Model-free online motion adaptation for energy efficient flights of multicopters

Xiangyu Wu[1], Jun Zeng[2], Andrea Tagliabue[3], and Mark W. Mueller[1]

***Abstract*—Limited flight distance and time is a common problem for multicopters. We propose a method for finding the optimal speed and heading of multicopters while flying a given path to achieve the longest flight distance or time. Since flight speed and heading are often free variables in multicopter path planning, they can be changed without changing the mission. The proposed method is based on a novel multivariable extremum seeking controller with adaptive step size. It (a) does not require any power consumption model of the vehicle, (b) can be executed online, (c) is computationally efficient and runs on low-cost embedded computers in real-time, and (d) converges faster than the standard extremum seeking controller with constant step size. We prove the stability of this proposed extremum seeking controller, and conduct outdoor experiments to validate the effectiveness of this method with different initial conditions, with and without payload. This method could be especially useful for applications such as package delivery, where the weight, size and shape of the payload vary between deliveries and the power consumption of the vehicle is hard to model. Experiments show that compared to flying at the maximum speed with a bad heading angle, flying at the optimal range speed and heading reduces the energy consumed per distance by 24.9% without payload and 33.5% with a box payload. In addition, compared to hovering, flying at the optimal endurance speed and heading reduces the the power consumption by 7.0% without payload and 12.6% with a box payload.**

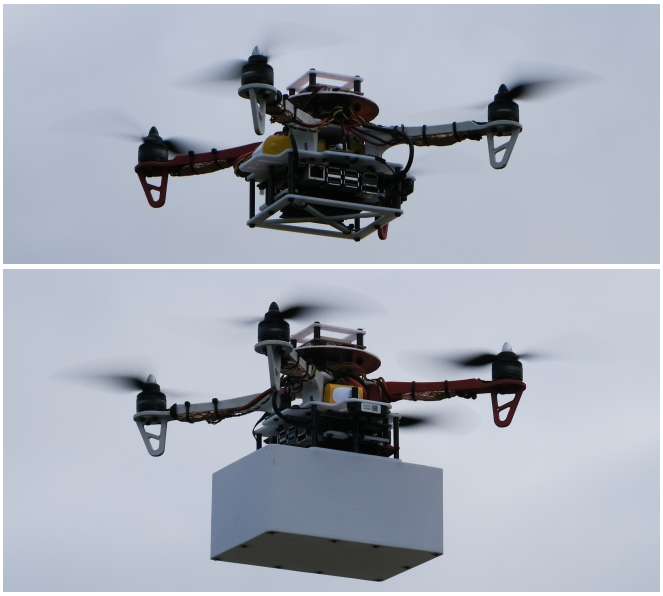


Figure 1. A quadcopter carrying payloads with differing mass and aerodynamics properties, as used in our experiments.

I. Introduction

Multicopters are used in a wide range of applications such as aerial photography [1], transportation [2], search and rescue [3], infrastructure inspection [4] and precision agriculture [5], thanks to their low cost, ease of control and high maneuverability. However, a primary limitation for current vehicles is their limited flight endurance and range [6].

One way to mitigate the limited flight range or endurance problem is through energy-efficient mechanical design. For example, in [7] a triangular quadcopter with one large rotor for lifting and three small rotors for control was proposed, which has the advantage of combining the energy efficiency of the large rotor and the fast control response of the small rotors. In [8], the authors designed a quadcopter with slightly-tilted motors which has a better control authority over the yaw. This results in a lower variance in motor forces for yaw control. Because a motor's power is a convex function of its thrust, this design helps to reduce the total power consumption of the motors. Hybrid quadcopters which are able to do both aerial and ground locomotion were introduced in [9] and [10]: when the vehicles operates in the ground locomotion mode on a flat ground, it only needs to overcome rolling resistance and uses much less power compared to flying. A hybrid power system for multicopters consisting of a lithium battery, a fuel cell and a hydrogen tank was introduced in [11], which enables longer flight time compared to traditional battery-only power systems. In [12], an in-flight battery switching system was proposed which enables a small quadcopter to dock an additional battery on a large quadcopter and increases its flight time.

Another category of methods focuses on developing algorithms to reduce the power consumption of existing multicopters. By planning energy efficient trajectories or by implementing energy aware control algorithms, these approaches do not require design changes on existing hardware and are thus economic to deploy. For example, in [13] the authors proposed a method for finding the minimum-energy trajectory between a predefined initial and final state of a quadcopter by solving an optimal control problem of the angular accelerations of the four propellers. This approach was extended in [14], where the fixed end-time trajectory optimization was extended to both free and fixed end-time solved with an indirect projected gradient algorithm to improve the numerical accuracy. Simulation results were shown to validate the effectiveness of

[1]Authors are with the High Performance Robotics Laboratory (HiPeR-Lab) at the Department of Mechanical Engineering, UC Berkeley. {wuxiangyu,mwm}@berkeley.edu

[2]Author is with the Hybrid Robotics Group at the Department of Mechanical Engineering, UC Berkeley. zengjunsjtu@berkeley.edu

[3]Author is with the Aerospace Controls Laboratory at the Department of Aeronautics and Astronautics, MIT. atagliab@mit.edu

the methods in both papers. In [15], the task of reaching a goal from a set of possible goals while using the least amount of energy was investigated. The energy-efficient path planning algorithm was based on model predictive control and disturbances from wind was considered. The authors tested their method in simulations and indoor experiments. In [16] and [17], the authors proposed energy-aware coverage path planning methods for photogrammetric sensing of large areas using multicopters. The methods find the optimal speed along the coverage path to minimize the energy usage during the mission. Outdoor experiments were conducted to validate their methods.

A necessary condition for model-based methods to perform well is accurate power consumption modeling. Power consumption models of multicopters can be derived by analyzing their electric and aerodynamic properties. For example, [18] [19] introduced power consumption models of the battery, electric speed controller and motor, and [20, chapter 5] introduced the aerodynamic power consumption of the propeller based on the momentum theory. Besides, some researchers proposed data-driven models by selecting variables that affect the power consumption (e.g. the vehicle's speed and acceleration, wind speed, and payload weight) as inputs and finding their relationship to power consumption through experimental data [16] [21].

However, there are often hard-to-model effects on the vehicle's power consumption such as changes in vehicle components' performance (e.g. batteries and motors) due to aging and temperature changes. In addition, the change in payload size, shape, or weight in applications such as package delivery and spraying (e.g. pesticide or fertilizer at farms) often requires re-identification of parameters in the power consumption model, which is time-consuming. These effects motivated us to propose, in our previous work [22], a model-free method for finding the flight speed which achieves the longest flight time or flight range given a predefined path. The method was based on extremum seeking control and does not require power consumption models of the multicopter.

Extremum seeking control is a model-free adaptive control technique for finding the local minimizer of a given, potentially time-varying, cost function by applying a persistently exciting periodic perturbation to a set of chosen inputs, and monitoring the corresponding output changes. A survey of the development of this control method can be found at [23]. It has applications in areas such as maximizing the energy generation of wind turbines [24] and photovoltaic power plants [25], and maximizing the pressure rise in axial-flow compressors [26]. Its applications in robotics can be found in a literature survey [27]. A common problem of extremum seeking controllers is slow convergence speed. In [28], which is our follow-up work of [22], we proposed a novel extremum seeking controller with adaptive step size which achieves faster convergence than the standard extremum seeking controller. In addition to flight speed, it could also find the optimal flight sideslip (heading) to achieve the longest flight range.

Nevertheless, in both [22] and [28], the experimental validation was conducted indoors where the setup was far from real-world applications: firstly, a motion capture system was used to provide state estimation of very high accuracy, which is not available outdoors. Secondly, because of the limited flight space, following the flight path (a circular path with a radius of about 2 meters) at a high speed required a significant centripetal force, which contributed a lot to the power consumption and is rare in real-world applications. Besides, no mathematical proof for stability was given for the extremum seeking controller with adaptive step size proposed in [28].

This work extends our previous related work [22] and [28] theoretically and experimentally. We demonstrate the ability of the online, adaptive, model-free method first proposed in our previous work of [28] in finding the speed and sideslip that achieve the longest flight range or endurance of multicopters in real-world environments. Not relying on any hardware changes and able to run in real-time on low-cost embedded computers, this method can be easily deployed on existing multicopters. In addition, based on a novel extremum seeking controller with adaptive step size, this method only requires three simple inputs: a user defined geometric path, electrical power measurements and speed measurements.

Compared with [22] and [28], the major extensions of this paper are:

(a) A stability proof of the proposed multivariable extremum seeking controller with adaptive step size.
(b) More practical outdoor experiments instead of previous work's indoor experiments with motion capture system for state estimation.
(c) Finding of the optimal endurance (time) flight speed and sideslip in addition to the optimal range (distance) flight speed and sideslip.

The remainder of this paper is structured as follows: Section II briefly describes the dynamics of multicopters and the effect of flight speed and sideslip on their power consumption. Section III states the energy optimal flight problem and derives the cost functions used in the optimal flight range and optimal flight endurance extremum seeking. Section IV gives a detailed explanation of the multivariable extremum seeking controller with adaptive step size and proves its stability. Section V shows the experimental results and Section VI summarizes this paper.

