Event-Triggered Control for Modular Aerial Vehicles

Miguel Ângelo Gomes Esteves da Costa
miguel.a.costa@tecnico.ulisboa.pt
Instituto Superior Técnico, Universidade de Lisboa

This thesis addresses the limitations of small-sized Unmanned Aerial Vehicles (UAVs) by proposing a modular approach, where multiple UAV modules form a single vehicle. This method aims to overcome the constraints of individual UAVs. The thesis focuses on proving the theoretical feasibility of this modular vehicle concept. It employs a control strategy involving full state feedback control within an event-triggered methodology, treating the system as an open hybrid automaton. The study uses a coaxial helicopter as the main module type and incorporates Event-Triggered Control and Linear Quadratic Regulator controller logic. The simulations demonstrate promising results, particularly in a scenario with two modules. The modular vehicle concept is envisioned to be scalable, potentially broadening its applications across various domains. This thesis not only enhances the versatility of UAVs but also depicts the Event-Triggered Control paradigm for designing and controlling autonomous modular vehicles, in an effort to broaden their capabilities for tasks previously constrained by their size and load-bearing capacities.

***Index Terms*—Hybrid control systems, event-triggered control, linear quadratic regulator, open hybrid automata.**

I. Introduction

In today's world, every industry faces the imperative of optimizing production costs and speed. The pressure to produce more, at an accelerated pace, has driven industries to adapt. The industrial revolution and technological advances have changed the way industries operate. Automation and efficiency have increased with the advent of robotics and fully automated Unmanned Vehicle Systems (UVSs) (1). These systems offer advantages over manned vehicles, including lower fuel costs, reduced hardware requirements, and safer operation in hostile environments.

In system engineering, devising an optimal control strategy is crucial. Control guides and regulates system behavior towards desired outcomes. Each algorithm has its strengths and weaknesses, which can complicate the selection process. Event-Triggered Control (ETC) computes inputs or actuator signals only at predefined events. These events can be triggered by factors such as time limits or deviations from the desired objective. This approach reduces computational workload, making it particularly desirable for complex plants with intricate control algorithms. ETC proves especially useful in limiting sensor and control computations to necessary instances, leading to efficient energy consumption and reducing the burden on computation-intensive controllers (2).

The calculation of the adequate input signals at event times is defined by the control strategy adopted. Many strategies can be used when controlling Unmanned Aerial Vehicles (UAVs), with Linear Quadratic Regulator (LQR) being one of the most common and versatile approaches (3). It relies on defining a quadratic cost function in an optimization problem to compute state and input values that minimize this function. Adjusting the cost function parameters prioritizes different aspects, influencing the feedback gain and system response.

The concept of modular aerial vehicles involves multiple independent flying modules that can operate individually or combine to form a unified vehicle. These modules can be reconfigured to suit various task requirements. The key advantage of a modular system over a generic UAV is its versatility. Each module can specialize in one aspect, enabling flexibility over thrust power and sensory capabilities based on module configuration. While this represents progress and a potential solution to UAV challenges, achieving full autonomy remains a significant endeavor (4).

A. Motivation

While UAVs offer distinct advantages over human-piloted vehicles, their small frames necessitate compact and lightweight electronic components. This can lead to increased costs due to the need for specialized hardware. In scenarios requiring significant force, UAVs may not be the most suitable option. This limitation has spurred the development of a bundled approach, where multiple UAVs function as a unified entity, providing enhanced capabilities for tasks like relocating hefty objects.

Modular UAVs allow for the integration of specialized sensors and actuators on different units, each tailored to specific subtasks. This approach optimizes resource allocation, as vehicles with varying capabilities can be deployed based on the task at hand. Besides physical constraints, UAVs face computational challenges, especially in fully automated operations. This demands robust control algorithms and substantial computing capacity. Various methodologies have been explored to address this, recognizing the complexity of controlling such systems.

B. An Introduction to ETC

Real-time control approaches assume the controller has access to the entire state space at any given moment, allowing for the computation of the control input in real time. However, in systems like modular UAV control, it's crucial to minimize computation and communication between components.

The Event-Triggered Control (ETC) paradigm reduces communication and computation time by assigning the calculations of the control responses at certain events. These events can depend for example on time of execution and/or state space error. ETC finds practical application in various modern systems, from simple tasks like room temperature control (5, Examle 1.9) to complex endeavors like autonomous drone navigation (6). In the case of temperature control, an event is defined (e.g., temperature falling outside the desired range), detected by a sensor (e.g., thermometer), processed by decision logic (e.g., algorithm determining response), and acted upon (e.g., activating the heater). This method is valuable for systems requiring extensive communication and computation, especially those with multimodal aspects, like modular UAV control.

C. State of the Art

The concept of modular vehicles has seen significant advancements, with notable contributions from the University of Pennsylvania GRASP Lab. Their work, introduced in (7), involved the development of the ModQuad, a flying modular structure capable of self-assembling in midair. The results from this work showcased the potential for autonomous reconfiguration in aerial systems. The nature of these modules has some particularities that can be considered as problematic for the robustness of the system. The research group has since endeavored to address these limitations through modifications in the modules' hardware (8), enabling detachment. Additionally, efforts have been made to enhance the software (9) in the event of rotor failure.

In this thesis, we propose a similar objective as the one from the research group at GRASP labs. The main contributions and differences when comparing the work done by GRASP and this thesis are the event-triggered-control approach and the envisioned new modules. For this particular problem, we envisioned using a coaxial helicopter (10), created by the same research group GRASP Labs, that emulates full actuation over forces and torques (six degrees of freedom) using only two actuators. The coaxial helicopter module addresses many of the difficulties encountered with the modules presented by GRASP. Specifically, the ModQuad employs quadrotors, which can make detachment challenging, and due to the multiple rotors, they are more susceptible to rotor failure. Both of these disadvantages can be mitigated by the proposed module. The attachment mechanism of the ModQuad, which utilizes magnets, was also examined, and it was envisioned that hook and loop connectors would better address the attachment issue, resulting in a more versatile connection (11).

D. Objectives

This thesis addresses the challenge of controlling a swarm of UAVs to function as a multimodule vehicle. We focus on simulating the control of two vehicles, studying their attachment, detachment, and task-following. Our goal is to develop a reliable control strategy that can withstand disturbances and errors while maintaining computational efficiency. Key considerations include selecting the appropriate control algorithm and designing an algorithm that balances performance and computational resources. Section IV showcases the implemented system's capability to mimic the dynamics of multiple modules in various configurations.

We adopt an LQR-based controller, complemented by an event-triggered approach for calculating trajectories and input signals. resulting in an Event-Triggered Linear Quadratic Controller. It recalculates control values if the system deviates significantly from the objective state, or a predetermined time limit is reached. By modeling the system as a hybrid automaton, we enable multiple modes of operation, including full connected modular vehicle and separated modules control.

II. ETC of Hybrid Automata

A hybrid dynamical system (or hybrid system) is a dynamical system that exhibits properties of both continuous and discrete systems. The framework of a hybrid system adopted in this thesis, follows the one presented in (5) as,

$$\begin{cases} x \in C & \dot{x} \in F(x) \\ x \in D & x^+ \in G(x) \end{cases} \tag{1}$$

This representation shows that the state of the system, $x \in \mathbb{R}^n$ can change according to the differential inclusion $\dot{x} \in F(x)$ while in the set C, and change according to a difference inclusion $x^+ \in G(x)$ while in the set D. For the sake of simplifying the terminology used in the upcoming sections, we label the behavior of a dynamical system describable by a differential equation or inclusion as "flow", and the behavior describable by a difference equation or inclusion as "jumps". Taking this nomenclature into consideration, the following names are given:

- A set $C \subset \mathbb{R}^n$, called the flow set;
- A set-valued mapping $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ with $C \subset \operatorname{dom} F$, called the flow map;
- A set $D \subset \mathbb{R}^n$, called the jump set;
- A set-valued mapping $G : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ with $D \subset \operatorname{dom} G$, called the jump map.

