Run-to-Run Disturbance Rejection for Feedforward Path Following of an Adaptively Controlled Unmanned Helicopter

Johann C. Dauer[1], Timm Faulwasser[2], and Sven Lorenz[1]

***Abstract*— We present a scheme for run-to-run disturbance rejection in optimization-based feedforward path following of a remotely piloted aircraft system (RPAS). The proposed scheme is based on the inter-run estimation of unknown disturbances, such as wind induced forces and model uncertainties. These disturbance estimates are introduced in an optimal control problem used to compute feedforward controls. In order to achieve good run-to-run disturbance rejection, the structure of the underlying stabilizing flight control of the RPAS is taken into account. In this work, we consider flight control based on adaptive reference following and the special case of the unmanned helicopter ARTIS. We present simulation results and flight test data. These results underpin that the proposed approach significantly decreases flight path deviations in a run-to-run fashion.**

***Index Terms*— feedforward path following, remotely piloted aircraft system, adaptive control, model reference control, run-to-run disturbance rejection, unmanned aircraft system, unmanned helicopter**

I. INTRODUCTION

One strength of unmanned helicopters, like ARTIS (Autonomous Rotorcraft Testbed of Intelligent Systems) shown in Figure 1, is the ability to fly through obstacle occupied environments. The problem of automatic navigation through these environments, however, is a very complex task. Even if the environment is known a priori, it involves finding the way through the environment without getting stuck in dead ends, considering constraints on the system dynamics as well as mission requirements on the flight performance, and stabilizing the helicopter along its route.

Consequently, the problem is often split into different sub-tasks. Common problem decompositions involve sensor fusion, path planning, and flight stabilization, cf. [1], [2], [3]. While the path-planner determines safe geometric paths through the obstacle field [4], the flight control system stabilizes the helicopter and provides the interface to steer the helicopter along the path [1], [2], [3]. Alternatively, planning can be done using motion primitives [5], [6].

The flight control system considered in this paper determines reference trajectories using a designed reference model [7], [8]. This reference model defines the desired closed-loop dynamics. If the path to be tracked has a non-trivial geometry with varying curvature these closed-loop dynamics have to be considered in order to achieve good path-following performance. Additionally, constraints on the dynamics of the unmanned helicopter—such as confidence regions of the software-components, requirements of the mission and the payload, or aerodynamic and structural limitations—have to be considered. Typical examples for such requirements are smoothness of the trajectories, a certain flight velocity, or limits of the acceleration.

Fig. 1: Unmanned helicopter ARTIS of DLR (German Aerospace Center).

In [9], [10] an optimization-based path following approach is proposed, which generates feedforward inputs to the underlying adaptive flight control system. An offline dynamic optimization uses the reference model of the flight control system as an approximation of the closed-loop dynamics. The resulting input signals are stored in a database and used during flight. Theses inputs steer the helicopter along nonlinear paths. However, if disturbances occur in flight that have not been considered during the optimization process, the helicopter might deviate from the desired path.

Here, we extend the approach of [9], [10] by considering the attenuation of disturbances in a run-to-run fashion. This extension focuses on the translational motion of the helicopter and compensates for acceleration components caused by model inaccuracies and unknown but constant wind conditions. We propose estimation of the unknown disturbances with a Kalman Filter. We validate the proposed approach by means of simulation results and preliminary flight test results. In order to demonstrate the quality and limitations of the proposed approach no outer position feedback for the adaptive flight controller is considered in simulation and flight tests.

This paper is structured as follows: In Section II the problem of iteratively improving the path-following performance in the context of adaptive flight control is introduced. In Section III-A, we introduce the optimal control formulation

[1]Johann Dauer and Sven Lorenz are with the Department of Unmanned Aircraft, Institute of Flight Systems, DLR (German Aerospace Center), 38108 Braunschweig, Germany. johann.dauer@dlr.de, sven.lorenz@dlr.de

[2]Timm Faulwasser is with the Institute for Applied Informatics, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany and the Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, Switzerland. timm.faulwasser@kit.edu

used to compute the feedforward commands. Section III-B recalls important aspects of the underlying adaptive flight control system, while in Section III-C the disturbance estimation and the proposed algorithm for run-to-run rejection are presented. Finally, we show simulation and flight test results in Section IV, respectively, Section V.

II. PROBLEM FORMULATION

Next, we introduce the problem of this paper. As it will be seen shortly, three different dynamic systems have to be considered: the real dynamics of the helicopter, its known approximated model, and the reference model of the flight control system. These three systems are closely linked by the flight control concept and motivate the way the disturbance estimator will be designed later on.

The dynamics of the unmanned helicopter can generically be written as

$$\dot{\mathbf{x}}_p(t) = \mathbf{f}_p(\mathbf{x}_p(t), \mathbf{d}(t), \boldsymbol{\delta}_p(t))), \ \mathbf{x}_p(0) = \mathbf{x}_{p,0} \tag{1a}$$

$$\mathbf{y}_p(t) = \mathbf{h}_p(\mathbf{x}_p(t)), \tag{1b}$$

where $\mathbf{x}_p \in \mathbb{R}^{n_x}$ represent the states and $\boldsymbol{\delta}_p \in \mathbb{R}^{n_\delta}$ is the real actuator input of the helicopter. In practice, these dynamics are only partially known. The time-varying and unknown input $\mathbf{d} \in \mathbb{R}^{n_d}$ describes exogenous disturbances (e.g. measurement error on the wind conditions).

