Event-Triggered Neural Network Adaptive Hybrid Attitude Control

João Pedro Silvestre[a], Joel Reis[b], Pedro Casau[c], Paulo Oliveira[d]

[a]*Electrical and Computer Engineering Department, University of California at Los Angeles, Los Angeles, CA 90095 USA*
[b]*Faculty of Science and Technology, University of Macau, Taipa, Macau*
[c]*Instituto de Telecomunições and Department of Electronics, Telecommunications and Informatics, University of Aveiro, Portugal.*
[d]*IDMEC-Institute of Mechanical Engineering and ISR-Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal.*

Abstract

In this paper, we develop an adaptive controller for rigid-body attitude tracking systems that combines a neural network architecture with an event-triggered mechanism. We design a quaternion-based nonlinear controller that employs a neural network to compensate for state-based disturbances and/or modeling errors. The attitude tracking error can be made arbitrarily small by appropriate tuning of the controller parameters. Using well-posed hybrid systems theory, we show that the proposed controller is robust to noise and does not suffer from chattering or unwinding phenomena. Afterward, we extend our design to accommodate an event-triggered mechanism featuring a sample-and-hold of the state signal, which lowers the required number of controller updates sent to the actuators, and therefore reduces wear on moving mechanical parts. The proposed design renders a compact neighborhood of the null error set semi-globally asymptotically stable for the closed-loop system. Simulation results are presented to assess and illustrate the performance attained by our solution.

Keywords: Application of nonlinear analysis and design; Adaptive control; Tracking; Artificial neural network; Event-triggered control.

1. Introduction

The problem of attitude control is crucial in many areas, including satellite navigation, underwater vehicles, and unmanned aerial vehicle operations. The problem has been addressed through various attitude parameterizations; however, some, such as Euler angles, suffer from inherent singularities. A common way to avoid this problem is to use unit-quaternions, which provide a singularity-free representation of the special orthogonal group and, compared to rotation matrices, offer a more compact parametrization. However, this approach may cause the closed-loop system to unwind, an issue that stems from the unit-quaternions' double cover of the attitude space. In summary, since each point in the attitude space has two different representations, there is no guarantee that a continuous controller will always drive the states to the nearest equilibrium, and, in many cases, the closed-loop system may need to *unwind* before converging to the reference. This problem has inspired several solutions, from discontinuous control [1, 2] to hybrid systems [3, 4], and while both approaches are able to solve it, as stated in [3], (memoryless) discontinuous feedback controllers are not robust to measurement noise, unlike hybrid controllers. The main cause is that an arbitrarily small amount of noise prevents the global stabilization of the closed-loop system, causing it to chatter. Thus, a hybrid controller is able to solve both the unwinding and chattering phenomena while being simultaneously robust to noise.

Email addresses: `joaosilvestre@ucla.edu` (João Pedro Silvestre), `joelreis@um.edu.mo` (Joel Reis), `pcasau@ua.pt` (Pedro Casau), `pjcro@isr.ist.utl.pt` (Paulo Oliveira)

Hybrid control in switched systems is important for two reasons: not only is it possible to avoid chattering and unwinding, but it also allows the implementation of switching mechanisms such as event-triggered techniques, while maintaining properties such as global asymptotic stability [5, 6]. These advantages propelled the usage of hybrid systems in many applications, from estimators [7, 8] to controllers [9], exploiting the hybrid systems' ability to connect disconnected sets using discrete jumps.

In practical controller implementations, it is required to sample data with a sufficiently small sampling period. However, this requirement may be difficult to impose in scenarios with limited communication bandwidth for signal transmission, transmission delay, and low computational power. Moreover, in most cases it is not required to sample the control signal periodically; thus, we need a protocol which lowers the total amount of data sent to a minimum, while keeping the desired performance. As discussed in [10], various methods, including input quantization, periodic sampling, and event-triggered control, can address the challenge of limited bandwidth. Unlike input quantization [11], which lowers the amount of data transmitted in each communication, periodic and event-triggered sampling aim to lower the number of messages. Periodic sampling is not ideal, as the frequency of messages should depend on how the error of the closed-system is currently behaving; otherwise, the performance and stability properties may deteriorate. Event-triggered control solves this problem by providing an update to the actuators based on the current error of the measurements or the performance of the system [6]. The latter yields several major advantages, since we can impose a minimum bound on the convergence speed.

The final topic addressed by this paper concerns adaptation. When a system is being deployed, adaptation is essential to maintain performance in the presence of several disturbances. If disturbances could be modeled meticulously, then it would be sensible to develop estimators using techniques such as adaptive backstepping [12], but in most cases it is hard to design adaptation laws for all the disturbances. By using a neural-network-based approach, it is possible to design a closed-loop system that adapts to any state-based disturbance. Such a design ensures that the closed-loop tracking error converges to a precompact neighborhood of the null error set. Different networks can be used, such as standard neural networks [13], recurrent neural networks [14], or radial basis function neural networks (RBFNNs) [15]. The latter allow a better design of the error set, and they are ideal when the state space is compact as in the unit-quaternion space, since in most implementations we define the centroids of the functions *a priori*. Deep architectures have also been employed in adaptive control on Euclidean state spaces, with real-time Lyapunov-based update laws for every layer of a feedforward network [16], and applied to sensor and actuator faults in strict-feedback systems [17]. Since such networks are not linear in their weights, these designs typically rely on an a priori known bound on the ideal parameters, enforced through a projection operator, and they do not consider an event-triggered actuation structure. In the attitude tracking problem the function to be approximated is defined on a low-dimensional compact set, so a single hidden layer already provides the required accuracy while retaining the adaptive control structure that our adaptation law and hybrid analysis exploit.

Most works in the literature that use adaptive neural networks in attitude control [18, 19, 20] lack the ability to solve the unwinding phenomenon. Other works, such as [21], address adaptive control without unwinding by using the sign function, but this approach prevents global results and makes the controller susceptible to chattering. The use of either continuous or discontinuous control in this context can cause chattering and may compromise the achievement of global results, as discussed in [22, 3]. Moreover, many event-triggered spacecraft controllers cannot tackle the unwinding phenomenon, focusing only on the chattering, which may be caused by either a sliding surface or the event-triggered system [10, 23, 24]. Furthermore, the majority of event-triggered solutions do not use Lyapunov-based jumps; instead, the jumps are based on the error of the measured states, rather than on the current performance. However, some works on event-triggered control or adaptive neural network control also focus on the problem of unwinding. Nonetheless, these works usually either use continuous controllers with rotation matrices [25], which leads to unstable equilibria, or use discontinuous controllers [26, 27], which, as mentioned previously, creates a chattering problem at the origin when noise is present. We consider event-triggered control, using a similar approach to [28], by exploiting hybrid systems to guarantee that the controller has semi-global asymptotic stability, and each jump is triggered by the Lyapunov function, ensuring no Zeno solutions. Finally, while there exist techniques combining neural network adaptation with event-triggered approaches [29, 30], to the best of our knowledge, our solution is the first tackling the problem of unwinding and chattering. Moreover, the strategies developed in [29, 30] do not mention any results on their robustness to noise.

Event-triggered actuation is one of several ways of restricting the instants at which the control signal acts, alongside impulsive [31, 32], intermittent [33] and memory sampled-data control [34], which prescribe these instants rather than deriving them from the closed-loop performance. Our triggering rule follows [6], where the condition is evaluated on the derivative of the Lyapunov function. That construction is not directly applicable here, since the Lyapunov function depends on the estimation errors, which are unavailable to the controller; we therefore bound it by an inequality that reduces to a computable condition. Relative to [35], the parametric estimator is replaced by a neural compensator, which removes the requirement that the disturbance be linearly parametrized in known regressors, and the sample-and-hold layer requires the analysis to be carried out for the sampled control law rather than for the continuously updated one.

In summary, we solve the problem of adaptive attitude tracking control by designing a novel controller that uses hybrid systems theory to semi-globally asymptotically stabilize the error set while avoiding the unwinding and chattering phenomena and remaining robust to measurement noise. Furthermore, by employing an RBFNN, our design is adaptable to any state-based disturbance and/or modeling error, and we demonstrate, through extensive simulation results, that, by increasing the number of neurons, we are able to improve the final accuracy, although at the expense of higher computational load. Most importantly, by using an event-triggered approach, our controller does not have Zeno solutions, and it only sends updates if the current value of the derivative of the Lyapunov function surpasses a designed threshold. By employing all these techniques, we are able, for the first time, to derive a hybrid neural network attitude controller that is robust and adaptable to any continuous state-based disturbance while reducing mechanical wear on the actuators. Furthermore, unlike other works in the literature [29], we solve the unavoidable problems of unwinding in continuous controllers and chattering in discrete ones.

Our contributions are threefold:

1. We derive a novel adaptive controller featuring an RBFNN and an event-triggered mechanism using hybrid systems theory. Unlike other works using continuous controllers and neural networks, our controller achieves semi-global results, creating a more robust and efficient controller.

2. By using an RBFNN, we ensure the closed-loop system can adapt to any continuous state-based disturbance or modeling error. Furthermore, our approach ensures the attitude tracking error can be made arbitrarily small through proper parameter tuning.

3. By leveraging a well-posed hybrid systems framework, we design an event-triggered adaptive controller that is robust and renders a precompact neighborhood of the null error set semi-globally asymptotically stable, achieving this without incurring the typical drawbacks that often arise when combining such different techniques.

2. Notation and Problem Formulation

2.1. Notation

The n-dimensional Euclidean space is denoted by $\mathbb{R}^n$. For any real vector $\boldsymbol{a} \in \mathbb{R}^3$ and matrix $\boldsymbol{M} \in \mathbb{R}^{N\times N}$, the Euclidean norm is denoted as $||\boldsymbol{a}||$, the Frobenius norm as $||\boldsymbol{M}||_F$, the matrix 1-norm is denoted by $||\boldsymbol{M}||_1$, the transpose operator is denoted as $(\bullet)^\mathsf{T}$ and the trace operator as $\mathrm{tr}(\boldsymbol{M})$. The diagonal operator $\mathrm{diag}(\boldsymbol{a})$ maps an n-dimensional vector to an n by n diagonal matrix where the i-th component of the vector $\boldsymbol{a}$ is mapped to the i-th row and i-th column of the resulting matrix. $\boldsymbol{I}_n$ denotes the n by n identity matrix. The operator $\tanh(\boldsymbol{a})$ is the element-wise hyperbolic tangent of vector $\boldsymbol{a}$, and we define the operator $\boldsymbol{D}_\epsilon(\boldsymbol{a}) : \mathbb{R}^3 \mapsto \mathbb{R}^{3\times 3}$ as $\boldsymbol{D}_\epsilon(\boldsymbol{a}) \triangleq \mathrm{diag}(\tanh(\boldsymbol{a}/\epsilon))$, where $\epsilon > 0$. The set of nonnegative real numbers is denoted as $\mathbb{R}_{\geq 0} \triangleq [0, +\infty[$, and $\mathbb{B}$ denotes a 3-dimensional closed unit ball. Given $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$, $\boldsymbol{S}(\boldsymbol{a})$ is the skew-symmetric matrix for which $\boldsymbol{S}(\boldsymbol{a})\boldsymbol{b}$ is the cross product between $\boldsymbol{a}$ and $\boldsymbol{b}$; considering $\boldsymbol{a} \triangleq (a_1, a_2, a_3) \in \mathbb{R}^3$, $\boldsymbol{S}(\boldsymbol{a})$ is given by:

$$\boldsymbol{S}(\boldsymbol{a}) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}.$$

A quaternion is denoted by:

$$\overline{\boldsymbol{q}} = \begin{bmatrix} q_0 \\ \boldsymbol{q} \end{bmatrix}, \tag{1}$$

where $q_0 \in \mathbb{R}$ denotes the scalar part of the quaternion and $\boldsymbol{q} \in \mathbb{R}^3$ its vector component. Considering $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$ as the imaginary components of the quaternion, the quaternion space $\mathbb{H}$ is denoted as:

$$\mathbb{H} \triangleq \{\overline{\boldsymbol{q}} \mid \overline{\boldsymbol{q}} = q_0 + q_1\boldsymbol{i} + q_2\boldsymbol{j} + q_3\boldsymbol{k},\ q_0, q_1, q_2, q_3 \in \mathbb{R}\}. \tag{2}$$

Moreover, the quaternion multiplication is defined as:

$$\overline{\boldsymbol{q}} \otimes \overline{\boldsymbol{r}} = q_0\overline{\boldsymbol{r}} + \begin{bmatrix} 0 & -\boldsymbol{q}^\mathsf{T} \\ \boldsymbol{q} & \boldsymbol{S}(\boldsymbol{q}) \end{bmatrix} \overline{\boldsymbol{r}}. \tag{3}$$

The complex conjugate of a quaternion is defined as:

$$\overline{\boldsymbol{q}}^\star = \begin{bmatrix} q_0 \\ -\boldsymbol{q} \end{bmatrix} \tag{4}$$

and its inverse as:

$$\overline{\boldsymbol{q}}^{-1} = \frac{\overline{\boldsymbol{q}}^\star}{||\overline{\boldsymbol{q}}||^2}, \tag{5}$$

where the quaternion norm $||\overline{\boldsymbol{q}}||$ is defined as

$$||\overline{\boldsymbol{q}}|| \triangleq \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}. \tag{6}$$

The space of unit-quaternions is denoted by the set $\mathbb{H}_1$, defined as:

$$\mathbb{H}_1 \triangleq \{\overline{\boldsymbol{q}} \in \mathbb{H} : ||\overline{\boldsymbol{q}}|| = 1\}.$$

Considering the quaternion representation of a vector $\boldsymbol{a} \in \mathbb{R}^3$ as the pure quaternion $\overline{\boldsymbol{p}}(\boldsymbol{a}) = [0\ \ \boldsymbol{a}^\mathsf{T}]^\mathsf{T}$, the rotation of a vector $\boldsymbol{a}$ by $\overline{\boldsymbol{q}} \in \mathbb{H}_1$ is given by:

$$\overline{\boldsymbol{p}}(\boldsymbol{b}) = \overline{\boldsymbol{q}} \otimes \overline{\boldsymbol{p}}(\boldsymbol{a}) \otimes \overline{\boldsymbol{q}}^\star, \tag{7}$$

where $\overline{\boldsymbol{p}}(\boldsymbol{b}) = [0\ \ \boldsymbol{b}^\mathsf{T}]^\mathsf{T}$.

In this work, two frames of reference are used: an inertial frame $\{I\}$ and a body-fixed frame $\{B\}$. The quaternion describing the orientation of the axes of frame $\{B\}$ with respect to frame $\{I\}$ is denoted by $\overline{\boldsymbol{q}} \in \mathbb{H}$.

2.2. Rigid Body Dynamics and Kinematics

In this work, we use the rigid body dynamics and kinematics from [35]. First, as shown in [36, Chapter 4], we define the rigid body dynamics as:

$$\boldsymbol{J}\dot{\boldsymbol{\omega}}_B = -\boldsymbol{S}(\boldsymbol{\omega}_B)\boldsymbol{J}\boldsymbol{\omega}_B + \boldsymbol{f}_d(\boldsymbol{v}_d) + \boldsymbol{\tau}, \tag{8}$$

where $\boldsymbol{J} \in \mathbb{R}^{3\times3}$ is a positive-definite inertia matrix with unknown components, $\boldsymbol{\tau} \in \mathbb{R}^3$ denotes the external torque applied to the rigid body, $\boldsymbol{\omega}_B \in \mathbb{R}^3$ is the angular velocity of the rigid body with respect to $\{I\}$, expressed in $\{B\}$, and $\boldsymbol{f}_d(\boldsymbol{v}_d) \in \mathbb{R}^3$ denotes a disturbance that is a function of the attitude and angular velocity, where $\boldsymbol{v}_d \triangleq [\overline{\boldsymbol{q}}^\mathsf{T}\ \boldsymbol{\omega}_B^\mathsf{T}]^\mathsf{T}$, $\boldsymbol{v}_d \in \mathbb{H}_1 \times \mathbb{R}^3$.

The following assumptions are used in the remainder of the paper.

Assumption 1. *The inertia matrix, $\boldsymbol{J}$, is contained within a compact set, defined as:*

$$\boldsymbol{J} \in \mathcal{X}_J \triangleq \{\boldsymbol{J} \in \mathbb{R}^{3\times3} |\ \ J_{\max}I \succeq \boldsymbol{J} \succeq J_{\min}I\}.$$

The controller of Section 3.2 employs a radial basis function neural network to compensate an unknown state-dependent term, decomposed there as $\boldsymbol{f}(\boldsymbol{v}) = \boldsymbol{W}^\top \boldsymbol{\Psi}(\boldsymbol{v}) + \boldsymbol{\xi}(\boldsymbol{v})$ in (26), where $\boldsymbol{W}$ is the ideal weight matrix and $\boldsymbol{\xi}$ the approximation error. The following two assumptions concern that network. They are stated here so that all standing hypotheses are collected in a single place, and they are used only in Section 4.

