Decoupled Initialization-Based Distributed Tube Model Predictive Control for Multi-Robot Systems Coordination

Abdullateef Ogundipe[1], Yiwen Chen[1], Auwal Shehu Tijjani[2], and Michael Defoort[1]

***Abstract*— This paper considers the problem of collision-free navigation for multiple nonholonomic (unicycle) mobile robots in the presence of bounded disturbances, subject to safety constraints ensuring avoidance of both obstacles and inter-robot collisions. We propose a two-stage hierarchical control architecture, namely Decoupled Initialization-Based Distributed Tube MPC (DI-DTMPC), for distributed nonholonomic multi-robot systems. In Stage 1, each robot independently solves a computationally cheap and decoupled nonlinear program in parallel to generate a warm-start trajectory toward its goal. Then, in the subsequent stage (i.e., Stage 2), the robots solve a full-constraint tube-based MPC problem in a distributed manner, incorporating both local and inter-robot coupled constraints, which is initialized and guided by the Stage 1 result. The proposed DI-DTMPC is validated through numerical simulations and hardware experiments, achieving reduction in travel time compared to single-stage DMPC. A quantitative comparative study with recent schemes from the literature demonstrates the efficiency of the proposed scheme in terms of cost performance.**

***Index Terms*— Collision-free navigation, Model predictive control (MPC), Multi-robot systems, Experimental validation.**

I. INTRODUCTION

Multi-robot system (MRS) coordination supports a broad class of real-world applications, including autonomous exploration, environmental monitoring [1], hazardous industrial inspection [2], and operations in remote or inaccessible areas. A particularly challenging and practically important instance is the collision-free navigation of multiple nonholonomic mobile robots governed by nonintegrable kinematic constraints that prevents lateral motion [3]. The nonholonomic constraint couples position and heading, complicating trajectory planning and constraint satisfaction compared to fully actuated platforms.

Among the control methods proposed for mobile robot coordination, model predictive control (MPC) has attracted sustained interest. Its ability to handle state and input constraints, together with its optimization-based formulation and predictive horizon, makes it well suited for proactive collision avoidance [4]. However, when multiple robots must be coordinated, the joint optimization becomes computationally intensive, motivating distributed and hierarchical formulations.

The theoretical foundations of centralized MPC, including stability, recursive feasibility, and constraint satisfaction for nonlinear systems, were established in earlier works of [4] and [5]. Although centralized MPC achieves globally optimal trajectories, its computational complexity grows rapidly with fleet size and it introduces a single point of failure, motivating distributed alternatives [6]. Sequential (non-cooperative) MPC solves the subproblems one robot at a time using predictions broadcast by already-solved neighbors, a notable example is the work of [7] for robust multi-vehicle guidance. While stable, sequential MPC incurs latency and cannot exploit parallelism.

On the other hand, distributed MPC (DMPC) for multi-robot coordination decomposes the global coordination problem into per-robot subproblems coupled through a communication graph. Moreover, distributed tube-based MPC schemes combine the robustness of tube MPC with distributed computation. Based on this concept, [8] develops a distributed tube-based MPC for uncertain nonlinear multi-agent systems, enforcing both local and inter-agent constraints via agent updates. A closely related work is [9], which introduces a parallelized robust DMPC for omnidirectional multi-robot systems by enforcing horizon-wide reference-tube consistency conditions around iteratively updated reference trajectories. While theoretically rigorous, [9] requires: (i) an iterative inner loop at every time step, and (ii) an offline reference-trajectory precomputation whose cost is quite high; all these factors limit real-time applicability of this scheme.

Another MPC structure is hierarchical MPC, which decomposes the controller into layers of decreasing time scale or constraint complexity [10]. The upper layer provides a simplified plan that guides and initializes the lower layer, which enforces the full constraint set. This architecture has been applied to multi-robot navigation [11], legged manipulators [12], etc. All of these works exploit the layer separation to reduce per-step computation and improve scalability, yet none has treated that of nonholonomic multi-robot tube MPC, to the best knowledge of the authors. Hence, the proposed DI-DTMPC in this paper fills this gap.

The contribution of this paper can be summarized as follows: A two-stage hierarchical MPC architecture, referred to as Decoupled Initialization-Based Distributed Tube MPC (DI-DTMPC) scheme, has been developed for collision-free navigation of nonholonomic mobile robots. Stage 1 solves

*The first and third authors acknowledges the financial support of Petroleum Technology Development Fund (PTDF). The second and fourth authors acknowledge financial support from the CNRS through the MITI interdisciplinary programs of the exploratory research program (project COSMOD5G) and by the ANR agency under TROCOOL project (ANR-25-CE10-4393).

