meteorologic objects, incorporating a priori knowledge on the general shape of thermals is advantageous for fast localization. We largely borrow the following survey from our pre-

vious work on updraft estimation [5]: "A method to map a single, but arbitrarily shaped thermal cell was presented in [6]. Therein, the problem of estimating the shape of a thermal cell is cast to the problem of estimating coefficients of B-Splines, which piecewise describe the structure of the flow field. A Kalman filter is proposed to recurrently solve the resultant linear estimation problem. Researchers of the same facility had presented a model-free, grid-based approach to thermal mapping already previously, using a set of Kalman filters to estimate the vertical wind speed in each cell individually [7, 8]. Amongst a comprehensive set of algorithms for autonomous soaring aircraft, a refined occupancy grid approach to thermal mapping was presented, implemented, and flight-tested by that group, more recently [9, 10]. Another interesting approach to model-free wind field estimation for autonomous soaring flight is presented in [11]. Instead of using a grid-based representation, the authors propose the method of Gaussian process (GP) regression to generate a continuous map of the wind field from local observations. Although not being based on a distinctive updraft model, physical properties such as spatio-temporal smoothness and expected drift can be accounted for by tailoring the structure of the covariance function, which correlates individual measurements taken. Given their model-free nature, all the approaches mentioned so far are capable of mapping an updraft flow field of unknown shape. On the downside, model-free approaches are inherently less efficient when fast localization of the core of a thermal cell is key.
In [12], the basic centroid method from [2] was enhanced by estimating the parameters of a circular bell curve thermal updraft observation model, applying a least-squares evolutionary-search method. The approach was impressively validated by placing third in a remote-controlled (RC) glider aircraft competition, outperforming all but the most experienced RC pilots. Two coupled extended Kalman filters (EKF) are used by [13] to produce a thermal location estimate. Simulation results indicate that a single thermal is located fairly quickly, enabling a guidance system to exploit the updraft. A similar way of characterizing a single thermal cell is presented in [14]. Therein, the authors propose to augment the local updraft velocity observation by an induced roll moment measurement in order to improve the observability of the estimated thermal updraft parameters. With the performance assessed by a comprehensive analysis of simulation results, the approach is finally validated by actual flight test results. Another EKF-based solution to the problem of estimating the characteristics of a single thermal is presented in [15]."
Lastly, the authors of the paper at hand themselves proposed a particle-filter-based updraft estimator [5, 16]. Given its non-parametric nature, this approach offers the unique ability to localize several idealized thermals at once. However, the complex shape and the chaotic dynamics of real thermals are hard to model. Therefore, applying model-based approaches that estimate parameters of a simple observation model is, to some extent, performance-limiting in a real-world scenario.

What if we could learn the shape and the dynamics of thermal updrafts directly based on interactions with the environment? Supervised learning of a sophisticated updraft model is not an option, as ground truth data are hardly available. Reinforcement learning (RL), in contrast, does not require a ground truth but only a measure for what the designer deems a favorable outcome. The authors themselves recently proposed and flight-test-validated a reinforcement learning approach to thermal updraft exploitation [17]. That approach, however, features a dedicated updraft estimator to localize thermals. The estimated thermal updraft positions are then fed to an RL control policy. A multi-agent reinforcement learning approach for the task of multiple gliders cooperatively exploiting an arbitrary number of thermals is presented in [18]. Still, the information on the positions of the updrafts is fed to the reinforcement learning agents. Also within the authors' previous work, a trained policy for longitudinal control was presented, which integrates detection and exploitation thermals [19]. A basic reinforcement learning approach to integrated localization and exploitation of thermal updrafts by controlling the aircraft bank angle was presented by [20]. The ability of a reinforcement learning agent to maximize the gain in height by thermal updraft exploitation solely based on the "glider's pooled experiences, collected over several days in the field" was impressively demonstrated by the same team [21].

The approach proposed herein addresses a similar problem. Instead of applying tabular Q-Learning with a coarse discretization of both the state-space and the action-space, we propose a stochastic pol-

icy gradient ascent algorithm to learn a continuous control policy. Furthermore, this paper reasons the superiority of a recurrent policy architecture for integrated updraft localization and exploitation. Applying a deep artificial neural network (ANN) featuring a *Long Short-Term Memory* (LSTM), the policy representation facilitates updraft localization through improved observability. Moreover, the recurrent structure also enables mapping the hard-to-model dynamics of thermal updrafts. The proposed guidance for updraft exploitation results from end-to-end type learning without the need for a separate updraft estimator or any preprocessing of the measured variables indeed.

The structure of the paper is as follows: In section 2, we set up the task of thermal updraft exploitation as a reinforcement learning problem. The recurrent policy structure proposed is described in section 3 on page 5. Furthermore, we outline the basic principles of model-free reinforcement learning and policy gradient ascent. In section 4 on page 8, we describe the specific training setup. Simulation results showcase how the integrated control policy allows the trained agent to localize and exploit thermal updrafts. Lastly, we draw some conclusions regarding feasibility and future extensions in section 5 on page 10.

2 Problem Statement

Thermal updrafts caused by atmospheric convection are the prime source of free energy available for harvesting at altitudes below the free convective layer. Thus, autonomous soaring for unmanned aircraft is mostly about localizing and exploiting thermal updrafts. Missions that require a fixed-wing aircraft to fly for a long time at a relatively low altitude, such as search and rescue or surveillance applications, particularly profit from thermal updraft exploitation.

2.1 Integrated Updraft Localization and Exploitation

As stated in the introduction section 1, most previously published approaches to autonomous thermal updraft exploitation feature a classical filter for estimating the center position of a simplified thermal updraft observation model. The outcome of such an estimator is then fed to a classical guidance and control scheme, which lets the aircraft circle the assumed center of the thermal cell. Being small-scale meteorological objects of complex shape, real thermal updrafts, unfortunately, can hardly be comprehensively mapped by simplified models of both the general shape and the updraft dynamics. Instead of estimating parameters of a questionable updraft model, we propose a machine learning approach. As valid ground truth data are not available, trying to simply substitute a classical updraft estimator through supervised learning would contradict the basic idea of not applying any questionable belief on the shape and the dynamics of real thermals. Reinforcement learning, in contrast, is right about learning directly from interactions with the environment without the need for any a priori expert knowledge. Still, a notion of what is a good or a bad outcome of the trained mapping is necessary (cf. section 3 on page 5). Whereas we can hardly judge the outcome of an updraft estimator, we can easily judge the change in the energy state (i.e., height and velocity) of the glider aircraft given a certain action. Therefore, we propose learning a more end-to-end type mapping from measurable variables to a bank command, leaving the representation of thermals to the internal state of the integrated updraft exploitation control policy.

