Opportunities and Challenges with Autonomous Micro Aerial Vehicles

Vijay Kumar and Nathan Michael

Abstract We survey the recent work on micro-UAVs, a fast-growing field in robotics, outlining the opportunities for research and applications, along with the scientific and technological challenges. Micro-UAVs can operate in three-dimensional environments, explore and map multi-story buildings, manipulate and transport objects, and even perform such tasks as assembly. While fixed-base industrial robots were the main focus in the first two decades of robotics, and mobile robots enabled most of the significant advances during the next two decades, it is likely that UAVs, and particularly micro-UAVs will provide a major impetus for the third phase of development.

1 Introduction

The last decade has seen many exciting developments in the area of micro Unmanned Aerial Vehicles (UAVs) that are between 0.1-0.5 meters in length and 0.1-0.5 kilograms in mass. Just as the incorporation of 2-D mobility reinvigorated robotics research in the 1990s, the ability to operate in truly three-dimensional environments is bringing in new research challenges along with new technologies and applications. Indeed by some estimates [52], the UAV market is estimated to exceed $60 B in the next three years, and this forecast is conservative since it does not account for the thousands of micro-UAVs that are likely to be fielded in the near future.

Our focus in this work is on UAVs that have gross weights of the order of 1 kg and below; although as described in [6, 9, 31, 41] the platform development represents a challenge in its own right. While commercial products ranging from 5 g. to 350 g. are available, most of these products do not carry the sensors and processors required for autonomous flight. Many of these small aircrafts do not have the endurance required for missions of longer than 5 minutes. Longer endurance requires bigger batteries, and with the current energy densities of Li-polymer batteries (of the order of several hundred Watt-hr/kg), the mass fraction used by batteries is significant, often between 25-50% of the gross weight.

There are many types of micro-UAVs that are in various phases of research, development and practice. Fixed-wing aircrafts are less adept than rotor crafts at maneuvering in constrained, 3-D environments. While avian-style flapping wing aircrafts provide more agility, our limited understanding of the aerodynamics and

Department of Mechanical Engineering and Applied Mechanics
GRASP Laboratory, University of Pennsylvania, Philadelphia, PA

the fluid-structure coupling in such aircrafts presents a formidable challenge [11]. Insect-style flapping wing vehicles provide the ability to hover in place while also enabling forward flight [2]. However, it is unclear that they represent a significant advantage over rotor crafts or ducted fans in terms of efficiency, endurance, or maneuverability, and they do incur a significant increase in complexity [44].

There are two configurations of rotor crafts that have gained acceptance in the research community. Co-axial rotor crafts, exemplified by the Skybotix Coax [9], are equipped with two counter-rotating, co-axial rotors and with a stabilizer bar [7]. Prototypes of less than 300 grams (without sensors or processors) with a hover time of nearly 20 minutes make them attractive for robotics applications. In addition, the stabilizer bar confers passive mechanical stability making them easy to control.

However, we argue (see next section) that multi-rotor aircrafts exemplified by quadrotors currently represent the best bet in terms of maneuverability and their ability to carry small payloads. Hence the rest of this paper will address the mechanics and control of quadrotors, and approaches to state estimation, mapping, planning, exploration and manipulation.

2 Rotor craft designs and scaling laws

In this section, we explore the effect of choosing length scales on the inertia, payload and ultimately angular and linear acceleration. In particular, we can analyze maneuverability in terms of the robot's ability to produce linear and angular accelerations from a hover state. If the characteristic length is L, the rotor radius R scales linearly with L. The mass scales as L^3 and the moments of inertia as L^5. On the other hand the lift or thrust, F, and drag, D, from the rotors scales with the cross-sectional area and the square of the blade-tip velocity, v. If the angular speed of the blades is defined by $\omega = \frac{v}{L}$, $F \sim \omega^2 L^4$ and $D \sim \omega^2 L^4$. The linear acceleration a scales as $a \sim \frac{\omega^2 L^4}{L^3} = \omega^2 L$.

For multi-rotor aircrafts like the quadrotor, thrusts from the rotors produces a moment with a moment arm L. Thus the angular acceleration $\alpha \sim \frac{\omega^2 L^5}{L^5} = \omega^2$. However, the rotor speed also scales with length since smaller motors produce less torque which limits their peak speed because of the drag resistance that also scales the same way as lift.

There are two commonly accepted approaches to scaling: Froude scaling and Mach scaling [55]. Mach scaling is used for compressible flows and essential assumes that the tip velocities are constant leading to $\omega \sim \frac{1}{R}$. In other words, the rotor speed scales inversely with length. Froude scaling is used for incompressible flows and assumes that for similar aircraft configurations, the Froude number, $\frac{v^2}{Lg}$, is constant. Here g is the acceleration due to gravity. This yields $\omega \sim \frac{1}{\sqrt{R}}$. Neither Froude or Mach number similitudes take motor characteristics into account. It is clear that the motor torque (τ) scales with length. The surface area, which goes as $R^2 \sim L^2$, and the volume of the core which scales as $R^3 \sim L^3$, are both important variables governing motor performance. It turns out Froude scaling ($\omega \sim \frac{1}{R}$) is consistent

with $\tau \sim L^2$ while Mach scaling is consistent with $\tau \sim L^3$. While the reality might be somewhere in between, these two limiting cases are meaningful for our analysis. Froude scaling suggests that the acceleration is independent of length while the angular acceleration $\alpha \sim L^{-1}$. On the other hand Mach scaling leads to the conclusion that $a \sim L$ while $\alpha \sim L^{-2}$. In other words, smaller aircrafts are much more agile. Note that this conclusion is based on the assumption that the propeller blades are rigid, the efficiency of the blade is independent of the length scale and the inertia associated with the blades can be neglected. These factors can be important but considering the inertia of the blade further emphasizes the benefits of scaling down — longer blades require larger cross-sections to minimize stresses and the inertia grows faster than L^5.

