An open benchmark for distributed formation flight control of Fixed-Wing Unmanned Aircraft Systems

Jan Bolting * **Martin Stolle** ** **Soheib Fergani** *
Jean-Marc Biannic ** **François Defaÿ** *

* *Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), 31055 Toulouse, France (e-mail: jan.bolting@isae.fr, soheib.fergani@isae.fr, francois.defay@isae.fr)*
** *Office National d'Études et de Recherches Aérospatiales (ONERA), 31055 Toulouse, France (e-mail: jean-marc.biannic@onera.fr, martin.stolle@onera.fr)*

Abstract: The capability of autonomous formation flight has the potential to significantly enhance the utility and efficiency of small low-cost Unmanned Aerial Systems (UAS). Formations of small, inexpensive fixed-wing UAS allow for the sharing of remote sensing functionality, mission-level redundancy and range enhancements due to aerodynamic interactions widely exploited by migratory birds. This article presents a benchmark problem for scalable distributed flight control of formations of UAS with only local relative state information, one of the open problems in this field as of today. The benchmark is openly available and comprises a nonlinear six degrees of freedom dynamics model of an electric glider UAS. In this article we furthermore introduce a nominal guidance frame that does not require state information of other UAS and point out a fundamental issue related to wake vortex tracking during formation maneuvers. A set of LQ baseline controllers that are part of the benchmark is presented along with simulation results.

Keywords: UAS, Formation flight, benchmark

1. INTRODUCTION

Migratory birds routinely perform flights of several thousand kilometers. They do so using efficient formation patterns that allow them to benefit from the induced airflow generated by the birds in front of them. Achievable energy savings of the order of 10 % are reported in the literature, Weimerskirch et al. (2001). For some time there have been efforts to imitate this behavior to enhance the range of aircraft. The NASA AFF program has demonstrated the feasibility of this approach for manned fighter aircraft in a 2-aircraft configuration. Maximum fuel flow reductions of 18 % for the follower are reported, Jake et al. (2003).

In terms of implementation effort, this strategy is a very attractive one, since no structural changes have to be made to the aircraft to gain considerable fuel savings (otherwise only achievable by considerable advancements in e.g. the fuel efficiency of the engines), only by adapting the flight control software and possibly adding sensors for relative localization.

In order to bring this capability to small unmanned aircraft, significant advances need to be made regarding relative guidance performance. Reported experimental results for 2-formations, Park (2004), and 3-formations, Gu et al. (2006), feature relative guidance errors of the order of meters. However, both theoretical considerations Hummel (1982) and flight experiments Jake et al. (2003) suggest that the UAS needs to stay in a window of wingtip-wingtip separations $(\Delta x\ \Delta y\ \Delta z)^T$ roughly specified by

$$-0.2b < \Delta y' < -0.1b$$
$$-0.1b < \Delta z' < 0$$

to benefit from significant energy savings. The longitudinal separation Δx is less critical and can be several wingspans Jake et al. (2003). For small UAS, the spanwidth b is of the order of $1\,m$, resulting in guidance error requirements of the order of $0.1\,m$ laterally and vertically.

Testing autonomous formation flight control algorithms using inexpensive UAS also constitutes an interesting way for the commercial aviation industry to build confidence in and evaluate the safety implications new formation flight control systems before implementing them on manned aircraft.

In control, binary properties of a system such as stability or stability robustness - the system can either shown to be closed-loop stable over a given set of uncertainties or not - can usually be evaluated unambiguously for a given system. In contrast to that, control performance is tied to the problem that is used to evaluate a certain control approach. Benchmark problems play an essential role in every performance-driven field of research, as they allow to quantitatively compare different approaches.

As of today in the field of aircraft formation flight control there is no lack of contributions and promising approaches. However there appears to be a lack of reference problems to compare them and to make an informed judgment about their performance and implementability. This gap is it we are attempting to address with the presented benchmark. Our modeling approach is focused on the current main issues of distributed formation control - mesh stability with minimum shared information and scalable performance - and intentionally does not take into account secondary aspects such as sensor modeling and state estimation.

To be useful, a benchmark must be accessible to other researchers. Therefore, the proposed benchmark is complemented by an implementation as an openly accessible Matlab/Simulink library. A set of LQ baseline controllers comes with the benchmark to provide a basic closed loop solution more advanced approaches can be compared to. Furthermore, a cross-platform 3D visualization environment is part of the benchmark.

This work pulls together a number of contributions such as trajectory design techniques to provide a standard test case that addresses important challenges of distributed formation flight control as of today. The article is structured as follows: after elaborating on the role of information constraints on formation control in section 2, the aircraft model and the simulation environment are presented in section 3. Sections 4 and 5 deal with the benchmark trajectory and baseline control laws, respectively. In section 6 first simulation results are presented and section 7 provides some concluding remarks.

