Energy-Aware Hybrid Event-Triggered Control for Micro-Drones

Jesus O. Rodriguez-Ramirez[1]0009-0007-2817-1329, Julio S. De La Trinidad-Rendon[1]0000-0003-3900-8563 , and Jose Martinez-Carranza[2]0000-0002-8914-1904 , and Hugo G. Gonzalez-Hernandez[1]0000-0001-6495-9980*
1 Tecnologico de Monterrey, School of Engineering and Sciences, Puebla, Mexico
2 Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Puebla, Mexico

http://www.imavs.org/

ABSTRACT

This paper addresses the important balance between control performance and energy use in networked micro-drones. We propose and test a family of hybrid event-triggered control (HETC) methods that are designed to be both dependable and efficient. The core of our strategy is the combination of a fixed safety timer, which ensures system safety, with an adjustable threshold that responds to the drone's state to balance communication, performance, and energy. We look at several practical ways to adjust the threshold, including methods based on integral, derivative, and state-feedback approaches. The effectiveness of these strategies is tested using detailed simulations of a 6-DOF quadrotor. The validation uses dour different flying paths which include active tracking in straight and curved trajectories as well as stationary hovering, to test the controller in various conditions. The results show meaningful reductions not only in communication frequency but also in a measure of total energy use. This study shows our practical HETC approach is a promising solution for devices with limited resources and provides a solid starting point for future research on more advanced triggers and testing with real-world network problems.

1 INTRODUCTION

The increasing autonomy of Micro Aerial Vehicles (MAVs) presents a modern control challenge: greater autonomy and performance require more advanced onboard computing and frequent communication, yet the physical limits of these devices, especially battery life, require careful use of resources [1]. This problem is most significant in Networked Control Systems (NCS), where MAVs must constantly exchange information with ground stations or other agents, making the energy used for communication a major limit for mission duration.

The traditional time-triggered control method, which uses updates at regular, frequent intervals, is a major source of this inefficiency. By sending control data at a fixed rate based on a worst-case scenario, these systems waste energy sending unneeded data during normal flight. Event-Triggered Control (ETC) solves this by changing the control approach from updating at set times to updating only when necessary [2, 3]. This non-periodic approach has been shown to greatly reduce network use, a topic explored in detailed reviews on ETC for nonlinear systems [4, 5]. Many adaptive/learning or CLF-based triggers require evaluating Lyapunov/ISS bounds (and their Lie derivatives) online and, in some cases, solving small optimization problems (e.g., QP/SDP) or running neural inference at control rates. These steps increase CPU load, memory footprint, and code size, which can exceed MAV-class microcontrollers. See emulation/self-triggered and PETC analyses in [6, 7, 8].

Research has shifted from fixed triggers to flexible and adaptive methods that are more efficient, including those based on fuzzy neural networks [9] and reinforcement learning [10]. For Unmanned Aerial Vehicles (UAVs) specifically, ETC has been applied to periodic communication schemes [11, 12], swarm control [13], and security [14]. A key challenge is making sure it performs well even with constant outside disturbances [15, 16] and modeling errors, which has led to more research on dependable and adaptive ETC for autonomous vehicles in general [17, 18].

Despite this progress, two key problems are often overlooked. First, many studies only focus on sending fewer data packets, ignoring the total energy used, which includes the energy the motors use to fix trajectory-tracking errors. A trigger that sends updates too slowly might save on communication, but lead to poor trajectory-tracking that requires powerful, energy-intensive corrections. Second, many of the best-performing trigger rules from theory can be too complex to run, making them hard to use on MAVs with limited computing power. By contrast, our rules use only vector norms, constant-gain arithmetic, and (optionally) first-order filters—no optimizers, factorizations, or large matrices—so the per-step cost is $O(1)$ floating-point operations, appropriate for Cortex-M–class MCUs [8].