For other types of rotor crafts, including co-axial rotor crafts, the linear acceleration scales the same way but the angular acceleration does not. This is because the moment arm associated with the rotors is exactly L. This moment arm does not scale the same way with coaxial helicopters. Similarly the scaling law for conventional helicopters and ducted fans appears to be different. Thus if our objective is to build small, highly maneuverable aircrafts, multi-rotor helicopters like the quadrotor appear to be the best configuration. While rotorcrafts with six and eight rotors have been developed and are commercially available [3], the main benefits appear to be redundancy due to the number of rotors and increased safety because of the compactness of a six-rotor design over a four-rotor design.

There are three design points that are illustrative of the quadrotor configuration. We use the Pelican quadrotor from Ascending Technologies [3] equipped with sensors (approx. 2 kg gross weight, 0.75 m diameter, and 4000 rpm nominal rotor speed at hover), consuming approximately 400 W of power. The Hummingbird quadrotor from Ascending Technologies (500 grams gross weight, approximately 0.5 m diameter, and 5000 rpm nominal rotor speed at hover) consumes about 75 W. Attempts to develop a smaller quadrotor at the University of Maryland [35] suggest that a quad rotor without sensors of mass 62 grams, 0.075 m diameter and 9000 RPM rotor speed consumes a little over 10 W of power.

3 Control

3.1 Dynamics

The dynamics of quadrotors can be simplified to rigid body dynamic models with approximations to the aerodynamic forces [33]. In Fig. 1, the inertial frame, $\mathscr{A}$, is defined by the triad $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$ with $\mathbf{a}_3$ pointing upward. The body frame, $\mathscr{B}$, is attached to the center of mass of the quadrotor with $\mathbf{b}_1$ coinciding with the preferred forward direction and $\mathbf{b}_3$ perpendicular to the plane of the rotors pointing vertically up during perfect hover (see Fig. 1). Let $\mathbf{r}$ denote the position vector of the center of mass C in $\mathscr{A}$. The vehicle has mass m and the components of the inertia tensor is given by the 3×3 matrix J along along the principal axes $\mathbf{b}_i$. The rotation matrix describing $\mathscr{B}$ in $\mathscr{A}$ is given by $R \in SO(3)$, while the angular velocity of the vehicle, $\Omega \in \mathbb{R}^3$, is defined as

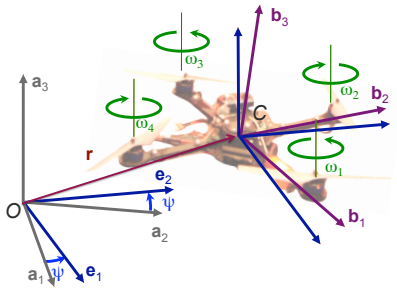

Fig. 1 The vehicle model. The position and orientation of the robot in the global frame are denoted by $\mathbf{r}$ and R, respectively.

$$\dot{R} = R\hat{\Omega}$$

where the operator $\hat{\cdot}$ is defined such that $\hat{x}y = x \times y$ for all $x, y \in \mathbb{R}^3$.

The forces on the system are gravity, in the $-\mathbf{a}_3$ direction, the lift forces from each of the rotors, F_i, and the drag moments from the rotors M_i, all in the $\mathbf{b_3}$ direction. Each rotor has an angular speed ω_i and produces a lift force $F_i = k_F \omega_i^2$ and drag moment $M_i = k_M \omega_i^2$. The constants, k_F and k_M, are related to the drag and lift coefficients, the cross sectional area and the rotor speed as discussed in Sect. 2. However, for a specific rotor, it is quite easy to determine these empirically. The thrust input is given by:

$$u_1 = \sum_{i=1}^{4} F_i$$

while the moment input vector is

$$\mathbf{u}_2 = L \begin{bmatrix} 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ \mu & -\mu & \mu & -\mu \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}$$

where L is the distance of the rotor axis from C, and $\mu = \frac{k_M}{Lk_F}$ is a non dimensional coefficient that relates the drag (moment) to the lift (force) produced by the propellor blades.

The dynamic model is given by:

$$m\ddot{\mathbf{r}} - mg\mathbf{e}_3 = u_1 R\mathbf{e}_3 \tag{1}$$

$$J\dot{\Omega} + \Omega \times J\Omega = \mathbf{u}_2 \tag{2}$$

where $\mathbf{e}_3 = [0, 0, 1]^{\mathrm{T}}$.

3.2 Control

The control problem, to track smooth trajectories $(R^{\text{des}}(t), \mathbf{r}^{\text{des}}(t)) \in SE(3)$, is challenging for several reasons. First, the system is underactuated — there are four inputs $(u_1, \mathbf{u}_2)$ while $SE(3)$ is six dimensional. Second, the aerodynamic model described above is only approximate. Finally, the inputs are themselves idealized. In practice, the motor controllers must generate the required speeds to realize these inputs. The dynamics of the motors and their interactions with the drag forces on the propellers can be quite difficult to model, although first order linear models are a useful approximation.

