

Prediction-Based Leader-Follower Rendezvous Model Predictive Control with Robustness to Communication Losses

Dženan Lapandić¹, Christos K. Verginis², Dimos V. Dimarogonas¹ and Bo Wahlberg¹

Abstract—In this paper we propose a novel distributed model predictive control (DMPC) based algorithm with a trajectory predictor for a scenario of landing of unmanned aerial vehicles (UAVs) on a moving unmanned surface vehicle (USV). The algorithm is executing DMPC with exchange of trajectories between the agents at a sufficient rate. In the case of loss of communication, and given the sensor setup, agents are predicting the trajectories of other agents based on the available measurements and prior information. The predictions are then used as the reference inputs to DMPC. During the landing, the followers are tasked with avoidance of USV-dependent obstacles and inter-agent collisions. In the proposed distributed algorithm, all agents solve their local optimization problem in parallel and we prove the convergence of the proposed algorithm. Finally, the simulation results support the theoretical findings.

I. INTRODUCTION

The problem of autonomous navigation and landing of unmanned aerial vehicles (UAVs) on other autonomous agents has drawn a considerable attention in recent years. The coordination and control of UAVs with optimization-based methods can be challenging in such scenarios due to the limited computational resources on-board. In that respect, a relevant methodology is cooperative distributed model predictive control whose aim is to optimize control inputs of a considered agent given the predicted trajectories of other (neighbouring) agents over the planning horizon. Each agent solves the finite-horizon distributed optimal control problem (DOCP), applies its control input and broadcasts the predicted trajectory to other agents [1]–[3].

The considered agents in our setup are equipped with sensors, i.e. a camera on the quadrotor (UAV) side, and a radar on the boat (USV) side, that enable the accurate position measurement of the neighbouring agents with a frequency significantly higher than the real-time execution frequency of the MPC. In the case of loss of communication between the agents, an agent is left only with the data history and locally available measurements. During challenging maneuvers such as landing or navigating through a space with moving obstacles, the frequency of execution of the DMPC relative to the moving speeds of the agents and obstacles can severely affect the performance and safety. This is especially

the case when the references are based only on the current measurements and not on the committed trajectories, which indeed occurs in the loss of communication scenario. Note that, the communication losses we consider can be both temporary and permanent, because even an occasional package drop for few seconds that can occur during the hardware-in-the-loop experiments can lead to disastrous outcomes.

The hierarchical approaches presented in the literature [4], [5] deal with the described problem by separating the control in different layers. Usually, there is a high-level planning layer and a safety layer, for example with Control Barrier Functions (CBF) operating on a much higher frequency that is tasked with safety in the presence of obstacles [6]. Fundamentally, the concept of MPC is to unify these layers. Both approaches have their advantages and disadvantages and all aspects of the chosen concept must be taken into account to avoid unwanted behaviors. One of the ways is to use a prediction scheme to complement the available data. In [7], the authors propose a vision-based framework with the state estimation for the ground vehicle considered as a moving target. Estimation-based control protocol proposed in [8] uses only local observations of the state of the neighboring agents for rendezvous and flocking control. In another estimation-based protocol, authors in [9] use particle filter to predict aircraft trajectories. Prediction-based navigation in a decentralized event-based scheme is studied in [10]. Moreover, learning-based prediction approaches for multi-agent systems in the recent years became very popular area of research [11]–[15]. Unfortunately, most of these approaches require extensive amount of data for the training purpose and direct transferability to our particular application is unknown.

In this paper, we case the multi-agent and heterogeneous (involving UAVs and USVs) rendezvous problem as a leader-follower network with one leader and one or more follower agents for autonomous landings. The leader in our case is the USV boat and UAVs are considered as followers. We formulate the problem as a cooperative distributed model predictive control problem with collision avoidance constraints. In the case when the predicted trajectory is available to the followers, and the follower is able to dynamically follow the given reference trajectory, the landing can be executed using a relatively simple MPC for a single agent follower. However, a sudden communication loss can severely destabilize the landing leading to a collision with obstacles on the boat. Based on this observation, we adopt a prediction scheme in the case of communication loss to enhance the safety and performance.

Moreover, we generalize the scheme to multiple-follower

This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) and the Swedish Research Council, Knut and Alice Wallenberg Foundation (KAW).

¹Dženan Lapandić, Dimos V. Dimarogonas and Bo Wahlberg are with Division of Decision and Control Systems, KTH Royal Institute of Technology, Stockholm, Sweden. lapandic,dimos,bo@kth.se

²Christos K. Verginis is with Division of Signals and Systems, Department of Electrical Engineering, Uppsala University, Uppsala, Sweden. christos.verginis@angstrom.uu.se

rendezvous in which the inter-agent collisions must be handled. There are several challenges in the inter-agent collision avoidance in this case. First, in the sequential application of the algorithm each agent is optimizing its control strategy based on the shared trajectories that other agents have committed to. In this case, each agent must wait until all other agents have shared their new trajectories to begin the computation of its next control input. Otherwise, using the old shared trajectories without additional safety measures can lead to a collision. Second, in the parallel version the agents operate with the most recent available shared trajectories and account for the worst-case deviations as a safety measure. However, this can be very conservative and lead to deadlock. Third, if one or more agents lose the communication for some duration safety can be compromised. This, in some sense, resembles a non-cooperative scenario and we propose appropriate safety measures.

The contributions of this paper are

- a rendezvous algorithm based on leader-follower DMPC formulation for autonomous landing in presence of obstacles
- a convergence proof for a probabilistic safe landing in the case of communication loss
- an inter-agent collision avoidance robust to communication losses

The paper is organized as follows. The problem formulation is given in Sec. II and the control architecture is described in Sec. III. In Sections V and IV, one-follower and multiple-follower cases are examined and the proposed rendezvous algorithm is stated. Finally, Sec. VI presents the simulation results and Sec. VII concludes the paper.