Finally, the defined system with its data as above is represented by the notation $\mathcal{H} = (C, F, D, G)$ or, briefly, by $\mathcal{H}$.

According to (5, Assumption 6.5) we name the following set of requirements as the hybrid basic conditions:

(A1) The sets C and D are closed;
(A2) $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is outer semicontinuous and locally bounded relative to C, $C \subset \operatorname{dom} F$, and $F(x)$ is convex for every $x \in C$;
(A3) $G : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is outer semicontinuous and locally bounded relative to D, $D \subset \operatorname{dom} G$.

A. Open Hybrid Automata

The Hybrid Automata (5, Section 1.4.2), can be used to model closed-loop systems, but they do not provide a suitable hybrid automata model for plants, because these do not encompass their inputs and outputs. In order to tackle this limitation, we resort to the concept of Open Hybrid Automaton (OHA) (12, Section 3.13.1). Similarly to the standard Hybrid Automaton, the data of an OHA also includes a directed graph $G = (Q, E)$ that describes possible mode transitions. However, the dynamics and the domain, guard and reset maps of the open hybrid automaton are modified as follows in order to encompass the influence of an input $u \in \mathbb{R}^k$:

$$\begin{aligned} f^\circ &: Q \times \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}^n \\ \mathrm{Domain}^\circ &: Q \rightrightarrows \mathbb{R}^n \times \mathbb{R}^k \\ \mathrm{Guard}^\circ &: E \rightrightarrows \mathbb{R}^n \times \mathbb{R}^k \\ \mathrm{Reset}^\circ &: E \times \mathbb{R}^n \times \mathbb{R}^k \rightrightarrows \mathbb{R}^n \times \mathbb{R}^k, \end{aligned} \tag{2}$$

where the superscript in f°, Domain°, Guard° and Reset° emphasizes the fact that this data characterizes an OHA. Lastly, the data of an OHA also includes a function $h : Q \times \mathbb{R}^n \to \mathbb{R}^m$ such that the output of the system is given by $y = h(q, x)$ for each $(q, x) \in Q \times \mathbb{R}^n$.

An Open Hybrid Automaton can be described as a hybrid dynamical system with inputs (cf. (13)) as,

$$\begin{aligned} &\begin{cases} \dot{q} = 0 \\ \dot{x} = f^\circ(q, x, u) \end{cases} && (q, x, u) \in C^\circ \\ &\begin{pmatrix} q^+ \\ x^+ \end{pmatrix} \in G^\circ(q, x, u) && (q, x, u) \in D^\circ \end{aligned} \tag{3}$$

where the jump map, $G^\circ(q, x, u)$, is

$$\bigcup_{\{q' \in Q : (x,u) \in \mathrm{Guard}^\circ(q,q')\}} \begin{pmatrix} q' \\ \mathrm{Reset}^\circ(q, q', x, u) \end{pmatrix}, \tag{4}$$

the flow set, C° is defined as,

$$\{(q, x, u) \in Q \times \mathbb{R}^n \times \mathbb{R}^k : (x, u) \in \mathrm{Domain}^\circ(q)\}, \tag{5}$$

and for some edge, $(q, q') \in E$, the jump map, D° is,

$$\bigcup_{q \in Q} \left(\{q\} \times \{(x, u) \in \mathbb{R}^n \times \mathbb{R}^k : (x, u) \in \mathrm{Guard}^\circ(q, q')\right). \tag{6}$$

With all being said, an OHA is identified by its data, as:

$$\mathcal{H}_a^\circ := (Q, E, f^\circ, \mathrm{Domain}^\circ, \mathrm{Guard}^\circ, \mathrm{Reset}^\circ, h),$$

and, for the sake of regularity, we take Assumption 1.

Assumption 1. *Given an Open Hybrid Automaton $\mathcal{H}_a^\circ := (Q, E, f^\circ, \mathrm{Domain}^\circ, \mathrm{Guard}^\circ, \mathrm{Reset}^\circ, h)$, the following hold:*

1) $\mathrm{Domain}^\circ(q)$ *is closed for each* $q \in Q$;
2) f° *is continuous;*
3) $\mathrm{Guard}^\circ(q, q')$ *is closed for each* $(q, q') \in E$;
4) Reset° *is outer semicontinuous and locally bounded relative to* D°;

Remark 1. *The data of an open hybrid automaton in (12, Section 3.13.1) also includes the set of initial states which we have omitted because we consider it to be the whole state space $Q \times \mathbb{R}^n$.*

The following definition of the execution of an open hybrid automaton is adapted from (12, Definition 3.16).

Definition 1. *An execution of an open hybrid automaton with initial condition $(q_0, x_0) \in Q \times \mathbb{R}^n$ is a tuple $\chi := (\mathcal{T}, q, x, u, y)$ satisfying the following:*

1) $(t, j) \mapsto (q, x, u)(t, j)$ *is a solution to* (3) *in the sense of (13, Definition 2.29);*
2) $\mathcal{T} := \{t_j\}_{0 \le j \le J}$ *with* $J := \sup_j \mathrm{dom}(q, x, u)$ *is the collection of jump times, i.e.,* $(t_{j-1}, j-1), (t_{j-1}, j) \in \mathrm{dom}(q, x, u)$ *for each* $j \in \{1, 2, \ldots, J\}$;
3) $y(t, j) = h((q, x)(t, j))$ *for each* $(t, j) \in \mathrm{dom}(q, x, u)$.

We say that an execution χ of an open hybrid automaton with initial condition $(q_0, x_0) \in Q \times \mathbb{R}^n$ is maximal if $(t, j) \mapsto (q, x, u)(t, j)$ is a maximal solution (5, Definition 2.7) to (3). *If $(t, j) \mapsto (q, x, u)(t, j)$ is complete (5, Definition 2.5), then we also say that χ is complete.*

B. Event-Triggered Control of a OHA

The assignment of expected input and trajectory values ($x^\star$ and $u^\star$) can be modeled as a discrete component which in itself has a jump map and jump set, depicting the ETC dynamics, and a new flow set, C that takes the previous set (C°) from (5).

Assumption 2. *Given an open hybrid automaton*

$$\mathcal{H}_a^\circ := (Q, E, f^\circ, \mathrm{Domain}^\circ, \mathrm{Guard}^\circ, \mathrm{Reset}^\circ, h)$$

with $h(q, x) = (q, x)$ for each $(q, x) \in Q \times \mathbb{R}^n$, there exists a unique complete (5, Definition 2.5) execution χ to $\mathcal{H}_a^\circ$.