The first challenge, the underactuation, can be overcome by recognizing that the quadrotor is differentially flat. See [37, 39] for a discussion of differential flatness. To see this, we consider the outputs $\mathbf{r}$ and ψ as shown in Fig. 1, and show that we can write all state variables and inputs as functions of the outputs and their derivatives. Derivatives of $\mathbf{r}$ yield the velocity $\mathbf{v}$, and the acceleration,

$$\mathbf{a} = \frac{1}{m} u_1 \mathbf{b}_3 + \mathbf{g}$$

By writing the unit vector:

$$\mathbf{e}_1 = [\cos\psi,\ \sin\psi,\ 0]^{\mathrm{T}}$$

we can define the body frame from ψ and $\mathbf{a}$ as follows:

$$\mathbf{b}_3 = \frac{\mathbf{a} - \mathbf{g}}{\|\mathbf{a} - \mathbf{g}\|},\ \mathbf{b}_2 = \frac{\mathbf{b}_3 \times \mathbf{e}_1}{\|\mathbf{b}_3 \times \mathbf{e}_1\|},\ \mathbf{b}_1 = \mathbf{b}_2 \times \mathbf{b}_3$$

provided $\mathbf{e}_1 \times \mathbf{b}_3 \neq 0$. This defines the rotation matrix R as a function of $\mathbf{a}$ and ψ. To write the angular velocity and the inputs as a function of the outputs and their derivatives, we write the derivative of acceleration or jerk,

$$\mathbf{j} = \frac{1}{m} \dot{u}_1 \mathbf{b}_3 + \frac{1}{m} u_1\ \Omega \times \mathbf{b}_3$$

and finally, the snap or the derivative of jerk:

$$\mathbf{s} = \frac{1}{m} \ddot{u}_1 \mathbf{b}_3 + \frac{2}{m} \dot{u}_1\ \Omega \times \mathbf{b}_3 + \frac{1}{m} u_1\ \dot{\Omega} \times \mathbf{b}_3 + \frac{1}{m} u_1\ \Omega \times (\Omega \times \mathbf{b}_3)$$

where

$$\dot{\Omega} = J^{-1}(\mathbf{u}_2 - \Omega \times J\Omega)$$

From the equations above it is possible to verify that there is a diffeomorphism between the 18×1 vector:

$$\left[\mathbf{r}^{\mathrm{T}}, \mathbf{v}^{\mathrm{T}}, \mathbf{a}^{\mathrm{T}}, \mathbf{j}^{\mathrm{T}}, \mathbf{s}^{\mathrm{T}}, \psi^{\mathrm{T}}, \dot{\psi}^{\mathrm{T}}, \ddot{\psi}^{\mathrm{T}}\right]^{\mathrm{T}}$$

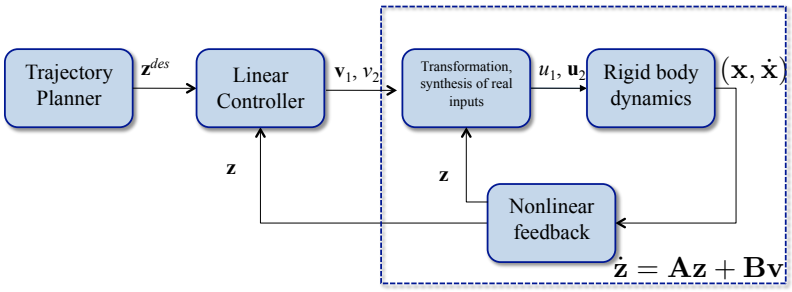

Fig. 2 Nonlinear feedback allows us to reduce the nonlinear system to a linear system (4).

and

$$R \times \left[\mathbf{r}^{\mathrm{T}}, \dot{\mathbf{r}}^{\mathrm{T}}, \Omega^{\mathrm{T}}, u_1, \dot{u}_1, \ddot{u}_1, \mathbf{u}_2^{\mathrm{T}}\right]^{\mathrm{T}}$$

Accordingly define the vector of flat outputs to be:

$$\mathbf{z} = [\mathbf{r}, \mathbf{v}, \mathbf{a}, \mathbf{j}, \psi, \dot{\psi}]^{\mathrm{T}} = [\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{z}_4, z_5, z_6]^{\mathrm{T}}$$

We can also define a vector of fictitious inputs

$$\mathbf{v} = \left[\mathbf{v}_1^{\mathrm{T}}, v_2\right]^{\mathrm{T}}$$

related to the original inputs by a nonlinear transformation of the form:

$$\begin{bmatrix} \mathbf{v}_1 \\ v_2 \end{bmatrix} = \mathbf{g}(\mathbf{z}) \begin{bmatrix} \ddot{u}_1 \\ \mathbf{u}_2 \end{bmatrix} + \mathbf{h}(\mathbf{z}) \tag{3}$$

so the state equations are linear:

$$\dot{\mathbf{z}} = \mathbf{A}\mathbf{z} + \mathbf{B}\mathbf{v} \tag{4}$$

with

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & 1 \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{I}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 0 \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}$$

This obviously makes the control problem trivial. See Fig. 2 for a graphical description of the controller design.

There are several difficulties following this naive approach. First, the linear controller based on (4) works only if the dynamics can be effectively linearized. This in turn depends on the cancelation of the dynamics in (3) which is difficult because the dynamic model only represents an approximation of the aerodynamic forces and our knowledge of the parameters in the model is not perfect. While parameter esti-

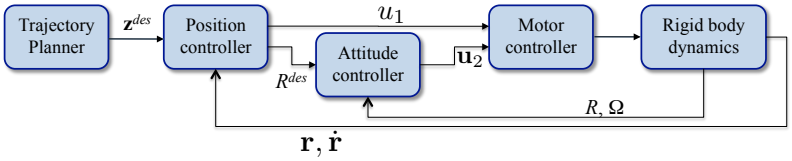

Fig. 3 The attitude controller achieves the desired orientation, which is in turn computed from the errors in position.

mation and adaptive control techniques (e.g., [40]) can be used to learn and adapt to these parameters, it is often not possible to get access to the low level signals involving higher order derivatives of the state and the inputs.

Indeed, the second challenge is to derive estimators that yield the extended state, $\mathbf{z}$, which includes not only the position and velocity, but also the acceleration and jerk. Knowledge of the thrust (u_1) and attitude ($\mathbf{b}_3$) allows us to estimate acceleration. Similarly, measuring the derivative of the thrust ($\dot{u}_1$), which is related to the rate of change of motor speeds, and the angular rates (Ω) allows us to estimate the jerk. However, this information is not usually available from motor drivers.