Assumption 2. *There exists* $\boldsymbol{\psi} \triangleq [\psi_1 \ \psi_2 \ \psi_3]^\top \in \mathbb{R}^3$ *such that* $\boldsymbol{\xi}$ *belongs to a compact set* $\mathbb{E} \triangleq \{\boldsymbol{\xi} \in \mathbb{R}^3 : |\xi_i| \leq \psi_i, \forall i \in \{1,2,3\}\}$.

Assumption 3. $\boldsymbol{W}$ *and* $\boldsymbol{\omega}_B$ *are each contained within a compact set, such that* $||\boldsymbol{W}||_F \leq W_{max}$ *and* $||\boldsymbol{\omega}_B|| \leq \omega_{max}$.

Notably, the assumption that $\boldsymbol{\omega}_B$ is bounded is a sufficient condition used to streamline the analysis rather than a restriction on the closed-loop behaviour. Its role is to fix a priori the compact set on which the radial basis function network is defined, and this set may be enlarged as needed: given any compact set of initial conditions, ω_{max} can be selected so that the corresponding closed-loop solutions remain within it, in which case the same conclusions hold. This is the sense in which our results are semi-global, namely that convergence is ensured for every initial condition, with ω_{max}, the gains and the network size chosen accordingly.

The attitude kinematics is expressed by:

$$\dot{\overline{\boldsymbol{q}}} = \overline{\boldsymbol{q}} \otimes \frac{1}{2}\overline{\boldsymbol{p}}(\boldsymbol{\omega}_B). \tag{9}$$

The tracking reference, which the system should follow, is defined as:

$$\boldsymbol{r} \triangleq \begin{bmatrix} \overline{\boldsymbol{q}}_r \\ \boldsymbol{\omega}_r \end{bmatrix} \in \mathbb{H}_1 \times \Omega_3, \tag{10}$$

where Ω_3 denotes a compact subset of $\mathbb{R}^3$, and $\overline{\boldsymbol{q}}_r$ describes the desired orientation with respect to $\{I\}$, and $\boldsymbol{\omega}_r$ is the angular velocity of the reference frame with respect to $\{I\}$, expressed in the reference frame.

The attitude tracking error is defined as:

$$\overline{\boldsymbol{q}}_e \triangleq \overline{\boldsymbol{q}}_r^\star \otimes \overline{\boldsymbol{q}} \triangleq \begin{bmatrix} q_{0e} \\ \boldsymbol{q}_e \end{bmatrix} \in \mathbb{H}_1. \tag{11}$$

Afterward, define the angular velocity error, expressed in $\{B\}$, as:

$$\boldsymbol{\omega}_e \triangleq \boldsymbol{\omega}_B - \boldsymbol{\omega}_r|_B \qquad\qquad \boldsymbol{\omega}_e \in \mathbb{R}^3. \tag{12}$$

where $\boldsymbol{\omega}_r|_B$ denotes the desired angular velocity, expressed in $\{B\}$, formulated as:

$$\begin{bmatrix} 0 \\ \boldsymbol{\omega}_r|_B \end{bmatrix} = \overline{\boldsymbol{q}}_e^\star \otimes \overline{\boldsymbol{p}}(\boldsymbol{\omega}_r) \otimes \overline{\boldsymbol{q}}_e. \tag{13}$$

The expression (13) is obtained by rotating the reference angular velocity through the attitude error, so that it is resolved in $\{B\}$. By taking the derivative of (11), we derive the error kinematics as:

$$\dot{\overline{\boldsymbol{q}}}_e = \dot{\overline{\boldsymbol{q}}}_r^\star \otimes \overline{\boldsymbol{q}} + \overline{\boldsymbol{q}}_r^\star \otimes \dot{\overline{\boldsymbol{q}}} = \frac{1}{2}\overline{\boldsymbol{q}}_e \otimes \overline{\boldsymbol{p}}(\boldsymbol{\omega}_e). \tag{14}$$

The last equality follows by substituting $\dot{\overline{\boldsymbol{q}}}_r^\star = -\frac{1}{2}\overline{\boldsymbol{p}}(\boldsymbol{\omega}_r) \otimes \overline{\boldsymbol{q}}_r^\star$ and (9), which give

$$\dot{\overline{\boldsymbol{q}}}_e = -\tfrac{1}{2}\overline{\boldsymbol{p}}(\boldsymbol{\omega}_r) \otimes \overline{\boldsymbol{q}}_r^\star \otimes \overline{\boldsymbol{q}} + \tfrac{1}{2}\overline{\boldsymbol{q}}_r^\star \otimes \overline{\boldsymbol{q}} \otimes \overline{\boldsymbol{p}}(\boldsymbol{\omega}_B) = \tfrac{1}{2}\overline{\boldsymbol{q}}_e \otimes \overline{\boldsymbol{p}}(\boldsymbol{\omega}_B - \boldsymbol{\omega}_r|_B),$$

and then using (12). We stress that both $\overline{\boldsymbol{q}}_e$ and $-\overline{\boldsymbol{q}}_e$ represent the same physical attitude error, so that the null error set consists of the two isolated points $\pm\overline{\boldsymbol{1}}$; it is precisely this double cover that motivates the hybrid variable h, whose role is to select the representation closest to the current state.

Finally, the error dynamics is given by:

$$\begin{aligned} \boldsymbol{J}\dot{\boldsymbol{\omega}}_e =& - \boldsymbol{S}(\boldsymbol{\omega}_r\big|_B + \boldsymbol{\omega}_e)\boldsymbol{J}(\boldsymbol{\omega}_r\big|_B + \boldsymbol{\omega}_e) - \boldsymbol{J}\boldsymbol{S}(\boldsymbol{\omega}_r\big|_B)\boldsymbol{\omega}_e - \boldsymbol{J}\dot{\boldsymbol{\omega}}_r\big|_B + \boldsymbol{\tau} + \boldsymbol{f}_d(\boldsymbol{v}_d) \\ =& \boldsymbol{F}_{\omega_e}(\boldsymbol{\omega}_r\big|_B, \boldsymbol{\omega}_e, \boldsymbol{v}_d, \dot{\boldsymbol{\omega}}_r\big|_B), \end{aligned} \tag{15}$$

where $\boldsymbol{F}_{\omega_e} : \Omega_3 \times \mathbb{R}^3 \times \mathbb{H}_1 \times \mathbb{R}^3 \times \mathbb{R}^3 \mapsto \mathbb{R}^3$ denotes the dynamics map.

2.3. Hybrid Systems

The controller is derived in the hybrid systems framework of [37, 38], where one defines a flow map $\boldsymbol{F}$, which governs the continuous time evolution of $\boldsymbol{x}$ on the flow set $\mathcal{C}$. Similarly, a jump map $\boldsymbol{G}$ governs the discrete time evolution of $\boldsymbol{x}$ on the jump set $\mathcal{D}$. Both these maps will be defined in the final stage of the controller design, where it is possible to take into account all the closed-loop dynamics. The overall hybrid system H is written as:

$$H : \begin{cases} \dot{\boldsymbol{x}} \in \boldsymbol{F}(\boldsymbol{x}) & \boldsymbol{x} \in \mathcal{C}, \\ \boldsymbol{x}^{+} \in \boldsymbol{G}(\boldsymbol{x}) & \boldsymbol{x} \in \mathcal{D}. \end{cases} \tag{16}$$

In this type of model, $\mathcal{C}$ is the set in which $\boldsymbol{x}$ has a continuous behavior over time $t \in \mathbb{R}_{\geq 0}$, and $\mathcal{D}$ denotes the set of points in which $\boldsymbol{x}$ has discrete jumps at time $j \in \mathbb{N} \triangleq \{0, 1, 2, ...\}$. A solution $\boldsymbol{x}$ to a hybrid system H is defined on a hybrid time domain $Q \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$, where $\forall (T, J) \in Q$, the set $Q \cap ([0, T] \times \{0, 1, ..., J\})$ can be written in the form $\bigcup_{j=0}^{J}([t_j, t_{j+1}] \times \{j\})$ for some finite time sequence $0 = t_0 \leq t_1 ... \leq t_J$. A hybrid trajectory is defined by the pair $(\boldsymbol{x}, \operatorname{dom} \boldsymbol{x})$, where $\operatorname{dom} \boldsymbol{x}$ denotes a hybrid time domain on which $\boldsymbol{x}$ is defined.

In hybrid systems, maximal solutions are not always complete, which requires a more careful approach to analyze properties such as attractiveness and stability. A pre-attractive set $\mathcal{A}$ is a set to which both every complete solution converges and every maximal noncomplete solution has a bounded distance. If a set $\mathcal{A}$ is both stable and pre-attractive, then this set is pre-asymptotically stable. Unlike asymptotic stability, the concept of pre-asymptotic stability is a more relaxed form of stability which accepts noncomplete maximal solutions.

Another common property is $\mathcal{KL}$ asymptotic stability, which requires defining beforehand what a $\mathcal{KL}$ function and a proper indicator are. On the one hand, a $\mathcal{KL}$ function $\beta(x, t) : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ is a function which is nondecreasing in its first argument and nonincreasing in its second argument, and furthermore, $\lim_{x \to 0} \beta(x, t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$ and $\lim_{t \to \infty} \beta(x, t) = 0$ for all $x \in \mathbb{R}_{\geq 0}$. On the other hand, considering that the basin of attraction is a set $\mathcal{U}$, a proper indicator $w(x) : \mathbb{R}^n \mapsto \mathbb{R}_{\geq 0}$ is a function which grows unbounded as x approaches either ∞ or the boundary of $\mathcal{U}$, and has a value of 0 if and only if $x \in \mathcal{A}$. Thus, as defined in [38, Definition 3.9], a set $\mathcal{A}$ is $\mathcal{KL}$ asymptotically stable on an open set $\mathcal{U}$ if for every proper indicator w of $\mathcal{A}$ on $\mathcal{U}$ there exists a class-$\mathcal{KL}$ function β such that each solution x satisfies:

$$w(x(t, j)) \leq \beta(w(0, 0), t + j). \tag{17}$$

As stated in [38, Theorem 3.22], if we can prove that our hybrid system follows the hybrid basic conditions [38, Definition 2.20] and the set $\mathcal{A}$ is asymptotically stable, then it is also $\mathcal{KL}$ asymptotically stable. This theorem is very important as $\mathcal{KL}$ asymptotic stability is equivalent to uniform asymptotic stability [38, Definition 3.7], as shown in [37, Theorem 3.20].

Additionally, [38, Definition 3.16] defines semi-global practical robust $\mathcal{KL}$ pre-asymptotic stability, a concept which guarantees that any solution to the closed-loop under a disturbance which follows [38, Assumption 3.25] will converge to within a bounded distance of the set $\mathcal{A}$.

The flow and jump sets of the closed-loop system are built from two mechanisms, each associated with a threshold. The first, developed in Section 3.2, is the hysteresis mechanism that governs the hybrid variable h: a jump occurs when the Lyapunov function evaluated at $-h$ is lower than its current value by at least a margin $\sigma > 0$, which yields the pair $(\mathcal{C}_1, \mathcal{D}_1)$ in (37). The second, developed in Section 3.3, is the event-triggering mechanism: a jump occurs, and the control law is resampled, when the derivative of the Lyapunov function along the sampled control law exceeds a threshold $\mu(\boldsymbol{x})$, which yields the pair $(\mathcal{C}_2, \mathcal{D}_2)$ in (38) and, in computable form, in (41). The closed-loop flow and jump sets are then $\mathcal{C} = \mathcal{C}_1 \cap \mathcal{C}_2$ and $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$. Both thresholds are defined only after the Lyapunov function is available, which is why their construction is deferred to Section 3.

3. Controller Design

In this section, we start by briefly describing the design reported in [35], and we expand on it by adding the RBFNN and the event-triggered boundary in the hybrid system.

3.1. Backstepping Design

In order to solve the tracking control problem, the first step is to carefully define the closed-loop state. In this work, the closed-loop state is written as $\boldsymbol{x} \triangleq (\overline{\boldsymbol{q}}_e\ \boldsymbol{\omega}_e\ \boldsymbol{r}\ \widetilde{\boldsymbol{W}}\ \widetilde{\boldsymbol{\psi}}\ h\ \boldsymbol{z}_s\ \boldsymbol{r}_s)$, $\boldsymbol{x} \in \mathcal{X} \triangleq \mathbb{H}_1 \times \mathbb{R}^3 \times \mathbb{H}_1 \times \Omega_3 \times \mathbb{R}^{N\times 3} \times \mathbb{R}^3 \times \{-1, 1\} \times \mathcal{X}_z \times \mathbb{H}_1 \times \Omega_3$, where $\widetilde{\boldsymbol{W}} \in \mathbb{R}^{N\times 3}$ is the neural network weights estimation error, $\widetilde{\boldsymbol{\psi}} \in \mathbb{R}^3$ is the estimation error of the neural network error bound, $\boldsymbol{r}$ is the reference defined in (10) and h is the hybrid controller state, and $\boldsymbol{z}_s$ and $\boldsymbol{r}_s$ are the sample-and-hold variables introduced in Section 3.3. The estimation states will be further contextualized in Section 3.2, where the corresponding estimation errors will also be formally defined.

In order to properly design the hybrid controller, let us start by considering a flow set $\mathcal{C}_1$ and a jump set $\mathcal{D}_1$, both to be designed after defining the Lyapunov function, and a hybrid controller state $h \in \{-1, 1\}$, whose role is to direct the controller and avoid unwinding, designed in [3] and given by:

$$\begin{cases} \dot{h} &= 0 \qquad \boldsymbol{x} \in \mathcal{C}_1, \\ h^+ &= -h \qquad \boldsymbol{x} \in \mathcal{D}_1. \end{cases} \tag{18}$$

Similarly to [3], in order to use Lyapunov's stability theory for hybrid systems, it is assumed that $\boldsymbol{\omega}_r$ is contained within a compact subset of $\mathbb{R}^3$, which is true according to (10), and that $\dot{\boldsymbol{\omega}}_r$ is available and is contained within $M\mathbb{B}$. In this context, $M > 0$ restricts every component of $\boldsymbol{\omega}_r$ to be Lipschitz continuous with Lipschitz constant M.

We start by considering the Lyapunov function:

$$V_1(\boldsymbol{x}) \triangleq 2k_1(1 - hq_{0e}) \qquad \forall \boldsymbol{x} \in \mathcal{X} \tag{19}$$

whose derivative, according to (3) and (14), is given by:

$$\dot{V}_1 = k_1 h \boldsymbol{q}_e^\mathsf{T} \boldsymbol{\omega}_e \tag{20}$$

where k_1 denotes a positive constant.