Given an open hybrid automaton satisfying Assumption 2, one may represent the state and input trajectories of a particular execution (5, Definition 2.6) as functions not only of hybrid time (5, Definition 2.3), but also of the initial conditions (q_0, x_0) as follows:

$$t \mapsto \begin{pmatrix} x^\star(t, j; q_0, x_0) \\ u^\star(t, j; q_0, x_0) \end{pmatrix} \qquad \forall (t, j) \in \mathrm{dom}(q^\star, x^\star, u^\star)$$

due to the uniqueness of complete executions to $\mathcal{H}_a^\circ$. Under the same assumption, it follows that

$$\begin{aligned} x^\star(t, j; q_0, x_0) &= x^\star(t - t_j, 0; q(t_j, j), x(t_j, j)) \\ u^\star(t, j; q_0, x_0) &= u^\star(t - t_j, 0; q(t_j, j), x(t_j, j)) \end{aligned}$$

for each $t \in I^j := \{t \ge t_j : (t, j) \in \mathrm{dom}(q, x, u)\}$ and each $j \in \{1, 2, \ldots, J\}$. Therefore, in order to follow an execution it is not necessary to keep track of the number of jumps, but merely the time since the last jump $\tau = t - t_j$ and the state of the hybrid automaton at that point $(q, x)(t_j, j)$. For this reason, and with a slight abuse of notation, we represent a segment of desired trajectories ($x^\star$) and desired input ($u^\star$) of a particular execution with initial condition $(q, x) \in Q \times \mathbb{R}^n$ as follows:

$$\tau \mapsto \begin{pmatrix} x^\star(\tau; q, x) \\ u^\star(\tau; q, x) \end{pmatrix}$$

for each $\tau \in [0, T]$ following the convention adopted in (5) where $T \in \mathbb{R}_{\ge 0}$ is the time of the jump.

The controller design aims to minimize transmissions between the controller and actuator by sending the entire control signal $u^\star$ to the actuator during initialization and for two scenarios: when the distance between the desired state trajectory $x^\star$ and the current state exceeds a threshold δ; or when a timer variable, tracking time since the last hybrid automaton reset, reaches its timeout.

More formally, the state of the hybrid closed-loop system $\mathcal{H} := (C, F, D, G)$ which implements the logic described above is denoted by $\xi := (q, x, \hat{x}, \tau) \in \Xi := Q \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{\geq 0}$, where $(q, x) \in Q \times \mathbb{R}^n$ denotes the operating mode and the state of the hybrid automaton $\mathcal{H}_a^\circ$, $\hat{x} \in \mathbb{R}^n$ is a memory variable that stores the value of the state x at jumps, and $\tau \in \mathbb{R}_{\geq 0}$ is the timer variable.

The flows of the hybrid closed-loop system are described as follows:

$$\dot{\xi} = f(\xi) := \begin{pmatrix} 0 \\ f^\circ(q, x, u^\star(\tau; q, \hat{x}) \\ 0 \\ 1 \end{pmatrix} \qquad \xi \in C \tag{7}$$

where

$$\begin{aligned} C := \{\xi \in \Xi : &\tau \in [0, T(q, \hat{x})], \\ &|x - x^\star(\tau; q, \hat{x})| \leq \delta, \\ &(q, x, u^\star(\tau; q, \hat{x})) \in C^\circ\}, \end{aligned} \tag{8}$$

with continuous $T(q, \hat{x})$. As can be verified in (7), the system is allowed to flow according to the dynamics specified by f as long as the state ξ belongs to C, i.e., if the timer τ belongs to $\tau \in [0, T(q, \hat{x})]$, the distance between the current value of the state $x \in \mathbb{R}^n$ and the trajectory $\tau \mapsto x^\star(\tau; q, \hat{x})$ does not exceed the parameter $\delta > 0$ and the combination of the values of the state and of the current input belong to the domain of the current operating mode, as specified in the given open hybrid automaton $\mathcal{H}_a^\circ$. The variable $\hat{x} \in \mathbb{R}^n$ stores the value of the state $x \in \mathbb{R}^n$ at the start of the trajectory $x^\star$. More precisely, $\hat{x}$ should be equal to $x^\star(0; q, \hat{x})$ save for poor initialization of the controller variables. The state variable $x \in \mathbb{R}^n$ evolves during flows according to the continuous dynamics specified in $\mathcal{H}_a^\circ$ considering that the input signal is $\tau \mapsto u^\star(\tau; q, \hat{x})$.

The jumps of the hybrid closed-loop system consist of the union of two distinct jump maps and jump sets as follows,

$$\xi^+ \in G(\xi) := G_1(\xi) \cup G_2(\xi) \quad \xi \in D := D_1 \cup D_2. \tag{9}$$

with the underlying assumption that $G_i(\xi) = \emptyset$ for each $\xi \notin D_i$ and each $i \in \{1, 2\}$. In the sequel, we describe each component of the jump map and jump set in detail.

The pair (G_1, D_1) represents the event-triggered mechanism and $\forall \xi \in D_1$,

$$G_1(\xi) := \{(g_q, g, \hat{g}, g_\tau) \in \Xi : g_q = q,\ g = \hat{g} = x,\ g_\tau = 0\} \tag{10}$$

where

$$D_1 := \{\xi \in \Xi : \tau \geq T(q, \hat{x}) \text{ or } |x - x^\star(\tau; q, \hat{x})| \geq \delta\}. \tag{11}$$

When either the timer τ exceed the limit set by $T(q, \hat{x})$ or when the state of the system deviates by an amount δ with respect to $x^\star$, the jump dynamics described by (10) do the following:

- Update the state of the memory variable $\hat{x} \in \mathbb{R}^n$ to match the current value of the state $x \in \mathbb{R}^n$;
- Keep the current operating mode $q \in Q$ unchanged;
- Reset the timer $\tau \in \mathbb{R}_{\geq 0}$ to 0.

These updates trigger the computation of a new execution $\chi = (\mathcal{T}, q^\star, x^\star, u^\star, y)$ of $\mathcal{H}_a^\circ$ with initial condition (q, x).

The pair (G_2, D_2) represents the jump dynamics of the open hybrid automaton $\mathcal{H}_a^\circ$. This refers to the jump dynamics introduced in Section II-A, which, in other words, means mode switching, and they are given by:

$$\begin{aligned} G_2(\xi) := \{&(g_q, g, \hat{g}, g_\tau) \in \Xi : \\ &(g_q, g) \in G^\circ(q, x, u^\star(\tau; q, \hat{x})), \\ &\hat{g} = g,\ g_\tau = 0\}, \end{aligned} \tag{12}$$

for each $\xi \in D_2 := \{\xi \in \Xi : (q, x, u^\star(\tau; q, \hat{x})) \in D^\circ\}$, where D° is given by (6).

Before delving into the theorem and the corresponding proof of pre-asymptotic stability of a hybrid closed loop, we must define the concept of uniformly pre-attractiveness, strongly forward pre-invariance and stability.

A compact set $\mathcal{A} \subset \mathbb{R}^n$ is said to be uniformly pre-attractive from a set $S \subset \mathbb{R}^n$ if every $\phi \in S_\mathcal{H}(S)$ is bounded and for every $\epsilon > 0$ there exists $\tau > 0$ such that $|\ \phi(t, j)\ |_\mathcal{A} \leq \epsilon$ for every $\phi \in S_\mathcal{H}(S)$ and $(t, j) \in \operatorname{dom} \phi$ with $t + j \geq \tau$ (5, Definition 6.24).

Let us call a set, $S \subset \mathbb{R}^n$, strongly forward pre-invariant if for every $\phi \in S_\mathcal{H}(S)$, rge $\phi \subset S$ (5, Definition 6.25).

Theorem 1. *If Assumptions 1 and 2 hold, and there exists a compact set $\mathcal{X} \subset \mathbb{R}^n$ such that*

$$\mathcal{A} := \{\xi \in \Xi : \hat{x} \in \mathcal{X},\ \tau \leq T(q, \hat{x}),\ x = x^\star(\tau; q, \hat{x})\}$$

is strongly forward pre-invariant and uniformly pre-attractive for $\mathcal{H}$, then $\mathcal{A}$ is pre-asymptotically stable for $\mathcal{H}$.

Proof.

