Event-Triggered Multi-Mode MPC with Application to Modular Aerial Vehicles

Pedro Casau and Bruno Guerreiro

***Abstract*—Our paper proposes a controller for global asymptotic stabilization of a nonlinear plant by combining multiple model-predictive controllers with event-triggered control. Given a collection of operating modes, the controller drives the state of the closed-loop system to a target operating mode using a sequence of optimal input trajectories. The computation of optimal trajectories takes place at events which are triggered either when the difference between the state of the closed-loop system and the optimal state trajectory exceeds a given threshold or when an internal timer expires. We demonstrate the proposed controller through simulations by applying it to the control of modular aerial vehicles.**

***Index Terms*—Sampled-data control, Optimal Control, Hybrid Systems.**

I. Introduction

Among the myriad of unmanned aerial vehicle (UAV) applications emerging in our society [1], [2], [3], modular aerial vehicles are steadily capturing the interest of the academic and industrial communities [4], [5]. They are composed of independent flying modules that can be rearranged to meet the requirements of a particular task. There are fundamental reasons why it might be preferable to utilize modular aerial vehicles over generic all-purpose UAVs, which include their flexibility and redundancy. In addition to modules dedicated to providing thrust, modular aerial vehicles can also provide increased flexibility by having dedicated sensing and power modules that can be swapped in and out as required. On the other hand, their redundancy implies a level of robustness to unforeseen and adverse scenarios that a group of non-modular UAVs cannot provide. Even though some initial research on the control of modular aerial vehicles has already been carried out (cf. [6]), there is still a long way to go towards full autonomy. Achieving such a feat would require: improved mechanical design, the synthesis of trajectory tracking controllers for multiple configurations, compliant control under human interaction, fast adaptation for in-flight reconfiguration, among others. In this paper, we are particularly focused on a very specific part of this ensemble of problems: the design of docking controllers for in-flight reconfiguration of modular aerial vehicles. Towards that end, we propose a solution that combines Event-triggered Control (ETC) and Model Predictive Control (MPC).

P. Casau is with the Department of Electronics, Telecommunications and Informatics (DETI) at the University of Aveiro, Portugal (email: pcasau@ua.pt). B. Guerreiro is with the NOVA School of Science and Technology (FCT-NOVA) and CTS/Uninova, Caparica, Portugal (email: bj.guerreiro@fct.unl.pt) P. Casau and B. Guerreiro are also with the Institute for Systems and Robotics (ISR/LARSyS), Lisbon, Portugal. This work was partially supported by Fundação para a Ciência e a Tecnologia (FCT) through LARSyS - FCT Project UIDB/50009/2020, the FCT project CAPTURE (PTDC/EEI-AUT/1732/2020) and by FCT Scientific Employment Stimulus grant CEECIND/04652/2017.

Event-triggered Control (ETC) describes a control approach in which the sampling of the outputs of a plant happens only if necessary to meet certain control objectives, typically asymptotic stability. This approach unlocks the possibility of achieving the desired performance with a great reduction in the average sampling frequency. This is particularly important in networked control systems, where communication resources might be shared by many components of the system, but also in computationally demanding controllers, such as Model-Predictive Control (MPC).

A standard approach to MPC requires the choice of a sampling frequency, the discretization of a system model at that frequency and the computation of control signals in a receding horizon fashion (see. e.g. [7]). A more recent approach to standard MPC is provided in [8] and it consists of the reformulation of the control problem within the framework of hybrid control, where discretization is not strictly necessary and the control horizon is given in hybrid time. However, the Hybrid MPC approach proposed in in [8] requires the use of optimal control solvers for hybrid systems, which are currently still under development.

The controller design presented in this paper follows more closely the approach proposed in [9], where an MPC controller is combined with a send-on-delta event-triggered approach to achieve robust asymptotic stabilization of a nonlinear plant. From a theoretical perspective our contributions are as follows: 1) the use of hybrid dynamical systems to extend the dual-mode MPC strategy of [9] to a multi-mode MPC strategy with milder constraints on the formulation of the optimal control problem; 2) the relaxation of terminal time constraints in the optimal control problem formulation; 3) the closed-loop system satisfies the hybrid basic conditions, hence it is endowed with robustness to state perturbations (cf. [10]); 4) the proposed controller does not induce complete discrete solutions, hence the closed-loop system is robustly non-Zeno (cf. [11]).

The general multi-mode MPC controller is then applied to the control of independent flying modules of a modular aerial vehicle. In particular, we assume that each flying module has the dynamics of a multirotor vehicle and we implement a dual-mode MPC controller that: 1) Implements a minimum-time optimal controller away from the desired setpoint; 2) Implements a locally stabilizing MPC controller near the desired setpoint. It is worth noting that these choices do not fit into the control design in [9], hence the need to introduce a new set of assumptions that allow for the realization of the desired goal. From another viewpoint, this implementation may also be regarded as one of uniting global and local controllers using a hybrid supervisor [12].

Finally, we acknowledge that there exist some competing alternatives to the proposed solution. Regular planning

strategies that do not account for the docking of vehicles can be used to obtain predefined trajectories that a separate controller can follow [13], [14]. There are also recent planning approaches that consider the trajectories that can ensure a docking point between vehicles [15], also considering a separate controller to follow the planned trajectory, as well as MPC strategies for integrated planning and control towards docking or similar maneuvers [16]. These strategies merit a full-fledged comparison that we do not pursue in the present paper.