Here, we consider a path-following control problem, i.e., the output[1] $\mathbf{y}_p \in \mathbb{R}^{n_y}$ of the helicopter dynamics (1) shall track a geometric reference path. Similar to [11], [12], the considered path $\mathcal{P} \subset \mathbb{R}^4$ is an explicitly parametrized curve including the position $\mathbf{r} \in \mathbb{R}^3$ and the yaw-angle ψ. The path $\mathcal{P}$ is given by

$$\mathcal{P} = \left\{ \mathbf{y} \in \mathbb{R}^4 \,|\, \theta \in [\theta_0, \theta_1] \rightarrow \mathbf{y} = \underbrace{(\mathbf{r}(\theta), \psi(\theta))^T}_{=:\mathbf{p}(\theta)} \right\}, \tag{2}$$

where the scalar variable θ is called the path parameter and represents the reference position on the path. It is important to note that this reference description does not involve a reference timing. In general, $\theta(t)$ is time-dependent but its time evolution is not known a priori, cf. [12], [11]. In order to compute the time evolution $t \mapsto \theta(t)$, we define an additional dynamic system with input v, which will be used to control the time evolution of $\theta(t)$. A double integrator has proven sufficient for the purpose of this paper, thus $\ddot{\theta} = v$ and a corresponding state vector $\mathbf{z} := (\theta, \dot{\theta})^T$.

To achieve path following for the helicopter ARTIS, a complicated control scheme is used. A simplified version of this scheme is sketched in Figure 2. It involves three main components: First, the flight control loops, marked by the light shaded box, ensure flight stabilization. It receives path-specific feedforward inputs $\mathbf{u}_c$ computed by the optimization block, which is the second main component in mid tone grey. As mentioned, the helicopter dynamics (1) are not known exactly. Furthermore, wind might act as exogenous disturbance on the helicopter. Thus, the model employed in the optimization-block is subject to plant-model mismatch. Here, we propose to account for this disturbance in a run-to-run fashion by means of a disturbance estimation, which is the third main component shaded in dark grey. Next, we outline the main functionalities of these three main components.

In the inner loop, adaptive flight control techniques [7] are used to control the helicopter with limited model knowledge, whereby the feedback part combines classical PID with an adaptive compensator based on artificial neural networks with a single hidden layer.

In a feedforward part of the flight controller, a known (approximative) model of the plant is used

$$\dot{\hat{\mathbf{x}}}_p(t) = \hat{\mathbf{f}}_p(\hat{\mathbf{x}}_p(t), \boldsymbol{\delta}_p(t)), \ \hat{\mathbf{x}}_p(0) = \hat{\mathbf{x}}_{p,0} \tag{3a}$$

$$\hat{\mathbf{y}}_p(t) = \hat{\mathbf{h}}_p(\hat{\mathbf{x}}_p(t)), \tag{3b}$$

where $\hat{\mathbf{x}}_p \in \mathbb{R}^{\hat{n}_x} \subset \mathbb{R}^{n_x}$ and $\hat{\mathbf{y}}_p \in \mathbb{R}^{n_y}$. In our case, the feedforward is based on the dynamic inversion of a model in the hover domain [8]. This technique uses designed reference dynamics, which are called the reference model, to generate reference trajectories $\mathbf{x}_m$ for the feedforward and reference outputs $\mathbf{y}_m$ for tracking. The reference model is given by

$$\dot{\mathbf{x}}_m(t) = \mathbf{f}_m(\mathbf{x}_m(t), \mathbf{d}(t), \mathbf{u}_c(t)), \ \mathbf{x}_m(0) = \mathbf{x}_{m,0} \tag{4a}$$

$$\mathbf{y}_m(t) = \mathbf{h}_m(\mathbf{x}_m(t)), \tag{4b}$$

where $\hat{\mathbf{x}}_m \in \mathbb{R}^{\hat{n}_x}$ and $\hat{\mathbf{y}}_m \in \mathbb{R}^{n_y}$. It expresses the desired closed-loop dynamics of the helicopter in terms of generating reference trajectories $\mathbf{x}_m$ based on the controller commands $\mathbf{u}_c$. As it will be evident later, the reference model also depends on the feedback $\boldsymbol{\delta}_{fb}$ of the flight control error $\mathbf{e}$. This particular feedback is called pseudo control hedging (PCH) [7]. While the details of this approach are beyond the scope of this paper, it is important to note that the reference model depends indirectly on the disturbance $\mathbf{d}$ and on the plant-model mismatch, i.e., difference between $\mathbf{f}_p$ and $\hat{\mathbf{f}}_p$.

Here, we aim at run-to-run improvement of path following in a feedforward fashion, whereby the feedforward commands $\mathbf{u}_c$ are computed by the optimization block, cf. Figure 2. To this end, an equivalent disturbance has to be estimated

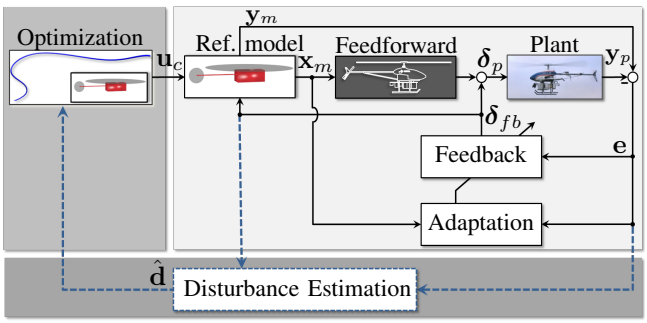


Fig. 2: General structure of the control architecture showing the focus of this paper with dashed lines.