- $\mathcal{H}$ is nominally well-posed, since it satisfies the hybrid basic conditions;
  It follows from Assumption 2 and the definition of an execution of an open hybrid automaton that $x^\star$ is continuous. It follows from Assumption 1 that C° is closed, and, since the Euclidean norm and T are continuous, then the flow set C in (8) is closed. Using similar arguments, it is also possible to prove that D° is closed due to Assumptions 2 and 1. It follows from continuity of f° in Assumption 1 and the continuity of $u^\star$ that the flow map (7) is outer semicontinuous, locally bounded and a singleton, hence convex-valued. The map G_1 in (10) is continuous, hence outer semicontinuous and locally bounded. It follows from Assumption 1 that G_2 in (12) is outer semicontinuous and locally bounded relative to D_2, hence we conclude that G is outer semicontinuous and locally bounded relative to D.
- It follows from continuity of $x^\star$ and T that $\mathcal{A}$ is compact;
  To prove that $\mathcal{A}$ is compact, we need to prove that it is both closed and bounded. Closedness of $\mathcal{A}$ follows from the continuity if T and $x^\star$. To see this, let $\gamma(\xi) := (T(q, \hat{x}) - \tau, x - x^\star(\tau; q, \hat{x}))$. Then, $\mathcal{A}$ can be rewritten as

  $$\mathcal{A} = \{\xi \in \Xi : \hat{x} \in \mathcal{X}, \gamma^{-1}([0, +\infty) \times \{0\})\}$$

  and $\gamma^{-1}([0, +\infty) \times \{0\})$ is the preimage of a closed set through a continuous map, hence it is closed (14, Lemma 2.7). Compactness of $\mathcal{X}$ implies that $\mathcal{X}$ is closed, hence $\mathcal{A}$ is the intersection of two closed sets, thus it is also closed. To prove that $\mathcal{A}$ is bounded, notice that Q is a finite discrete set, thus it is also compact. It follows from compactness of $\mathcal{X}$ and continuity of T that $T(Q \times \mathcal{X})$ is compact, thus $\tau < +\infty$ for each $\xi \in \mathcal{A}$. In particular, we have that $\mathcal{S} := \{(q, \hat{x}, \tau) \in Q \times \mathcal{X} \times \mathbb{R}_{\geq 0} : \tau \leq T(q, \hat{x})\}$ is compact. Since $x^\star$ is continuous, $x^\star(\mathcal{S})$ is compact. Finally, since $x \in x^\star(\mathcal{S})$ for each $\xi \in \Xi$, we conclude that $\mathcal{A}$ is bounded, which concludes the proof that it is compact.

- From the assumptions in the theorem's statement, we prove pre-asymptotic stability of $\mathcal{A}$ using (5, Proposition 7.5).

If each maximal solution (5, Definition 2.7) to $\mathcal{H}$ is complete, then solutions to $\mathcal{H}$ inherit the properties of the executions of the open hybrid automaton $\mathcal{H}_a^\circ$. If we construct an algorithm that finds complete bounded executions $\chi^\star = (\mathcal{T}, q^\star, x^\star, u^\star, y)$ to $\mathcal{H}_a^\circ$ such that

$$|(q^\star, x^\star)(t, j)|_{\mathcal{A}^\circ} \leq \beta((q^\star, x^\star)(0, 0), t + j)$$

for some class-$\mathcal{KL}$ function (5, Definition 3.38) β, then each maximal solution to $\mathcal{H}$ from $\mathcal{A}$ satisfies

$$|(q, x)(t, j)|_{\mathcal{A}^\circ} \leq \beta((q, x)(0, 0), t + j).$$

III. ETC of Modular Aerial Vehicles

The general model of the state space representation of a standard UAV (15) incorporates the concept of multiple reference frames and six degrees of motion, resulting in an overly complex state space system. In order to focus more effectively on the control problem at hand, it is in our best interest to propose a simplified version of the state space system.

A. Two Dimensional Simplification

To simplify the modeling of the vehicles at hand, certain assumptions were made, including:

- Aerodynamic drag forces are negligible;
- All vehicle modules are equivalent, having the same properties such as mass;
- Other external disturbances are neglected;
- The modules are considered to have three degrees of freedom.

The system is simplified by considering the z-axis as fixed, assuming a closed-loop control for constant altitude. Additionally, with constrained roll and pitch motions, the system is effectively treated as two-dimensional, focusing solely on angular motion in the yaw direction within the XY plane. This simplified state space representation is defined solely in the inertial frame, where angular positions and velocities are considered unidimensional values. Analyzing the forces and torques on each module provides insight into the system's dynamics.

Newton's second law for motion and rotation of each module define the dynamics of the first module as,

$$m\ddot{p}_1 = F_1 \tag{13}$$

$$J\ddot{\theta}_1 = M_1 \tag{14}$$

where $m \in \mathbb{R}$ and $J \in \mathbb{R}$ are the mass and inertia of each module, $p_1 \in \mathbb{R}^2$ and $\theta_1 \in \mathbb{R}$ the linear and angular acceleration of the first module and finally $F_1 \in \mathbb{R}^2$ and $M_1 \in \mathbb{R}$ the force and torque applied to the same module. The dynamics of the second module are obtained in the same manner, and a state system representation of a two vehicle system can be defined as,

$$\begin{cases} \dot{p}_i = v_i \\ \dot{v}_i = \dfrac{F_i}{m} \\ \dot{\theta}_i = \omega_i \\ \dot{\omega}_i = \dfrac{M_i}{J} \end{cases} \quad \text{with } i \in \{1, 2\} \tag{15}$$

When the modules are attached, the dynamics of the system are defined by the interaction of the modules with each other. Contrary to (15) the dynamics of a module connected to another is not solely dependent on its own inputs, since the actuation and motion of the connected module will interfere.

Figure 1 depicts the system diagram illustrating two interconnected modules. The diagram showcases the contact forces (F_c) acting between the modules, along with the position vectors (p_1 and p_2) representing the positions of each module's center of mass. Additionally, the vectors r_1 and r_2 are illustrated, representing the displacement vectors from the center of mass to the respective contact point.

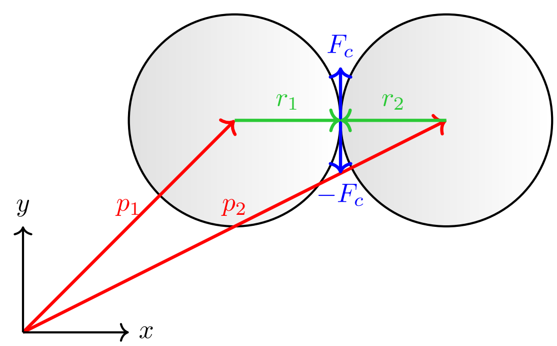

Figure 1: System diagram illustrating the contact force (F_c), position vectors (p_1 and p_2) and displacements from body masses to contact point (r_1 and r_2) of the two interconnected modules.

Just like the system defined in (15), this system is characterized by Newton's second law for both motion and rotation of each module. However, in this case, we have to factor in the contact force (F_c) and its impact on the system.