However, this model of exact linearization is useful since it allows us to design trajectories in the 18-dimensional space of flat outputs and their derivatives which are guaranteed to respect the dynamics and constraints we might want to impose on the state variables.

In most previous work [9, 15, 43], the control problem is addressed by decoupling the position control and attitude control subproblems as illustrated in Fig. 3. The position controller is obtained by projecting the position error (and its derivatives) along $\mathbf{b}_3$ and applying the input u_1 that cancels the gravitational force and provides the appropriate proportional plus derivative feedback:

$$u_1 = m\mathbf{b}_3^{\mathrm{T}} \left(\ddot{\mathbf{r}}^{\mathrm{des}} + K_d(\dot{\mathbf{r}}^{\mathrm{des}} - \dot{\mathbf{r}}^{\mathrm{des}}) + K_p(\mathbf{r}^{\mathrm{des}} - \mathbf{r}^{\mathrm{des}}) - \mathbf{g} \right). \tag{5}$$

The attitude controller varies based on the representation which is either using Euler angles, quaternions or rotation matrices. Euler angle representations have singularities and are suitable only for small excursions from the hover position. In most cases, it is sufficient to use linear controllers that are based on the linearization of the plant dynamics around the hover position [9, 16, 29, 33, 43]. The use of quaternions permits the exact cancellation of dynamics and a nonlinear controller that is exponentially stable almost everywhere in $SO(3)$ [53]. A similar result with rotation matrices is available in [23]. In both these papers, the error is defined on the rotation group and does not require the error to be small.

In [23], the two controllers are shown to result in a nonlinear controller that explicitly track trajectories in $SE(3)$. The key idea is to design exponentially converging controllers in $SO(3)$ using an accurate measure of the error in rotations instead of taking linear approximations:

$$\hat{e}_R = \frac{1}{2}\left((R^{\text{des}})^{\text{T}}R - R^{\text{T}}R^{\text{des}}\right) \tag{6}$$

which yields a skew-symmetric matrix representing the axis of rotation required to go from R to R^{des} and with the magnitude that is equal to the sine of the angle of rotation. Computing the proportional plus derivative of the error on $SO(3)$ and compensating for the nonlinear inertial terms gives us:

$$\mathbf{u}_2 = J(-k_R e_R - k_\Omega e_\Omega) + \Omega \times J\Omega, \tag{7}$$

If we do not consider constraints on the state or the inputs, (6-7) achieve asymptotic convergence to specified trajectories in $SE(3)$ [23]. From a practical standpoint it is possible to neglect the nonlinear $\Omega \times J\Omega$ term in the controller and achieve satisfactory performance [27]. Finally, as shown in [29], it is possible to combine this controller with attitude only controllers to fly through vertical windows or land on inclined perches with close to zero normal velocity.

trajectory controllers allows the robot to build up momentum and reorient itself while coasting with the generated momentum.

3.3 Adaptation and Learning

The dynamic models suffer from two types of limitations. First, such parameters as the location of the center of mass, the moments of inertia and the motor time constants are not precisely known. Second, the aerodynamic models are only approximate.

The first difficulty is overcome using parameter estimation algorithms. Because the unknown parameters appear linearly in the equations of motion (as in the case for robot manipulators [10, 38, 54]), we can write the state equations in discrete time as follows,

$$\mathbf{y}_{k+1} = \theta^T \Phi_k$$

θ is the *parameter vector*, Φ_k and $\mathbf{y}_k$ are the regressor and the measurement at the k^{th} time step. A simple linear least-squares method can be used to estimate the unknown parameters as shown in [28] either in a batch or in a recursive algorithm provided the dynamics are persistently excited. These methods can also be used to determine the offsets in IMU readings and for online calibration [48].

Adapting to varying aerodynamic conditions such as those encountered in narrow passages or perturbations due to wind gusts is harder because of the interaction between the time scales of estimation and control. Model Reference Adaptive Control techniques can be used in such settings, although it is necessary to get good measurements of the inputs (motor currents or speeds) and state variables for effective adaptation.

Iterative learning has been used effectively in [26, 29] for acrobatic maneuvers. Such techniques allow the robot to learn trajectories and inputs without knowing a precise aerodynamic model.

Regardless of the specific platform, it is unlikely that a conventional model-based approach to control can work without a robust adaptation mechanism. The small length scales and inertias lead to variations in dynamics that are very difficult to model and impossible to reason about in real time. However it is also unlikely if purely data-driven approaches can be used for control of micro-UAVs. While apprenticeship methods and variants of reinforcement learning algorithms (see, for example, [1]) have achieved remarkable results, they require an expert human operator to generate data for model and control identification. Further, it is unclear if these methods can generalize the results to cases not a priori encountered, where training data is not available. Indeed, in much of the work considered in our own group [29, 51], it is very challenging if not impossible for a trained human operator to fly the robot in the specified manner.

4 Planning

Incremental search [24] and sampling based techniques [22], which are excellent for planning in configuration spaces, are not particularly well-suited for planning motions for underactuated systems. RRT methods and their variants can solve problems with dynamic constraints. For example, in [48], a RRT planner is used to generate trajectories online through a cluttered environment with models acquired by a laser and a camera, but for dynamic models obtained by linearization around the hover operating point. However, the complexity of a 12-dimensional state space with four inputs makes such techniques impractical for planning fast motions through constrained environments. Smaller problems, for example planning motions in the plane, can be solved using reachability algorithms [13], but it is difficult to explore using the full state space using such approaches.

An alternative approach is to use a combination of planning algorithms for configuration spaces along with controller synthesis techniques to ensure the UAVs can execute the planned trajectory. For example, RRT-like techniques have been used with LQR-like control synthesis techniques to find trajectories and sufficing (and even optimal) control policies [50]. Similarly, uncertainty in dynamics and estimation can be addressed using LQG techniques with motion planners [5]. However, techniques like this have yet to be applied to 3-D motion planning of UAVs.

