Trajectory Generation and Tracking for an Agile Fixed-Wing VTOL Aircraft

Ezra Tal, Gilhyun Ryou, and Sertac Karaman

Abstract—**This extended abstract summarizes our recent work on trajectory generation and flight control for agile maneuvering of a tailsitter flying wing unmanned aerial vehicle. Our work shows differential flatness of a realistic model of the tailsitter dynamics with global aerodynamics equations. Based on the derived flatness transform, we propose an algorithm for real-time generation of aerobatic trajectories in the flat output space. A novel control system, specifically designed for agile flight, enables robust tracking of the generated trajectories through incremental nonlinear dynamic inversion (INDI) and incorporates flatness-based feedforward references for jerk and yaw rate. Our algorithms enable full use of the vehicle's extensive flight envelope, achieving accurate tracking of challenging aerobatic maneuvers, including inverted, sideways, and (partially) stalled flight.**

Supplemental Material

Video of the flight experiments can be found on the project website https://aera.mit.edu/projects/TailsitterAerobatics.

I. Introduction

A large portion of academic and industry research on agile aerial navigation focuses on multicopters, particularly quadcopters. These aircraft are relatively simple to design and control, and their agility has been convincingly demonstrated in recent works. Nonetheless, multicopters are at a disadvantage when considering high-speed or long-distance flight as their power consumption grows almost monotonously with increasing airspeed. On the other hand, fixed-wing aircraft provide much better range and endurance, but lack the agility to maneuver—let alone take-off and land—in confined spaces.

Transitioning aircraft have the potential to provide the best of both worlds by combining vertical take-off and landing (VTOL) with efficient horizontal flight. This versatility is relevant to many real-world applications. For example, in search and rescue scenarios, transitioning aircraft can quickly reach remote locations and closely inspect (indoor) areas of interest. Tailsitter VTOL aircraft, specifically, transition between hover and forward flight by pitching, so that their rotors transition between lift generation and forward propulsion based on their attitude. The tailsitter *flying wing* shown in Fig. 1, which lacks fuselage, tail, and vertical stabilizers, profits from reduced mass and drag and has a relatively simple aerodynamic and mechanical design. By placing flaps that act as both elevator and aileron, i.e., elevons, in the rotor wash, the aircraft remains controllable throughout its flight envelope, including static hover conditions. Instead of vertical surfaces, differential thrust is used to provide active directional stabilization. This arrangement increases maneuverability, as it permits uncoordinated flight, where the vehicle incurs nonzero lateral velocity, enabling maneuvers such as fast skidding turns and knife-edge flight.

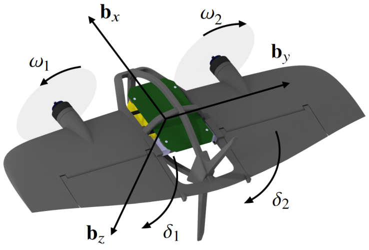


Fig. 1: Tailsitter flying wing body-fixed reference frame and control inputs.

In this extended abstract, we summarize our recent work on trajectory generation and control for agile flight using the tailsitter flying wing aircraft [1], [2], [3]. Our work contains several contributions: We show differential flatness of a global flight dynamics model, including aerodynamics equations based on φ-theory [4]. Based on the derived flatness transform, we propose an algorithm for generating fast and agile tailsitter trajectories at low computational cost, i.e., suitable for online motion planning applications. Our algorithm is capable of generating aerobatics maneuvers that exploit the entire flight envelope, including challenging conditions, such as knife-edge and inverted flight. Furthermore, we propose a control algorithm for tracking of such aerobatic trajectories. The controller exploits differential flatness to track (feedforward) inputs corresponding to the reference position, velocity, acceleration, and jerk (the third derivative of position), as well as yaw angle and yaw rate. We apply incremental nonlinear dynamic inversion (INDI) to achieve accurate trajectory tracking despite model discrepancies. All contributions are demonstrated in extensive flight experiments.