This paper is organized as follows. In Section II we present some notation and the required preliminary material on hybrid systems. In Section III we present the controller design and the main results of this paper. In Section IV, we demonstrate how the multirotor dynamical system can be cast as a triple integrator by means of an input transformation and we provide the details of the implementation of the proposed controller for this particular case. In Section V, we present some simulation results for both the triple integrator and the multirotor vehicle. Finally, in Section VI we provide some concluding remarks. The proofs in this paper were ommitted due to space constraints, but they will appear elsewhere.

II. Notation & Preliminaries

The Cartesian product $\mathbb{R}^n = \mathbb{R} \times \ldots \times \mathbb{R}$ of n copies of the real line together with scalar multiplication and component-wise addition of vectors is known as n-dimensional Euclidean space. The Euclidean metric topology is the one induced by the metric $x \mapsto |x| := \sqrt{x^\top x}$ with a basis of open balls $c + \delta\mathbb{B} := \{x \in \mathbb{R}^n : |x - c| < \delta\}$ for each $c \in \mathbb{R}^n$ and each $\delta > 0$. Given a set S, $\overline{S}$ denotes its closure. The Minkowski sum of two sets $A, B \subset \mathbb{R}^n$ is given by $A + B := \{a + b : a \in A,\ b \in B\}$.

A set-valued map M from $S \subset \mathbb{R}^m$ to the power set of some Euclidean space $\mathbb{R}^n$ is represented by $M : S \rightrightarrows \mathbb{R}^n$. The domain of a set-valued map is given by $\operatorname{dom} M := \{x \in \mathbb{R}^n : M(x) \neq \emptyset\}$. Given a subset S of $\mathbb{R}^m$, a set-valued map $M : S \rightrightarrows \mathbb{R}^n$ is said to be outer semicontinuous (relative to S) if its graph, given by $\operatorname{gph} M := \{(x, y) \in S \times \mathbb{R}^n : y \in M(x)\}$, is closed (relative to $S \times \mathbb{R}^n$). The set-valued map M is locally bounded at $x \in S$ if there exists a neighborhood U_x of x such that $M(U_x) \subset \mathbb{R}^n$ is bounded. It is locally bounded (relative to S) if it is locally bounded at each $x \in S$. It is convex-valued if $M(x)$ is convex for each $x \in S$.

A hybrid system $\mathcal{H}$ with state space $\mathbb{R}^n$ is defined as follows:

$$\begin{aligned} \dot{\xi} &\in F(\xi) \quad \xi \in C \\ \xi^+ &\in G(\xi) \quad \xi \in D \end{aligned} \tag{1}$$

where $\xi \in \mathbb{R}^n$ is the state, $C \subset \mathbb{R}^n$ is the flow set, $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is the flow map, $D \subset \mathbb{R}^n$ denotes the jump set, and $G : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ denotes the jump map. A solution ϕ to $\mathcal{H}$ is parametrized by (t, j), where t denotes ordinary time and j denotes the jump time, and its domain $\operatorname{dom}\phi \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$ is a hybrid time domain: for each $(T, J) \in \operatorname{dom}\phi$, $\operatorname{dom}\phi \cap ([0, T] \times \{0, 1, \ldots J\})$ can be written in the form $\cup_{j=0}^{J-1}([t_j, t_{j+1}], j)$ for some finite sequence of times $0 = t_0 \leq t_1 \leq t_2 \leq \cdots \leq t_J$, where $I_j := [t_j, t_{j+1}]$ and the t_j's define the jump times. A solution ϕ to a hybrid system is said to be *maximal* if it cannot be extended by flowing nor jumping and *complete* if its domain is unbounded.

The *hybrid basic conditions* provide a set of sufficient conditions for well-posedness and they are as follows (cf. [10, Assumption 6.5]):

(A1) C and D are closed subsets of $\mathbb{R}^n$;

(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, and $D \subset \operatorname{dom} G$.

A compact set $\mathcal{A}$ is said to be: *stable* for (1) if for every $\epsilon > 0$ there exists $\delta > 0$ such that every solution ϕ to (1) with $|\phi(0,0)|_{\mathcal{A}} \leq \delta$ satisfies $|\phi(t,j)|_{\mathcal{A}} \leq \epsilon$ for all $(t, j) \in \operatorname{dom}\phi$; *globally pre-attractive* for (1) if every solution ϕ to (1) is bounded and, if it is complete, then also $\lim_{t+j\to+\infty} |\phi(t,j)|_{\mathcal{A}} = 0$; *globally pre-asymptotically stable* for (1) if it is both stable and pre-attractive. If every maximal solution to (1) is complete then one may drop the prefix "pre".

A directed graph $\mathcal{G} := (\mathcal{Q}, \mathcal{E})$ is an ordered pair consisting of two finite sets $\mathcal{Q}$ and $\mathcal{E}$. The elements of $\mathcal{Q}$ are called *nodes* and the elements of $\mathcal{E}$ are called *directed edges*. Each directed edge $e = (t, h)$ is an ordered pair consisting of a tail $t \in \mathcal{Q}$ and a head $h \in \mathcal{Q}$. A *directed walk* W on a graph $\mathcal{G}$ is a sequence of nodes $\{q_j\}_{0 \leq j \leq k} \subset \mathcal{Q}$ represented by

$$W = q_0 q_1 \ldots q_k \tag{2}$$

such that, for each $j \in \{1, \ldots, k\}$, (q_{j-1}, q_j) is a directed edge of $\mathcal{G}$, i.e., $(q_{j-1}, q_j) \in \mathcal{E}$. We say that a directed graph $\mathcal{G} := (\mathcal{Q}, \mathcal{E})$ is acyclic if, for each $q \in \mathcal{Q}$, there are no directed walks on $\mathcal{G}$ from q to q.