2. IMPACT OF INFORMATION CONSTRAINTS

A variety of approaches for formation flight control have been proposed in the past. A focus has been formation scalability, i.e. the capability to add members to a formation without adapting control laws and without violating performance bounds. This is expressed by the requirement of mesh stability. In mesh stable formations, position errors stay locally confined and are not amplified while propagating through the formation.

As a general trend, the control problem becomes more challenging in inverse proportion to the available feedback information. Reduced feedback information is an attractive property, since less sensors and less inter-vehicle communications are necessary. Centralized approaches, Nageli et al. (2014), pose no particular challenge, since standard control techniques can readily be applied, but require large amounts of inter-vehicle communication to obtain full state information.

Mature linear techniques exist, Shaw and Hedrick (2007), for mesh stable control with leader information, i.e. the relative position of each formation member w.r.t the leader aircraft is available for feedback.

The proposed benchmark addresses a more challenging scenario, distributed control with local state information, i.e. each member has only access to its own relative position and velocity w.r.t. to the preceding aircraft. There are fundamental results that linear interconnected systems with at least two integrators such as aircraft formations are always mesh instable if only local feedback information is available (Pant et al. (2001)), ruling out linear control techniques. Very few solutions based on nonlinear techniques such as Port-Hamiltonian systems, Knorn et al. (2014) and Sliding Mode control, Galzi and Shtessel (2006); Bolting et al. (2016), have been proposed so far to tackle this problem and no experimental results are known to the authors. Furthermore, the existing approaches have been applied to very simplified models, basically point-mass approximations, neglecting a large part of the complex dynamics and constraints of a fixed-wing UAS.

3. AIRCRAFT MODEL

3.1 Aerodynamics

There is an abundance of possibilities to model the aerodynamic coefficients of an aircraft, reaching from look-up tables to neural networks. Following the principle of minimum necessary complexity, here a classic, mostly linear in the states model of the aerodynamic coefficients has been adopted, Beard and McLain (2012). The aerodynamic force and moment coefficients are given by

$$C_D = C_{D_0} + (C_{L_0} + C_{L_\alpha}\alpha)^2 \frac{1}{\pi e \Lambda} + C_{D_{\delta_e}}\delta_e + C_{D_q} q \quad (1)$$

$$C_Y = C_{y_\beta}\beta + C_{Y_p} p + C_{Y_r} r + C_{Y_{\delta_a}}\delta_a + C_{Y_{\delta_r}}\delta_r \quad (2)$$

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + C_{L_q} q + C_{L_{\delta_e}}\delta_e \quad (3)$$

$$C_l = C_{l_\beta}\beta + C_{l_p} p + C_{l_r} r + C_{l_{\delta_a}}\delta_a + C_{l_{\delta_r}}\delta_r \quad (4)$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_q} q + C_{m_{\delta_e}}\delta_e \quad (5)$$

$$C_n = C_{n_\beta}\beta + C_{n_p} p + C_{n_r} r + C_{n_{\delta_a}}\delta_a + C_{n_{\delta_r}}\delta_r \quad (6)$$

Estimates of the coefficients have been obtained with the software AVL from a 3D model of a small electric glider airplane (Multiplex Cularis), see Stolle (2016).

3.2 Kinematics

Using 1-6, the aerodynamic forces and moments in the body frame are computed as

$$\mathbf{F} = \mathbf{R_{b \leftarrow s}} \bar{q} S \begin{pmatrix} C_D \\ C_Y \\ C_L \end{pmatrix} + \begin{pmatrix} T \\ 0 \\ 0 \end{pmatrix} \quad (7)$$

$$\mathbf{M} = \mathbf{R_{b \leftarrow s}} \begin{bmatrix} \bar{q}Sb & 0 & 0 \\ 0 & \bar{q}Sc & 0 \\ 0 & 0 & \bar{q}Sb \end{bmatrix} \begin{pmatrix} C_l \\ C_m \\ C_n \end{pmatrix} \quad (8)$$

with $\mathbf{R_{b \leftarrow s}}$ the rotation matrix between stability frame and body frame, the vector of forces $\mathbf{F}$ and the vector of moments $\mathbf{M}$, assuming that the thrust vector is aligned with the body frame x axis and coincides with the center of gravity. Translational accelerations in the body frame result to

$$\dot{\mathbf{v}} = \frac{1}{m}\mathbf{F} - (\omega \times \mathbf{v}) \quad (9)$$

where $\mathbf{v}$ is the translational velocity of the body frame w.r.t the inertial frame expressed in body frame axes, and ω^b is the angular velocity of the body frame w.r.t the inertial frame expressed in body frame axes. The rotation rate of the earth can be neglected in this particular case due to the short duration of the considered UAS flight maneuvers.