This paper explores a practical alternative: a set of simple heuristic adaptive triggers, when used in a dependable hybrid system, can save a large amount of energy without the complexity of other methods. We propose a Hybrid Event Triggered Control (HETC) framework that combines an adaptive

*Email address(es): hgonz@tec.mx

http://www.imavs.org/

event-trigger with a fixed safety timer ($T_{\max}$) to guarantee stability. Our main goal is to measure the balance between trajectory-tracking accuracy, communication frequency, and, most importantly, total energy use.

1.1 Quadrotor Dynamic Model

We model the quadrotor as a 6-DOF rigid body. Let $\{W\}$ be the fixed world frame and $\{B\}$ be the frame attached to the drone's body. The state is $(\boldsymbol{p}, \boldsymbol{v}, \boldsymbol{R}, \boldsymbol{\Omega}) \in \mathrm{SE}(3)(3) \times \mathbb{R}^3$, where $\boldsymbol{p}, \boldsymbol{v} \in \mathbb{R}^3$ are the position and velocity in $\{W\}$, $\boldsymbol{R} \in \mathrm{SO}(3)$ is the rotation matrix from $\{B\}$ to $\{W\}$, and $\boldsymbol{\Omega} \in \mathbb{R}^3$ is the angular velocity in $\{B\}$. The dynamics are given by [19]:

$$\dot{\boldsymbol{p}} = \boldsymbol{v} \tag{1}$$

$$m\dot{\boldsymbol{v}} = mg\boldsymbol{e}_3 - \boldsymbol{R}T\boldsymbol{e}_3 + \boldsymbol{d}_f(t) \tag{2}$$

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\hat{\boldsymbol{\Omega}} \tag{3}$$

$$\boldsymbol{J}\dot{\boldsymbol{\Omega}} = -\boldsymbol{\Omega} \times (\boldsymbol{J}\boldsymbol{\Omega}) + \boldsymbol{\tau} + \boldsymbol{d}_\tau(t) \tag{4}$$

where m is the mass, $\boldsymbol{J}$ is the inertia matrix, g is gravity, $\boldsymbol{e}_3 = [0, 0, 1]^T$, T is the total thrust, and $\boldsymbol{\tau}$ is the torque vector. The terms $\boldsymbol{d}_f(t)$ and $\boldsymbol{d}_\tau(t)$ are bounded external disturbances. The hat map $\hat{\cdot}$ is the skew-symmetric matrix representation.

1.2 Baseline Control Architecture and Implementation

Our method builds on a standard nested-loop controller:

- **Inner Attitude Loop:** A high-rate controller (e.g., 200 Hz or higher) regulates the drone's rotation by computing the motor torques $\boldsymbol{\tau}$ required to track a desired orientation $\boldsymbol{R}_d(t)$.
- **Outer Position Loop:** A continuous-time nonlinear controller (e.g., geometric on $\mathrm{SE}(3)(3)$) tracks the reference path $\boldsymbol{p}_d(t)$ and its derivatives. Its output is the nominal control signal $\boldsymbol{u}(t) = [T(t), \boldsymbol{R}_d(t)]^T$.

In practice, $\boldsymbol{u}(t)$ cannot be transmitted continuously due to communication and energy constraints. Instead, it is sampled at discrete times $\{t_k\}_{k \in \mathbb{N}_0}$, which the proposed HETC framework determines.

Between update instants, the motors apply the most recently transmitted command using a Zero-Order Hold (ZOH) mechanism. The executed control signal is therefore

$$\boldsymbol{u}_{drone}(t) = \boldsymbol{u}(t_k), \quad \forall t \in [t_k, t_{k+1}), \tag{5}$$

i.e., constant between successive triggers. This implementation introduces a discrepancy between the nominal signal $\boldsymbol{u}(t)$ and the applied signal $\boldsymbol{u}_{drone}(t)$, defined as the measurement error $\boldsymbol{e}_m(t) = \boldsymbol{u}(t) - \boldsymbol{u}_{drone}(t)$. The event-triggering mechanism monitors and regulates this error.

