IEEE Control Systems Letters paper presented at
2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

# Design of a Distributed Switching Model Predictive Control for Quadrotor UAVs Aggregation

Chrystian Pool Edmundo Yuca Huanca, Gian Paolo Incremona and Patrizio Colaneri

*Abstract*— **This letter proposes a novel distributed model predictive control (MPC) strategy to address the swarm aggregation of a team of quadrotor unmanned aerial vehicles (UAVs). First, a switched formulation of the quadrotor model is derived by mapping the UAVs dynamics into a set of finite motion modes. Then, relying on a suitably selected control Lyapunov function (CLF), the inter-agent collisions and the aggregation task are taken into account to design a switching MPC (SMPC) strategy. A clustering method is also introduced to define the communication network among the agents, which is essential to sequentially solve the optimal control problem. Finally, the efficacy of the proposal, also in comparison with other methodologies, is satisfactorily shown in simulation.**

*Index Terms*— **Switched systems, model predictive control, quadrotor UAVs, distributed control, consensus.**

## I. Introduction

In recent years, thanks to the non-stop advances in communication, sensing and processing technology, the application domain of quadrotor unmanned aerial vehicles (UAVs) has grown in several fields from military purposes [1], to agriculture [2], to racing [3], among many others. Their main attractiveness relies on autonomously performing activities in scenarios where human integrity might be compromised. However, and despite the enormous research on single UAV flight control, the use of multiple UAVs in an organized swarm has demonstrated to improve the single performance of the agents as well as the efficiency of the entire group.

In the literature, many works were devoted to address the design of controllers for UAVs swarms. In [4], for instance, a multicopter geometrically constrained trajectory planning is investigated. Among decentralized solutions, a hybrid control algorithm to achieve swarm formation is proposed in [5], while in [6] a decentralized model predictive control (MPC), defined by inter-agent bearings to avoid collisions, is presented. Among results with distributed topology, instead, a consensus based control strategy for swarms of UAVs under a time-varying topology for flight formation, swarm tracking, and social foraging is presented in [7]. A distributed trajectory optimization algorithm for safe multi-agent trajectory planning is proposed in [8], while an observer-based optimal consensus controller for a multi-UAVs system, which integrates linear quadratic regulator technique and linear matrix inequalities, is introduced in [9]. In the stochastic framework, [10] discusses a chance-constraint formulation to account for position and velocity uncertainty, while achieving obstacles avoidance.

Having in mind multi-UAVs aggregation, localization requires continuous sensor information. However, computational and communication limitations make some simplifications mandatory to reduce the curse of dimensionality. Spurred by these motivations, this letter takes inspiration from [11], [12], where a differential wheeled robot dynamics is constrained to only two motion modes, and from [13], which proposes an order-reduction of a remote controlled car model, recasting it as a hybrid system. Indeed, all these works address the previously mentioned problems, which arise together when a swarm of agents with high order dynamics is studied, e.g., an UAVs swarm, as in this letter. The main purpose of this article is to overcome these challenges by proposing a novel optimization-based strategy, exploiting switched systems, in order to solve a swarm aggregation problem. Indeed, aggregation processes are very common, e.g., in biological systems, and they often are a needed prerequisite for many collective systems for accomplishing other cooperative tasks.

Specifically, this letter proposes an original swarm aggregation strategy, that combines the advantages of switched systems with those of distributed MPC. Differently from [11], [12], where only two switching planar modes are considered, the proposed aggregation strategy is extended to the tridimensional case. Moreover, while [12] presents a control Lyapunov function (CLF) based approach, in this letter, starting from a suitable CLF, a novel distributed switching MPC (SMPC) is proposed, capable of dealing with an arbitrary number of modes capturing the UAVs motion, and employing a clustering method to define the data exchange among agents. This allows achieving both optimal control performance and computational simplifications for the aggregation algorithm, which has to face the issue that it needs to scale well with the swarm population, as finally assessed in simulation on a large-scale case study.

## II. Modelling and Problem Formulation

This section introduces the considered quadrotor UAV model and the problem formulation.

### A. Quadrotor UAV model

Here, we focus on a general aggregation problem in which the UAVs could collide among each other. Since in practice, in this case, it is convenient to provide velocity references to

This work was supported by the Italian Ministry for Research in the framework of the 2017 Program for Research Projects of National Interest (PRIN) under Grant 2017YKXYXJ.

Chrystian Pool Edmundo Yuca Huanca, Gian Paolo Incremona and Patrizio Colaneri are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy (e-mails: *chrystianpool.yuca@polimi.it*, *gianpaolo.incremona@polimi.it*, *patrizio.colaneri@polimi.it*).

the internal control loops, the differential kinematic model of the UAVs is hereafter considered. Moreover, we assume that a local replanner is used to allow smoothness in velocity and acceleration (see, e.g., [14]) even in the case of switching policies, with local controllers capable of perfectly tracking the desired velocities.

Consider Fig. 1, and let $[p_x\ p_y\ p_z\ \phi\ \theta\ \psi]^\top$ be the vector containing the linear and angular position of the quadrotor in the world frame $O_\mathrm{W}-X_\mathrm{W}Y_\mathrm{W}Z_\mathrm{W}$, and $[v_u\ v_v\ v_w\ w_p\ w_q\ w_r]^\top$ be the vector containing the linear and angular velocities in the body frame $O_\mathrm{B}-X_\mathrm{B}Y_\mathrm{B}Z_\mathrm{B}$. In the following, the dependence of all the variables on time will be omitted when obvious.