2.2 3-DoF Glider Dynamics in the Presence of Wind

Despite the reinforcement learning approach proposed in section 3 on page 5 is classified as being model-free, realistically, an environment simulation is always necessary to solve flight control problems by reinforcement learning as crashing a real aircraft in the early training phase is not an option. Within our previous work, we contributed by presenting the unsimplified three degrees of freedom (3-DoF) equations of motion describing the dynamics of a glider aircraft in the presence of an arbitrary wind

field [17]. Assuming a flat, non-rotating Earth, the basic equations of motion are given by:

$$\dot{\mathbf{p}} = \mathbf{v}_K \tag{1}$$

$$\dot{\mathbf{v}}_K = \frac{1}{m}\left(\mathbf{f}_A + \mathbf{f}_G\right) \tag{2}$$

All variables in Eq. (1) and Eq. (2) are denoted with respect to a local Earth-fixed, north-east-down (NED) frame of reference. The flight-mechanical state $\mathbf{x}$ consists of the position $\mathbf{p}$ of the aircraft with mass m and the track velocity $\mathbf{v_K}$:

$$\mathbf{x} = \begin{bmatrix} {}^{E}\mathbf{p} \\ {}^{E}\mathbf{v}_K \end{bmatrix} \tag{3}$$

Assuming zero sideslip, the control vector $\mathbf{u}$ consists of the angle of attack α and the air-relative bank angle μ:

$$\mathbf{u} = [\alpha,\ \mu]^{\mathsf{T}} \tag{4}$$

Whereas the gravitational force in Eq. (2) simply reads $\mathbf{f}_G = [0,\ 0,\ mg]^{\mathsf{T}}$, the reader is referred to the aforementioned paper [17] for a sound description of how the aerodynamic force $\mathbf{f}_A\left(\mathbf{x},\ \mathbf{u},\ \mathbf{v}_W\right)$ in NED-coordinates results from state variables, control variables, and an arbitrary wind vector $\mathbf{v}_W$.

2.3 Markov Decision Process

For applying reinforcement learning to integrated updraft localization and exploitation, we need to formally frame the problem in terms of a Markov decision process (MDP). An MDP is defined by a tuple $\langle \mathscr{S},\ \mathscr{A},\ P(\cdot),\ R(\cdot),\ \gamma \rangle$. For the task at hand, the components of that defining tuple are as follows:

- State-Space $\mathscr{S}$:
  The Markovian system state $\mathbf{s}$ consists of the flight-mechanical state as given in Eq. (3) and the wind velocity at the position of the aircraft:

  $$\mathscr{S} = \{\mathbf{p},\ \mathbf{v}_K,\ \mathbf{v}_W\} \tag{5}$$

  We assume a fully observable state in the remainder of this paper.
- Action-Space $\mathscr{A}$:
  The action $\mathbf{a}$ is given by commanding a bank angle μ:

  $$\mathscr{A} = \{\mu\} \tag{6}$$

  subject to the control constraint:$-30^\circ \leq \mu \leq 30^\circ$. Within the scope of this paper, we set the angle of attack to a fixed value of 6° with reference to the zero-lift angle of attack $\alpha_{c_L=0}$.
- State-Transition-Probability $P(\cdot)$:
  The probability of action $\mathbf{a}$ taken in state $\mathbf{s}$ leading to the subsequent state $\mathbf{s}'$ combines the (deterministic) 3-DoF glider dynamics $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\ \mathbf{u})$ (cf. section 2.2 on the preceding page) subject to the stochastic occurrence of thermal updrafts.
- Reward-Function $R(\cdot)$:
  Solving an MDP is about finding a policy π – a stochastic control law in terms of classical control – that maximizes the expectation of the cumulative future reward. Consequently, the reward function represents the objective. The objective of integrated updraft localization and exploitation is to maximize the energy harvested from the environment[1]. We propose to use the integrated output signal of a total energy compensated vertical speed indicator as a reward signal [22]. The energy-equivalent climb rate $\dot{e}$ takes both changes in height $\dot{h}$ and a changing airspeed $\dot{V}_A$ into account:

[1] Note that localizing an updraft is only a means but not the objective of the integrated approach proposed.

$$R : \mathscr{S} \to \mathbb{R},\ \mathbf{s} \mapsto r = \left(\underbrace{\dot{h} + \frac{V_A \dot{V}_A}{g}}_{=\dot{e}} \right) \Delta t \tag{7}$$

- Discount-Factor γ:
  The factor γ discounts future reward. Whereas a small value makes the agent act greedily, a value close to one lets the agent act foresighted. Only for the sake of stable convergence of the reinforcement learning approach, we apply a slight discount of $\gamma = 0.99$.

3 Reinforcement Learning Approach

3.1 Model-Free Policy-Based Reinforcement Learning

Reinforcement learning is one of the three main domains of machine learning. In contrast to supervised learning, an agent does not learn the correct behavior from a large data set of examples but through direct interaction with the environment. At time instance t_k, the agent observes the system state $\mathbf{s}_k$ and selects an action $\mathbf{a}_k$, according to a policy $\pi_\theta(\mathbf{a}|\mathbf{s})$ subject to parameters θ. After the interaction, the agents receives a reward scalar r_{k+1} (according to Eq. (7), herein) [23]. Figure 1 illustrates the closed-loop model-free reinforcement learning principle. The term "model-free" refers to the fact that the agent does not know the state transition model $P(\cdot)$ of the environment and does not seek to explicitly learn it. Instead, the agent directly learns a behavior strategy that maximizes the agent's expectation of the discounted, cumulative future reward. This strategy is represented by a stochastic policy π_θ, which specifies the probability for taking an action $\mathbf{a}$ given a system state $\mathbf{s}$:

$$\pi_\theta(\mathbf{a}|\mathbf{s}) = P(\mathbf{a} \in \mathscr{A} \,|\, \mathbf{s} \in \mathscr{S}) \tag{8}$$