Now, we consider the virtual control law $-k_a h \boldsymbol{q}_e$, where k_a is a positive gain, and define a new Lyapunov function as:

$$V_2(\boldsymbol{x}) \triangleq V_1 + \frac{1}{2}(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T} \boldsymbol{J} (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e) \qquad \forall \boldsymbol{x} \in \mathcal{X}, \tag{21}$$

with derivative given by:

$$\dot{V}_2 = (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T}(k_1 h \boldsymbol{q}_e + \boldsymbol{J}(\dot{\boldsymbol{\omega}}_e + k_a h \dot{\boldsymbol{q}}_e)) - k_a k_1 \boldsymbol{q}_e^\mathsf{T} \boldsymbol{q}_e. \tag{22}$$

To stabilize the closed-loop system, let us consider a control law which cancels out the dynamics terms from (15) by feedforwarding the reference-based terms, and adds a component $-k_2(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)$, where k_2 is a positive gain. Furthermore, we consider that there is a modeling error caused by an incorrect value of the moment of inertia, $\widehat{\boldsymbol{J}} \in \mathbb{R}^{3\times 3}$, held by the controller. A preliminary control law is thus set as:

$$\begin{aligned} \boldsymbol{\tau}_1(\boldsymbol{x}) \triangleq & - k_1 h \boldsymbol{q}_e - k_2(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e) - \widehat{\boldsymbol{J}} k_a h \dot{\boldsymbol{q}}_e + \boldsymbol{S}(\boldsymbol{\omega}_r\big|_B + \boldsymbol{\omega}_e)\widehat{\boldsymbol{J}}(\boldsymbol{\omega}_r\big|_B + \boldsymbol{\omega}_e) \\ & + \widehat{\boldsymbol{J}}(\boldsymbol{S}(\boldsymbol{\omega}_r\big|_B)\boldsymbol{\omega}_e) + \widehat{\boldsymbol{J}} \dot{\boldsymbol{\omega}}_r\big|_B. \end{aligned} \tag{23}$$

Using (23) and the error dynamics (15), and by considering the modeling error $\widetilde{\boldsymbol{J}} \triangleq \boldsymbol{J} - \widehat{\boldsymbol{J}}$, the expression for $\dot{V}_2$ in (22) becomes:

$$\dot{V}_2 = -\rho + (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T} \boldsymbol{f}(\boldsymbol{v}), \tag{24}$$

where ρ is a positive function given by $\rho(\boldsymbol{x}) \triangleq k_a k_1 \boldsymbol{q}_e^\mathsf{T} \boldsymbol{q}_e + k_2(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T}(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)$ and $\boldsymbol{f}(\boldsymbol{v}) \in \mathbb{R}^3$ is given by:

$$\boldsymbol{f}(\boldsymbol{v}) \triangleq \boldsymbol{f}_d(\boldsymbol{v_d}) - \boldsymbol{S}(\boldsymbol{\omega}_r\big|_B + \boldsymbol{\omega_e})\widetilde{\boldsymbol{J}}(\boldsymbol{\omega}_r\big|_B + \boldsymbol{\omega_e}) - \widetilde{\boldsymbol{J}}(\boldsymbol{S}(\boldsymbol{\omega}_r\big|_B)\boldsymbol{\omega_e}) - \widetilde{\boldsymbol{J}}\dot{\boldsymbol{\omega}}_r\big|_B + \widetilde{\boldsymbol{J}} k_a h \dot{\boldsymbol{q}}_e, \tag{25}$$

where $\boldsymbol{v} \triangleq [\overline{\boldsymbol{q}}^\mathsf{T}\ \boldsymbol{\omega}_B^\mathsf{T}\ \boldsymbol{r}^\mathsf{T}, \dot{\boldsymbol{\omega}}_r]^\mathsf{T}$, $\boldsymbol{v} \in \mathbb{H}_1 \times \mathbb{R}^3 \times \mathbb{H}_1 \times \Omega_3 \times M\mathbb{B}$.

3.2. Adaptive Neural Network

Our objective now is to design an RBFNN to compensate for $\boldsymbol{f}(\boldsymbol{v})$, introduced in (25). Thus, we start by writing $\boldsymbol{f}(\boldsymbol{v})$ as:

$$\boldsymbol{f}(\boldsymbol{v}) = \boldsymbol{W}^\mathsf{T}\boldsymbol{\Psi}(\boldsymbol{v}) + \boldsymbol{\xi}(\boldsymbol{v}), \tag{26}$$

where $\boldsymbol{W}$ is an $\mathbb{R}^{N\times 3}$ weight matrix, $\boldsymbol{\xi}(\boldsymbol{v}) \in \mathbb{R}^3$ is the approximation error, and $\boldsymbol{\Psi}(\boldsymbol{v})$ is a radial basis function defined as $\boldsymbol{\Psi}(\boldsymbol{v}) \triangleq [g_1(\boldsymbol{v}) \ \ldots \ g_n(\boldsymbol{v})]$, where $g_i(\boldsymbol{v}) \triangleq \exp(-||\boldsymbol{v} - \mu_i||^2/\sigma_{\boldsymbol{W}}^2)$, and μ_i denotes the centroid of the function g_i, and $\sigma_{\boldsymbol{W}}$ denotes the standard deviation, and N is the number of neurons.

While it is not possible to guarantee that an arbitrary function will be estimated with an RBFNN, the RBFNN universal approximation theorem [39, Theorem 2] states that it is possible to create a neural network with one hidden layer that is able to approximate a continuous function $\boldsymbol{f}(\boldsymbol{v})$ with some error $\boldsymbol{\xi}(\boldsymbol{v})$.

Assumptions 2 and 3, stated in Section 2, concern the network just introduced. We note that $||\boldsymbol{\Psi}(\boldsymbol{v})|| \leq \sqrt{N}$, since each g_i takes values in $(0, 1]$.

Now, we consider the estimates $\widehat{\boldsymbol{W}} \in \mathbb{R}^{N\times 3}$ and $\widehat{\boldsymbol{\psi}} \in \mathbb{R}^3$. Based on (23), and according to (26), the updated proposed control law follows as:

$$\boldsymbol{\tau}(\boldsymbol{x}) \triangleq \boldsymbol{\tau}_1(\boldsymbol{x}) - \widehat{\boldsymbol{W}}^\mathsf{T}\boldsymbol{\Psi}(\boldsymbol{v}) - \boldsymbol{D}_\epsilon(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)\widehat{\boldsymbol{\psi}}. \tag{27}$$

Let us now consider the estimation errors $\widetilde{\boldsymbol{W}} \in \mathbb{R}^{N\times 3}$ and $\widetilde{\boldsymbol{\psi}} \in \mathbb{R}^3$, defined as $\widetilde{\boldsymbol{W}} \triangleq \boldsymbol{W} - \widehat{\boldsymbol{W}}$ and $\widetilde{\boldsymbol{\psi}} \triangleq \boldsymbol{\psi} - \widehat{\boldsymbol{\psi}}$, respectively. Considering a positive definite matrix gain $\boldsymbol{K}_{\boldsymbol{W}} \in \mathbb{R}^{N\times N}$ and the gain $k_{\boldsymbol{\psi}} \in \mathbb{R}_{>0}$, and using (21), we establish a third and final Lyapunov candidate function given by:

$$V(\boldsymbol{x}) \triangleq V_2 + \frac{1}{2}\operatorname{tr}\left(\widetilde{\boldsymbol{W}}^\mathsf{T}\boldsymbol{K}_{\boldsymbol{W}}^{-1}\widetilde{\boldsymbol{W}}\right) + \frac{1}{2k_{\boldsymbol{\psi}}}\widetilde{\boldsymbol{\psi}}^\mathsf{T}\widetilde{\boldsymbol{\psi}} \quad \forall \boldsymbol{x} \in \mathcal{X}. \tag{28}$$

The derivative of (28) can be written as:

$$\begin{aligned}\dot{V} = &- \rho - (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T}[\boldsymbol{D}_\epsilon(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)\widehat{\boldsymbol{\psi}} \\ &- \widetilde{\boldsymbol{W}}^\mathsf{T}\boldsymbol{\Psi}(\boldsymbol{v}) - \boldsymbol{\xi}(\boldsymbol{v})] - \operatorname{tr}\left(\widetilde{\boldsymbol{W}}^\mathsf{T}\boldsymbol{K}_{\boldsymbol{W}}^{-1}\dot{\widehat{\boldsymbol{W}}}\right) - \frac{1}{k_{\boldsymbol{\psi}}}\widetilde{\boldsymbol{\psi}}^\mathsf{T}\dot{\widehat{\boldsymbol{\psi}}}.\end{aligned} \tag{29}$$

To cancel some of the terms in the Lyapunov function, we propose the following adaptation laws:

$$\dot{\widehat{\boldsymbol{W}}} \triangleq \boldsymbol{K}_{\boldsymbol{W}}\boldsymbol{\Psi}(\boldsymbol{v})(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T} - \gamma_{\boldsymbol{W}}\boldsymbol{K}_{\boldsymbol{W}}\widehat{\boldsymbol{W}}, \tag{30a}$$

$$\dot{\widehat{\boldsymbol{\psi}}} \triangleq k_{\boldsymbol{\psi}}\boldsymbol{D}_\epsilon(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e) - \gamma_{\boldsymbol{\psi}} k_{\boldsymbol{\psi}}\widehat{\boldsymbol{\psi}}, \tag{30b}$$

where $\gamma_{\boldsymbol{W}} > 0$ and $\gamma_{\boldsymbol{\psi}} > 0$ are leakage terms whose function is to ensure that the neural network's weights do not overfit the disturbances. In the absence of these terms, changes in disturbances may lead to slower adaptation.

Remark 1. *The analysis that follows does not exploit the radial structure of* $\boldsymbol{\Psi}$. *It requires only that the unknown parameters admit an update law derived from the Lyapunov analysis, as in* (30a) *and* (30b), *and that* $\boldsymbol{\Psi}$ *be continuous and bounded on the operating set, so that Lemmas 4.1 and 4.2 remain valid; other approximators with these properties may be used without affecting the robustness or the Zeno-free guarantees. The radial basis function network is adopted here for its low complexity: the regressor is evaluated on a compact set, so the centroids are fixed a priori, each evaluation requires* $O(N)$ *operations, and the bound* $||\boldsymbol{\Psi}(\boldsymbol{v})|| \leq \sqrt{N}$ *follows by construction. This is well suited to a design whose purpose is to reduce the computational and communication burden on the platform.*

Using (30), the expression in (29) simplifies down to:

$$\begin{aligned}\dot{V} = &- (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T}(\boldsymbol{D}_\epsilon(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)\boldsymbol{\psi} + \boldsymbol{\xi}(\boldsymbol{v})) \\ &- \rho + \gamma_{\boldsymbol{W}}\operatorname{tr}\left(\widetilde{\boldsymbol{W}}^\mathsf{T}\widehat{\boldsymbol{W}}\right) + \gamma_{\boldsymbol{\psi}}\widetilde{\boldsymbol{\psi}}^\mathsf{T}\widehat{\boldsymbol{\psi}}.\end{aligned} \tag{31}$$

Now, we use the fact from [40, Lemma 1] that for any $\eta \in \mathbb{R}$, considering a $k_D = 0.2785$, the inequality $0 \leq |\eta| - \eta \tanh(\eta/\epsilon) \leq k_D \epsilon$ holds. Thus, using Assumption 2, we can write:

$$\dot{V} \leq -\rho + k_D \epsilon \mathbf{1}^{\mathsf{T}} \boldsymbol{\psi} + \gamma_{\boldsymbol{W}} \operatorname{tr}\left(\widetilde{\boldsymbol{W}}^{\mathsf{T}} \widehat{\boldsymbol{W}}\right) + \gamma_{\psi} \widetilde{\boldsymbol{\psi}}^{\mathsf{T}} \widehat{\boldsymbol{\psi}}, \tag{32}$$

where $\mathbf{1} \in \mathbb{R}^3$ denotes a column vector of ones.

Now, we use Cauchy's inequality and Young's product inequality to derive $\operatorname{tr}(\boldsymbol{A}^{\mathsf{T}} \boldsymbol{B}) \leq ||\boldsymbol{A}||_F ||\boldsymbol{B}||_F$ and $ba - a^2 \leq \frac{1}{2} b^2 - \frac{1}{2} a^2$, and note that:

$$\begin{aligned} \operatorname{tr}\left(\widetilde{\boldsymbol{W}}^{\mathsf{T}} \widehat{\boldsymbol{W}}\right) &\leq \operatorname{tr}\left(\widetilde{\boldsymbol{W}}^{\mathsf{T}} \boldsymbol{W}\right) - \operatorname{tr}\left(\widetilde{\boldsymbol{W}}^{\mathsf{T}} \widetilde{\boldsymbol{W}}\right) \\ &\leq ||\widetilde{\boldsymbol{W}}||_F ||\boldsymbol{W}||_F - ||\widetilde{\boldsymbol{W}}||_F^2 \\ &\leq \frac{1}{2} ||\boldsymbol{W}||_F^2 - \frac{1}{2} ||\widetilde{\boldsymbol{W}}||_F^2, \end{aligned} \tag{33}$$

Now, using the fact that:

$$\begin{aligned} \widetilde{\boldsymbol{\psi}}^{\mathsf{T}} \widehat{\boldsymbol{\psi}} &\leq ||\widetilde{\boldsymbol{\psi}}|| ||\boldsymbol{\psi}|| - ||\widetilde{\boldsymbol{\psi}}||^2 \\ &\leq \frac{1}{2} ||\boldsymbol{\psi}||^2 - \frac{1}{2} ||\tilde{\boldsymbol{\psi}}||^2, \end{aligned} \tag{34}$$

we can write:

$$\dot{V} \leq -\rho_2 + Z, \tag{35}$$

where both ρ_2 and Z are positive and given by: $\rho_2 \triangleq \rho + \frac{\gamma_{\boldsymbol{W}}}{2} ||\widetilde{\boldsymbol{W}}||_F^2 + \frac{\gamma_\psi}{2} ||\widetilde{\boldsymbol{\psi}}||^2$ and $Z \triangleq k_D \epsilon \mathbf{1}^{\mathsf{T}} \boldsymbol{\psi} + \frac{\gamma_\psi}{2} ||\boldsymbol{\psi}||^2 + \frac{\gamma_{\boldsymbol{W}}}{2} ||\boldsymbol{W}||_F^2$.

Now, we establish the boundary between a flow set $\mathcal{C}_1$ and a jump set $\mathcal{D}_1$, used to define h as in (18). In summary, our objective is to define a boundary which ensures that h directs the controller towards the nearest equilibrium set, but avoids chattering. Thus, we consider $0 < \sigma < 4k_1$; we define $\boldsymbol{z}$ as the Lyapunov function states, $\boldsymbol{z} \triangleq (\overline{\boldsymbol{q}}_e \ \boldsymbol{\omega}_e \ \widehat{\boldsymbol{W}} \ \widehat{\boldsymbol{\psi}})$, $\boldsymbol{z} \in \mathcal{X}_z \triangleq \mathbb{H}_1 \times \mathbb{R}^3 \times \mathbb{R}^{N \times 3} \times \mathbb{R}^3$, and define the boundary using $\mathcal{D}_1 \triangleq \{\boldsymbol{x} \in \mathcal{X} : V(\boldsymbol{z}, \boldsymbol{r}, h) - V(\boldsymbol{z}, \boldsymbol{r}, -h) \geq \sigma\}$. The notation $V(\boldsymbol{z}, \boldsymbol{r}, h)$ is used here instead of $V(\boldsymbol{x})$ to differentiate $V(\boldsymbol{z}, \boldsymbol{r}, -h)$ and $V(\boldsymbol{z}, \boldsymbol{r}, h)$.

This choice of $\mathcal{D}_1$ changes the value of h whenever the value of the Lyapunov function using $-h$, $V(\boldsymbol{z}, \boldsymbol{r}, -h)$, is lower than the value before the jump, $V(\boldsymbol{z}, \boldsymbol{r}, h)$. The jump in the Lyapunov function can be computed using (28), yielding:

$$V(\boldsymbol{z}, \boldsymbol{r}, h) - V(\boldsymbol{z}, \boldsymbol{r}, -h) = 2hk_a \boldsymbol{q}_e^{\mathsf{T}} \boldsymbol{J} \boldsymbol{\omega}_e - 4hk_1 q_{0e}. \tag{36}$$

Finally, since we do not know the value of $\boldsymbol{J}$, we use Assumption 1 and, as done in [35], we define:

$$\begin{aligned} \mathcal{C}_1 &\triangleq \{\boldsymbol{x} \in \mathcal{X} : \min_{\boldsymbol{X} \in \mathcal{X}_J} (2hk_a \boldsymbol{q}_e^{\mathsf{T}} \boldsymbol{X} \boldsymbol{\omega}_e - 4hk_1 q_{0e}) \leq \sigma\}, \\ \mathcal{D}_1 &\triangleq \{\boldsymbol{x} \in \mathcal{X} : \min_{\boldsymbol{X} \in \mathcal{X}_J} (2hk_a \boldsymbol{q}_e^{\mathsf{T}} \boldsymbol{X} \boldsymbol{\omega}_e - 4hk_1 q_{0e}) \geq \sigma\}. \end{aligned} \tag{37}$$

The jump condition is straightforward to verify in practice, as it reduces to an SDP optimization problem with an affine cost function. Thus, it is fairly trivial to solve in real-time.

3.3. Event-Triggered Control

A key challenge in this work lies in designing an event-triggered adaptive controller with neural networks, where the Lyapunov derivative is not directly computable due to the NN structure. To address this, we construct a triggering condition based on measurable quantities and carefully define the hybrid flow and jump sets to ensure Zeno-free behavior. The use of hysteresis switching, following the approach in [3], prevents chattering and unwinding. This integrated design allows us to rigorously prove semi-global asymptotic

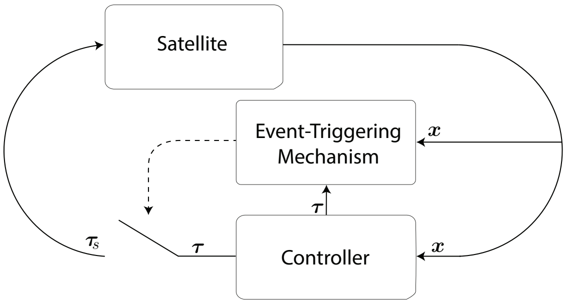

Figure 1: Example of an architecture with event-triggered sampling of the actuation signal.

stability to a small error neighborhood, which, to the best of our knowledge, has not been achieved in previous related works.