II. Dynamics and power consumption

In this section we define the reference frames and briefly introduce the quadcopter dynamics, and the effects of flight speed and sideslip on the power consumption of the vehicle. A power consumption model is given to help understanding some counter-intuitive effects observed in the experiments, e.g. the power consumption first decreases and then increases as the flight speed increases. This power consumption model is not used for the flight range and endurance optimization.

A. Quadcopter dynamics

As shown in Figure 2, two sets of reference frames are defined – an inertial frame I attached to the ground and a body-fixed frame B. The quadcopter is modeled as a rigid body with six degrees of freedom: three degrees of freedom

from the linear translation and three degrees of freedom from rotation. Each propeller j ($j = 1, ..., 4$), produces a thrust $\boldsymbol{f}_j$ and a torque $\boldsymbol{\tau}_j$, both expressed in frame B.

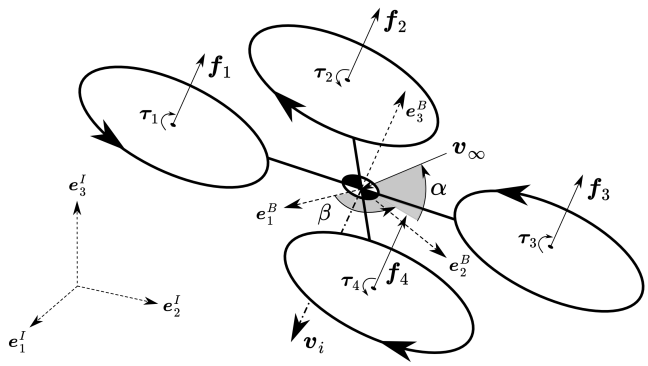


Figure 2. I represents the inertial frame and B represents the quadcopter's body frame. We additionally show the thrust and torque from the j-th propeller $\boldsymbol{f}_j$ and $\boldsymbol{\tau}_j$, the free-stream velocity $\boldsymbol{v}_\infty$, the induced velocity $\boldsymbol{v}_i$, angle of attack α, and the sideslip β, shown positive in the diagram.

The orientation of the vehicle with respect to the free-stream velocity $\boldsymbol{v}_\infty$ is described by the angle of attack α and the sideslip β, as shown in Figure 2. The angle of attack α is defined as the angle between $\boldsymbol{v}_\infty$ and the rotor plane; the sideslip is defined as the angle between $\boldsymbol{e}_1^B$ and the projection of $\boldsymbol{v}_\infty$ in the rotor plane.

With $\boldsymbol{x} \in \mathbb{R}^3$ and its derivatives denoting the translational position, velocity and acceleration of the vehicle, $\boldsymbol{g}$ denoting the acceleration due to gravity, all expressed in the inertial frame I, the translational dynamics of the vehicle are given by

$$m\ddot{\boldsymbol{x}} = m\boldsymbol{g} + \boldsymbol{R}\sum_j \boldsymbol{f}_j + \boldsymbol{f}_d, \tag{1}$$

where $\boldsymbol{f}_d$ is the drag force and $\boldsymbol{R}$ is the rotation matrix from the body frame B to the inertial frame I. The drag force $\boldsymbol{f}_d$ is a function of the rotation $\boldsymbol{R}$ and free-stream speed v_∞. In addition, with $\boldsymbol{\omega}$ and its derivative denoting the angular velocity and acceleration, expressed in the body-fixed frame B, the rotational dynamics of the vehicle is given by

$$\dot{\boldsymbol{R}} = \boldsymbol{R}S(\boldsymbol{\omega}), \tag{2}$$

$$\boldsymbol{I}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \boldsymbol{I}\boldsymbol{\omega} + \sum_j \boldsymbol{\tau}_j, \tag{3}$$

where $S(\boldsymbol{\omega})$ is the skew-symmetric matrix form of the vector cross product. Control of the quadcopter is achieved by controlling the thrust and torque generated by each rotor. A more detailed description of the quadcopter's dynamics and control can be found in e.g. [29].

B. Power consumption

When the flight speed increases, the power consumption of the vehicle usually first decreases and then increases. This effect can be explained by the momentum theory [20, chapter 2.14]. In forward flight, the propeller's power consumption p is computed as

$$p = \kappa\left(v_i + v_\infty \sin\alpha\right)\sum_j \|\boldsymbol{f}_j\|_2, \tag{4}$$

where $v_\infty = \|\boldsymbol{v}_\infty\|_2$ is the magnitude of the free-stream velocity and v_i represents the induced velocity applied by the propeller to the surrounding air. The scalar κ is an empirical correction factor. The induced velocity v_i is implicitly defined by

$$v_i = \frac{v_h^2}{\sqrt{(v_\infty\cos\alpha)^2 + (v_\infty\sin\alpha + v_i)^2}}, \quad v_h = \sqrt{\frac{mg}{8\rho\pi r^2}}, \tag{5}$$

where ρ is the density of the air and r is the radius of the propellers.

Compared to hovering, flying at a low speed can help increasing the free-stream speed v_∞ and decreasing the power consumption. When the speed is further increased, the drag force $\boldsymbol{f}_d$ becomes significant and large motor thrusts are needed for the flight, which cause power consumption to increase. This effect can be validated by solving the equations (4) and (5) numerically [20, chapter 2.14].

In addition, because the vehicle is usually not perfectly axisymmetric (especially when carrying payload), flying at the same speed but with different sideslip produces different drag force $\boldsymbol{f}_d$. This will cause the thrust forces $\boldsymbol{f}_j$ to be different, leading to different power consumption.

III. Problem Statement

In this work, we want to find the energy-efficient flight speed and sideslip angle to mitigate the common problem of the limited flight range and endurance of multicopters. We choose to optimize the flight speed and sideslip angle because they affect the vehicle's power consumption and are typically additional (redundant) degrees of freedom in multicopter's path planning, where the flight missions require the vehicle to track user-defined geometric paths.

When the goal is to achieve the longest flight endurance (time), we define the cost function as the instantaneous electric power p_e, while for the optimal range (distance) goal, the cost function is instantaneous electric power over speed p_e/v.

Consider a multicopter with energy $E \in [0, E_{\text{full}}]$ stored in the battery during the operation. Denote its electric power consumption to be p_e when it's flying with a constant speed $v := \|\boldsymbol{v}\|_2$ and sideslip angle β. If the goal is to reach the longest flight endurance, we define the total flight time t_{flight} as

$$t_{\text{flight}} := \int_{t_0}^{t_{\text{end}}} dt = \int_0^{E_{\text{full}}} \frac{1}{p_e} dE = \frac{1}{p_e} E_{\text{full}}, \tag{6}$$

where t_0 and t_{end} represent the initial and final time of the flight. Thus, maximizing the flight time corresponds to minimizing the instantaneous electrical power consumption p_e If the goal is to reach the longest flight range, we define the total flight distance d_{flight} as

$$d_{\text{flight}} := \int_{t_0}^{t_{\text{end}}} v\ dt = \int_0^{E_{\text{full}}} \frac{v}{p_e} dE = \frac{v}{p_e} E_{\text{full}}. \tag{7}$$

Thus, maximization of the flight range corresponds to minimization of p_e/v.

In order to minimize the flight range or endurance cost, a model-free method is preferable, which can handle hard-to-model effects (e.g. components aging and temperature change) and payload changes. This motivates us to use an extremum seeking controller (with adaptive step size) to find the optimal flight speed and sideslip angle. The required inputs to the extremum seeking controller are the instantaneous energy cost and a user defined geometric path. The outputs of it are the vehicle's reference speed and sideslip angle commands which are then used to convert the geometric path into a reference trajectory to be tracked by the low-level controllers. By adding a small, periodic perturbation to the reference speed and sideslip angle and monitoring the corresponding changes in the cost function, the extremum seeking controller is able to adapt its output to a small neighbourhood around the optimal flight speed and sideslip angle.

IV. Model-free speed and sideslip adaptation

In this section we introduce the extremum seeking controller with adaptive step size. It is able to achieve faster convergence than the standard extremum seeking controller with a fixed step size, by taking a smaller step size when the estimated gradient has a large magnitude or variance and vice versa.

A. Extremum seeking controller with adaptive step size

A block diagram of the proposed adaptive-step-size, multivariable, extremum seeking controller is shown in Figure 3. It is able to find the vehicle flight speed r_v^* and sideslip r_β^* which locally minimize the cost function $y = h(r_v, r_\beta)$ given that the cost function is locally convex, where r_v and r_β are the reference speed and sideslip. The derivation of the cost functions is in Section III.

1) Gradient estimation: The extremum seeking controller approximates the gradient of the cost function and integrates the negative of the estimated gradient to minimize the cost [30], under the assumption that the speed and sideslip setpoint $\hat{r}_v$ and $\hat{r}_\beta$ changes much slower than the perturbation and demodulation signals.

