View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by NASA Technical Reports Server

Coordinated Turn Trajectory Generation and Tracking Control for Multi-rotors Operating in Urban Environment

Vahram Stepanyan [1]
Universities Space Research Association, Columbia, MD 21046

Kalmanje Krishnakumar [2] and Corey Ippolito [3]
NASA Ames Research Center, Moffett Field, CA 94035

The paper presents an efficient trajectory generation and tracking approach for multi-rotor air vehicles operating in urban environment, which takes into account uncertainties in the urban wind field and in the vehicle's parameters. Generated trajectories are sufficiently smooth, based on the differential flatness of the vehicle's dynamics and optimal in the sense of minimum agility and time. They pass through given set of way points, guarantee flight without a side-slip, and satisfy vehicle's dynamics and actuator constraints. In addition, an algorithm is presented to compute the required power to traverse the generated trajectory. Presented algorithms are implementable in real time using on-board computers. They do not take into account the vehicle's existing flight controller, hence there is no guarantee that the controller will be able to provide acceptable tracking of the generated trajectory, especially in the presence of atmospheric disturbances. To this end, we propose an adaptive augmentation algorithm to improve vehicle's performance by taking into account the effects of disturbances and on-line estimates of vehicle's existing flight controller's gains. The algorithms

[1] Principal Scientist, NASA Ames Research Center/Mail Stop 269/1, AIAA Senior Member, email: vahram.stepanyan@nasa.gov
[2] Technical Lead, Intelligent Systems Division, NASA Ames Research Center/Mail Stop 269/3, AIAA Associate Fellow, email: kalmanje.krishnakumar@nasa.gov
[3] Task Lead, Intelligent Systems Division, NASA Ames Research Center/Mail Stop 269/3, AIAA Senior Member, email: corey.a.ippolito@nasa.gov

have been verified by simulations using DJI S1000 octocopter's model.

I. Introduction

Drones are becoming increasingly popular for research, commercial and military applications due to their affordability resulting from their small size, low cost and simple hardware structure. One of the critical aspects of these uses is the reliable navigation and control of the drones in an urban environment, where a complex and uncertain wind field can be dangerous for flight operations.

Many efforts have been directed to compensating for the wind effects in control settings. In [7], the wind effects are estimated by a nonlinear disturbance observer and used to design a path following controller. In [1], a controller is presented to achieve trajectory tracking for kinematic models of unmanned aerial vehicles. In [8], a linear observer with integral action is used to stabilize a quad-rotor at hover flight taking into account only the wind effects in roll and pitch angles. In [4], L1 adaptive control augmentation of the baseline outer-loop controller is used for position tracking in the presence of wind disturbances. In [11], path-following guidance method is presented in the presence of quasi-constant but unknown wind disturbances. A quaternion-based adaptive attitude control for a quad-rotor in the presence of external disturbances is considered in [13].

Trajectory generation problems in the wind field have received less attention. In [3], a time optimal trajectory generation method in known constant in time and linear in space wind field is presented for the kinematic model of quad-rotors. In [17] and [12], minimum time algorithm and trochoid curves are respectively used for path planning in known steady uniform wind fields for fixed wing UAVs. These approaches are not applicable in urban environment since the wind field may not be uniform or known. For this reason wind estimation techniques have to be employed to accommodate for the trajectory generation.

One way to estimate the wind components is using air data measurements from available on-board sensors (see for example [2], [5] and references therein). While this approach may be suitable for fixed wing UAVs, no reliable air data sensors have been reported for the multi-rotor UAVs in the literature to our best knowledge. In [16], we have proposed an adaptive wind estimation based

approach to generate a feasible trajectory for multi-rotors. However, this trajectory generation methods do not address the air vehicle desired orientation design along the trajectory, which is critical from the sensors field of view perspective and for the operations effectiveness.

In this paper we build on the results of [16] and present an efficient trajectory generation and tracking approach for multi-rotor air vehicles operating in urban environment, which takes into account uncertainties in the urban wind field and in the vehicle's parameters. It is assumed that the inertial position and velocity of the vehicle's center of mass, orientation angles and the angular rates are available for feedback through on-board sensor package. In addition, it is assumed that the estimation algorithms from [16] are implemented on the on-board computer to accurately estimate the atmospheric drag forces and moments, the wind linear and angular velocities and accelerations in real time. The approach includes a set of real-time algorithms to generate trajectories of the vehicle's center of mass and yaw angle, compute the required power to traverse the generated trajectory and augment the existing flight controller with an adaptive outer-loop. The generated trajectories are sufficiently smooth, based on the differential flatness of the vehicle's dynamics and optimal in the sense of minimum agility and time. In addition to passing through a given set of way points, they guarantee flight without a side-slip, and satisfy vehicle's dynamic and actuators constraints. The trajectory generation and power computation algorithms are computationally inexpensive and easily implementable in real-time using on-board computers. They do not depend on the vehicle's existing flight controller, and can be implemented on any vehicle. However, the accuracy of tracking these trajectories may vary vehicle-to-vehicle, especially in the presence of atmospheric disturbances. The proposed adaptive augmentation algorithm provides a robust and accurate tracking performance by explicitly taking into account the effects of atmospheric disturbances and on-line estimates of uncertainties in the vehicle's control coefficients. The benefits of the approach have been demonstrated through the simulation for an octocopter flying in the cityscape with a simulated wind field.

II. Motivation

Many drone operations require directed sensors such as camera or Lidar when operating in GPS degraded environment. These sensors need to be aligned with the direction of motion in order

to provide necessary information for the operation. However majority of path/trajectory planning algorithms care only about the vehicle's center of gravity position in the environment, which may result in loosing some obstacles from the field of view. Therefore there is a need to define the vehicle's orientation along the trajectory at any time instance. One way of doing it is to use the differential flatness properties of the multi-rotor's dynamics following [9]. Two singularities arise when going forward with this approach. First singularity results from the fact that when way-points do not make a straight line, the velocity vector has to instantaneously change the direction at a way point, which is possible when the velocity is zero there. This is not an efficient planning strategy. In addition, when the velocity is zero, the air vehicle's orientation cannot be defined using approach given in [9]. This scenario is displayed in Fig. 1. Second singularity results when the air vehicle flies vertically up or down. In this case the velocity vector is parallel to the thrust and the approach from [9] is not applicable as well.

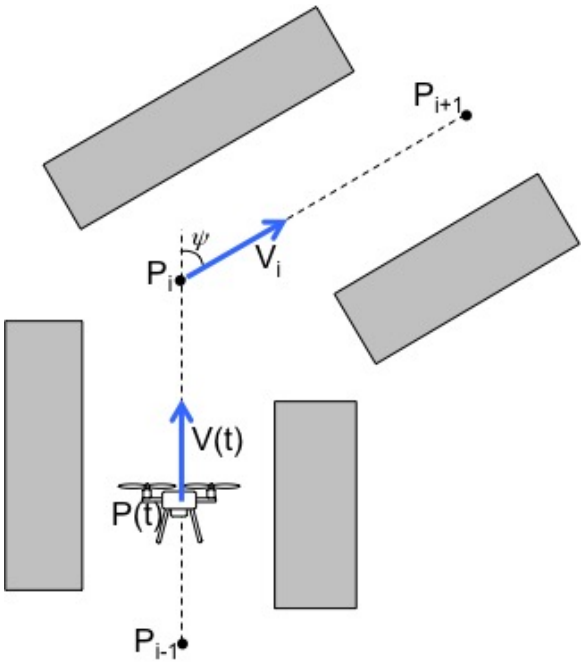

Fig. 1 Instantaneous velocity vector rotation at the way-point.

III. Drone's Dynamic Model

A. Equations of Motion

The dynamics of the multi-rotor vehicle's center of mass in the East-North-Up Earth (inertial) frame (F_E) are given by

$$\begin{aligned} \dot{\boldsymbol{r}}(t) &= \boldsymbol{v}(t) \\ m\dot{\boldsymbol{v}}(t) &= R_{B/E}(t)\boldsymbol{e}_3^B f_T(t) + \boldsymbol{f}_D(t) + m\boldsymbol{g}\,, \end{aligned} \tag{1}$$

where $\boldsymbol{r}(t) = [x(t)\ y(t)\ z(t)]^\top$ is the position of the center of mass in F_E, $\boldsymbol{v}(t) = [v_x(t)\ v_y(t)\ v_z(t)]^\top$ is the inertial velocity, m is the mass, $f_T(t)$ is the total thrust generated by the rotors, $R_{B/E}(t)$ is the rotation matrix from the body frame F_B (Forward-Left-Up) to F_E, $\boldsymbol{e}_3^B = [0\ 0\ 1]^\top$ is the third unit vector of F_B, $\boldsymbol{f}_D(t)$ is the aerodynamic drag force and $\boldsymbol{g} = [0\ 0\ -g]^\top$ is the gravity acceleration.

The vehicle's rotational dynamics about the center of mass are given in the frame F_B as

$$\begin{aligned} \dot{R}_{B/E}(t) &= R_{B/E}(t)\boldsymbol{\omega}^\times(t) \\ J\dot{\boldsymbol{\omega}}(t) &= -\boldsymbol{\omega}(t) \times J\boldsymbol{\omega}(t) + J_m\omega_m(t)\bar{\boldsymbol{\omega}}(t) + \boldsymbol{\tau}(t) + \boldsymbol{\tau}_D(t)\,, \end{aligned} \tag{2}$$

where $\boldsymbol{\omega}(t) = [p(t)\ q(t)\ r(t)]^\top$ is the angular rate of F_B with respect to the inertial frame F_E expressed in F_B, $J = diag(J_1,\ J_2,\ J_3)$ is the vehicle's inertia matrix (the body frame is aligned with the principal axes of inertia), J_m is the rotor inertia about the axis of rotation (assuming identical for all of them), $\bar{\boldsymbol{\omega}}(t) = [-q(t)\ p(t)\ 0]^\top$, $\omega_m(t) = \sum_{i=1}^n (-1)^i \Omega_i(t)$, $\Omega_i(t)$ is the i-th rotor angular rate about its axis of rotation, $\boldsymbol{\tau}(t)$ is the torque generated by the rotors, $\boldsymbol{\tau}_D(t)$ is the aerodynamic rotational drag torque.