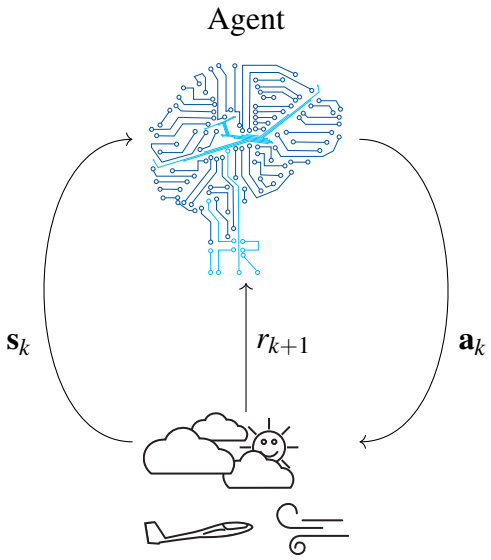


Fig. 1 Closed-loop reinforcement learning principle.

Policy gradient approaches try to directly learn optimal values for θ by maximizing the expectation of the discounted, cumulative future reward over an episode of length T. Finding the optimal policy

turns into a stochastic optimization problem:

$$\underset{\theta}{\text{maximize}} \left\{ J(\theta) = E_\theta \left(\sum_{k=1}^{T} \gamma^k r_k \right) \right\} \quad \text{s.t. } \langle \mathscr{S},\ \mathscr{A},\ P(\cdot),\ R(\cdot),\ \gamma \rangle \tag{9}$$

The most common approach to solving this type of optimization problem is basic gradient ascent:

$$\theta' = \theta + \lambda\ \nabla_\theta J(\theta) \tag{10}$$

It maximizes the objective function $J(\theta)$ by taking small steps in the direction of the gradient $\nabla_\theta J(\theta)$. The learning rate λ is a hyper-parameter for controlling the step size. Applying gradient ascent to a model-free approach needs some modifications as the objective function depends on the unknown transition probability $P(\cdot)$. Estimating the gradient of the objective function with respect to the policy parameterization $\nabla_\theta J(\theta)$ from a trajectory $\mathscr{T}$ of sampled interactions avoids the explicit evaluation of $P(\cdot)$:

$$\nabla_\theta J(\theta) = E_{\mathscr{T}\sim\pi_\theta} \left(\sum_{k=1}^{T} \gamma^k r_k \cdot \nabla_\theta \log p(\mathscr{T} \mid \theta) \right) \tag{11}$$

$$\approx \frac{1}{N} \sum_{\mathscr{T}} \left(G(\mathscr{T}) \sum_{k=0}^{T-1} \nabla_\theta \log \pi_\theta\left(\mathbf{a}_k \mid \mathbf{s}_k\right) \right) \tag{12}$$

Therein, $G(\mathscr{T})$ denotes the sampled return of a trajectory. The so-called score function gradient estimator (Eq. 12) results from the policy gradient theorem [23] and renders the core element of the *REINFORCE* algorithm [24]. It approximates the expectation in Eq. 11 by evaluating a batch of sampled trajectories and yields an unbiased estimator for the gradient of the cost function.

3.2 Proximal Policy Optimization

Simple learning approaches like gradient ascent can achieve good results when applied to simple problems but suffer from poor convergence properties when facing complex problems. One major drawback is the constant step size. *Trust Region Policy Optimization* (TRPO) [25] introduces a constraint into the optimization problem and maximizes a surrogate objective function with so that the Kullback-Leibler divergence between two policies π_θ and π'_θ is bounded. This leads to monotonic policy improvement with adaptive step size. For a formal derivation of the algorithm, we refer to the original publication [25]. *Proximal Policy Optimization* (PPO) [26] is a modification of TRPO, wherein the constraint optimization problem is approximated with an unconstrained problem by clipping the surrogate objective function:

$$J^{\text{CLIP}}\left(\theta'\right) = E_{\mathscr{T}\sim\pi_\theta} \left(\min \left(\frac{\pi_{\theta'}}{\pi_\theta} \hat{A}_\theta,\ \text{clip} \left(\frac{\pi_{\theta'}}{\pi_\theta},\ 1-\varepsilon,\ 1+\varepsilon \right) \hat{A}_\theta \right) \right) \tag{13}$$

Therein, the estimated advantage $\hat{A}_k$ accounts for how much the policy has improved, while the clipping value ε is a tunable hyper-parameter. Due to its convincing results in many benchmark problems and the simple implementation, PPO has become a standard policy optimization algorithm for complex control tasks. Again we refer to the original publication for a formal derivation [26].

3.3 Control Policy Representation

For the problem of estimating the position of an idealized thermal from a scalar measurement, observability can only be achieved by a sequence of measurements [17]. More generally speaking: From a single measurement $\mathbf{y}_k \in \mathbb{R}^n$ not more than n parameters of a (steady-state) system are observable at a time. However, only a limited number of sensors is realistically applicable for small fixed-wing air-

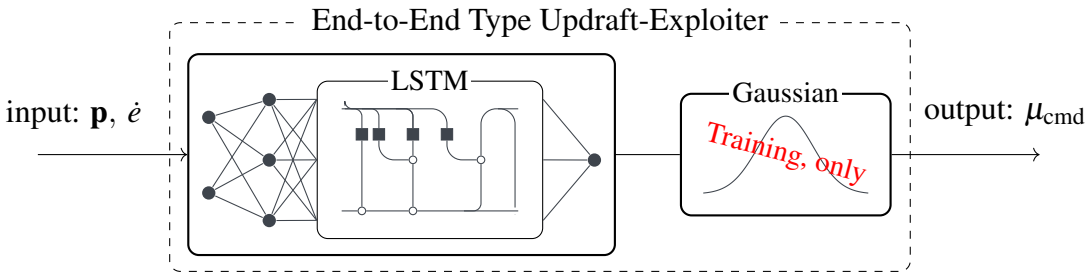


Fig. 2 Recurrent policy architecture for integrated updraft localization and exploitation.