This work uses a Lyapunov-based approach similar to the algorithm presented in [6]. We start by defining the sample-and-hold hybrid variables as $\boldsymbol{z}_s \triangleq (\overline{\boldsymbol{q}}_{es}\ \boldsymbol{\omega}_{es}\ \widehat{\boldsymbol{W}}_s\ \widehat{\boldsymbol{\psi}}_s)$, where the subscript s denotes the sampled variable and $\boldsymbol{z}_s \in \mathcal{X}_z$, together with the sampled reference $\boldsymbol{r}_s \in \mathbb{H}_1 \times \Omega_3$. Holding the reference as well is what makes the actuation piecewise constant between jumps, as intended by the architecture of Fig. 1.

In our work, the main objective underlying the event-triggered mechanism is to ease the communication between the controller and the actuator. We employ the architecture shown in Fig. 1, where $\boldsymbol{\tau}_s$ denotes the sampled control law, *i.e.* $\boldsymbol{\tau}(\boldsymbol{z}_s, \boldsymbol{r}_s, h)$. Since we want to reduce the communication between the actuators and the controller, it is expected that the error might converge slower due to the usage of sampled signals. However, we want to ensure that whenever the performance dips below a predefined boundary, the control input is updated. Thus, we define the value of the derivative of the Lyapunov function using the latest sampled control input as $\nabla_{\boldsymbol{z}} V(\boldsymbol{x}) \boldsymbol{f}_{\boldsymbol{z}}(\boldsymbol{x}, \boldsymbol{\tau}_s)$, where $\nabla_{\boldsymbol{z}} V(\boldsymbol{x})$ denotes the gradient of the Lyapunov function with respect to $\boldsymbol{z}$ and $\boldsymbol{f}_{\boldsymbol{z}}$ denotes the flow of the state $\boldsymbol{z}$, given by $\boldsymbol{f}_{\boldsymbol{z}} \triangleq (\dot{\overline{\boldsymbol{q}}}_e\ \dot{\boldsymbol{\omega}}_e\ \dot{\widehat{\boldsymbol{W}}}\ \dot{\widehat{\boldsymbol{\psi}}})$.

Now, we bound $\nabla_{\boldsymbol{z}} V(\boldsymbol{x}) \boldsymbol{f}_{\boldsymbol{z}}(\boldsymbol{x}, \boldsymbol{\tau}_s)$ by a function $\mu(\boldsymbol{x})$, and whenever the performance dips below $\mu(\boldsymbol{x})$, the control input is sampled. In hybrid systems, this method can be implemented by defining a jump set $\mathcal{D}_2$, which dictates when a new sampling is required. Thus, we define $\mathcal{D}_2$ and the flow set $\mathcal{C}_2$ as:

$$
\begin{aligned}
\mathcal{C}_2 &\triangleq \{\boldsymbol{x} \in \mathcal{X} : \nabla_{\boldsymbol{z}} V(\boldsymbol{x}) \boldsymbol{f}_{\boldsymbol{z}}(\boldsymbol{x}, \boldsymbol{\tau}_s) \leq \mu(\boldsymbol{x})\}, \\
\mathcal{D}_2 &\triangleq \{\boldsymbol{x} \in \mathcal{X} : \nabla_{\boldsymbol{z}} V(\boldsymbol{x}) \boldsymbol{f}_{\boldsymbol{z}}(\boldsymbol{x}, \boldsymbol{\tau}_s) \geq \mu(\boldsymbol{x})\},
\end{aligned} \tag{38}
$$

To design $\mu(\boldsymbol{x})$, we start by defining the error between the computed control input and the latest sample as $\boldsymbol{\tau}_e \triangleq \boldsymbol{\tau} - \boldsymbol{\tau}_s$, followed by rewriting the boundary of $\mathcal{C}_2$ and $\mathcal{D}_2$ as:

$$
\nabla_{\boldsymbol{z}} V(\boldsymbol{x}) \boldsymbol{f}_{\boldsymbol{z}}(\boldsymbol{x}, \boldsymbol{\tau}_s) = \dot{V} - (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^{\mathsf{T}} \boldsymbol{\tau}_e \tag{39}
$$

Unlike the example in [6], we cannot compute the value of the Lyapunov function or its derivative due to a lack of knowledge of the current adaptive error states. However, using (31) to define $\boldsymbol{\mu}$, we can avoid this problem. Furthermore, to avoid Zeno solutions, we want to guarantee each jump from $\mathcal{D}_2$ will end in $\mathcal{C}_2$. Thus, we add a constant $\upsilon > 0$, which ensures that a neighborhood of the set $\mathcal{B} \triangleq \{\boldsymbol{x} \in \mathcal{X} : \overline{\boldsymbol{q}}_e = h\overline{\mathbf{1}}, \boldsymbol{\omega}_e = \mathbf{0}, \widetilde{\boldsymbol{W}} = \mathbf{0}, \widetilde{\boldsymbol{\psi}} = \mathbf{0}\}$ is inside the flow set, where $\overline{\mathbf{1}}$ is the identity quaternion. In Lemma 4.3, it will be proven that using this approach there are no Zeno solutions, as the time between jumps cannot tend to zero, and Proposition 4.6 extends this conclusion to the case in which the triggering condition is evaluated on noisy measurements. Finally, we consider $0 < \gamma < 1$ and define:

$$
\begin{aligned}
\mu(\boldsymbol{x}) \triangleq &- (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^{\mathsf{T}} (\boldsymbol{D}_\epsilon (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e) \boldsymbol{\psi} + \boldsymbol{\xi}(\boldsymbol{v})) \\
&- \gamma \rho + \gamma_{\boldsymbol{W}} \operatorname{tr}\left(\widetilde{\boldsymbol{W}}^{\mathsf{T}} \widehat{\boldsymbol{W}}\right) + \gamma_{\boldsymbol{\psi}} \widetilde{\boldsymbol{\psi}}^{\mathsf{T}} \widehat{\boldsymbol{\psi}} + \upsilon.
\end{aligned} \tag{40}
$$

Next, we develop (39) as:

$$\begin{aligned}\mathcal{C}_2 &\triangleq \{\boldsymbol{x} \in \mathcal{X} : -(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T} \boldsymbol{\tau}_e \leq (1-\gamma)\rho + \upsilon\},\\ \mathcal{D}_2 &\triangleq \{\boldsymbol{x} \in \mathcal{X} : -(\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T} \boldsymbol{\tau}_e \geq (1-\gamma)\rho + \upsilon\}.\end{aligned} \tag{41}$$

Notably, the bound between $\mathcal{C}_2$ and $\mathcal{D}_2$ is now computable. Recall from (24) that:

$$\rho(\boldsymbol{x}) \triangleq k_a k_1 \boldsymbol{q}_e^\mathsf{T} \boldsymbol{q}_e + k_2 (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e)^\mathsf{T} (\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e),$$

which depends only on the tracking error $\boldsymbol{q}_e$, on the angular velocity error $\boldsymbol{\omega}_e$, on the hybrid state h and on the design gains. It is therefore directly computable from the measured signals. Consequently, the triggering condition (41) is evaluated from available signals only, without requiring knowledge of $\boldsymbol{W}$, $\boldsymbol{\psi}$ or of the disturbance $\boldsymbol{f}(\boldsymbol{v})$. Next, we define the flow set $\mathcal{C} \triangleq \mathcal{C}_1 \cap \mathcal{C}_2$ and the jump set $\mathcal{D} \triangleq \mathcal{D}_1 \cup \mathcal{D}_2$. Furthermore, writing $m(h) \triangleq \min_{\boldsymbol{X} \in \mathcal{X}_J} (2hk_a \boldsymbol{q}_e^\mathsf{T} \boldsymbol{X} \boldsymbol{\omega}_e - 4hk_1 q_{0e})$ for the quantity compared against σ in (37), the jump map of h is defined as

$$G_h(\boldsymbol{x}) \triangleq -h\,\overline{\mathrm{sgn}}\big(m(h)\big), \qquad \overline{\mathrm{sgn}}(r) \triangleq \begin{cases} \mathrm{sgn}(r) & r \neq 0, \\ \{-1, 1\} & r = 0, \end{cases} \tag{42}$$

so that h is flipped exactly when $m(h) > 0$. Since $m(h) \geq \sigma > 0$ on $\mathcal{D}_1$, the flip is forced there. A consequence of this definition is that h may also flip during a resampling triggered by $\mathcal{D}_2$ when $0 < m(h) < \sigma$, which is harmless since $m(h) > 0$ implies that the jump does not increase the Lyapunov function.

Finally, we define the maps $\boldsymbol{G}$: $\mathcal{X} \rightrightarrows \mathcal{X}$ and $\boldsymbol{F}$: $\mathcal{X} \rightrightarrows \mathcal{X}_f$, where $\mathcal{X}_f \triangleq \mathbb{H} \times \mathbb{R}^3 \times \mathbb{H} \times M\mathbb{B} \times \mathbb{R}^{N\times 3} \times \mathbb{R}^3 \times \{0\} \times \mathcal{X}_z \times \mathbb{H} \times \Omega_3$. Thus, following the structure used in (16), we define the overall closed-loop system as:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{q}}_e \\ \dot{\boldsymbol{\omega}}_e \\ \dot{\boldsymbol{q}}_r \\ \dot{\boldsymbol{\omega}}_r \\ \dot{\widetilde{\boldsymbol{W}}} \\ \dot{\tilde{\boldsymbol{\psi}}} \\ \dot{h} \\ \dot{\boldsymbol{z}}_s \\ \dot{\boldsymbol{r}}_s \end{bmatrix} \in \begin{bmatrix} \frac{1}{2}\overline{\boldsymbol{q}}_e \otimes \overline{\boldsymbol{\omega}}_e \\ \boldsymbol{J}^{-1}\boldsymbol{F}_{\omega_e}(\boldsymbol{x}, \boldsymbol{\tau}_s) \\ \frac{1}{2}\overline{\boldsymbol{q}}_r \otimes \overline{\boldsymbol{\omega}}_r \\ M\mathbb{B} \\ -\dot{\hat{\boldsymbol{W}}} \\ -\dot{\hat{\boldsymbol{\psi}}} \\ 0 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} & \boldsymbol{x} \in \mathcal{C}, \\ \begin{aligned} \boldsymbol{z}^+ &= \boldsymbol{z} \\ \boldsymbol{r}^+ &= \boldsymbol{r} \\ h^+ &\in G_h(\boldsymbol{x}) \\ \boldsymbol{z}_s^+ &= \boldsymbol{z} \\ \boldsymbol{r}_s^+ &= \boldsymbol{r} \end{aligned} & \boldsymbol{x} \in \mathcal{D}, \end{cases} \tag{43}$$

4. Controller Properties

This section presents the properties that ensure the state converges to a neighborhood of the set $\mathcal{B}$. To ease the notation, we consider the set $\mathcal{A}_V \triangleq \{\boldsymbol{x} \in \mathcal{C}_1 : \mu(\boldsymbol{x}) \geq 0\}$, and the maximum value of the Lyapunov function (28) in the set $\mathcal{A}_V$ as V_{max_A}. This set is crucial since we would like to prove it is semi-globally asymptotically stable, as $\mu(\boldsymbol{x})$ is in practice a bound for the performance of the closed-loop system. However, if $\mathcal{A}_V$ is large enough, it is possible for a jump from inside the set to end outside. Thus, since $\dot{V} > 0$ only occurs inside $\mathcal{A}_V$, we consider the set $\mathcal{A} \triangleq \{V \leq V_{max_A}\}$, *i.e.*, the set which contains all the points inside the sublevel set $V(\boldsymbol{x}) = V_{max_A}$. This set has two properties which will allow us to prove semi-global asymptotic stability:

1) $\dot{V} < 0, \forall \boldsymbol{x} \in \mathcal{X} \setminus \mathcal{A}$;

2) $\boldsymbol{G}(\mathcal{A} \cap \mathcal{D}) \subset \mathcal{A}$;

Thus, the proposed controller has the following major properties:

1) The hybrid system follows the hybrid basic conditions.

2) The states are bounded and every solution is a maximal solution.

3) Between each jump, the solution must flow.

4) The set $\mathcal{A} \triangleq \{\boldsymbol{x} \in \mathcal{X} : V \leq V_{max_A}\}$ is semi-globally asymptotically stable for the closed-loop system (43).

5) The time between consecutive jumps admits a strictly positive lower bound when the triggering condition is evaluated on noisy measurements.

6) The ultimate bound on the tracking error is given explicitly in terms of the approximation error and the design parameters.

Lemma 4.1. *The closed-loop hybrid system* (43) *follows the hybrid basic conditions by satisfying the properties:*

H1) $\mathcal{C}$ *and* $\mathcal{D}$ *are closed sets.*

H2) The map $\boldsymbol{F}$: $\mathcal{X} \rightrightarrows \mathcal{X}_f$ *is outer-semicontinuous, convex-valued, locally bounded, and* $\boldsymbol{F}(\boldsymbol{x}) \neq \emptyset$ *for all* $\boldsymbol{x} \in \mathcal{C}$.

H3) The map $\boldsymbol{G}$: $\mathcal{X} \rightrightarrows \mathcal{X}$ *is outer-semicontinuous, locally bounded, and* $\boldsymbol{G}(\boldsymbol{x}) \neq \emptyset$ *for all* $\boldsymbol{x} \in \mathcal{D}$.

Proof. Starting with the first condition, using the fact that the union and intersection of closed sets are closed, it can be proven that $\mathcal{C}$ and $\mathcal{D}$ are closed sets by proving that their components, $\mathcal{C}_1$, $\mathcal{C}_2$, $\mathcal{D}_1$, and $\mathcal{D}_2$, are closed. From (37), $\mathcal{C}_1 \cup \mathcal{D}_1 = \mathcal{X}$, and using (38), $\mathcal{C}_2 \cup \mathcal{D}_2 = \mathcal{X}$, thus, since $\mathcal{X}$ is closed, and each set includes its boundary, it is only necessary to prove that each boundary is continuous. Starting with (37), the boundary is a minimization on $\boldsymbol{J}$ of a continuous function, whereby we use the maximum theorem [41, Theorem 9.14] under Assumption 1 to prove the continuity of the boundary. The continuity of the second boundary (38) is trivial using the rewritten boundary (41). Then, since $\mathcal{C}_1$, $\mathcal{C}_2$, $\mathcal{D}_1$, and $\mathcal{D}_2$ are closed, the sets $\mathcal{C}$ and $\mathcal{D}$ are also closed. According to (43), $\boldsymbol{F}(\boldsymbol{x})$ is a locally-bounded continuous map, single-valued for each $\boldsymbol{x}$, with the exception of the states $\boldsymbol{\omega}_r$ and $\boldsymbol{\omega}_e$, which depend on $\dot{\boldsymbol{\omega}}_r \in M\mathbb{B}$, which is available. $M\mathbb{B}$ is convex and bounded, and since it has no dependence on $\boldsymbol{x}$, it is outer-semicontinuous. $\boldsymbol{G}(\boldsymbol{x})$ is defined for all values of $\boldsymbol{x} \in \mathcal{D}$, and since $G_h(\boldsymbol{x})$ is outer-semicontinuous, as noted after (42), $\boldsymbol{G}(\boldsymbol{x})$ is outer-semicontinuous for all values of $\boldsymbol{x} \in \mathcal{D}$. □

Lemma 4.2. *Consider the closed-loop hybrid system* (43)*, control torque* (27)*, and the adaptation laws* (30a) *and* (30b)*. Then, all the states are bounded and maximal solutions of the closed-loop system are complete.*