1.3 Total Energy Use Model

System efficiency is evaluated with an energy model that accounts for both communication and actuation costs:

$$E = \int_0^{t_f} P_{base}\, dt + N_{pkts}\, E_{tx} + \int_0^{t_f} P_{act}(\boldsymbol{u}_{drone}(t))\, dt, \tag{6}$$

where P_{base} is the constant baseline consumption of onboard electronics, N_{pkts} the number of transmitted packets, and E_{tx} the energy per packet. Since P_{base} is independent of the control scheme, comparisons focus on:

- **Communication Energy (E_{comm}):** $N_{pkts}\, E_{tx}$, reduced by event-triggering strategies.
- **Actuation Energy (E_{act}):** The integral of motor power, commonly approximated as $P_{act} \propto T(t)^2$.

This formulation highlights the trade-off: frequent updates improve tracking but increase communication cost, whereas sparse updates reduce communication but may require larger corrective maneuvers and thus higher motor energy. Effective triggers balance these contributions to minimize $E_{comm} + E_{act}$.

The energy model in Eq. 6, which separates baseline, communication, and actuation terms, follows standard UAV power formulations and ETC studies that include communication cost [1, 11, 12].

2 PROPOSED HYBRID ADAPTIVE CONTROL FRAMEWORK

The framework is intended to be reliable, efficient, and suitable for resource-constrained platforms.

If the baseline continuous controller guarantees closed-loop stability under sample-and-hold with period $h \leq T_{\max}$ (standard emulation condition), then enforcing the time guard preserves this property. From sampled-data and PETC theory, the resulting hybrid loop is input-to-state stable (ISS) with respect to disturbances and measurement error. The event condition further improves accuracy between mandatory watchdog updates [8, 7, 6].

2.1 Hybrid Event- and Time-Triggering Law

The triggering mechanism combines an event condition with a periodic safeguard. Let t_k denote the last update time and $\boldsymbol{e}_m(t) = \boldsymbol{u}(t) - \boldsymbol{u}(t_k)$ the measurement error. The next update time t_{k+1} is defined as

$$t_{k+1} = \inf\{t > t_k \mid \underbrace{\|\boldsymbol{e}_m(t)\| \geq \sigma(t)}_{\text{Event Condition}} \vee \underbrace{t - t_k \geq T_{\max}}_{\text{Time Guard}}\}. \tag{7}$$

The event condition enforces updates when the control error exceeds the adaptive threshold $\sigma(t)$, while the time guard $T_{\max}$ prevents excessive inter-update intervals. Together, these conditions guarantee stability and avoid unnecessary communication.

http://www.imavs.org/

2.2 *Adaptive Threshold Structure*

The threshold $\sigma(t)$ is defined using the error vector $\boldsymbol{x}_e = [\boldsymbol{e}_p^T, \boldsymbol{e}_v^T]^T$:

$$\sigma(t) = \underbrace{\beta(t)}_{\text{Adaptive Offset}} + \underbrace{\alpha \left\|\boldsymbol{e}_p(t)\right\| + \gamma \left\|\boldsymbol{e}_v(t)\right\|}_{\text{Error-Dependent Scaling}}, \tag{8}$$

where α and γ are fixed gains, and $\beta(t)$ is a time-varying offset. The offset term allows the trigger sensitivity to be adjusted online according to mission demands or system behavior.

The gains α, γ, and the adaptation parameters were chosen to balance trigger sensitivity and stability. Values were tuned to maintain RMS tracking error below $0.1\,\mathrm{m}$ while avoiding excessive updates and preserving stability, following the typical trade-offs reported in event-triggered UAV literature.