III. Controller Design

Suppose that we are given the dynamical system

$$\dot{x} = f(x, u) \tag{3}$$

where $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, $x \in \mathbb{R}^n$ denotes the state and $u \in \mathbb{R}^m$ denotes the input. The goal of the present work is to develop a multi-mode event-triggered controller that drives the state trajectories of (3) to a target operating mode for all possible initial conditions.

To this end, we represent each operating mode with a label q that belongs to a *finite* set $\mathcal{Q} \subset \mathbb{N}$ and to each mode we associate an open set $U_q \subset \mathbb{R}^n$, so that the collection of all such sets covers $\mathbb{R}^n$, i.e., $\bigcup_{q \in \mathcal{Q}} U_q = \mathbb{R}^n$. Let $\mathcal{Q}_0 \subset \mathcal{Q}$ denote the set of all possible end nodes that are associated with the target operating modes. To achieve the desired goal of steering the state trajectories of (3) to the target operating modes, it is necessary to define the path (or paths) between each operating mode q in $\mathcal{Q} \backslash \mathcal{Q}_0$ to the set $\mathcal{Q}_0$, and we choose to do that using a directed acyclic graph $\mathcal{G}$ with the properties specified in the following assumption.

Assumption 1. *Given a set of nodes $\mathcal{Q}$ and edges $\mathcal{E}$ that define a directed acyclic graph $\mathcal{G} := (\mathcal{Q}, \mathcal{E})$ and a collection of open sets $\{U_q\}_{q \in \mathcal{Q}}$ satisfying $\bigcup_{q \in \mathcal{Q}} U_q = \mathbb{R}^n$, the following hold: 1) for each node $p \in \mathcal{Q}_0$ and for each node $q \in \mathcal{Q} \backslash \mathcal{Q}_0$ there is a directed walk on $\mathcal{G}$ from q to p; 2) for each $q \in \mathcal{Q}_0$, the set U_q is bounded.*

Switching from a mode $q \in \mathcal{Q}$ to another mode $p \in \mathcal{Q}$ requires that the state x moves from U_q to U_p. In this direction, let

$$\mathcal{S} := \mathcal{E} \cup \{(q', p') :\in \mathcal{Q}_0^2 : q' = p'\} \tag{4}$$

represent all the possible transitions between operating modes for the closed-loop system, including the edges $\mathcal{E}$ of the graph $\mathcal{G}$ and self-loops at the end nodes. Then, for each $(q, p) \in \mathcal{S}$ and for each $\widehat{x} \in \mathbb{R}^n$, we generate the input trajectories that drive the state of the system (3) from the initial condition $\widehat{x} \in \overline{U_q}$ to U_p by solving the following optimal control problem:

$$\begin{array}{ll} \text{minimize} & J_{q,p}(x(.), u(.)) \\ \text{subject to} & \dot{x}(t) = f(x(t), u(t)),\ u(t) \in \mathcal{U} \\ & x(0) = \widehat{x}, x(T) \in \Gamma_{q,p} \end{array} \tag{5}$$

where $\mathcal{U} \subset \mathbb{R}^m$ represents input constraints, $\Gamma_{q,p}$ denotes the terminal constraints, $T \geq 0$ is the terminal time (not necessarily fixed), $x(.)$ and $u(.)$ denote admissible state and input trajectories for (3)[1], respectively, and

$$J_{q,p}(x(.), u(.)) := h_{q,p}(x(T), T) + \int_0^T g_{q,p}(x(t), u(t), t)dt$$

is the cost functional, where $h_{q,p} : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ and $g_{q,p} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ represent the terminal and running costs for each $(q, p) \in \mathcal{S}$, respectively.

Assumption 2. *Given a graph $\mathcal{G} := (\mathcal{Q}, \mathcal{E})$ and a collection of open sets $\{U_q\}_{q\in\mathcal{Q}}$ satisfying Assumption 1, let $\mathcal{S}$ be given by* (4) *and let*

$$M_p := \mathbb{R}^n \setminus \left(\cup_{q'\in\mathcal{Q}\setminus\{p\}} U_{q'}\right) \tag{6}$$

for each $p \in \mathcal{Q}$. For each $(q, p) \in \mathcal{S}$ and each $\widehat{x} \in \overline{U_q}$, there is a unique solution

$$\tau \mapsto \begin{pmatrix} x^\star(\tau; \widehat{x}, q, p) \\ u^\star(\tau; \widehat{x}, q, p) \end{pmatrix} \tag{7}$$

to (5) *defined on $[0, T]$ such that $(\tau; \widehat{x}, q, p) \mapsto x^\star(\tau; \widehat{x}, q, p)$ is continuous. Moreover, given $\delta > 0$, there exists a continuous function $(\widehat{x}, q, p) \mapsto \eta(\widehat{x}, q, p)$ satisfying $\eta(\widehat{x}, q, p) \leq T$, such that $x^\star(\tau; \widehat{x}, q, p) + \delta\mathbb{B} \subset M_p$ for each $\tau \in [\eta(\widehat{x}, q, p), T]$.*

Assumption 2 imposes several regularity properties on the solutions to (5) that are very important to the controller design.[2] Uniqueness of solutions for each $(\widehat{x}, q, p)$ allows us to define the optimal control and state trajectories unambiguously. The continuity of $x^\star$ not only as a function of τ but also as a function of $(\widehat{x}, q, p)$ and the continuity of η are used to prove that the closed-loop hybrid system satisfies the hybrid basic conditions. Finally, and most importantly, we assume that each optimal state trajectory is able to drive the state from the initial mode $\overline{U_q}$ to the target mode U_p, and that this is achieved robustly, in the sense that there is a margin $\delta > 0$ between the tail end of each state trajectory and all the operating modes that are not the target mode p.