It is assumed that all motors generate thrust in the positive z-direction in F_B frame ($\boldsymbol{e}_3^B$), and $f_T(t) = \sum_{i=1}^n f_i(t)$, where $f_i(t)$ is the thrust generated by the i-th rotor at time t.

B. Atmospheric Effects

The aerodynamic drag force is modeled in the body frame as $\boldsymbol{f}_D^B = [-c_{D_x} v_{a_x}^B |v_{a_x}^B| \quad - c_{D_y} v_{a_y}^B |v_{a_y}^B| \quad - c_{D_z} v_{a_z}^B |v_{a_z}^B|]^\top$, where the drag coefficients c_{D_i} are constant for each axis $i = x, y, z$, $\boldsymbol{v}_a^B(t) = \boldsymbol{v}^B(t) - \boldsymbol{w}^B(t)$ is the vehicle's relative to the air velocity expressed in the body frame, and $\boldsymbol{w}^B(t)$ is the wind inertial velocity expressed in the body frame.The drag force is translated to

the inertial frame as $\boldsymbol{f}_D = R_{B/E}\boldsymbol{f}_D^B$. The rotational drag torque is modeled in the body frame as $\boldsymbol{\tau}_D^B = [-c_{\tau_x}\omega_{a_x}^B|\omega_{a_x}^B| \quad -c_{\tau_y}\omega_{a_y}^B|\omega_{a_y}^B| \quad -c_{\tau_z}\omega_{a_z}^B|\omega_{a_z}^B|]^\top$, where coefficients c_{D_i} are constant for each axis $i = x, y, z$, and $\boldsymbol{\omega}_a^B(t) = \boldsymbol{\omega}(t) - \boldsymbol{\omega}_c^B(t)$ is the vehicle's relative to air angular rate expressed in the body frame, which includes the air mass circulation rate (or vorticity) $\boldsymbol{\omega}_c^B(t)$ expressed in the body frame. We refer interested reader to [16] for details. Since the drag coefficients and the wind components (velocities and accelerations) can be unknown, for the purposes of this paper we use the estimates $\hat{\boldsymbol{f}}_D(t)$, $\hat{\boldsymbol{\tau}}_D^B(t)$, $\hat{\boldsymbol{w}}^B(t)$, $\hat{\boldsymbol{\omega}}_c^B(t)$, $\hat{C}_D(t)$ and $\hat{C}_\tau(t)$ and their derivatives computed according to algorithms presented in [16].

IV. Trajectory Generation

In this section we present a trajectory generation algorithm that takes into account effects of atmospheric disturbances in the multi-copter's dynamics using the estimates $\hat{\boldsymbol{w}}(t)$, $\hat{\boldsymbol{\omega}}_c(t)$, $\hat{\boldsymbol{s}}_v(t) = \hat{\boldsymbol{f}}_D(t)/m$ and $\hat{\boldsymbol{s}}_\omega(t) = J^{-1}\hat{\boldsymbol{\tau}}_D^B(t)$ and their derivatives from the previous section. For this purpose, we consider the simplified equation of motion

$$\dot{\boldsymbol{v}}(t) = \bar{f}(t)R_{B/E}(t)\boldsymbol{e}_3^B + \boldsymbol{g} + \hat{\boldsymbol{s}}_v(t)\,, \tag{3}$$

where the rotation matrix $R_{B/E}(t)$ evolves according to equation

$$\dot{R}_{B/E}(t) = R_{B/E}(t)\boldsymbol{\omega}^\times(t)\,, \tag{4}$$

and the mass-normalized total thrust $\bar{f}(t) = \frac{f_T(t)}{m}$ and the angular rate $\boldsymbol{\omega}(t)$ are viewed as control inputs. The justification of this simplification is that the controller designed for the angular rate dynamics

$$\dot{\boldsymbol{\omega}}(t) = -J^{-1}\boldsymbol{\omega}(t) \times J\boldsymbol{\omega}(t) + J_m\omega_m(t)J^{-1}\bar{\boldsymbol{\omega}}(t) + J^{-1}\boldsymbol{\tau}(t) + \hat{\boldsymbol{s}}_\omega(t) \tag{5}$$

can provide fast and accurate tracking of the angular rate commands in the presence of rotational drag with or without wind.

Our objective is to generate a minimum time trajectory without side-slip through given way-points $P_i,\ i = 1, \ldots, N$, where P_1 can be the vehicle's current position. In particular, a straight segment $(P_{i-1},\ P_i)$ is generated when the way-points $P_{i-1},\ P_i,\ P_{i+1}$ are aligned for each $i =$

$2, \ldots, N$. Otherwise, two new points P_i^- and P_i^+ are placed equidistant from P_i respectively on the intervals $[P_{i-1}\ P_i]$ and $[P_i\ P_{i+1}]$ such that the triangle $(P_i^-,\ P_i,\ P_i^+)$ is obstacle free. Then a straight segment $(P_{i-1},\ P_i^-)$ and a curved segment $(P_i^-,\ P_i^+)$ are generated, and P_i is replaced with P_i^+ for the next step.

To generate a straight portion $(P_{i-1},\ P_i)$ we use the modified jerk minimization approach of [16], which takes into account the estimate $\hat{\boldsymbol{s}}_v(t)$ of the aerodynamic drag and its derivative. In this case, we need only a single axis motion primitive generated via an optimal control problem for the system

$$\dddot{s}(t) = u(t) \tag{6}$$

with performance index

$$J = \int_0^{t_{f,i}} u^2(\tau) d\tau \tag{7}$$

and a proper selection of initial $s_0,\ \dot{s}_0,\ \ddot{s}_0$ and final $s_{f,i},\ \dot{s}_{f,i},\ \ddot{s}_{f,i}$ conditions and corresponding transformation of dynamic constraints. The resulting closed form solution $s(t), 0 \leq t \leq t_{f,i}$ is a 5th order polynomial in time (we refer the interested reader to [10] for details).

The boundary conditions are set as follows. Let vector $\boldsymbol{p}_i$ denote the radius-vector of the way-point P_i. Then the direction of the trajectory $\boldsymbol{r}_i(t)$ between way-points P_{i-1} and P_i is given by a unit vector $\boldsymbol{h}_{i-1} = (\boldsymbol{p}_i - \boldsymbol{p}_{i-1})/d_i$, where $d_i = dist(P_{i-1}, P_i)$. Hence, $\boldsymbol{r}_i(t_{f,i-1}+\tau) = \boldsymbol{p}_{i-1} + s(\tau)\boldsymbol{h}_{i-1}$ for $0 \leq \tau \leq t_{f,i}$. It follows that the initial conditions can be selected as $s_0 = 0,\ \dot{s}_0 = v_{i-1},\ \ddot{s}_0 = a_{i-1}$, where $v_{i-1} = \|\dot{\boldsymbol{r}}_{i-1}(t_{f,i-1})\|$ is the speed and $a_{i-1} = \|\ddot{\boldsymbol{r}}_{i-1}(t_{f,i-1})\|$ is the acceleration, which are available from the previous portion for $i > 1$ or are the vehicle's speed and acceleration at current time t for $i = 1$. The final conditions are set to $s_{f,i} = d_i,\ \dot{s}_{f,i} = v_i$, and $\ddot{s}_{f,i} = 0$, where the final speed v_i is left free if the next leg of the trajectory is along the same line or computed from the perspective of a feasible turn, which will be detailed shortly.

To set the dynamic constraints we notice that equation (3) implies that the mass-normalized total thrust magnitude necessary to traverse the trajectory $\boldsymbol{r}_i(t)$ satisfies the constraint

$$\bar{f}(t) \ = \|\ddot{s}(t)\boldsymbol{h}_{i-1} - \boldsymbol{g} - \hat{\boldsymbol{s}}_v(t)\| \tag{8}$$

The orientation of the thrust vector is defined in this case only by the pitch angle, since the roll angle is zero for the this trajectory. Therefore, following the steps from [16] we obtain

$$|\omega_y^2(t)| \leq \frac{1}{\bar{f}(t)} \left\| \dddot{s}(t)\boldsymbol{h}_{i-1} - \dot{\hat{\boldsymbol{s}}}_v(t) \right\| . \tag{9}$$

Obviously, $\omega_x = 0$ and trajectory $\boldsymbol{r}_i(t)$ can be traversed using only two control inputs $\bar{f}$ and ω_y. ω_z will be defined later to make side-slip angle equal to zero.

Taking into account physical constraints on the total thrust

$$0 \leq f_{\min} \leq \bar{f}(t) \leq f_{\max} , \tag{10}$$

and limitations on the angular rates due to sensors

$$-\omega_{\max} \leq \boldsymbol{\omega}_a^B(t) \leq \omega_{\max} , \tag{11}$$

which directly takes into account the estimate of wind vorticity $\hat{\boldsymbol{\omega}}_c^B(t)$, we compute and time-to-go $t_{f,i}$ following the steps from [16].

It can be shown that the presented algorithm always finds a feasible trajectory, which is optimal in the sense of performance index (7) (aggressiveness of the traverse as explained in [10]) and suboptimal in the sense of time-to-go within a user defined margin (see details in [16]).