To implement this mechanism, we introduce three alternative adaptation laws for $\beta(t)$, each emphasizing a different objective:

2.3 *Adaptation Rule 1: Integral Control for Rate Regulation*

Objective Regulate the communication rate by driving the average number of updates, N_{avg}, toward a target N^*.

Rule An integral controller with leakage:

$$\dot{\beta}(t) = -k_I\big(N_{\text{avg}}(t) - N^*\big) - \lambda\big(\beta(t) - \beta_0\big), \tag{9}$$

where $\beta(t)$ is the adaptive offset. Increasing β raises the trigger threshold, reducing update frequency; decreasing β lowers the threshold, increasing updates. The leakage term, governed by λ, prevents windup by returning β toward a nominal value β_0 when the rate stabilizes.

Implementation Considerations Performance depends on the averaging window used for N_{avg}: shorter windows improve responsiveness but amplify noise, whereas longer windows provide smoother estimates at the cost of slower adaptation. Gains k_I and λ must be tuned to achieve stability without oscillation.

2.4 *Adaptation Rule 2: Derivative Feedback*

Objective To adjust the trigger based on the rate of change of the position error, increasing sensitivity when the error grows and reducing it when the error decreases.

Rule

$$\beta(t) = \beta_0 - k_D \frac{d}{dt} \left\|\boldsymbol{e}_p(t)\right\|, \tag{10}$$

where a positive derivative decreases the offset β, making an update more likely, and a negative derivative increases it, reducing update frequency. This predictive adjustment helps limit maximum error growth during disturbances and can lower the total actuation energy required for correction.

Implementation Considerations Direct differentiation is noise-sensitive. A filtered derivative estimate (e.g., via a first-order high-pass filter or observer) is required. The filter cutoff frequency must be chosen carefully to balance responsiveness and robustness.

2.5 *Adaptation Rule 3: State-Feedback for Context-Awareness*

Objective Define a rule that adapts to the drone's operating condition (e.g., transit versus hover) without requiring memory of past data or derivative estimates.

Rule

$$\beta(t) = \beta_0 + k_p \left\|\boldsymbol{p}(t)\right\| + k_v \left\|\boldsymbol{v}(t)\right\|, \tag{11}$$

where $\boldsymbol{p}(t)$ and $\boldsymbol{v}(t)$ are the position and velocity. When the drone is moving quickly or operating farther from the target, the threshold increases, reducing communication demand. During hovering or precise maneuvers, the threshold decreases, prioritizing accuracy.

Implementation Considerations The parameters k_p and k_v must be scaled to the mission's workspace and expected speeds. This rule is less adaptable to changing tasks than the other laws but remains straightforward and reliable.

3 SIMULATION STUDY AND DISCUSSION OF RESULTS

3.1 *Simulation Environment*

The proposed HETC framework was implemented in MATLAB/Simulink using a 6-DOF quadrotor model with parameters corresponding to the Crazyflie 2.1 micro-drone ($m = 0.031$ kg).

As a baseline, we employed a geometric tracking controller on SE(3)(3) [19], which directly regulates both position and orientation.

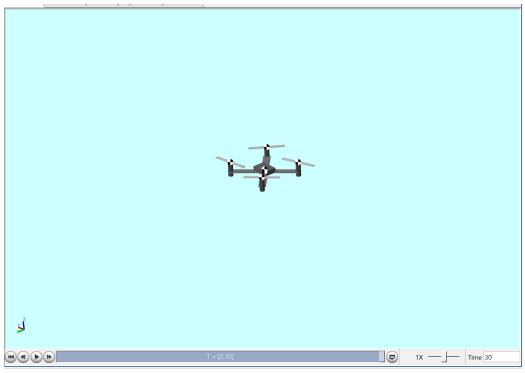

Figure 1: Simulation environment used for the evaluation of the proposed hybrid event-triggered control.