Proof. Using [38, Proposition 2.34], the first step is to prove that the closed-loop system follows the condition (VC), i.e., $\boldsymbol{F}(\boldsymbol{x}) \cap \boldsymbol{T}_c(\boldsymbol{x}) \neq \emptyset$, where $\boldsymbol{T}_c(\boldsymbol{x})$ denotes the tangent cone to $\mathcal{C}$ at $\boldsymbol{x}$. Firstly, the tangent cone to any point in the unit-quaternion space is the hyperplane space tangent to it, $\mathbb{H}_P$. Moreover, the tangent cone to any point in the set $\{-1, 1\}$ is $\{0\}$ and to any point in the set $\mathbb{R}^n$ is the space $\mathbb{R}^n$ of the same dimension. Thus, given that $\boldsymbol{T}_c(\boldsymbol{x}) \triangleq \mathbb{H}_P \times \mathbb{R}^3 \times \mathbb{H}_P \times \mathbb{R}^3 \times \mathbb{R}^{N\times 3} \times \mathbb{R}^3 \times \{0\} \times \mathcal{X}_z \times \mathbb{H}_P \times \mathbb{R}^3$, and the derivative of a unit-quaternion is on the plane tangent to it, *i.e.*, $\overline{\boldsymbol{q}} \cdot (\frac{1}{2}\overline{\boldsymbol{q}} \otimes \overline{\boldsymbol{p}}(\boldsymbol{\omega})) = 0$, using (43), the hybrid system follows (VC), and it has a nontrivial solution for every point $\boldsymbol{x} \in \mathcal{C} \cup \mathcal{D}$. Furthermore, from the derivative of the Lyapunov function (35) and the boundary between $\mathcal{D}_2$ and $\mathcal{C}_2$ (41), the value of the Lyapunov function will always decrease outside of a precompact neighborhood of $\mathcal{B}$. Moreover, given the definition of the boundary (37), any jump from $\mathcal{D}_1$ to $\mathcal{C}_1$ will decrease the value of the Lyapunov function, and using (43), it can be verified that all other jumps do not alter the Lyapunov function. Furthermore, from (28), V is positive definite with respect to the set $\mathcal{B}$, where it is equal to zero, and if $||\boldsymbol{z}|| \to +\infty$, then $V \to +\infty$, meaning that V is radially unbounded. To verify this fact, we recall that $\overline{\boldsymbol{q}}_e$ is contained within $\mathbb{H}_1$, which implies that if $||\boldsymbol{z}|| \to +\infty$, then at least one component of $\boldsymbol{\omega}_e$, $\widetilde{\boldsymbol{W}}$, or $\widetilde{\boldsymbol{\psi}}$ tends to $\pm\infty$. However, given that $\boldsymbol{K}_W$ and $\boldsymbol{J}$ are positive definite matrices, V only depends on $\boldsymbol{\omega}_e$, $\widetilde{\boldsymbol{W}}$, or $\widetilde{\boldsymbol{\psi}}$ through quadratic terms, which implies that V is radially unbounded. Thus, all Lyapunov function states, $\boldsymbol{z}$, are bounded. Furthermore, since $\boldsymbol{z}_s$ is only dependent on $\boldsymbol{z}$ and $\overline{\boldsymbol{q}}_r$, $\boldsymbol{\omega}_r$, $\boldsymbol{r}_s$ and h are all contained within compact sets, all states are bounded. Thus, no solution will have finite escape times. Finally, since $\mathcal{C} \cup \mathcal{D} = (\mathcal{C}_1 \cup \mathcal{D}) \cap (\mathcal{C}_2 \cup \mathcal{D}) = \mathcal{X} \cap \mathcal{X} = \mathcal{X}$, it is clear that $\boldsymbol{G}(\mathcal{D}) \subset \mathcal{C} \cup \mathcal{D}$ and the only remaining case from [38, Proposition 2.34] is that every maximal solution is complete. □

Lemma 4.3. *Consider the closed-loop hybrid system* (43)*, control torque* (27)*, and the adaptation laws* (30a) *and* (30b)*. Then, between each jump, the solution must flow, i.e., there are no Zeno solutions.*

Proof. Every jump updates the sampled variables and applies G_h, which by (42) flips h precisely when $m(h) > 0$; we therefore distinguish a jump triggered by $\mathcal{D}_2$ from one triggered by $\mathcal{D}_1$. Let us start by analyzing what occurs when updating $\boldsymbol{z}_s$. Considering $0 < \gamma < 1$, the control law will be updated and $\boldsymbol{\tau}_e$ will become $\mathbf{0}$, thus, according to (41), $\boldsymbol{x}$ will jump to a value where $0 \leq (1-\gamma)\rho + \upsilon$, which holds for every state since $\rho \geq 0$ and $\upsilon > 0$, so the states will be inside $\mathcal{C}_2$. Furthermore, given that the states are bounded, $\dot{V}$ is also bounded, which implies that $\boldsymbol{x}$ must flow between jumps from $\mathcal{D}_2$ to $\mathcal{C}_2$, *i.e.*, the time between jumps has a lower bound. The second jump occurs from $\mathcal{D}_1$ to $\mathcal{C}_1 \cap \mathcal{C}_2$, since both h and $\boldsymbol{z}_s$ are updated. Since $\sigma > 0$, it is known, from (37), that before the jump $m(h) \geq \sigma > 0$. Thus $\overline{\mathrm{sgn}}(m(h)) = 1$ and (42) gives $h^+ = -h$. Since $m(-h) = -\max_{\boldsymbol{X}\in\mathcal{X}_J}(2hk_a\boldsymbol{q}_e^\intercal \boldsymbol{X}\boldsymbol{\omega}_e - 4hk_1 q_{0e}) \leq -m(h)$, after the jump $m(h^+) \leq -\sigma < \sigma$. Therefore, according to (37), the solution will be outside $\mathcal{D}_1$. Thus, since $\boldsymbol{z}_s$ is also updated, the same logic applies and $\boldsymbol{x}$ must flow between jumps from $\mathcal{D}_1$ to $\mathcal{C}_1$. Therefore, both possible jumps will ensure that $\boldsymbol{x}$ must flow between jumps, and since from Lemma 4.2, all the states are bounded, no Zeno solutions may occur.

□

Lemma 4.4. *Consider the closed-loop hybrid system* (43)*, control torque* (27) *held at the sampled value* $\boldsymbol{\tau}_s$ *between consecutive jumps, and the adaptation laws* (30a) *and* (30b)*. Then, the set* $\mathcal{N} \triangleq \{\boldsymbol{x} \in \mathcal{X} : V < V_{max_A}\}$ *is a precompact neighborhood of the set* $\mathcal{B}$*.*

Proof. Let us start by reconsidering the Lyapunov function (28), where V is positive outside of the set $\mathcal{B}$. Furthermore, $\mathcal{A}_V \subset \mathcal{A}$ and $\mathcal{A}_V = \{\boldsymbol{x} \in \mathcal{C}_1 : \mu(\boldsymbol{x}) \geq 0\}$. Since $\rho = 0$, $\widetilde{\boldsymbol{W}} = \mathbf{0}$, $\widetilde{\boldsymbol{\psi}} = \mathbf{0}$ and $\boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e = \mathbf{0}$ on $\mathcal{B}$, evaluating μ there gives $\mu(\boldsymbol{x}) = \upsilon > 0$. Moreover, $m(h) = -4k_1 < \sigma$ on $\mathcal{B}$, and hence $\mathcal{B} \subset \mathcal{C}_1$. Therefore, $\mathcal{B} \subset \mathcal{A}_V$. Moreover, since $\mu = \upsilon > 0$ and $m(h) = -4k_1 < \sigma$ on $\mathcal{B}$, continuity implies that $\mathcal{A}_V$ contains points outside $\mathcal{B}$. Hence, $\mathcal{B} \neq \mathcal{A}_V$ and $V_{max_A} > 0$. To prove that $\mathcal{A}$ is compact, we recall that V_{max_A} is the maximum value of the Lyapunov function inside $\mathcal{A}_V$, and using the same steps from (31) to (35), we can derive $\mu(\boldsymbol{x}) \leq -\rho_2 + (1-\gamma)\rho + Z + \upsilon$. Moreover, since υ is a chosen constant, and using Assumptions 2 and 3, Z is bounded, for any γ larger than zero, $\rho_2 - (1-\gamma)\rho$ is a radially unbounded function with respect to $\boldsymbol{z}$. Thus, $\mathcal{A}_V$ is bounded. Since $\mathcal{A}_V$ is closed, it is compact, and therefore V_{max_A} is finite. Consequently, all the states are bounded inside $\mathcal{A}$. Since V is continuous, $\mathcal{A}$ is closed and hence compact. Finally, since $\overline{\mathcal{N}} \subseteq \mathcal{A}$, the set $\mathcal{N}$ is precompact. □

Theorem 4.5. *Consider the closed-loop hybrid system* (43)*, control torque* (27) *held at the sampled value* $\boldsymbol{\tau}_s$ *between consecutive jumps, and the adaptation laws* (30a) *and* (30b)*. Then, the set* $\mathcal{A}$ *is semi-globally asymptotically stable for the closed-loop system* (43)*.*

Proof. We begin by reconsidering the Lyapunov function (28) and verifying, using (43), that the change across each jump, ΔV, is not positive. Next, using [37, Theorem 8.2] and considering the auxiliary function

$V_{aux} = \max(V - V_{max_A}, 0)$, the complementary set of $\mathcal{C}$ as $\mathcal{C}'$ and of $\mathcal{D}$ as $\mathcal{D}'$, we define:

$$u_C(\boldsymbol{x}) \triangleq \begin{cases} \mu(\boldsymbol{x}) & \boldsymbol{x} \in \mathcal{C} \setminus \mathcal{A}, \\ 0 & \boldsymbol{x} \in \mathcal{C} \cap \mathcal{A}, \\ -\infty & \boldsymbol{x} \in \mathcal{C}', \end{cases} \tag{44}$$

$$u_D(\boldsymbol{x}) \triangleq \begin{cases} \Delta V(\boldsymbol{x}) & \boldsymbol{x} \in \mathcal{D} \setminus \mathcal{A}, \\ 0 & \boldsymbol{x} \in \mathcal{D} \cap \mathcal{A}, \\ -\infty & \boldsymbol{x} \in \mathcal{D}'. \end{cases} \tag{45}$$

Since all maximal solutions are bounded and complete, the growth of V_{aux} along solutions to (43) is bounded by $u_C(\boldsymbol{x})$ and $u_D(\boldsymbol{x})$ on $\mathcal{X}$. Both are nonpositive: on $\mathcal{C} \setminus \mathcal{A}$ we have $u_C = \mu < 0$, since $\mathcal{A}_V \subset \mathcal{A}$ and $\mu < 0$ outside $\mathcal{A}_V$ by definition, while $u_C = 0$ on $\mathcal{C} \cap \mathcal{A}$ by construction; and $u_D = \Delta V \leq 0$ across every jump. It follows from [37, Theorem 8.2] that every maximal solution converges to the set

$$\overline{{u_C}^{-1}(0)} \cup \left({u_D}^{-1}(0) \cap \boldsymbol{G}({u_D}^{-1}(0))\right). \tag{46}$$

Since $\mu < 0$ strictly on $\mathcal{C} \setminus \mathcal{A}$, we have $u_C^{-1}(0) = \mathcal{C} \cap \mathcal{A}$. The set $u_D^{-1}(0)$ may contain points of $\mathcal{D}_2 \setminus \mathcal{A}$ at which h is not flipped and $\Delta V = 0$, but every such point has $\boldsymbol{\tau}_e \neq \mathbf{0}$, whereas every point in the image of $\boldsymbol{G}$ has $\boldsymbol{\tau}_e = \mathbf{0}$; such points therefore do not belong to $\boldsymbol{G}(u_D^{-1}(0))$. On $\mathcal{D}_1$ the jump flips h and $\Delta V \leq -\sigma < 0$. Hence the set above is contained in $\mathcal{A}$, and the closed-loop system under (43) converges to $\mathcal{A}$.

Finally, given that $\mathcal{A}$ is compact, and that V_{aux} is continuous, locally Lipschitz, and positive definite with respect to $\mathcal{A}$, and both $u_C(\boldsymbol{x}) \leq 0$ and $u_D(\boldsymbol{x}) \leq 0$, we use [42, Theorem 7.6] to prove that the set $\mathcal{A}$ is stable. Therefore, the set $\mathcal{A} \triangleq \{V \leq V_{max_A}\}$ is semi-globally asymptotically stable for the closed-loop system. □

Remark 2. *By using [38, Theorem 3.22], the set $\mathcal{A}$ is $\mathcal{KL}$ asymptotically stable for the closed-loop system, and therefore it is semi-globally uniformly asymptotically stable. Thus, using [38, Theorem 3.26], the set $\mathcal{A}$ exhibits semi-global practical robust KL asymptotic stability under disturbances that follow [38, Assumption 3.25]. Therefore, for many practical disturbances such as measurement noise or random disturbances, all solutions will converge to a bounded distance of the set $\mathcal{A}$.*

Now, we consider noisy measurements. To this end, we partition the state as $\boldsymbol{x} = (\overline{\boldsymbol{q}}_e, \boldsymbol{x}_{\text{euc}})$, where $\boldsymbol{x}_{\text{euc}}$ groups all Euclidean components (including $\boldsymbol{\omega}_e$). We define the *noisy state* $\tilde{\boldsymbol{x}} \in \mathcal{X}$ by replacing $\overline{\boldsymbol{q}}_e$ with $\overline{\boldsymbol{q}}_e \otimes \overline{\boldsymbol{n}}_q$ for noise $\overline{\boldsymbol{n}}_q \in \mathbb{H}_1$, and $\boldsymbol{\omega}_e$ with $\boldsymbol{\omega}_e + \boldsymbol{n}_{\boldsymbol{\omega}}$, while all other components remain unchanged.

To define the distance between a noisy measurement and the true state, we equip $\mathcal{X}$ with the product metric combining the geodesic distance $d_q(\boldsymbol{q}_1, \boldsymbol{q}_2) = 2\arccos(\boldsymbol{q}_1 \cdot \boldsymbol{q}_2)$ on $\mathbb{H}_1$ and the Euclidean norm on $\boldsymbol{x}_{\text{euc}}$, the distance between $\tilde{\boldsymbol{x}}$ and $\boldsymbol{x}$ simplifies by invariance to:

$$d(\tilde{\boldsymbol{x}}, \boldsymbol{x}) = \sqrt{d_q(\overline{\boldsymbol{q}}_e \otimes \overline{\boldsymbol{n}}_q, \overline{\boldsymbol{q}}_e)^2 + \|\boldsymbol{n}_{\boldsymbol{\omega}}\|^2} = \sqrt{\theta_n^2 + \|\boldsymbol{n}_{\boldsymbol{\omega}}\|^2} \triangleq \delta,$$

where $\theta_n \in [0, 2\pi]$ is the rotation angle of $\overline{\boldsymbol{n}}_q$; we call δ the noise level of $\tilde{\boldsymbol{x}}$. In the next result, the controller evaluates Θ and Φ at $\tilde{\boldsymbol{x}}$.

Proposition 4.6. *Consider the closed-loop hybrid system* (43)*, the control torque* (27) *held at the sampled value $\boldsymbol{\tau}_s$ between consecutive jumps, and the adaptation laws* (30a) *and* (30b)*, and suppose the triggering condition* (41) *and the sampled control law are evaluated at noisy states of level at most δ, and that the resulting perturbation satisfies [38, Assumption 3.25]. Suppose in addition that $\dot{\boldsymbol{\omega}}_r$ is Lipschitz continuous with constant M_2. Then, for every compact set of initial conditions $\mathcal{K} \subset \mathcal{X}$ there exist $\delta^\star > 0$ and $T_{\min} > 0$ such that, for every $\delta < \delta^\star$, consecutive jump times of any maximal solution from $\mathcal{K}$ satisfy $t_{j+1} - t_j \geq T_{\min}$. In particular, the closed-loop system admits no Zeno solutions.*

Proof. Let $\boldsymbol{s} \triangleq \boldsymbol{\omega}_e + k_a h \boldsymbol{q}_e$ and define:

$$\Theta(\boldsymbol{x}, \boldsymbol{\tau}_s) \triangleq -\boldsymbol{s}^\mathsf{T} \boldsymbol{\tau}_e - (1-\gamma)\rho(\boldsymbol{x}) - \upsilon, \qquad \Phi(\boldsymbol{x}) \triangleq m(h) - \sigma,$$

so that $\mathcal{D}_2 = \{\Theta \geq 0\}$ and $\mathcal{D}_1 = \{\Phi \geq 0\}$ by (41) and (37).

By Lemma 4.1 the closed loop satisfies the hybrid basic conditions, by Lemma 4.4 the set $\mathcal{A}$ is compact, and by Theorem 4.5 together with [38, Theorem 3.22] it is $\mathcal{KL}$ pre-asymptotically stable. Fixing any $\epsilon > 0$, [38, Theorem 3.26] then provides $\delta_1 > 0$ and a compact set $\mathcal{S}$, a sublevel set of a proper indicator of $\mathcal{A}$, with $\mathcal{A} \subset \operatorname{int} \mathcal{S}$ and such that every solution from $\mathcal{K}$ with $\delta < \delta_1$ remains in $\mathcal{S}$.