[1]Note that this output does not represent the complete output used for the adaptive flight controller. It represents only the coordinates of the path definition. This simplification of notation is also included in Figure 2 where the cascaded nature of the adaptive flight controller is omitted for clarity.

which contains the time varying external disturbance and the modeling error $\boldsymbol{\Delta} = \dot{\mathbf{x}}_p - \hat{\dot{\mathbf{x}}}_p$:

$$\hat{\mathbf{d}}(t) \approx \boldsymbol{\Delta}(t) + \mathbf{d}(t) \tag{5}$$

Depending on its definition, the disturbance might also depend on the states of the plant. In this context, this state dependency is not resolved and rather represented by a time-wise signal only. Now, we are ready to formally define the problem of run-to-run disturbance rejection for feedforward path following.

Problem 1 (Disturbance Rejection for Path Following): Given the approximate helicopter dynamics (3), an adaptive flight controller with the nominal reference model (4), and a path $\mathcal{P}$. At each iteration k, estimate a disturbance $\hat{\mathbf{d}}^k$ and compute a corresponding transition time $\tau^k \in (0, \infty)$, a feedforward control $\mathbf{u}_c^k : [0, \tau] \to \mathcal{U}$, and a path evolution $\theta^k : [0, \tau^k] \to [\theta_0, \theta_1]$, which achieve the following:

i) *Reduction of the path deviation:* The path error

$$E^k = \int_0^{\tau^{k+1}} \left\| \mathbf{h}_p(\mathbf{x}^k(t)) - \mathbf{p}(\theta^k(t)) \right\|^2 dt \leq E^{k-1}$$

is reduced between flight trails $(k-1)$ and k.

ii) *Forward motion:* The system (4) moves in finite time τ^k from $\mathbf{h}_p(\mathbf{x}_0^k)$ in forward direction to the final point θ_1, i.e., for all $t \in [0, \tau^k] : \dot{\theta}^k(t) \geq 0$ and $\mathbf{y}_m(\tau^k) = \mathbf{p}(\theta_1)$.

iii) *Constraint satisfaction:* The input and state constraints of (4) are satisfied for all times, i.e., for all $t \in [0, \tau^k] : \mathbf{u}_c^k(t) \in \mathcal{U}$ and $\mathbf{x}_m^k(t, x_{m,0}^k; \mathbf{u}_c^k(\cdot)) \in \mathcal{X}$.

III. ITERATIVE PATH FOLLOWING

Next, we discuss all three main components of Figure 2 starting with the optimization based command generation. Afterwards some important aspects of the flight control system are outlined. Finally, the disturbance estimation is introduced, including the resulting algorithm for the iterative path-tracking improvement.

A. Feedforward Path Following

In order to compute path-specific feedforward inputs u_c, we rely on ideas from model predictive path-following control [11], [12]. Similar to our previous work [10] we formulate an optimal control problem (OCP), which is solved in a receding horizon fashion. That is, instead of solving the OCP over the complete horizon τ, we solve a sequence of OCPs over a shorter optimization horizon t_I, which are shifted forward along the path until its end is reached. This approach speeds up the computation but comes at the cost of approximating the optimal solution.

The OCP minimizes an objective by varying the decision variables $\mathbf{u} = (\mathbf{u}_\omega^T, u_t, v)^T$, containing the flight control variables introduced later and the path parameter input v:

$$\begin{aligned} J\left(\mathbf{x}_0, \mathbf{u}(\cdot)\right) = \int_{t_i}^{t_i+t_I} & \left\| (\mathbf{e}_\mathcal{P}^T(t), \dot{\mathbf{e}}_\mathcal{P}^T(t))^T \right\|_{\mathbf{Q}_e}^2 \\ & + \left\| \mathbf{z}(t) - \mathbf{z}_r(t) \right\|_{\mathbf{Q}_z}^2 + \left\| (\mathbf{u}^T(t), \dot{a}_t(t))^T \right\|_{\mathbf{R}}^2 dt \end{aligned}$$

Here, $\mathbf{e}_\mathcal{P}$ refers to a vector representing the path following error. The matrices $\mathbf{Q}_e, \mathbf{Q}_z, \mathbf{R}$ contain weighting parameters. The first term in the cost functional penalizes path deviation. Introducing the time derivative of the path following error to the objective avoids solutions where the resulting position trajectory oscillates around the desired path. The second term accounts for the deviation from the desired reference behavior, and the third term introduces a regularization, $\mathbf{u}$ and the thrust a_t, to smooth the optimal trajectories.