Generally the notation $\times$ is considered the vector product, and it's not applicable to two-dimensional vectors. With this being said, for the considered two-dimensional simplification, for any vectors a and b in $\mathbb{R}^2$, where $a = \begin{bmatrix} a_1 & a_2 \end{bmatrix}^T$ and $b = \begin{bmatrix} b_1 & b_2 \end{bmatrix}^T$, we introduce the redefined operation expressed as:

$$a \times b = a_1 b_2 - a_2 b_1 \tag{16}$$

With the new $\times$ operation in mind, the differential equations governing the sysem when the modules are attached are,

$$m\ddot{p}_1 = F_1 + F_c \tag{17}$$

$$J\ddot{\theta}_1 = M_1 + r_1 \times F_c \tag{18}$$

where $F_c \in \mathbb{R}^2$ is the contact force. The dynamics of the second module are obtained in the same manner and defined by:

$$m\ddot{p}_2 = F_2 - F_c \tag{19}$$

$$J\ddot{\theta}_2 = M_2 + r_2 \times (-F_c). \tag{20}$$

Let's examine the position in time of the contact point between the two modules, denoted as $p_c(t)$. From Figure 1 it becomes apparent that a path to p_c can be obtained by summing the position vector (p_i) to the displacement vector (r_i) for any module. Equation (21) illustrates the equivalence of these paths:

$$p_c(t) = p_1 + r_1 = p_2 + r_2. \tag{21}$$

To maintain continuous contact between the modules, the contact force F_c must adhere to specific constraints outlined by Equation (21) and its derivatives over time. Thus, the following constraint must be met:

$$\frac{d^2}{dt^2}(p_1 + r_1 - p_2 - r_2) = 0 \tag{22}$$

For the differential equations and constraint equations, it is crucial that all involved parameters are either system inputs, state space variables, or can be expressed in terms of these parameters. Therefore, we must define F_c, r_1 and r_2 as functions of the inputs and state space variables

The displacement vectors r_1 and r_2 are vectors with magnitude R such that, $r_1 = -r_2$. Having said this, and from (21) we can write both vectors as a function of the state space variables p_1 and p_2 as,

$$r_1 = -r_2 = \frac{p_2 - p_1}{2}. \tag{23}$$

It will be useful to define the parameters $\dot{r_1}$ and $\dot{r_2}$ as a function of the state space variables, and with this in mind we consider,

$$\dot{r_i} = S(\omega_i) r_i \tag{24}$$

Where the matrix $S(x)$ can be defined as

$$S(x) = \begin{bmatrix} 0 & -x \\ x & 0 \end{bmatrix}, \tag{25}$$

and the vector $S(\omega_i)r_i$ is a new, perpendicular vector to r_i with magnitude $\omega_i R$.

The constraint equation (22) can be rewritten as,

$$\ddot{p_1} - \ddot{p_2} + S(\dot{\omega}_1)r_1 + S(\omega_1)\dot{r_1} - S(\dot{\omega}_2)r_2 - S(\omega_2)\dot{r_2} = 0 \tag{26}$$

This equation contains parameters that are not state space, nor input variables, and for that reason through extensive reworking of the equation we obtained the force of contact expression,

$$\Lambda = \left(\frac{2}{m} + \frac{\|r_1\|^2}{J} + \frac{\|r_2\|^2}{J}\right) I_2 - \frac{r_1 r_1^T}{J} - \frac{r_2 r_2^T}{J}$$

$$\Psi = \frac{F_2 - F_1}{m} + \frac{S(M_2)r_2 - S(M_1)r_1}{J} + \omega_1^2 r_1 - \omega_2^2 r_2$$

$$F_c = \Lambda^{-1}\Psi \tag{27}$$

Equation (21) establishes the crucial link between module positions and the contact point. Taking the first derivative, we obtain,

$$v_c = \dot{p_c} = v_1 + S(\omega_1)r_1 = v_2 + S(\omega_2)r_2. \tag{28}$$

This equation signifies the required relationship between angular and linear velocities between the individual modules and the fully connected vehicle (operating in mode $q = 1$).

Given the previous deductions and differential equations obtained, the proposed systems of differential equations that govern both modes of operation and define the flow map are formally represented as,

$$f^\circ(q, x, u) = \dot{x} \qquad (q, x, u) \in C^\circ.$$

For the sake of simplicity, and for the rest of this thesis, the state vector elements may be referred to as either x_i or their physical meanings, for each $i \in \{1, 2\}$ module, the positions and velocities (p_i and v_i) and the angular positions and velocities (θ_i and ω_i) such that,

$$p_1 = [x_1\ x_2]^T,\ v_1 = [x_3\ x_4]^T,\ \theta_1 = x_5,\ \omega_1 = x_6,$$
$$p_2 = [x_7\ x_8]^T,\ v_2 = [x_9\ x_{10}]^T,\ \theta_2 = x_{11},\ \omega_2 = x_{12}.$$

Similarly, the control vector elements may be referred to as either u_i or by their physical meanings. For each $i \in \{1, 2\}$ module, the input forces (F_i) and input torques (M_i) are defined such that,

$$F_1 = [u_1\ u_2]^T,\ M_1 = u_3,$$
$$F_2 = [u_4\ u_5]^T,\ M_2 = u_6.$$

From this point forward, this representation will be frequently employed to simplify and improve the understanding of many expressions.

For the mode, $q = 0$, we define the state space system, $f^\circ(0, x, u)$, to be given by (15).

For the mode, $q = 1$, we define the state space system, $f^\circ(1, x, u)$, similarly to the previous mode, yet $\dot{v_i}$ are given by (17) and (19) and similarly, $\dot{\omega_i}$ are given by (18) and (20).

B. Trajectory Generation

For an isolated module, i, the input vector $(u(t))^{\mathbf{i}}$ guides the module by minimizing the difference between the state vector $(x(t))^{\mathbf{i}}$ and the reference signal $(r(t))^{\mathbf{i}}$. For this thesis, we focus on defining the OHA for controlling only two modules. With this being said, we consider,

$$x(t) = \begin{bmatrix} (x(t))^{\mathbf{1}} \\ (x(t))^{\mathbf{2}} \end{bmatrix},\ u(t) = \begin{bmatrix} (u(t))^{\mathbf{1}} \\ (u(t))^{\mathbf{2}} \end{bmatrix},\ r(t) = \begin{bmatrix} (r(t))^{\mathbf{1}} \\ (r(t))^{\mathbf{2}} \end{bmatrix} \tag{29}$$

The notation $[.]^{\mathbf{i}}$ (or $(.)^{\mathbf{i}}$) indicates the values are related to the i-th module.

The concept of trajectory generation relies on the idea of calculating the desired input, $u^\star(\tau)$. By inputting the plant with $u^\star(\tau)$ the modules follow the desired trajectory, $x^\star(\tau)$, for the proceeding T seconds, such that $\tau \in [0;T]$.

$$u^\star(\tau) = \begin{bmatrix} (u^\star(\tau))^{\mathbf{1}} \\ (u^\star(\tau))^{\mathbf{2}} \end{bmatrix}, \qquad x^\star(\tau) = \begin{bmatrix} (x^\star(\tau))^{\mathbf{1}} \\ (x^\star(\tau))^{\mathbf{2}} \end{bmatrix}. \tag{30}$$

The functions $x^\star(\tau)$ and $u^\star(\tau)$ are functions redefined at any instance of event and for that reason are defined only for the values of τ.

The reference signals have the same structure as the state space vector, i.e.

$$(r(t))^i = \begin{bmatrix} p(t) \\ v(t) \\ \theta(t) \\ \omega(t) \end{bmatrix}_{\mathrm{r}}^{\mathbf{i}}. \tag{31}$$

The notation $[.]_{\mathrm{r}}$ (or $(.)_{\mathrm{r}}$) indicates the variables are related to the reference signal.

In theory, we defined the reference signals to be distinct for the instances we desire separated modules (mode $q = 0$) and we set the same reference signals ($(r(t))^1 = (r(t))^2$) for the instances we desire the modules to connect (mode $q = 1$).

The diagrams in Figures 2 and 3 depict precisely the dynamics of each mode in the context of two modules, following their references. The circles describe the position of the modules in time, and the dashed lines represent the position values of the reference signals.

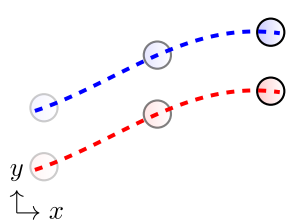

Figure 2: Expected dynamics for mode $q = 0$.