Model predictive control (MPC) techniques represent a third approach that can be used to solve planning and control problems for underactuated systems [20, 56]. These techniques are promising since they combine open loop (optimal) motion planning with feedback control — by generating open loop trajectories based on environmental models periodically with a time interval that is much smaller than the horizon of planning, corrective motions can be generated to accommodate changes in the environment. However, with such approaches, convergence guarantees are difficult to prove. It is possible to prove stability of the MPC algorithm when the linearized model is fully controllable about the goal position [56] (which is generally possible when the goal corresponds to a static hover position), or if a control Lyapunov function can be synthesized for goal positions [17]. Guarantees aside, the synthesis of optimal control solutions even with a finite horizon and a terminal cost

function can be difficult with limiting on-board processing resources. Thus it appears to be difficult to directly apply such techniques to the trajectory generation of a quadrotor with guarantees.

It appears that a hierarchical approach that combines incremental search or sampling based techniques in configuration space with optimal control techniques that refines configuration space trajectories in state space is the best framework to solve such problems. If a configuration space planner can be used first to establish waypoints and constraints, optimal trajectories that respect these constraints and the dynamics of the UAV can be generated as a second step. In [27], the property of differential flatness is used to develop an algorithm that enables the generation of optimal trajectories through a series of keyframes or waypoints in the set of positions and orientations, while ensuring safe passage through specified corridors and satisfying constraints on achievable velocities, accelerations and inputs. Since the cost function and all the constraints can be written as algebraic functions of the flat output vector, $\mathbf{z}$, the general setting reduces to solving the problem:

$$\min_{\mathbf{v}(t)} \int_0^{\mathrm{T}} L(\mathbf{z})dt, \;\; s.t.\; g(\mathbf{z}) \leq 0 \tag{8}$$

A simple choice for $L(\mathbf{z})$ is the square of the norm of the input vector, which turns out to be the equivalent of finding the trajectory that minimizes the snap and the yaw acceleration along the trajectory. It also has the added benefit of yielding a convex cost function. Recall that trajectories in this flat space automatically satisfy the dynamic equations of motion. Thus the only constraints in $g(\mathbf{z}) \leq 0$ are those on the position (obstacles), velocity (maximum angular rates because of gyro saturation), accelerations (saturation of the IMU), and inputs (propellors can only exert positive lift). All except the position constraints are linear. By linearizing the position constraints the optimization in (8) becomes a convex program. The unconstrained problem, the minimum snap trajectory optimization, yields an analytical solution - a seventh degree polynomial function of time for which we can introduce a polynomial basis for the trajectories. We can similarly use polynomial functions (if necessary of higher order) to satisfy all the constraints in (8). The resulting trajectories have interesting time scaling properties [27] and can be refined efficiently for different values of T to obtain the fastest trajectory to satisfy all the constraints. Finally the quadratic program can be solved in real time quite efficiently, and even in a distributed MPC-like setting for multiple quadrotors at speeds approaching 20 Hz. [51].

5 State Estimation and Perception

State estimation is a fundamental necessity for any application involving autonomous UAVs. However, platform design, mobility and payload constraints place considerable restrictions on available computational resources and sensing. The agility and three-dimensional mobility of the vehicle require sensors that provide low-latency information about the three-dimensional environment surrounding the vehicle. Al-

though in open outdoor domains, this problem is seemingly solved with onboard GPS, IMU and camera sensors [45, 47], indoor domains and cluttered outdoor environments still pose a considerable challenge. In such complex environments, the vehicle must be able to localize, detect or perceive obstacles, generate trajectories to navigate around the obstacles and track the trajectories with reasonable accuracy. Any failure to successfully achieve any of these requirements may in fact lead to a complete failure of the vehicle. Further, outdoor environmental effects (e.g. obscurants [46], wind, direct sunlight, GPS-shadowing) and indoor structural considerations (e.g. obstacles, tight corridors, vehicle-induced wind [33]) can challenge the consistency and accuracy of estimation algorithms that are not designed to directly consider these issues.

The fusion of information from multiple onboard sensors such as IMU, laser and cameras (monocular, stereo and RGB-D) do much to address these issues but come with a cost on processing demands, payload and power. Thus there is a real need to find a balance between sensor availability, onboard and offboard processing and operating conditions (which in turn lead to restrictions on the kind of environments in which the UAV can operate).

Initial developments in the area focused on systems capable of navigating indoor environments with algorithms leveraging laser and IMU information to generate a map and localize within the map [4, 14]. Processing for estimation and mapping is shared between local and external computational resources. Unlike the previous work, a monocular camera approach is employed for full pose estimation and localization of ground terrain in GPS-denied environments in [8] but with offboard processing. However, a major concern with offboard processing is the need to maintain uninterrupted, low-latency communication. While this is possible in some indoor and outdoor environments, it inhibits the ability of the system to operate autonomously throughout more general and complex environments. Additionally, the added time cost of external information exchange reduces the performance of the onboard feedback control due to the communication incurred time-delays.

To address these issues, in [48] we considered a similar problem but required the development of an implementation that permitted all processing to occur on the vehicle in real-time. The advancements made in this work were in the form of system design and algorithm optimization to permit autonomous navigation using an IMU, camera and laser to generate three-dimensional maps throughout large and multi-story environments using only limited onboard processing. Further, as all processing occurred in real-time and on the vehicle, we were able to leverage the feedback from the state estimation to drive model-based adaption to account for external disturbances due to gusting wind and ground effects.