II. Differentially Flat Dynamics Model

The vehicle translational and rotational dynamics are modeled using the Newton-Euler equations

$$\ddot{\mathbf{x}} = \dot{\mathbf{v}} = g\mathbf{i}_z + m^{-1}\mathbf{R}^i_\alpha \mathbf{f}^\alpha, \tag{1}$$

$$\dot{\boldsymbol{\xi}} = \frac{1}{2}\boldsymbol{\xi} \circ \boldsymbol{\Omega}, \tag{2}$$

$$\dot{\boldsymbol{\Omega}} = \mathbf{J}^{-1}(\mathbf{m} - \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega}), \tag{3}$$

with $\mathbf{x}$ and $\mathbf{v}$ respectively the center of mass position and

E. Tal, G. Ryou, and S. Karaman are with the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology. This work was supported by the Army Research Office through grant W911NF1910322. {eatal, ghryou, sertac}@mit.edu

$$\mathbf{f}^\alpha = \sum_{i=1}^{2} \underbrace{\begin{bmatrix} \cos\bar{\alpha}\ (1 - c_{D_T}) \\ 0 \\ \sin\bar{\alpha}\ (c_{L_T} - 1) \end{bmatrix} T_i}_{\mathbf{f}^\alpha_{T_i}} + \sum_{i=1}^{2} \underbrace{- \begin{bmatrix} 0 \\ 0 \\ c^\delta_{L_T} \cos\bar{\alpha}\ T_i + c^\delta_{L_V} \|\mathbf{v}\| \mathbf{i}_x^\top \mathbf{v}^\alpha \end{bmatrix} \delta_i}_{\mathbf{f}^\alpha_{\delta_i}} - \begin{bmatrix} c_{D_V} \mathbf{i}_x^\top \mathbf{v}^\alpha \\ 0 \\ c_{L_V} \mathbf{i}_z^\top \mathbf{v}^\alpha \end{bmatrix} \|\mathbf{v}\| \tag{4}$$

$$\mathbf{m} = \begin{bmatrix} l_{T_y} \mathbf{i}_z^\top \mathbf{R}^b_\alpha (\mathbf{f}^\alpha_{T_2} - \mathbf{f}^\alpha_{T_1}) \\ c_{\mu_T} (T_1 + T_2) \\ l_{T_y} \mathbf{i}_x^\top \mathbf{R}^b_\alpha (\mathbf{f}^\alpha_{T_1} - \mathbf{f}^\alpha_{T_2}) \end{bmatrix} + \begin{bmatrix} \cos\alpha_T \\ 0 \\ -\sin\alpha_T \end{bmatrix} \sum_{i=1}^{2} \mu_i + \begin{bmatrix} l_{\delta_y} \cos\alpha_0\ \mathbf{i}_z^\top (\mathbf{f}^\alpha_{\delta_2} - \mathbf{f}^\alpha_{\delta_1}) \\ l_{\delta_x} \mathbf{i}_z^\top (\mathbf{f}^\alpha_{\delta_1} + \mathbf{f}^\alpha_{\delta_2}) \\ l_{\delta_y} \sin\alpha_0\ \mathbf{i}_z^\top (\mathbf{f}^\alpha_{\delta_2} - \mathbf{f}^\alpha_{\delta_1}) \end{bmatrix}. \tag{5}$$