A. Notation

We denote the discrete time step with t and set of agents with $N := \{l, f_1, \dots, f_M\}$. The state trajectories are denoted with $x_i(t)$, the predicted nominal state trajectories with $\hat{x}_i(k|t)$ and optimal state trajectories with $\hat{x}_i(k|t)$ for an agent i at time step $k + t$ predicted at time step t . The collection of trajectories from time step t until $t + N$ is denoted with $x_i(jt)$; $B_r := \{a : \|a - r\| \leq r\}$ is a ball of radius r ; \oplus is Minkowski sum addition defined as $x \oplus S := \{x + a : a \in S\}$. For an element $i \in N$, we denote a set excluding the element i with $N_{-i} := N \setminus \{i\}$.

II. PROBLEM FORMULATION

Consider a multi-agent system consisting of one leader and M follower agents that are dynamically decoupled and behaving according to the nonlinear discrete time dynamics

$$x_i(t+1) = f_i(x_i(t), u_i(t)) + w_i(t), \quad (1)$$

where $i \in N = \{l, f_1, \dots, f_M\}$, l denotes the leader, f_1, \dots, f_M denote the followers, M is the number of follower agents, $x_i \in X_i \subset \mathbb{R}^{n_i}$, $u_i \in U_i \subset \mathbb{R}^{m_i}$ denote the state and input of an agent i that are subject to the state and input constraints X_i and U_i , respectively, and $w_i \in W_i \subset \mathbb{R}^{n_i}$ is unknown but bounded disturbance in a compact set W_i .

The follower agents have the same dynamics and state space of a quadrotor UAV as in [16] which are different from the leader ones modeled as a 3DoF boat USV model [17]. We assume that the landing platform is rigidly attached to the boat USV and neglect the heave (vertical motion), roll and pitch motion. Furthermore, the first three elements of state vectors of all agents denote the position $p_i \in \mathbb{R}^3$.

The overall multi-agent system dynamics in stack-vector form are

$$x(t+1) = f(x(t), u(t)) + w(t), \quad (2)$$

where $x = [x_l^T, x_{f_1}^T, \dots, x_{f_M}^T]^T$, $u = [u_l^T, u_{f_1}^T, \dots, u_{f_M}^T]^T$, $w = [w_l^T, w_{f_1}^T, \dots, w_{f_M}^T]^T$. The state and input constraints of the overall system are $x \in X := X_l \times X_{f_1} \times \dots \times X_{f_M}$, and $u \in U := U_l \times U_{f_1} \times \dots \times U_{f_M}$, the set of all disturbances is $w \in W := W_l \times W_{f_1} \times \dots \times W_{f_M}$.

We formulate the inter-agent collision avoidance constraints as

$$h_{ij}(x_i(t), x_j(t)) = 0, \quad \forall i, j \in N, i \neq j, \quad (3)$$

where $h_{ij} : X_i \times X_j \rightarrow \mathbb{R}$ is a function that encodes collisions and will be defined later. Given a solution $x(t)$ of the system (2), if the constraint functions (3) are satisfied for all $t \geq t_0$ then $x(t)$ is a collision-free solution, i.e. $x(t) \in F \subset X$, where F denotes the collision-free space.

This paper considers the problem of navigating the leader agent to follow a given reference and follower agents to a rendezvous position with respect to the leader while avoiding inter-agent collisions. We denote the leader reference $x_{r,l}(t) = x_r(t)$ and references of the followers $x_{r,i}(x_l(t), c_i) := [p_l(t) + c_i, 0_{n_i-3}]^T$ with respect to the position $p_l(t)$ of the leader x_l and a given offset $c_i \in \mathbb{R}^3$ encoding the particular landing position of an i -th agent, $i \in N_{-l}$. With a slight abuse of notation, the problem treated in the paper can be stated as follows.

Problem 1. Consider a multi-agent system (1). Design a control policy u such that

$$\begin{aligned} & x(t) \in F, \quad t \geq t_0 \\ & \lim_{t \rightarrow \infty} (x_l(t), u_l(t)) = (x_r(t), 0_{n_l}) \\ & \lim_{t \rightarrow \infty} (x_i(t), u_i(t)) = (x_{r,i}(x_l(t), c_i), 0_{n_i}), \forall i \in N_{-l} \end{aligned}$$

Moreover, we assume that agents are able to communicate and share their current and predicted positions $\hat{z}_i(jt)$ asynchronously. Because the follower agents have different dynamics from the leader, and different state vectors, we define the following mapping from the leader state space X_l to the follower state space X_i , $i \in N_{-l}$

$$\hat{z}_i(t) = H \hat{x}_l(t), \quad (4)$$

where $H = \text{diag}(I_3, 0_{(n_i-3) \times (n_l-3)})$. This mapping effectively maps only the position of the leader to the follower state space. For the follower agents it holds $\hat{z}_i(t) = \hat{x}_i(t)$, $i \in N_{-l}$.

When the communication is without losses, all follower agents have access to $\hat{z}_i(jt - 1)$ for all $i \in N$, where all data

arrived between time steps $t - 1$ and t is cast as data from time step $t - 1$. However, we want that the designed control policy u is able to achieve the goal stated in Problem 1 even in the case of communication loss. In the context of this paper, the communication loss is considered as an inability of an agent to retrieve the latest shared data from another agent. More formally, the communication loss is an inability of an agent i at time step t to retrieve the shared data $\hat{z}_j(jt - k)$ from another agent j generated at time step $t - k$, where $k \geq 1$. Thus, we will impose stronger assumptions on the behaviour of the followers that will be stated in Sections IV and V.

III. CONTROL ARCHITECTURE

A. Distributed MPC Formulation

We choose to address Problem 1 using distributed MPC where each agent solves a distributed optimal control problem for a planning horizon of N time steps and applies the first control input during the control horizon Δt [1]. Note that all time steps in the control architecture are discrete $t \geq N_0$ and Δt denotes the continuous time duration between two time steps. The procedure is then repeated at every time step for each agent.

Let $\hat{x}_i(kjt)$ be the nominal state trajectory at time $k + t$, $k = 0, 1, \dots, N$ calculated at time instant t , where $\hat{x}_i(0jt) = x_i(t)$, governed by the following difference equation

$$\hat{x}_i(k + 1jt) = f_i(\hat{x}_i(kjt), u_i(kjt)) \quad (5)$$

Each agent is provided with the state reference trajectory at time step t until $t + N$ which is given as $x_{r,i}(jt) = [x_{r,i}(0jt), \dots, x_{r,i}(Njt)]^T$. The references are provided by the proposed algorithm which will be elaborated in Sections V and IV. The control objective of each agent at time step t is to minimize the following cost function

$$J_i(\hat{x}_i(jt), u_i(jt), x_{r,i}(jt), N, t) = \sum_{k=0}^N k \hat{x}_i(kjt)^T Q_i x_{r,i}(kjt) \quad (6)$$

while respecting the constraints, where the summand represents the stage cost and Q_i is a positive-definite weighting matrix. We denote also $J = \sum_{i \geq N} J_i$.