Angular accelerations about body frame axes follow to

$$\dot{\omega} = -\mathbf{J}^{-1}(\omega \times (\mathbf{J}\,\omega)) + \mathbf{J}^{-1}\mathbf{M} \quad (10)$$

with the inertia tensor $\mathbf{J}$. With these accelerations, standard techniques are used to propagate angular and translational velocities and positions over time to compute the aircraft's full state, which is given by

$$\mathbf{x} = \left(\mathbf{p}^T \; \mathbf{v}^T \; \mathbf{q}^T \; \omega^T \right)^T \tag{11}$$

with NED position $\mathbf{p}$, NED velocity $\mathbf{v}$, the attitude quaternion $\mathbf{q}$ and the rotation rate vector ω.

Exogenous perturbations The aircraft dynamics are subject to exogenous disturbances due to atmospheric free air turbulence. These are taken into account as additional system inputs, extending the input vector to

$$\mathbf{u} = \left(\delta_e \; \delta_a \; \delta_r \; \delta_{en} \; \mathbf{v}_w^T \; \omega_w^T\right)^T \tag{12}$$

3.3 Actuators and Engine

The control surface servo actuators have been identified in previous work, Ameho (2013), as second order systems

$$\begin{pmatrix} \dot{\delta}_i \\ \ddot{\delta}_i \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ -3948 & 100.5 \end{bmatrix} \begin{pmatrix} \delta_i \\ \dot{\delta}_i \end{pmatrix} + \begin{bmatrix} 0 \\ 3948 \end{bmatrix} \delta_{i,c} \tag{13}$$

This model is augmented by adding realistic saturations on deflection and deflection rate

$$|\delta_i| \leq \frac{\pi}{4} \tag{14}$$

$$|\dot{\delta}_i| \leq 5.8 \frac{rad}{s} \tag{15}$$

The engine is modeled as first order system

$$\dot{\delta}_{en} = \tau_{en}^{-1}(\delta_{en,c} - \delta_{en}) \tag{16}$$

The thrust T is then computed using a simple model based on momentum theory (Beard and McLain (2012)) as

$$T = \frac{\rho}{2} S_{en} C_{en} ((k_{en}\delta_{en})^2 - V_a^2) \tag{17}$$

with the propeller disc surface S_{en}, the thrust constant C_{en}, the engine constant k_{en} and the airspeed V_a.

3.4 Wind

The wind is assumed to be composed of a constant ambient part and a time-varying stochastic part, inducing velocities

$$\mathbf{v}_w(t) = \mathbf{v}_{w,a} + \mathbf{v}_{w,s}(t) \tag{18}$$

and induced angular rates $\omega_w(t)$. Wind time series are generated according to the Dryden turbulence spectrum.

4. BENCHMARK TRAJECTORY

The benchmark includes a 3D trajectory composed of C_2-continuous cubic Bézier splines, based on the work presented by Yang and Sukkarieh (2008). The trajectory features a linear part to evaluate cruise flight performance, followed by an approximately helical part to evaluate climb and turn maneuvers. See Looye (2011) for the compelling properties of helical trajectories for the evaluation of automatic flight control laws.