velocity in the world-fixed reference frame, $\boldsymbol{\xi}$ the vehicle attitude, $\boldsymbol{\Omega}$ and $\mathbf{m}$ respectively the angular velocity and moment in the body-fixed reference frame (see Fig. 1), m the mass, $\mathbf{J}$ the inertia tensor, and g the gravitational acceleration. The thrust and aerodynamic force $\mathbf{f}^\alpha$ is expressed in the zero-lift reference frame (i.e., the body-fixed frame rotated around the negative $\mathbf{b}_y$-axis by the zero-lift angle of attack α_0) and transformed to the world-fixed frame using the rotation matrix $\mathbf{R}^i_\alpha$. We employ φ-theory to model the aerodynamic force and moment [4]. This parametrization provides a simple global model that includes dominant contributions over the entire flight envelope, including post-stall conditions. It avoids the singularity that traditional methods incur near hover, where the angle of attack and the sideslip angle are undefined. The resulting expressions for the force and moment are given by (4) and (5), where c denotes the φ-theory parameters, T_i and μ_i are respectively the thrust and torque due to rotor i, l indicates geometric properties of the wing, and $\bar{\alpha} = \alpha_0 + \alpha_T$ with α_T the thrust angle.

Under some assumptions, the nonlinear dynamics system described by (1) through (5) is differentially flat[1]. This entails that its state and input variables can be directly expressed as a function of the flat output

$$\boldsymbol{\sigma}(t) = [\mathbf{x}(t)^\top\ \psi(t)]^\top, \tag{6}$$

consisting of the position and yaw angle, and a finite number of its derivatives. Intuitively, the flatness transform is derived by rewriting (1) to obtain the force vector $\mathbf{R}^i_\alpha \mathbf{f}^\alpha$ as a function of the second derivative of position, i.e., acceleration. The vehicle attitude and collective thrust are then uniquely defined by three constraints:

(i) the yaw angle ψ,
(ii) the fact that $\mathbf{i}_y^\top \mathbf{f}^\alpha = 0$ according to (4), and
(iii) the forces in the vehicle symmetry plane, i.e., $\mathbf{i}_x^\top \mathbf{f}^\alpha$ and $\mathbf{i}_z^\top \mathbf{f}^\alpha$.

By twice taking the derivative of the resulting expression for the attitude, we obtain the angular velocity and angular acceleration as a function of the velocity, acceleration, jerk, snap, yaw, yaw rate, and yaw acceleration. Finally, the moment is obtained through inversion of (3), so that the rotor speeds and flap deflections can be obtained using (5).

III. Aerobatic Trajectory Generation

Trajectory generation algorithms for fixed-wing aircraft often avoid the relatively complicated flight dynamics and instead use kinematics models, such as Dubins paths [7]. However, when considering fast and agile flight, realistic aircraft dynamics and control input constraints must be considered, so that the resulting trajectory is dynamically feasible, i.e., so that it can be accurately tracked in flight. Directly using the six-degree-of-freedom (6DOF) nonlinear flight dynamics model is computationally costly, making it unsuitable for online applications [8]. Hence, existing works resort to simplifications, such as point-mass equations of motion [9], [10], [11], pre-computed maneuvers [12], [13], or local models [14]. In practice, these methods impose limitations on the generated trajectories, especially when planning aerobatic maneuvers that rapidly progress through unconventional flight conditions. Alternatively, flatness-based algorithms, widely employed for quadcopters, generate trajectories in the relatively simple flat output space. Dynamic feasibility of the output trajectory is evaluated using the corresponding control input trajectory (obtained using the flatness transform), enabling computationally efficient trajectory generation with realistic feasibility constraints. Flatness has also been considered for fixed-wing aircraft dynamics [6], but its application has been limited to strongly simplified models that do not capture the dynamics of agile aerobatics [15], [16], [17].

We propose a novel flatness-based trajectory generation algorithm that considers the global 6DOF flight dynamics model described in Section II, including aerodynamics equations. As such, it is capable of generating aerobatic maneuvers that

[1] For an introduction to differential flatness, see, e.g., [5], [6].