Figure 1 shows the simulated quadrotor environment used to test the proposed control scheme.

http://www.imavs.org/

Performance and energy efficiency were evaluated on a set of 30-seconds trajectory experiments designed to combine periods of active motion and near-stationary hovering. This provided a test of controller behavior under varied conditions. Four representative trajectories were considered:

- **Circle:** A horizontal circular path of 1m radius at constant altitude.
- **Lemniscate:** A figure-eight path designed to test adaptability.
- **Square:** A stepwise path with sharp corners.
- **L path:** A piecewise-linear "L-shaped" trajectory with alternating segments.

Together, these trajectories provide a controlled testbed for analyzing how the hybrid triggering mechanism balances communication savings and actuation energy across diverse operating conditions. Reported results for UAV/VTOL systems show 40–80% packet-rate reductions with less than 10–15% degradation in tracking accuracy under ETC and PETC schemes [3, 4, 5, 11, 12, 13].

3.2 Discussion of Results

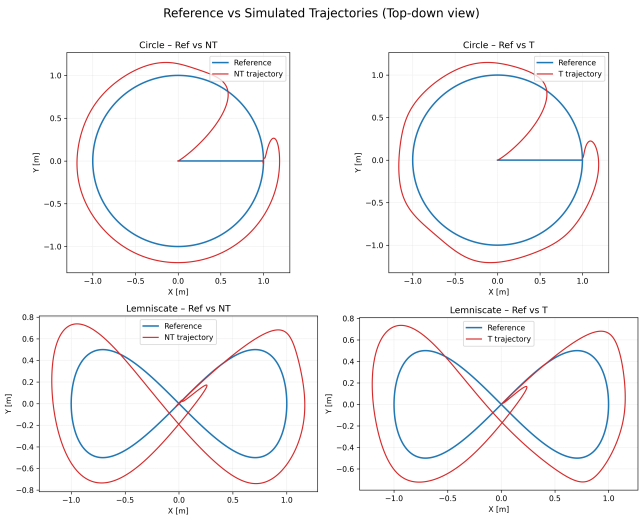

Figure 2: Top-down view of the reference and simulated trajectories for circular and lemniscate paths.

Figure 2 compares the reference and simulated trajectories for the circular and lemniscate paths under non-triggered and hybrid event-triggered control. The NT trajectories correspond to the non-triggered (periodic) controller, while T trajectories correspond to the hybrid event-triggered implementation.

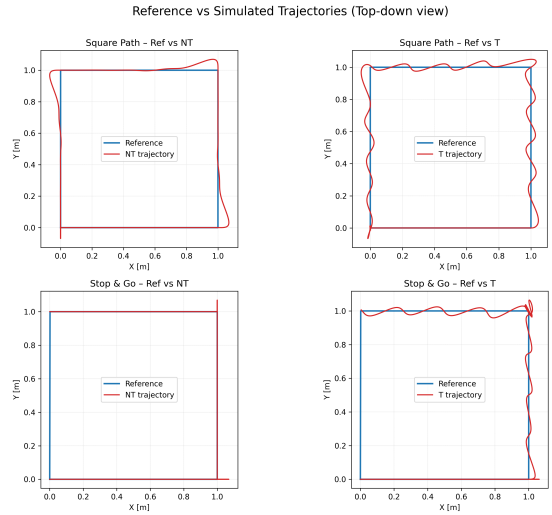

Figure 3: Comparison of reference and simulated trajectories for the square and stop-and-go paths.

Figure 3 illustrates trajectory tracking for square and stop-and-go paths, showing the effect of the hybrid trigger compared to the fixed-rate baseline. The non-triggered (NT) case uses fixed-rate updates, while the triggered (T) case applies the proposed hybrid event-triggered policy.

Straight-line segments exhibit small oscillations because low curvature reduces trigger activity, causing delayed corrections, whereas curved paths naturally maintain frequent updates and smoother tracking, the oscillations also contribute greatly to an increase on actuation energy as shown on table 1.