To approximate the gradient of the cost function, sinusoidal perturbations $\boldsymbol{p}(t) = [a_v \sin(\omega_v t), a_\beta \sin(\omega_\beta t)]^T$ are added to the speed setpoint $\hat{r}_v$ and sideslip setpoint $\hat{r}_\beta$. The effect of the perturbations on the cost function's value is then monitored. Firstly, the cost function's value is high pass filtered to remove the steady state components (η_v and η_β) and output the cost changes because of the perturbations ($y - \eta_v$ and $y - \eta_\beta$). These values are then multiplied with the demodulation signals $\boldsymbol{d}(t) = [\sin(\omega_v t), \sin(\omega_\beta t)]^T$. If the cost function's value change is in phase with the perturbations, which means that cost value increases as the inputs' values increase, the outputs ξ_v and ξ_β will be positive. If they are out of phase, the outputs will be negative. After this, ξ_v and ξ_β are low pass filtered to remove the high frequency noises in them, and their outputs q_v and q_β are approximations of the cost function's gradients. A more detailed and formal explanation of how the gradient is approximated is shown in the remainder of this subsection. For simplicity in notation we denote the setpoint $\hat{\boldsymbol{r}} = [\hat{r}_v, \hat{r}_\beta]^T$ and the setpoint with perturbation $\boldsymbol{r} = \hat{\boldsymbol{r}} + \boldsymbol{p}(t) = [r_v, r_\beta]^T$.

The first-order Taylor series expansion of the cost function at the setpoint $\hat{\boldsymbol{r}}$ is

$$y = h(\hat{\boldsymbol{r}} + \boldsymbol{p}(t)) = h(\hat{\boldsymbol{r}}) + \boldsymbol{p}(t)^T \frac{\partial h(\hat{\boldsymbol{r}})}{\partial \boldsymbol{r}} + O(\|\boldsymbol{a}\|^2), \quad (8)$$

where $\boldsymbol{a} = [a_v, a_\beta]^T$ and O means on the same order. The first order high pass filters suppress the low frequency signal $h(\hat{\boldsymbol{r}})$, and their outputs are given by

$$\begin{bmatrix} y - \eta_v \\ y - \eta_\beta \end{bmatrix} = \boldsymbol{p}(t)^T \frac{\partial h(\hat{\boldsymbol{r}})}{\partial \boldsymbol{r}} + O(\|\boldsymbol{a}\|^2), \quad (9)$$

where η_v and η_β are the low frequency components of the cost function removed by the high pass filters.

Multiplication with the demodulation signal $\boldsymbol{d}(t)$ results in

$$\boldsymbol{\xi} = \begin{bmatrix} \xi_v \\ \xi_\beta \end{bmatrix} = \boldsymbol{d}(t)\boldsymbol{p}(t)^T \frac{\partial h(\hat{\boldsymbol{r}})}{\partial \boldsymbol{r}} + O(\|\boldsymbol{a}\|^2), \quad (10)$$

which has a low frequency component (or average) given by

$$\left(\lim_{t\to\infty} \boldsymbol{d}(t)\boldsymbol{p}(t)^T dt\right) \frac{\partial h(\hat{\boldsymbol{r}})}{\partial \boldsymbol{r}}. \quad (11)$$

Thus, outputs of the first order low pass filters, q_v and q_β, are estimates of the cost function's gradient with respect to the speed and sideslip.

2) Step size adapter: The difference between the proposed extremum seeking controller and the standard multivariable extremum seeking controller [31] is the step size adapters, which are defined as follows:

$$\dot{m}_v = \gamma_v(-m_v + q_v^2), \quad \dot{m}_\beta = \gamma_\beta(-m_\beta + q_\beta^2), \quad (12)$$

$$g_v = \frac{q_v}{\sqrt{m_v + \epsilon}}, \quad g_\beta = \frac{q_\beta}{\sqrt{m_\beta + \epsilon}}, \quad (13)$$

where m_v, m_β are estimates of the second moments of the output of the low pass filters q_v and q_β, and ϵ is a small positive constant preventing dividing by zero. Equations in (12) are essentially first order low pass filters for q_v^2 and q_β^2, and γ_v and γ_β are their cut-off frequencies respectively. The idea comes from the adaptive moment estimation algorithm (Adam) [32], which is commonly used in the stochastic optimization of objective functions in machine learning, such as training neural networks.

The adapters take in the output of the low pass filters q_v and q_β, which are the gradient estimates, and outputs g_v and g_β. They are then passed to the integrators to perform gradient descent. The effective step size for gradient descent is $k_v g_v/q_v$ for the speed optimization and $k_\beta g_\beta/q_\beta$ for the sideslip optimization, and the step size adapters change them by setting g_v and g_β. The second moment of the initial outputs from the low pass filters are used to initialize m_v and m_β in (12).

In (13), by dividing q_v and q_β with the square root of their corresponding second moments, the outputs g_v and g_β of the adapters will be approximately bounded by ± 1, since $|\mathbb{E}[q_l]|/\sqrt{\mathbb{E}[q_l^2]} \leq 1$ ($\mathbb{E}$ denotes expected value, and q_l being either q_v or q_β). As a result, the descent rates for speed and slideslip are bounded by k_v and k_β. This can be understood

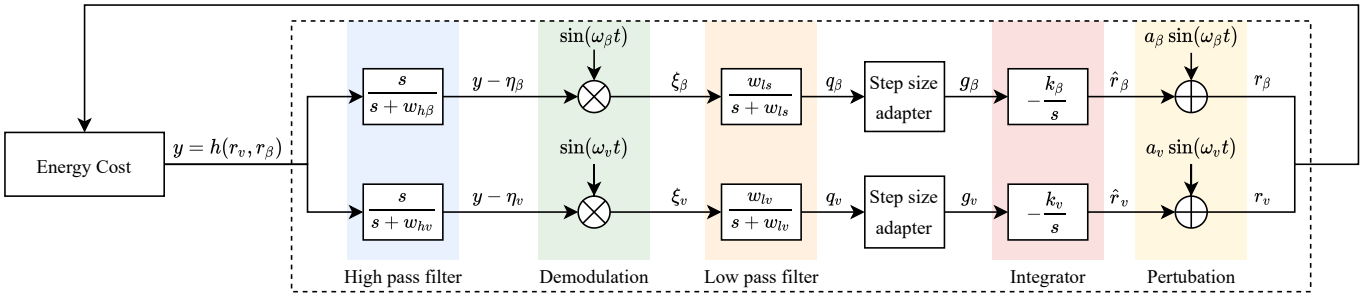


Figure 3. Block diagram of the adaptive step size multivariable extremum seeking controller (in the dashed rectangle). The goal of the controller is to find the optimal sideslip r_β and r_v to minimize the cost function $y = h(r_v, r_\beta)$. The frequencies of the high pass and low pass filters are set, respectively, to ω_{hv} and ω_{lv} for speed, and $\omega_{h\beta}$ and $\omega_{l\beta}$ for sideslip. The scalar k_v and k_β are related to the step size of the extremum seeking controller and both of them should be positive numbers to minimize the cost function. The standard extremum seeking controller with sinusoidal perturbations does not have the step size adapter and the outputs of the low pass filters directly go to the integrator, while the remaining structure of the algorithm is exactly the same. The step size adapter is detailed in Section IV-A2.

as establishing a trust region around the current parameter value, beyond which the current gradient estimation can be inaccurate. In addition, the adapters output small values when the gradient estimates have large uncertainty (m_v and m_β are large) and vice versa, which is good for the stability of the extremum seeking controller.

3) Parameter selection: The parameters for the extremum seeking controller are selected as

$$\begin{aligned}
&\omega_v = \omega\omega_v^{'} = O(\omega), \quad \omega_\beta = \omega\omega_\beta^{'} = O(\omega),\\
&\omega_{hv} = \omega\delta w_{hv}^{'} = O(\omega\delta), \quad \omega_{h\beta} = \omega\delta w_{h\beta}^{'} = O(\omega\delta),\\
&\omega_{lv} = \omega\delta w_{lv}^{'} = O(\omega\delta), \quad \omega_{l\beta} = \omega\delta w_{l\beta}^{'} = O(\omega\delta),\\
&k_v = \omega\delta k_v^{'} = O(\omega\delta), \quad k_\beta = \omega\delta k_\beta^{'} = O(\omega\delta),\\
&\gamma_v = \omega\delta\gamma_v^{'} = O(\omega\delta), \quad \gamma_\beta = \omega\delta\gamma_\beta^{'} = O(\omega\delta),
\end{aligned} \tag{14}$$

where δ and ω are small positive constants, and $\omega_v^{'}$, $\omega_\beta^{'}$, $\omega_{hv}^{'}$, $\omega_{h\beta}^{'}$, $\omega_{lv}^{'}$, $\omega_{l\beta}^{'}$, $k_v^{'}$, $k_\beta^{'}$, $\gamma_v^{'}$ and $\gamma_\beta^{'}$ are positive constants. ω needs to be small to make the perturbation slow compared to the dynamics of the multicopter, such that the dynamics of the multicopter does not interfere with the extremum seeking controller. The requirement for δ to be small is related to the stability of the controller. In addition, the magnitude of the perturbation signals, a_v and a_β are also required to be small. We will show the reasons for δ and perturbation magnitudes to be small in the stability analysis in Section IV-B. In addition, for this multivariable extremum seeking controller to work for both the speed and the sideslip simultaneously, their perturbation frequencies ω_v and ω_β should be different.