The controller design relies on the computation of the optimal state and input trajectories given in (7) at events, which we split into two separate kinds: send-on-delta and time-triggered. The timer associated with the time-triggered approach is denoted by $\tau \in \mathbb{R}_{\geq 0}$ and it has the hybrid dynamics $\dot{\tau} = 1$ if $\tau \in [0, \eta(\widehat{x}, q, p)]$, and $\tau^+ = 0$ if $\tau \geq \eta(\widehat{x}, q, p)$, where η is given in Assumption 2. The flow and jump sets associated with the time-triggered events are given by:

$$C_\tau := \{\xi \in \Xi : \tau \leq \eta(\widehat{x}, q, p)\} \tag{8a}$$
$$D_\tau := \{\xi \in \Xi : \tau \geq \eta(\widehat{x}, q, p)\}, \tag{8b}$$

respectively, where $\xi := (x, \widehat{x}, q, p, \tau)$ represents the state of the closed-loop system and $\Xi := \mathbb{R}^{2n} \times \mathcal{S} \times \mathbb{R}_{\geq 0}$ represents the state space.

The flow and jump sets associated with the send-on-delta event-triggered mechanism are given by:

$$C_\delta := \{\xi \in \Xi : |x - x^\star(\tau; \widehat{x}, q, p)| \leq \delta\} \tag{9a}$$
$$D_\delta := \{\xi \in \Xi : |x - x^\star(\tau; \widehat{x}, q, p)| \geq \delta\} \tag{9b}$$

respectively, with $\delta > 0$ given in Assumption 2. Combining the (8b) with (9), we obtain the closed-loop hybrid system $\mathcal{H} := (C, F, D, G)$, given by:

$$\begin{array}{l} \dot{\xi} \in F(\xi) := (f(x, u^\star(\tau; \widehat{x}, q, p)), 0, 0, 0, 1)\ \xi \in C \\ \xi^+ \in G(\xi) := \{\xi' \in \Xi : x' = \widehat{x}' = x, x \in \overline{U_{q'}}, \tau' = 0\} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \xi \in D \end{array} \tag{10}$$

where $\xi' := (x', \widehat{x}', q', p', \tau')$, $C := C_\delta \cap C_\tau$ and $D := D_\delta \cup D_\tau$ are the flow and jump sets, respectively.

The hybrid basic conditions [10, Assumption 6.5] guarantee that a hybrid system is endowed with robustness to small perturbations as discussed in [10, Chapter 7] and this motivates the following assumption and subsequent lemma.

Assumption 3. *The flow map F in* (10) *satisfies (A2).*

Lemma 1. *Suppose that Assumptions 2 and 3 hold. Then, the hybrid system $\mathcal{H} := (C, F, D, G)$ in* (10) *satisfies the hybrid basic conditions.*

Remark 1. *It is the case in Section IV (and in optimal control problems more generally) that the set of admissible input trajectories is piecewise continuous, thus the flow set F does not necessarily satisfy Assumption 3. In that case, we may consider the regularization $\widehat{F}(\xi) := \bigcap_{\delta>0} \overline{\text{con}} F((\xi+\delta\mathbb{B})\cap C)$ as introduced in [10, Definition 4.13], where $\overline{\text{con}}(S)$ denotes the closure of the convex hull of a set S. However, it must be demonstrated that each solution to $\mathcal{H} := (C, F, D, G)$ in* (10) *is also a solution to the regularized hybrid system $\widehat{\mathcal{H}} := (C, \widehat{F}, D, G)$.*

The objective of the proposed controller is to globally asymptotically stabilize the set

$$\begin{array}{l} \mathcal{A} := \{\xi \in \Xi : x = x^\star(\tau; \widehat{x}, q, p), \widehat{x} \in M_q, \\ \qquad\qquad \tau \leq \eta(\widehat{x}, q, p), q \in \mathcal{Q}_0, q = p\} \end{array} \tag{11}$$

with M_q given in (6), using an event-triggered multi-mode MPC approach with the following properties: 1) regardless of the initial condition $x_0 \in \mathbb{R}^n$ there exists a path through the graph $\mathcal{G}$ that corresponds to the application of a sequence of optimal controllers which take the state of the system from the initial condition x_0 to U_q for some $q \in \mathcal{Q}_0$; 2) the control signal operates in open-loop between updates, which occur

[1] An admissible input trajectory $u(.)$ is a piecewise continuous function and an admissible state trajectory $x(.)$ is an absolutely continuous function that satisfies (3) almost everywhere.

[2] In practice, Assumption 2 imposes constraints on the algorithm that computes the solution to (5).

either when the state deviates from the expected trajectory by at least an amount δ or when the timer expires, i.e., when $\tau = \eta(\widehat{x}, q, p)$; 3) the controller drives state trajectories to the solution of (5) for a collection of possible target modes $q \in \mathcal{Q}_0$; 4) the event-triggered mechanism allows for corrections in the initialization of the controller and it protects against the influence of disturbances and unmodeled dynamics by triggering the computation of new optimal control signals when the tracking error exceeds δ. Under the previous assumptions, it is possible to prove the following theorem.