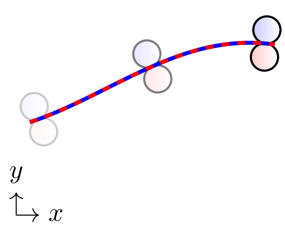

Figure 3: Expected dynamics for mode $q = 1$.

For the mode $q = 0$, we see the modules' center of masses accurately follow their reference positions. For the mode $q = 1$, the two reference signals coincide, and the point accurately following the references is the contact point, i.e., the center of mass of the modular vehicle.

The controller, ETC paradigm, computes and applies control inputs exclusively when certain events (9) occur within the system. Having said this, given a specific event instance t_0, the controller unit takes the reference signal $r(t)$ for $t \in [t_0; t_0 + T]$. Given the reference signal for the desired interval, and the current OHA state $\xi^+ = G(\xi)$ it computes the pair $(x^\star(\tau), u^\star(\tau))$ for $\tau \in [0;T]$ that resolves the initial value problem,

$$\begin{cases} \dot{x}^\star(\tau) = f^\circ(\hat{q}, x^\star(\tau), u^\star(\tau)) \\ x^\star(0) = \hat{x}. \end{cases} \tag{32}$$

With $\hat{x} = x(t_0)$ representing the initial value of $x^\star(\tau)$, used for the calculation of $(x^\star(\tau), u^\star(\tau))$. The initial mode value, $\hat{q}$, defines the current mode of operation and the dynamics of the system to consider ($f^\circ(0,\cdot)$ or $f^\circ(1,\cdot)$). The $u^\star(\tau)$, will determine the input to be applied to the plant for $t \in [t_0; t_0 + T]$, unless a mode switch is triggered (6) or an ETC event occurs (11).

As it was previously stated, the solution to the control problem (and consequently to (32)), is based on an event-triggered Linear Quadratic Regulator (LQR) approach. The feedback gain matrix, $K \in \mathbb{R}^{3\times 6}$, was designed to control a single module, defined by its state space variable $(x)^i \in \mathbb{R}^6$, by calculating an input control vector $(u)^i \in \mathbb{R}^3$. For multiple modules, multiple vectors are calculated and concatenated as shown in (30).

The reference signals are defined *a priori* and constitute one of the arguments required for the calculation of the control vector. For the attached case ($q = 0$), we want to track the center of mass of each module to follow their references. In the case of connected modules ($q = 1$), we wish to have the contact point follow the reference.

The standard LQR control law for calculating the value of the input, u, given a state vector, x taking into consideration the reference signal r, is given by,

$$u = -K\,(x - r) = -K \underbrace{\begin{bmatrix} p - (p)_{\mathrm{r}} \\ v - (v)_{\mathrm{r}} \\ \theta - (\theta)_{\mathrm{r}} \\ \omega - (\omega)_{\mathrm{r}} \end{bmatrix}}_{\mathbf{e}}. \tag{33}$$

The gain matrix is calculated and defined following an LQR methodology. The expression (33) illustrates the relationship between the gain matrix elements and the elements in the error vector $\mathbf{e} \in \mathbb{R}^6$. The elements of $\mathbf{e}$ must be defined to correctly depict the position, velocity, angular and rotation errors, related to a given module (or modular vehicle).

1) Trajectory for Disconnected Modules

For the case of disconnected modules, the solution to (32) stems from the standard control law (33) since the state vector $(x^\star(\tau))^{\mathbf{i}}$ depicts the dynamics of the modules in regard to their center of mass and for that reason we should minimize the error between it and its corresponding reference signal $(r(t_0+\tau))^{\mathbf{i}}$. In this way we obtain for any i-th module,

$$(u^\star(\tau))^{\mathbf{i}} = -K\left[(x^\star(\tau))^{\mathbf{i}} - (r(t_0+\tau))^{\mathbf{i}}\right], \tag{34}$$

2) Trajectory for Connected Modules

The objective behind the attachment of two or more modules is to create a singular, more able vehicle composed by many modules. Considering the expected dynamics depicted by Figure 3, we can conclude that the error vector from (33), must be accurately defined with respect to the full modular vehicle, i.e. the state values to be considered are related to the modular vehicle and not the individual modules. Let us start by defining the error vector in regard to the first module, $(\mathbf{e})^{\mathbf{1}}$.

The position tracking error should depict the deviation between p_c and the reference positioning, $(p)_r^{\mathbf{1}}$.

$$\begin{bmatrix} e_1 \\ e_2 \end{bmatrix}^{\mathbf{1}} = p_c - (p)_r^{\mathbf{1}} \tag{35}$$

The position tracking error should depict the deviation between v_c and the reference positioning, $(v)_r^{\mathbf{1}}$.

$$\begin{bmatrix} e_3 \\ e_4 \end{bmatrix}^{\mathbf{1}} = v_c - (v)_r^{\mathbf{1}} \tag{36}$$

From Figure 4 we establish what we consider should be the orientation, θ of the modular vehicle.

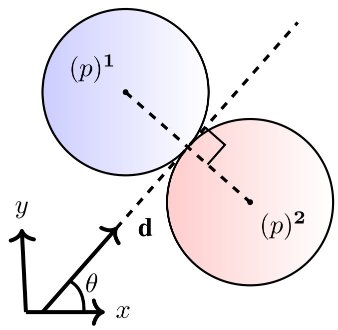

Figure 4: Depiction of the modules for mode $q = 1$. The vector $\mathbf{d}$ is perpendicular to $\left((p)^{\mathbf{2}} - (p)^{\mathbf{1}}\right)$ and the angle θ is the angle $\mathbf{d}$ makes with the x-axis.

Defining the direction vector as, $\mathbf{d} = S(1)\left((p)^{\mathbf{2}} - (p)^{\mathbf{1}}\right)$, we find θ by obtaining the angle between $\mathbf{d}$ and the x-axis. Finally we define the fifth element of $(\mathbf{e})^{\mathbf{1}}$ as,

$$(e_5)^{\mathbf{1}} = \theta - (\theta)_r^{\mathbf{1}}. \tag{37}$$

Note that this difference has to be reframed to the interval $[-\pi, \pi)$.

Given the attachment configuration of the modules, where both modules are connected to a fixed point, then, $(\omega)^{\mathbf{1}} = (\omega)^{\mathbf{2}} = \omega$. With this being said, we obtain,

$$(e_6)^{\mathbf{1}} = \omega - (\omega)_r^{\mathbf{1}}. \tag{38}$$

Having defined the error vector, we can rewrite (33) for any i module in mode $q = 1$, such that:

$$(u^\star(\tau))^{\mathbf{i}} = -K \left(\begin{bmatrix} p_c(\tau) \\ v_c(\tau) \\ \theta(\tau) \\ \omega(\tau) \end{bmatrix} - (r(t_0 + \tau))^{\mathbf{i}} \right) \tag{39}$$

Where (30) becomes the solution to the initial value problem, (32).

From the definition of the desired control input calculation, if we ensure f° is continuous between events, we guarantee continuous desired input $u_j^\star(\tau)$ for any $j \in \{1, 2, ..., J\}$ jump, as found necessary from Theorem 1.

C. Closed-Loop Hybrid System

In the Section II-A, the general definition for an OHA was presented. Now, an overview of the multimodal aspect of the dynamical system will be provided, defining the state's dynamics by

$$\mathcal{H}_a^\circ := (Q, E, f^\circ, \mathrm{Domain}^\circ, \mathrm{Guard}^\circ, \mathrm{Reset}^\circ, h).$$

The two available modes and their transitions are defined by the graph $G = (Q, E)$ where,

$$\begin{aligned} Q &:= \{0, 1\} \\ E &:= \{(0, 1), (1, 0)\}. \end{aligned} \tag{40}$$

The $f^\circ(q, x, u)$, was previously defined at the end of Section III-A describing the flow of both modes of the system.