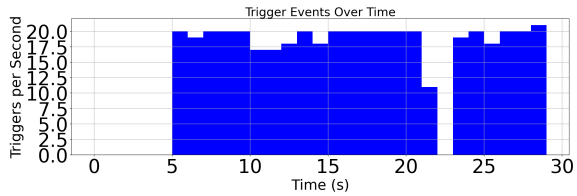

Figure 4: Visualization of the HETC trigger signal over the a multi-phase trajectory to summarize different trials. The histogram shows the number of trigger events on that second

The simulation results confirm that the proposed HETC framework reduces resource consumption while maintaining tracking performance. The observed behavior is consistent across the evaluated trajectories and ablation studies.

3.2.1 Analysis of Triggering Behavior

Figure 4 illustrates the distribution of trigger events over time. The trigger activity closely follows the multi-phase trajectory:

- **Dynamic Tracking:** Frequent updates are generated to maintain path accuracy, resembling periodic control. Small gaps indicate instances where communication is avoided without performance loss.

- **Hovering:** With the drone stabilized, the tracking error remains negligible and the trigger rarely activates, leading to substantial communication savings relative to periodic control.

- **Reorientation and Settling:** The yaw maneuver induces a transient error, producing a short burst of updates. After the maneuver, the trigger rate decreases again as the system stabilizes.

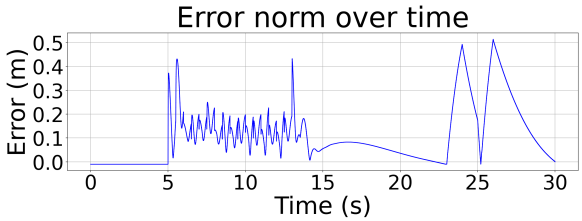

Figure 5: Evolution of the position tracking error norm, $||e_p(t)||$ (in meters) vs Time in seconds

The evolution of the position error norm corresponds closely to the trigger activity shown in Fig. 4:

- **Dynamic Tracking:** During trajectory following, the error remains small (typically below 0.3 m) with an oscillatory pattern characteristic of continuous corrective actions. The high trigger rate is consistent with the need for frequent adjustments.

- **Hovering:** Once stabilized in hover, the tracking error quickly decreases to near zero, confirming that extended periods without triggers occur because the control objective is maintained.

- **Yaw Reorientation:** A reorientation maneuver produces temporary error spikes of up to approximately 0.55 m, after which the controller restores accuracy, demonstrating robustness to internal disturbances.

- **Final Settling:** Following the maneuver, the error again converges to near zero, and the system resumes stable, low-energy operation.

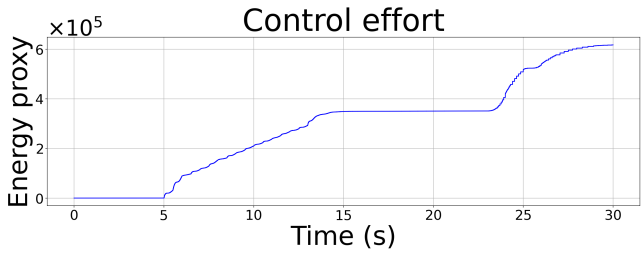

Figure 6: Cumulative actuation energy (E_{act}) vs Time in seconds, proxied by the integral of the squared output of the controllers with no HETC.

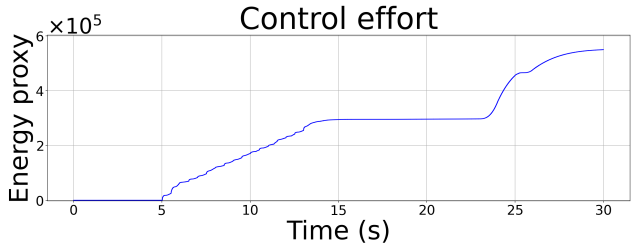

Figure 7: Cumulative actuation energy (E_{act}) vs Time in seconds, proxied by the integral of the squared output of the controllers with HETC.