craft. Therefore, mapping a thermal of complex shape requires evaluating a sequence of measurements. Consequently, a policy architecture requires an internal state to allow for the emulation of an updraft estimator. Moreover, if we want to keep track of the evolution of a thermal updraft, a classical unbiased estimator requires not only an observation model but also a process model. To be capable of mapping the dynamics of the real system, however, the artificial neural network that implements the updraft estimator needs to be a dynamic system itself. We propose a deep artificial neural network featuring a *Long Short-Term Memory* as depicted in Fig. 2 to represent the control policy for updraft exploitation. Given its feedback connections, the LSTM enhances observability and allows for mapping the system dynamics of a thermal updraft. Note that thermal updraft estimates, however, are only internally encoded by the ANN. The integrated control policy directly maps from aircraft position $\mathbf{p}$ and the energy-equivalent climb rate $\dot{e}$ to a bank angle command μ_{cmd}. Note that the input variables to the end-to-end type control policy can directly be obtained from sensor measurements without the need for any preprocessing. Instead, the input is encoded by a fully connected hidden layer consisting of 32 neurons with $\tanh(\cdot)$ activation functions. Downstream the first hidden layer, the hidden state vector of the LSTM is of dimension 32. The output of the LSTM is finally decoded to output the expected value for the aircraft bank angle, normalized to $\pm 30^\circ$.

3.4 Training Setup

The training environment for the end-to-end type updraft exploiter is set up as an extension of the *Gym* library [27]. Its core is the 3-DoF glider dynamics as described in section 2.2 and the thermal updraft model presented in [28]. For each episode during training, the velocity of the horizontal wind is drawn from a half-normal distribution with scale parameter $\sigma = 1\,\mathrm{m/s}$. The wind direction is drawn from a uniform distribution. Forty individual thermals are randomly scattered within a radius of 2 km. The simulated thermals drift at half the speed of the horizontal wind. During training, the actions of the agent are drawn from a Gaussian distribution to ensure exploration in the policy parameter space. The parameters associated to the environment simulation are listed in Table 1 on page 13.

Both the ANN policy representation (cf. Fig. 2) and the reinforcement learning algorithm of *Proximal Policy Optimization* are implemented using the *PyTorch* library [29]. The most relevant reinforcement learning (hyper-)parameters applied for training the end-to-end type control policy for integrated updraft localization and exploitation are listed in Table 2 on page 13. To foster transparency and allow other researchers and practitioners to potentially adopt and build upon our approach, the source code of our reinforcement learning framework including the environment simulation is made publicly available on *GitHub*[2].

[2] `https://github.com/ifrunistuttgart/RL_Integrated-Updraft-Exploitation`

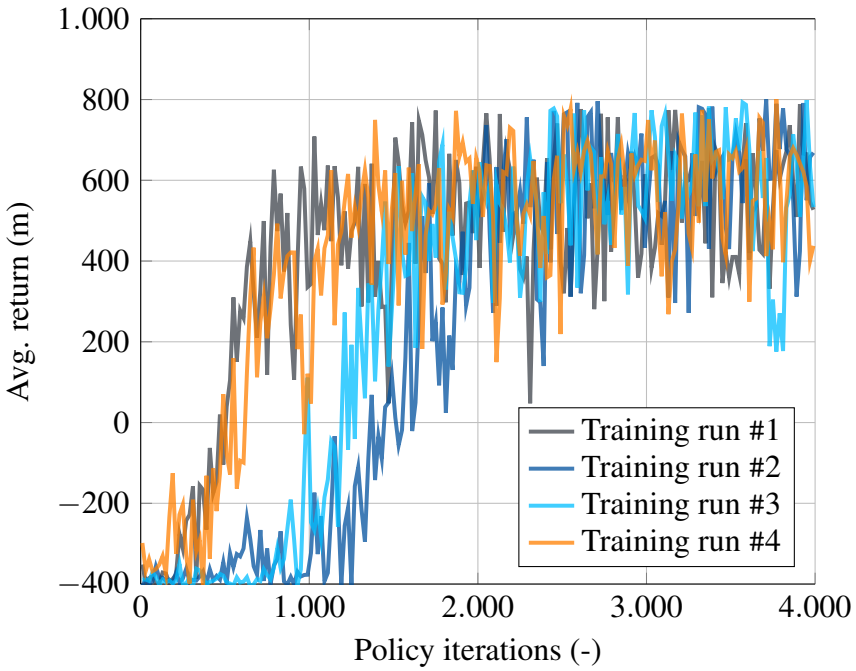


Fig. 3 Evolution of average returns during training.