The function Θ is continuously differentiable in $\boldsymbol{x}$ and Lipschitz in $\dot{\boldsymbol{\omega}}_r$ with some constant c_τ on $\mathcal{S}$; since $\dot{\boldsymbol{\omega}}_r$ is Lipschitz with constant M_2, the map $t \mapsto \Theta$ is absolutely continuous along flows. The function Φ need not be differentiable, being a pointwise minimum, but its growth is controlled by that of the individual smooth functions in the family. Writing $b(\boldsymbol{X}, \boldsymbol{x}) \triangleq 2hk_a \boldsymbol{q}_e^\mathsf{T} \boldsymbol{X} \boldsymbol{\omega}_e - 4hk_1 q_{0e}$, so that $m(h) = \min_{\boldsymbol{X} \in \mathcal{X}_J} b(\boldsymbol{X}, \boldsymbol{x})$, and letting $\boldsymbol{X}^\star$ attain the minimum at $\boldsymbol{x}(s,j)$, for every $t \geq s$ in the same interval of flow:

$$m(h)\big|_{\boldsymbol{x}(t,j)} \leq b(\boldsymbol{X}^\star, \boldsymbol{x}(t,j)) = m(h)\big|_{\boldsymbol{x}(s,j)} + \int_s^t \big\langle \nabla_{\boldsymbol{x}} b(\boldsymbol{X}^\star, \cdot), \dot{\boldsymbol{x}} \big\rangle \, \mathrm{d}r,$$

the inequality because $\boldsymbol{X}^\star \in \mathcal{X}_J$ and the equality by the definition of $\boldsymbol{X}^\star$. Since $\mathcal{S}$ and $\mathcal{X}_J$ are compact, $\boldsymbol{F}$ is locally bounded by Lemma 4.1 and $\boldsymbol{\tau}_s$ ranges over the image of $\mathcal{S}$ under (27), the quantity:

$$\Lambda \triangleq \max\Big\{ \sup \big|\langle \nabla_{\boldsymbol{x}} \Theta, \boldsymbol{f} \rangle\big| + c_\tau M_2, \ \sup \big|\langle \nabla_{\boldsymbol{x}} b(\boldsymbol{X}, \cdot), \boldsymbol{f} \rangle\big| \Big\},$$

with the suprema over $\boldsymbol{x} \in \mathcal{S}$, $\boldsymbol{f} \in \boldsymbol{F}(\boldsymbol{x})$, $\boldsymbol{\tau}_s \in \boldsymbol{\tau}(\mathcal{S})$ and $\boldsymbol{X} \in \mathcal{X}_J$, is finite and bounds the growth rate of both Θ and Φ along any interval of flow in $\mathcal{S}$. Both are uniformly continuous, with respect to d, on the compact set $\{\boldsymbol{y} : d(\boldsymbol{y}, \mathcal{S}) \leq \delta_1\}$, which contains every noisy state, and Θ uniformly in $\boldsymbol{\tau}_s$; let ϖ be a common modulus of continuity there. Define:

$$\delta^\star \triangleq \min\Big\{ \delta_1, \ \sup\big\{ \delta \geq 0 : \varpi(\delta) \leq \tfrac{1}{8} \min\{\upsilon, \sigma\} \big\} \Big\}.$$

Write $\operatorname{dom} \boldsymbol{x} = \bigcup_j ([t_j, t_{j+1}] \times \{j\})$ with $t_0 = 0$, so that the j-th jump occurs at ordinary time t_j and maps $\boldsymbol{x}(t_j, j-1)$ to $\boldsymbol{x}(t_j, j)$. For $g \in \{\Theta, \Phi\}$, with $\boldsymbol{\tau}_s$ held at its post-jump value, and for every t in the interval of flow following t_j, it holds that:

$$g(\tilde{\boldsymbol{x}}(t,j)) \leq g(\tilde{\boldsymbol{x}}(t_j, j)) + 2\varpi(\delta) + \Lambda(t - t_j),$$

since g changes by at most $\varpi(\delta)$ on each passage between the true and the noisy state, at t_j and at t, and by at most $\Lambda(t - t_j)$ along the flow between them.

Every jump resets the sampled variables, and $\boldsymbol{\tau}$ and $\boldsymbol{\tau}_s$ are then evaluated at the same noisy state, so $\boldsymbol{\tau}_e = \boldsymbol{0}$ there and $\Theta(\tilde{\boldsymbol{x}}(t_j, j), \boldsymbol{\tau}_s) = -(1-\gamma)\rho - \upsilon \leq -\upsilon$. By the bound above, no jump associated with $\mathcal{D}_2$ can occur before $t_j + (\upsilon - 2\varpi(\delta))/\Lambda$.

Next, the sign in the m is evaluated at the noisy state, where m takes a value $\tilde{m}$ with $|\tilde{m} - m(h)| \leq \varpi(\delta)$. If $\tilde{m} > 0$ then $h^+ = -h$ and $m(h^+) \leq -m(h) \leq \varpi(\delta) - \tilde{m} < \varpi(\delta)$; otherwise $h^+ = h$ and $m(h^+) = m(h) \leq \tilde{m} + \varpi(\delta) \leq \varpi(\delta)$. Hence $\Phi(\tilde{\boldsymbol{x}}(t_j, j)) \leq -\sigma + 2\varpi(\delta)$ and, by the bound above, no jump associated with $\mathcal{D}_1$ can occur before $t_j + (\sigma - 4\varpi(\delta))/\Lambda$.

Since $\varpi(\delta) \leq \frac{1}{8}\min\{\upsilon, \sigma\}$ for $\delta < \delta^\star$, both post-jump values are strictly negative, so the noisy state lies outside $\mathcal{D}_1 \cup \mathcal{D}_2$ and the solution must flow after every jump. Combining the two bounds:

$$t_{j+1} - t_j \geq \frac{\min\{\upsilon, \sigma\} - 4\varpi(\delta)}{\Lambda} \geq \frac{\min\{\upsilon, \sigma\}}{2\Lambda} \triangleq T_{\min} > 0.$$

Thus $\operatorname{dom} \boldsymbol{x}$ is unbounded in the t direction and no Zeno solutions exist. $\square$

Finally, the following corollary establishes explicit ultimate bounds on the attitude and angular velocity tracking errors.

Corollary 4.7. *Consider the closed-loop hybrid system* (43)*, the control torque* (27) *held at the sampled value $\boldsymbol{\tau}_s$ between consecutive jumps, and the adaptation laws* (30a) *and* (30b)*, and let:*

$$B_q \triangleq \sqrt{\frac{Z+\upsilon}{\gamma k_a k_1}}, \qquad B_\omega \triangleq \sqrt{\frac{Z+\upsilon}{\gamma k_2}} + k_a B_q.$$

Suppose that $B_q < 1$ and:

$$4k_1\sqrt{1-B_q^2} - 2k_a J_{\max} B_q B_\omega > \sigma, \tag{47}$$

then $\|\boldsymbol{q}_e\| \le B_q$ and $\|\boldsymbol{\omega}_e\| \le B_\omega$ for all $\boldsymbol{x} \in \mathcal{A}_V$, and it holds that:

1. *If $V_{max_A} < 2k_1$, then for all $\boldsymbol{x} \in \mathcal{A}$ it holds that:*

$$\|\boldsymbol{q}_e\| \le \bar{B}_q \triangleq \frac{1}{2k_1}\sqrt{V_{max_A}(4k_1 - V_{max_A})}, \qquad \|\boldsymbol{\omega}_e\| \le \bar{B}_\omega \triangleq \sqrt{\frac{2V_{max_A}}{J_{\min}}} + k_a \bar{B}_q,$$

2. *For every $r_q, r_\omega > 0$, the design parameters can be selected so that $V_{max_A} < 2k_1$ and* (47) *are satisfied, and both $\bar{B}_q < r_q$, and $\bar{B}_\omega < r_\omega$ hold.*

3. *If $V_{max_A} < 2k_1$, then, for every compact set of initial conditions $\mathcal{K} \subset \mathcal{X}$, there exist $\delta^\star > 0$ and a nondecreasing $\rho_{\mathcal{K}} : [0, \delta^\star) \to \mathbb{R}_{\ge 0}$ with $\rho_{\mathcal{K}}(\delta) \to 0$ as $\delta \to 0$, such that every maximal solution from $\mathcal{K}$ evaluated at noisy states of level at most $\delta < \delta^\star$ satisfies:*

$$\limsup_{t+j\to\infty} \|\boldsymbol{q}_e\| \le \bar{B}_q + \rho_{\mathcal{K}}(\delta), \qquad \limsup_{t+j\to\infty} \|\boldsymbol{\omega}_e\| \le \bar{B}_\omega + \rho_{\mathcal{K}}(\delta).$$

Proof. The proof of Lemma 4.4 establishes $\mu(\boldsymbol{x}) \le -W_2 + (1-\gamma)\rho(\boldsymbol{x}) + Z + \upsilon$ with $W_2 = \rho + \frac{\gamma_{\boldsymbol{W}}}{2}\|\widetilde{\boldsymbol{W}}\|_F^2 + \frac{\gamma_{\boldsymbol{\psi}}}{2}\|\widetilde{\boldsymbol{\psi}}\|^2$, that is:

$$\mu(\boldsymbol{x}) \le -\gamma\rho(\boldsymbol{x}) - \tfrac{\gamma_{\boldsymbol{W}}}{2}\|\widetilde{\boldsymbol{W}}\|_F^2 - \tfrac{\gamma_{\boldsymbol{\psi}}}{2}\|\widetilde{\boldsymbol{\psi}}\|^2 + Z + \upsilon.$$

On $\mathcal{A}_V$ the three subtracted terms are nonnegative, so each is bounded by $Z+\upsilon$. In particular $\gamma\rho \le Z+\upsilon$, and since $\rho = k_a k_1\|\boldsymbol{q}_e\|^2 + k_2\|\boldsymbol{s}\|^2$ with $\boldsymbol{s} = \boldsymbol{\omega}_e + k_a h\boldsymbol{q}_e$, both summands are bounded by $(Z+\upsilon)/\gamma$, giving $\|\boldsymbol{q}_e\| \le B_q$ and $\|\boldsymbol{s}\| \le \sqrt{(Z+\upsilon)/(\gamma k_2)}$. Since $|h| = 1$, applying the triangle inequality to $\boldsymbol{\omega}_e = \boldsymbol{s} - k_a h\boldsymbol{q}_e$ yields $\|\boldsymbol{\omega}_e\| \le B_\omega$. Moreover, if $hq_{0e} \le 0$ at some $\boldsymbol{x} \in \mathcal{A}_V$, then $m(h) \ge 4k_1\sqrt{1-B_q^2} - 2k_a J_{\max} B_q B_\omega > \sigma$, which contradicts $\mathcal{A}_V \subseteq \mathcal{C}_1$. Hence, $hq_{0e} > 0$ on $\mathcal{A}_V$.

For the first claim: On $\mathcal{A}$ every term of (28) is bounded by V_{max_A}. From $2k_1(1-hq_{0e}) \le V_{max_A}$ and $|h| = 1$ we get $hq_{0e} \ge 1 - V_{max_A}/(2k_1)$, which is positive by hypothesis, so $\|\boldsymbol{q}_e\|^2 = 1 - q_{0e}^2 \le \bar{B}_q^2$. From $\frac{1}{2}\boldsymbol{s}^\mathsf{T}\boldsymbol{J}\boldsymbol{s} \le V_{max_A}$ and Assumption 1, $\|\boldsymbol{s}\| \le \sqrt{2V_{max_A}/J_{\min}}$, and the triangle inequality gives $\|\boldsymbol{\omega}_e\| \le \bar{B}_\omega$.

For the second claim: Using the bounds established above and the fact that $hq_{0e} > 0$ on $\mathcal{A}_V$, we obtain

$$V_{max_A} \le \frac{2(Z+\upsilon)}{\gamma k_a} + \frac{J_{\max}(Z+\upsilon)}{2\gamma k_2} + \frac{Z+\upsilon}{\gamma_{\boldsymbol{W}}\lambda_{\min}(K_{\boldsymbol{W}})} + \frac{Z+\upsilon}{\gamma_{\boldsymbol{\psi}} k_\psi}. \tag{48}$$

Indeed, this follows from $1 - \sqrt{1-r} \le r$ for $r \in [0,1]$ and from the bounds on $\boldsymbol{q}_e$, $\boldsymbol{s}$, $\widetilde{\boldsymbol{W}}$, and $\widetilde{\boldsymbol{\psi}}$ on $\mathcal{A}_V$.

Fix $\epsilon, \gamma_W, \gamma_\psi, \upsilon > 0$ so that $Z_\upsilon = Z + \upsilon$ is sufficently small. With these parameters fixed, choose k_2, $\lambda_{\min}(K_W)$, and k_ψ sufficiently large so that $\sqrt{\frac{2V_{max_A}}{J_{\min}}} < \frac{r_\omega}{2}$.

Then, pick k_1 sufficiently large so that $B_q < 1$ and σ sufficiently small so that (47) and $\bar{B}_q < \min\left\{r_q, \frac{r_\omega}{2k_a}\right\}$ hold. It follows that $\bar{B}_q < r_q$ and $\bar{B}_\omega < r_\omega$.

For the last claim, as in the proof of Proposition 4.6, for every compact set $\mathcal{K} \subset \mathcal{X}$ there exist $\delta^\star > 0$ and a nondecreasing function $\rho_{\mathcal{K}} : [0, \delta^\star) \to \mathbb{R}_{\ge 0}$, with $\rho_{\mathcal{K}}(\delta) \to 0$ as $\delta \to 0$, such that every maximal solution from $\mathcal{K}$ under measurement noise of level at most $\delta < \delta^\star$ satisfies $\limsup_{t+j\to\infty} d(\boldsymbol{x}(t,j), \mathcal{A}) \le \rho_{\mathcal{K}}(\delta)$.

Since $\mathcal{A}$ is compact, for every $\boldsymbol{x}$ there exists $\boldsymbol{y} \in \mathcal{A}$ such that $d(\boldsymbol{x}, \boldsymbol{y}) = d(\boldsymbol{x}, \mathcal{A})$. Moreover, the maps $\boldsymbol{x} \mapsto \|\boldsymbol{q}_e\|$ and $\boldsymbol{x} \mapsto \|\boldsymbol{\omega}_e\|$ are 1-Lipschitz with respect to d. Therefore,

$$\|\boldsymbol{q}_e(\boldsymbol{x})\| \leq \bar{B}_q + d(\boldsymbol{x}, \mathcal{A}), \qquad \|\boldsymbol{\omega}_e(\boldsymbol{x})\| \leq \bar{B}_\omega + d(\boldsymbol{x}, \mathcal{A}).$$

Taking the limit superior gives the claim. □

Remark 3. *The constant Z is finite by Assumptions 2 and 3, but it is not computable, since Assumption 2 only asserts the existence of $\boldsymbol{\psi}$ and Assumption 3 bounds $\boldsymbol{W}$ without identifying it. Corollary 4.7 is therefore to be read as a statement about how the ultimate error scales with the design parameters rather than as a numerical bound. Since $\boldsymbol{q}_e$ is the vector part of a unit quaternion, $||\boldsymbol{q}_e|| \leq 1$ always, so the bound on the attitude error becomes informative only once the gains are large enough that $\gamma k_a k_1 > Z + \upsilon$.*

Remark 4. *The construction is not specific to the rigid body dynamics. It requires only a backstepping-compatible cascade in which the uncertainty is matched with the control input, so that it can be compensated by an adaptive term in the usual manner, and a function ρ that is computable from measured signals. Any adaptive backstepping design meeting these two conditions can be equipped with the proposed event-triggered layer, including on Euclidean state spaces. The hybrid variable h is the single component specific to the present setting, where it is required because the unit-quaternion parametrization double covers the attitude space; when this obstruction is absent, the same design applies with the hysteresis logic omitted. We treat attitude tracking because it is precisely the case in which the obstruction and the sampling mechanism have to be handled simultaneously, and the same construction carries over to other parametrizations of this problem, such as tracking on $SE(3)$.*

Remark 5. *Impulsive effects on the plant [31, 32], such as jumps $\boldsymbol{\omega}_B^+ = \boldsymbol{\omega}_B + \boldsymbol{\iota}(\boldsymbol{\omega}_B)$ occurring on a set $\mathcal{D}_\iota$, are naturally accommodated in this framework by enlarging the jump set to $\mathcal{D} \cup \mathcal{D}_\iota$. Provided $\mathcal{D}_\iota$ is closed and the impulse map is outer-semicontinuous and locally bounded, the hybrid basic conditions of Lemma 4.1 continue to hold, and the conclusions of Theorem 4.5 are preserved whenever the impulses do not increase the Lyapunov function, or increase it by a bounded amount and are separated by a sufficiently long dwell time.*

5. Simulation Results

5.1. Nominal Study

To analyze the simulation results, three different parameters were used in order to measure the performance. The first was the angle α, which denotes the angle in degrees of the axis-angle rotation denoted by $h\overline{\boldsymbol{q}}_e$. The second was the norm of the angular velocity, and the third was a measure of the total energy consumed by the controller, given by:

$$E \triangleq \sqrt{\int_0^t \boldsymbol{\tau}^\intercal \boldsymbol{\tau} d\sigma}. \tag{49}$$

There were three external disturbances: a constant disturbance equal to $\boldsymbol{\tau}_d \triangleq [1/\sqrt{3}\ 1/\sqrt{3}\ 1/\sqrt{3}]^\intercal$ N·m, a torque given by $-2|\boldsymbol{\omega}_B|$ N·m, and a random disturbance torque described by a normal distribution with standard deviation of $[1\ 1\ 1]^\intercal$ N·m. Furthermore, the output of each sensor was corrupted by white noise, resulting in an additive white Gaussian noise with a variance of $[5\times10^{-5}\ 5\times10^{-5}\ 5\times10^{-5}\ 5\times10^{-5}]^\intercal$ in the quaternion measurements and $[5\times10^{-5}\ 5\times10^{-5}\ 5\times10^{-5}]^\intercal$ in the angular velocity measurements. Finally, there was a modeling error derived from a difference between the nominal moment of inertia used to develop the controller, $\widehat{\boldsymbol{J}}$, and its actual value in the dynamics $\boldsymbol{J}$, given by:

$$\boldsymbol{J} = \begin{bmatrix} 40 & 1 & 0.5 \\ 1 & 15 & 2 \\ 0.5 & 2 & 20 \end{bmatrix} \quad \widehat{\boldsymbol{J}} = \begin{bmatrix} 36 & 0 & 0 \\ 0 & 18 & 0 \\ 0 & 0 & 25 \end{bmatrix}.$$

Table 1: Parameter values for the simulations.