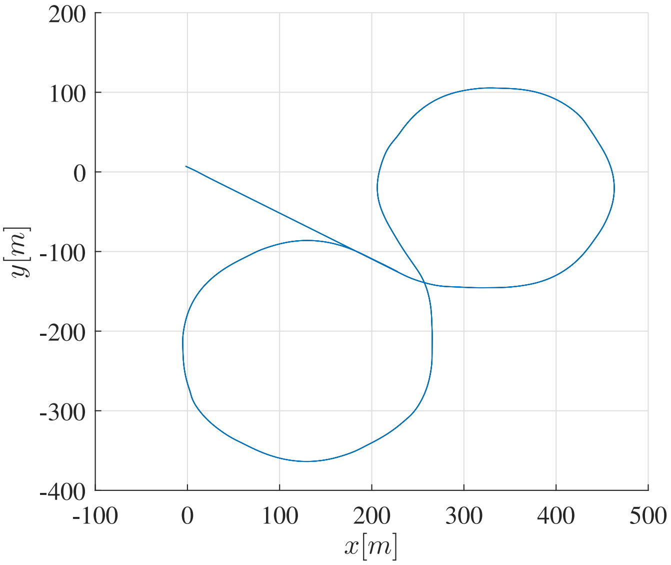

Fig. 1. Top view of benchmark trajectory

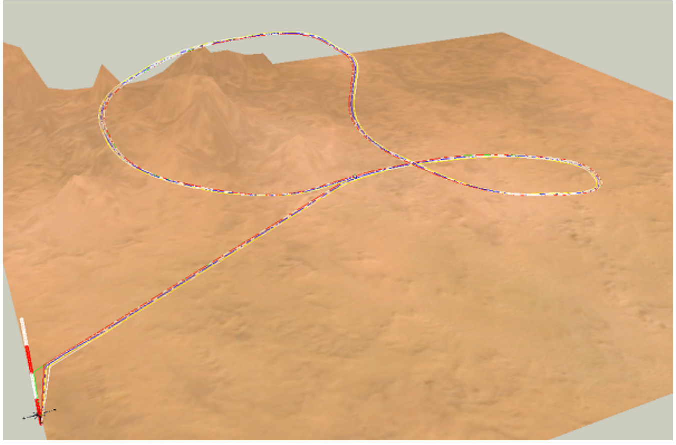

Fig. 2. Isoview of benchmark trajectory in UAV3D

5. LQ BASELINE CONTROL LAWS

To provide a standard to compare new control approaches, a set of baseline LQ control laws is provided as part of the benchmark. For controller synthesis, the respective subsystems are extracted from the full linear model that results from linearizing the nonlinear aircraft model at cruise flight trim conditions. The structure of the control laws is based on the usual assumptions of weak coupling between lateral and longitudinal dynamics and timescale separation of attitude dynamics and translational dynamics.
The synthesis procedure is the same for all four subsystems. It is shortly recalled here, and the specific design systems for every controller are given in the respective subsections.

The time-invariant linear quadratic regulator (LQR) minimizes a quadratic integral performance index

$$J = \int_{t0}^{\infty} (\mathbf{x}(t)^T \mathbf{Q} \mathbf{x}(t) + \mathbf{u}(t)^T \mathbf{R} \mathbf{u}(t)) dt \tag{19}$$

for a given linear system

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{20}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^m$ are the state and input vectors respectively and $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{m\times m}$ are positive definite design matrices. To track a given reference state $\mathbf{x}_c$, the tracking error dynamics are derived (dropping in the following the explicit dependence on time) as

$$\begin{aligned}\dot{\mathbf{x}}_e &= \dot{\mathbf{x}} - \dot{\mathbf{x}}_c && (21)\\ &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} - \dot{\mathbf{x}}_c && (22)\end{aligned}$$

The reference state is either constant, as in the static relative position tracking case, or unknown, since the reference state (e.g. the commanded bank angle) is generated by the outer loop controller. Its derivative is thus treated as unknown disturbance and set to zero for controller design, leading to the state error dynamics that are identical with the state dynamics.

$$\dot{\mathbf{x}}_e = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{23}$$

Tracking is thus approached by a simple change of state coordinates

$$\dot{\mathbf{x}}_e = \mathbf{x} - \mathbf{x}_c \tag{24}$$

To cope with steady state tracking errors, integral action is added by augmenting the system (24) with integral states

$$\begin{pmatrix}\dot{\mathbf{x}}_e\\ \dot{\mathbf{x}}_i\end{pmatrix} = \begin{bmatrix}\mathbf{A} & \mathbf{0}\\ \mathbf{G} & \mathbf{0}\end{bmatrix}\begin{pmatrix}\mathbf{x}_e\\ \mathbf{x}_i\end{pmatrix} + \begin{bmatrix}\mathbf{B}\\ \mathbf{0}\end{bmatrix}\mathbf{u} \tag{25}$$

where G is a sparse matrix defining the states that are selected for integral action. As is common practice, wind disturbances and actuator and engine dynamics are neglected for the LQR design systems. A linear model including models of actuators and engine is used for fast evaluation of candidate controllers during controller tuning.

5.1 Coordinate frames

The system to be controlled is a formation of n UAS flying in an arbitrary pattern. In the literature, a variety of guidance frames has been proposed, ranging from the predecessor's body frame to a common guidance frame attached to the formation's barycenter.

The primary objective of tight formation flight is to keep each UAS at the position of maximum efficiency in the wake of its predecessor. Being induced by the aerodynamic flow, the wake vortices are approximately aligned with the predecessor's wind frame, neglecting trajectory curvature. For maximum energy savings, the follower thus needs to keep its relative position constant in this frame. Since small UAS typically are not equipped with sensors for angle of attack and side slip angle, a local guidance frame (index g) is introduced as an approximation. Its x axis unit vector $\mathbf{x}_g$ is aligned with each predecessor's NED velocity vector. Its y axis should be aligned with the y axis of the predecessor's body frame, since the central axis of the vortices lies somewhere on the predecessor's wing. To avoid having to communicate the NED speed and attitude of each UAS to its follower, the nominal speed and acceleration vector of each UAS on the current formation trajectory is used to derive a nominal guidance frame. These two vectors are computed sequentially for each UAS, since the attitude of each nominal local guidance frame defines the nominal NED position of the next one. For two subsequent UAS, predecessor p and follower f, the nominal NED speed of p is computed as