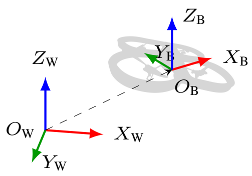


Fig. 1. Schematic rendering of the quadrotor UAV's frames.

Relying on Euler equations for the spatial motion of a rigid body, the relation between the body and the world frame is

$$v = R_v v_\mathrm{B}, \tag{1a}$$

$$w = R_w w_\mathrm{B}, \tag{1b}$$

where $v := [\dot{p}_x\ \dot{p}_y\ \dot{p}_z]^\top$, $w := [\dot{\phi}\ \dot{\theta}\ \dot{\psi}]^\top$, $v_\mathrm{B} := [v_u\ v_v\ v_w]^\top$, $w_\mathrm{B} := [w_p\ w_q\ w_r]^\top$, while

$$R_v := \begin{bmatrix} c(\theta)c(\psi) & s(\phi)s(\theta)c(\psi) - c(\theta)s(\psi) & c(\phi)s(\theta)c(\psi) + s(\phi)s(\psi) \\ c(\theta)s(\psi) & s(\phi)s(\theta)s(\psi) + c(\theta)c(\psi) & c(\phi)s(\theta)s(\psi) - s(\phi)c(\psi) \\ -s(\theta) & s(\phi)c(\theta) & c(\phi)c(\theta) \end{bmatrix},$$

$$R_w := \begin{bmatrix} 1 & s(\phi)\tan(\theta) & c(\phi)\tan(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)c(\theta)^{-1} & c(\phi)c(\theta)^{-1} \end{bmatrix}.$$

with $c(\cdot) := \cos(\cdot)$, and $s(\cdot) := \sin(\cdot)$. Now, by posing the quadrotor pose and orientation in the state vector $x := [p_x\ p_y\ p_z\ \phi\ \theta\ \psi]^\top \in \mathbb{R}^6$, and the velocity vector in the input vector $u := [v_\mathrm{B}^\top\ w_\mathrm{B}^\top]^\top \in \mathbb{R}^6$, then the differential kinematic model (1) can be rewritten as

$$\dot{x} = f(x, u), \quad x(0) = x_0. \tag{2}$$

### B. The switched quadrotor model

Motivated by the need to reduce the curse of dimensionality when dealing with multi-quadrotor systems, we model the quadrotor dynamics (2) as a switched system, switching among a finite set of operating modes. Let $m$ be the number of predefined motion modes such that $\sigma \in \mathcal{M} := \{1, \dots, m\}$ is the so-called switching signal. The latter defines the current motion mode so that $u(\sigma) \in \mathcal{U}_\mathrm{sw}$ can be modelled as a switching input, with $\mathcal{U}_\mathrm{sw}$ containing the following vectors

$$u(\sigma) := \begin{cases} 0_{6\times 1}, & \sigma = 1 \\ \bar{v}\mathrm{e}_{\sigma-1}, & \sigma = 2, 3, 4 \\ \bar{w}\mathrm{e}_{\sigma-1}, & \sigma = 5, 6, 7 \\ -\bar{v}\mathrm{e}_{\sigma-7}, & \sigma = 8, 9, 10 \\ -\bar{w}\mathrm{e}_{\sigma-7}, & \sigma = 11, 12, 13 \end{cases} \tag{3}$$

where $\bar{v}$ and $\bar{w}$ are predefined longitudinal and rotational velocities, and $\mathrm{e}_\sigma \in \mathbb{R}^6$ is the canonical unit vector. Then, the switched time-invariant system corresponding to (2) is

$$\dot{x} = f_\sigma(x, u(\sigma)), \quad x(0) = x_0, \tag{4}$$

where $f_\sigma = [f_{x_\sigma}, f_{y_\sigma}, f_{z_\sigma}, f_{\phi_\sigma}, f_{\theta_\sigma}, f_{\psi_\sigma}]^\top$ belongs to the set $\mathcal{Q} := \{f_1, \dots, f_m\}$.

*Remark 2.1 (Modes selection):* Note that the modes selection is not trivial. Indeed, the set of vector fields $\mathcal{Q}$ must include at least one non-zero longitudinal velocity component in at least one of its elements, since pure angular velocities in all modes do not contribute to any displacement in the tridimensional space. ▽

For design purposes, the set of vector fields must also include a zero vector, meaning that the quadrotor is capable of keeping still. In this paper, without loss of generalization, we assume $m = 13$ motion modes, with 6 velocity vectors having only one positive non-zero component and zero for the others, 6 velocity vectors defined as the previous ones but with negative values only, and one fully zero vector. This modes selection, based on a specific motion policy and according to Remark 2.1, allows the UAVs to move in any possible direction of the world frame.