Theorem 1. *Suppose that Assumptions 1, 2 and 3 hold. If there exists $\mu > 0$ such that $x^\star(\tau; \widehat{x}, q, \underline{q}) + \delta\overline{\mathbb{B}} \subset U_q$ for each $\tau \in [0, \eta(\widehat{x}, q, p)]$, each $\widehat{x} \in M_q + \mu\overline{\mathbb{B}}$ and each $q \in \mathcal{Q}_0$, then, the set $\mathcal{A}$ in* (11) *is globally asymptotically stable for $\mathcal{H}$ in* (10)*.*

Note that, since the image of the jump set of (10) through the jump map does not intersect the jump set, each maximal solution to (10) has a positive lower bound to the time between jumps (cf. [17]). In the next section, we demonstrate the application of the proposed controller to the stabilization of multirotor vehicles.

IV. Application to Multirotor Control

In this section, we apply the multi-mode MPC strategy that was presented in Section III to the control of multirotor aerial vehicles. This class of vehicles can be described by the following dynamical system:

$$\dot{p_I} = v_I, \qquad \dot{v_I} = Rr\mu + g, \qquad \dot{R} = RS(\omega_B), \tag{12}$$

where $S : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$ is the unique skew-symmetric matrix such that $S(a)b = a \times b$ for all $a, b \in \mathbb{R}^3$, $g \in \mathbb{R}^3$ is the gravity vector, $r \in \mathsf{S}^2 := \{x \in \mathbb{R}^3 : |x| = 1\}$ is a constant unitary vector that represents the direction of the thrust in body-fixed coordinates, $p_I \in \mathbb{R}^3$ denotes the position with respect to an inertial reference frame, $v_I \in \mathbb{R}^3$ denotes the linear velocity of the vehicle with respect to the inertial frame, $R \in \mathsf{SO(3)} := \{R \in \mathbb{R}^{3\times 3} : R^\top R = I_3,\ \det(R) = 1\}$ is a rotation matrix that maps vectors in a body-fixed frame to the inertial frame, $\mu \in \mathbb{R}$ represents the specific thrust and $\omega_B \in \mathbb{R}^3$ is the angular velocity of the vehicle in the body-fixed frame (cf. [18]) Letting

$$a_I := Rr\mu + g \tag{13}$$

we transform the dynamical system (12) into

$$\dot{p_I} = v_I, \qquad \dot{v_I} = a_I, \qquad \dot{a_I} = \overline{u} \tag{14}$$

where

$$\overline{u} := R \underbrace{\begin{bmatrix} r & -S(r\mu) \end{bmatrix}}_{M(\mu)} \begin{bmatrix} \dot{u} \\ \omega_B \end{bmatrix}. \tag{15}$$

The kernel of $M(u)$ is given by

$$\ker M(\mu) = \begin{cases} \{0\} \times \mathbb{R}^3 & \text{if } \mu = 0 \\ \operatorname{span}\left\{ \begin{bmatrix} 0 & r^\top \end{bmatrix}^\top \right\} & \text{if } \mu \neq 0 \end{cases},$$

which implies that the component of the angular velocity ω_B that is aligned with r has no influence over the value of μ. Additionally, the matrix $M(\mu)$ has full row rank for each $\mu \neq 0$, therefore it is invertible to the right with right inverse given by $M(\mu)^\dagger = \begin{bmatrix} r & -S(r\mu^{-1})^\top \end{bmatrix}^\top$ for each $u \neq 0$. The assignment $\begin{bmatrix} \dot{\mu} & \omega_B^\top \end{bmatrix}^\top = M(\mu)^\dagger R^\top \overline{u}$ is a solution to (15) for each $u \neq 0$ and it is in fact the solution to (15) with the lowest norm. We conclude that (13) allows us to transform the problem of stabilizing the position of a multirotor vehicle with dynamics (12) into one of stabilizing three triple integrators corresponding to the three spatial coordinates, while avoiding the condition $\mu = 0$. To achieve this goal, we start with the design of a dual-mode MPC for the triple integrator and, subsequently, we present the simulation results for the stabilization of multirotor vehicles using the transformation outlined above.

A. Dual-Mode MPC for Global Asymptotic Stabilization of a Triple Integrator

Motivated by the application of multirotor control, particularly the realization of docking maneuvers between two modules of a modular aerial vehicle and following the controller design outlined in Section III, we propose a dual-mode MPC that combines a minimum-time controller with finite-time linear quadratic regulator. In this direction, we consider that the logic variable q has one of two possible values, i.e., $q \in \mathcal{Q} := \{1, 2\}$, and we define the domains in which each mode operates as follows:

$$U_1 := \{x \in \mathbb{R}^3 : x^\top P x > \underline{\epsilon}\}, \tag{16a}$$

$$U_2 := \{x \in \mathbb{R}^3 : x^\top P x < \overline{\epsilon}\}, \tag{16b}$$

for some $0 < \underline{\epsilon} < \overline{\epsilon}$ and a positive definite matrix $P \in \mathbb{R}^{3\times 3}$. In other words, the controller corresponding to $q = 1$ is selected away from the origin and the controller corresponding to $q = 2$ is selected in a neighborhood of the origin. The sets $\{U_q\}_{q\in\mathcal{Q}}$ are open, and the difference between $\underline{\epsilon}$ and $\overline{\epsilon}$ guarantees that $\bigcup_{q\in\mathcal{Q}} U_q = \mathbb{R}^3$. Under this construction, the directed acyclic graph that determines the sequence of controllers to be employed is composed of two nodes and a single edge $(1, 2)$ that emphasizes the fact that the optimal controller of $q = 1$ drives the state to $q = 2$. In this case $\mathcal{Q}_0 = \{2\}$ and U_2 is bounded, which completes the proof of Assumption 1.