To generate curved portion $(P_i^-,\ P_i^+)$ of the trajectory, we first solve a minimum time bank angle command problem using second order simplified system with torque as an input

$$\begin{aligned} &\min\ t_f, \\ &\ddot{s}(t) = \tau(t) \\ &s(0) = 0,\ \dot{s}(0) = 0,\ s(t_f) = \phi_{\max},\ \dot{s}(t_f) = \omega_{\max}, \\ &|\tau| \leq \tau_{\max} , \end{aligned} \tag{12}$$

where $\phi_{\max}$, $\omega_{\max}$ and $\tau_{\max}$ are respectively maximum allowable bank angle, angular rate and torque values. This constraint optimization problem is solved using Pontryagin's maximum principle. The

resulting closed form solution is given by the equation

$$s(t) = \begin{cases} \pi_1(t)\,, & 0 < t \le t_1 \\ \pi_2(t)\,, & t_1 < t \le t_2 \\ \pi_3(t)\,, & t_2 < t \le t_3 \end{cases} \tag{13}$$

where time instances $t_1,\ t_2,\ t_3$ and polynomials $\pi_1(t),\ \pi_2(t),\ \pi_3(t)$ are defined as follows. If $\sqrt{\tau_{\max}\phi_{\max}} \le \omega_{\max}$, then

$$t_1 = \sqrt{\tfrac{\phi_{\max}}{\tau_{\max}}},\quad t_2 = t_1,\quad t_3 = 2t_1$$

$$\pi_1(t) = \tfrac{\tau_{\max}}{2}t^2 - \phi_{\max},\quad \pi_2(t) = 0,\quad \pi_3(t) = -\tfrac{\tau_{\max}}{2}t^2 + 2\tau_{\max}t_1 t - \phi_{\max} - \tau_{\max}t_1^2\,,$$

otherwise

$$t_1 = \tfrac{\omega_{\max}}{\tau_{\max}},\quad t_2 = t_1 + \tfrac{\phi_{\max}}{\omega_{\max}},\quad t_3 = t_1 + t_2$$

$$\pi_1(t) = \tfrac{\tau_{\max}}{2}t^2 - \phi_{\max},\quad \pi_2(t) = \omega_{\max}t - \phi_{\max} - \tfrac{\tau_{\max}}{2}t_1^2, \tag{14}$$

$$\pi_3(t) = -\tfrac{\tau_{\max}}{2}t^2 + (\omega_{\max} + \tau_{\max}t_1)t - \phi_{\max} - \tfrac{\tau_{\max}}{2}(t_1^2 + t_2^2)\,.$$

The minimum time is t_3. The minimum time bank angle command is given by $\phi_{com}(t) = \phi_{\max} + s(t)$.

Next we generate a normal acceleration command according to equation

$$a_n(t) = a_{\max}\sin(\phi_{com}(t))/\sin(\phi_{\max})\,, \tag{15}$$

which reaches from zero to the maximum allowable acceleration $a_{\max}$ in minimum time. The trajectory is generated according to equations

$$\ddot{\boldsymbol{r}}(t) = a_n \boldsymbol{n}_v(t)\,, \tag{16}$$

where the unit vector $\boldsymbol{n}_v(t)$ is normal to velocity in the plane of way-points $P_i^-,\ P_i,\ P_i^+$ at each time instance t. The initial position of this trajectory is way-point P_i^- and the initial velocity is $v_i\boldsymbol{h}_{i-1}$ from the previous step. Since the acceleration is always perpendicular to velocity, the magnitude of v_i is constant. To provide sufficient smoothness of the trajectory (continuous derivatives up to forth order), we impose a constraint for the trajectory to pass through way-point P_i^+ with a velocity $v_i\boldsymbol{h}_i$ and zero acceleration, where $\boldsymbol{h}_i$ is the unit vector in the direction of $[P_i,\ P_{i+1}]$. To this end we

compute v_i such that the condition $a_n^2(t_*)R(t_*) = v_i^2$ is satisfied for some t_* before the trajectory reaches the bisector of angle $< P_i^- P_i P_i^+$, where $R(t_*)$ is the distance of $\boldsymbol{r}(t_*)$ from the bisector. Then we set $a_n^2(t) = a_n^2(t_*)$ for $t > t_*$ until trajectory reaches the bisector, which implies that this portion of the trajectory is a circular arc of radius $R(t_*)$ (see 2 for illustration). The rest of trajectory is symmetrical about the bisector until it reaches way-point P_i^+.

We notice that computed value v_i guarantees feasibility of the curved portion of trajectory between way-points P_i^- and P_i^+ with a circular arc in the middle. However to guarantee that the previous straight portion between way-points P_{i-1} and P_i^- is feasible without velocity reversal, we set $v_i = \min\left(v_i, \sqrt{2a_{\max} d_{i-1} - v_{i-1}^2}\right)$, where v_{i-1} is its initial speed and $d_{i-1} = dist(P_{i-1}, P_i^-)$ is the length.

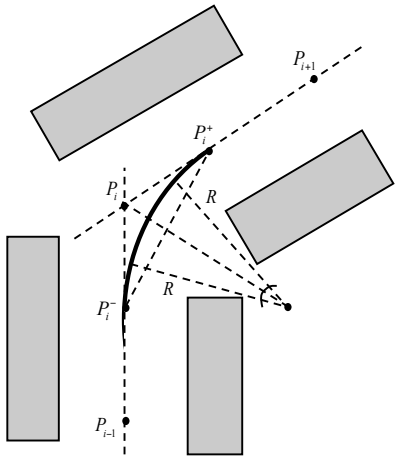

Fig. 2 Way points make a triangle.

The resulting trajectory is sufficiently smooth and has a non-zero traverse velocity. Therefore the approach from [9] can be applied to compute the orientation angles of the body frame such that the side-slip angle is zero. To this end, we first obtain the necessary mass-normalized thrust vector $\boldsymbol{f}_T(t) = \bar{f}(t) R_{B/E}(t) \boldsymbol{e}_3^B$ to move the vehicle's center of mass along the trajectory $\boldsymbol{r}(t) = [x(t)\ y(t)\ z(t)]^\top$ as

$$\boldsymbol{f}_T(t) = \ddot{\boldsymbol{r}}(t) - \boldsymbol{g} - \hat{\boldsymbol{s}}_v(t)\,, \tag{17}$$

which implies that

$$\|\boldsymbol{f}_T(t)\| = \|\ddot{\boldsymbol{r}}(t) - \boldsymbol{g} - \hat{\boldsymbol{s}}_v(t)\| = \bar{f}(t)\,, \tag{18}$$

which is non-zero unless the vehicle is not in flight. Therefore

$$\boldsymbol{e}_3^B(t) = \frac{\boldsymbol{f}_T(t)}{\bar{f}(t)}\,. \tag{19}$$

Next we compute a unit vector in the direction of velocity, which is always defined.

$$\boldsymbol{e}_{\boldsymbol{v}}(t) = \frac{\dot{\boldsymbol{r}}(t)}{\|\dot{\boldsymbol{r}}(t)\|}\,. \tag{20}$$

Let $\boldsymbol{e}_3^B(t)$ and $\boldsymbol{e}_{\boldsymbol{v}}(t)$ are not parallel. Then the body frame second direction can be computed as

$$\boldsymbol{e}_2^B(t) = \frac{\boldsymbol{e}_2^B(t) \times \boldsymbol{e}_{\boldsymbol{v}}(t)}{\|\boldsymbol{e}_2^B(t) \times \boldsymbol{e}_{\boldsymbol{v}}(t)\|}\,. \tag{21}$$

Therefore the vehicle's longitudinal axis direction at each time instance can be defined as

$$\boldsymbol{e}_1^B(t) = \boldsymbol{e}_2^B(t) \times \boldsymbol{e}_3^B(t)\,, \tag{22}$$

which completes the definition of rotation matrix from the vehicle's body to inertial frame, from which corresponding orientation angles are readily computed.

We notice that $\boldsymbol{e}_3^B(t)$ and $\boldsymbol{e}_{\boldsymbol{v}}(t)$ are parallel only when the vehicle flies vertically. In this case $\boldsymbol{e}_3^B(t) = [0\ 0\ 1]^\top$, and the other two axes are horizontal, that is $\phi(t) = 0$ and $\theta(t) = 0$ for all t. The yaw angle command $\psi(t)$ is computed as the minimum time solution of the optimization problem (12) with the final condition corresponding to the direction of next portion of the trajectory and constraints corresponding to yawing motion of the vehicle.

This completes the minimum time trajectory generation with zero side-slip angle, which comprises of position $\boldsymbol{r}_c(t)$, velocity $\boldsymbol{v}_c(t)$, acceleration $\boldsymbol{a}_c(t)$, and orientation angle $\phi_c(t),\ \theta_c(t),\ \psi_c(t)$ commands as sufficiently smooth time functions.

Remark IV.1 *We notice that the estimates of the drag force, drag torque, wind linear velocity, wind angular velocity and their derivatives are available at current time and at the vehicle's current position. Therefore, we use their extrapolation in the trajectory generation assuming constant derivatives along the trajectory. For example, we use* $\hat{\boldsymbol{w}}(\tau) = \hat{\boldsymbol{w}}(t) + (\tau - t)\dot{\hat{\boldsymbol{w}}}(t)$ *as the wind linear*

velocity estimate in forward time τ, where $\hat{\boldsymbol{w}}(t)$ and $\dot{\hat{\boldsymbol{w}}}(t)$ are the estimates at current time t. This approach is justified by the observations that the trajectory generation algorithm is very fast, and new estimated values can be used to regenerate the trajectory as soon as they became available in the next sensor sampling time step, which also includes the elapsed time for the estimation.

V. Required Power Computation

Let the remaining trajectory to be tracked by the UAV be $\boldsymbol{r}_c(\tau),\ \tau \geq t$, where t is the current time, and τ is the forward time along the desired trajectory. It follows from equation (17) that the thrust vector $\boldsymbol{T}_c(\tau)$ necessary to track the desired trajectory must at any forward time instance τ satisfy the equation

$$\boldsymbol{T}_c(\tau) = m\boldsymbol{a}_c(\tau) - m\boldsymbol{g} - \hat{\boldsymbol{s}}_v(\tau)\,, \tag{23}$$

where the estimate $\hat{\boldsymbol{s}}_v(\tau)$ is generated according to Remark IV.1 as $\hat{\boldsymbol{s}}_v(\tau) = \hat{\boldsymbol{s}}_v(t) + (\tau - t)\dot{\hat{\boldsymbol{s}}}_v(t)$, assuming the $\dot{\boldsymbol{s}}_v(t)$ is constant along the desired trajectory in forward time.