B. Stability Analysis

In this section we present the stability proof of the multivariable extremum seeking controller with adaptive step size using the averaging method. A similar method was used in [31] to prove the stability of a single variable standard extremum seeking controller and was used in [33] to prove the stability of a multivariable Newton-based extremum seeking controller.

1) System dynamics: We assume that the cost function in Section III has a local minimum at $\boldsymbol{r}^* = [r_v^*, r_\beta^*]^T$, such that

$$\nabla h(\boldsymbol{r}^*) = 0, \quad \nabla^2 h(\boldsymbol{r}^*) > 0. \tag{15}$$

The system dynamics in Figure 3 can be summarized as

$$\begin{aligned}
&\dot{\hat{r}}_v = -k_v\frac{q_v}{\sqrt{m_v+\epsilon}}, \quad \dot{\hat{r}}_\beta = -k_\beta\frac{q_\beta}{\sqrt{m_\beta+\epsilon}},\\
&\dot{q}_v = -\omega_{lv}q_v + \omega_{lv}(y-\eta_v)\sin w_v t,\\
&\dot{q}_\beta = -\omega_{l\beta}q_\beta + \omega_{l\beta}(y-\eta_\beta)\sin w_\beta t,\\
&\dot{\eta}_v = -\omega_{hv}\eta_v + \omega_{hv}y, \quad \dot{\eta}_\beta = -\omega_{h\beta}\eta_\beta + \omega_{h\beta}y,\\
&\dot{m}_v = \gamma_v(-m_v + q_v^2), \quad \dot{m}_\beta = \gamma_\beta(-m_\beta + q_\beta^2).
\end{aligned} \tag{16}$$

In order to analyze the stability at this local minimum, we denote $\tau = \omega t$ and

$$\begin{aligned}
\tilde{r}_v = \hat{r}_v - r_v^*, \quad \tilde{r}_\beta = \hat{r}_\beta - r_\beta^*,\\
\tilde{\eta}_v = \hat{\eta}_v - h(r_v^*, r_\beta^*), \quad \tilde{\eta}_\beta = \hat{\eta}_\beta - h(r_v^*, r_\beta^*).
\end{aligned} \tag{17}$$

Hence, the system dynamics with small perturbations can be written as

$$\frac{d}{d\tau}\begin{bmatrix}\tilde{r}_v\\ \tilde{r}_\beta\\ q_v\\ q_\beta\\ \tilde{\eta}_v\\ \tilde{\eta}_\beta\\ m_v\\ m_\beta\end{bmatrix} = \delta\begin{bmatrix}(-k_v^{'}q_v)/\sqrt{m_v+\epsilon}\\ (-k_\beta^{'}q_\beta)/\sqrt{m_\beta+\epsilon}\\ \omega_{lv}^{'}(v(\tilde{\boldsymbol{r}}+\bar{\boldsymbol{p}}(\tau)) - \tilde{\eta}_v)\sin w_v^{'}\tau - \omega_{lv}^{'}q_v\\ \omega_{l\beta}^{'}(v(\tilde{\boldsymbol{r}}+\bar{\boldsymbol{p}}(\tau)) - \tilde{\eta}_\beta)\sin w_\beta^{'}\tau - \omega_{l\beta}^{'}q_\beta\\ -\omega_{hv}^{'}\tilde{\eta}_v + \omega_{hv}^{'}v(\tilde{\boldsymbol{r}}+\bar{\boldsymbol{p}}(\tau))\\ -\omega_{h\beta}^{'}\tilde{\eta}_v + \omega_{h\beta}^{'}v(\tilde{\boldsymbol{r}}+\bar{\boldsymbol{p}}(\tau))\\ \gamma_v^{'}(-m_v+q_v^2)\\ \gamma_\beta^{'}(-m_\beta+q_\beta^2)\end{bmatrix}, \tag{18}$$

where $\tilde{\boldsymbol{r}} = [\tilde{r}_v, \tilde{r}_\beta]^T$, $\bar{\boldsymbol{p}}(\tau) = \boldsymbol{p}(t/\omega)$ and $v(\tilde{\boldsymbol{r}}+\bar{\boldsymbol{p}}(\tau)) = h(\boldsymbol{r}^* + \tilde{\boldsymbol{r}} + \bar{\boldsymbol{p}}(\tau)) - h(\boldsymbol{r}^*)$. As cost function $h(\cdot)$ holds local minimum at $\boldsymbol{r}^*$ shown in (15) the function $v(\cdot)$ holds local minimum at 0,

$$v(0) = 0, \ \nabla v(0) = 0, \ \nabla^2 v(0) > 0. \tag{19}$$

In order to provide compact notations, we denote $\nabla^2 v(0) = H$ for later discussions.

2) Averaging method: The system dynamics under small perturbations (18) is in the form where the averaging method is applicable [34, chapter 10.4] (δ is a small positive parameter).

Its corresponding averaged system dynamics can be described as follows,

$$\frac{d}{d\tau}\begin{bmatrix}\tilde{r}_v^a\\ \tilde{r}_\beta^a\\ q_v^a\\ q_\beta^a\\ \tilde{\eta}_v^a\\ \tilde{\eta}_\beta^a\\ m_v^a\\ m_\beta^a\end{bmatrix} = \delta\begin{bmatrix}(-k_v^{'}q_v^a)/\sqrt{m_v^a+\epsilon}\\ (-k_\beta^{'}q_\beta^a)/\sqrt{m_\beta^a+\epsilon}\\ \omega_{lv}^{'}\frac{1}{\Pi}\int_0^\Pi(v(\tilde{\boldsymbol{r}}^{\boldsymbol{a}}+\bar{\boldsymbol{p}}(\sigma))\sin\omega_v^{'}\sigma d\sigma-\omega_{lv}^{'}q_v^a\\ \omega_{l\beta}^{'}\frac{1}{\Pi}\int_0^\Pi(v(\tilde{\boldsymbol{r}}^{\boldsymbol{a}}+\bar{\boldsymbol{p}}(\sigma))\sin\omega_\beta^{'}\sigma d\sigma-\omega_{l\beta}^{'}q_\beta^a\\ -\omega_{hv}^{'}\tilde{\eta}_v^a+\omega_{hv}^{'}\frac{1}{\Pi}\int_0^\Pi v(\tilde{\boldsymbol{r}}^{\boldsymbol{a}}+\bar{\boldsymbol{p}}(\sigma))d\sigma\\ -\omega_{h\beta}^{'}\tilde{\eta}_\beta^a+\omega_{h\beta}^{'}\frac{1}{\Pi}\int_0^\Pi v(\tilde{\boldsymbol{r}}^{\boldsymbol{a}}+\bar{\boldsymbol{p}}(\sigma))d\sigma\\ \gamma_v^{'}(-m_v^a+q_v^{a2})\\ \gamma_\beta^{'}(-m_\beta^a+q_\beta^{a2})\end{bmatrix}, \tag{20}$$

where the superscript a denotes the variables of the averaged system, and Π is the least common period of sinusoidal functions with frequencies of $\omega_v^{'}$ and $\omega_\beta^{'}$.

The equilibrium point of the averaged system (20) is denoted as $[\tilde{r}_v^{a,e}, \tilde{r}_\beta^{a,e}, q_v^{a,e}, q_\beta^{a,e}, \tilde{\eta}_v^{a,e}, \tilde{\eta}_\beta^{a,e}, m_v^{a,e}, m_\beta^{a,e}]^T$ which satisfies:

$$q_v^{a,e} = q_\beta^{a,e} = 0, \tag{21}$$

$$m_v^{a,e} = m_\beta^{a,e} = 0, \tag{22}$$

$$\frac{1}{\Pi}\int_0^\Pi(v(\tilde{\boldsymbol{r}}^{\boldsymbol{a,e}}+\bar{\boldsymbol{p}}(\sigma))\sin\omega_v^{'}\sigma d\sigma = 0, \tag{23}$$

$$\frac{1}{\Pi}\int_0^\Pi(v(\tilde{\boldsymbol{r}}^{\boldsymbol{a,e}}+\bar{\boldsymbol{p}}(\sigma))\sin\omega_\beta^{'}\sigma d\sigma = 0, \tag{24}$$

$$\tilde{\eta}_v^{a,e} = \tilde{\eta}_\beta^{a,e} = \frac{1}{\Pi}\int_0^\Pi v(\tilde{\boldsymbol{r}}^{\boldsymbol{a,e}}+\bar{\boldsymbol{p}}(\sigma))d\sigma, \tag{25}$$