4 Training Results

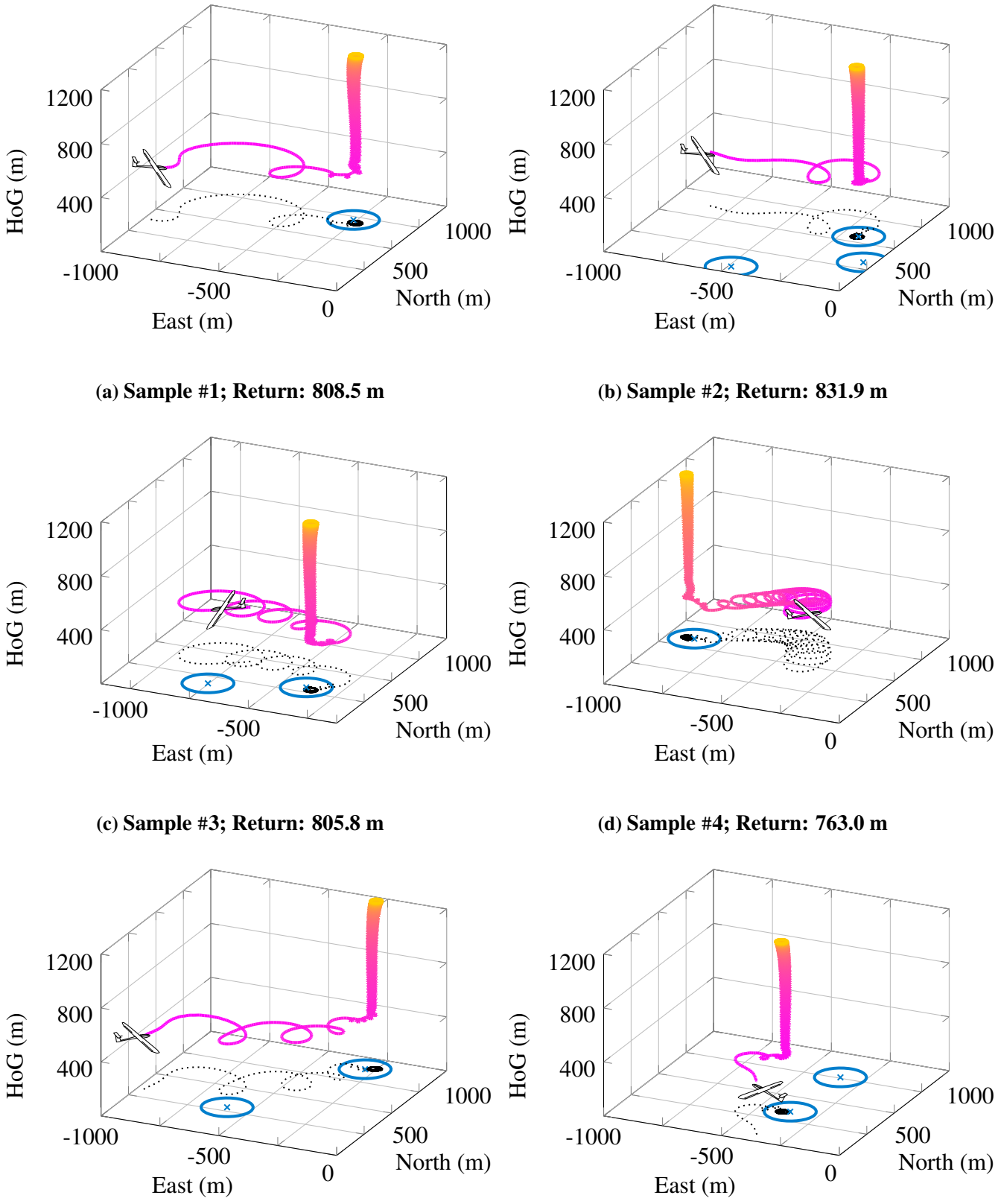
We trained the updraft exploiter over 4e3 policy iterations. As training is a random process, Fig. 3 shows the results of four independent training runs with identically set up experiments subject to different random seeds. The return reflects the altitude gained during an episode. An episode ends when the glider touches the ground or 30 minutes of flight time have elapsed. The stochastic environment can cause high fluctuations in subsequent roll-outs. Therefore, the returns displayed are averaged over ten subsequent iterations. All four training runs show a volatile but increasing evolution of the average return.

Training run #3 results in the policy with the highest final average return. All results presented in the following are subject to the policy resulting from that experiment after 4e3 iterations. Unlike during training, the actions are not drawn from Gaussians. Instead, the direct outcome of the policy net is applied. The results discussed in the subsequent sections provide an insight into the trained agent´s behavior.

4.1 Localization and Exploitation of Stationary Thermal Updrafts

Figure 4 on the next page shows glider position trajectories of multiple episodes with randomly sampled, stationary updraft distributions. All samples show the agent performing some exploratory soaring until an updraft is reached. Inside the updraft, the agent is circling until the end of the episode to maximize the altitude gain. In all six episodes displayed, a final return of approximately 800 meters is reached. The concentric circling of an updraft is made possible by the recurrent architecture of the policy, encoding the estimated center position of the updraft in its internal state.

Within the appendix, we showcase the inferior behavior of a similarly trained agent, the policy representation of whom does not feature an LSTM but an additional feed-forward layer, instead. Given the sparse input of aircraft position and energy-equivalent climb rate, only the recurrent architecture ensures observability in the first place.



(a) Sample #1; Return: 808.5 m

(b) Sample #2; Return: 831.9 m

(c) Sample #3; Return: 805.8 m

(d) Sample #4; Return: 763.0 m

(e) Sample #5; Return: 820.7 m

(f) Sample #6; Return: 835.1 m

Fig. 4 Position trajectories for randomly sampled initial conditions and updraft distributions subject to the proposed control policy featuring an LSTM: The trained agent is capable of centering stationary thermals.

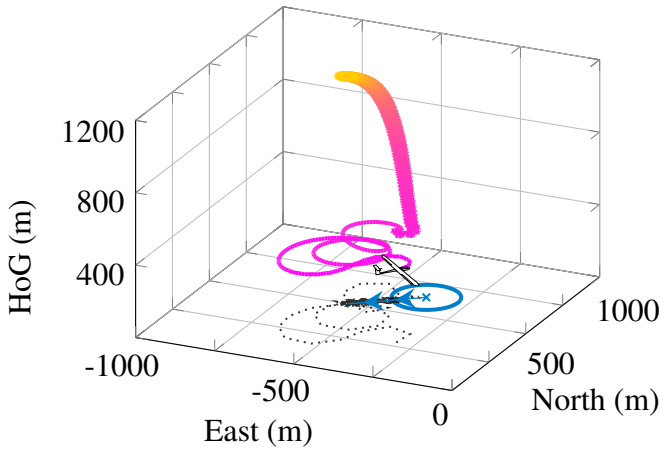
4.2 Localization and Exploitation of Drifting Thermal Updrafts

In a real-world scenario, thermal updrafts drift according to horizontal winds. Figure 5 on the following page shows glider position trajectories of multiple episodes with randomly sampled, non-stationary updraft distributions. Not only does the glider face horizontal winds, but the thermals drift away with the wind. In the samples depicted in Fig. 5, arrows indicate the respective horizontal wind speed and wind direction. Just as for the stationary updraft examples depicted in Fig. 4, all samples show the agent staying within the center of a localized updraft until the end of the respective episode. The trained agent not featuring a recurrent policy architecture is not even capable of centering a stationary thermal updraft (cf. Fig. 7 on page 12). The agent subject to the policy featuring an LSTM, in contrast, keeps track of non-stationary thermals, even. A dynamical system itself, the architecture allows learning to encode the updraft dynamics. The agent subject to the recurrent policy architecture proposed integrates an estimator for updraft localization, including a latent updraft state-transition model.