[1]Abdullateef Ogundipe, Yiwen Chen, and Michael Defoort are with LAMIH UMR CNRS 8201, Universite Polytechnique Hauts-De-France, Valenciennes, France. abdullateef.ogundipe@uphf.fr, yiwen.chen@uphf.fr, and michael.defoort@uphf.fr

[2]Auwal Shehu Tijjani is with the Centre for Digital Systems, IMT Nord Europe, Institut Mines-Télécom, Univ. Lille, F-59000, Lille, France. auwal.shehu@imt-nord-europe.fr

decoupled NLPs in parallel (without inter-robot coupling constraints) to generate warm-start (initialization) trajectories. Then, Stage 2 solves the full-constraint robust tube MPC initialized and guided by Stage 1 outputs via a consistency constraint. This constitutes a novel application of hierarchical MPC to distributed nonholonomic tube MPC. The proposed scheme is validated not only in numerical simulations but also in hardware experiments.

The paper is structured as follows: Section II presents the problem formulation. Section III details the DI-DTMPC design. Section IV discusses simulation and experimental results. Section V concludes this paper and outlines future research directions.

Notation: $\mathbb{R}$ denotes the set of real numbers. The notation $x[\kappa|k]$ denotes a predicted trajectory at time step k. The term $x[k]$ denotes the actual robot state while κ indexes the prediction step. For $x \in \mathbb{R}^n$, the Euclidean norm is $\|x\| := \sqrt{x^\top x}$, and with $P \in \mathbb{R}^{n\times n}$ positive definite, the weighted norm is $\|x\|_P := \sqrt{x^\top P x}$. Let $A, B \subset \mathbb{R}^n$; their Minkowski sum and Pontryagin difference are defined as $A \oplus B := \{a + b \in \mathbb{R}^n \mid a \in A, b \in B\}$ and $A \ominus B := \{a \in \mathbb{R}^n \mid a + b \in A, \ \forall b \in B\}$. Lastly, $\mathbf{1}$ denotes a vector of all ones.

II. Problem Formulation

A. Nonholonomic Robot Dynamics

Consider a multi-agent system consisting of $N^\star$ nonholonomic mobile robots whose dynamics can be described by:

$$\begin{bmatrix}\dot{\chi}_i(t)\\ \dot{y}_i(t)\\ \dot{\theta}_i(t)\end{bmatrix} = \begin{bmatrix} v_i(t)\cos(\theta_i(t))\\ v_i(t)\sin(\theta_i(t))\\ \omega_i(t)\end{bmatrix}, i \in \mathcal{V} := \{1,\dots,N^\star\} \quad (1)$$

where $x_i(t) = [\chi_i(t), y_i(t), \theta_i(t)]^T \in \mathbb{R}^3$ denotes the robot state consisting of planar position and orientation, and $u_i(t) = [v_i(t), \omega_i(t)]^T \in \mathbb{R}^2$ is the control input comprising the linear and angular velocities.
For implementation purpose, it is assumed that the control inputs remain constant over each sampling interval $[kT, (k+1)T)$, where $T > 0$ denotes the sampling period. Under this assumption, the discrete-time kinematic model is written as:

$$x_i[k+1] = f(x_i[k], u_i[k], d_i[k]), \quad x_i[0] = x_{i,0} \quad (2)$$

where $f(\cdot)$ is computed from (1) either using an Euler or high-order Runge–Kutta (e.g., RK4) method. It is worth noting that $d_i[k]$ represents disturbance which is assumed to be constrained, i.e., $d_i \in \mathcal{D}_i \subset \mathbb{R}^3$ with $\mathcal{D}_i$ a bounded set.

B. Communication network structure

The coupled constraints among robots can be described by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ represent the vertex set and the edge set respectively. Each edge $(i, j) \in \mathcal{E}$ represents a bidirectional communication link between the i-th and the j-th robot. (i.e., $i \in \mathcal{N}_j$ and $j \in \mathcal{N}_i$, where $\mathcal{N}_i$ is the set of all neighbors of the i-th robot).

C. Control Objective

The control objective is to design a distributed controller for each robot $i \in \mathcal{V}$ that drives it to its desired configuration x_i^g with input $u_{x_i^g}$, while satisfying input, state, and dynamic constraints. Hence, each robot $i \in \mathcal{V}$ should respect non-coupled (i.e., local) and coupled (i.e., neighbor-dependent) state constraints.

1) Local constraints (Obstacle avoidance): The local constraints aim to avoid a set of known static obstacles $\mathcal{S}$ in the environment, which is defined as follows:
$$h_i(x_i) \le 0, \quad (3)$$
where $h_i(x_i) := [-(h_i^{\text{obs}}(x_i))^T, (-h_i^{\min}(x_i))^T, (h_i^{\max}(x_i))^T]^T$ and
$$h_i^{\text{obs}}(x_i) := \text{dist}(q_i, \mathcal{O}_{s\in\mathcal{S}}) - d_{\min} \ge 0, \quad (4)$$
where $q_i[k] = [\chi_i, y_i]^T \in \mathbb{R}^2$ represents the position of the i-th robot. $\mathcal{O}_{s\in\mathcal{S}} \subset \mathbb{R}^2$ denotes the obstacle region and $\text{dist}(q_i, \mathcal{O}_{s\in\mathcal{S}})$ denotes the Euclidean distance from the robot's planar position to the obstacle region, and $d_{\min} > 0$ is a predefined safety margin. Furthermore, $h_i^{\min}(x_i) := q_i - [\chi_i^{\min}, y_i^{\min}]^T$ and $h_i^{\max}(x_i) := q_i - [\chi_i^{\max}, y_i^{\max}]^T$, constrain the trajectories of robots in a bounded rectangular region $[\chi_i^{\min}, \chi_i^{\max}] \times [y_i^{\min}, y_i^{\max}]$.
2) Coupled constraints (Inter-agent collision avoidance): The coupled constraints aim to avoid collisions with neighboring robots to ensure security. The pairwise collision avoidance constraint between agent i and agent j can be described by
$$c_{i,r}(x_i, x_{j\in\mathcal{N}_i}) := g_{\min}^2 - \|q_i - q_{j\in\mathcal{N}_i}\|^2 \le 0 \quad (5)$$
where $g_{\min}$ is the minimum allowed distance between any two robots to avoid collisions.