Thus far, the discussion focuses on autonomous navigation, where the vehicle plans and controls to goals provided by an external entity. A remaining question is the introduction of perception, planning and control to permit autonomous exploration, where perception algorithms must also allow the UAV to reason about the environment to determine control policies that will yield maximal information for mapping. However, a major challenge in moving toward this direction is the lack of three-dimensional sensors that can be mounted on UAVs, which are required for

3-D exploration. Unfortunately, rich sensor sources such as three-dimensional laser range finders and omni-directional cameras either do not fit the vehicle payload constraints or are prohibitive given the limited computational resources. As such, it is necessary to focus on new algorithmic methods to explore an environment given limited sensing and computational resources. A current strategy we are pursuing in ongoing research [49] is the application of stochastic-differential equations to establish information frontiers between spatial regions that represent the known, explored environment and regions that represent the unexplored environment. The approach strives to find a balance between the computationally complexity of analyzing a full three-dimensional map and the limited field-of-view of onboard sensing. The area of autonomous exploration and perception is clearly an area with rich research possibilities that will become increasingly viable as computing and sensing options improve in time.

6 Other Challenges

6.1 Scaling and SWaP constraints

One of the key challenges in creating small autonomous UAVs are the so-called size, weight and power constraints. Packaging constraints are severe. Sensors and processors have to be smaller due to the limitations on payload. Because of this, it is difficult to create autonomous quadrotors (with onboard computation and sensing) at small length scales. The smallest autonomous quadrotors capable of exploring, mapping and scouting an unknown three-dimensional building-like environment have a characteristic length of approximately 0.75 m, mass of a little less than 2 kg., and power consumptions over 400 Watts leading to a mission life of around 10-20 minutes [48]. The main reason for the size is the need to carry three-dimensional sensors like Hokuyo laser range finders or Microsoft Kinect cameras. This in turn leads to high power consumption. Many impressive advances have been made in mapping and estimation for autonomous navigation using just an IMU and a camera [8]. Recent results point to algorithms that yield estimates of 3-D metric information from just monocular vision combined with a good IMU [21, 36]. This suggests that the sensor payload challenges associated with scaling can be overcome in the near future.

However, the net payload constraints are still significant if the UAV needs to be able to transport or manipulate a payload. Since the linear acceleration scales with L (Sect. 2), it is impossible to design small UAVs that are able to overcome this fundamental constraint. Current UAVs with $L \sim 1$m have a maximum payload of around 1 kg. One way to overcome this constraint is by using multiple UAVs to cooperatively transport or manipulate payloads. Recent work suggests that the challenges in coordinating multiple UAVs and adapting individual vehicles to constraints imposed by other vehicles is possible in different settings ranging from payloads suspended from UAVs [12, 18, 19, 32] to payloads rigidly grasped by UAVs [30].

6.2 Grasping and manipulation

There are many challenges in aerial grasping for micro-UAVs. The biggest challenge arises from their limited payload. While multiple UAVs can be coordinated to carry payloads with grippers [30], the end effector or gripper has to be light weight and capable of grasping complex shapes. Second, the dynamics of the robot are significantly altered by the addition of payloads. While this can be beneficial to tasks when aerial robots need to sense the payload that has been grasped, it is important to also be able to compensate for and adapt to changes in the dynamics caused by the grasped payload. It is clear that the design of claws for grasping represents a challenging mechanism design problem where the compliance and damping must be finely tuned to grasping. Finally, all the challenges associated with grasping objects (approaching, contacting, and securing the grasp) make this a significant challenge.

Preliminary work in this direction has appeared in conferences over the last two years. The difficulties associated with the analysis of the flight dynamics and stability are explained with the help of an approximate model in [42]. The mechanics and design for aerial grasping are addressed in [28, 30]. Parameter estimation methods for estimating the grasped payload and the ability to adapt to the payloads are investigated in [28]. The application to construction of structures is discussed in [25] in which the sensed disturbance forces are used to verify successful grasping and assembly. Micro-UAVs afford opportunities for truly three-dimensional grasping since they can, in principle, approach objects or assemblies from any direction, and because they can sense disturbance forces without additional sensors. This is a fertile area of future research.

6.3 Adaptation to complex environments with changing dynamics

As discussed earlier, it is very difficult to model micro-UAVs with a high degree of precision because of the complexity of modeling air drag, the interactions between the motor, rotor and the fluid through which the propellor blades must move, the dynamics of the flexible propellor blade and the different nonlinearities and saturation effects in the sensors and actuators. And such difficulties get compounded when the rigid body dynamics interact with the aero dynamics and the fluid-structure coupling effects become significant, as is the case in flapping-wing vehicles or rotor crafts with long blades. As discussed earlier in Sect. 3.3, adaptive control and iterative learning techniques can be used to handle some of these challenges. However, parameterizing the set of uncertainties and ensuring the appropriate level of sensing and actuation to identify these parameters may not always be possible. Methods such as the ones described in [26, 28, 29] are good starting points for such studies.

The effects of changes in the aerodynamics in three-dimensional environments are much harder to study. A study of wind gusts in [57] illustrates the challenges in modeling and experimentation. For small aircrafts, small, local variations in wind conditions can be significant. Transitions between indoor and outdoor environments can induce large perturbations. Even without wind gusts, changes in elevation can dramatically alter the lift generated by individual propellors resulting in significant disturbances to the vehicle. Some of these phenomena are studied for mod-

estly changing environments in [48] where the inputs required to compensate for the changes can be parameterized by a small set of trim parameters. In these studies the sensed information was limited to gross position and velocity information which in turn limits the level of adaptation that is possible. If aerial vehicles are to become as reliable and easy-to-use as ground vehicles, it is necessary to develop techniques that will enable safe and robust low-level navigation behaviors in complex environments.