The overall OCP reads as follows:

$$\underset{\mathbf{u}^k(\cdot)}{\text{minimize}} \quad J\left(\mathbf{x}_0^k, \mathbf{u}^k(\cdot)\right) \tag{6a}$$

subject to the dynamics

$$\dot{\mathbf{x}}_m(t) = \mathbf{f}_m(\mathbf{x}_m(t), \hat{\mathbf{d}}^k(\theta(t)), \mathbf{u}(t)), \quad \mathbf{x}_m(0) = \mathbf{x}_{m,0} \tag{6b}$$

$$\dot{\mathbf{z}}(t) = \mathbf{l}(\mathbf{z}(t), v(t)), \quad \mathbf{z}(0) = (\theta_0, 0)^T \tag{6c}$$

$$\dot{\mathbf{e}}_\mathcal{P}(t) = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \mathbf{f}_m(\mathbf{x}_m(t), \hat{\mathbf{d}}^k(\theta(t)), \mathbf{u}(t)) - \frac{\partial \mathbf{p}}{\partial z_1} z_2(t). \tag{6d}$$

The dynamics (6b) are the reference model of the underlying adaptive flight stabilization (4), cf. Section III-B. The term $\mathbf{l}(\cdot)$ is the double integrator of the path parameter dynamics. The disturbance term is replaced by its estimate $\hat{\mathbf{d}}^k$. Furthermore, constraints on states and inputs, as well as additional constraints are considered

$$\forall t \in [t_i, t_i + t_I] : \mathbf{x}_m(t) \in \mathcal{X}, \quad \mathbf{u}^k(t) \in \mathcal{U} \tag{6e}$$

$$\forall t \in [t_i, t_i + t_I] : \dot{\theta}(t) \geq 0 \tag{6f}$$

$$\dot{\theta}(\tau) = 0. \tag{6g}$$

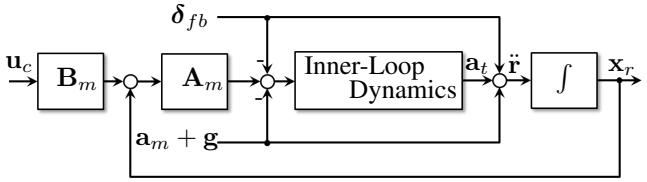
These constraints correspond to the controller's validated state-space region. The remaining constraints ensure that the movement along the path is forward only and the helicopter stops at the end of the path using τ as the time when the helicopter reaches the path's end.

It is worth noting that the disturbance estimate $\hat{\mathbf{d}}$ enters the OCP. We explain in Section III-C how the estimate $\hat{\mathbf{d}}$ is obtained.

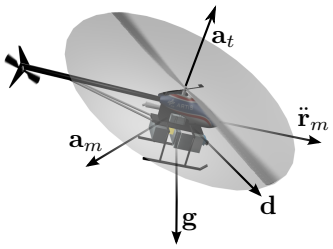
B. Model Based Adaptive Control

This section motivates that from perspective of the reference model the disturbance is in fact the feedback used for PCH. Furthermore, the outer loop reference model is presented in more detail to enable understanding the mechanics of the disturbance estimate.

Flight control of ARTIS, like most helicopter flight control systems, is based on a cascaded approach. The inner dynamics contain the rigid body rotational kinematics, the dynamics of the main and tail rotor and the engine dynamics. It also contains saturation of actuators, thrust and other known dynamical limits. From perspective of the outer loop these dynamics can be considered as an actuator. The outer-loop dynamics are the translational motion of the helicopter. Each loop has its own reference model. In this paper for the disturbance compensation we focus on the outer-loop, while the inner-loop dynamics are considered undisturbed. The remaining tracking error caused by omitting the disturbance for the inner-loops for a wind disturbance example will be shown in the Section IV.



(a) Simplified outer-loop reference model including PCH.



(b) Illustration of the signals.

Fig. 3: The translational reference model.

The aspect of the translation reference model important for this paper is shown in Figure 3. The complete architecture can be found in [8]. The dynamics of the reference model are designed using the matrices $\mathbf{A}_m$ and $\mathbf{B}_m$. Note that while these matrices might indicate a linear reference model, the nonlinearity is introduces by the acceleration componenten $\mathbf{a}_t$ creating an overall nonlinear reference model $\mathbf{f}_m$. The feedback signal $\boldsymbol{\delta}_{fb}$ contains linear velocity error feedback in form of a PI controller but also the adaptive compensator. The structure shown in the figure is a special form of PCH as introduced by [8] such that the PCH does not require state measurements of the actual plant, but uses the reference model states instead. Two benefits are involved with this structure: first, the reference model will not diverge from the plant due to inner-loop saturations; and second, the actuator dynamics are removed from the error states and thus hidden from the adaptation, an effect referred to as hedging [7].

Figure 3b illustrates of the principal behind the reference model signals. The acceleration of the helicopter $\ddot{\mathbf{r}}_m$ consists of different components, including gravity $\mathbf{g}$ and the acceleration caused by the main rotor thrust $\mathbf{a}_t$. A property of helicopters in the depicted classical rotor configuration is a significant amount of aerodynamic and dynamic coupling. The known part of this acceleration $\mathbf{a}_m$ is based on the same model used for feedforward of Figure 2. All remaining acceleration components caused by model uncertainty and inaccurate wind information can be considered as a disturbance acceleration $\mathbf{d}$. From this discussion it is evident that the feedback signal $\boldsymbol{\delta}_{fb}$ is the only feedback into the reference model that can change the reference trajectory compared to the optimal ones. Furthermore, this feedback influences the reference model similar to a disturbance acceleration, which indicates a close relation between the equivalent disturbance and the PCH feedback.