$$\mathbf{v}_f^e = \mathbf{v}_p^e + \omega_p^e \times \Delta\mathbf{p}_f^e \tag{26}$$

with the commanded separation vector $\Delta\mathbf{p}_f^e$. Acceleration is given by

$$\dot{\mathbf{v}}_f^e = \dot{\mathbf{v}}_p^e + \omega_p^e \times (\omega_p^e \times \Delta\mathbf{p}^e) \tag{27}$$

and the instantaneous rotation rate ω_f^e of the guidance frame due to trajectory curvature by

$$|\omega_f^e| = \left(\frac{|\dot{\mathbf{v}}_f^e|}{|\Delta\mathbf{p}^e|}\right)^{\frac{1}{2}} \tag{28}$$

$$\omega_f^e = |\omega_f^e|\frac{\dot{\mathbf{v}}_f^e \times \mathbf{v}_f^e}{|\dot{\mathbf{v}}_f^e \times \mathbf{v}_f^e|} \tag{29}$$

The x axis unit vector of the local guidance frame results then to

$$\mathbf{u}_x = |\mathbf{v}_f^e|^{-1}\mathbf{v}_f^e \tag{30}$$

The orientation of the z axis unit vector $\mathbf{u}_z$ is derived from the simplifying assumption that gravitational acceleration and centrifugal forces due to trajectory curvature are compensated for by the thrust and drag along $\mathbf{x}_g$ and by the aerodynamic lift in a plane normal to $\mathbf{x}_g$. Further assuming that the aerodynamic lift Z lies in the x-z plane of the body frames, first the total centrifugal and gravitational acceleration acting on the aircraft is computed as

$$\mathbf{a}_t = |\omega_p|^2\mathbf{r} + (0\,0\,g)^T \tag{31}$$

The z unit vector $\mathbf{u}_z$ is then computed by projecting and normalizing the total acceleration on the z-y plane of the g frame

$$\begin{aligned}\mathbf{a}_{t,zy} &= \mathbf{a}_t - \frac{\mathbf{a}_t \cdot \mathbf{u}_x}{\mathbf{u}_x \cdot \mathbf{u}_x}\cdot\mathbf{u}_x && (32)\\ \mathbf{u}_z &= |\mathbf{a}_{t,zy}|^{-1}\mathbf{a}_{t,zy} && (33)\end{aligned}$$

and $\mathbf{u}_y$ completes a right-handed Cartesian coordinate frame

$$\mathbf{u}_y = \mathbf{u}_z \times \mathbf{u}_x \tag{34}$$

leading to the corresponding rotation matrix

$$\mathbf{R}_{\mathbf{e}\leftarrow\mathbf{g}} = [\mathbf{u}_x\ \mathbf{u}_y\ \mathbf{u}_z] \tag{35}$$

The guidance frame and its computation should be illustrated by an appropriate figure, this is deferred to the final version due to time constraints.

5.2 Predecessor tracking

Longitudinal The control inputs of the longitudinal position control loop are the commanded pitch angle Θ_c and the engine command $\delta_{en,c}$. In the absence of external disturbances it regulates the relative position and velocity errors asymptotically to zero. Using Θ_c as control input instead of the angle of attack α, which is more directly related to aerodynamic lift, is imposed by the usual absence of AoA sensors on board of small UAS. To obtain a good approximation of α, the pitch angle w.r.t to the local guidance frame is used. The design system is given by

$$\begin{pmatrix}\Delta\dot{p}_x\\ \Delta\dot{p}_z\\ \Delta\dot{v}_x\\ \Delta\dot{v}_z\\ \Delta\dot{p}_{x,int}\\ \Delta\dot{p}_{z,int}\end{pmatrix} = \begin{bmatrix}\mathbf{A}_{lon,pos} & \mathbf{0}\\ [\mathbf{I}_2\ \mathbf{0}_2] & \mathbf{0}\end{bmatrix}\begin{pmatrix}\Delta p_x\\ \Delta p_z\\ \Delta v_x\\ \Delta v_z\\ \Delta p_{x,int}\\ \Delta p_{z,int}\end{pmatrix} + \mathbf{B}_{lon,pos}\begin{pmatrix}\delta_{en}\\ \Theta_c\end{pmatrix} \tag{36}$$

Lateral The lateral control strategy is derived from the simplified system

$$n_y = \frac{Z}{mg}\sin(\Phi) \tag{37}$$

$$Z \approx mg \tag{38}$$

$$\Rightarrow n_y \approx \sin^{-1}(\Phi) \tag{39}$$

thus lateral load factors in the local guidance frame are generated by inclining the lift vector via the local bank angle Φ. The resulting LQR design system is given by

$$\begin{pmatrix} \Delta\dot{p}_y \\ \Delta\dot{v}_y \\ \Delta\dot{p}_{y,int} \end{pmatrix} = \begin{bmatrix} \mathbf{A}_{lat,pos} & 0 \\ [1\ 0] & 0 \end{bmatrix} \begin{pmatrix} \Delta p_y \\ \Delta v_y \\ \Delta p_{y,int} \end{pmatrix} + \mathbf{B}_{lat,pos}\,(n_{y,c}) \tag{40}$$

and the commanded bank angle in the local guidance frame is computed using (39).