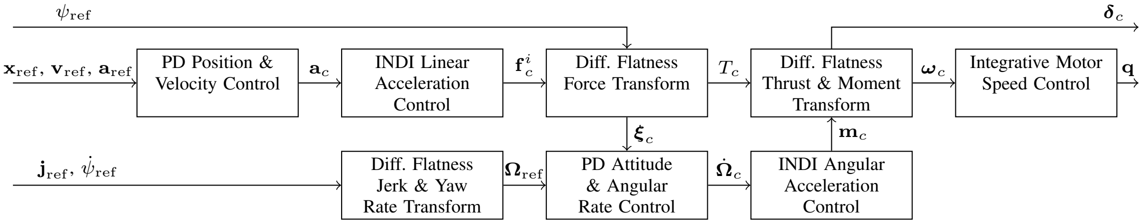

ψ_{ref}
$\boldsymbol{\delta}_c$
$\mathbf{x}_{\text{ref}}, \mathbf{v}_{\text{ref}}, \mathbf{a}_{\text{ref}}$
PD Position & Velocity Control
$\mathbf{a}_c$
INDI Linear Acceleration Control
$\mathbf{f}^i_c$
Diff. Flatness Force Transform
T_c
Diff. Flatness Thrust & Moment Transform
$\boldsymbol{\omega}_c$
Integrative Motor Speed Control
$\mathbf{q}$
$\boldsymbol{\xi}_c$
$\mathbf{m}_c$
$\mathbf{j}_{\text{ref}}, \dot{\psi}_{\text{ref}}$
Diff. Flatness Jerk & Yaw Rate Transform
$\boldsymbol{\Omega}_{\text{ref}}$
PD Attitude & Angular Rate Control
$\dot{\boldsymbol{\Omega}}_c$
INDI Angular Acceleration Control

Fig. 2: Overview of trajectory-tracking control architecture.

exploit the entire flight envelope, including (partially) stalled, sideways, and inverted flight. The tailsitter flatness transform has a similar structure as the well-known quadcopter flat transform, in the sense that snap and yaw acceleration roughly correspond to the control inputs. Hence, their reduction also increases feasibility of tailsitter trajectories, akin to the premise of minimum-snap trajectory generation algorithms for quadcopters [18], [19]. This enables the application of similar algorithms toward aerobatic tailsitter trajectory generation.

In order to demonstrate our approach, we use the formulation by [19] to describe the trajectory with piecewise polynomial functions that are defined in terms of their derivatives at the waypoints. First, we obtain the minimum-snap time allocation $\mathbf{t}^*$ over the segments between the waypoints $\tilde{\boldsymbol{\sigma}}$ by solving

$$\begin{aligned} \underset{\boldsymbol{\sigma},\mathbf{t}}{\text{minimize}} \quad & \int_0^{\bar{T}} \left\| \frac{d^4\mathbf{x}}{dt^4} \right\|^2 + \mu_\psi \left(\frac{d^2\psi}{dt^2} \right)^2 dt \\ \text{subject to} \quad & \boldsymbol{\sigma} = \boldsymbol{\chi}\left(\mathbf{t}, \tilde{\boldsymbol{\sigma}}, \dot{\tilde{\boldsymbol{\sigma}}}, \ddot{\tilde{\boldsymbol{\sigma}}}, \dots\right), \\ & \sum\nolimits_{j=1}^{m} t_j = \bar{T}, \end{aligned} \tag{7}$$

where $\bar{T}$ is a rough estimate of the total trajectory time. The minimum-snap trajectory $\boldsymbol{\sigma} = \boldsymbol{\chi}\left(\mathbf{t}, \tilde{\boldsymbol{\sigma}}, \dot{\tilde{\boldsymbol{\sigma}}}, \ddot{\tilde{\boldsymbol{\sigma}}}, \dots\right)$, where $\dot{\tilde{\boldsymbol{\sigma}}}$, $\ddot{\tilde{\boldsymbol{\sigma}}}$ etc. denote optional derivative constraints at (some) waypoints, is efficiently obtained in closed form using matrix multiplications [19]. Next, we minimize the scale factor c that is applied to $\mathbf{t}^*$ to obtain the quickest minimum-snap trajectory $\boldsymbol{\sigma} = \boldsymbol{\chi}\left(c\mathbf{t}^*, \tilde{\boldsymbol{\sigma}}, \dot{\tilde{\boldsymbol{\sigma}}}, \ddot{\tilde{\boldsymbol{\sigma}}}, \dots\right)$ that is in the feasible set