where the superscript e denotes the variables for the equilibrium point. We consider $\tilde{r}_v^{a,e}$ and $\tilde{r}_\beta^{a,e}$ as perturbations with second-order Taylor series expansion over a_v and a_β,

$$\begin{aligned}\tilde{r}_v^{a,e} &= b_{1,v}a_v + b_{2,v}a_\beta\\ &\quad + b_{3,v}a_v^2 + b_{4,v}a_va_\beta + b_{5,v}a_\beta^2 + O(\|\boldsymbol{a}\|^3),\end{aligned} \tag{26}$$

$$\begin{aligned}\tilde{r}_\beta^{a,e} &= b_{1,\beta}a_v + b_{2,\beta}a_\beta\\ &\quad + b_{3,\beta}a_v^2 + b_{4,\beta}a_va_\beta + b_{5,\beta}a_\beta^2 + O(\|\boldsymbol{a}\|^3),\end{aligned} \tag{27}$$

where $b_{i,v}$ and $b_{i,\beta}$ $(i = 1,..,5)$ are constant numbers. By substituting (26), (27) into (23), (24), integrating and equating the like powers of a_v and a_β, we can find that the first-order coefficients and second-order coefficients for the mixing terms are zero, and $\tilde{r}_v^{a,e}$ and $\tilde{r}_\beta^{a,e}$ can be written as:

$$\tilde{r}_v^{a,e} = b_{3,v}a_v^2 + b_{5,v}a_\beta^2 + O(\|\boldsymbol{a}\|^3), \tag{28}$$

$$\tilde{r}_\beta^{a,e} = b_{3,\beta}a_v^2 + b_{5,\beta}a_\beta^2 + O(\|\boldsymbol{a}\|^3). \tag{29}$$

In addition, by substituting (28), (29) into (25) and integrating, we can get

$$\tilde{\eta}_v^{a,e} = \tilde{\eta}_\beta^{a,e} = \frac{1}{4}(H_{11}a_v^2 + H_{22}a_\beta^2) + O(\|\boldsymbol{a}\|^3). \tag{30}$$

3) Stability of the averaged and original systems: At the equilibrium point of the averaged system in (20), the Hessian $J_r^{a,e}$ is a block-diagonal matrix as follows,

$$J_r^{a,e} = \delta\begin{bmatrix}A & 0_{4\times4}\\ B & -\text{diag}(\omega_{hv}^{'},\omega_{h\beta}^{'},\gamma_v^{'},\gamma_\beta^{'})\end{bmatrix}, \tag{31}$$

where $A, B \in \mathbb{R}^{4\times4}$,

$$A = \begin{bmatrix}0 & 0 & -k_v^{'}/\sqrt{\epsilon} & 0\\ 0 & 0 & 0 & -k_\beta^{'}/\sqrt{\epsilon}\\ A_{31} & A_{32} & -\omega_{lv}^{'} & 0\\ A_{41} & A_{42} & 0 & -\omega_{l\beta}^{'}\end{bmatrix}, \tag{32}$$

$$B = \begin{bmatrix}B_{11} & B_{12} & 0 & 0\\ B_{21} & B_{22} & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}, \tag{33}$$

with expressions of two matrices,

$$\begin{bmatrix}A_{31} & A_{32}\end{bmatrix}^T = \frac{\omega_{lv}^{'}}{\Pi}\int_0^\Pi\frac{\partial v(\tilde{\boldsymbol{r}}^{a,e}+\bar{\boldsymbol{p}}(\sigma))}{\partial\tilde{\boldsymbol{r}}^{a,e}}\sin\omega_v^{'}\sigma d\sigma, \tag{34}$$

$$\begin{bmatrix}A_{41} & A_{42}\end{bmatrix}^T = \frac{\omega_{l\beta}^{'}}{\Pi}\int_0^\Pi\frac{\partial v(\tilde{\boldsymbol{r}}^{a,e}+\bar{\boldsymbol{p}}(\sigma))}{\partial\tilde{\boldsymbol{r}}^{a,e}}\sin\omega_\beta^{'}\sigma d\sigma, \tag{35}$$

$$\begin{bmatrix}B_{11} & B_{12}\end{bmatrix}^T = \frac{\omega_{hv}^{'}}{\Pi}\int_0^\Pi\frac{\partial v(\tilde{\boldsymbol{r}}^{a,e}+\bar{\boldsymbol{p}}(\sigma))}{\partial\tilde{\boldsymbol{r}}^{a,e}}d\sigma, \tag{36}$$

$$\begin{bmatrix}B_{21} & B_{22}\end{bmatrix}^T = \frac{\omega_{h\beta}^{'}}{\Pi}\int_0^\Pi\frac{\partial v(\tilde{\boldsymbol{r}}^{a,e}+\bar{\boldsymbol{p}}(\sigma))}{\partial\tilde{\boldsymbol{r}}^{a,e}}d\sigma. \tag{37}$$

Hence, the block-lower-triangular matrix $J_r^{a,e}$ in (31) is Hurwitz if and only if that all diagonal submatrices are Hurwitz. Since $\delta, \gamma_v^{'}, \gamma_\beta^{'}, \omega_{hv}^{'}$ and $\omega_{h\beta}^{'}$ are positive constants, it remains to prove A as Hurwitz for stability.

With a first-order Taylor expansion we can get that

$$\begin{bmatrix}A_{31} & A_{32}\\ A_{41} & A_{42}\end{bmatrix} = \frac{1}{2}\begin{bmatrix}\omega_{lv}^{'}a_v & 0\\ 0 & \omega_{l\beta}^{'}a_\beta\end{bmatrix}H + O(\|\boldsymbol{a}\|). \tag{38}$$

The characteristic polynomial of A with roots λ can be written by computing the determinant of $\lambda I - A$,

$$\begin{aligned}&\det(\lambda I - A)\\ &= \det\left(\lambda I\left(\lambda I + \delta\begin{bmatrix}\omega_{lv}^{'} & 0\\ 0 & \omega_{l\beta}^{'}\end{bmatrix}\right)\right.\\ &\quad\left.+\frac{\delta^2}{\sqrt{\epsilon}}\begin{bmatrix}A_{31} & A_{32}\\ A_{41} & A_{42}\end{bmatrix}\begin{bmatrix}k_v^{'} & 0\\ 0 & k_\beta^{'}\end{bmatrix}\right)\\ &= \det\left(\lambda^2 I + \lambda\delta\begin{bmatrix}\omega_{lv}^{'} & 0\\ 0 & \omega_{l\beta}^{'}\end{bmatrix}\right.\\ &\quad\left.+\frac{\delta^2}{2\sqrt{\epsilon}}\begin{bmatrix}\omega_{lv}^{'}a_v & 0\\ 0 & \omega_{l\beta}^{'}a_\beta\end{bmatrix}H\begin{bmatrix}k_v^{'} & 0\\ 0 & k_\beta^{'}\end{bmatrix} + O(\delta^2\|\boldsymbol{a}\|)\right),\end{aligned} \tag{39}$$

which can be expanded to a 4th order polynomial of λ. Under the assumptions that $\|\boldsymbol{a}\|$ is small and that the Hessian H in (19) is positive, the roots of this 4th order polynomial

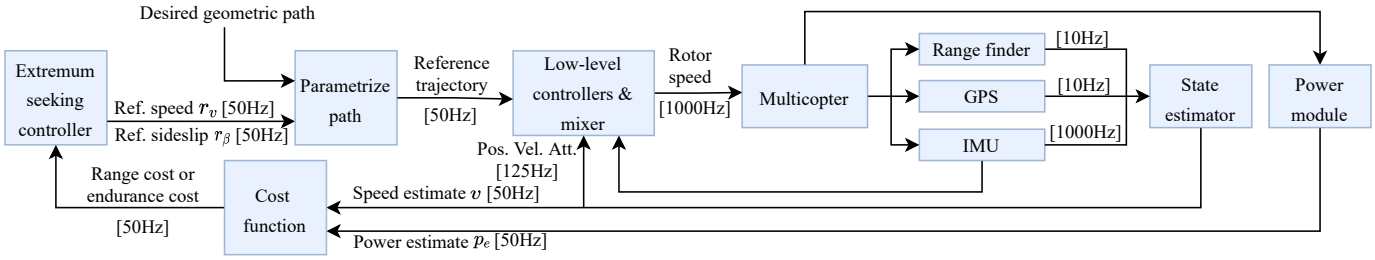


Figure 4. Control architecture for the model-free adaptive flight range or endurance optimization of a multicopter. The details of the extremum seeking controller block is shown in the dashed rectangle in Figure 3. The extremum seeking controller runs on the onboard computer (Jetson Nano) while the low-level controllers and the state estimator run on the Pixracer flight controller.

can be shown have negative real parts using the Routh-Hurwitz criterion [35, Chap. 6.2], implying that A is Hurwitz. Therefore, $J_r^{a,e}$ is proven as Hurwitz.