We formulate the distributed optimal control problem with respect to the objective.

Problem 2. *Let the states of the agents at time t be $x_i(t)$, $i \geq N$. Given the references $x_{r,i}(jt)$ and the predicted trajectories of other agents $\hat{z}_j(jt)$, $j \geq N - i$, the distributed optimal control problem is formulated as*

$$\min_{u_i(jt)} J_i(\hat{x}_i(jt), u_i(jt), x_{r,i}(jt), N, t) \quad (7a)$$

subject to

$$\hat{x}_i(k + 1jt) = f_i(\hat{x}_i(kjt), u_i(kjt)), \quad (7b)$$

$$\hat{x}_i(kjt) \geq X_i, \quad (7c)$$

$$u_i(kjt) \geq U_i, \quad (7d)$$

$$\hat{x}_i(kjt) \geq X_{i,j}(\hat{z}_j(kjt)), \quad \text{for } i \geq N - i, j \geq N - i, \quad (7e)$$

for $k = 0, 1, \dots, N$, for all $i \geq N$.

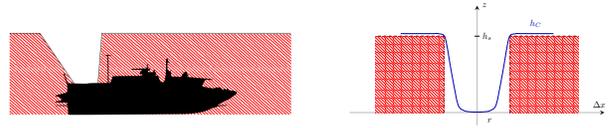


Fig. 1. Restricted area and side view of the constraint h_C from (8)

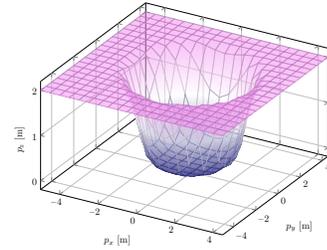


Fig. 2. h_C from (8), for $h_s = 2, r = 2.5, \beta = 1$

Set X_i denotes the set of model state constraints and U_i the input constraints. $X_{i,j}(\hat{z}_j(kjt))$ is the set of spatiotemporal safety constraints with respect to the other agents that will be formulated in the next section. Note that the leader is not subjected to the inter-agent collision constraints (7e), but only the follower agents.

B. Spatiotemporal Safety Constraints

The desired specification on UAV movements in the 3D space are to fly above some prescribed height and avoid obstacles. However, in the problem of UAV landing on a boat, we are particularly interested in the obstacles that arise from the boat shape, equipment and its movement. These specific obstacles surrounding the landing area on the boat are depicted in Figure 1.

This area can be modeled as two convex areas with a binary variable that depends on the altitude of the UAV and determines which of the two constraints should be enforced within the mixed-integer optimization problem [18]. However, it is of interest to model the restricted flying area with one nonlinear continuous function such that the usage of mixed-integer programming is avoided due to the computational burden it can cause.

In the following considerations, we assume the landing platform is circular with the radius r , $r_{safe} < r$ is the safety radius needed for safe landing and h_s is the safety height below which the UAV is not allowed to descent unless above the landing platform. Let $z_l = [p_{x_c}, p_{y_c}, 0_{n_f}^T, 2]^T \geq X_f$ be the center of the boat landing platform and $x_f = [p_x, p_y, p_z, 0_{n_f}^T, 3]^T \geq X_f$ be the position of the UAV.

We define the constraint function h_C in a novel way as

$$h_C(x_f, z_l) := p_z \frac{h_s}{1 + e^{-\beta((p_x - p_{x_c})^2 + (p_y - p_{y_c})^2 - r^2)}} \quad (8)$$

where $\beta > 0$ is a tuning parameter that determines the slope of the funnel.

Requiring that $h_C(x_f, z_l) = 0$, the UAV will always be above the boat position-dependent constraint function. The boundary of the restricted area defined with h_C in 3D is visible on Fig. 2.

The spatiotemporal constraint imposed on the position of the follower $\hat{x}_i(kjt)$, $i \in \mathcal{N}_l$ at time step $t+k$, predicted at time t , for $i = 0, 1, \dots, N$, with respect to the future trajectory of the leader position $\hat{z}_l(kjt)$ is

$$h_C(\hat{x}_i(kjt), \hat{z}_l(kjt)) = 0 \quad \text{for all } k = 0, \dots, N. \quad (9)$$

Therefore, we define the safe set $X_{i,l}(\hat{z}_l(kjt))$ in (7e) for all $i \in \mathcal{N}_l$ as

$$X_{i,l}(\hat{z}_l(kjt)) := \{ \hat{x}_i(kjt) : h_C(\hat{x}_i(kjt), \hat{z}_l(kjt)) = 0 \}. \quad (10)$$

Note that in the current formulation we assume that the height of the landing platform is zero.

C. Inter-agent Collision Avoidance

The previous section defines the safety constraint function that each follower enforces in its optimization problem with respect to the leader. The inter-agent collision avoidance condition is formulated on a set of follower agents $\mathcal{M} = \mathcal{N}_l$ as

$$kC(\hat{x}_i(kjt), \hat{z}_j(kjt)) \leq R, \quad \text{for } j \in \mathcal{M}_i \quad (11)$$

for $k = 0, 1, \dots, N$, where $R > 0$ is the minimal distance between the follower agents. The matrix $C = \text{diag}(1, 1, c, 0_{n_i-3})$, with $c > 1$, determines the shape of the super ellipsoidal set constraint such that it prevents the collision and the downwash effect that can occur if a UAV ends up below another.