7 Conclusion

Micro UAVs are potentially game changers in robotics. They can operate in constrained three-dimensional environments, explore and map multi-story buildings, manipulate and transport objects, and even perform such tasks as assembly. Our recent experiments with quadrotors in collapsed buildings in Sendai, Japan in July 2011 [34] demonstrated many benefits of using autonomous quadrotors for mapping unknown environments, searching in collapsed buildings and exploration in settings that are too dangerous for human rescue workers. Just as the advent of mobile robots led to a flurry of activity with new research problem areas, micro-UAVs will inevitably lead robotics research in new and exciting directions.

References

1. Abbeel, P.: Apprenticeship learning and reinforcement learning with application to robotic control. Ph.D. thesis, Stanford University, Stanford, CA (2008)
2. Aeroenvironment nano hummingbird (2011). URL `http://www.avinc.com/nano`
3. Ascending Technologies, GmbH. URL `http://www.asctec.de`
4. Bachrach, A.G.: Autonomous flight in unstructured and unknown indoor environments. Master's thesis, MIT, Cambridge, MA (2009)
5. van der berg, J., Abbeel, P., Goldberg, K.: Lqg-mp: Optimized path planning for robots with motion uncertainty and imperfect state information. Intl. J. Robot. Research **30**(7), 895–913 (2011). DOI 10.1177/0278364911406562
6. Bermes, C.: Design and dynamic modeling of autonomous coaxial micro helicopters. Ph.D. thesis, ETH Zurich, Switzerland (2010)
7. Bermes, C., Schafroth, D., Bouabdallah, S., Siegwart, R.: Modular simulation model for coaxial rotary wing mavs. In: Proc. of The 2nd International Symposium on Unmanned Aerial Vehicles (2009)
8. Blosch, M., Weiss, S., Scaramuzza, D., Siegwart, R.: Vision based MAV navigation in unknown and unstructured environments. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 21–28. Anchorage, AK (2010)
9. Bouabdallah, S.: Design and control of quadrotors with applications to autonomous flying. Ph.D. thesis, Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland (2007)
10. Craig, J., Hsu, P., Sastry, S.: Adaptive control of mechanical manipulators. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., vol. 3, pp. 190 – 195 (1986). DOI 10.1109/ROBOT.1986.1087661
11. Faruque, I., Humbert, J.S.: Dipteran insect flight dynamics. part 2: Lateral-directional motion about hover. Journal of Theoretical Biology **265**(3), 306 – 313 (2010). DOI DOI:10.1016/j.jtbi.2010.05.003
12. Fink, J., Michael, N., Kim, S., Kumar, V.: Planning and control for cooperative manipulation and transportation with aerial robots. Intl. J. Robot. Research **30**(3) (2011)

13. Gillula, J.H., Huang, H., Vitus, M.P., Tomlin, C.J.: Design of guaranteed safe maneuvers using reachable sets: Autonomous quadrotor aerobatics in theory and practice. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 1649–1654. Anchorage, AK (2010)
14. Grzonka, S., Grisetti, G., Burgard, W.: Towards a navigation system for autonomous indoor flying. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 2878–2883. Kobe, Japan (2009)
15. Gurdan, D., Stumpf, J., Achtelik, M., Doth, K., Hirzinger, G., Rus, D.: Energy-efficient autonomous four-rotor flying robot controlled at 1 khz. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom. Roma, Italy (2007)
16. Huang, H., Hoffman, G.M., Waslander, S.L., Tomlin, C.J.: Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 3277–3282. Kobe, Japan (2009)
17. Jadbabaie, A., Hauser, J.: On the stability of receding horizon control with a general terminal cost. IEEE Trans. Autom. Control **50**(5), 674 – 678 (2005)
18. Jiang, Q., Kumar, V.: The direct kinematics of objects suspended from cables,. In: ASME Intl. Design Eng. Tech. Conf. & Comput. and Inf. in Eng. Conf. (2010)
19. Jiang, Q., Kumar, V.: The inverse kinematics of 3-d towing. In: J. Lenarcic, M.M. Stanisic (eds.) Advances in Robot Kinematics, pp. 321–328 (2010)
20. Kim, H., Shim, D., Sastry, S.: Nonlinear model predictive tracking control for rotorcraft-based unmanned aerial vehicles. In: Proc. of the Amer. Control Conf., vol. 5, pp. 3576–3581. Anchorage, AK (2002)
21. Kneip, L., Martinelli, A., Weiss, S., Scaramuzza, D., Siegwart, R.: Closed-form solution for absolute scale velocity determination combining inertial measurements and a single feature correspondence. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 4546–4553 (2011)
22. Lavalle, S.M.: Planning Algorithms. Cambridge University Press (2006)
23. Lee, T., Leok, M., McClamroch, N.: Geometric tracking control of a quadrotor uav on SE(3). In: Proc. of the IEEE Conf. on Decision and Control (2010)
24. Likhachev, M., Gordon, G., Thrun, S.: ARA*: Anytime A* with provable bounds on sub-optimality. Advances in Neural Information Processing Systems **16** (2003)
25. Lindsey, Q., Mellinger, D., Kumar, V.: Construction of cubic structures with quadrotor teams. In: Proc. of Robot.: Sci. and Syst. Los Angeles, CA (2011)
26. Lupashin, S., Schollig, A., Sherback, M., D'Andrea, R.: A simple learning strategy for high-speed quadrocopter multi-flips. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 1642–1648. Anchorage, AK (2010)
27. Mellinger, D., Kumar, V.: Minimum snap trajectory generation and control for quadrotors. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom. Shanghai, China (2011)
28. Mellinger, D., Lindsey, Q., Shomin, M., Kumar, V.: Design, modeling, estimation and control for aerial grasping and manipulation. In: Proc. of the IEEE/RSJ Intl. Conf. on Intell. Robots and Syst. San Francisco, CA (2011). To Appear
29. Mellinger, D., Michael, N., Kumar, V.: Trajectory generation and control for precise aggressive maneuvers with quadrotors. In: Proc. of the Intl. Sym. on Exp. Robot. Delhi, India (2010)
30. Mellinger, D., Shomin, M., Michael, N., Kumar, V.: Cooperative grasping and transport using multiple quadrotors. In: Intl. Sym. on Distributed Auton. Syst. Lausanne, Switzerland (2010)
31. Mettler, B.: Modeling small-scale unmanned rotorcraft for advanced flight control design. Ph.D. thesis, Carnegie Mellon University, Pittsburgh, PA (2001)
32. Michael, N., Fink, J., Kumar, V.: Cooperative manipulation and transportation with aerial robots. Auton. Robots **30**(1), 73–86 (2011)
33. Michael, N., Mellinger, D., Lindsey, Q., Kumar, V.: The grasp multiple micro uav testbed. IEEE Robot. Autom. Mag. (2010)
34. Michael, N., Tadokoro, S., Nagatani, K., Ohno, K.: Experiments with air ground coordination for search and rescue in collapsed buildings (2011). Working Paper
35. Miller, D.S., Gremillion, G., Ranganathan, B., Samuel, P.D., Zarovy, S., Costello, M., Mehta, A., Humbert, J.S.: Challenges present in the development and stabilization of a micro quadrotor helicopter. In: Autonomous Weapons Summit and GNC Challenges for Miniature Autonomous Systems Workshop (2010)