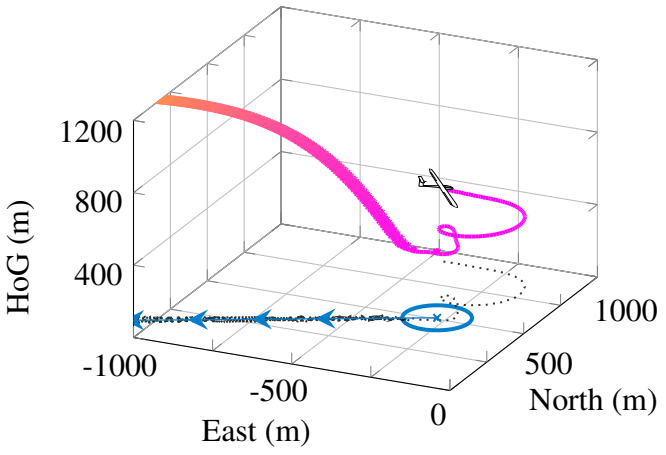
5 Conclusion

A reinforcement learning approach to the problem of autonomous thermal updraft localization and exploitation has been presented. An integrated control policy featuring a *Long Short-Term Memory* has been proposed to control the bank angle of the aircraft such that as much free energy as possible is harvested. The optimization is solely based on interactions with the (simulated, for now) environment. Simulation results have showcased the ability of the soaring agent to autonomously localize and exploit thermal updrafts in a stochastic environment. We both reasoned and demonstrated that for learning to map directly measurable variables to the control variable, a policy featuring a recurrent structure results in superior behavior of an agent trained on integrated updraft localization and exploitation.

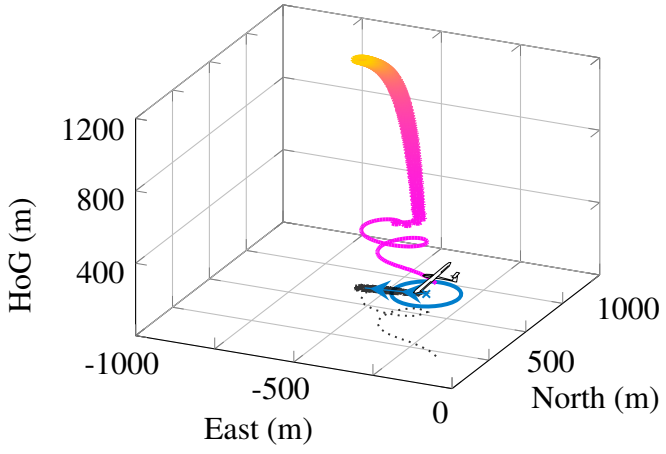
Naturally, the authors aim at demonstrating the feasibility and performance of the approach proposed through flight-test demonstrations. As we showed before, low-cost off-the-shelf embedded hardware can evaluate a recurrent artificial neural network control policy [17, 30]. By reapplying the unaltered reinforcement learning setup for samples taken through real-world interactions, the approach enables adapting to the complex shape and dynamics of real-world thermal updrafts.



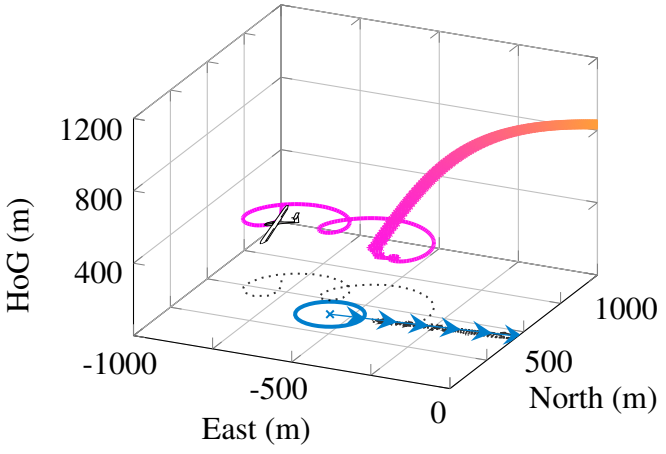
(a) Sample #1; Return: 829.1 m



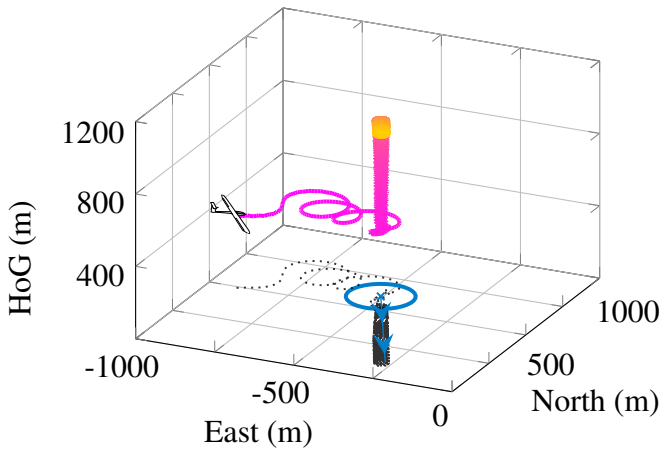
(b) Sample #2; Return: 829.4 m



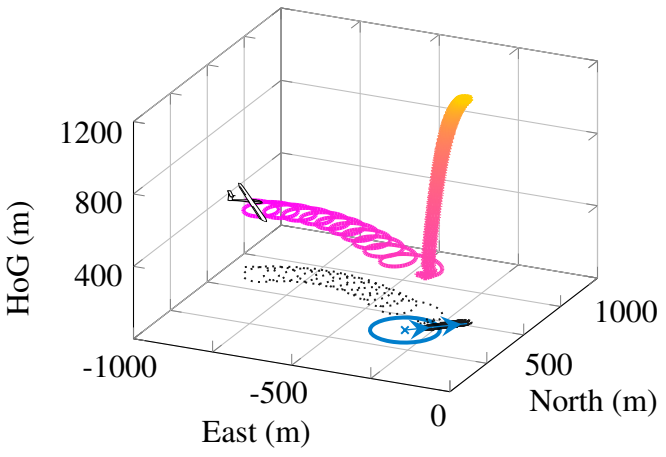
(c) Sample #3; Return: 828.6 m



(d) Sample #4; Return: 827.5 m



(e) Sample #5; Return: 829.6 m



(f) Sample #6; Return: 807.8 m

Fig. 5 Position trajectories for randomly sampled initial conditions and updraft distributions resulting from a control policy featuring an LSTM: The trained agent is capable of tracking and exploiting drifting thermals.