Given the required thrust $\boldsymbol{T}_c(\tau)$ and the free stream velocity $\boldsymbol{v}_a(\tau) = \boldsymbol{v}_c(\tau) - \hat{\boldsymbol{w}}(\tau)$, the power consumed by motors can be computed using the momentum theory [6]. From the conservation of momentum, the power P_j consumed by the jth motor to generate thrust T_j can be computed as

$$P_j = T_j[v_{a,j}\sin\alpha_j + v_{i,j}]\,, \tag{24}$$

where $v_{a,j}$ is the free stream airspeed for the jth rotor, α_j is the angle of attack, and $v_{i,j}$ is the induced velocity, which is perpendicular to the rotor's disc. We notice that the $v_{a,j}\sin\alpha_j$ is the component of the free stream velocity perpendicular to the rotor's disc and is identical for all rotors. That is, $v_a = \|\boldsymbol{v}_a\|$ is the vehicle's true airspeed and α is the vehicle's angle of attack defined to be positive for the forward motion. On the other hand, $v_{i,j}$ depends on the rotor's spin rate and can be computed from the conservation of momentum by solving the equation

$$v_{i,j} = \frac{T_j}{2\rho A\sqrt{(v_a\cos\alpha)^2 + (v_a\sin\alpha + v_{i,j})^2}}\,, \tag{25}$$

where A is the rotor's disc area.

We notice that from the given trajectory the individual motor thrust is not available, therefore we compute the approximate induced velocity $v_{i,j}$ using the average thrust $\frac{T_c(t)}{n}$ in (25), where n is

the number of rotors. Then the induced velocity is the same for all rotors and at each time instance is computed as the maximal real root of the quartic equation

$$v_i^4(\tau) + 2v_a(\tau)\sin\alpha(\tau)v_i^3(\tau) + v_a^2(\tau)v_i^2(\tau) = \frac{T_c^2(\tau)}{(2\rho An)^2}\,, \tag{26}$$

which has at least two real roots.

The last unknown term $\sin\alpha(\tau)$ in (26) can be computed from the projection of $\boldsymbol{v}_a(\tau)$ in the direction of thrust vector $\boldsymbol{T}_c(\tau)$. Expressing $\boldsymbol{v}_a(\tau)$ in the body frame as $\boldsymbol{v}_a^B(\tau) = R_{E/B}\boldsymbol{v}_a(\tau)$, we notice that the third component $v_{az}^B(\tau)$ of $\boldsymbol{v}_a^B(\tau)$ is aligned with the thrust direction. Therefore

$$v_{az}^B(\tau) = (\boldsymbol{e}_3^B)^\top(t)\boldsymbol{v}_a(\tau) = \frac{\boldsymbol{T}_c^\top(\tau)}{\|\boldsymbol{T}_c(\tau)\|}\boldsymbol{v}_a(\tau),$$

where we denote $T_c(\tau) = \|\boldsymbol{T}_c(\tau)\|$, and

$$\sin\alpha(\tau) = \frac{v_{az}^B(\tau)}{\|\boldsymbol{v}_a^B(\tau)\|} = \frac{v_{az}^B(\tau)}{v_a(\tau)} = \frac{\boldsymbol{T}_c^\top(\tau)\boldsymbol{v}_a(\tau)}{T_c(\tau)v_a(\tau)}\,. \tag{27}$$

The required total power at any time instance $\tau \geq t$ can be readily computed by adding up the equations (24)

$$P_c(\tau) = \sum_{j=1}^{n} P_j(\tau) = \sum_{j=1}^{n} T_j(\tau)\left[\frac{\boldsymbol{T}_c^\top(\tau)\boldsymbol{v}_a(\tau)}{T_c(\tau)} + v_i(\tau)\right] = \boldsymbol{T}_c^\top(\tau)\boldsymbol{v}_a(\tau) + v_i(\tau)T_c(\tau)\,. \tag{28}$$

Equation (28) is used to estimate the power at current time instance t, which is necessary to traverse the generated trajectory $\boldsymbol{r}_c(t+\tau), 0 \leq \tau \leq t_f$, where $t_f = \sum_{i=1}^{N} t_{f,i}$ is the time-to-go.

VI. Trajectory Tracking Control Design

The trajectory generation algorithm presented in the previous section does not depend on the vehicle's existing flight controller, therefore acceptable tracking performance may not be always achievable, especially in the presence of atmospheric disturbances. To this end, we design an adaptive outer-loop augmentation algorithm for performance improvement by taking into account the effects of disturbances and on-line estimates of vehicle's existing flight controller gains. The schematics of this augmentation is displayed in Fig. 3.

Let the desired trajectory be given as the altitude $z_c(t)$ and orientation angle $\phi_c(t),\ \theta_c(t),\ \psi_c(t)$ commands, which corresponds to estimated atmospheric disturbances. We assume that the existing flight controller is a Proportional-Integral-Derivative (PID) control in each channel.

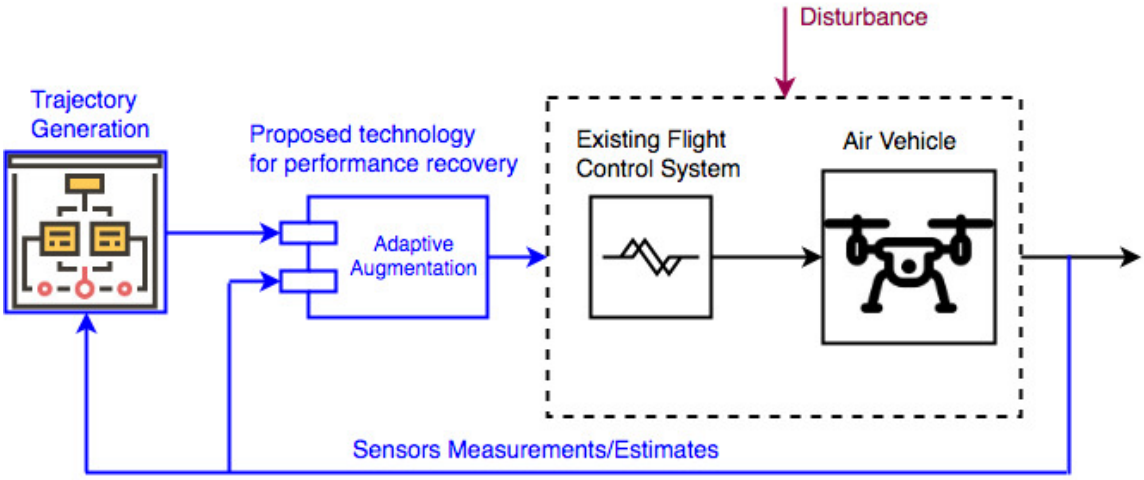

Fig. 3 Semantics of outer-loop augmentation.

First, we consider the altitude control problem, which is described by the equation

$$\ddot{z}(t) = \bar{f}(t)\cos\phi(t)\cos\theta(t) - g + \hat{s}_{vz}(t) \tag{29}$$

with control input $\bar{f}(t) = u_{z,bl}(t) + u_{z,a}(t)$, where

$$u_{z,bl}(t) = \left(k_p + \frac{k_i}{s} + k_d s\right)(z_{com}(t) - z(t))$$

is the baseline PID control with unknown gains $k_p,\ k_i,\ k_d > 0$ and $u_{z,a}(t)$ is the adaptive augmentation to be defined later. In the dynamics (29) we use estimates $\hat{s}_{vz}(t)$ instead of the actual drag force $s_{vz}(t)$ motivated by the convergence properties of this estimated given in [16] and avoiding double estimation in the adaptive control algorithm.

Introducing the tracking error $e_z(t) = z(t) - z_{com}(t)$ and its integral $e_{iz}(t)$, the dynamics (29) can be written as

$$\begin{aligned}
\dot{e}_{iz}(t) &= e_z(t) \\
\dot{e}_z(t) &= e_{vz}(t) \\
\dot{e}_{vz}(t) &= [k_p e_z(t) + k_i e_{iz}(t) + k_d e_{vz}(t) + u_a(t)]\cos\phi(t)\cos\theta(t) - g + \hat{s}_{vz}(t) - a_{z,com}(t)\,,
\end{aligned} \tag{30}$$

where $e_{vz}(t) = \dot{z}(t) - v_{z,com}(t)$ is the vertical velocity tracking error. Next, we introduce a prediction

model for the error dynamics (30) following the steps from [14] as

$$\begin{aligned}
\dot{\hat{e}}_{iz}(t) &= e_z(t) - \lambda \tilde{e}_{iz}(t) \\
\dot{\hat{e}}_{iz}(t) &= e_{vz}(t) - \lambda \tilde{e}_z(t) \\
\dot{\hat{e}}_{vz}(t) &= [\hat{k}_p(t) e_z(t) + \hat{k}_i(t) e_{iz}(t) + \hat{k}_d(t) e_{vz}(t) + u_a(t)] \cos\phi(t) \cos\theta(t) \\
&- g + \hat{s}_{vz}(t) - a_{z,com}(t) - \lambda \tilde{e}_{vz}(t)
\end{aligned} \tag{31}$$

where $\tilde{e}_{iz}(t) = e_{iz}(t) - \hat{e}_{iz}(t)$, $\tilde{e}_z(t) = e_z(t) - \hat{e}_z(t)$ and $\tilde{e}_{vz}(t) = e_{vz}(t) - \hat{e}_{vz}(t)$ are the prediction errors, $\lambda > 0$ is a design parameter, and variables with "hat" notations are the estimates of corresponding quantities without "hat", which are updated on-line according to adaptive laws