5.3 Attitude tracking

The inner loops consist of separate pitch angle and bank angle tracking laws. Both angles are not the usual Euler angles w.r.t. to the NED frame but are defined as the rotation angles needed to rotate the body frame into the local guidance frame. That being said, the resulting commanded attitude rotation matrix w.r.t. to the NED frame is composed as

$$\mathbf{R}_{\mathbf{e}\leftarrow\mathbf{b},c} = \mathbf{R}_{\mathbf{e}\leftarrow\mathbf{g}}\,\mathbf{R}_{\mathbf{g}\leftarrow\mathbf{b},c}(\Phi_c, \Theta_c) \tag{41}$$

where

$$\mathbf{R}_{\mathbf{g}\leftarrow\mathbf{b},c}(\Phi_c, \Theta_c) = \begin{bmatrix} \cos\Phi_c & \sin\Phi_c\cos\Theta_c & \sin\Phi_c\cos\Theta_c \\ -\sin\Phi_c & \cos\Phi_c\cos\Theta_c & \cos\Theta_c\sin\Theta_c \\ 0 & -\sin\Theta_c & \cos\Phi_c \end{bmatrix} \tag{42}$$

Note that the local heading angle is not actively controlled and as such is set to zero.

Longitudinal The pitch attitude design system is given by

$$\begin{pmatrix} \Delta\dot{\Theta} \\ \dot{q} \\ \Delta\dot{\Theta}_{int} \end{pmatrix} = \begin{bmatrix} \mathbf{A}_{lon,att} & 0 \\ [1\ 0] & 0 \end{bmatrix} \begin{pmatrix} \Delta\Theta \\ q \\ \Delta\Theta_{int} \end{pmatrix} + \mathbf{B}_{lon,att}\,(\delta_e) \tag{43}$$

Lateral The lateral attitude law is driven by two requirements: tracking a commanded local bank angle and adding damping to the weakly damped dutch roll mode. The second one is achieved by adding the yaw rate r to the state vector of the design system and choosing an appropriate weight in the LQR synthesis matrix $\mathbf{Q}$. The design system is thus:

$$\begin{pmatrix} \Delta\dot{\Phi} \\ \dot{p} \\ \dot{r} \\ \Delta\dot{\Phi}_{int} \end{pmatrix} = \begin{bmatrix} \mathbf{A}_{lat,att} & 0 \\ [1\ 0\ 0] & 0 \end{bmatrix} \begin{pmatrix} \Delta\Phi \\ p \\ r \\ \Delta\Phi_{int} \end{pmatrix} + \mathbf{B}_{lat,att} \begin{pmatrix} \delta_a \\ \delta_r \end{pmatrix} \tag{44}$$

6. SIMULATION RESULTS

Flight on the benchmark trajectory has been simulated for a formation of three UAS both in calm air and under turbulent wind with $\mathbf{v}_{w,a} = (-3\ -3\ 0)^T\ \frac{m}{s}$ i.e. about 28 % of the nominal airspeed of 15 $\frac{m}{s}$. The separation vectors have been selected as $\Delta\mathbf{p}_i(-b\,b\,0)^T$ for $i = 1...3$.

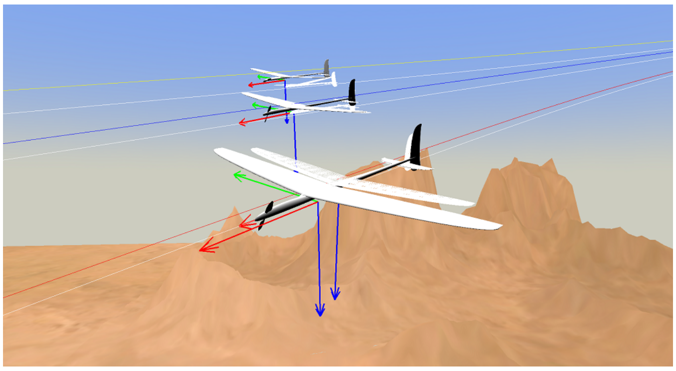

Fig. 3. N=3 formation in UAV3D

6.1 Visualization

A lightweight visualization environment based on the Java jmonkey game engine (see jmonkeyengine.org (2016)) has been developed to display the attitude and positions of multiple UAS in a synthetic 3D environment. It has proven to enhance productivity while developing and debugging guidance laws. It is interfaced with the Simulink dynamics simulation via a UDP (User Datagram Protocol) network link. Since the visuals have to be updated only at a moderate rate of about 30 Hz for human perception, sending UDP packets adds a small overhead to the overall simulation time. The tool can be easily extended and currently provides several generic UAS 3D models as well as the possibility to display wake vortices.