The Hurwitz Jacobian $J_r^{a,e}$ indicates that the equilibrium point of the averaged system (20) is locally exponentially stable if a_v and a_β are sufficiently small. According to the averaging theorem [34, Thm. 10.4], there exists $\bar{a}$ and $\bar{\delta}$ such that for all $\|\boldsymbol{a}\| \in (0, \bar{a})$, $\delta \in (0, \bar{\delta})$, the original system dynamics (18) have a unique exponentially stable periodic solution of period Π, which for all $\tau > 0$

$$\tilde{r}_v^{\Pi}(\tau) - b_{3,v}a_v^2 - b_{5,v}a_\beta^2 \leq O(\delta + \|\boldsymbol{a}\|^3), \tag{40}$$

$$\tilde{r}_\beta^{\Pi}(\tau) - b_{3,\beta}a_v^2 - b_{5,\beta}a_\beta^2 \leq O(\delta + \|\boldsymbol{a}\|^3), \tag{41}$$

which implies that the error terms $\tilde{r}_v^{\Pi}(\tau)$ and $\tilde{r}_\beta^{\Pi}(\tau)$ converge to an $O(\delta + \|\boldsymbol{a}\|^2)$ neighbourhood of zero. This means that the flight speed and sideslip found by the extremum seeking controller are periodic, and converge to an $O(\delta + \|\boldsymbol{a}\|^2)$ neighbourhood of their optimal values r_v^* and r_β^* (i.e. values that minimize the cost functions defined in Section III).

In summary, the proposed extremum seeking controller is locally stable – starting from an initial condition near the cost function's local minimum, it will converge to a neighbourhood around that local minimum if the perturbation is sufficiently small and the cost function is locally convex at the local minimum.

V. Experimental results

Outdoor experiments were conducted to validate the effectiveness of the extremum seeking controller with adaptive step size to find the optimal flight speed and sideslip. The proposed method was shown to have better convergence speed than standard extremum seeking control.

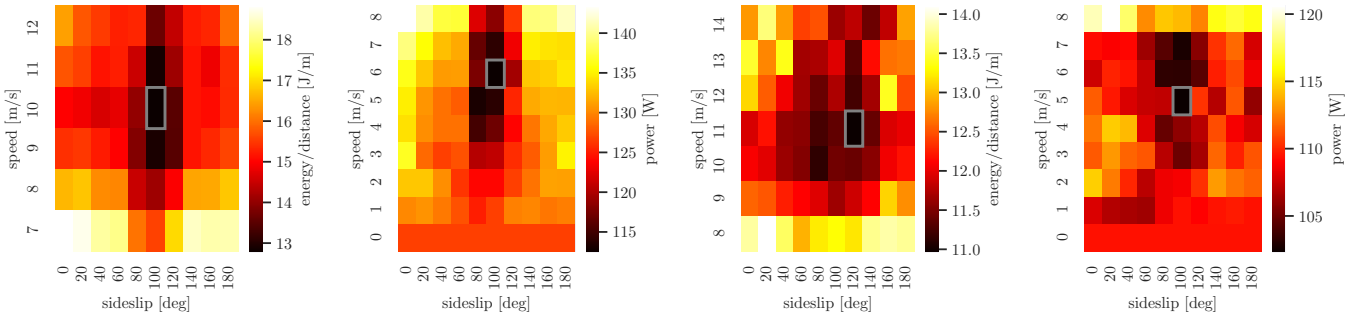
A. Experiment setup

1) System setup: The experiments were performed with a custom-built quadcopter carrying different payloads – with and without an additional box (as shown in Figure 1). The weight of the vehicle without the additional box payload was 0.9 kg, and the box weighs 0.1 kg and has a size of 180×115×80 mm. The distance between the hubs of two diagonal motors is 330 mm and the propeller is 203 mm in diameter. The extremum seeking controller was run on an onboard computer (Jetson Nano), and an mRo Pixracer R15 flight controller ran the standard PX4 firmware [36] including the state estimator and low-level controllers. The Jetson Nano and the Pixracer communicate through a UART link using mavros. The experiments were conducted at a flat grass land at the Richmond Field Station, Richmond, CA (37.916588 N, -122.336667 E).

The control architecture for the vehicle is shown in Figure 4. The extremum seeking controller (with or without adaptive step size) takes in the desired geometric path and instantaneous range cost or endurance cost. The power measurement p_e comes from a power module (Holybro PM06 v2) connected to the battery, and the speed measurement v comes from a state estimator based on a GPS, a range finder (for measuring the flight height) and an IMU. The extremum seeking controller outputs the reference tangential speed r_v and sideslip r_β along the desired path, which are used to parameterize the geometric path into a reference trajectory. The reference trajectory is then tracked by the low-level position and attitude controller, which is a cascaded PID controller.

2) Extremum seeking parameter values: The values of parameters of the standard extremum seeking controller and our proposed adaptive step size extremum seeking controller used throughout the experiments are shown in Table I. The perturbation frequency of reference speed w_v was set to 1 rad/s, and the perturbation frequency of reference sideslip w_β was set to 0.5 rad/s. Multivariable extremum seeking control requires $w_v \neq w_\beta$. While increasing the perturbation frequencies is helpful to improve the convergence rate of the extremum seeking controller, one should make sure they are not too large for the vehicle to track. The cutoff frequencies of the high-pass and low-pass filters were set to be the same as their corresponding perturbation frequency, which was found to work well in the experiments. The magnitude of speed perturbation a_v was set to be 0.5 m/s and the magnitude of sideslip a_β was set to 10 degrees. As long as perturbation can induce large enough change in the cost function compared to measurement noise, it's preferable to keep the perturbation magnitude small. This is helpful in keeping the neighborhood small around the optimal value that the ESC converges to. Finally, the two parameters in the step size adapter γ_v and γ_β are selected depending on how noisy the q_v^2 and q_β^2 are – the lower the parameters' values, the more aggressive q_v^2 and q_β^2 are filtered.

To make a fair comparison, we kept all the control parame-



(a) Range cost with box payload. (b) Endurance cost with box payload. (c) Range cost with no box payload. (d) Endurance cost with no box payload.

Figure 5. Ground truth data of the cost functions' values, with and without an additional box payload. Each square in the heat maps corresponds to 20 seconds' data collected at 50Hz. The optimal value in each case is encircled with a grey rectangle. (a) The range cost with the box payload reaches its minimum at about 10 m/s in speed and 100 degrees in sideslip, which is 24.9% lower than the cost when flying at the maximum speed of 12 m/s with 0 degree in sideslip. (b) The endurance cost with the box payload reaches its minimum at about 6 m/s and 100 degrees in sideslip, which is 12.6% lower than the cost at hovering. (c) The range cost with no box payload reaches its minimum at about 11 m/s and 120 degrees in sideslip, which is 33.5% lower than the cost when flying at the maximum speed of 14 m/s and 0 degree in sideslip. (d) The endurance cost with the box payload reaches its minimum at about 5 m/s and 100 degrees in sideslip, which is 7.0% lower than the cost at hovering.

ters for the two different methods to be the same except k_v and k_β, since they have different meanings for the two methods: the k_v and k_β values are the step sizes for the standard method but are only part of the step sizes for the adaptive method, as shown in Section IV-A. They were empirically tuned in experiments for the two different methods to each achieve the fastest convergence rate.

Table I
VALUES OF EXTREMUM SEEKING PARAMETERS

Parameter	Proposed method	Standard method
a_v	0.5 m/s	0.5 m/s
w_v, w_{hv}, w_{lv}	1 rad/s	1 rad/s
k_v	0.11	0.05
a_β	10°	10°
$w_\beta, w_{h\beta}, w_{l\beta}$	0.5 rad/s	0.5 rad/s
k_β	0.04	0.04
γ_v, γ_β	0.5	N/A

B. Performance evaluation

In the experiments, the quadcopter was commanded to fly along a circular path with constant height and 30 meters in radius.

1) Cost values: To verify that the proposed extremum seeking controller is able to converge close to the optimal speed and sideslip, we experimentally evaluated the optimal range and optimal endurance cost functions. When the vehicle is carrying the box payload, the values of the cost functions at various speed and sideslip are shown at Figure 5(a) and Figure 5(b), while Figure 5(c) and Figure 5(d) show the values without box.