The best way to measure the system's success is its impact on total energy use, as modeled in Eq. (6). While the analysis of the trigger signal in Fig. 4 confirmed a large reduction in communication energy (E_{comm}), a detailed look at the motor energy (E_{act}) shows an even better result.

Figures 6 and 7 present a comparison of the cumulative motor energy for the baseline periodic controller, The motor energy is estimated by calculating the integral of the squared thrust command over the simulation time, the first one being the system without our event triggering logic.

Looking closely at the plots shows two key things:

- **Similar Power Profiles:** Both the periodic and HETC systems show similar power usage patterns that match the mission phases. As expected, energy use increases rapidly during active flight and is almost flat when hovering.

- **Reduced Total Actuation Energy:** But there is a big difference in the total energy used. The periodic controller's motor energy continually climbs, reaching a final value of about 3.5 x10^5 units. In contrast, the HETC system's total motor energy is lower, settling around 3.0 x10^5 units. This is a reduction of about 14% in the energy used by the motors.

http://www.imavs.org/

http://www.imavs.org/

Table 1: ΔPkts [%] = packet reduction, E-cons = Energy consumed by both communication and actuation

Trajectory	ΔPkts [%]	E-cons	RMS error [m]
Circle (R=1 m)	28	↓	0.07
Lemniscate	24	↓	0.06
Square (step corners)	39	↑↑	0.018
L path (piecewise-linear)	41	↑↑	0.022

Table 1 summarizes the results per trajectory obtained with the proposed HETC compared to the periodic baseline. For smooth trajectories such as the circle and lemniscate, communication demand is reduced by about 25–30% with negligible loss of tracking accuracy (RMS error below 0.1 m) and even slight energy savings, since fewer corrective maneuvers are required. In contrast, trajectories with sharp corners or long straight segments (square and L-path) achieve higher packet savings (around 40%), but this comes at the cost of increased actuation effort. The oscillatory corrections required in these cases lead to additional motor energy consumption (indicated qualitatively in the table), even though the RMS error remains low. These results highlight the trade-off: smoother paths yield balanced savings across communication and energy, while step-like trajectories shift the burden toward actuation, reducing the overall energy benefit.

4 CONCLUSION AND FUTURE WORK

This paper presented a hybrid control framework that combines event- and time-based triggers to address energy efficiency in MAVs. By integrating a safety timer with simple adaptive trigger rules, the method balances control performance with reduced communication and actuation energy.

The simulation study, based on a total energy model, shows that these adaptive strategies lower both communication frequency and overall energy consumption compared to periodic and fixed-threshold approaches. All proposed rules are applied within the same framework, ensuring practicality for MAV-class hardware.

Future work will focus on extending the evaluation beyond ideal simulations to include more realistic network conditions with delays and packet losses, as well as hardware-in-the-loop experiments. Another direction is the development of triggers derived from control-theoretic formulations, such as those based on Control Lyapunov Functions (CLFs), with the goal of assessing whether their theoretical performance gains justify the additional computational complexity on embedded platforms.

Moreover an important limitation observed in the present study is that the triggering mechanism and control design do not explicitly account for the inertia of the vehicle. As a result, aggressive maneuvers and step-like trajectory segments often introduce large overshoots and oscillations, which in turn increase actuation energy consumption. While the proposed adaptive thresholds successfully regulate communication effort, they are less effective in anticipating the high corrective demands associated with inertial dynamics. Future work will focus on incorporating inertia-aware terms into the triggering condition, such as acceleration, jerk, or command-rate penalties, to prevent delayed updates during rapid maneuvers. By making the event-trigger more sensitive to inertial effects, it is expected that both overshoot and oscillatory energy losses can be reduced, improving overall efficiency without compromising tracking performance.

REFERENCES

[1] Randal W. Beard and Timothy W. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, Princeton, NJ, 2012.