Moreover, the condition (11) can be stated in form (3) as

$$h_{ij}(\hat{x}_i(kjt), \hat{z}_j(kjt)) := kC(\hat{x}_i(kjt), \hat{z}_j(kjt)) \leq R = 0$$

and the safe set $X_{i,j}(\hat{z}_j(kjt))$ in (7e) for all $i \in \mathcal{M}$ and $j \in \mathcal{M}_i$ is defined as

$$X_{i,j}(\hat{z}_j(kjt)) := \{ \hat{x}_i(kjt) : h_{ij}(\hat{x}_i(kjt), \hat{z}_j(kjt)) = 0 \}.$$

D. EKF Predictor

In order to handle the communication losses, we equip the agents with an Extended Kalman Filter (EKF) as an N -step predictor of the future trajectory of the other agents. EKF Predictor is formulated for a general nonlinear system $x^+ = f(x)$ with linearized dynamics around a measured state and unknown input. The N -step prediction is done by repeating the prediction step N times in open loop.

For brevity, we do not state the EKF matrices but refer the reader to the relevant literature [19]. The outputs of the EKF Predictor used in the control architecture are the predicted trajectories $\hat{z}_{p,i}(jt)$ and the prediction covariance matrices $P_{p,i}(jt)$. The prediction covariance matrix $P_{p,i}(kjt)$ provides us with the estimate of a super ellipsoidal set determined by its eigenvalues $\lambda_k(P_{p,i})$.

IV. RENDEZVOUS ALGORITHM

The multi-agent formulation in which we have multiple followers is challenging due to the possible inter-agent collisions even in the case when all agents share their committed trajectories. In this section, we formulate the algorithm for multiple-follower rendezvous landing.

We assume that the landing platform is large enough to accommodate M agents that can land simultaneously at pre-specified positions $c_i \in B_{r_{safe}}$ relative to the center of the landing platform. Moreover, due to the application tested in hand [20], we will constrain the scheme to M agents, and additional follower agents can be included if a prioritization procedure is included in the proposed algorithm. We state the following assumption on the initial conditions and feasibility that will be used to analyze the algorithm convergence.

Assumption 1. All agents at time t_0 have initial conditions $x_i(t_0)$, $i \in \mathcal{N}$ such that

$$x_i(t_0) \in X_{i,j}(z_j(t_0)) \quad \text{for } i \neq l, j \in \mathcal{N}_i.$$

Moreover, it holds that

$$kc_i - c_j \leq R, \quad \text{for all } i, j \in \mathcal{M}, i \neq j.$$

Before we state the algorithm, let us introduce a data collection $D_i(t)$ that is available to an agent i at time step t as $D_i(t) = \{ \hat{z}_j(jt), g_j, z_{N-i} \}$ consisting of the shared future trajectories of all other agents broadcast at time $t_j < t$, $g_j \in \mathcal{N}_i$.

Algorithm 1 Multiple-Follower Rendezvous Algorithm

Require: initial states $x_i(0)$ at time $t = 0$, landing locations c_i according to Assumption 1 for $i \in \mathcal{M}$, a tolerance parameter ε ;

- 1: **for** each agent $i \in \mathcal{M}$ **do**
 - 2: update $D_i(t)$ and $\hat{z}_j(jt)$, $j \in \mathcal{N}_i$ using (13)
 - 3: $\hat{x}_i(jt), \hat{u}_i(jt)$ solve Problem 2
 - 4: **if** $k\hat{x}_i(t) - \hat{z}_l(t) + c_i \leq \varepsilon$ **then**
 - 5: apply $\hat{u}_i(0jt)$
 - 6: broadcast $\hat{x}_i(jt)$
 - 7: $t = t + 1$
-

Remark 1. An issue that can occur in the multi-agent case with inter-agent collision avoidance in general, is that the agents can end up in a deadlock and be prevented to effectively find a way to navigate to the goal position. In that case, the deadlock can be resolved by forcing the agents to solve Problem 2 sequentially [2], [21]. This would guarantee that each agent takes into account the current predicted trajectory of other agents and waits until the process is completed. Thus, the generated trajectories will not end up in a deadlock. Note that communication in this case is required. Therefore, Algorithm 1 requires that the first iteration of the algorithm is done in a sequential manner and that the initial feasibility is established.

V. CONVERGENCE

Let us consider the one-follower case of the rendezvous landing problem, $N = fl, fg$, in which the leader reference $x_{r,l}(t)$ is given, and the follower reference is based on the position of the leader $x_{r,f}(t) = z_l(t)$ that corresponds to the center of the landing platform. In order to analyze system behavior, we introduce an assumption on the follower's capability to track the leader with respect to the leader dynamics and spatiotemporal constraints.

Assumption 2. There exists a control law $\kappa : X_f \times X_f \rightarrow U_f$ such that

$$\|kx_f^+ - z_l^+ - k^2 \rho kx_f - z_l k^2\| \leq \rho \quad (12)$$

with $\rho \in (0, 1)$ and

$$\begin{aligned} x_f^+ &= f_f(x_f, \kappa(x_f, z_l)) \\ x_l^+ &= f_l(x_l, u_l), \\ z_l &= Hx_l, \\ h_C(x_f, z_l) &= 0 \end{aligned}$$

for all $x_f \in X_f, x_l \in X_l$, and $u_l \in U_l$.

Assumption 2 states that for any control action the leader takes, there exists a control law for the follower that will reduce the distance between them in every consecutive time step. However, in the case of communication losses, the follower must be capable to asymptotically reduce the distance regardless of the leader's behaviour.

Furthermore, Assumption 2 is similar to Assumptions 4 and 6 in [22] from which the notion of incremental stability as in [23, Def. 2.1], [24] can be elaborated. Compared to [22], the reference z_l in our case has different dynamics than the followers' and it does not take into account the control action of the leader.

Referring to Problem 2, let the value function at time step t be

$$\begin{aligned} V_N(x_f(t), z_l(t)) &= \min_{u_f(jt)} J_f(\hat{x}_f(jt), u_f(jt), \hat{z}_l(jt), N, t) \\ &= \sum_{k=0}^{N-1} \|\hat{x}_f(kjt) - \hat{z}_l(kjt)\|_{Q_f}^2 \end{aligned}$$

We define the region of attraction $X_f^{ROA}(z_l) := \{x_f \in X_f : V_N(x_f, z_l) \leq V_{N, \max}\}$ of the MPC controller as the set of states which can be steered to the desired leader state z_l in N or fewer steps.

The convergence result for the case of one follower is based on ensuring that for all initial states in the region of attraction, the value function is a Lyapunov function decreasing at each time step.