### C. Multi-UAVs switched system and problem statement

Having in mind a team of quadrotors, consider now a system composed of $n$ agents, and let $\mathcal{N} := \{i \in [1, \dots, n]\}$ be the set comprising all the agent indexes. In the following, the apex $^{[i]}$ will be used to indicate the $i$th subsystem. The multi-quadrotor system model is obtained by redefining the vectors $x$ and $u$ as $x := [x^{[1]}, \dots, x^{[n]}]^\top$ and $u := [u^{[1]}, \dots, u^{[n]}]^\top$, respectively. Analogously, a switching string $\Sigma := \{\sigma^{[1]}, \dots, \sigma^{[n]}\} \in \mathcal{S} := \mathcal{M}^n$ is introduced such that

$$\dot{x} = f_\Sigma(x, u(\Sigma)), \quad x(0) = x_0, \tag{5}$$

where $u(\Sigma) \in \mathcal{V} := \mathcal{U}_\mathrm{sw}^n$, and $f_\Sigma \in \mathcal{Q}_n := \{f_1, \dots, f_{m^n}\}$.

Therefore, given the switched model (5) constrained to $m^n$ motion modes, the control problem to solve consists of designing a control strategy capable of aggregating the quadrotors, while avoiding collisions among them, and taking into account the switching nature of the system dynamics.

## III. Control Lyapunov Function Baseline

To solve the problem formulated in §II, let us introduce the design of a CLF for the switched system (5). To streamline the exposition, we refer to [15], [16] for further details on CLFs.

Having in mind an aggregation objective for UAVs with rigid body properties, it is natural to consider for each agent the pose dynamics (1a) and the reciprocal Euclidean distance between the $i$th and $j$th centroids, i.e.,

$$d_{ij} = \sqrt{(p_x^{[i]} - p_x^{[j]})^2 + (p_y^{[i]} - p_y^{[j]})^2 + (p_z^{[i]} - p_z^{[j]})^2},$$

$\forall\, i, j \in \mathcal{N}$, $i < j$. In order to take into account the space occupancy of UAVs, let $\bar{d} \in \mathbb{R}$ be the reference between two agents, and $\underline{d} \in \mathbb{R}$ a distance lower bound. Note that

$\underline{d}$ is instrumental to manage the collision avoidance among agents, and such a parameter is selected relying on $\bar{d}$, the maximum longitudinal velocity $\bar{v}$ and, considering practical implementation, also on the sampling time $T$, such that $\ell < \underline{d} < \bar{d} - 2\bar{v}T$, with $\ell$ being the length of the UAV's frame.

Then, posing $\tilde{d}_{ij} := d_{ij} - \bar{d}$, and $\tilde{d}$ as the vector containing all the distance errors, consider the following function

$$V(\tilde{d}) := \sum_{\substack{i,j \in \mathcal{N} \\ i<j}} \underbrace{\gamma_{ij} \frac{\tilde{d}_{ij}^2}{2}}_{V_{ij}}, \tag{6}$$

with $\gamma_{ij}$ being a weight defined as

$$\gamma_{ij} := \begin{cases} 1, & d_{ij} \geq \bar{d} \\ \frac{(\bar{d}-\underline{d})^2}{(d_{ij}-\underline{d})^2}, & d_{ij} < \bar{d} \end{cases} \tag{7}$$

The weight (7) is a continuous function of $d_{ij}$, which is aimed at avoiding that the UAVs move away from $\bar{d}$. As customary in the theory of switching control [17], the Lyapunov function induced by the argmin-based switching control action is continuous but only piecewise differentiable so that the notion of the Dini derivative should be used.

*Remark 3.1 (Switching law):* Note that, in the considered CLF baseline, making the zero mode possible only when the agents aggregate, the switching law can be designed as the $\arg\min(D_+ V)$, where $D_+$ is the Dini derivative. In the case of two or more equal minima, one could select the first one in order. Typically, this kind of strategy leads to a sliding trajectory around the discontinuity points of the CLF. In our case, depending on the selected switching inputs as in (3), it is possible to show that such a derivative is negative, and zero only when the agents are aggregated. $\triangledown$

By virtue of the decreasing property of the selected CLF, it can be employed as value function for establishing stability of a finite horizon optimal control problem in discrete time-domain.

## IV. The Proposed Distributed Switching Model Predictive Control

Motivated by the reasoning of the previous section, we present now an alternative aggregation approach based on SMPC. This approach keeps the spirit of the CLF based law, and, by virtue of the guarantees mentioned above, it exploits the CLF in (6) as cost function to be minimized subject to the switched dynamics in (5).

The SMPC is designed to be solved at each time instant $k \in \mathbb{N}_0$, so that by introducing the sampling time $T > 0$, and discretizing the UAVs dynamics by using forward Euler method, one has

$$x_{k+1} = x_k + T f_{\Sigma_k}(x_k, u_k(\Sigma_k)) = f_{\mathrm{d}_{\Sigma_k}}(x_k, u_k(\Sigma_k)), \tag{8}$$

with $f_{\mathrm{d}_{\Sigma_k}}$ belonging to the set of vector fields $\mathcal{Q}_{\mathrm{d}_n} := \{f_{\mathrm{d}_1}, \ldots, f_{\mathrm{d}_{m^n}}\}$. Letting $N \geq 1$, the prediction horizon is instead defined as $\mathcal{T}_k := \{k, \ldots, k+N-1\}$, while the variable $t$ is used to span along the prediction horizon, i.e., $t \in \mathcal{T}_k$. The cost function of the SMPC, relying on the function designed in (6), is chosen as

$$J_k := \sum_{t \in \mathcal{T}_k} \sum_{\substack{i,j \in \mathcal{N} \\ i<j}} \gamma_{ij_t} \frac{\tilde{d}_{ij_t}^2}{2}. \tag{9}$$