$$\Sigma_T = \left\{ \boldsymbol{\sigma} \middle| \mathbf{u}(t) \in \mathcal{U} \quad \forall t \in \left[0, c\bar{T}\right] \right\}, \tag{8}$$

where the control input trajectory $\mathbf{u}$ is obtained from $\boldsymbol{\sigma}$ using the flatness transform described in Section II, and $\mathcal{U}$ is the set of permissible control inputs, i.e., the bounded set defined by the minimum and maximum allowed rotor speeds and flap deflections. Through extensive experimental evaluation on various trajectories, we found that application of the flatness transform to evaluate (8) does indeed give a useful prediction of the critical trajectory time or speed where a stark increase in tracking error on the real vehicle occurs (see Section V for an example).

IV. Trajectory-tracking Flight Control

Flight control design for tailsitter aircraft is complicated by the change of dynamics between hover and forward flight. Existing designs address this transition in various ways, e.g., blending separate controllers [20], [21], gain scheduling [22], [23], or pre-planned transition maneuvers [24]. When performing agile maneuvering at large angle of attack, the aircraft continuously enters and exits the transition regime, and it is preferable to utilize a global controller without blending or switching [25]. However, accurately modeling the global dynamics is challenging and discrepancies may lead to impaired performance [26]. Robustification can be used to design a performant controller that does not rely on an accurate model of the vehicle dynamics [27]. Incremental nonlinear dynamic inversion (INDI) increases robustness by only using a local control effectiveness model to incrementally update the control inputs based on inertial measurements [28], [29]. While INDI has been applied for robust control of a tailsitter flying wing, its use has been limited to coordinated flight at small flight path angles [30].

We present a novel tailsitter flight control design, specifically for tracking agile trajectories. Our design combines INDI with the global model from Section II to achieve robust tracking of challenging agile maneuvers, including uncoordinated, inverted, and (partially) stalled flight. We employ the flatness transform to incorporate feedforward inputs for jerk and yaw rate that enable accurate tracking of fast-changing acceleration references. Flatness also enables direct nonlinear inversion so that, contrary to existing INDI implementations, our controller does not rely on local linearization of the dynamics for inversion.

The controller, depicted in Fig. 2, tracks the dynamic trajectory reference (6). It uses cascaded PD position and velocity controllers with acceleration feedforward to obtain the linear acceleration command $\mathbf{a}_c$, which is robustly tracked using INDI control. The INDI controller estimates any external (i.e., unmodeled) force by comparing the measured acceleration to the expected acceleration according to (1), as follows:

$$\mathbf{f}_{\text{ext}} = m\left(\tilde{\mathbf{a}}_{\text{lpf}} - g\mathbf{i}_z\right) - \mathbf{R}^i_\alpha \mathbf{f}^\alpha_{\text{lpf}}, \tag{9}$$

where $\tilde{\mathbf{a}}_{\text{lpf}}$ is the low-pass filtered and gravity-corrected IMU acceleration measurement transformed to the world-fixed frame, and $\mathbf{f}^\alpha_{\text{lpf}}$ is the estimated force according to (4) based on low-pass filtered motor speed and flap deflection measurements. Substitution of (9) into (1) gives

$$\begin{aligned} \mathbf{a} &= g\mathbf{i}_z + m^{-1}\left(\mathbf{R}^i_\alpha \mathbf{f}^\alpha + \mathbf{f}_{\text{ext}}\right) \\ &= g\mathbf{i}_z + m^{-1}\left(\mathbf{R}^i_\alpha \mathbf{f}^\alpha + \left(m\left(\tilde{\mathbf{a}}_{\text{lpf}} - g\mathbf{i}_z\right) - \mathbf{R}^i_\alpha \mathbf{f}^\alpha_{\text{lpf}}\right)\right) \\ &= \tilde{\mathbf{a}}_{\text{lpf}} + m^{-1}\left(\mathbf{f}^i - \mathbf{f}^i_{\text{lpf}}\right). \end{aligned} \tag{10}$$