The complete reference dynamics can be found in [9]. These dynamics involve significant nonlinearities that make the optimization computationally difficult. For this reason, a reformulation is proposed in [10], which exploits the facts that parts of the dynamics are invertible. This way, artificial input variables can be introduced becoming the decision variables for optimization without changing the reference model itself. Using the optimized state trajectories of the reformulated problem, the original commands $\mathbf{u}_c \in \mathbb{R}^4$ can then be calculated in an algebraic fashion. The new inputs are the second derivatives of the rotational rates $\mathbf{u}_\omega \in \mathbb{R}^3$, and the second derivative of the mass normalized thrust $u_t \in \mathbb{R}$. These reformulated internal dynamics can be written as

$$\ddot{\boldsymbol{\omega}}(t) = \mathbf{u}_\omega(t) \tag{7a}$$

$$\ddot{a}_t(t) = u_t(t) \tag{7b}$$

$$\dot{\underline{q}}(t) = \frac{1}{2}\underline{q}(t) \otimes \underline{\omega}(t), \tag{7c}$$

where $\underline{\omega}$ represents the pure imaginary quaternion of ω and $\otimes$ the quaternion product. The inner-loop states are the attitude $\underline{q} \in \mathbb{R}^4$ represented by a quaternion, body-fixed rotation rates $\boldsymbol{\omega} \in \mathbb{R}^3$ and its first and second derivative $\dot{\boldsymbol{\omega}}, \ddot{\boldsymbol{\omega}}$. The engine is modeled using a second-order system having the thrust normalized by the helicopter's mass $a_t \in \mathbb{R}$ as well as its derivative $\dot{a}_t$ as states.

For the outer-loop, the abbreviation $\mathrm{R}[\mathbf{a}]$ is used to represent the rotation of a vector $\mathbf{a}$, which is achieved by using the formula for quaternion rotation $\underline{a}^R = \underline{q} \otimes \underline{a} \otimes \bar{\underline{q}}$ and selecting the imaginary part of the $\underline{a}^R$. Using the assumption that the thrust is aligned with the main rotor shaft, the translation dynamics already extended by an estimate of the disturbance $\hat{\mathbf{d}}$ are given by

$$\ddot{\mathbf{r}}_m(t) = R\left[\begin{pmatrix} 0 \\ 0 \\ -a_t(t) \end{pmatrix}\right] + \mathbf{g} + \mathbf{a}_m(\mathbf{x}_m(t)) + \hat{\mathbf{d}}(t). \tag{8}$$

The overall state vector of the reference model is then compiled to $\mathbf{x}_m = (\dot{\mathbf{r}}^T, \underline{q}^T, \boldsymbol{\omega}^T, \dot{\boldsymbol{\omega}}^T, a_t, \dot{a}_t)^T \in \mathbb{R}^{15}$.

C. Disturbance Estimation

The path tracking error, which is the difference between the desired and the actual position of the helicopter, contains two components. First, the feedback $\boldsymbol{\delta}_{fb}$ has an influence on the signals of the reference model, such that the reference signals do not match the solution of the OCP exactly. It has been shown in the previous section that this feedback acts like a disturbance acceleration from the perspective of the reference model. Second, there is always a difference between the reference model and the state of the plant called the reference tracking error $\mathbf{e}$. Its velocity component $\mathbf{e}_{\dot{r}} = \dot{\mathbf{r}}_m - \dot{\mathbf{r}}_p$ is used here. The deviation is caused by model inaccuracies of the feedforward part as well as external disturbances like unmeasured wind. In the following this deviation is considered to be caused by a disturbance acceleration $\mathbf{a}_d := \dot{\mathbf{e}}_{\dot{r}}$, such that

$$\ddot{\mathbf{r}}_p(t) = \ddot{\mathbf{r}}_m(t) + \mathbf{a}_d(t). \tag{9}$$

Unfortunately, classical helicopter configurations are subject to a significant amount of vibration. As the small

scale helicopters considered here are limited in payload capabilities, the inertial measurement units (IMU) are thus limited in quality by weight as well as achievable sampling rates. These two aspects render the direct measurement of this disturbance infeasible. It hence has to be estimated.

A linear Kalman Filter (KF) is proposed as estimator in this work. Nevertheless, there is a great variety of alternatives to a KF based approach. One aspect shall be briefly discussed. The KF used here estimates the disturbance forward in time. In contrast, approaches like optimal smoothing [13] also include future measurements. This backward smoothing enhances the estimate. However, as it is aimed to extend this work in respect to an onboard closed-loop formulation at some point, the backward smoothing cannot be applied.

The reference acceleration of (9) is recorded during flight and can be considered as input for the estimation $\mathbf{u}_{est} = \ddot{\mathbf{r}}_m$. The states of the estimator are augmented by $\mathbf{a}_d$, such that $\mathbf{x}_{est} = (\dot{\mathbf{r}}_p^T, \mathbf{a}_d)^T$. The linear process for estimation of a slowly varying $\mathbf{a}_d$ can now be written as

$$\dot{\mathbf{x}}_{est}(t) = \begin{pmatrix} \mathbf{0} & \mathbf{I}_3 \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \mathbf{x}_{est}(t) + \begin{pmatrix} \mathbf{I}_3 \\ \mathbf{0} \end{pmatrix} \mathbf{u}_{est}(t) + \mathbf{n}_x, \quad (10)$$

where $\mathbf{n}_x$ represents the process noise and $\mathbf{I}_3$ the 3×3 identity matrix. The correction is performed using the velocity measurements and the measurement model

$$\mathbf{y}(t) = \begin{pmatrix} \mathbf{I}_3 \\ \mathbf{0} \end{pmatrix}^T \mathbf{x}_{est} + \mathbf{n}_y, \quad (11)$$

assuming the measurement noise $\mathbf{n}_y$.