Appendix

Benchmark: Feed-Forward Control Policy

For comparison, we provide an insight into the behavior of an equally trained soaring agent subject to a control policy that does not feature an internal state but only consists of feed-forward layers. Figure 6 depicts the architecture of the feed-forward deep artificial neural network applied. For the Markov property to hold, we augmented input by the aircraft velocity vector $\mathbf{v}_K$. Two exemplary resultant position trajectories, subject to the same conditions as those in Fig. 4 on page 9, are shown in Fig. 7 . The feed-forward control policy lacks an internal state. The agent is not capable of localizing the center of an updraft. As outlined in [5], the updraft center position is not observable based on a single snapshot of the variables input to the ANN. The glider starts to circle, once it has hit an updraft. However, it loses track of the (steady-state!) thermal in both samples depicted. Consequently, the return achieved is much lower and more dependent on the stochastic updraft distribution as for the samples depicted in Fig. 4 on page 9 subject to the agent whose policy features a recurrent structure. Moreover, a soaring agent not featuring a recurrent structure will never be capable of learning the dynamics of an updraft, no matter what variables are input to the policy.

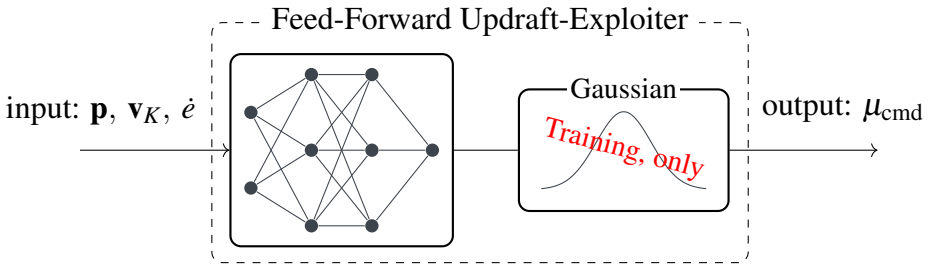
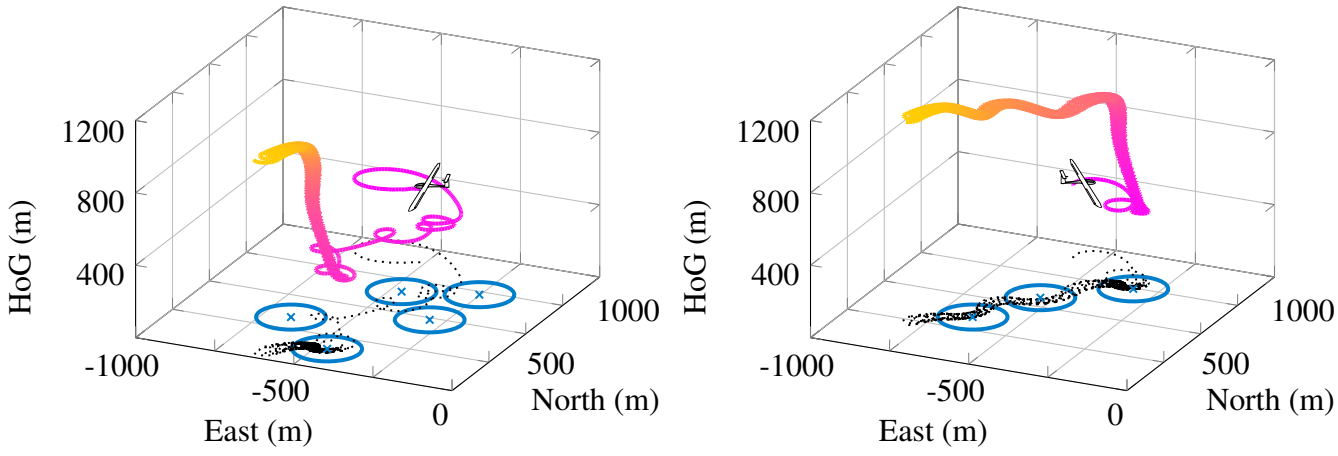


Fig. 6 Benchmark feed-forward policy architecture.



(a) Sample #1; Return: 484.5 m **(b) Sample #2; Return: 683.2 m**

Fig. 7 Position trajectories for randomly sampled initial conditions and updraft distributions subject to a feed-forward control policy: The trained agent is *NOT* capable of centering stationary thermals.

Parameter Listing

The complete parametrization of the environment simulation is listed in Table 1. The most relevant parameters for the reinforcement learning setup are listed in Table 2.

Table 1 Environment parameters

(a) Basic physical parameters

Parameter	Identifier	Unit	Value
Gravitational accel.	g	m/s^2	9.810
Air density	ρ	kg/m^3	1.225

(b) Glider parameters

Parameter	Identifier	Unit	Value
Aircraft mass	m	kg	5.000
Wing area	S	m^2	0.790
Aspect ratio	Λ	-	23.60
Oswald factor	e	-	0.900
Zero-lift drag coefficient	C_{D0}	-	0.015

(c) Updraft simulation

Parameter	Identifier	Unit	Value
Conv. scale	w^*	m/s	2.061
Conv. layer	z_i	m	1496

Table 2 Reinforcement learning (hyper-)parameters

Parameter	Value
Batch size	4096
Sequence length	256
Learning rate, actor	1e−5
Learning rate, critic	1e−4
Discount factor	0.99
PPO clipping value	0.2

References

[1] Michael Allen. Autonomous soaring for improved endurance of a small uninhabitated air vehicle. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*, page 1025, 2005. DOI: 10.2514/6.2005-1025.

[2] Michael Allen and Victor Lin. Guidance and Control of an Autonomous Soaring Vehicle with Flight Test Results. In *45th AIAA Aerospace Sciences Meeting and Exhibit*, page 867, 2007. DOI: 10.2514/6.2007-867.

[3] Daniel Edwards. Implementation Details and Flight Test Results of an Autonomous Soaring Controller. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, Guidance, Navigation, and Control and Co-located Conferences. American Institute of Aeronautics and Astronautics, 2008. DOI: 10.2514/6.2008-7244.