$$\begin{aligned}
\dot{\hat{k}}_p(t) &= \gamma \tilde{e}_{vz}(t) e_z(t) \cos\phi(t) \cos\theta(t) \\
\dot{\hat{k}}_i(t) &= \gamma \tilde{e}_{vz}(t) e_{iz}(t) \cos\phi(t) \cos\theta(t) \\
\dot{\hat{k}}_d(t) &= \gamma \tilde{e}_{vz}(t) e_{vz}(t) \cos\phi(t) \cos\theta(t)
\end{aligned} \tag{32}$$

with $\gamma > 0$ being the adaptive learning rate. The adaptive augmentation $u_{z,a}(t)$ is defined as

$$u_{z,a}(t) = -\hat{k}_p(t) e_z(t) - \hat{k}_i(t) e_{iz}(t) - \hat{k}_d(t) e_{vz}(t) + \frac{g - a_{z,com}(t) - \hat{s}_{vz}(t)}{\cos\phi(t) \cos\theta(t)}$$

which results in the following prediction error dynamics

$$\begin{aligned}
\dot{\tilde{e}}_{iz}(t) &= -\lambda \tilde{e}_{iz}(t) \\
\dot{\tilde{e}}_{iz}(t) &= -\lambda \tilde{e}_z(t) \\
\dot{\tilde{e}}_{vz}(t) &= [\tilde{k}_p(t) e_z(t) + \tilde{k}_i(t) e_{iz}(t) + \tilde{k}_d(t) e_{vz}(t)] \cos\phi(t) \cos\theta(t) - \lambda \tilde{e}_{vz}(t)
\end{aligned} \tag{33}$$

It can be easily shown that the error system (33) along with the adaptive laws (32) is stable with quantifiable error bounds, which can be derived following the steps from [16].

Next, we consider the orientation control problem for the system

$$\begin{aligned}
\dot{\boldsymbol{E}}(t) &= H(t) \boldsymbol{\omega}(t) \\
\dot{\boldsymbol{\omega}}(t) &= -J^{-1} \boldsymbol{\omega}(t) \times J \boldsymbol{\omega}(t) + J_m \omega_m(t) J^{-1} \bar{\boldsymbol{\omega}}(t) + J^{-1} \boldsymbol{\tau}(t) + \hat{\boldsymbol{s}}_\omega(t)\,,
\end{aligned} \tag{34}$$

where we denote

$$\boldsymbol{E} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \quad H = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}.$$

The objective is to design adaptive augmentation $\boldsymbol{\tau}_a(t)$ such that the Euler angle vector $\boldsymbol{E}(t)$ tracks the commend $\boldsymbol{E}_{com}(t) = [\phi_{com}(t) \;\; \theta_{com}(t) \;\; \psi_{com}(t)]^\top$, assuming that the existing baseline controller is PID

$$\boldsymbol{\tau}_{z,bl}(t) = \left(K_p + \frac{K_i}{s} + K_d s\right) \boldsymbol{e}_E(t)$$

with unknown diagonal gain matrices $K_p,\ K_i,\ K_d > 0$, where $\boldsymbol{e}_E(t) = \boldsymbol{E}(t) - \boldsymbol{E}_{com}(t)$ is the tracking error. Using time scale separation and dynamic inversion techniques, we first derive an expression for the desired angular rate

$$\boldsymbol{\omega}_{com}(t) = H^{-1}(t)[-c_{1E}\boldsymbol{e}_{iE}(t) - c_{2E}\boldsymbol{e}_E(t) + \dot{\boldsymbol{E}}_{com}(t)]\,, \tag{35}$$

where $\boldsymbol{e}_{iE}(t)$ is the integral of the tracking error, $c_{1E} > 0$ and $c_{2E} > 0$ are properly chosen control gains, and $H^{-1}(t)$ is the inverse of $H(t)$ given by

$$H^{-1}(t) = \begin{bmatrix} 1 & 0 & -\sin\theta(t) \\ 0 & \cos\phi(t) & \sin\phi(t)\cos\theta(t) \\ 0 & -\sin\phi(t) & \cos\phi(t)\cos\theta(t) \end{bmatrix}.$$

This results in an exponentially stable error system

$$\begin{aligned} \dot{\boldsymbol{e}}_{iE}(t) &= \boldsymbol{e}_E(t) \\ \dot{\boldsymbol{e}}_E(t) &= -c_{1E}\boldsymbol{e}_{iE}(t) - c_{2E}\boldsymbol{e}_E(t)\,. \end{aligned} \tag{36}$$

Introducing the angular rate tracking error $\boldsymbol{e}_\omega(t) = \boldsymbol{\omega}(t) - \boldsymbol{\omega}_{com}(t)$ and deriving the error equation

$$\begin{aligned} \dot{\boldsymbol{e}}_\omega(t) &= -J^{-1}\boldsymbol{\omega}(t) \times J\boldsymbol{\omega}(t) + J_m\omega_m(t)J^{-1}\bar{\boldsymbol{\omega}}(t) + \hat{\boldsymbol{s}}_\omega(t) - \boldsymbol{\omega}_{com}(t) \\ &+ J^{-1}\left(K_p\boldsymbol{e}_E(t) + K_i\boldsymbol{e}_{iE}(t) + K_d\dot{\boldsymbol{e}}_E(t) + \boldsymbol{\tau}_a(t)\right), \end{aligned} \tag{37}$$

we can estimate the unknown quantities from the following prediction model

$$\begin{aligned} \dot{\hat{\boldsymbol{e}}}_{\boldsymbol{\omega}}(t) &= -J^{-1}\boldsymbol{\omega}(t) \times J\boldsymbol{\omega}(t) + J_m\omega_m(t)J^{-1}\bar{\boldsymbol{\omega}}(t) + \hat{\boldsymbol{s}}_\omega(t) - \boldsymbol{\omega}_{com}(t) \\ &+ J^{-1}\left(\hat{K}_p(t)\boldsymbol{e}_E(t) + \hat{K}_i(t)\boldsymbol{e}_{iE}(t) + \hat{K}_d(t)\dot{\boldsymbol{e}}_E(t) + \boldsymbol{\tau}_a(t)\right) - \lambda_\omega\tilde{\boldsymbol{e}}_\omega(t)\,, \end{aligned} \tag{38}$$

where $\tilde{\boldsymbol{e}}_\omega(t) = \boldsymbol{e}_\omega(t) - \hat{\boldsymbol{e}}_\omega(t)$ is the prediction error and $\lambda_\omega > 0$ is a design parameter. It follows that the prediction error satisfies equation

$$\dot{\tilde{\boldsymbol{e}}}_{\boldsymbol{\omega}}(t) = J^{-1}\left(\tilde{K}_p(t)\boldsymbol{e}_E(t) + \tilde{K}_i(t)\boldsymbol{e}_{iE}(t) + \tilde{K}_d(t)\dot{\boldsymbol{e}}_E(t)\right) - \lambda_\omega \tilde{\boldsymbol{e}}_\omega(t)\,, \tag{39}$$

from which the following adaptive laws can be obtained using Lyapunov analysis

$$\begin{aligned} \dot{\hat{K}}_p(t) &= \gamma_\omega J^{-1}\tilde{\boldsymbol{e}}_\omega(t)\boldsymbol{e}_E^\top(t) \\ \dot{\hat{K}}_i(t) &= \gamma_\omega J^{-1}\tilde{\boldsymbol{e}}_\omega(t)\boldsymbol{e}_{iE}^\top(t) \\ \dot{\hat{K}}_d(t) &= \gamma_\omega J^{-1}\tilde{\boldsymbol{e}}_\omega(t)\dot{\boldsymbol{e}}_E^\top(t) \end{aligned} \tag{40}$$

where $\gamma_\omega > 0$ is the adaptive learning rate. The augmenting adaptive control is designed as

$$\begin{aligned} \boldsymbol{\tau}_a(t) &= \boldsymbol{\omega}(t) \times J\boldsymbol{\omega}(t) - J_{r3}\Omega(t)\bar{\boldsymbol{\omega}}(t)\hat{K}_p(t)\boldsymbol{e}_E(t) - \hat{K}_i(t)\boldsymbol{e}_{iE}(t) - \hat{K}_d(t)\dot{\boldsymbol{e}}_E(t) \\ &+ J[\boldsymbol{\omega}_{com}(t) - \hat{\boldsymbol{s}}_\omega(t) - c_\omega \boldsymbol{e}_\omega(t)]\,, \end{aligned} \tag{41}$$

where $c_\omega > 0$ is the control gain. This adaptive augmentation scheme guarantees convergence of $\boldsymbol{e}_\omega(t)$ to zero and user regulated bounds for the parameters estimation errors $\tilde{K}_p(t)$, $\tilde{K}_i(t)$ and $\tilde{K}_d(t)$ [14].

VII. Simulation Results

Using the dynamic model of DJI S1000 octocopter, we conducted MatLab simulations to demonstrate the performance of presented algorithms. A set of Way-points is generated by means of A^* path planning algorithm for a cityscape given by the 3D digital map, which takes into account obstacle information provided by on-board sensors. We generate a trajectory through way-points using the presented algorithm, assuming that the first way-point is the octocopter's initial condition. Then we re-plan the trajectory from vehicle's current position through remaining way-points every 5 sec, which corresponds to time interval required by the sensor information processing and way-points generation. Figure 4 displays the generated 3D trajectory with corresponding way-points, where red "star" markers indicate original and inserted way-points, which are traversed by the generated trajectory, and black markers indicates corner way-points, which are left out.