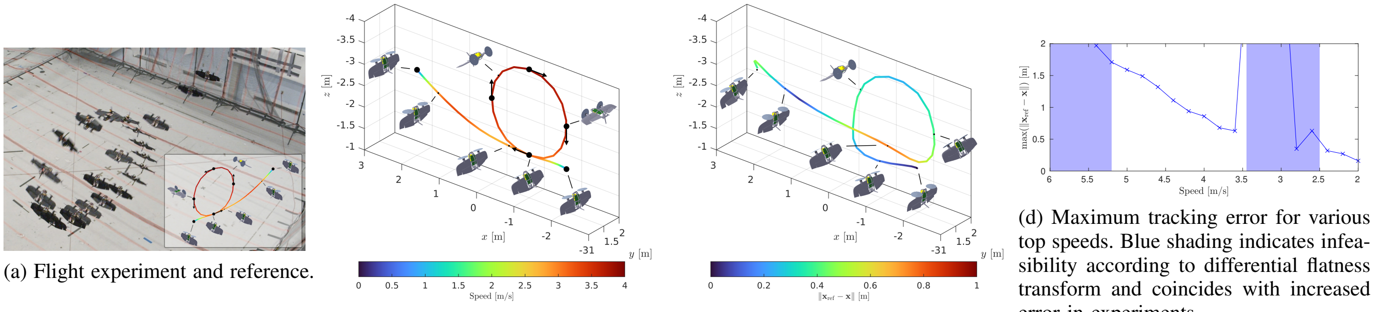
Solving (10) for $\mathbf{f}^i$ gives an incremental expression for the force command that corresponds to the commanded acceleration, as follows:

$$\mathbf{f}^i_c = m(\mathbf{a}_c - \tilde{\mathbf{a}}_{\text{lpf}}) + \mathbf{f}^i_{\text{lpf}}. \tag{11}$$

This incremental control law enables the controller to achieve the commanded acceleration despite potential modeling discrepancies and external forces, without depending on integral action. If the commanded acceleration is not yet attained, the force command will be adjusted further in subsequent control updates until the first term in (11) vanishes. Based on the force command $\mathbf{f}^i_c$, the commanded attitude $\boldsymbol{\xi}_c$ is obtained using the flatness transform described in Section II. Note that the flatness transform enables fully nonlinear inversion, without local linearization of (4) This nonlinear inversion provides more accurate control commands when large force changes occur, such as may happen during aggressive maneuvers with fast-changing acceleration references.

The attitude command is tracked using the PD controller

$$\dot{\boldsymbol{\Omega}}_c = \mathbf{K}_{\boldsymbol{\xi}} \boldsymbol{\zeta}_e + \mathbf{K}_{\boldsymbol{\Omega}}\left(\boldsymbol{\Omega}_{\text{ref}} - \boldsymbol{\Omega}_{\text{lpf}}\right), \tag{12}$$



(a) Flight experiment and reference.

(b) Reference with waypoints.

(c) Experiment.

(d) Maximum tracking error for various top speeds. Blue shading indicates infeasibility according to differential flatness transform and coincides with increased error in experiments.

Fig. 3: Loop trajectory reference and experimental results.

which also incorporates the angular velocity feedforward input $\boldsymbol{\Omega}_{\text{ref}}$ used to the track the reference jerk and yaw rate. An incremental controller similar to (10) is used to track the resulting angular acceleration command, as follows:

$$\mathbf{m}_c = \mathbf{J}(\dot{\boldsymbol{\Omega}}_c - \dot{\boldsymbol{\Omega}}_{\text{lpf}}) + \mathbf{m}_{\text{lpf}}. \tag{13}$$

Finally, the motor speed and flap deflection commands are obtained from the moment command $\mathbf{m}_c$ through nonlinear inversion of (5).