Parameter	Value
k_1	10
k_2	20
k_a	1
$\boldsymbol{K_W}$	$0.2\boldsymbol{I}_N$
k_{ψ}	0.01
$\gamma_{\boldsymbol{W}}$	0.01
γ_{ψ}	0.1
N	256
$\sigma_{\boldsymbol{W}}$	6
υ	10^{-2}
γ	0.1

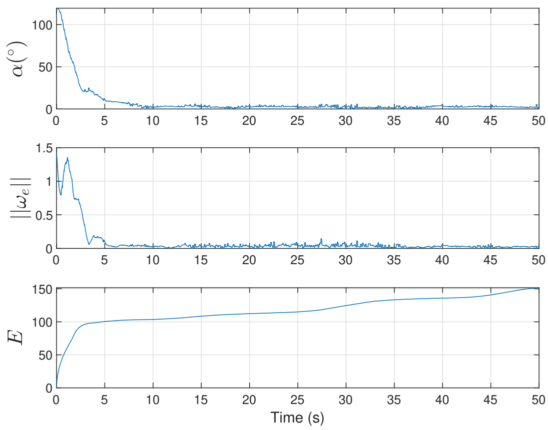

Figure 2: Tracking results for the disturbed system using the reference $\boldsymbol{\omega}_r = [\cos(t/5), \sin(t/6), \cos(t/7)]$.

Thus, $\boldsymbol{f}_d(\boldsymbol{v}_d)$ is given by $\boldsymbol{f}_d(\boldsymbol{v}_d) \triangleq \boldsymbol{\tau}_d - 2|\boldsymbol{\omega}_B|$ and $\boldsymbol{f}(\boldsymbol{v})$ is given by (25). To ease the implementation, we note, as shown in [35], that the minimization in the boundary between $\mathcal{C}_1$ and $\mathcal{D}_1$ is the minimization of an affine function, which enables a smaller computational load by considering that there is a component-wise bound on $\boldsymbol{J}$. The controller parameters used can be found in Table 1.

The first simulation consisted of tracking the reference $\boldsymbol{\omega}_r = [\cos(t/5)\ \sin(t/6)\ \cos(t/7)]$, and the tracking results can be seen in Fig. 2. In Fig. 3, the RBFNN output is compared with the actual total disturbance.

Fig. 2 shows that the controller is capable of tracking time-varying references, even in the presence of different disturbances and a modeling error. In Fig. 3, it can be seen that after 5 seconds, the network was able to compensate for the disturbances, albeit with a small residual error. Furthermore, the disturbance estimation of the controller has a similar accuracy to more classic techniques, as can be seen in Fig. 4, where the estimation error of $\boldsymbol{f}(\boldsymbol{v})$ without $-2|\boldsymbol{\omega}_B|$ was compared with the accuracy of the controller in [35], which uses the classic adaptive backstepping approach with a hybrid controller.

The tracking error can decrease even more by increasing the number of neurons; however, this strategy leads to an increase in computational load and may lead to an increase in samples. To verify this fact, we measured the average sampling interval (T_{avg}) in four different cases, $N = 128$, $N = 256$, $N = 512$, and $N = 1024$, maintaining the reference, the disturbances, and the modeling error. The results can be seen in Fig. 5.

Since the event-triggered mechanism depends heavily on the current convergence rate, it can be verified,

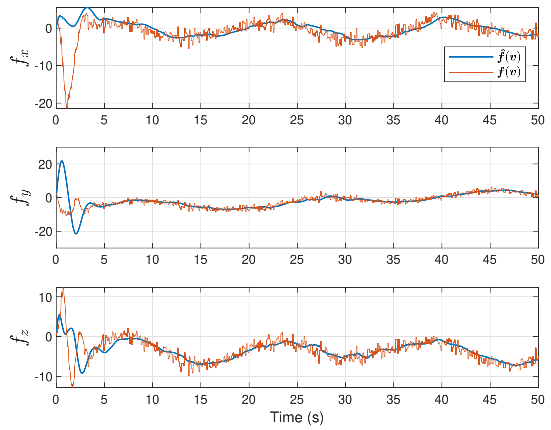

Figure 3: Comparison between the output of the neural network $\hat{\boldsymbol{f}}(\boldsymbol{v})$ and the sum of all disturbances and modeling error $\boldsymbol{f}(\boldsymbol{v})$.

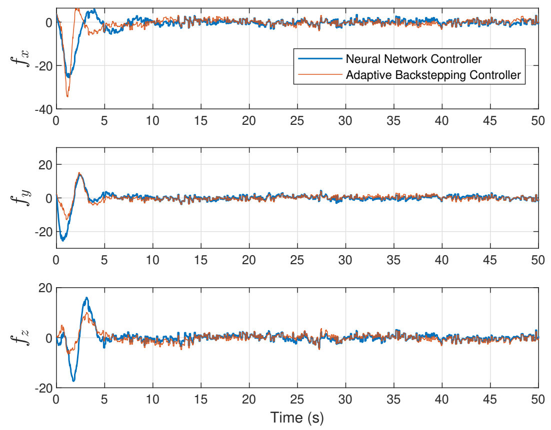

Figure 4: Comparison between the estimation error of a classic adaptive backstepping controller and the neural network event-triggered controller.

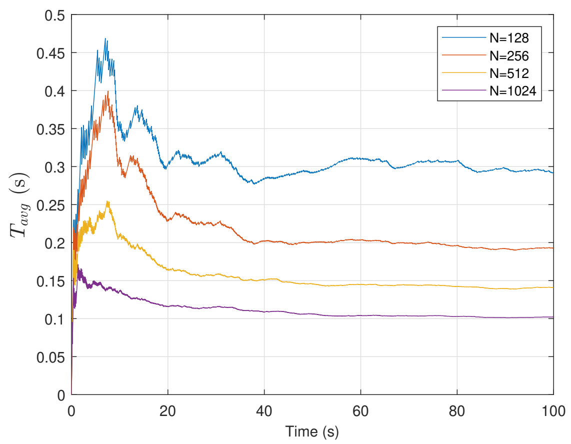

Figure 5: Comparison of the average time between samples, T_{avg}, using different sizes of neural networks.

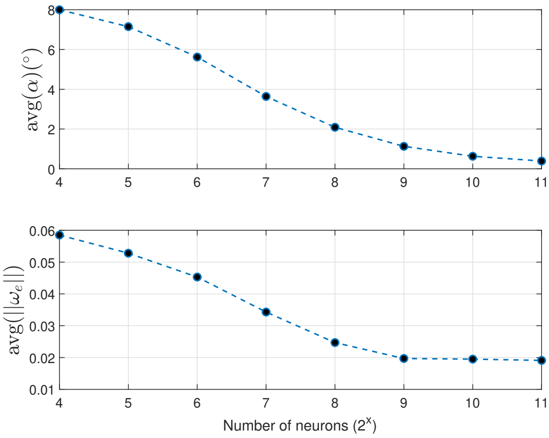

Figure 6: Average attitude and angular velocity error in increasing neural network size in the last 10 seconds of the simulation.

using the Lyapunov function (35), that the switching depends on several factors, namely the reference and the parameters used. Due to the shape of the reference and the disturbances, the computed control law requires a high number of updates. By using a larger number of neurons, the ideal approximation error decreases, and the size of the error set decreases as well. In that case, the controller requires more samples to keep the tracking error within the more restrictive error set, especially when dealing with time-varying references.

To verify that the error decreases when the size of the neural network increases, we measured the average α and $||\boldsymbol{\omega}_e||$ in the last 10 seconds of a 100-second simulation, keeping the same disturbances and parameters, except for N. The simulations were performed without measurement noise to verify more easily the decrease in the size of the error set. The main reason behind this experiment is that, from the bound of Corollary 4.7, an increased gain will result in a lower error, but a larger network is not guaranteed to have the same effect due to the $||\boldsymbol{W}||_F^2$ component of Z. The main objective was to verify an intuitive result, *i.e.*, a larger network implies a lower attitude and angular velocity error, but, using theoretical results, we cannot ascertain whether this conjecture is true, how small the error set will get, or how much of a difference a larger network will make, due to the lack of knowledge of $||\boldsymbol{W}||^2$, even if $||\boldsymbol{\psi}||^2$ decreases.

In our case, Fig. 6 shows that the increase in the number of neurons leads to a decrease in the size of the error set. Even if the results might suggest using larger networks, as seen in Fig. 7, a greater number of neurons increases the number of samples. Thus, due to the event-triggered mechanism, there is a trade-off between the number of samples and the average tracking error.

To visualize how the sampling process works, Fig. 7 presents the control law in the first 15 seconds of the first simulation, with $N = 256$, including both the sampled signal, the computed control law, and the sampling time. Furthermore, Fig. 8 displays the evolution of the inter-event times, where it can be seen that it does not drop below 0.15 seconds.

Fig. 7 shows that the sampling depends on the current performance, even if the value of the Lyapunov function cannot be computed. Namely, there is a clear contrast between the concentration of samples near the 10-second mark and the time between samples at the beginning. However, even with time windows with a significant concentration of samples, from Fig. 5, it can be seen that the time between samples tends towards an average of 0.2 seconds.

5.2. Monte Carlo Study

The results reported so far correspond to a single operating point. To assess the controller over a range of conditions, a Monte Carlo study of 100 runs was carried out at each of three measurement noise levels, for a total of 300 runs of 100 s each. All quantities that are randomized, together with the parameters that differ from Table 1, are collected in Table 2; the remaining gains and the network are those of Table 1.

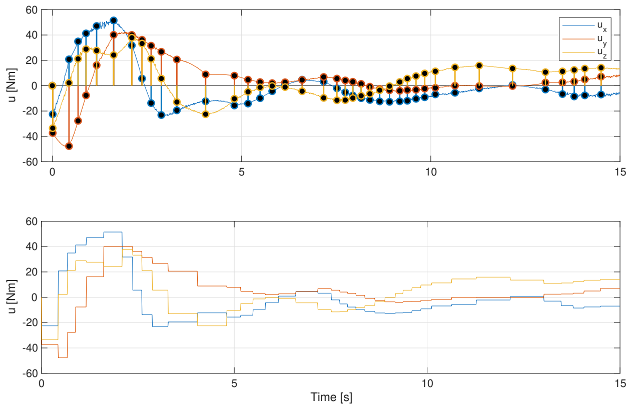

Figure 7: Computed control law, sampling times, and information received by the actuators.

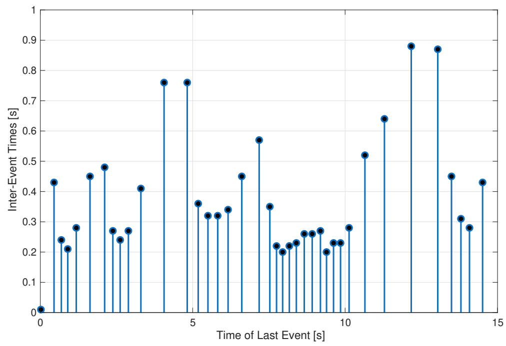

Figure 8: Evolution of inter-event times. Each point represents the time elapsed since the previous event, plotted against the time at which the event occurred.

Table 2: Parameters and sampling distributions for the Monte Carlo study. $\mathcal{U}$ denotes the uniform distribution and each component is drawn independently.

Quantity	Value or distribution
Runs per noise level	100
Horizon	100 s
k_1	20
υ	1
γ	0.1
$\overline{\boldsymbol{q}}(0)$	uniform on $\mathbb{H}_1$
$\boldsymbol{\omega}_B(0)$	$\mathcal{U}[-1,1]$ rad $\cdot$ s^{-1}
a	$\mathcal{U}[0,2]$ rad $\cdot$ s^{-1}
ω_i	$\mathcal{U}[0,1/2]$ rad $\cdot$ s^{-1}
C	$\mathcal{U}[0,2/\sqrt{3}]$ N $\cdot$ m
v	$\mathcal{U}[0,1]$
Noise variance	$\{2,4,10\} \times 10^{-5}$

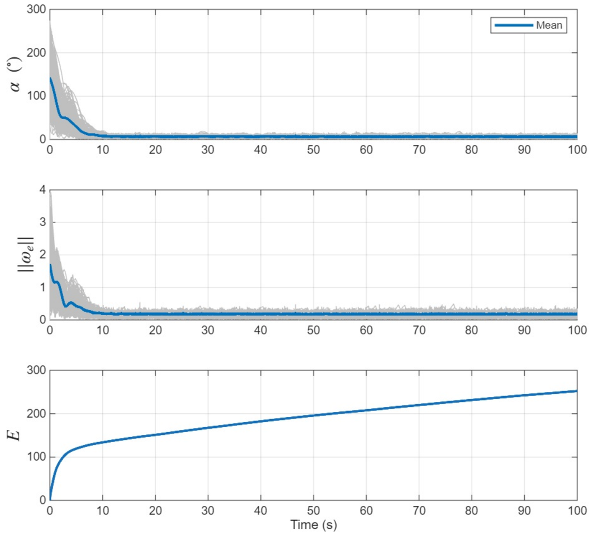

Figure 9: Attitude error, angular velocity error and consumed energy over the Monte Carlo study. Individual runs in grey, mean in colour.

The initial attitude is drawn uniformly on $\mathbb{H}_1$ by normalizing a vector of four independent standard normal variates. Since this is the Haar distribution on the double cover, the two representations of the reference attitude occur with equal probability, and the hysteresis mechanism is therefore exercised in both directions across the study. The initial angular velocity is drawn uniformly in $[-1, 1]$ rad$\cdot$s^{-1} per component.

The reference is $\boldsymbol{\omega}_r = a[\cos(\omega_1 t)\ \sin(\omega_2 t)\ \cos(\omega_3 t)]^\mathsf{T}$, with the amplitude a and the three rates ω_i drawn independently, so that both the magnitude and the frequency content of the reference vary from run to run. The reference attitude is obtained by integrating $\boldsymbol{\omega}_r$, and $\dot{\boldsymbol{\omega}}_r$ is supplied to the feedforward analytically.

The disturbances are randomized as well: the constant disturbance is $C\mathbf{1}$ with C drawn per run, and the random disturbance torque has variance v per component, also drawn per run. Together with the three noise levels, this means that no two runs share the same operating point.

Fig. 9 shows α and $||\boldsymbol{\omega}_e||$ for every run together with their mean, and the mean of E. Every run converges to a neighborhood of the null error set: averaged over the last 10 s and over the study, the attitude error is 6.33° with a standard deviation of 0.86° and the angular velocity error is 0.176 with a standard deviation of 0.017. The spread visible during the transient reflects the range of initial conditions, references and disturbance magnitudes rather than a loss of accuracy, and the same spread appears in the consumed energy, 252 ± 126.

Fig. 10 shows the distribution of the inter-event times pooled over all runs. The controller issued on average 165.3 updates per run, with a standard deviation of 39.6, against the 10^4 updates that a periodic implementation at the rate at which the triggering condition is evaluated would require, a reduction of over 98%. The mean inter-event time is 0.606 s and the distribution is concentrated between roughly 0.2 s and 1.2 s. The smallest interval observed is 0.01 s, which coincides with the period at which the triggering condition is evaluated.

5.3. Unwinding and Chattering

Finally, to show the importance of tackling the unwinding phenomenon and using hybrid control, two additional simulations were performed. Using the same disturbances as the last simulation, the controller tracked a constant reference, and after 20 seconds, the reference was switched. To contrast our solution with a continuous controller, we implemented the controller proposed in [43], similar to the solution shown in [30]. The results were compared and can be seen in Fig. 11.