6.2 Calm atmosphere

Position errors in the local guidance frames along with the longitudinal control inputs under calm air conditions are shown in fig. 4. Simulating the benchmark formation in calm air clearly reveals two deficiencies of the baseline controllers. The first is their expected mesh instability. The second one is related to the selection of guidance frames, see section 5. Since the local guidance frames have been selected to track the predecessor's wake vortices, changes of trajectory curvature create perturbations w.r.t. the NED frame that are dependent on the separation vectors and vehicle index. This leads - with linear controllers - to position errors that grow with the vehicle index. Tracking the vortex during maneuvers is therefore not a scalable guidance strategy. This constitutes an open problem.

6.3 Turbulent atmosphere

Position errors in the local guidance frames along with the longitudinal control inputs under wind and turbulence are shown in fig. 4. Position errors are considerably larger during the initial cruise flight phase. On the helical part of the trajectory, perturbations due to reference frame perturbations dominate. The helical trajectory clearly reveals another shortcoming of the baseline control strategy. As the formation descends and the changing trajectory heading leads to tailwind, the engine inputs saturate as the longitudinal position controller attempts to maintain

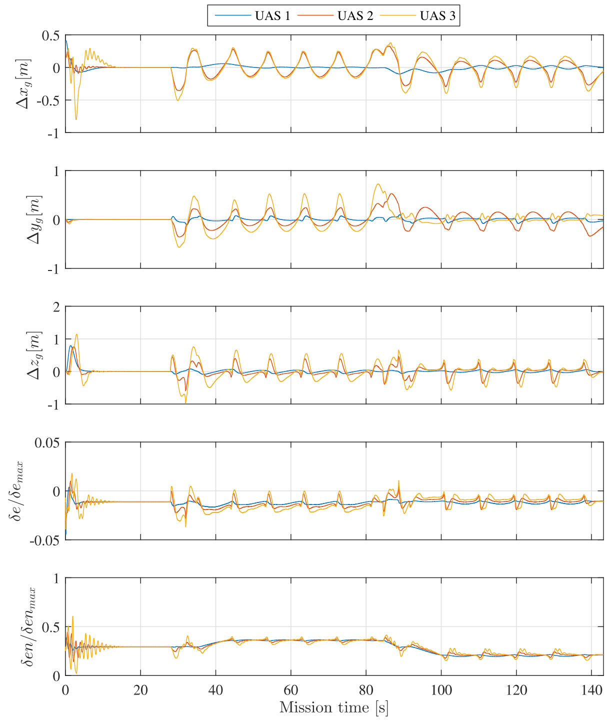

Fig. 4. Position errors in g frame and longitudinal control inputs for calm atmosphere

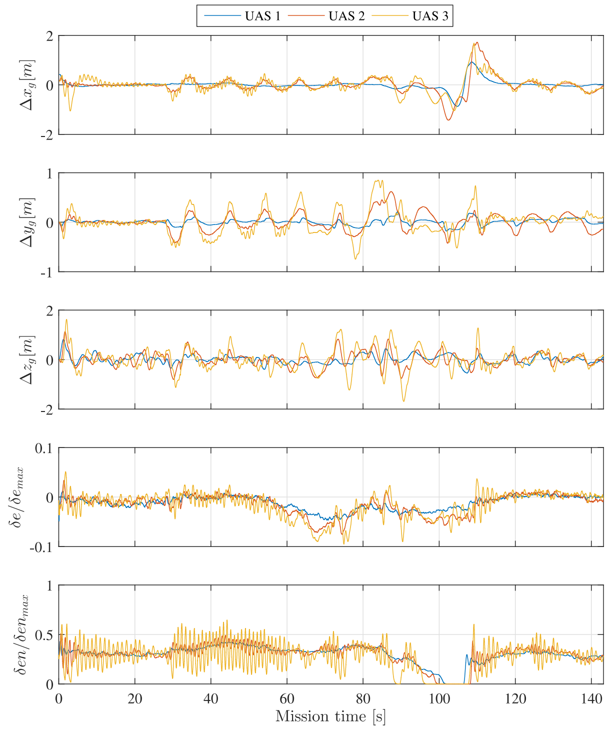

Fig. 5. Position errors in g frame and longitudinal control inputs for turbulent atmosphere

constant ground speed. It seems therefore advisable to add some kind of wind estimation scheme into the guidance strategy to maintain constant airspeed.