36. Mourikis, A.I., Trawny, N., Roumeliotis, S.I., Johnson, A.E., Ansar, A., Matthies, L.: Vision-aided inertial navigation for spacecraft entry, descent, and landing. IEEE Trans. Robot. **25**(2), 264–280 (2009)
37. Murray, R.M., Rathinam, M., Sluis, W.: Differential flatness of mechanical control systems: A catalog of prototype systems. In: Proceedings of the 1995 ASME International Congress and Exposition (1995)
38. Niemeyer, G., Slotine, J.J.: Performance in adaptive manipulator control. In: Proc. of the IEEE Conf. on Decision and Control, vol. 2, pp. 1585–1591 (1988). DOI 10.1109/CDC.1988.194595
39. Nieuwstadt, M.J.V., Murray, R.M.: Real-time trajectory generation for differentially flat systems. International Journal of Robust and Nonlinear Control **8**, 995–1020 (1998)
40. Ortega, R., Spong, M.W.: Adaptive motion control of rigid robots: a tutorial. In: Proc. of the IEEE Conf. on Decision and Control, vol. 2, pp. 1575–1584 (1988). DOI 10.1109/CDC.1988.194594
41. Pines, D., Bohorquez, F.: Challenges facing future micro air vehicle development. AIAA Journal of Aircraft **43**(2), 290–305 (2006)
42. Pounds, P., Dollar, A.: Hovering stability of helicopters with elastic constraints. In: ASME Dynamic Systems and Control Conference (2010)
43. Purwin, O., D'Andrea, R.: Performing aggressive maneuvers using iterative learning control. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 1731–1736. Kobe, Japan (2009)
44. Ratti, J., Vachtsevanos, G.: Towards energy efficiency in micro hovering air vehicles. In: IEEE Aero. Conf., pp. 1–8. Big Sky, MT (2011)
45. Saripalli, S., Montgomery, J.F., Sukhatme, G.S.: Vision-based autonomous landing of an unmanned aerial vehicle. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., pp. 2799–2804. Washington, DC (2002)
46. Sevcik, K.W., Kuntz, N., Oh, P.Y.: Exploring the effect of obscurants on safe landing zone indentification. J. Intell. Robotic Syst. **57**(1-4), 281–295 (2010)
47. Sharp, C.S., Shakernia, O., Sastry, S.S.: A vision system for landing an unmanned aerial vehicle. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom., vol. 2, pp. 1720–1727. Seoul, Korea (2001)
48. Shen, S., Michael, N., Kumar, V.: 3D estimation and control for autonomous flight with constrained computation. In: Proc. of the IEEE Intl. Conf. on Robot. and Autom. Shanghai, China (2011)
49. Shen, S., Michael, N., Kumar, V.: Exploration and control for autonomous mapping with aerial robots. Tech. rep., University of Pennsylvania (2011)
50. Tedrake, R.: LQR-Trees: Feedback motion planning on sparse randomized trees. In: Proc. of Robot.: Sci. and Syst. Seattle, WA (2009)
51. Turpin, M., Michael, N., Kumar, V.: Trajectory design and control for aggressive formation flight with quadrotors. In: Proc. of the Intl. Sym. of Robot. Research. Flagstaff, AZ (2011)
52. U.S. Military Unmanned Aerial Vehicles (UAV) Market Forecast 2010–2015 (2011). URL `http://www.uavmarketresearch.com/`
53. Wen, J., Kreutz-Delgado, K.: The attitude control problem. IEEE Trans. Autom. Control **36**(10), 1148–1162 (1991)
54. Whitcomb, L., Rizzi, A., Koditschek, D.: Comparative experiments with a new adaptive controller for robot arms. IEEE Trans. Robot. Autom. **9**(1), 59–70 (1993). DOI 10.1109/70.210795
55. Wolowicz, C.H., Bowman, J.S., Gilbert, W.P.: Similitude requirements and scaling relationships as applied to model testing. Tech. rep., NASA (1979)
56. Yu, J., Jadbabaie, A., Primbs, J., Huang, Y.: Comparison of nonlinear control design techniques on a model of the caltech ducted fan. In: IFAC World Congress, IFAC-2c-112, pp. 53–58 (1999)
57. Zarovy, S., Costello, M., Mehta, A., Flynn, A., Gremillion, G., Miller, D., Ranganathan, B., Humbert, J.S., Samuel, P.: Experimental study of gust effects on micro air vehicles. In: AIAA Conference on Atmospheric Flight Mechanics, AIAA-2010-7818. American Institute of Aeronautics and Astronautics (2010)