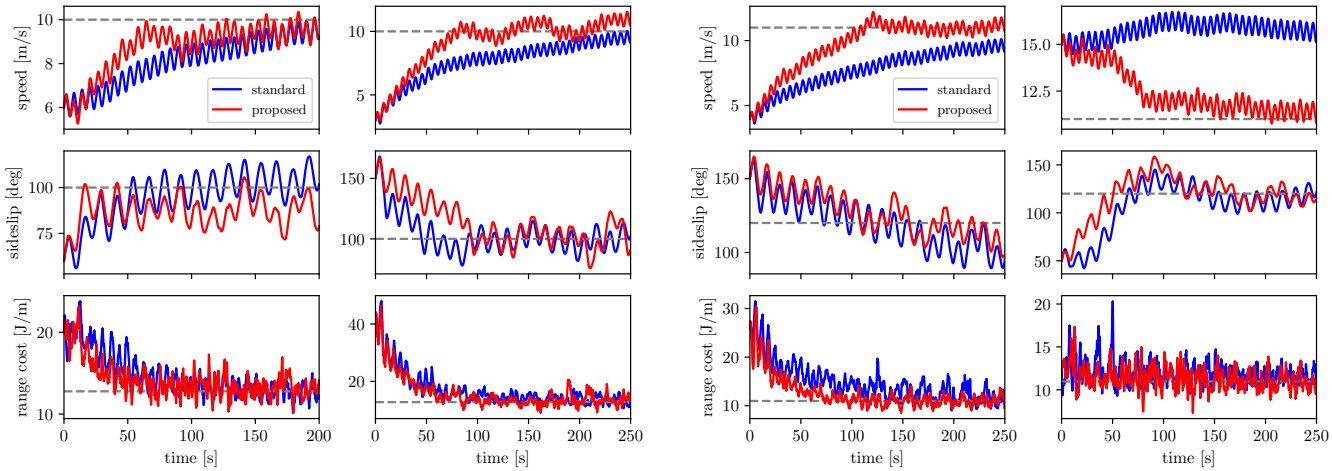
The data shows the importance of flying at the energy efficient speed and sideslip: compared to flying at the maximum speed with a bad sideslip, flying at the optimal range speed and sideslip reduces the energy consumption per distance by 24.9% without payload and 33.5% with a box payload. Besides, compared to hovering, flying at the optimal endurance speed and sideslip reduces the the power consumption by 7.0% without payload and 12.6% with a box payload. By looking at the cost values when flying at the same speed but with different sideslips, we can see finding the optimal sideslip is important in reducing energy cost. In addition, by looking at the cost values when flying at the optimal sideslip but at different speeds, we can see that flying at the fastest speed doesn't achieve the longest flight distance, and in order to achieve the longest flight time (e.g. in surveillance applications), flying at a suitable speed is better than hovering.

2) Optimal range speed and sideslip seeking: When the goal is to find the speed and sideslip which achieve the optimal flight range (distance), the performance of the proposed and standard extremum seeking controller was compared with two different payloads and different initial conditions.

When the additional box payload was carried, the performance of the proposed and standard extremum seeking controller is compared in Figure 6(a) in two tests with different initial conditions. In the first test (column 1), the proposed method converged close to the optimal value in about 50 seconds, while the standard method converged close to the optimal value in about 200 seconds; in the second test (column 2), the proposed method converged close to the optimal value in about 100 seconds, while the standard method converged close to the optimal value in about 250 seconds.

When there was no additional box payload, the performance of the proposed and standard extremum seeking controller is compared in Figure 6(b). In the first test, the proposed method converged close to the optimal value in about 125 seconds, while the standard method converged close to the optimal value in about 250 seconds; in the second test, the proposed method converged close to the optimal value in about 100 seconds, the standard method converged close to the optimal sideslip but didn't converge to the optimal speed in 250 seconds.

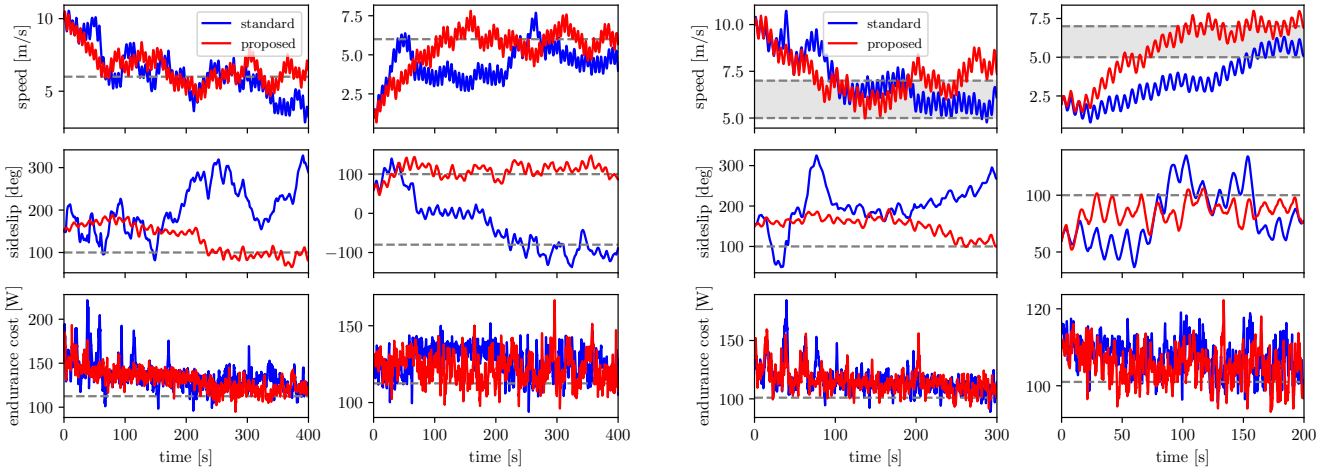
3) Optimal endurance speed and sideslip seeking: When the goal is to find the speed and sideslip which achieve the optimal flight endurance (time), the performance of the proposed



(a) Carrying an additional box payload.

(b) Without carrying an additional box payload.

Figure 6. Optimal range speed and sideslip searching performance comparison between the proposed method (red lines) and the standard method (blue lines). The ground truth values for optimal speed and sideslip are marked as grey dashed lines and grey shaded regions (values from Figure 5). The results when carrying an additional box payload are shown in (a) and the results with no additional box payload are shown in (b). Each column in the subfigures represent a test with a different initial speed and sideslip.



(a) Carrying an additional box payload.

(b) Without carrying an additional box payload.

Figure 7. Optimal endurance speed and sideslip searching performance comparison between the proposed method (red lines) and the standard method (blue lines). The ground truth values for optimal speed and sideslip are marked as grey dashed lines and grey shaded regions (values from Figure 5). The results when carrying an additional box payload are shown in (a) and the results with no additional box payload are shown in (b). Each column in the subfigures represent a test with a different initial speed and sideslip. In the second test of (a), the optimal sideslip is marked at both 100 degrees and -80 degrees. This is because the vehicle is mirror symmetric, such that a sideslip offset of 180 degrees has the same effect on the vehicle's consumption. In (b), the optimal speed is marked as a range between 5 - 7 m/s, because the cost function values are very close in this range with less than 1% difference.

and standard extremum seeking controller was compared with different payloads and different initial conditions. The cost function was instant power.

When the additional box payload was carried, the performance was compared in Figure 7(a) in two tests with different initial conditions. The proposed method converged close to the optimal value in about 250 seconds in the first test (column 1), and about 100 seconds in the second test (column 2). The standard method didn't converge to the optimal sideslip in 400 seconds in the first test, and converged close to the optimal value after about 200 seconds in the second test.

When there was no additional box payload, the performance of the proposed and standard extremum seeking controller is compared in Figure 7(b). In the first test, the proposed method converged close to the optimal value in about 250 seconds, while sideslip of the standard method didn't converge in 300 seconds; in the second test, the proposed method converged close to the optimal value in about 75 seconds, while the standard method converged close to the optimal value in about 175 seconds.

C. Cost of extremum seeking

Since the perturbations are applied by the extremum seeking controller, the power consumption of the vehicle will be higher than flight without perturbations. In this subsection, we compare the optimal values of the cost function without perturbation (i.e. optimal cost values in Figure 5) with the average cost values when flying at the same mean speed and sideslip but with perturbations applied. The cost values with perturbations were also collected experimentally by using the same hardware setup as the collection of ground truth data in Figure 5. The increase in optimal range or endurance cost, with and without box are summarized in Table II.

Table II
OPTIMAL COST INCREASE DUE TO PERTURBATION

Optimization goal	Payload	Cost without perturbation	Cost with perturbation	Cost increase
range	box	12.8 J/m	13.2 J/m	3.1 %
range	none	11.0 J/m	11.4 J/m	4.0 %
endurance	box	112.5 W	116.4 W	3.5 %
endurance	none	101 W	105.2 W	4.2 %

In summary, increase in cost was 3.1 - 4.2 % because of perturbations applied by the extremum seeking controller. To prevent this power consumption increase from happening, the extremum seeking controller can be enabled only when there is a model change (e.g. picking up a new payload), and it should be disabled after convergence.

VI. CONCLUSION

An online, adaptive, model-free method for finding the speed and sideslip that maximize the flight range or endurance of multicopters is proposed in this work. Not dependent on any power consumption model of the vehicle, it is able to adapt to different payloads and is easy to deploy. The proposed method can mitigate the common problem of limited flight range and endurance of multicopters. Based on a novel multivariable extremum seeking controller with adaptive step size, it is able to achieve faster convergence compared to the standard extremum seeking controller with fixed step size.

Through realistic outdoor experiments, we show that this method is able to find the optimal speed and sideslip correctly under different payloads and initial conditions. By flying at the optimal range speed and sideslip, the quadcopter's energy usage per distance can be reduced by 24.9% without payload and 33.5% with the box payload. Also, by flying at the optimal endurance speed and sideslip the power consumption can be reduced by by 7.0% without payload and 12.6% with the box payload.