The free parameters, namely the covariance matrices of process and measurement noise are determined using a derivative free, global optimization process. The optimization of these parameters is performed once using flight test data. The objective function of this optimization contains an estimation error defined as

$$\mathbf{e}_{est}(t) = \dot{\mathbf{r}}_p(t) - \dot{\mathbf{r}}_p(t_0) - \int_{t_0}^{t} \ddot{\mathbf{r}}_m(\tau) + \mathbf{a}_d(\tau) d\tau, \quad (12)$$

which is in essence the velocity difference of the real plant to the reference model corrected by the disturbance estimate. The objective function additionally contains standard deviation of the estimation error to penalize the noisiness of the estimate.

The complete disturbance to be considered in the reference model and optimization for path tracking at iteration k is given by

$$\hat{\mathbf{d}}^k(t) = \hat{\mathbf{d}}^{k-1}(t) + \boldsymbol{\delta}_{fb}(t) + \mathbf{a}_d(t). \quad (13)$$

Different possibilities have been investigated how this disturbance estimate can be considered within the receding horizon optimization. To this end, disturbances have been generated in a full system simulation using a wind emulation and an aerodynamic wind influence model. The results of the OCP under ideal wind knowledge have then be compared to different approaches where wind is assumed unknown and the above procedure to estimate disturbance acceleration has been applied. It has been found that strong simplifications as assuming a constant mean value over the receding horizon do significantly change the generated optimal trajectories.

The trend of the disturbance is hence important as well. To this end, the following approach has proven successful: Within the optimization horizon $t \in [t_i, t_i + t_I]$, the disturbance estimate $\hat{\mathbf{d}}(t)$ is locally represented by a sixth order polynomial. As the path parameter is strictly increasing due to the constraint in (6f) it is possible to replace the time-wise representation of $\hat{\mathbf{d}}(t)$ using the path parameter θ

$$\hat{\mathbf{d}}^k(\theta) \approx \sum_{j=0}^{6} c_j^k \theta^j. \quad (14)$$

In other words, for the re-optimization it is assumed that the disturbance is repeatable at each position of the path rather than a certain flight time. The parameters of the polynomial are determined using a regression over the optimization horizon with some time margin t_m for smoothness of the polynomial at the interval boarder, $t \in [t_i - t_m, t_i + t_I + t_m]$. This representation of the disturbance can be considered within the optimization as it simply extends the reference model as shown in (8). The optimal time-wise evolution of the path parameter $\theta^k(t)$ at iteration k is used to calculate an expected evolution of the disturbance. This disturbance signal is stored together with the input history $\mathbf{u}_c^k(t)$ in the maneuver database and used in the feedforward during flight as shown in Figure 3a by extending the signal $(\mathbf{a}_m + \mathbf{g})$ by the disturbance estimate $\hat{\mathbf{d}}^k$.

The proposed scheme for run-to-run disturbance rejection in feedforward path following can now be summarized. It involves three main steps:

0 : Initialize with $k = 0$, path $\mathcal{P}$, and disturbance estimate $\hat{\mathbf{d}}^k(t) = 0$.
1 : Optimization-based computation of feedforward commands $\mathbf{u}_c^k(t)$ for the flight control system based on past disturbance data.
2 : Application of $\mathbf{u}_c^k(t)$ to flight control system during the flight along the path.
3 : Estimation of the disturbances $\hat{\mathbf{d}}^k(t)$ based on recorded flight data.
4 : $k \to k + 1$. If tracking error is insufficient, goto 1.

One limitation of the proposed approach is that, by design, disturbances cannot be compensated if they do not reoccur systematically during each iteration like (e.g. wind gusts).

IV. SIMULATION RESULTS

Next, we present results for an example path using a system simulation of the ARTIS helicopter. This simulation includes all software components present during flight. Among others, it contains the flight control system outlined here, sensor fusion and mission management. The identified hover domain model is used as simulated dynamics of the helicopter and sensor properties are emulated as well. Uniform static wind is introduced, which generates forces and moments affecting the helicopter's rigid body motion using

equivalent aerodynamic surfaces and lever arms. The optimization has been implemented using the code generation of the ACADO Toolkit, cf. [14]. Details on the implementation can be found in [10].