The minimum-time controller is selected when $q = 1$, and the minimum-time control signal for a particular initial condition $\widehat{x}$ is obtained by solving the optimization problem (5) with $J_{1,2}(x(.), u(.)) = T$, $x(T) = 0$, $|u(t)| \leq 1$ for all $t \in [0, T]$, free terminal time T and $f(x, u) = Ax + Bu$, where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

It follows from [19, Theorems 6.6-8] that the solution to minimum-time optimal control problem is not singular, thus it is unique and has a control signal $u^\star(.; \widehat{x}, 1, 2)$ that is piecewise constant and satisfies $u^\star(\tau; \widehat{x}, 1, 2) \in \{-1, 1\}$ for each $\tau \in [0, T]$. We set the value of the function η for each $\widehat{x} \in U_1$ and $(q, p) = (1, 2)$ to be the terminal time T of the corresponding solution to (5). Since $x^\star(T; \widehat{x}, 1, 2) = x^\star(\eta(\widehat{x}, 1, 2); \widehat{x}, 1, 2) = 0$ by design, then the reachability condition in Assumption 2 is satisfied for

any $\delta > 0$ satisfying $\delta \leq \min\{|x| : x^\top P x = \underline{\epsilon}\}$. As explained in [19], there exists a state-feedback law $x \to \kappa(x)$ such that $\tau \mapsto \kappa(x^\star(\tau; \widehat{x}, q, p)) = u^\star(\tau; \widehat{x}, q, p)$, where $(x^\star(.; \widehat{x}, q, p), u^\star(.; \widehat{x}, q, p))$ is the unique continuou solution to (5). Using the previous definition of κ given in [20], one could compute $x^\star$ by determining the unique solution to the initial value problem

$$\begin{cases} \dot{x} = Ax + B\kappa(x) \\ x(0) = \widehat{x} \end{cases}. \tag{17}$$

This concludes the verification of the requirements in Assumption 2 for $(q, p) = (1, 2)$.

It is important to note that the flow map F as defined in (10) does not satisfy (A2) because $\tau \mapsto u^\star(\tau; \widehat{x}, 1, 2)$ is not continuous. Therefore, we consider the Krasovskii regularization of F for the sake of satisfying Assumption (A2) which, as pointed out in Remark 1, does not influence the analysis because maximal solutions to (10) with and without regularization of the flow map coincide.

When $q = 2$, the linear quadratic regulator is used instead of the minimum-time controller. The control signal and state trajectories $(x^\star, u^\star)$ are obtained from the solution to the optimization problem (5) subject to the input constraints $|u(\tau)| \leq 1$ for each $\tau \geq 0$ and with cost functional given by: $J_{2,2}(x(.), u(.)) := \int_0^{+\infty} x(\tau)^\top Q x(\tau) + u(\tau)^\top R u(\tau) d\tau$ where $Q \in \mathbb{R}^{3\times 3}$ is a positive definite matrix, $R > 0$, and $x(T) = x(\infty)$ is free. The solution to the optimization problem is the solution to the initial value problem

$$\begin{cases} \dot{x^\star}(\tau) = Ax^\star(\tau) + Bu^\star(\tau) \\ x^\star(0) = \widehat{x} \end{cases} \qquad \tau \geq 0 \tag{18}$$

with $u^\star(\tau) = -R^{-1}B^\top P x^\star(\tau)$ for each $\tau \geq 0$, where $P \in \mathbb{R}^{3\times 3}$ is the solution to the Riccati equation

$$0 = PA + A^\top P + Q - PBR^{-1}B^\top P. \tag{19}$$

Considering that P in (16) is also the solution to the Riccati equation, we have that U_2 is strongly forward invariant for the closed-loop system (10) and, for small enough $\overline{\epsilon} > 0$, the input $u^\star$ satisfies the input constraints $|u^\star(\tau; \widehat{x}, q, p)| \leq 1$ for each $\widehat{x} \in U_2$ and $q = p = 2$. It is also the case that $(\tau, \widehat{x}, q, p) \mapsto x^\star(\tau; \widehat{x}, q, p)$ is continuous and unique, since it is the solution to (18). Choosing $\widehat{x} \mapsto \eta(\widehat{x}, 2, 2)$ to be equal to a constant $\overline{\tau}$ that satisfies $\overline{\tau} > \lambda^{-1} \log\left(\underline{\epsilon}\,\overline{\epsilon}^{-1}\right)$, $\lambda := \lambda_{min}(Q)/\lambda_{max}(P)$ we have that η is continuous and that the reachability requirement in Assumption 2 is satisfied for some $\delta > 0$. In fact, the LQR controller associated with $q = 2$ also meets the assumptions of Theorem 1 for some $\delta > 0$ due to the forward invariance of the sublevel sets of $x \mapsto x^\top P x$.

In the following section, we present simulation results that illustrate the behavior of the closed-loop system for the stabilization of the triple integrator and of a multirotor vehicle.