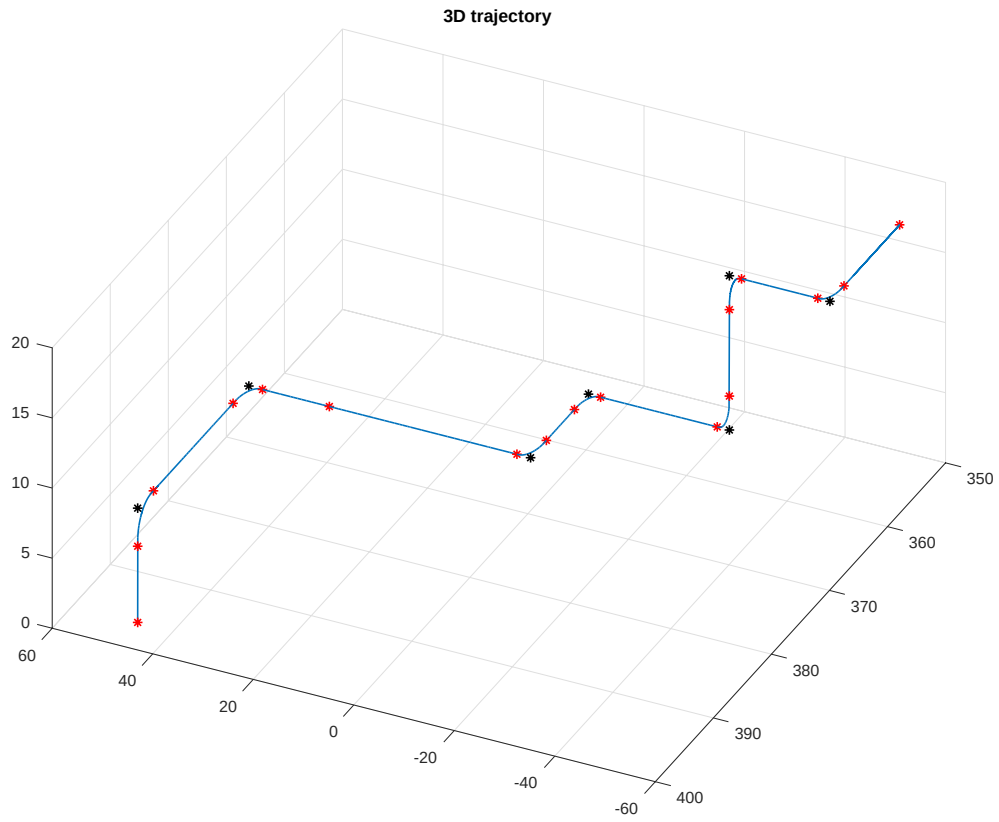

Fig. 4 Generated 3D trajectory and corresponding way-points.

For this simulation study, the wind field with variable linear and angular velocities in all directions is introduced such that the wind maximum velocity reaches $6m/s$, and the maximum vorticity reaches $1.7rad/sec$.

The wind estimates are computed from the on-line estimation of resulting linear and rotational drag force and torque according to algorithms presented in [16]. Figures 5 and 6 display the performance of adaptive estimation algorithms for linear and rotational drag respectively. A good convergence can be observed in all drag components.

Figures 7 and 8 display the estimated and actual wind linear and angular velocity components along the trajectory of flight. The observed spikes are numerical errors resulting from frequent zero-crossing of *sign* function, which is involved in computations of wind components via inversion of the corresponding drag component estimate. For example, $\hat{w}_z^B(t)$ is computed as $\hat{w}_z^B(t) = v_z^B(t) -$

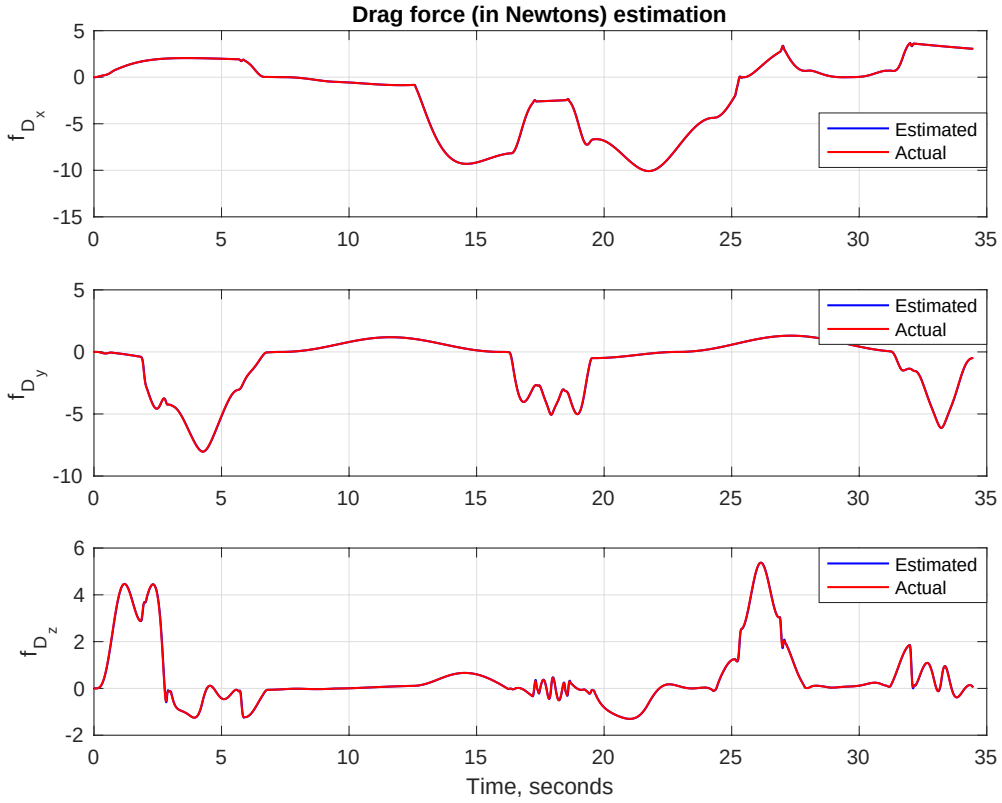

Fig. 5 Drag force estimation.

$\text{sign}(\hat{s}^B_{vz}(t))\sqrt{\frac{m}{c_{Dz}}|\hat{s}^B_{vz}(t)|}$, where $\hat{s}_{vz}(t)$ is directly estimated [16].

The estimated wind linear and angular velocities are used to generate (regenerate) minimum time trajectory and corresponding required power, and estimated drag force and torque are used in control augmentation algorithm to generate necessary total thrust and 3-axis torque, which consequently are translated into individual motor thrust trough the control allocation technique from [15].

Figures 9 and 10 display the tracking performance of the augmenting controller in positions and Euler angles commands. It can be observed that the close tracking is achieved despite severe wind condition, and that during the tracking of the generated trajectories the roll and pitch angles do not exceed angle limits of $45deg$ set in the trajectory generation algorithm.

Figures 11 and 12 display the total thrust and required power computed along the trajectory.

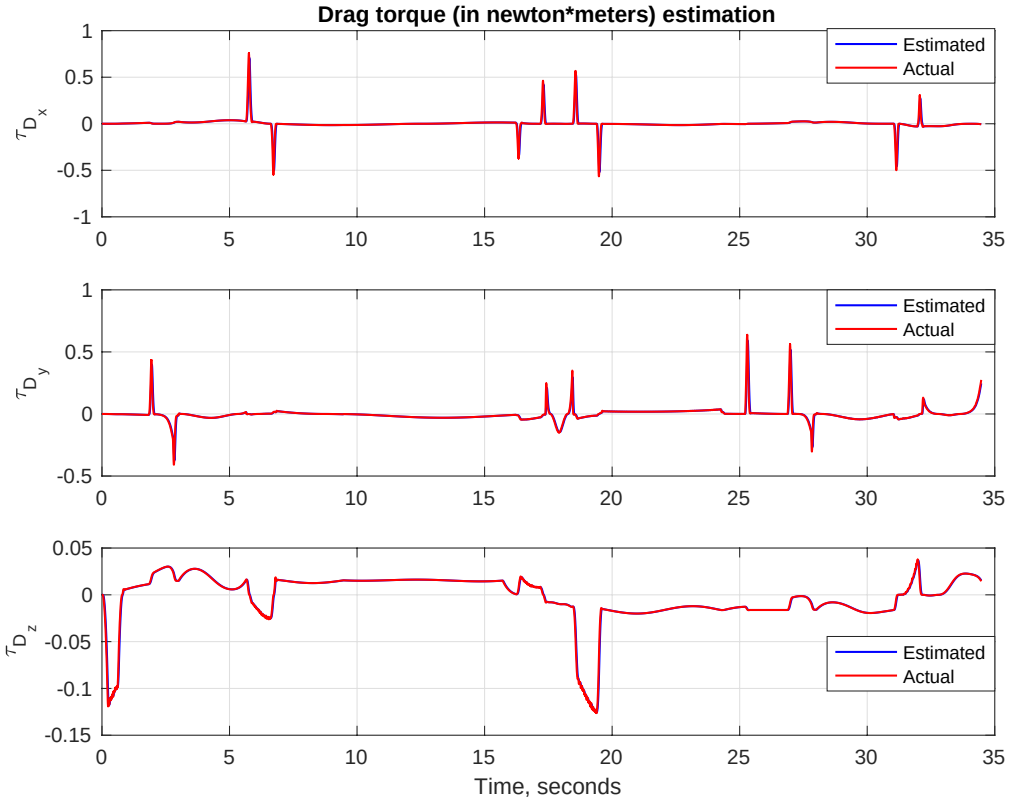

Fig. 6 Drag torque estimation.

VIII. Conclusion

We have presented a minimum time no side-slip trajectory generation algorithm for multi-rotor drones flying in urban wind field. This algorithm is based on closed form solutions and is computational very fast, which allows on-line planning and re-planning as new information about the wind filed or obstacles becomes available. The adaptive augmentation control algorithm is designed to estimate the existing flight control gains and provide acceptable tracking of the generated trajectories. The benefits of presented algorithms were demonstrated in simulations.