Theorem 1 (Convergence with one follower). *Let Assumption 2 hold. For any $V_{N, \max} \in \mathbb{R}_{>0}$, there exist constants $\gamma < 1$, and $N_0 \in \mathbb{N}$, such that for all $N > N_0$ and all initial conditions in the region of attraction $x_f(0) \in X_f^{ROA}(z_l(0))$, there exists $\alpha_N \in \mathbb{R}_{>0}$ such that the multi-agent system (1), with $w_i = 0, i \in \{1, \dots, N\}$, satisfies*

$$\begin{aligned} \|kx_f - z_l k_{Q_f}^2 - V_N(x_f, z_l) - \gamma kx_f - z_l k_{Q_f}^2\| \\ V_N(x_f^+, z_l^+) - V_N(x_f, z_l) \leq \alpha_N kx_f - z_l k_{Q_f}^2, \end{aligned}$$

for all $t \geq 0$. Furthermore, for all $x_f(0) \in X_f^{ROA}(z_l(0))$ the follower converges to the leader-dependent rendezvous location exponentially, i.e. $\|x_f(t) - z_l(t)\| \leq \rho^t$.

The proof is given in Appendix A.

Remark 2. Note that the optimization problem in Prob. 2 does not use the terminal ingredients and thus they are not

used in Theorem 1, although it is a common way to prove the stability of MPC scheme [25], [26]. In this work, we avoid usage of the terminal ingredients by considering a sufficiently long planning horizon N in the region of attraction X^{ROA} . This builds upon the methodology suggested in [22], [27]. Moreover, the initial feasibility in the region of attraction is implicitly assumed in Theorem 1 by the same principle, i.e. by assuming that there exists N_0 for which the optimization problem in Prob. 2 is feasible for all initial conditions in X^{ROA} . Also note that the disturbances are not considered in the theorem.

It is straightforward to show, using the same argument as in Theorem 1, that the leader agent will follow its reference trajectory as well, thus achieving the objective of Problem 1.

A. Robustness to Communication Losses

In case of communication loss, the latest available shared trajectory of the leader used as the follower reference is shifted and the missing part of the trajectory is predicted with the EKF predictor. If the latest time of arrival t_a of the shared trajectory is $t_a < t - k, 0 < k < N$, then the trajectory is shifted for k time steps and the rest is predicted as follows

$$\hat{z}_i(ljt) = \begin{cases} \hat{z}_i(l + kjt_a, i), & \text{for } l < N - k \\ \hat{z}_{p,i}(l + kjt_a, i), & \text{for } l \geq N - k \end{cases} \quad (13)$$

for $l = 0, 1, \dots, N$. However, the predicted trajectory has some uncertainty such that $z_i(ljt) \in P_i(ljt)$ where the set $P_i(ljt) := \{b : kb k_{P_{p,i}(ljt)}^2 \leq s\}$ is determined by the prediction covariance matrix $P_{p,i}(ljt)$ and a parameter $s = 2 \ln(1 - p)$, that depends on the chosen probability confidence $p \in (0, 1)$. In case $k = N - 1$ which means that no predicted future steps are available, the data collection is updated only with the EKF Predictor.

Remark 3. Note that the next state uncertainty can be estimated without using EKF Predictor as $z_i(l + 1jt) \in B_{r_i}(ljt)$ where the choice of the safety radius r_i determines the conservativeness of the used set estimates. By setting r_i to

$$r_i = \max_{x_i \in X_i, u_i \in U_i} \|kx_i^+ - f(x_i, u_i)k\| \quad (14)$$

one can guarantee that the next state is within the ball of the given radius. However, this is an overly conservative approach given that the uncertainty sets grow and can become very large at the end of the horizon thus preventing the follower agents to land. In that case, the collision checking can be restricted only for the one-step ahead prediction, i.e.

$$\|k\hat{x}_i(1jt) - \hat{x}_j(1jt)k\| \leq R + r_j, \quad \text{for } j \in \{1, \dots, i\}$$

and use the worst case radius r_j as in (14).

Theorem 2 (p -probabilistically safe landing). *Let the conditions of Theorem 1 hold. Given the probabilistic confidence $p \in (0, 1)$, the radius of the landing platform r , and the*

radius necessary for the safe landing r_{safe} , if the following condition holds

$$r_{safe} + \sqrt{s\lambda_{\max}(P_{p,i}(t))} < r$$

where $s = 2\ln(1/p)$, then the landing is considered as probabilistically safe with probability p .

Proof: The proof is based on the worst-case estimate of the landing position. Given the probabilistic confidence p and the covariance matrix $P_{p,i}(t)$, the worst-case distance from the actual landing position $z_l(t)$ and its estimate $\hat{z}_l(t)$ is $d = \sqrt{s\lambda_{\max}(P_{p,i}(t))}$. Thus, $\|z_l(t) - \hat{z}_l(t)\| = d < r - r_{safe}$ which means that the follower applying the control input obtained with Prob. 2 is guaranteed to land inside of the landing platform with probability p . \square

The extension to the multiple-follower case is given by the following result:

Corollary 1. *Let Assumption 2 and 1 hold. Moreover, let the conditions of Theorem 2 hold for all follower agents $i \geq 1$. Then, Algorithm 1 converges and all follower agents meet on the leader landing platform without a collision.*

Proof: The proof is based on two parts. First, that the all agents converge to the landing platform and second, that their trajectories are collision-free. Given that all stated conditions hold there exists a feasible configuration for all follower agents to rendezvous on the leader landing platform. Moreover, all follower agents also satisfy the conditions from Theorem 2, and thus there exist a feasible landing trajectory robust to communication losses. Because each agent is solving the optimization problem in Problem 2, the inter-agent collision avoidance is enforced in every feasible landing trajectory. Therefore, all agents rendezvous on the leader landing platform without a collision. \square

VI. SIMULATION RESULTS

In this section we present a landing scenario with $M = 6$ agents. The leader is unable to communicate with the follower agents and thus follower agents must use EKF Predictor to estimate the position of the leader. Moreover, the leader measurements are taken from real-world experiments and thus have disturbances. The follower agents share their predicted trajectories until time step $k = 10$ (continuous time $t_c = 2s$) when one of the agents (Agent f_1) also loses the communication with the rest of the agents. Then the rest of the agents in the scenario must also predict the future trajectory of Agent f_1 and Agent f_1 predicts the trajectories of all other agents in the scenario.