V. Simulation Results

To illustrate the behavior of the closed-loop system resulting from the interconnection of the triple integrator with the controller proposed in Section III, we present the simulation results in Figure 1 which represent the evolution of the state of the system and of the error between the state and the expected trajectory for varying levels of input-matched perturbations, i.e., $u \equiv u(\tau; \widehat{x}, q, p) + \rho \sin(0.2\pi t)$ for all $t \geq 0$ and for $\rho \in \{0, 0.01, 0.1\}$. In particular, the simulations represented in Figure 1 are obtained for a randomly selected initial condition:

$$x(0,0) = \begin{bmatrix} -0.2949 & 0.1868 & 0.1704 \end{bmatrix}^\top \tag{20}$$

with controller parameters $\overline{\tau} = 1$, $\delta = 0.05$, $\overline{\epsilon} = 0.1$ and $\underline{\epsilon} = 0.5\overline{\epsilon}$. In addition, we correctly initialize the memory variable $\widehat{x}$ to match the initial condition $x(0,0)$ in (20), the timer variable τ starts at 0 and the initial mode is $q(0,0) = 1$. This initialization improves the transient response, because it prevents the controller update that would be necessary in order to trigger the correct mode of operation. The star shaped markers in Figure 1 identify the events of the closed-loop system (10), while the vertical dashed lines identify the subset of events where the controller switches from the minimum-time controller to the linear quadratic regulator and vice-versa.

As expected, the tracking error increases with the magnitude of the disturbances. When $\rho = 0$, it is possible to verify that the difference between the actual trajectory and the expected trajectory remains zero for all $t \geq 0$. When $\rho = 0.01$, the state trajectory does not deviate significantly from the state trajectory of the nominal case and, in particular, it does not come close to triggering the $\delta = 0.05$ threshold. In both of these cases the switch of the controller mode happens at $t = \eta(\widehat{x}(0,0), 1, 2) \approx 1.6$, that is, when the timer expires. However, when $\rho = 0.1$, the behavior of the closed-loop system is very different. In this case, there are several violations of the δ threshold which trigger updates to the control signal, but we should point out that: 1) mode switching can be mitigated by careful selection of the controller parameters; 2) this simulation demonstrates that the closed-loop system is resilient to the influence of input-matched perturbations with a maximum magnitude of 10% of the actuator saturation level, since the tracking error remains bounded in all cases; 3) the average sampling frequency remains small despite the influence of perturbations.

To illustrate the application of the proposed controller to the stabilization of multirotor vehicles, we carry out two separate simulation runs of three triple integrators in accordance with the dynamics (14). The initial conditions have been selected at random from a uniform distribution.

Figure 2 illustrates the position of two multirotor vehicles which are steered into each other's proximity in order to perform a docking maneuver. Due to the saturation of the controllers the maneuvers are not very aggressive which can be seen by the fact that none of the vehicles deviates significantly from a leveled configuration (cf. 3). The reader may find the source code of the simulations presented here at https://github.com/pcasau/ACC2023.

VI. Conclusions

In this paper, we have presented a controller design that combines a multiple mode approach to Model Predictive Control (MPC) with event-triggered control, in order to attain global asymptotic stabilization of a target operating mode for the closed-loop system. Under the proposed approach, events are triggered when a timer state expires or sooner than that if

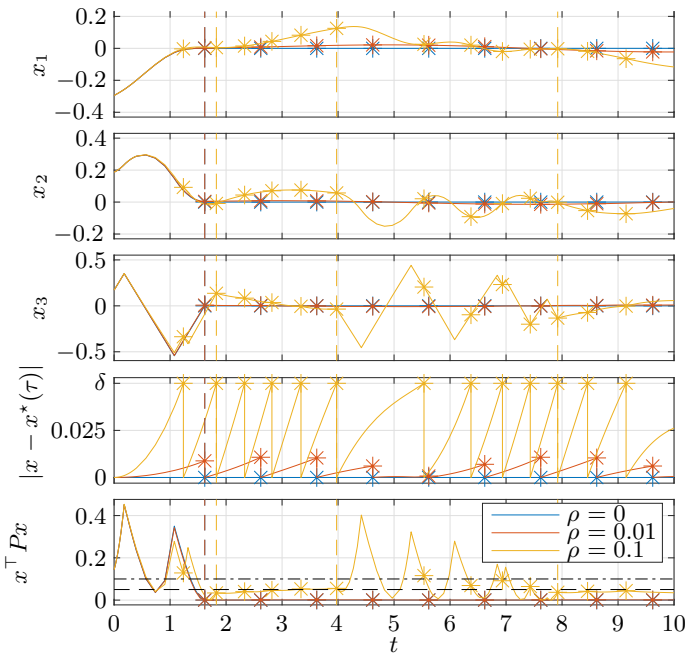

Fig. 1. Simulation results for the stabilization of the triple integrator under varying levels of input-matched perturbations $u \equiv u(\tau; \widehat{x}, q, p) + \rho \sin(0.2\pi t)$.

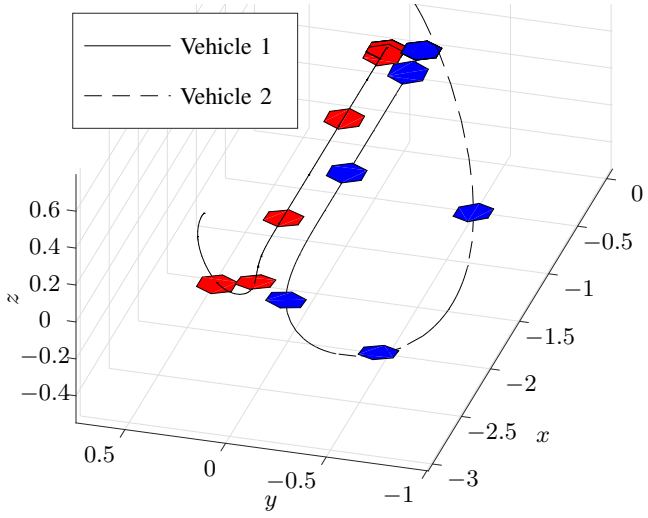

Fig. 2. Simulation of a docking maneuver using two multirotor vehicles. Snapshots of the position of the vehicles are taken one second apart from each other.