Acknowledgment

This work was supported by the NASA Ames UAS Traffic Management (UTM) Sub-project, under the NASA's Safe Autonomous Systems Operations (SASO) Project. The authors gratefully acknowledge all members of the SAFE50 team for engaging in many hours of discussions on various

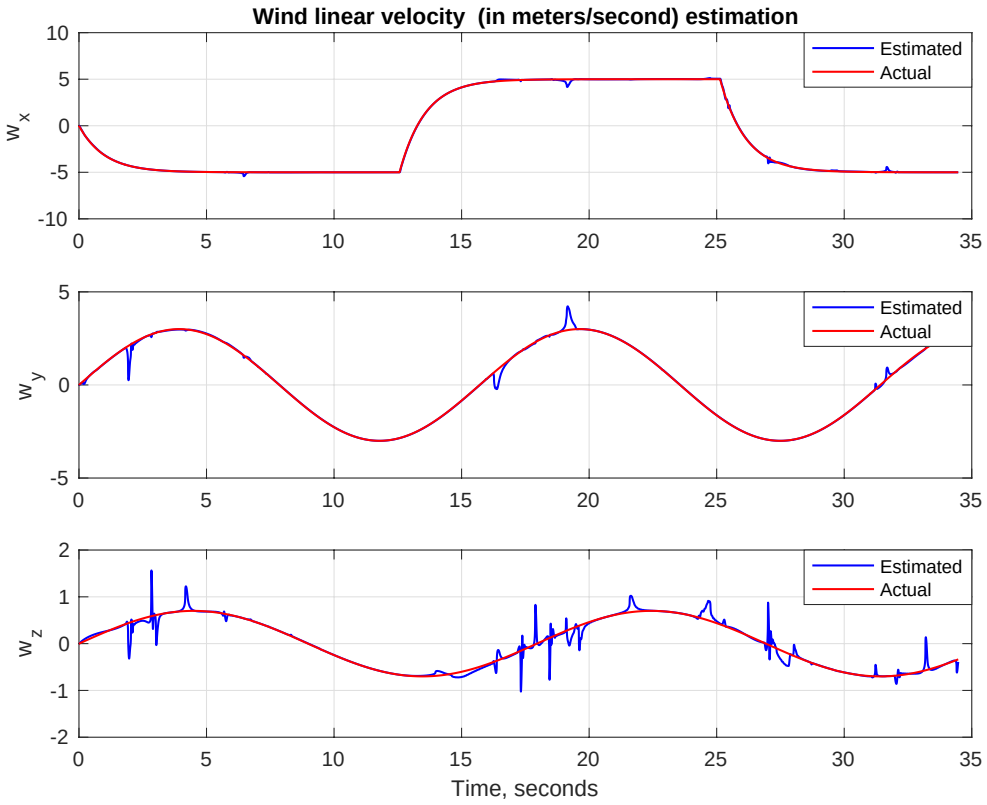

Fig. 7 Wind linear velocity estimation.

topics discussed in this paper.

References

[1] A. Ailon and I. Zohar. Control of Unmanned Aerial Vehicle With Restricted Input in the Presence of Additive Wind Perturbations. *In Proc. of the 11th International Conference on Control, Automation and Systems, Gyeonggido, Korea*, pages 873–878, 2011.

[2] A. Chakrabarty and J. Langelaan. UAV Flight Path Planning in Time Varying Complex Wind-fields. *In Proc. of the American Control Conference, Washington, DC*, pages 2568–2574, 2013.

[3] J.A. Guerrero, J.A. Escareno, and Y. Bestaoui. Quad-rotor MAV Trajectory Planning in Wind Fields. *In Proc. of the IEEE International Conference on Robotics and Automation, Karlsruhe, Germany*, pages 770–775, 2013.

[4] M. Q. Huynh, W. Zhao, and L. Xie. L1 Adaptive Control for Quadcopter: Design and Implementation. *In Proc. of the 13th International Conference on Control, Automation, Robotics and Vision, Marina Bay Sands, Singapore*, pages 1496–1501, 2014.

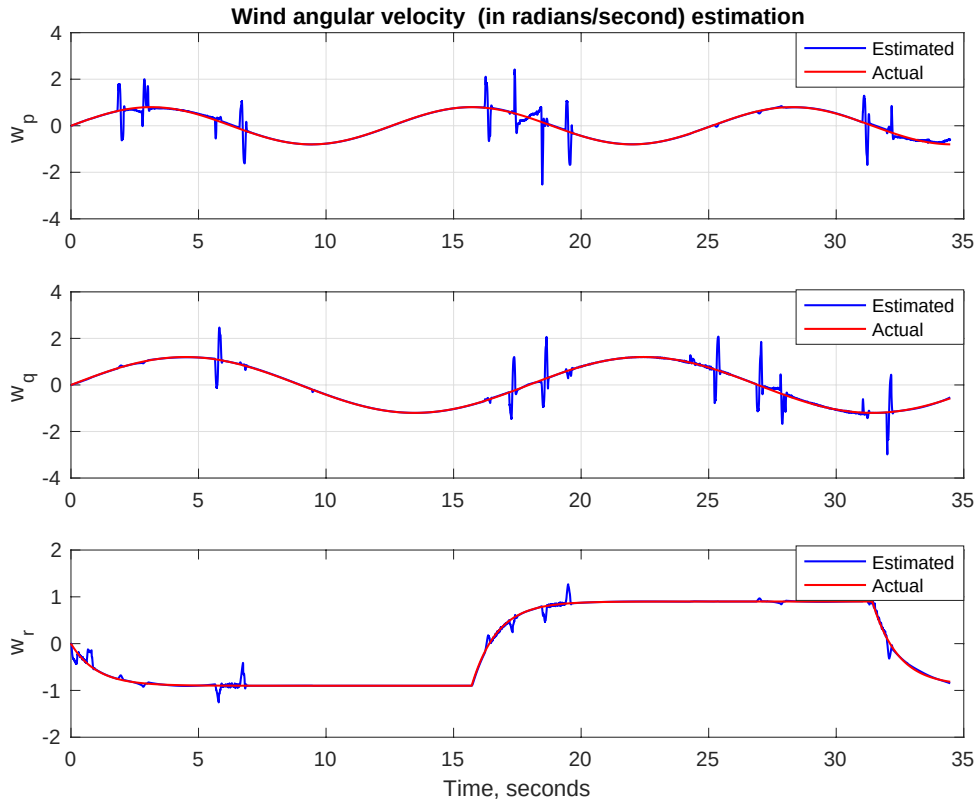

Fig. 8 Wind angular velocity estimation.

[5] T. Larrabee, H. Chao, M. Rhudy, Y. Gu, and M. R. Napolitano. Wind Field Estimation in UAV Formation Flight. *In Proc. of the American Control Conference, Portland, Oregon*, pages 5408–5413, 2014.

[6] G. J. Leishman. *Principles of Helicopter Aerodynamics.* Cambridge University Press, 2nd ed., Cambridge, 2006.

[7] C. Liu, O. McAreet, and W. Chen. Path Following for Small UAV s in the Presence of Wind Disturbance. *In Proc. of the UKACC International Conference on Control, Cardiff, UK*, pages 613–618, 2012.

[8] A. Martinez-Vasquez, A. Rodriguez-Mata, and I. Gonzalez-Hernandez. Linear Observer for Estimating Wind Gust in UAV's. *In Proc. of the 12th International Conference on Electrical Engineering, Computing Science and Automatic Control, Mexico City, Mexico*, pages 873–878, 2015.

[9] D. Mellinger and V. Kumar. Minimum Snap Trajectory Generation and Control for Quadrotors. *In Proc. of the IEEE International Conference on Robotics and Automation, Shanghai, China*, pages 2520–2525, 2011.

[10] M. W. Mueller, M. Hehn, and R. D'Andrea. A Computationally Efficient Motion Primitive for Quadro-

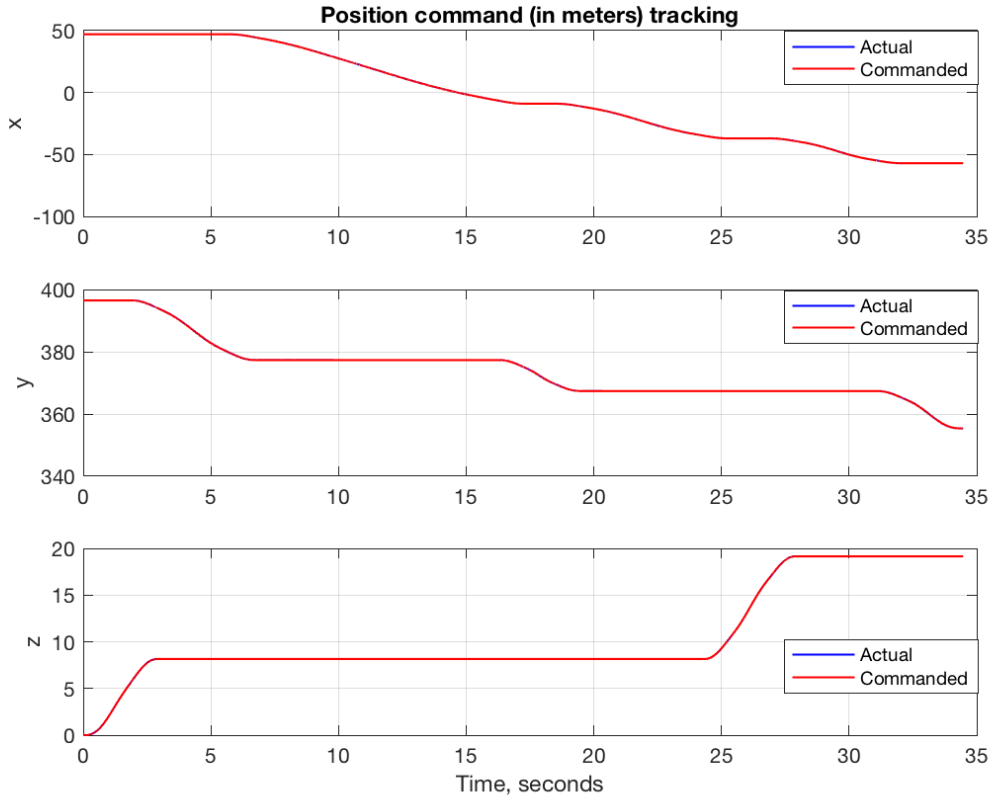

Fig. 9 Trajectory command and tracking performance.

copter Trajectory Generation. *IEEE Transactions on Robotics*, 31(6):820–825, December 2015.