with the weight $\gamma_{ij_t}$ as in (7). Therefore, letting $\boldsymbol{\Sigma}_k := \{\boldsymbol{\sigma}_k^{[1]}, \ldots, \boldsymbol{\sigma}_k^{[n]}\}$ and $\boldsymbol{\sigma}_k^{[i]} := [\sigma_k^{[i]}, \ldots, \sigma_{k+N-1}^{[i]}]^\top$, the SMPC formulation results

$$\begin{aligned} & \min_{\boldsymbol{\Sigma}_k} J_k \\ & \text{subject to, } \forall\, t \in \mathcal{T}_k, \\ & \quad x_{t+1} = f_{\mathrm{d}_{\Sigma_t}}(x_t, u_t(\Sigma_t)), \quad x_k = \tilde{x}_k, \\ & \quad x_t \in \mathcal{X}, \;\; u_t(\Sigma_t) \in \mathcal{V}, \end{aligned} \tag{10}$$

where $\tilde{x}_k$ expresses the measured state at each $k \in \mathbb{N}_0$. Moreover, in (10), the set of the state constraint is

$$\begin{aligned} \mathcal{X} := \{ & x \in \mathbb{R}^{6n} \mid d_{ij_t} > \underline{d}, \forall i,j \in \mathcal{N}, i<j, \\ & \wedge\; |\phi_t^{[i]}| \leq \bar{\phi}, \; |\theta_t^{[i]}| \leq \bar{\theta}, \; |\psi_t^{[i]}| \leq \bar{\psi}, \forall i \in \mathcal{N}\}, \end{aligned} \tag{11}$$

in order to avoid collisions among the quadrotors, and limit the angle variation according to positive thresholds $\bar{\phi}$, $\bar{\theta}$, $\bar{\psi}$.

It is worth highlighting that, due to the combinatorial nature of the strategy, the formulated SMPC problem (10) can be computationally demanding. More specifically, the computational complexity is indeed given by the number of evaluations of the nonlinear dynamics $f\big(x_t^{[i]}, u_t^{[i]}(\sigma_t^{[i]})\big)$ that have to be performed for computing the optimal sequence $\mathbf{u}_k^* = [u_k^*(\sigma_k^*), \ldots, u_{k+N-1}^*(\sigma_{k+N-1}^*)]$. The number of possible control sequences belonging to the set $\mathcal{W}$ is given by the cardinality $|\mathcal{W}| = m^N$, so that, for $n$ UAVs, the whole complexity is $n \cdot m^N$. As a consequence, some simplifications are introduced as follows.

*1) Switching policies:* In order to alleviate the computational burden required to solve the optimal control problem (10), the set of possible sequences $\mathcal{W}$ is reduced by introducing policies in order to determine the permitted switching between the considered modes. For example, one can impose that only one agent per time can modify its dynamics, or, depending on the task's constraints, some sequence can be discarded (e.g., employing only the heading, the forward and the vertical motions in the case of a non-holonomic way of navigation). This implies the selection of the optimal input sequence be within a subset of feasible sequences, i.e., $\mathbf{u}_k^{*[i]} \in \mathcal{F} \subset \mathcal{W}$.

*2) Clustering:* Despite the introduction of switching policies, the number of agents is still a critical parameter that compromises real-time computations and could possibly affect fault-tolerance, organizational complexity and maintenance problems. Therefore, a distributed control method is a valid solution to further reduce the complexity, locally solving the optimal control problem inside subsystems that share limited resources over a constrained communication network. Making reference to the distributed solution in [18, §5.2.1], a clustering approach is hereafter introduced.

Let $\mathcal{C}_k^{[i]}$ be the set defined by the clustering method as

$$\mathcal{C}_k^{[i]} := \{j \in \mathcal{N} \mid d_{ij} \leq r_{\mathrm{cl}},\ \forall j \neq i\}, \tag{12}$$

where $r_{\mathrm{cl}}$ is a predefined radius of the sphere cluster attached to the $i$th robot centroid. Consider now Fig. 2, where each quadrotor locally generates the the optimal input sequence $\mathbf{u}_k^{*[i]} := [u_k^*(\sigma_k^{*[i]}), \ldots, u_{k+N-1}^*(\sigma_{k+N-1}^{*[i]})]^\top$. The $i$th quadrotor receives at each sampling time the state feedback set

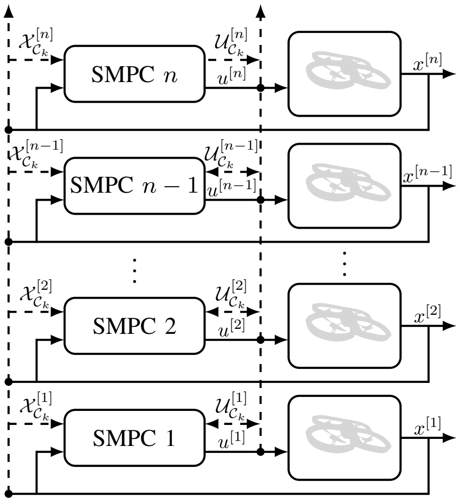

$\mathcal{X}_{\mathcal{C}_k}^{[n]}$
$\mathcal{U}_{\mathcal{C}_k}^{[n]}$
SMPC $n$
$u^{[n]}$
$x^{[n]}$
$\mathcal{X}_{\mathcal{C}_k}^{[n-1]}$
$\mathcal{U}_{\mathcal{C}_k}^{[n-1]}$
SMPC $n-1$
$u^{[n-1]}$
$x^{[n-1]}$
$\mathcal{X}_{\mathcal{C}_k}^{[2]}$
$\mathcal{U}_{\mathcal{C}_k}^{[2]}$
SMPC 2
$u^{[2]}$
$x^{[2]}$
$\mathcal{X}_{\mathcal{C}_k}^{[1]}$
$\mathcal{U}_{\mathcal{C}_k}^{[1]}$
SMPC 1
$u^{[1]}$
$x^{[1]}$