In addition to multicopters, this method can also be applied to fixed wing aerial robots to find the optimal flight speed (to achieve the longest flight time or distance) whose sideslip is usually not a free degree of freedom in path planning.

ACKNOWLEDGEMENTS

This work was partially supported by the J.K. Zee Fellowship, the UC Berkeley Graduate Division Block Grant Award, and the USDA AI Institute for Next Generation Food Systems (AIFS), USDA award number 2020-67021-32855. Research was also partially sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-20-2-0105. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

The experimental testbed at the HiPeRLab is the result of contributions of many people, a full list of which can be found at hiperlab.berkeley.edu/members/.

REFERENCES

[1] H. Kang, H. Li, J. Zhang, X. Lu, and B. Benes, "Flycam: Multitouch gesture controlled drone gimbal photography," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3717–3724, 2018.

[2] A. Tagliabue, M. Kamel, R. Siegwart, and J. Nieto, "Robust collaborative object transportation using multiple mavs," *The International Journal of Robotics Research*, vol. 38, no. 9, pp. 1020–1044, 2019.

[3] R. Bähnemann, D. Schindler, M. Kamel, R. Siegwart, and J. Nieto, "A decentralized multi-agent unmanned aerial system to search, pick up, and relocate objects," in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, 2017, pp. 123–128.

[4] Y. Wu, Y. Qin, Z. Wang, and L. Jia, "A uav-based visual inspection method for rail surface defects," *Applied Sciences*, vol. 8, no. 7, 2018.

[5] Y. Chen, S. Huang, and R. Fitch, "Active slam for mobile robots with area coverage and obstacle avoidance," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 3, pp. 1182–1192, 2020.

[6] K. Karydis and V. Kumar, "Energetics in robotic flight at small scales," *Interface focus*, vol. 7, no. 1, p. 20160088, 2017.

[7] S. Driessens and P. Pounds, "The triangular quadrotor: A more efficient quadrotor configuration," *IEEE Transactions on Robotics*, vol. 31, pp. 1–10, 10 2015.

[8] C. Holda, B. Ghalamchi, and M. W. Mueller, "Tilting multicopter rotors for increased power efficiency and yaw authority," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2018, pp. 143–148.

[9] A. Kalantari and M. Spenko, "Modeling and performance assessment of the hytaq, a hybrid terrestrial/aerial quadrotor," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1278–1285, 2014.

[10] A. Tagliabue, S. Schneider, M. Pavone, and A.-a. Agha-mohammadi, "Shapeshifter: A multi-agent, multi-modal robotic platform for exploration of titan," in *2020 IEEE Aerospace Conference*, 2020, pp. 1–13.

[11] H. T. Arat and M. G. Sürer, "Experimental investigation of fuel cell usage on an air vehicle's hybrid propulsion system," *International Journal of Hydrogen Energy*, vol. 45, no. 49, pp. 26 370–26 378, 2020, progress in Hydrogen Production and Utilization.

[12] K. P. Jain and M. W. Mueller, "Flying batteries: In-flight battery switching to increase multirotor flight time," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 3510–3516.

[13] F. Morbidi, R. Cano, and D. Lara, "Minimum-energy path generation for a quadrotor uav," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 1492–1498.

[14] F. Morbidi and D. Pisarski, "Practical and Accurate Generation of Energy-Optimal Trajectories for a Planar Quadrotor," in *IEEE International Conference on Robotics and Automation*, Xi'an, China, May 2021. [Online]. Available: https://hal.archives-ouvertes.fr/hal-03178460

[15] N. Bezzo, K. Mohta, C. Nowzari, I. Lee, V. Kumar, and G. Pappas, "Online planning for energy-efficient and disturbance-aware uav operations," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 5027–5033.

[16] C. Di Franco and G. Buttazzo, "Energy-aware coverage path planning of uavs," in *2015 IEEE International Conference on Autonomous Robot Systems and Competitions*, 2015, pp. 111–117.

[17] T. M. Cabreira, C. D. Franco, P. R. Ferreira, and G. C. Buttazzo, "Energy-aware spiral coverage path planning for uav photogrammetric applications," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3662–3668, 2018.

[18] C. Ampatis and E. Papadopoulos, "Parametric design and optimization of multi-rotor aerial vehicles," in *Applications of Mathematics and Informatics in Science and Engineering*. Springer, 2014, pp. 1–25.

[19] N. Michel, A. K. Sinha, Z. Kong, and X. Lin, "Multiphysical modeling of energy dynamics for multirotor unmanned aerial vehicles," in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, June 2019, pp. 738–747.

[20] G. J. Leishman, *Principles of helicopter aerodynamics*. Cambridge university press, 2006.

[21] D. Hong, S. Lee, Y. H. Cho, D. Baek, J. Kim, and N. Chang, "Least-energy path planning with building accurate power consumption model of rotary unmanned aerial vehicle," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14 803–14 817, 2020.

[22] A. Tagliabue, X. Wu, and M. W. Mueller, "Model-free online motion adaptation for optimal range and endurance of multicopters," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 5650–5656.

[23] Y. Tan, W. H. Moase, C. Manzie, D. Nešić, and I. M. Y. Mareels, "Extremum seeking from 1922 to 2010," in *Proceedings of the 29th Chinese Control Conference*, 2010, pp. 14–26.

[24] J. Creaby, Y. Li, and J. E. Seem, "Maximizing wind turbine energy capture using multivariable extremum seeking control," *Wind Engineering*, vol. 33, no. 4, pp. 361–387, 2009.

[25] S. J. Moura and Y. A. Chang, "Lyapunov-based switched extremum seeking for photovoltaic power maximization," *Control Engineering Practice*, vol. 21, no. 7, pp. 971–980, 2013.

[26] Hsin-Hsiung Wang, S. Yeung, and M. Krstic, "Experimental application of extremum seeking on an axial-flow compressor," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 2, pp. 300–309, 2000.

[27] B. Calli, W. Caarls, P. Jonker, and M. Wisse, "Comparison of extremum seeking control algorithms for robotic applications," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 3195–3202.

[28] X. Wu and M. W. Mueller, "In-flight range optimization of multicopters using multivariable extremum seeking with adaptive step size," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 1545–1550.

[29] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[30] M. A. Rotea, "Analysis of multivariable extremum seeking algorithms," in *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No. 00CH36334)*, vol. 1, no. 6. IEEE, 2000, pp. 433–437.

[31] M. Krstić and H.-H. Wang, "Stability of extremum seeking feedback for general nonlinear dynamic systems," *Automatica*, vol. 36, no. 4, pp. 595–601, 2000.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

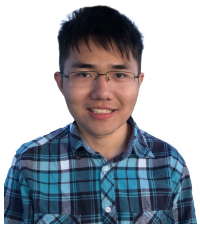
[33] A. Ghaffari, M. Krstić, and D. Nešić, "Multivariable newton-based extremum seeking," *Automatica*, vol. 48, no. 8, pp. 1759–1767, 2012.

[34] H. Khalil, *Nonlinear Systems*, 3rd ed., ser. Pearson Education. Prentice Hall, 2002.

[35] N. Nise, *Control Systems Engineering*, 7th ed. Wiley, 2015.

[36] PX4. (2021) Px4 drone autopilot. [Online]. Available: https://github.com/PX4/PX4-Autopilot/tree/v1.10.1

Xiangyu Wu received his bachelor of science degree from Beijing Institute of Technology, China in 2017 and master of science degree from University of California, Berkeley, USA in 2019. He is currently a Ph.D. candidate at the High Performance Robotics Lab at UC Berkeley. His current research interests are the state estimation and path planning of multi-copters.



Jun Zeng received his B.S.E degree from Shanghai Jiao Tong University (SJTU), China in 2016 and Dipl. Ing. from Ecole Polytechnique, France in 2017. He is currently a Ph.D. candidate supervised by Koushil Sreenath at Hybrid Robotics Group of Mechanical Engineering at University of California, Berkeley, USA. His research interests lies at the intersection of control, optimization and learning on robotics.

Andrea Tagliabue Andrea Tagliabue is a Ph.D candidate with the Laboratory for Information and Decision Systems at MIT. Prior to that, he was a Robotics Engineer at Caltech, affiliate with NASA's Jet Propulsion Laboratory. He received a M.S. in Robotics, Systems and Control from ETH Zurich, and a B.S. with honours in Automation Engineering from Politecnico di Milano. He was a visiting researcher at U.C. Berkeley, and a research assistant at the ETH's Autonomous Systems Lab. His research interests include learning-based methods for control, planning and state estimation for aerial robots.

Mark W. Mueller Mark w. Mueller is an assistant professor of Mechanical Engineering at the University of California, Berkeley, and runs the High Performance Robotics Laboratory (HiPeRLab). He received a Dr.Sc. and M.Sc. from the ETH Zurich in 2015 and 2011, respectively, and a BSc from the University of Pretoria in 2008. His research interests include aerial robotics, their design and control, and especially the interactions between physical design and algorithms.