The state and input constraints on the models are defined similarly to [28, Sec.5.1]. The initial positions of follower agents are $x_i(0) = [5 \cos(2i\pi/M), 5 \sin(2i\pi/M), 10, 0_6^T]^T$, and the leader is at origin $x_l(0) = 0_6$. The radius of the whole landing platform for all agents is $5r_{safe}$, and the safe radius for landing is $r_{safe} = 0.5m$. The matrices $Q_{f_i} = \text{diag}(10, 10, 5, 1, 1, 1, 1, 1, 1)$, $Q_l = \text{diag}(10, 10, 10, 1, 1, 1)$, thus primary penalizing the position in x and y and then in z for quadrotors, and orientation ψ for the boat. $\lambda_{\max}(Q_{f_i}) =$

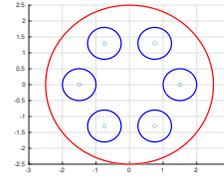


Fig. 3. The red circle marks the boundary of the landing platform, while the blue circles are the landing locations with safety radius $r_{safe} = 0.5m$. The distance from the center of the platform to a landing location is $3r_{safe}$.

10, and we pick $V_{N,max} = 240$ such that $\bar{\gamma} = 1.99$ and $N = 20$, and the region of attraction is large enough and encompass the initial displacement with a sufficient margin.

The landing locations are equidistantly distributed as on Fig. 3 and assigned to an agent positioned at the opposite side diagonally. Assumption 1 is satisfied with $R = 2r_{safe}$, and $C = I_3$. The landing is considered safe if the conditions of Theorem 2 hold with $p = 0.95$. Assuming that all agents will have identical estimation of the landing platform position and inter-agent collisions are handled it is sufficient to consider the safety with respect to the outer boundary. Thus, with $p = 0.95$, $\lambda_{\max}(P_{p,l}(t)) < (5r_{safe} - 3r_{safe})^2/s = 0.17$. From the experimental results, $\lambda_{\max}(P_{p,l}(t)) < 0.06$ thus satisfying Theorem 2. Algorithm 1 and Prob. 2 are implemented with CasADi [29] and results are shown on Figures 4 and 5. All trajectories are collision-free, and the predicted trajectory of Agent f_1 by other agents and vice versa do not induce much conservativeness to Algorithm. This is mainly because the first part of the trajectory is generated using the shift mechanism as in (13) and the small eigenvalues of the covariance matrices compared to the considered safety radii.

VII. CONCLUSION

In this paper, we presented a rendezvous algorithm based on the leader-follower scheme and distributed MPC with robustness to communication losses. The algorithm is designed for autonomous landing of multiple quadrotors on moving unmanned surface vehicles. The convergence analysis of the algorithm is presented and the effectiveness of the proposed algorithm is demonstrated with the simulation of a landing scenario. In the future work, we aim to include the disturbances in the analysis and quantify the upper bounds such that the convergence is preserved. Moreover, it will be interesting to include learning methods in order to speed-up the computation.

APPENDIX A: PROOF OF THEOREM 1

Proof: The proof proceeds in two parts. The first part shows the boundedness of $V_N(x_f(t), z_l(t))$ for all $t \geq 0$, and the second part proves that the value function decreases for all states in the region of attraction at every time step.

By the definition of the value function

$$\begin{aligned} V_N(x_f(t), z_l(t)) &= k\hat{x}_f(0|t) - \hat{z}_l(0|t)k_{Q_f}^2 \\ &= kx_f(t) - z_l(t)k_{Q_f}^2 \end{aligned}$$

Moreover, for $x_f(t) \in \mathcal{X}_f^{ROA}(z_l)$

$$kx_f(t) - z_l(t)k_{Q_f}^2 \leq V_N(x_f(t), z_l(t)) \leq V_{N,max}$$

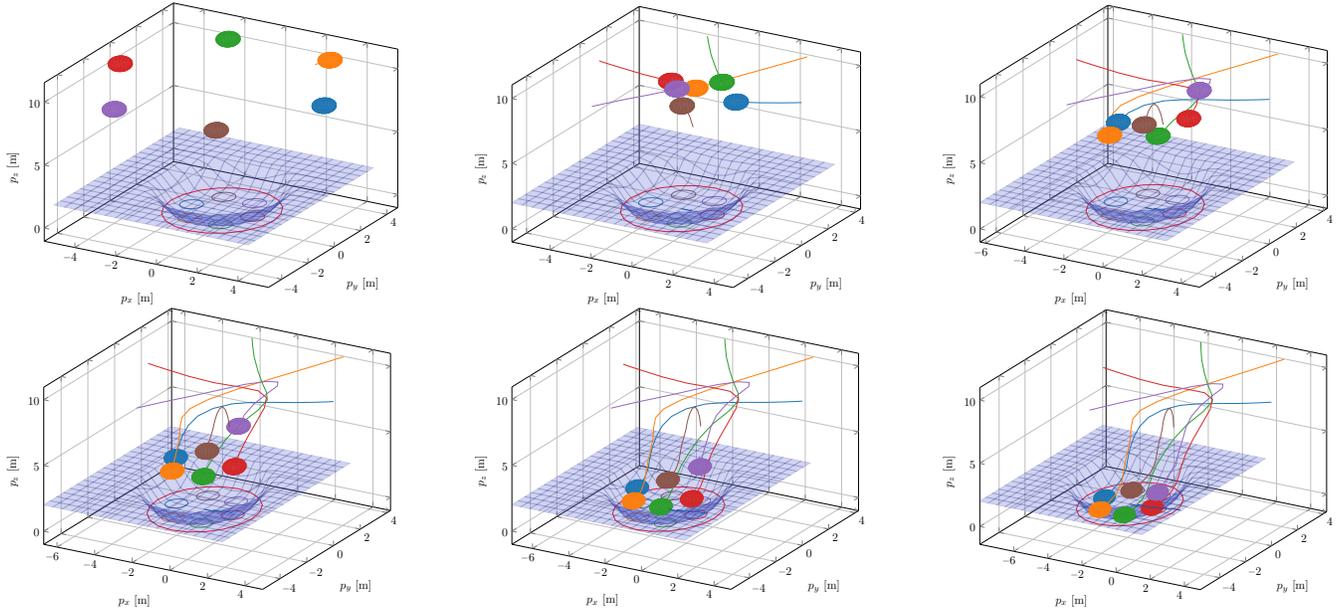


Fig. 4. 3D view of a landing scenario at discrete time steps $t = \{0, 5, 10, 13, 17, 23\}$ with $\Delta t = 0.2s$ between two time steps. The followers are unable to communicate with the leader and a loss of communication with Agent f_1 occurs during the experiment at $t = 10$. Agent f_1 is marked in orange.