Fig. 2. Sequential distributed SMPC architecture.

$$\mathcal{X}_{\mathcal{C}_k}^{[i]} := \{x_k^{[j]} \in \mathbb{R}^6,\ \forall j \neq i,\ j \in \mathcal{C}_k^{[i]}\}, \tag{13}$$

and the predicted input set

$$\mathcal{U}_{\mathcal{C}_k}^{[i]} := \{\mathbf{u}_k^{[j]},\ \forall j \neq i,\ j \in \mathcal{C}_k^{[i]}\}, \tag{14}$$

in order to compute, according to the receding horizon principle, the input $u_k^{[i]}$ in a decreasing sequential order, according to a predefined indexing criterion. This implies that the $j$th SMPC, $j = n, \ldots, 2$, retrieves no predicted input set from subsystem $\kappa = j-1, \ldots, 1$. To solve this issue, the optimization problem feasibility is guaranteed by letting the $j$th SMPC, $j = n, \ldots, 2$, assume specific trajectories for the predicted input subset $\mathcal{U}_k^{[\kappa]}$, $\kappa = j-1, \ldots, 1$ (for instance the quadrotors are kept still along the prediction horizon).

*3) Distributed SMPC:* For each quadrotor the SMPC problem is finally formulated by defining the cost as

$$J_k^{[i]} := \sum_{t \in \mathcal{T}_k} \sum_{j \in \mathcal{C}_k^{[i]}} \gamma_{ij_t} \frac{\tilde{d}_{ij_t}^2}{2}, \tag{15}$$

which takes into account the quadrotors inside the cluster $\mathcal{C}_k^{[i]}$. The proposed SMPC problem is then stated as

$$\begin{aligned} &\min_{\boldsymbol{\sigma}_k} J_k^{[i]} \\ &\text{subject to, } \forall\, t \in \mathcal{T}_k,\ \forall\, \kappa \in \mathcal{C}_k^{[i]} \wedge \kappa = i, \\ &\quad x_{t+1}^{[\kappa]} = f_{\mathrm{d}_{\sigma_t^{[\kappa]}}}(x_t^{[\kappa]}, u_t^{[\kappa]}(\sigma_t^{[\kappa]}))\,, \quad x_k^{[\kappa]} = \tilde{x}_k^{[\kappa]}, \\ &\quad \hat{x}_t^{[i]} \in \widehat{\mathcal{X}},\ \ u_t^{[\kappa]}(\sigma_t^{[\kappa]}) \in \mathcal{U}_{\mathrm{sw}}, \end{aligned} \tag{16}$$

with $\hat{x}_t^{[i]} \in \mathbb{R}^{6|\mathcal{C}_k^{[i]}|}$ being the vector containing all the states of the agents in the cluster, while the state constraint set is

$$\begin{aligned} \widehat{\mathcal{X}} := \{\hat{x}_t^{[i]} \in \mathbb{R}^{6|\mathcal{C}_k^{[i]}|} \mid d_{ij_t} > \underline{d},\ \forall j \in \mathcal{C}_k^{[i]},\ i < j \\ \wedge\ |\phi_t^{[i]}| \leq \bar{\phi},\ |\theta_t^{[i]}| \leq \bar{\theta},\ |\psi_t^{[i]}| \leq \bar{\psi}\}. \end{aligned} \tag{17}$$

*Remark 4.1 (Sub-optimality):* The distributed SMPC solution depends on the predefined indexing criterion. Then, other sub-optimal solutions can be found by reassigning the indexes, solving an optimization problem aimed at searching for the index criterion which generates the smallest cost. ▽

*Remark 4.2 (Scalability):* It is worth highlighting that, by virtue of the clustering method and the collision avoidance mechanism, which imply a maximum number of agents belonging to the same subgroup, if the total number of quadrotors increased, this would not correspondingly affect the computational complexity to solve (16). ▽

## V. Case-Study

In this section, the proposed distributed SMPC is assessed in simulation relying on different sizes of the UAVs swarm.

### A. Settings

The simulated test benchmark consists of different scenarios to evaluate the performance of the proposed algorithm on UAVs swarms with a various population size, ranging from $n = 10$ to 150 agents. The quadrotors have a random initial distribution over a tridimensional grid in a $6 \times 6 \times 6$ m cube space and are charaterized by the parameters listed in Table I. Finally, some performance metrics are introduced.