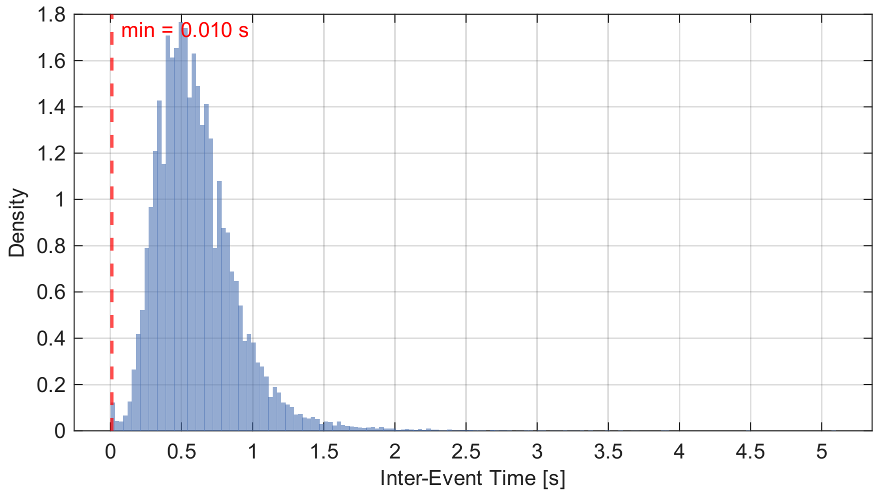

Figure 10: Distribution of the inter-event times pooled over the Monte Carlo study.

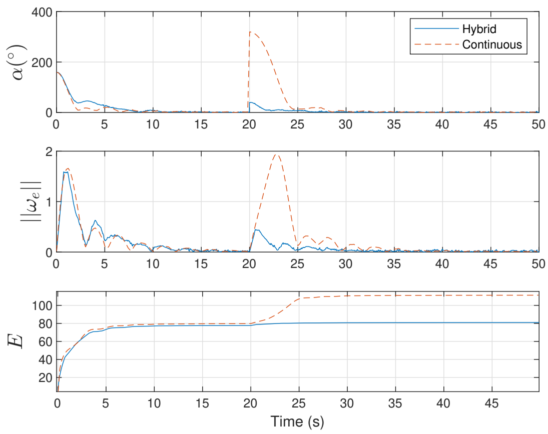

Figure 11: Comparison between a hybrid and a continuous controller in the presence of the unwinding phenomenon.

Both cases require a similar amount of energy to converge. The key difference appears after the reference switches. In the first scenario, both reference quaternions had a positive real component, namely $\overline{\boldsymbol{q}}_{r1} = [0.1736, -0.6411, 0.3846, 0.6411]^{\mathsf{T}}$ and $\overline{\boldsymbol{q}}_{r2} = [0.1736, 0.6411, -0.3846, -0.6411]^{\mathsf{T}}$. Since quaternion space is continuous, the closest equilibrium in the second case is $-\overline{\boldsymbol{q}}_{r2}$. However, the continuous controller can only converge to $\overline{\boldsymbol{q}}_{r2}$, unlike the hybrid controller, which reached $-\overline{\boldsymbol{q}}_{r2}$.

One can conclude that the continuous controller sees a greater error, since it can only converge to one equilibrium. The parameter α is given by $\alpha = h\overline{\boldsymbol{q}}_e$, thus, when h does not adapt to the nearest equilibrium, it can perceive an error greater than $180°$. In practice, this will cause the closed-loop system to spend much larger amounts of energy, as shown in Fig. 11, so that the event-triggered hybrid design saves a significant amount of control effort over the continuous one. The two implementations were also comparable in cost: the simulations required 1.65 s and 1.57 s respectively. The cost per evaluation of the control law grows linearly with the network size, since (27) requires $O(N)$ operations to evaluate $\boldsymbol{\Psi}$, so the reduction in the number of updates translates directly into a proportional reduction in the on-board computation devoted to the network.

The unwinding problem is unavoidable when using continuous controllers. Another notable example is

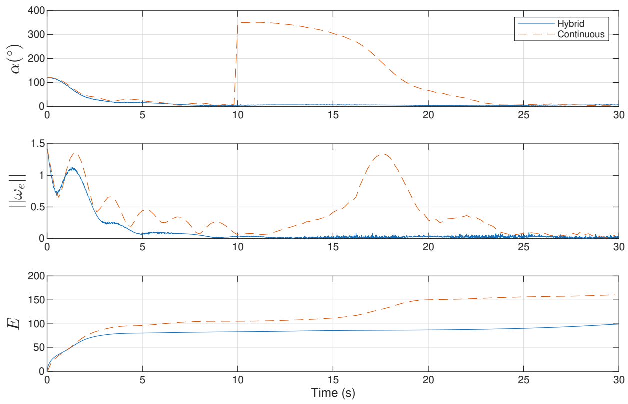

Figure 12: Comparison with the controller developed in [29].

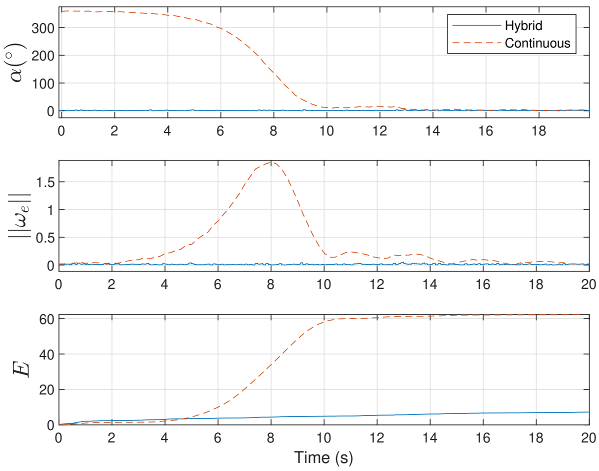

Figure 13: Simulation demonstrating the problem arising from unstable equilibria in a continuous controller.

[29]: although their controller is also an adaptive neural network controller, it cannot guarantee convergence to the nearest equilibrium. Using the same conditions as the previous examples, Fig. 12 displays what happens when a new reference is given, such that the nearest equilibrium is $q_{0e} = -1$.

Our solution is capable of tackling not only the unwinding phenomenon, but also other problems which may stem from the quaternion space. One common issue is unstable equilibria, which are present in continuous controllers, as in [43]. They cause the controller to be much slower near the unstable equilibrium point, as can be seen in Fig. 13. Discontinuous solutions such as [26], on the other hand, are prone to chattering in the presence of noise, as shown in [3]. Furthermore, to the best of our knowledge, our work is the first attitude controller using neural networks and event-triggered control, which is able to show semi-global convergence and stability results as in Theorem 4.5.

6. Conclusions

In this paper, we presented a novel adaptive attitude controller that fuses event-triggered control, adaptive neural networks, and hybrid systems theory. Firstly, we employed the backstepping technique together with a neural network to construct an adaptive controller that can handle any state-dependent disturbances or modeling errors, while ensuring the tracking error can be made arbitrarily small through appropriate parameter tuning. Afterwards, an event-triggered mechanism was designed by implementing a sample-and-hold of the state signal, rendering a precompact neighborhood of the null error set semi-globally asymptotically stable. To the best of our knowledge, our solution differs from the current state of the art, since it does not have any unstable equilibrium points and does not suffer from the unwinding or chattering phenomena. Simulations were presented in order to show both the adaptation capability and the performance of the event-triggered controller. Beyond spacecraft, the proposed architecture is directly applicable to platforms in which actuator updates carry a mechanical or energetic cost, such as reaction wheel assemblies, and underwater vehicles operating under limited communication bandwidth. In the future, this controller may be extended to deal with actuator faults or underactuated control.

Acknowledgements

This work was supported in part by the University of Macau, Macau, China, under Project SRG2024-00012-FST.

References

[1] R. Kristiansen and P. J. Nicklasson, "Satellite attitude control by quaternion-based backstepping," in *Proceedings of the 2005, American Control Conference, 2005.*, pp. 907–912, IEEE, 2005.

[2] P. M. Tiwari, S. Janardhanan, and M. un Nabi, "Spacecraft anti-unwinding attitude control using second-order sliding mode," *Asian Journal of Control*, vol. 20, no. 1, pp. 455–468, 2018.

[3] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel, "Quaternion-based hybrid control for robust global attitude tracking," *IEEE Transactions on Automatic control*, vol. 56, no. 11, pp. 2555–2566, 2011.

[4] R. Schlanbusch, A. Loria, and P. J. Nicklasson, "On the stability and stabilization of quaternion equilibria of rigid bodies," *Automatica*, vol. 48, no. 12, pp. 3135–3141, 2012.

[5] P. Casau, J. Reis, and C. Silvestre, "Event-triggered trajectory tracking control of an underactuated autonomous surface vessel," in *2022 American Control Conference (ACC)*, pp. 1757–1762, IEEE, 2022.

[6] A. Seuret and C. Prieur, "Event-triggered sampling algorithms based on a lyapunov function," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pp. 6128–6133, IEEE, 2011.

[7] T.-H. Wu, E. Kaufman, and T. Lee, "Globally asymptotically stable attitude observer on $so(3)$," in *2015 54th IEEE Conference on Decision and Control (CDC)*, pp. 2164–2168, IEEE, 2015.

[8] S. Berkane, A. Abdessameud, and A. Tayebi, "Hybrid attitude and gyro-bias observer design on $so(3)$," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 6044–6050, 2017.

[9] T. Lee, "Robust global exponential attitude tracking controls on $so(3)$," in *2013 American Control Conference*, pp. 2103–2108, IEEE, 2013.

[10] B. Wu, Q. Shen, and X. Cao, "Event-triggered attitude control of spacecraft," *Advances in Space Research*, vol. 61, no. 3, pp. 927–934, 2018.

[11] B. Wu, "Spacecraft attitude control with input quantization," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 1, pp. 176–181, 2016.

[12] J. Zhou and C. Wen, *Adaptive backstepping control of uncertain systems: Nonsmooth nonlinearities, interactions or time-variations.* Springer, 2008.

[13] M. Bisheban and T. Lee, "Geometric adaptive control with neural networks for a quadrotor in wind fields," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 4, pp. 1533–1548, 2020.

[14] K. Nouri, R. Dhaouadi, and N. B. Braiek, "Adaptive control of a nonlinear DC motor drive using recurrent neural networks," *Applied Soft Computing*, vol. 8, no. 1, pp. 371–382, 2008.

[15] G. Yu, J. Reis, and C. Silvestre, "Quadrotor neural network adaptive control: Design and experimental validation," *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2574–2581, 2023.

[16] O. S. Patil, D. M. Le, M. L. Greene, and W. E. Dixon, "Lyapunov-derived control and adaptive update laws for inner and outer layer weights of a deep neural network," *IEEE Control Systems Letters*, vol. 6, pp. 1855–1860, 2021.

[17] S. Guo, J. Li, and G.-H. Yang, "Deep neural network-based adaptive supervisory control for strict-feedback nonlinear systems with sensor and actuator faults," *Applied Mathematics and Computation*, vol. 511, p. 129745, 2026.

[18] W. MacKunis, F. Leve, P. Patre, N. Fitz-Coy, and W. Dixon, "Adaptive neural network-based satellite attitude control in the presence of CMG uncertainty," *Aerospace Science and Technology*, vol. 54, pp. 218–228, 2016.

[19] H. Leeghim, Y. Choi, and H. Bang, "Adaptive attitude control of spacecraft using neural networks," *Acta Astronautica*, vol. 64, no. 7-8, pp. 778–786, 2009.

[20] A.-M. Zou, K. D. Kumar, and Z.-G. Hou, "Quaternion-based adaptive output feedback attitude control of spacecraft using chebyshev neural networks," *IEEE transactions on neural networks*, vol. 21, no. 9, pp. 1457–1471, 2010.

[21] X. Shao, Q. Hu, D. Li, Y. Shi, and B. Yi, "Composite adaptive control for anti-unwinding attitude maneuvers: An exponential stability result without persistent excitation," *IEEE Transactions on aerospace and electronic systems*, vol. 59, no. 2, pp. 1051–1066, 2022.

[22] S. P. Bhat and D. S. Bernstein, "A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon," *Systems & Control Letters*, vol. 39, no. 1, pp. 63–70, 2000.

[23] Y. Liu, B. Jiang, J. Lu, J. Cao, and G. Lu, "Event-triggered sliding mode control for attitude stabilization of a rigid spacecraft," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 9, pp. 3290–3299, 2018.

[24] S. Cao, L. Guo, and Z. Ding, "Event-triggered anti-disturbance attitude control for rigid spacecrafts with multiple disturbances," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 2, pp. 344–357, 2021.

[25] C. Tan, G. Xu, L. Dong, H. Zhao, J. Li, and S. Zhang, "Neural network-based finite-time fault-tolerant control for spacecraft without unwinding," *International Journal of Aerospace Engineering*, vol. 2021, pp. 1–10, 2021.

[26] M. N. Hasan, M. Haris, and S. Qin, "Finite-time active fault-tolerant attitude control for flexible spacecraft with vibration suppression and anti-unwinding," *Advances in Space Research*, vol. 71, no. 9, pp. 3644–3660, 2023.

[27] S. M. Amrr, A. Banerjee, and M. Nabi, "Robust attitude control of rigid spacecraft based on event-triggered approach with anti-unwinding," in *2019 Fifth Indian Control Conference (ICC)*, pp. 510–515, IEEE, 2019.

[28] X.-Z. Zhu, P. Casau, and C. Silvestre, "Event-triggered global trajectory tracking control of a quadrotor: Synthesis, simulations, and experiments," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 13, pp. 6144–6165, 2021.

[29] W. Liu, Y. Geng, B. Wu, and D. Wang, "Neural-network-based adaptive event-triggered control for spacecraft attitude tracking," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 10, pp. 4015–4024, 2019.

[30] H. Xie, B. Wu, and W. Liu, "Adaptive neural network model-based event-triggered attitude tracking control for spacecraft," *International Journal of Control, Automation and Systems*, vol. 19, no. 1, pp. 172–185, 2021.

[31] H. Zheng and Y. Tian, "Exponential stability of time-delay systems with highly nonlinear impulses involving delays," *Mathematical Modelling and Control*, vol. 5, no. 1, p. 103, 2025.

[32] R. Kumar, S. Das, and Y. Cao, "Effects of infinite occurrence of hybrid impulses with quasi-synchronization of parameter mismatched neural networks," *Neural Networks*, vol. 122, pp. 106–116, 2020.

[33] S. Shanmugam, R. Vadivel, S. Sabarathinam, P. Hammachukiattikul, and N. Gunasekaran, "Enhancing synchronization criteria for fractional-order chaotic neural networks via intermittent control: An extended dissipativity approach," *Mathematical Modelling and Control*, vol. 5, no. 1, p. 31, 2025.

[34] Y. Cao, K. Udhayakumar, K. P. Veerakumari, and R. Rakkiyappan, "Memory sampled data control for switched-type neural networks and its application in image secure communications," *Mathematics and Computers in Simulation*, vol. 201, pp. 564–587, 2022.

[35] J. P. Silvestre, P. Casau, and P. Oliveira, "Adaptive hybrid global attitude tracking controller," *Nonlinear Analysis: Hybrid Systems*, vol. 55, p. 101555, 2025.

[36] Y. Yang, *Spacecraft Modeling, Attitude Determination, and Control Quaternion-based Approach.* CRC Press, 02 2019.

[37] R. Goebel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems," in *Hybrid dynamical systems*, Princeton University Press, 2012.

[38] R. G. Sanfelice, "Hybrid feedback control," in *Hybrid Feedback Control*, Princeton University Press, 2020.

[39] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural computation*, vol. 3, no. 2, pp. 246–257, 1991.

[40] M. M. Polycarpou, "Stable adaptive neural control scheme for nonlinear systems," *IEEE Transactions on Automatic control*, vol. 41, no. 3, pp. 447–451, 1996.

[41] R. K. Sundaram, *A first course in optimization theory.* Cambridge university press, 1996.

[42] R. G. Sanfelice, R. Goebel, and A. R. Teel, "Invariance principles for hybrid systems with connections to detectability and asymptotic stability," *IEEE Transactions on Automatic Control*, vol. 52, no. 12, pp. 2282–2297, 2007.

[43] C. Zhang, M.-Z. Dai, J. Wu, B. Xiao, B. Li, and M. Wang, "Neural-networks and event-based fault-tolerant control for spacecraft attitude stabilization," *Aerospace Science and Technology*, vol. 114, p. 106746, 2021.