7. CONCLUSION

A new benchmark for distributed formation flight control for small UAS has been presented, along with simulation results for a set of baseline controllers. The results confirm their expected mesh instability as well as the ability of the chosen reference trajectory to reveal this and various other deficiencies of the baseline guidance strategy. These deficiencies provide a starting point for future improvements. It is demonstrated that tracking the predecessor's wake vortex during maneuvers is not a scalable guidance strategy.

We invite other researchers working in this domain to adopt this benchmark to evaluate new promising control and guidance approaches and to contribute to it e.g. by proposing and implementing new benchmark trajectories or by adding to the documentation. The benchmark and the tools it is built on are version controlled for guaranteed comparability and openly available, Bolting (2016d,b,a,c). The implementation depends on a number of Matlab/Simulink toolboxes and blocksets. To reduce this threshold, we are currently evaluating the possibility of a hosted benchmark that is accessible via the Internet.

REFERENCES

Ameho, Y. (2013). *Du pilotage d'une famille de drones à celui d'un drone hybride via la commande adaptative.* Ph.D. thesis, Toulouse, ISAE.

Beard, R.W. and McLain, T.W. (2012). *Small Unmanned Aircraft: Theory and Practice.* Princeton University Press, Princeton, NJ, USA.

Bolting, J. (2016a). Formation Flying Benchmark. https://github.com/jtec/ffb. URL `https://github.com/jtec/ffb`.

Bolting, J. (2016b). Simulink uav simulation blockset. https://github.com/jtec/uavsimblockset. URL `https://github.com/jtec/uavsimblockset`.

Bolting, J. (2016c). Spline Trajectory Generator library. https://github.com/jtec/STRAGE. URL `https://github.com/jtec/STRAGE`.

Bolting, J. (2016d). UAV3D Visualization environment. https://github.com/jtec/UAV3D. URL `https://github.com/jtec/UAV3D`.

Bolting, J., Fergani, S., Biannic, J.M., Defay, F., and Stolle, M. (2016). Discrete sliding mode control of small uas in tight formation flight under information constraints. In *Proceedings of IFAC Symposium on Automatic Control in Aerospace (ACA) 2016.*

Galzi, D. and Shtessel, Y. (2006). UAV formations control using high order sliding modes. In *American Control Conference, 2006*, 6–pp. IEEE.

Gu, Y., Seanor, B., Campa, G., Napolitano, M.R., Rowe, L., Gururajan, S., and Wan, S. (2006). Design and flight testing evaluation of formation control laws. *Control Systems Technology, IEEE Transactions on*, 14(6), 1105–1112.

Hummel, D. (1982). Aerodynamic aspects of formation flight in birds. *Journal of theoretical Biology.*

Jake, V., Ray, R., Ennix, K., and Walsh, K. (2003). F/A-18 Performance Benefits Measured During the Autonomous Formation Flight Project. Technical report, NASA/TM-2002-210732, Sept.

jmonkeyengine.org (2016). jmonkeyengine.org Java Game Engine. https://jmonkeyengine.org/. URL `https://jmonkeyengine.org/`.

Knorn, S., Donaire, A., Agüero, J.C., and Middleton, R.H. (2014). Passivity-based control for multi-vehicle systems subject to string constraints. *Automatica*, 50(12), 3224–3230.

Looye, G. (2011). Helical flight path trajectories for autopilot evaluation. In *Advances in Aerospace Guidance, Navigation and Control*, 79–90. Springer.

Nageli, T., Conte, C., Domahidi, A., Morari, M., and Hilliges, O. (2014). Environment-independent formation flight for micro aerial vehicles. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, 1141–1146. IEEE.

Pant, A., Seiler, P., Koo, T.J., and Hedrick, K. (2001). Mesh stability of unmanned aerial vehicle clusters. In *American Control Conference, 2001. Proceedings of the 2001*, volume 1, 62–68. IEEE.

Park, S. (2004). *Avionics and Control System Development for Mid-Air Rendezvous of Two Unmanned Aerial Vehicles*. Ph.D. thesis, MIT.

Shaw, E. and Hedrick, J.K. (2007). Controller design for string stable heterogeneous vehicle strings. In *Decision and Control, 2007 46th IEEE Conference on*, 2868–2875. IEEE.

Stolle, M. (2016). *Towards vision-based autonomous cross-country soaring*. Ph.D. thesis.

Weimerskirch, H., Martin, J., Clerquin, Y., Alexandre, P., and Jiraskova, S. (2001). Energy saving in flight formation. *Nature*, 413(6857), 697–698. URL `http://dx.doi.org/10.1038/35099670`.

Yang, K. and Sukkarieh, S. (2008). 3D smooth path planning for a UAV in cluttered natural environments. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, 794–800. doi: 10.1109/IROS.2008.4650637.