[2] Paulo Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685, 2007.

[3] WPMH Heemels, KH Johansson, and P Tabuada. An introduction to event-triggered and self-triggered control. *Proceedings of the 51st IEEE Conference on Decision and Control*, pages 3270–3285, 2012.

[4] Rida Yaqoob and et al. A recent survey of event-triggered control of nonlinear systems. *arXiv preprint arXiv:2202.01684*, 2021.

[5] Derui Ding, Qing-Long Han, and et al. Dynamic event-triggered control and estimation: A survey. *International Journal of Systems Science*, 52(13):2353–2376, 2021.

[6] Alvaro Anta and Paulo Tabuada. To sample or not to sample: Self-triggered control for nonlinear systems. *Automatica*, 46(5):949–958, 2010.

[7] Arjan Postoyan, Paulo Tabuada, Dragan Nešić, and Alvaro Anta. A framework for event-triggered and self-triggered control. *IEEE Transactions on Automatic Control*, 60(10):2661–2675, 2015.

[8] W. P. M. H. Heemels, M. C. F. Donkers, and A. R. Teel. Periodic event-triggered control for linear systems. *IEEE Transactions on Automatic Control*, 58(4):847–861, 2013.

[9] Kangkang Sun, Jianbin Qiu, Hamid Reza Karimi, and Yili Fu. Event-triggered robust fuzzy adaptive finite-time control of nonlinear systems with prescribed performance. *IEEE Transactions on Fuzzy Systems*, 29(6):1460–1471, 2021.

[10] Umair Siddique, Hassan Khalil, and Xiang Xu. Adaptive event-triggered reinforcement learning control for complex nonlinear systems. *arXiv preprint arXiv:2403.12069*, 2024.

[11] Ángel Cuenca, Oscar Reinoso, and Pablo Gil. Periodic event-triggered sampling and dual-rate control for a wireless ncs with applications to uavs. *IEEE Transactions on Industrial Electronics*, 65(8):6510–6520, 2018.

[12] Tong WU, Jie WANG, and Bailing TIAN. Periodic event-triggered formation control for multi-uav systems with collision avoidance. *Chinese Journal of Aeronautics*, 35(8):193–203, 2022.

[13] Xiuxia Yang, Hao Yu, Yi Zhang, and Wenqiang Yao. Distributed event-triggered control for uav swarm target fencing with network connectivity preservation and collision avoidance. *Defence Technology*, 2025.

[14] Shijie Han, Chen Wang, and Xiaoyu Liu. Event-triggered attitude controller design for uavs under cyber-attacks. In *IEEE International Symposium on Industrial Electronics (ISIE)*, pages 1099–1104. IEEE, 2024.

[15] Peng Zhao, Yuan Liu, and Han Xu. Event-triggered trajectory tracking control for quadrotor uavs subject to external disturbances. *International Journal of Robust and Nonlinear Control*, 2025. Early Access.

[16] Liang Kong, Jun Wu, and Tianyang Zhang. Robust event-triggered tracking control for a uav with non-vanishing disturbances. *Automatica*, 159:111224, 2024.

[17] Hongguang Meng, Derui Ding, Hao Su, and Li Wang. Event-triggered robust output regulation for nonlinear systems: A time regularization approach. *International Journal of Robust and Nonlinear Control*, 32(17):7403–7421, 2022.

[18] Mehmet Arsal and Tansel Yucelen. Adaptive event-triggered consensus control for auvs with output feedback and energy efficiency. *Journal of Marine Science and Engineering*, 12(4):511, 2024.

[19] Taeyoung Lee, Melvin Leok, and N. Harris McClamroch. Geometric tracking control of a quadrotor UAV on SE(3). In *49th IEEE Conference on Decision and Control (CDC)*, pages 5420–5425, Atlanta, GA, USA, 2010. IEEE.

http://www.imavs.org/