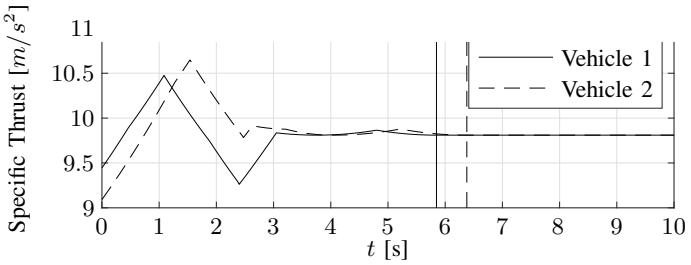

Fig. 3. The evolution of the thrust with time for two multirotor vehicles. The vertical lines represent, for each vehicle, the time where the LQR is selected for all corresponding triple integrators.

the state of the system deviates from the expected trajectory by a predefined amount. In either case, the event triggers the computation of a new control signal for the given mode, and resets the timer to zero. We demonstrate the application of this controller to the control of multirotor vehicles and we resort to simulation results in order to illustrate the behavior of the closed-loop system.

References

[1] M. Giles, "Zipline Launches the world's fastest commercial delivery drone," 2018.

[2] C. Huang, F. Gao, J. Pan, Z. Yang, W. Qiu, P. Chen, X. Yang, S. Shen, and K. T. T. Cheng, "ACT: An Autonomous Drone Cinematography System for Action Scenes," in *Proceedings of the 2018 IEEE International Conference on Robotics and Automation*, pp. 7039–7046, IEEE, 2018.

[3] S. Weeks, R. M. Osorno, B. Prestwich, L. Sanford, and A. Amin, "Additive Manufacturing Drone Design Challenge," *2020 Intermountain Engineering, Technology and Computing, IETC 2020*, 2020.

[4] M. D. Patterson, J. R. Quinlan, W. J. Fredericks, E. Tse, and I. Bakhle, "A modular unmanned aerial system for missions requiring distributed aerial presence or payload delivery," in *AIAA SciTech Forum - 55th AIAA Aerospace Sciences Meeting*, no. January, 2017.

[5] M. A. Ferreira, M. F. T. Begazo, G. C. Lopes, A. F. Oliveira, E. L. Colombini, and A. Simões, "Drone Reconfigurable Architecture (DRA): a Multipurpose Modular Architecture for Unmanned Aerial Vehicles (UAVs)," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 99, no. 3-4, pp. 517–534, 2020.

[6] D. Saldana, B. Gabrich, G. Li, M. Yim, and V. Kumar, "ModQuad: The flying modular structure that self-assembles in Midair," in *IEEE International Conference on Robotics and Automation*, pp. 691–698, 2018.

[7] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability," *Automatica*, vol. 34, no. 10, pp. 1421–1426, 1998.

[8] B. Altin and R. Sanfelice, "Asymptotically stabilizing model predictive control for hybrid dynamical systems," in *Proceedings of the American Control Conference*, vol. 2019-July, pp. 3630–3635, 2019.

[9] H. Li and Y. Shi, "Event-triggered robust model predictive control of continuous-time nonlinear systems," *Automatica*, vol. 50, no. 5, pp. 1507–1513, 2014.

[10] R. Goebel, R. Sanfelice, and A. Teel, *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, 2012.

[11] P. Casau, R. Sanfelice, and C. Silvestre, "On the Robustness of Nominally Well-Posed Event-Triggered Controllers," *IEEE Control Systems Letters*, vol. 6, 2022.

[12] R. G. Sanfelice, A. R. Teel, and R. Goebel, "Supervising a family of hybrid controllers for robust global asymptotic stabilization," in *47th IEEE Conference on Decision and Control*, pp. 4700–4705, IEEE, 2008.

[13] D. Mellinger, A. Kushleyev, and V. Kumar, "Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 477–483, IEEE, 2012.

[14] J. Tordesillas and J. P. How, "MADER: Trajectory Planner in Multiagent and Dynamic Environments," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 463–476, 2022.

[15] J. Pinto, B. J. Guerreiro, and R. Cunha, "Planning Parcel Relay Manoeuvres for Quadrotors," in *2021 International Conference on Unmanned Aircraft Systems,*, pp. 137–145, 2021.

[16] F. Matos and B. Guerreiro, "Model predictive control strategies for parcel relay manoeuvres using drones," in *2021 International Young Engineers Forum*, pp. 32–37, IEEE, 2021.

[17] J. Chai, P. Casau, and R. G. Sanfelice, "Analysis and design of event-triggered control algorithms using hybrid systems tools," *International Journal of Robust and Nonlinear Control*, vol. 30, no. 15, pp. 5936–5965, 2020.

[18] 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.

[19] M. Athans and P. L. Falb, *Optimal Control: An Introduction to the Theory and Its Applications*. Dover Books on Engineering, Mineola NY: Dover Publications, 2007.

[20] L. Y. Pao and G. F. Franklin, "Proximate time-optimal control of third-order servomechanisms," *IEEE Transactions on Automatic Control*, vol. 38, no. 4, pp. 560–580, 1993.