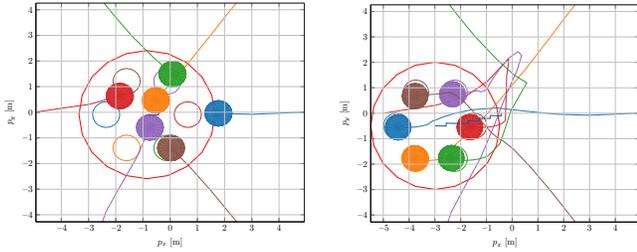


Fig. 5. Left: Top view of the second situation at $t = 5$, when it might look from the perspective view that the agents are too close, shows that the inter-agent collision avoidance constraint is enforced. Right: Final situation at $t = 23$. h_C is removed for better visibility.

thus, there exists $\bar{\gamma} \geq 1$ for which $\bar{\gamma} k x_f(t) - z_l(t) k_{Q_f}^2 = V_{N,max}$ and $\gamma \leq \bar{\gamma} \leq 1$ such that

$$V_N(x_f(t), z_l(t)) \leq V_{N,max} - \gamma k x_f(t) - z_l(t) k_{Q_f}^2$$

The second part of the proof uses the contraction of the error from Assumption 2. Let a feasible (suboptimal) input sequence for the next time step be $\tilde{u}_f(jt+1)$, defined as

$$\tilde{u}_f(kjt+1) = \begin{cases} u_f(k+1jt) & \text{for } k = 0, 1, \dots, N-1 \\ \kappa(\hat{x}_f(Njt), \hat{z}_l(Njt)) & \text{for } k = N \end{cases}$$

consisting of the shifted optimal input from the previous step and some appended $u_f^N = \kappa(\hat{x}_f(Njt), \hat{z}_l(Njt)) \in U_f$ that satisfies Assumption 2. Assuming there are no disturbances, then $x_f(t+1) = \tilde{x}_f(0jt+1) = f_f(x_f(t), u_f(0jt))$ and

$$\tilde{x}_f(kjt+1) = \begin{cases} \hat{x}_f(k+1jt) & \text{for } k < N \\ f_f(\hat{x}_f(Njt), u_f^N) & \text{for } k = N \end{cases}$$

Moreover, due to Assumption 2

$$k \tilde{x}_f(Njt+1) - \hat{z}_l(Njt+1) k^2 - \rho \|\hat{x}_f(Njt) - \hat{z}_l(Njt)\|^2$$

and

$$k \tilde{x}_f(Njt+1) - \hat{z}_l(Njt+1) k_{Q_f}^2 - \rho \frac{\lambda_{max}(Q_f)}{\lambda_{min}(Q_f)} \|\hat{x}_f(Njt) - \hat{z}_l(Njt)\|_{Q_f}^2$$

Because $V_N(x_f(t), z_l(t)) \leq \gamma k x_f(t) - z_l(t) k_{Q_f}^2$, then $\exists k \geq 1$, there exist $N > 1$ and $\bar{\gamma} \geq 1$ such that

$$k \tilde{x}_f(kjt) - \hat{z}_l(kjt) k_{Q_f}^2 \leq \frac{V_N(x_f(t), z_l(t))}{N} - \frac{\gamma}{N} k x_f(t) - z_l(t) k_{Q_f}^2$$

Let us consider the the value function at time step $t+1$, then

$$\begin{aligned} & V_N(x_f(t+1), z_l(t+1)) \\ &= \sum_{k=0}^N k \tilde{x}_f(kjt+1) - \hat{z}_l(kjt+1) k_{Q_f}^2 \\ &= \sum_{k=0}^{N-1} k \tilde{x}_f(kjt+1) - \hat{z}_l(kjt+1) k_{Q_f}^2 \\ &\quad + k \tilde{x}_f(Njt+1) - \hat{z}_l(Njt+1) k_{Q_f}^2 \\ &= \sum_{k=1}^N \|\hat{x}_f(kjt) - \hat{z}_l(kjt)\|_{Q_f}^2 \\ &\quad + k \tilde{x}_f(Njt+1) - \hat{z}_l(Njt+1) k_{Q_f}^2 \end{aligned}$$

$$\begin{aligned}
&= \sum_{k=0}^N \left\| \hat{x}_f(k/t) \quad \hat{z}_l(k/t) \right\|_{Q_f}^2 \quad \left\| \hat{x}_f(0/t) \quad \hat{z}_l(0/t) \right\|_{Q_f}^2 \\
&\quad + k\tilde{x}_f(N/t + 1) \quad \hat{z}_l(N/t + 1) k_{Q_f}^2 \\
&= V_N(x_f(t), z_l(t)) \quad kx_f(t) \quad z_l(t) k_{Q_f}^2 \\
&\quad + k\tilde{x}_f(N/t + 1) \quad \hat{z}_l(N/t + 1) k_{Q_f}^2 \\
&\quad V_N(x_f(t), z_l(t)) \quad kx_f(t) \quad z_l(t) k_{Q_f}^2 \\
&\quad + \rho \frac{\gamma}{N} \frac{\lambda_{max}(Q_f)}{\lambda_{min}(Q_f)} kx_f(t) \quad z_l(t) k_{Q_f}^2 \\
&\quad V_N(x_f(t), z_l(t)) \quad \alpha_N kx_f(t) \quad z_l(t) k_{Q_f}^2
\end{aligned}$$

where $\alpha_N := 1 - \rho \frac{\gamma}{N} \frac{\lambda_{max}(Q_f)}{\lambda_{min}(Q_f)}$. Thus by choosing $N > N_0 := \frac{\bar{\gamma} \lambda_{max}(Q_f)}{\lambda_{min}(Q_f)}$, $\alpha_N > 0$. Finally, the value function $V_N(x_f(t), z_l(t))$ is decreasing for all $t \in [0, N]$.