TABLE I
Parameters.

| $\ell$ | $\bar{v}$ | $\bar{w}$ | $\bar{d}$ | $\underline{d}$ | $r_{\mathrm{cl}}$ | $N$ |
|---|---|---|---|---|---|---|
| $0.225\,\mathrm{m}$ | $0.1\,\mathrm{m\,s^{-1}}$ | $0.2\,\mathrm{rad\,s^{-1}}$ | $3.5\ell$ | $2\ell$ | $4\ell$ | 3 |

Let a cluster of UAVs be the maximal connected subgraph of the graph defined by the UAVs positions, where two agents are considered to be adjacent if another one cannot fit in between them, that is if $d_{ij} < 2\bar{d}$ [11]. Therefore, the first index is given by the ratio between the number of quadrotors within a cluster, as defined above, namely $n_{\mathcal{C}}$, and the total number of agents $n$, that is

$$\mu_{\mathrm{a}} := \frac{n_{\mathcal{C}}}{n}, \tag{18}$$

with full aggregation corresponding to $\mu_{\mathrm{a}}^* = 1$. The second index is the packing density ratio between the sum of the sphere volumes of diameter $\bar{d}$ for each agent, and the volume of the smallest sphere encapsulating the previous ones, i.e.,

$$\mu_{\mathrm{p}} := \frac{n\bar{d}^3}{\left[2\left(\max_{i \in \mathcal{N}} \left\| [p_x^{[i]}\, p_y^{[i]}\, p_z^{[i]}]^\top - b \right\|_2\right) + \bar{d}\right]^3}, \tag{19}$$

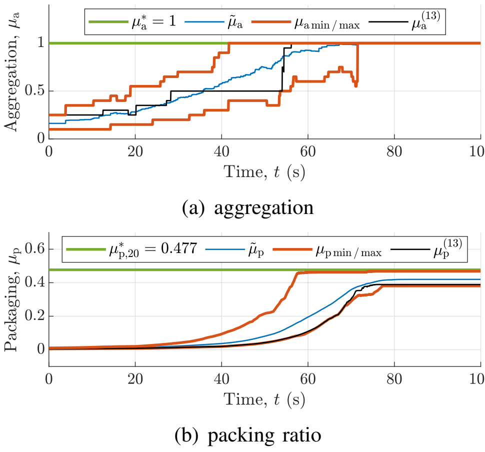
where $b$ is the barycentre of the network. Note that the sphere packing problem has been widely studied in the literature, see e.g., [19], and optimal density values for systems up to 200 equal spheres are listed in [20]. Then, the percentage error between the density ratio $\mu_\mathrm{p}$ and its optimal value given by $\mu_\mathrm{p}^*$ is computed as

$$\mu_\delta := \left| \frac{\mu_\mathrm{p} - \mu_\mathrm{p}^*}{\mu_\mathrm{p}^*} \right| 100\%. \tag{20}$$

Also the computation time to solve the SMPC, i.e., $\mu_{t_\mathrm{c}}$, and the objective cost value, i.e., $\mu_J$, are adopted. The average values of the performance metrics, denoted as $\tilde{\mu}_{\mathrm{a,p},\delta,t_\mathrm{c},J}$, are computed over 25 simulations of $100\,\mathrm{s}$, and with $T = 0.01\,\mathrm{s}$.

### *B. Results on the 20-UAVs scenario*

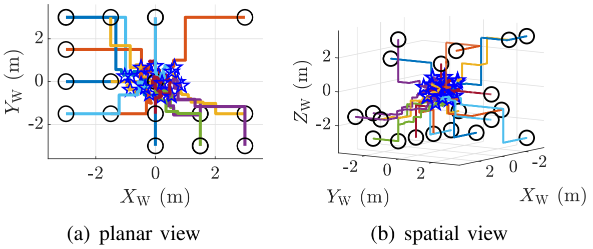
First, an illustrative example with $n = 20$ UAVs is shown. Fig. 3 illustrates the time evolution of the average performance metrics $\tilde{\mu}_\mathrm{a}$ in (18) and $\tilde{\mu}_\mathrm{p}$ in (19) (blue lines), respectively. Moreover, the evolutions of their maximum and minimum values (red lines), the maximum aggregation ratio $\mu_\mathrm{a}^* = 1$, the optimal packing density ratio $\mu_{p,20}^* = 0.477$ (green lines), and the median simulation (black line), indicated by the apex $^{(13)}$, are illustrated. Relying on the median simulation, Fig. 4 shows the trajectories of the quadrotors, where the initial and final positions are represented by black circles and blue stars, respectively. It is worth highlighting that the swarm aggregation occurs in the vicinity of the origin, while, as expected, the collision among the quadrotors is avoided.



(a) aggregation

(b) packing ratio

Fig. 3. Time evolution of the performance metrics when the distributed SMPC is used: $\mu_\mathrm{a}$ (a), and $\mu_\mathrm{p}$ (b).



(a) planar view

(b) spatial view

Fig. 4. Swarm trajectories ($n = 20$). Planar view (a), spatial view (b).

Finally, note that the proposed distributed SMPC has been tested on a laptop with 12th Gen Intel(R) Core(TM) i7-12700H processor, achieving an average computational time of $3.8 \times 10^{-4}\,\mathrm{s}$ to solve (16).