The $\mathrm{Domain}^\circ(q)$ depicts for any given mode, q, the set of state, x and input, u, values where the state flows. Hence we define it for each mode as,

$$\begin{aligned} \mathrm{Domain}^\circ(0) &= \{(x, u) \in \mathbb{R}^4 \times \mathbb{R}^2 : \|p_1 - p_2\|_2 \geq R\} \\ \mathrm{Domain}^\circ(1) &= \{(x, u) \in \mathbb{R}^4 \times \mathbb{R}^2 : \|p_1 - p_2\|_2 \leq R\} \end{aligned}$$

The $\mathrm{Guard}^\circ(q, q')$ depicts for any given edge, (q, q'), the set of state, x and input, u, values where state jumps from mode q to q'.

Firstly, for the situation defined by the edge $(0, 1)$, the instance of attachment requires direct contact between modules, $(x, u) \in \mathrm{Domain}^\circ(1)$. It must also be found that the vehicles aren't distancing from each other. Defining $d(t) = \|p_1(t) - p_2(t)\|_2$ as the distance in time between the two modules at any given moment t. Non-increasing distance between the two modules implies a non-positive time derivative, $\dot{d}(t)$.

$$\dot{d}(t) \leq 0 \Leftrightarrow (p_1 - p_2) \cdot (v_1 - v_2) \leq 0 \tag{41}$$

This provides the formalization of $\mathrm{Guard}^\circ(0, 1)$, as,

$$\{(x, u) \in \mathrm{Domain}^\circ(1) : (p_1 - p_2) \cdot (v_1 - v_2) \leq 0\}.$$

Moving on, to the situation defined by the edge $(1, 0)$, the instance of detachment requires direct contact between modules, $(x, u) \in \mathrm{Domain}^\circ(1)$. But it must also be found that the forces applied to the modules are such that the attachment force, F_c, reaches its limit (16). Thus, detachment occurs when a contact force with a separation component (the direction of separation of the vehicles, $(p_2 - p_1)/\|p_2 - p_1\|_2$) exceeds the given maximum attachment force, σ.

$$F_c^T \cdot (p_2 - p_1)/\|p_2 - p_1\|_2 \geq \sigma.$$

This provides the formalization of $\mathrm{Guard}^\circ(1, 0)$, as,

$$\{(x, u) \in \mathrm{Domain}^\circ(1) : F_c^T \cdot (p_2 - p_1)/\|p_2 - p_1\|_2 \geq \sigma\}.$$

The $\mathrm{Reset}^\circ(q, q', x, u)$ maps for any given edge, (q, q'), state, x and input, u, the correspondent jump values $(x, u)^+$. For the scenario depicted by the edge, $(0, 1)$, we must take into account the collision dynamics. Assuming two vehicles with independent dynamics and no net external forces, we may consider conservation of linear momentum,

$$\begin{aligned} P_1 + P_2 =& P^+ \Leftrightarrow \\ m_1 v_1 + m_2 v_2 =& (m_1 + m_2) v^+ \end{aligned}$$

After the collision, the two modules are connected and thus, the center of mass, must have $v_c = v^+$ velocity. Assuming that both modules have the same mass, we obtain,

$$v_c^+ = \frac{v_1 + v_2}{2} \tag{42}$$

Similarly to the conservation of linear momentum, the conservation of angular momentum must also be ensured. The angular momentum of a body is given by,

$$L = J \cdot \omega = \frac{1}{2} m r^2 \cdot \omega$$

where $J \in \mathbb{R}$ depicts the moment of inertia (17) of a single detached circular module of mass m and radius r. The conservation of the angular momentum is given by,

$$\begin{aligned} L_1 + L_2 &= L^+ \\ J(\omega_1 + \omega_2) &= J^+ \cdot \omega^+. \end{aligned}$$

When connected, the full connected modular vehicle has a new moment of inertia, due to the new axis of rotation at the connection point (17),

$$J^+ = 2(J + md^2) = 3mr^2 = 6J.$$

with $d = r$ being the distance from the module's center of mass to the new rotational axis.

The relationship between the angular velocities of the modules before connecting (ω_1 and ω_2) and the angular velocity of the full system after connection (ω^+) can be expressed as,

$$\omega^+ = \frac{1}{6}(\omega_1 + \omega_2). \tag{43}$$

The angular velocity of the full connected module at any given moment, are the same as the angular velocity of each module, since they are connected. On the other hand, the linear velocities of each module are derived from (28) obtaining the expressions,

$$\begin{aligned} v_1^+ &= v^+ + S(\omega^+) r_1 \\ v_2^+ &= v^+ + S(\omega^+) r_2. \end{aligned} \tag{44}$$

From the described ETC dynamics in Section II-B, it is said that a mode change implies the recalculation of the desired state ($x^\star$) and input ($u^\star$) vectors. Moreover, as per (32), at any given instant $t = t_j + \tau$, the control input is given by, $u(t) = u^\star(\tau)$. Where t_j is the instance of the j-th event and $u^\star(\tau)$ is the desired input calculated for $\tau \in [0; T]$. Taking all of this into consideration, for the j-th event, if $(q, q', x, u) \in \{0\} \times \{1\} \times \text{Guard}^\circ(0, 1)$, then

$$\text{Reset}^\circ(0, 1, x, u) = \left(\begin{bmatrix} p_1 \\ v_1^+ \\ \theta_1 \\ \omega^+ \\ p_2 \\ v_2^+ \\ \theta_2 \\ \omega^+ \end{bmatrix}, u^\star(0) \right). \tag{45}$$

For the second scenario, for the edge, $(1, 0)$, the separation doesn't contain any discrete dynamics with respect to the state x. Only the new control input values ($u^\star(\tau)$) are reassigned. Thus, if $(q, q', x, u) \in \{1\} \times \{0\} \times \text{Guard}^\circ(1, 0)$, then

$$\text{Reset}^\circ(1, 0, x, u) = (x, u^\star(0)). \tag{46}$$

By combining this $\mathcal{H}_a^\circ$ with the model for the event-triggering hybrid system from Section II-B and the control law presented in Section III-B, the closed-loop hybrid system is completely defined.

IV. Simulation Results

The 2-dimensional graphs, shown in this section, depict positions of modules over time by dashed lines, and the reference positions are shown as thicker, lighter lines. Circular modules are depicted at calculation instances. These graphs provide insights into position and time. The time domain graphs are self-explanatory, where the vertical lines mark ETC events.

Figure 5 depicts the dynamics of two modules, initially connected ($q = 1$) and tracking their coinciding reference. As the reference signals diverge, it initiates the disconnected mode ($q = 0$), only to eventually converge back into each other, concluding the simulation in a connected state ($q = 1$).

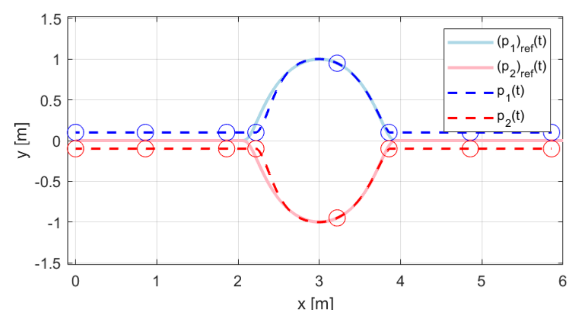

Figure 5: Plot of the positions ($p_1(t)$ and $p_2(t)$) and corresponding references ($(p_1)_{ref}(t)$ and $(p_2)_{ref}(t)$), for the case of two modules separating and joining.