[11] I. Sarras and H. Siguerdidjane. On the Guidance of a UAV Under Unknown Wind Disturbances. *In Proc. of the IEEE Multi-conference on Systems and Control, Antibes, France*, pages 820–825, 2014.

[12] S. Schopferer and T. Pfeifer. Performance-Aware Flight Path Planning for Unmanned Aircraft in Uniform Wind Fields. *In Proc. of the International Conference on Unmanned Aircraft Systems, Denver, Colorado*, pages 1138–1147, 2015.

[13] Q. Shen, D. Wang, S. Zhu, and E. K. Poh. Fault-Tolerant Attitude Tracking Control for a Quadrotor Aircraft. *In Proceedings of the 53rd IEEE Conference on Decision and Control, DOI: 10.1109/CDC.2014.7040349*, pages 6129 – 6134, 2014.

[14] V. Stepanyan and K. Krishnakumar. Adaptive Control with Reference Model Modification. *AIAA Journal of Guidance, Control, and Dynamics*, 35(4):1370–1374, 2012.

[15] V. Stepanyan, K. Krishnakumar, and A. Bencomo. Identification and reconfigurable Control of Impaired Multi-Rotor Drones. *In Proc. of the AIAA Guidance, Navigation, and Control Conference, San Diego, CA*, 2016.

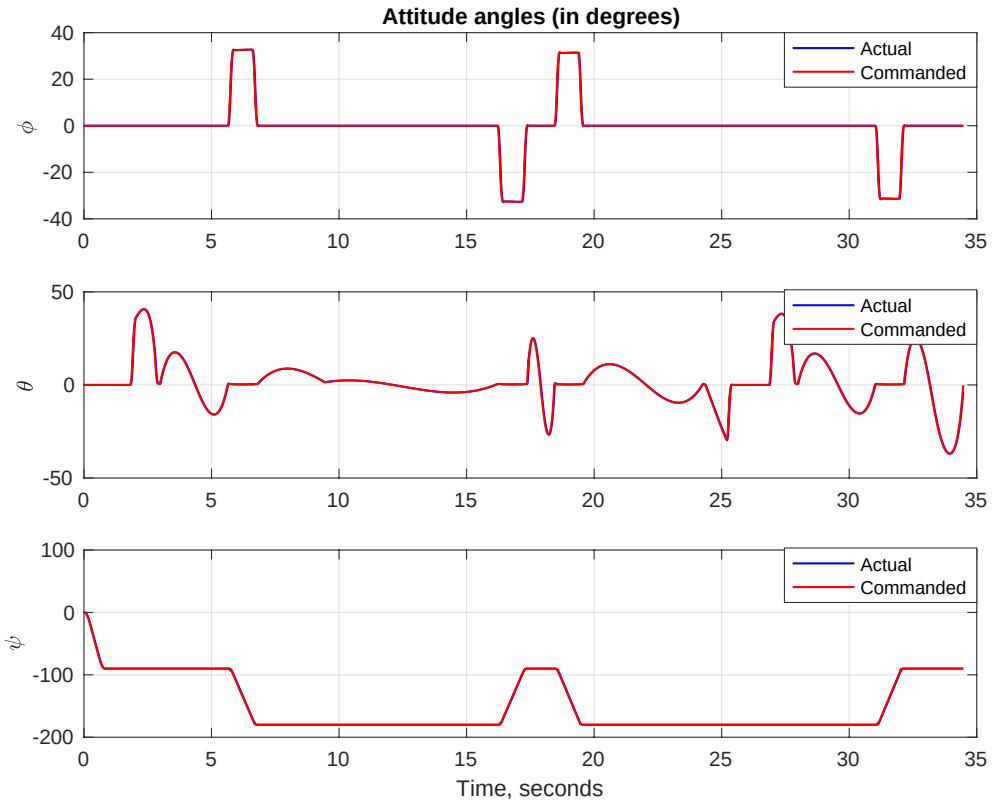

Fig. 10 The octocopter attitude angles in degrees.

[16] Vahram Stepanyan and Kalmanje Krishnakumar. Estimation, Navigation and Control of Multi-Rotor Drones in an Urban Wind Field. *In Proc. of the AIAA Information Systems-AIAA Infotech @ Aerospace, Grapevine, TX*, 2017.

[17] L. Techy and C. A. Woolsey. Minimum-Time Path Planning for Unmanned Aerial Vehicles in Steady Uniform Winds. *AIAA Journal of Guidance, Control, and Dynamics*, 32(6):1736–1746, 2009.

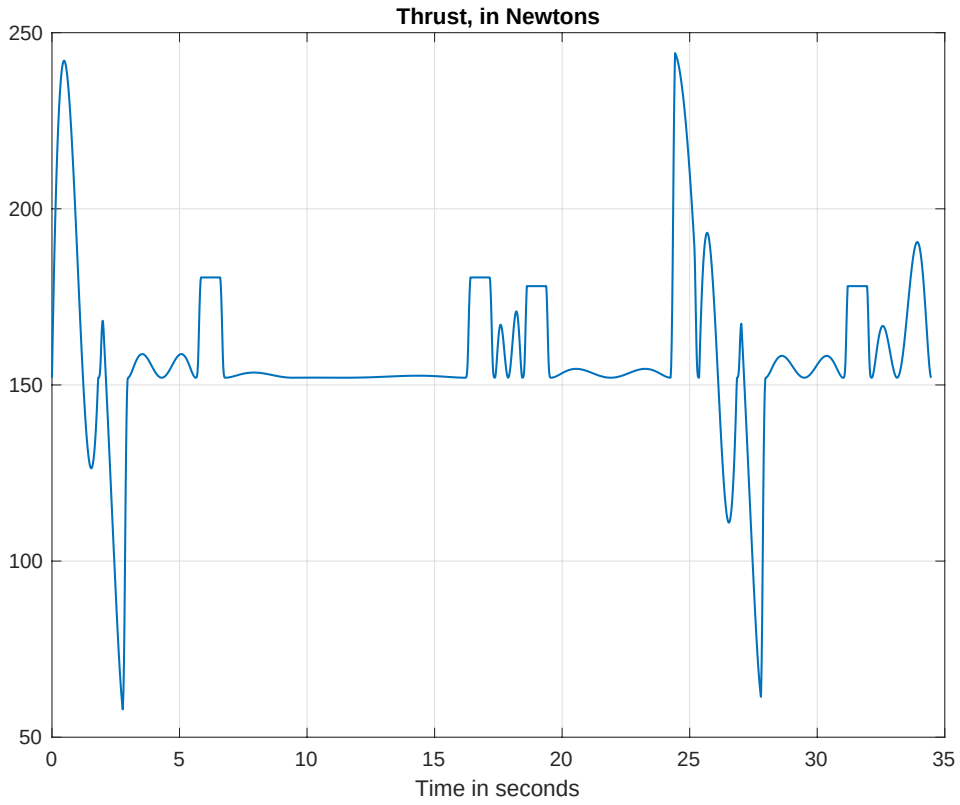

Fig. 11 Total thrust along the trajectory.

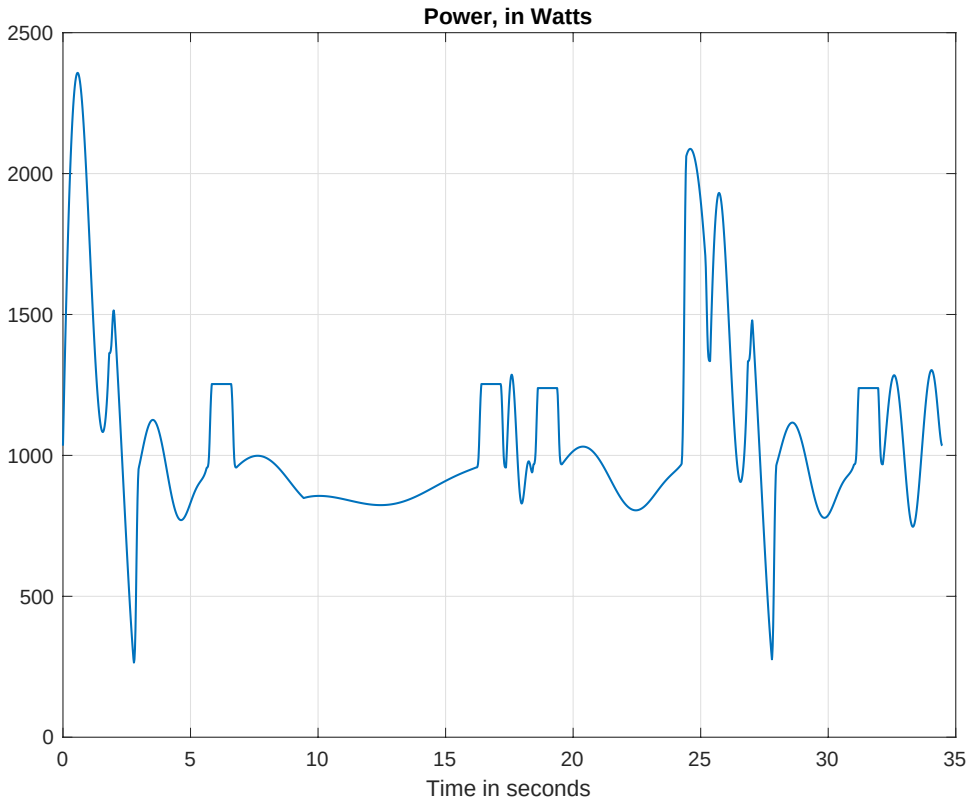

Fig. 12 Required power along the trajectory.