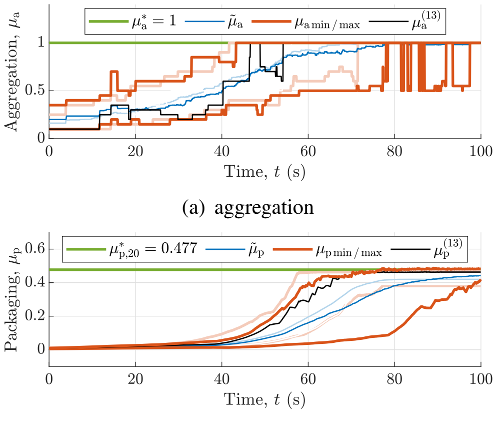
*Comparison and discussion:* In order to further assess the proposed distributed SMPC (briefly, dSMPC), the results are hereafter compared with those achieved by applying a centralized SMPC (briefly, cSMPC), and a distributed version of the CLF approach (briefly, dCLF) as baseline. The outcome of the simulations is reported in Table II.

TABLE II
AVERAGE PERFORMANCE METRICS.

| Strategy | $\mu_\mathrm{a}$ [1] | $\tilde{\mu}_\mathrm{p}$ | $\tilde{\mu}_\delta$ | $\tilde{\mu}_{t_\mathrm{c}}$ (s) | $\tilde{\mu}_J$ (m$^2$) [2] |
|---|---|---|---|---|---|
| dSMPC | 1 | 0.4197 | 0.12 | $3.8 \times 10^{-4}$ | 169.455 |
| cSMPC | 1 | 0.443 | 0.0712 | 1.0109 | 166.352 |
| dCLF | 1 | 0.4 | 0.1614 | $1.98 \times 10^{-4}$ | 174.775 |

[1] The metric $\tilde{\mu}_\mathrm{a}$ is recorded at $t = 100\,\mathrm{s}$.

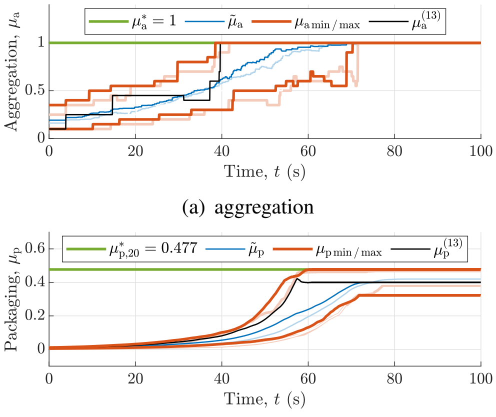
[2] As for the dCLF strategy, $\tilde{\mu}_J$ is computed over the last $N$ time instants.



(a) aggregation

(b) packing ratio

Fig. 5. Time evolution of the performance metrics when the centralized SMPC is used (partially transparent lines represent the metrics in the case of distributed SMPC): $\mu_\mathrm{a}$ (a), and $\mu_\mathrm{p}$ (b).



(a) aggregation

(b) packing ratio

Fig. 6. Time evolution of the performance metrics when the distributed CLF baseline is used (partially transparent lines represent the metrics in the case of distributed SMPC): $\mu_\mathrm{a}$ (a), and $\mu_\mathrm{p}$ (b).

Figs 5 and 6 show the time evolution of the average performance metrics $\tilde{\mu}_\mathrm{a}$ and $\tilde{\mu}_\mathrm{p}$ for the cSMPC and the dCLF strategies, respectively, where the transparent lines are the corresponding signals in the case of dSMPC. The results highlight that the three strategies are finally (at $t = 100\,\mathrm{s}$) capable of fully aggregating the system while fulfilling the collision avoidance objective. As expected, the cSMPC performs better in terms of packing with respect to dSMPC and dCLF. The dSMPC performance is however better than the one achieved with the dCLF approach. These results are also coherent with the values of the objective cost $\tilde{\mu}_J$. Nevertheless, a higher average computational time is evident (about 4 orders of magnitude) for the cSMPC approach, while the dSMPC takes only almost twice as long as the time taken by the dCLF, which is the fastest method since no predictions occur.

### C. Results on larger-scale UAVs scenarios

In order to assess the scalability of the proposal in more complex scenarios, different population sizes of the UAVs swarm, ranging from $n = 10$ to 150 UAVs, are now considered. Specifically, 25 simulations were performed for each value $n$, and the time taken for solving the distributed SMPC problem was recorded. Fig. 7 shows a box plot of these times, and it is worth highlighting that, despite the increasing number of UAVs, the solution to (16) is still feasible with an average computational time of $8.3633 \times 10^{-4}$ s. Clearly, as expected, the computational time, depending on the clustering mechanism, slightly rises with the higher number of UAVs (see Remark 4.2), but, even with the most demanding setting ($n = 150$), the average value for the proposal is below $T = 0.01\,\mathrm{s}$ (i.e., $1.2 \times 10^{-3}\,\mathrm{s}$).

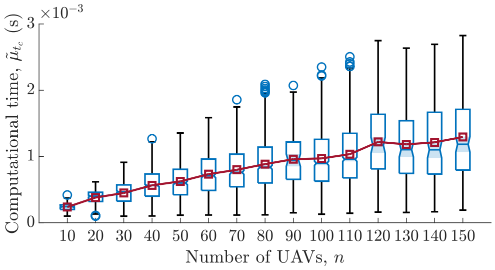


Fig. 7. Box plot of the computational times to solve (16). Each box represents values obtained from 25 simulations with different sizes of the swarm population.