V. Flight Experiments

We have validated our algorithms in extensive experiments, including aerobatic maneuvers, drone racing trajectories, and an airshow-like aerobatic sequence for three tailsitter aircraft. Videos are available at https://aera.mit.edu/projects/TailsitterAerobatics. All experiments were conducted in an 18 m × 8 m motion capture space using the 3D-printed vehicle described in [31]. The φ-theory aerodynamic parameters used for trajectory generation and control were refined using experimental data to improve model accuracy.

Figure 3 shows a loop trajectory that was generated using five waypoints (of which two coincide) with tangential velocity constraints on a vertical circle with 1 m radius, and start and end points constrained to static hover. As shown in Fig. 3d, the loop trajectory has several feasibility boundaries. When flown slowly (i.e., below 2.5 m/s), the trajectory is feasible and flown in hover attitude. When flown faster (i.e., around 4.5 m/s), the vehicle performs a loop, making a full upward pitch rotation. Intermediate speeds (i.e., around 3 m/s) are too slow to perform a loop and require the vehicle to quickly pitch back down at the top of the circular segment, rendering the trajectory infeasible due to flap deflection limits. The maximum position tracking error obtained from flight experiments shows a stark increase in this region of infeasibility and also increases as the infeasibility boundary at very high speed (i.e., 5.2 m/s) is approached. The trajectory with a maximum speed of 3.8 m/s is shown in Fig. 3b and Fig. 3c. The loop maneuver is successfully performed in the flight experiment with a maximum position error of 74 cm.

References

[1] E. Tal and S. Karaman, “Global trajectory-tracking control for a tailsitter flying wing in agile uncoordinated flight,” in *AIAA Aviation 2021 Forum*, 2021, p. 3214.

[2] E. Tal, G. Ryou, and S. Karaman, “Aerobatic trajectory generation for a VTOL fixed-wing aircraft using differential flatness,” *arXiv preprint arXiv:2207.03524*, 2022.

[3] E. Tal and S. Karaman, “Global incremental flight control for agile maneuvering of a tailsitter flying wing,” *arXiv preprint arXiv:2207.13218*, 2022.

[4] L. R. Lustosa, F. Defaÿ, and J.-M. Moschetta, “Global singularity-free aerodynamic model for algorithmic flight control of tail sitters,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 42, no. 2, pp. 303–316, 2019.

[5] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, “Sur les systèmes non linéaires différentiellement plats,” *CR Acad. Sci. Paris*, pp. 619–624, 1992.

[6] P. Martin, “Contribution à l’étude des systèmes différentiellement plats,” Ph.D. dissertation, École Nationale Supérieure des Mines de Paris, 1992.

[7] H. Chitsaz and S. M. LaValle, “Time-optimal paths for a Dubins airplane,” in *IEEE Conference on Decision and Control (CDC)*, 2007, pp. 2379–2384.

[8] A. J. Barry, T. Jenks, A. Majumdar, H.-T. Lin, I. G. Ros, A. A. Biewener, and R. Tedrake, “Flying between obstacles with an autonomous knife-edge maneuver,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 2559–2559.

[9] H. van der Plas and H. Visser, “Trajectory optimisation of an aerobatic air race,” *The Aeronautical Journal*, vol. 113, no. 1139, pp. 1–8, 2009.

[10] M. A. Morales, F. J. Silvestre, and A. B. G. Neto, “Equations of motion for optimal maneuvering with global aerodynamic model,” *Aerospace Science and Technology*, vol. 77, pp. 206–216, 2018.