[4] Klas Andersson, Isaac Kaminer, Vladimir Dobrokhodov, and Venanzio Cichella. Thermal Centering Control for Autonomous Soaring; Stability Analysis and Flight Test Results. *Journal of Guidance, Control, and Dynamics*, 35(3):963–975, may 2012. DOI: 10.2514/1.51691.

[5] Stefan Notter, Pascal Groß, Philipp Schrapel, and Walter Fichter. Multiple Thermal Updraft Estimation and Observability Analysis. *Journal of Guidance, Control, and Dynamics*, 43(3):490–503, 2020. DOI: 10.2514/1.G004205.

[6] John Bird and Jack Langelaan. Spline mapping to maximize energy exploitation of non-uniform thermals. *Technical Soaring*, 37(3):38–44, 2013.

[7] Nathan Depenbusch and Jack Langelaan. Coordinated Mapping and Exploration for Autonomous Soaring. In *Infotech@ Aerospace 2011*, page 1436. American Institute of Aeronautics and Astronautics, 2011. DOI: 10.2514/6.2011-1436.

[8] Kwok Cheng and Jack W Langelaan. Guided exploration for coordinated autonomous soaring flight. In *AIAA Guidance, Navigation, and Control Conference*, page 0969, 2014.

[9] Nathan T. Depenbusch, John J. Bird, and Jack W. Langelaan. The AutoSOAR Autonomous Soaring Aircraft, Part 1: Autonomy Algorithms. *Journal of Field Robotics*, 35(6):868–889, 2018. DOI: 10.1002/rob.21782.

[10] Nathan T. Depenbusch, John J. Bird, and Jack W. Langelaan. The AutoSOAR Autonomous Soaring Aircraft, Part 2: Hardware Implementation and Flight Results. *Journal of Field Robotics*, 35(4):435–458, 2018. DOI: 10.1002/rob.21747.

[11] Nicholas R. J. Lawrance and Salah Sukkarieh. Path Planning for Autonomous Soaring Flight in Dynamic Wind Fields. In *2011 IEEE international conference on robotics and automation*, pages 2499–2505. IEEE, 2011. DOI: 10.1109/ICRA.2011.5979966.

[12] Daniel J. Edwards and Larry M. Silberberg. Autonomous Soaring: The Montague Cross-Country Challenge. *Journal of Aircraft*, 47(5):1763–1769, 2010. DOI: 10.2514/1.C000287.

[13] Aaron D. Kahn. Atmospheric thermal location estimation. *Journal of Guidance, Control, and Dynamics*, 40(9):2363–2369, September 2017. DOI: 10.2514/1.g002782.

[14] Philipp Oettershagen, Thomas Stastny, Timo Hinzmann, Konrad Rudin, Thomas Mantel, Amir Melzer, Bartosz Wawrzacz, Gregory Hitz, and Roland Siegwart. Robotic Technologies for Solar-Powered UAVs: Fully Autonomous Updraft-Aware Aerial Sensing for Multiday Search-and-Rescue Missions. *Journal of Field Robotics*, 35(4):612–640, 2018. DOI: 10.1002/rob.21765.

[15] Iain Guilliard, Rick Rogahn, Jim Piavi, and Andrey Kolobov. Autonomous Thermalling as a Partially Observable Markov Decision Process. In *Robotics: Science and Systems XIV*. Robotics: Science and Systems Foundation, jun 2018. DOI: 10.15607/rss.2018.xiv.068.

[16] Stefan Notter, Philipp Schrapel, Pascal Groß, and Walter Fichter. Estimation of Multiple Thermal Updrafts Using a Particle Filter Approach. In *AIAA Guidance, Navigation, and Control Conference*, AIAA SciTech Forum. American Institute of Aeronautics and Astronautics, 2018. DOI: 10.2514/6.2018-1854.

[17] Stefan Notter, Fabian Schimpf, and Walter Fichter. Hierarchical Reinforcement Learning Approach Towards Autonomous Cross-Country Soaring. In *AIAA Scitech 2021 Forum*. American Institute of Aeronautics and Astronautics, jan 2021. DOI: 10.2514/6.2021-2010.

[18] Fabian Schimpf, Stefan Notter, Pascal Groß, and Walter Fichter. Multi-Agent Reinforcement Learning for Thermalling in Updrafts. In *AIAA Scitech 2021 Forum*, page 0864, 2021. DOI: 10.2514/6.2021-0864.

[19] Stefan Notter, Markus Zürn, Pascal Groß, and Walter Fichter. Reinforced Learning to Cross-Country Soar in the Vertical Plane of Motion. In *AIAA Scitech 2019 Forum*, 2019. DOI: 10.2514/6.2019-1420.

[20] Gautam Reddy, Antonio Celani, Terrence J. Sejnowski, and Massimo Vergassola. Learning to Soar in Turbulent Environments. *Proceedings of the National Academy of Sciences*, 113(33):E4877–E4884, aug 2016. DOI: 10.1073/pnas.1606075113.

[21] Gautam Reddy, Jerome Wong-Ng, Antonio Celani, Terrence J. Sejnowski, and Massimo Vergassola. Glider Soaring via Reinforcement Learning in the Field. *Nature*, 562:236–239, October 2018. DOI: 10.1038/s41586-018-0533-0.

[22] Pascal Groß, Stefan Notter, and Walter Fichter. Estimating Total Energy Compensated Climb Rates from Position Trajectories. In *AIAA Scitech 2019 Forum*, page 0828, 2019. DOI: 10.2514/6.2019-0828.

[23] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, USA, 1st edition, 1998.

[24] Ronald J. Williams. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning*, 8(3):229–256, May 1992. DOI: 10.1007/BF00992696.

[25] John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust Region Policy Optimization, 2015.

[26] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, 2017.

[27] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. OpenAI Gym. `https://arxiv.org/pdf/1606.01540.pdf`, 2016.

[28] Michael J. Allen. Updraft Model for Development of Autonomous Soaring Uninhabited Air Vehicles. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, page 1510, 2006. DOI: 10.2514/6.2006-1510.

[29] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[30] Marc R. Schlichting, Stefan Notter, and Walter Fichter. Long Short-Term Memory for Spatial Encoding in Multi-Agent Path Planning. *Journal of Guidance, Control, and Dynamics*, 2022. Articles in Advance. DOI: 10.2514/1.G006129.