The simulation effectively illustrated the transition between modes of operation. Specifically, the moments of contact and separation are represented by a pair of circles in the figure, signifying that the impact moment leads to the recalculation of the desired trajectories. Furthermore, during the instances where the modules were in a constant mode, they accurately tracked their reference signals.

The previous simulation assumed ideal conditions, where the applied forces and torques precisely followed the designed values, denoted as $u^\star(\tau)$. Consider a scenario where the desired input is calculated correctly, but the actuation is compromised, resulting in,

$$F_1(t) = F_1^\star(\tau) - \begin{bmatrix} \mu \\ \mu \end{bmatrix} sin(10t)$$

$$F_2(t) = F_2^\star(\tau) - \begin{bmatrix} \mu \\ \mu \end{bmatrix} sin(10t)$$

The ensuing simulations were conducted employing sinusoidal references, and all parameters were held constant, except for the disturbance coefficient μ. In these experiments, the ETC error threshold (11) was maintained at a constant value of $\delta = 0.2$.

The situation, depicted in Figure 6, illustrates the outcome of the control system in the presence of a minor disturbance coefficient ($\mu = 0.05$).

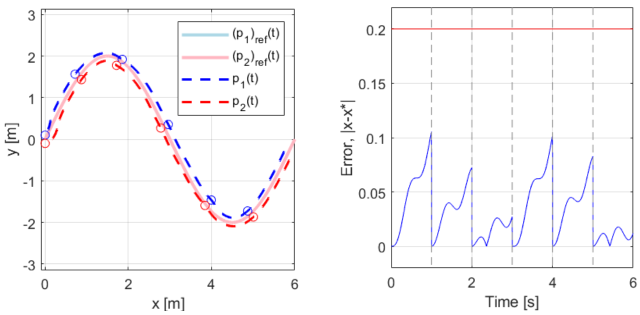

Figure 6: Dynamics and error plots, for $\delta = 0,2$ and $\mu = 0,05$.

The small dimension of μ is insufficient to yield significant deviations from the desired trajectory. While there are some minor visible deviations, it is evident from the error plot that the deviation is not significant enough to surpass the threshold values. Furthermore, the event instances remain equidistant, separated by one second.

The final situation depicted in Figure 7 now exhibits a substantial disturbance, providing a compelling demonstration of the system's stability.

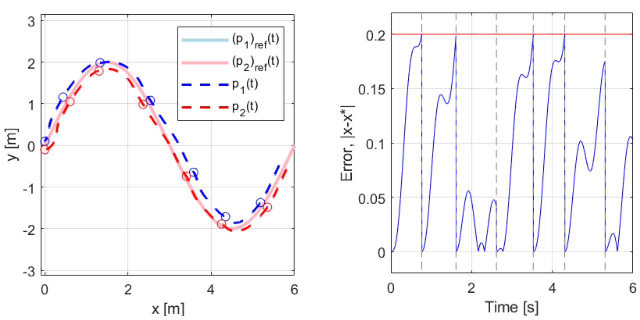

Figure 7: Dynamics and error plots, for $\delta = 0,2$ and $\mu = 0,15$.

The error plot shows that the disturbance exceeds the defined maximum threshold (δ) leading to, six event-triggered input recalculation, compared to five in the previous examples. Although the error is less evident in the positioning plot, it's clear from the error plot that the results are not ideal. However, the ETC mechanism effectively maintains system control. The sensitivity of the system to the deviations can be managed by adjusting the values of the threshold (δ), resulting in a more accurately referenced path, though at the cost of multiple unexpected recalculations of the control input.

V. Conclusion

This thesis introduces event-triggered control for multiple modules of aerial vehicles, using the paradigm of hybrid systems. The work includes a formal definition of the system in the form of an Open Hybrid Automaton. The stability of this system is theoretically proved, leading to the definition of the plant system and its corresponding event-triggered control strategy. The state-space equations governing the dynamics of the plant are derived, encompassing all possible dynamics. The thesis concludes with simulations in MATLAB, demonstrating the functioning of the closed-loop system and the dynamics of the modules under complex reference trajectories. It's shown that the system maintains its course even in adverse situations and disturbances.

Future research questions involve scaling the system to N modules and its impact on dynamics and outcomes, though this was beyond the scope of this thesis. The potential applications of multiple N modules, capable of achieving more than a single UAV, are of great interest. The thesis highlights the benefit of the Event-Triggered Control paradigm in reducing computational strain for real-time control strategies with complex systems, by focusing computation on event instances.

References

[1] N. S. Labib, M. R. Brust, G. Danoy, and P. Bouvry, "The rise of drones in internet of things: A survey on the evolution, prospects and challenges of unmanned aerial vehicles," IEEE Access, vol. 9, pp. 115 466–115 487, 2021.

[2] W. P. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in 2012 ieee 51st ieee conference on decision and control (cdc). IEEE, 2012, pp. 3270–3285.

[3] M. Okasha, J. Kralev, and M. Islam, "Design and experimental comparison of pid, lqr and mpc stabilizing controllers for parrot mambo mini-drone," Aerospace, vol. 9, no. 6, p. 298, 2022.

[4] D. Saldana, B. Gabrich, G. Li, M. Yim, and V. Kumar, "Modquad: The flying modular structure that self-assembles in midair," in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 691–698.

[5] R. Goedel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems: modeling stability, and robustness," 2012.

[6] W. Tong, W. Jie, and T. Bailing, "Periodic event-triggered formation control for multi-uav systems with collision avoidance," Chinese Journal of Aeronautics, vol. 35, no. 8, pp. 193–203, 2022.

[7] D. Saldana, B. Gabrich, G. Li, M. Yim, and V. Kumar, "Modquad: The flying modular structure that self-assembles in midair," in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 691–698.

[8] D. Saldana, P. M. Gupta, and V. Kumar, "Design and control of aerial modules for inflight self-disassembly," IEEE Robotics and Automation Letters, vol. 4, no. 4, pp. 3410–3417, 2019.

[9] N. Gandhi, D. Saldana, V. Kumar, and L. T. X. Phan, "Self-reconfiguration in response to faults in modular aerial systems," IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 2522–2529, 2020.

[10] J. Paulos, B. Caraher, and M. Yim, "Emulating a fully actuated aerial vehicle using two actuators," in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 7011–7016.

[11] HookandLoop.com, "Duragrip® and velcro® brand hook and loop products," https://www.hookandloop.com/products, accessed: 2023-10-09.

[12] J. Lygeros, C. Tomlin, and S. Sastry, "Hybrid systems: modeling, analysis and control," Electronic Research Laboratory, University of California, Berkeley, CA, Tech. Rep. UCB/ERL M, vol. 99, p. 6, 2008.

[13] R. Sanfelice, Hybrid Feedback Control, ser. Princeton Series in Applied Mathematics Series. Princeton University Press, 2021.

[14] J. Lee, Introduction to topological manifolds. Springer Science & Business Media, 2010, vol. 202.

[15] T. Hamel, R. Mahony, R. Lozano, and J. Ostrowski, "Dynamic modelling and configuration stabilization for an x4-flyer." IFAC Proceedings Volumes, vol. 35, no. 1, pp. 217–222, 2002.

[16] HookandLoop.com, "Velcro® brand sew on hook and loop fasteners," https://www.hookandloop.com/hook-and-loop-brands/velcro/velcror-brand-sew-on-hook-and-loop-fasteners, accessed: 2023-10-09.

[17] M. Spiegel, SCHAUM'S OUTLINE SERIES ON THEORETICAL MECHANICS. McGraw Hill Education, 1982.