[11] P. Pashupathy, A. Maity, H. Hong, and F. Holzapfel, “Unspecified final-time nonlinear suboptimal guidance of aerobatic aircraft in air race,” *Aerospace Science and Technology*, vol. 116, p. 106817, 2021.

[12] J. M. Levin, M. Nahon, and A. A. Paranjape, “Real-time motion planning with a fixed-wing UAV using an agile maneuver space,” *Autonomous Robots*, vol. 43, no. 8, pp. 2111–2130, 2019.

[13] C.-J. Kim, M. J. Heo, J. W. Hwang, H. G. Lyu, J. Y. Lee *et al.*, “Development of real-time maneuver library generation technique for implementing tactical maneuvers of fixed-wing aircraft,” *International Journal of Aerospace Engineering*, vol. 2020, 2020.

[14] L. L. Beyer, N. Balabanska, E. Tal, and S. Karaman, “Multi-modal motion planning using composite pose graph optimization,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 9981–9987.

[15] J. Hauser and R. Hindman, “Aggressive flight maneuvers,” in *IEEE Conference on Decision and Control (CDC)*, 1997, pp. 4186–4191.

[16] J. Hall and T. McLain, “Aerobatic maneuvering of miniature air vehicles using attitude trajectories,” in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008, p. 7257.

[17] A. Bry, C. Richter, A. Bachrach, and N. Roy, “Aggressive flight of fixed-wing and quadrotor aircraft in dense indoor environments,” *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 969–1002, 2015.

[18] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 2520–2525.

[19] C. Richter, A. Bry, and N. Roy, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments,” in *International Symposium on Robotics Research (ISRR)*. Springer, 2016, pp. 649–666.

[20] Y. Ke, K. Wang, and B. M. Chen, “Design and implementation of a hybrid UAV with model-based flight capabilities,” *IEEE/ASME Trans. on Mechatronics*, vol. 23, no. 3, pp. 1114–1125, 2018.

[21] J. L. Forshaw, V. J. Lappas, and P. Briggs, “Transitional control architecture and methodology for a twin rotor tailsitter,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 37, no. 4, pp. 1289–1298, 2014.

[22] Y. Jung and D. H. Shim, “Development and application of controller for transition flight of tail-sitter UAV,” *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1, pp. 137–152, 2012.

[23] L. R. Lustosa, “The φ-theory approach to flight control design of hybrid vehicles,” Ph.D. dissertation, ISAE-SUPAERO, 2017.

[24] R. Chiappinelli and M. Nahon, “Modeling and control of a tailsitter UAV,” in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2018, pp. 400–409.

[25] P. Hartmann, C. Meyer, and D. Moormann, “Unified velocity control and flight state transition of unmanned tilt-wing aircraft,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 40, no. 6, pp. 1348–1359, 2017.

[26] R. Ritz and R. D’Andrea, “A global controller for flying wing tailsitter vehicles,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 2731–2738.

[27] J. M. Barth, J.-P. Condomines, M. Bronz, J.-M. Moschetta, C. Join, and M. Fliess, “Model-free control algorithms for micro air vehicles with transitioning flight capabilities,” *International Journal of Micro Air Vehicles*, vol. 12, 2020.

[28] P. Smith, “A simplified approach to nonlinear dynamic inversion based flight control,” in *AIAA Atmospheric Flight Mechanics Conference*, 1998, pp. 4461–4469.

[29] B. Bacon and A. Ostroff, “Reconfigurable flight control using nonlinear dynamic inversion with a special accelerometer implementation,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2000, pp. 4565–4579.

[30] E. J. Smeur, M. Bronz, and G. C. de Croon, “Incremental control and guidance of hybrid aircraft applied to a tailsitter unmanned air vehicle,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 43, no. 2, pp. 274–287, 2020.

[31] M. Bronz, E. Tal, F. Favalli, and S. Karaman, “Mission-oriented additive manufacturing of modular mini-UAVs,” in *AIAA Scitech 2020 Forum*, 2020, p. 0064.