## VI. Conclusions

This letter has proposed a distributed SMPC strategy to solve a rendezvous problem for a scalable UAVs swarm. The proposed approach is based on a switched formulation of the UAV model suitably quantizing the velocity inputs. Under the assumption that the UAVs location is available, a clustering method is first defined, and a sequential distributed MPC, with collision avoidance properties, is then solved. Future works will be devoted to the application on a real setup, and the extension to general trajectory optimization problems and more complex scenarios, e.g., in presence of disturbances, delays or fixed and moving obstacles. The use of alternative collision avoidance constraints (for instance reformulating them in linear terms, see e.g., [21]) is also of interest.

## References

[1] G. Udeanu, A. Dobrescu, and M. Oltean, "Unmanned aerial vehicle in military operations," *Scientific Research and Education in the Air Force*, vol. 18, pp. 199–206, 2016.

[2] A. Rejeb, A. Abdollahi, K. Rejeb, and H. Treiblmaier, "Drones in agriculture: A review and bibliometric analysis," *Computers and Electronics in Agriculture*, vol. 198, p. 107017, 2022.

[3] R. Spica, E. Cristofalo, Z. Wang, E. Montijano, and M. Schwager, "A real-time game theoretic planner for autonomous two-player drone racing," *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1389–1403, 2020.

[4] Z. Wang, X. Zhou, C. Xu, and F. Gao, "Geometrically constrained trajectory optimization for multicopters," *IEEE Transactions on Robotics*, vol. 38, no. 5, pp. 3259–3278, 2022.

[5] M. A. Toksöz, S. Oğuz, and V. Gazi, "Decentralized formation control of a swarm of quadrotor helicopters," in *IEEE 15th International Conference on Control and Automation*, Edinburgh, UK, 2019, pp. 1006–1013.

[6] J. Erskine, R. Balderas-Hill, I. Fantoni, and A. Chriette, "Model predictive control for dynamic quadrotor bearing formations," in *IEEE International Conference on Robotics and Automation*, Xi'an, China, 2021, pp. 124–130.

[7] R. Carli, G. Cavone, N. Epicoco, M. Di Ferdinando, P. Scarabaggio, and M. Dotoli, "Consensus-based algorithms for controlling swarms of unmanned aerial vehicles," in *Ad-Hoc, Mobile, and Wireless Networks*, L. A. Grieco, G. Boggia, G. Piro, Y. Jararweh, and C. Campolo, Eds. Cham: Springer International Publishing, 2020, pp. 84–99.

[8] J. Park, D. Kim, G. C. Kim, D. Oh, and H. J. Kim, "Online distributed trajectory planning for quadrotor swarm with feasibility guarantee using linear safe corridor," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4869–4876, 2022.

[9] A. Zaidi, M. Kazim, R. Weng, D. Wang, and X. Zhang, "Distributed observer-based leader following consensus tracking protocol for a swarm of drones," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 3, p. 64, 2021.

[10] T. Wakabayashi, Y. Suzuki, and S. Suzuki, "Dynamic obstacle avoidance for multi-rotor UAV using chance-constraints based on obstacle velocity," *Robotics and Autonomous Systems*, vol. 160, p. 104320, 2023.

[11] M. Gauci, J. Chen, W. Li, T. J. Dodd, and R. Groß, "Self-organized aggregation without computation," *The International Journal of Robotics Research*, vol. 33, no. 8, pp. 1145–1161, 2014.

[12] C. Yuca Huanca, G. P. Incremona, R. Groß, and P. Colaneri, "Design of a switched control Lyapunov function for mobile robots aggregation," in *19th International Conference on Informatics in Control, Automation and Robotics - ICINCO*, Lisbon, Portugal, 2022, pp. 492–499.

[13] T. Wood, P. M. Esfahani, and J. Lygeros, "Hybrid modelling and reachability on autonomous RC-cars," in *4th IFAC Conference on Analysis and Design of Hybrid Systems*, vol. 45, no. 9, Eindhoven, The Netherlands, 2012, pp. 430–435.

[14] H. Oleynikova, M. Burri, Z. Taylor, J. Nieto, R. Siegwart, and E. Galceran, "Continuous-time trajectory optimization for online UAV replanning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Daejeon, Korea (South), 2016, pp. 5332–5339.

[15] E. Sontag and H. Sussmann, "Nonsmooth control-Lyapunov functions," in *34th IEEE Conference on Decision and Control*, vol. 3, New Orleans, LA, USA, 1995, pp. 2799–2805.

[16] E. D. Sontag, "Control-Lyapunov functions," in *Open Problems in Mathematical Systems and Control Theory*, V. Blondel, E. D. Sontag, M. Vidyasagar, and J. C. Willems, Eds. London: Springer London, 1999, pp. 211–216.

[17] P. Colaneri, J. C. Geromel, and A. Astolfi, "Stabilization of continuous-time switched nonlinear systems," *Systems & Control Letters*, vol. 57, no. 1, pp. 95–103, 2008.

[18] P. Christofides, R. Scattolini, D. Peña, and J. Liu, "Distributed model predictive control: A tutorial review and future research directions," *Computers & Chemical Engineering*, vol. 51, pp. 21–41, 04 2013.

[19] J. Conway and N. Sloane, *Sphere Packings, Lattices and Groups*. Springer New York, NY, 1988.

[20] W. Q. Huang and L. Yu, "Serial symmetrical relocation algorithm for the equal sphere packing problem," 2012.

[21] M. Farina, A. Perizzato, and R. Scattolini, "Application of distributed predictive control to motion and coordination problems for unicycle autonomous robots," *Robotics and Autonomous Systems*, vol. 72, pp. 248–260, 2015.