In the example, the reference path is a clover leaf with three leafs viewed from top. Additionally a height profile is generated. The heading ψ is defined in forward direction such that the helicopter is in side-slip free motion. The position component of the path is defined as

$$\mathbf{r} = \begin{pmatrix} R\cos\left(3\theta - \frac{3}{2}\pi\right)\cos(\theta) \\ R\cos\left(3\theta - \frac{3}{2}\pi\right)\sin(\theta) \\ H\sin(4\theta) \end{pmatrix}, \quad \begin{aligned} R &= 25\,\mathrm{m}, \\ H &= 3\,\mathrm{m}. \end{aligned} \tag{15}$$

The flight performance of one example of these non-stationary, non-trim paths is shown in Figure 4, where the position from different views and the absolute path error $\|\mathbf{e}_p\|$ is presented. The desired flight velocity is 8 m/s, while a wind of 10 m/s has been added, which is assumed to be entirely unknown. The helicopter starts at $\mathbf{r}_0 = (x_0, y_0, z_0)^T$ from hover conditions and begins flying in southward direction. After completion of the path, the helicopter stops at the initial position in hover condition. The flight is performed several times using the iterative process outlined above. All flights have been performed without position feedback, which means the position measurement for the flight controller has been disabled in order to highlight inaccuracies of the presented method.

It can be seen that the path deviation $\|\mathbf{e}_p\|$ of the first flight is significant and exceeds 10 m. Already with the second iteration, the path error is reduced to around 1-2 m. It is only slightly improved by additional iterations.

The remaining flight path errors are caused by two sources: first, the wind model also introduces disturbances of the rotational dynamics caused by a lever arm between the neutral point of the equivalent aerodynamic surfaces and the center of gravity of the helicopter. These disturbances are not estimated and considered within the optimization. Second, the simulation also contains emulation of delays present in the software and hardware interfaces also present in real flight.

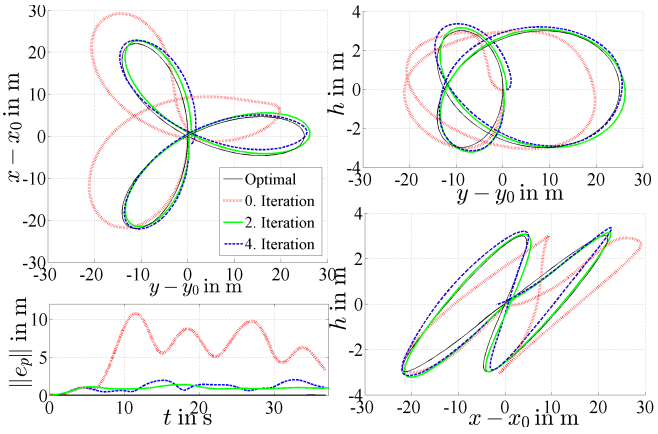


Fig. 4: Simulation results of a clover leaf reference path with 10 m/s wind disturbance flown without position feedback.

V. PRELIMINARY FLIGHT TEST RESULTS

In September 2014, flight tests of the path presented in the simulation chapter have been conducted. Figure 5 shows the flight from a ground camera perspective. In white, the reference path is shown, which is defined by (15). The actual path flown is drawn in green. The flight path error remains in the range of 2 m in the presented case. Wind conditions on the trial day were gentle breeze (number 3 on the Beaufort scale). Equivalent to the simulation no position feedback was used to determine the limits of the presented iterative optimization-based approach.

In contrast to the simulation, the wind conditions were not stationary. In order to decrease the effect of time varying winds like gusts, for each iteration k, the flight was repeated three times without reoptimization. Each flight results in a separate time history of the disturbance. All flights involved in one iteration were used to find a mean disturbance value for each time step, that is

$$\bar{\mathbf{d}}(t) := \frac{1}{3}\sum_{i=1}^{3} \hat{\mathbf{d}}_i(t). \tag{16}$$

Figure 6 shows quantitative results of the performed flight test. From top to bottom the iterations are shown starting without disturbance compensation up to the second iteration. Iterating further did not improve the tracking performance. The left side shows the path error caused by a deviation of the signals of the reference model. As can be seen these values decrease with each iteration. The right side shows the actual helicopter position in comparison with the desired path. Here, the influence of the time varying wind is clearly visible in the different curves. Nevertheless, an improvement is still visible from around 5 m down to around 2.5 m. Although these results strongly suggest that the proposed approach does improve the tracking performance, a deep flight test analysis is necessary to discuss all aspects of the validation of the presented algorithm. Lessons learned from these experiments are that it would be beneficial to test the approach with even stronger wind. Additionally, the amount of wind variation plays a significant role. This variation could be reduced either

Fig. 5: Ground camera perspective of the flight test of the clover leaf path flown without position feedback.

by flying in higher altitudes or by introducing a virtual and thus ideally known disturbance.

Figure 7 shows the Frobenius Norm of the matrix storing all weights of the adaptive compensator, i.e. the sum of all squared elements. With each iteration, this norm decreases significantly. In other words, the model error that the adaptive compensator has to adapt to decreases as well. This decrease indicates that the unknown part of the disturbance also decreases and both, model discrepancies and external disturbances, are captured adequately by the proposed approach.

VI. CONCLUSIONS

This paper presented a methodology for combining feedforward path following with run-to-run disturbance estimation and rejection. It has been shown that the adaptive feedback of a reference model-based approach can be used as a first disturbance estimate. This estimate has to be extended using the velocity control error. A Kalman Filter can be used to tackle this second disturbance component. Simulations have shown significant improvement of the path tracking for an unmanned helicopter estimating outer-loop disturbances in time-constant wind fields. Preliminary analyses of flight test results confirm the tracking improvement to the extent possible under the wind conditions of the flight test.

Future work will include a detailed analysis of the conducted flight tests. It remains for investigation to assess potential additional improvement by extending the estimation for inner-loop disturbances.