Using the decrease property and the boundedness in the region of attraction $X_f^{ROA}(z_l)$ proven in the first part, $V_N(x_f(t), z_l(t))$ is a Lyapunov function in $X_f^{ROA}(z_l)$. Thus, the error $kx_f(t) \quad z_l(t)k$ exponentially goes to zero, which concludes the proof. \square

REFERENCES

- [1] Matthias A Müller, Marcus Reble, and Frank Allgöwer. Cooperative control of dynamically decoupled systems via distributed model predictive control. *International Journal of Robust and Nonlinear Control*, 22(12):1376–1397, 2012.
- [2] Panagiotis D Christofides, Riccardo Scattolini, David Munoz de la Pena, and Jinfeng Liu. Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering*, 51:21–41, 2013.
- [3] Christos K Verginis, Alexandros Nikou, and Dimos V Dimarogonas. Communication-based decentralized cooperative object transportation using nonlinear model predictive control. In *2018 European control conference (ECC)*, pages 733–738. IEEE, 2018.
- [4] Brett T Lopez, Jean-Jacques Slotine, and Jonathan P How. Robust collision avoidance via sliding control. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2962–2969. IEEE, 2018.
- [5] Noel Csomay-Shanklin, Andrew J Taylor, Ugo Rosolia, and Aaron D Ames. Multi-rate planning and control of uncertain nonlinear systems: Model predictive control and control lyapunov functions. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 3732–3739. IEEE, 2022.
- [6] Ugo Rosolia, Andrew Singletary, and Aaron D Ames. Unified multirate control: From low-level actuation to high-level planning. *IEEE Transactions on Automatic Control*, 67(12):6627–6640, 2022.
- [7] Davide Falanga, Alessio Zanchettin, Alessandro Simovic, Jeffrey Delmerico, and Davide Scaramuzza. Vision-based autonomous quadrotor landing on a moving platform. In *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*.
- [8] Yeongho Song, Hojin Lee, Cheolhyeon Kwon, Hyo-Sang Shin, and Hyondong Oh. Distributed estimation of stochastic multiagent systems for cooperative control with a virtual network. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2022.
- [9] Ioannis Lymperopoulos and John Lygeros. Adaptive aircraft trajectory prediction using particle filters. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, page 7387, 2008.
- [10] Spyros Maniatopoulos, Dimos V Dimarogonas, and Kostas J Kyriakopoulos. A decentralized event-based predictive navigation scheme for air-traffic control. In *2012 American Control Conference (ACC)*, pages 2503–2508. IEEE, 2012.
- [11] Panna Felsen, Patrick Lucey, and Sujoy Ganguly. Where will they go? predicting fine-grained adversarial multi-agent motion using conditional variational autoencoders. In *Proceedings of the European conference on computer vision (ECCV)*, pages 732–747, 2018.
- [12] Jiachen Li, Hengbo Ma, and Masayoshi Tomizuka. Interaction-aware multi-agent tracking and probabilistic behavior prediction via adversarial learning. In *2019 international conference on robotics and automation (ICRA)*, pages 6658–6664. IEEE, 2019.
- [13] Tianyang Zhao, Yifei Xu, Mathew Monfort, Wongun Choi, Chris Baker, Yibiao Zhao, Yizhou Wang, and Ying Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12126–12134, 2019.
- [14] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. Evolve-graph: Multi-agent trajectory prediction with dynamic relational reasoning. *Advances in neural information processing systems*, 33:19783–19794, 2020.
- [15] Yulong Cao, Chaowei Xiao, Anima Anandkumar, Danfei Xu, and Marco Pavone. Advdo: Realistic adversarial attacks for trajectory prediction. In *European Conference on Computer Vision*, pages 36–52. Springer, 2022.
- [16] Dženan Lapandić, Christos K Verginis, Dimos V Dimarogonas, and Bo Wahlberg. Robust trajectory tracking for underactuated quadrotors with prescribed performance. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 3351–3358. IEEE, 2022.
- [17] Thor I Fossen. *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.
- [18] Linnea Persson and Bo Wahlberg. Model predictive control for autonomous ship landing in a search and rescue scenario. In *AIAA Scitech 2019 Forum*, page 1169, 2019.
- [19] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.
- [20] Olov Andersson, Patrick Doherty, Mårten Lager, Jens-Olof Lindh, Linnea Persson, Elin A Topp, Jesper Tordenlid, and Bo Wahlberg. WARA-PS: a research arena for public safety demonstrations and autonomous collaborative rescue robotics experimentation. *Autonomous Intelligent Systems*, 1(1):1–31, 2021.
- [21] Arthur Richards and Jonathan P How. Robust distributed model predictive control. *International Journal of control*, 80(9):1517–1531, 2007.
- [22] Johannes Köhler, Matthias A Müller, and Frank Allgöwer. Nonlinear reference tracking with model predictive control: An intuitive approach. In *2018 European Control Conference (ECC)*, pages 1355–1360. IEEE, 2018.
- [23] David Angeli. A lyapunov approach to incremental stability properties. *IEEE Transactions on Automatic Control*, 47(3):410–421, 2002.
- [24] Duc N Tran, Björn S Rüffer, and Christopher M Kellett. Incremental stability properties for discrete-time systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 477–482. IEEE, 2016.
- [25] David Q Mayne, James B Rawlings, Christopher V Rao, and Pierre OM Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.
- [26] Hong Chen and Frank Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.
- [27] Andrea Boccia, Lars Grüne, and Karl Worthmann. Stability and feasibility of state constrained mpc without stabilizing terminal constraints. *Systems & control letters*, 72:14–21, 2014.
- [28] Dženan Lapandić, Linnea Persson, Dimos V. Dimarogonas, and Bo Wahlberg. Aperiodic communication for mpc in autonomous cooperative landing. *IFAC-PapersOnLine*, 54(6):113–118, 2021. 7th IFAC Conference on Nonlinear Model Predictive Control NMPC 2021.
- [29] Joel A E Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36, 2019.