ACKNOWLEDGMENT

The authors would like to thank the team of the unmanned aircraft department at DLR. Special thanks go to Prof. Dominique Bonvin who enabled a guest stay at the Laboratoire d'Automatique at EPFL where parts of the results have been derived.

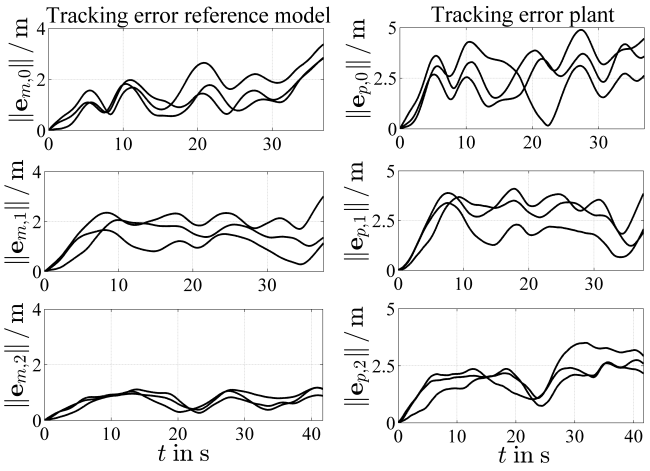


Fig. 6: Path tracking errors of the three flight test iterations. The left shows the position error of the reference model while the right displays the helicopter's total position error.

REFERENCES

[1] G. Conte, S. Duranti, and T. Merz, "Dynamic 3d-path following for an autonomous helicopter," in *IFAC /EURON Symposium on Intelligent Autonomous Vehicles*, Lisbon, Portugal, July 2004.

[2] M. Takahashi, G. Schulein, and M. Whalley, "Flight control law design and development for an autonomous rotorcraft," in *Proceedings of 64th American Helicopter Society International Annual Forum*, vol. 3, Montreal, Canada, May 2008, pp. 1768–1787.

[3] S. Lorenz and F. Adolf, "A decoupled approach for trajectory generation for an unmanned rotorcraft," in *Advances in Aerospace Guidance, Navigation and Control*, F. Holzapfel and S. Theil, Eds. Berlin: Springer, 2011, pp. 3–14.

[4] F. Adolf and F. Andert, "Rapid multi-query path planning for a vertical take-off and landing unmanned aerial vehicle," *Journal of Aerospace Computing Information and Communication*, vol. 8, pp. 310–327, Nov. 2011.

[5] N. Dadkhah and B. Mettler, "Survey of Motion Planning Literature in the Presence of Uncertainty: Considerations for UAV Guidance," *Journal of Intelligent & Robotic Systems*, vol. 65, pp. 233–246, 2012.

[6] G. Q. Huang, Y. P. Lu, and Y. Nan, "A survey of numerical algorithms for trajectory optimization of flight vehicles," *Science China Technological Sciences*, vol. 55, no. 9, pp. 2538–2560, 2012.

[7] E. N. Johnson and S. K. Kannan, "Adaptive Trajectory Control for Autonomous Helicopters," *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 3, pp. 524–538, 2005.

[8] S. Lorenz, "Open-Loop Reference System for Nonlinear Control Applied to Unmanned Helicopters," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 1, pp. 259–269, 2012.

[9] J. Dauer, T. Faulwasser, S. Lorenz, and R. Findeisen, "Optimization-based feedforward path following for model reference adaptive control of an unmanned helicopter," in *Proceedings of AIAA Guidance, Navigation, and Control Conference*, Boston, MA, Aug. 2013.

[10] J. Dauer, T. Faulwasser, and S. Lorenz, "Computational aspects of optimization-based path following of an unmanned helicopter," in *Proceedings of the 18th European Conference on Mathematics for the Industry*, Taormina, Italy, Aug. 2014.

[11] T. Faulwasser and R. Findeisen, "Nonlinear model predictive control for constrained output path following," *IEEE Transactions on Automatic Control. Accepted for publication*, 2015, arXiv:1502.02468.

[12] T. Faulwasser, *Optimization-based solutions to constrained trajectory-tracking and path-following problems*, ser. Contributions in Systems Theory and Automatic Control, O. U. Magdeburg and I. für Automatisierungstechnik, Eds. Aachen, Germany: Shaker, 2013.

[13] A. Gelb, J. F. Kasper, R. A. Nash, C. F. Price, and A. A. Sutherland, *Applied Optimal Estimation*, A. Gelb, Ed. Cambridge: The M.I.T. Press, 2001.

[14] B. Houska, H. Ferreau, and M. Diehl, "ACADO toolkit – an open-source framework for automatic control and dynamic optimization," *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011.

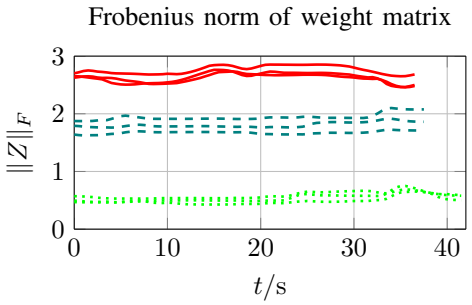


Fig. 7: Frobenius Norm of the weight matrix: The red solid lines show the three flights without disturbance compensation, the dark green and dashed lines the first iteration and light green and dotted lines the second iteration.