Accordingly, the feasible state sets are formulated as $\mathcal{X}_i := \{x_i \in \mathbb{R}^3 \mid h_i(x_i) \le 0\}$ and $\mathcal{X}_{i,c}(x_{j\in\mathcal{N}_i}) := \{x_i \in \mathbb{R}^3 \mid c_{i,r}(x_i, x_{j\in\mathcal{N}_i}) \le 0\}$. Moreover, the control input $u_i(t)$ are supposed to be bounded and subject to input constraints $u_i \in \mathcal{U}_i \subseteq \mathbb{R}^2$, where $\mathcal{U}_i := [v_i^{\min}, v_i^{\max}] \times [\omega_i^{\min}, \omega_i^{\max}]$.

III. Proposed DI-DTMPC scheme

This section presents the design of the proposed DI-DTMPC scheme for multi-robot motion planning. The main concept is dividing the computational process as follows.

A. Stage 1: Warm-Start less-constrained DMPC

At this stage, each agent computes a trajectory to its goal only with uncoupled constraints. Hence, the local optimization problem is formulated as:

$$\min_{x_i^{\text{warm}}[\kappa|k],\, u_i^{\text{warm}}[\kappa|k]} \sum_{\kappa=k}^{k+N-1} \Big(\|x_i^{\text{warm}}[\kappa|k] - x_i^g\|_{Q_i}^2 + \left\|u_i^{\text{warm}}[\kappa|k] - u_{x_i^g}\right\|_{R_i}^2 \Big) \quad (6a)$$

$$\text{s.t.} \quad x_i^{\text{warm}}[\kappa+1|k] = f\big(x_i^{\text{warm}}[\kappa|k], u_i^{\text{warm}}[\kappa|k], 0\big) \quad (6b)$$

$$h_i(x_i^{\text{warm}}[\kappa|k]) \le -\delta_i \mathbf{1}_{|\mathcal{S}|+4}, \quad (6c)$$

$$u_i^{\text{warm}}[\kappa|k] \in \mathcal{U}_i \quad (6d)$$

where $x_i^g = f(x_i^g, u_{x_i^g}, 0)$ is a steady state of dynamics (2) without disturbance. Meanwhile, x_i^g is also the goal of the i-th agent. Q_i, R_i are weight matrices and $x_i^{\text{warm}}[0|0] = x_{i,0}$. The initial value $x_i^{\text{warm}}[k|k]$ will be defined later based on the full-constrained DTMPC of the previous time step. $\delta_i > 0$ is a positive margin.

Remark 3.1: It is worth noting that (6a) is computationally cheap. The generated warm-start trajectory, i.e., $\{x_i^{\text{warm}}[\kappa|k], u_i^{\text{warm}}[\kappa|k]\}$, will be used in the subsequent stage as reference trajectory. That is the reason why a more strict safety margin δ_i is proposed for the uncoupled constraint $h_i(\cdot)$ i.e satisfying the obstacles $\mathcal{S}$ and the χ,y boundaries constraints (6c).

At the initial stage, the robots are assumed to be sufficiently separated, and the DMPC dynamics, subject only to bounded control inputs, evolve in a continuous manner. This leads to the following assumption:

Assumption 1: For $i \in \mathcal{V}$, the initial warm-start trajectory and controller $(x_i^{\text{warm}}[\cdot|0], u_i^{\text{warm}}[\cdot|0])$ satisfy coupled constraint with safety margin $\epsilon_i \geq 0$:

$$c_{i,r}(x_i^{\text{warm}}[\kappa|0], x_j^{\text{warm}}[\kappa|0]) \leq -\epsilon_i \mathbf{1}_{|\mathcal{N}_i|} \tag{7}$$

for all $\kappa \in \{0, \ldots, N-1\}$, $j \in \mathcal{N}_i$.

B. Stage 2: Full-constrained DTMPC

In this stage, the full-constrained DTMPC problem is employed. It relies on nominal trajectories generated by the disturbance-free dynamics. The i-th agent predicts its nominal state $x_i[\kappa|k], \kappa = k, \cdots, k+N$ with input $u_i[\kappa|k], \kappa = k, \cdots, k+N-1$ in terms of the following nominal dynamics without disturbances

$$x_i[\kappa+1|k] = f(x_i[\kappa|k], u_i[\kappa|k], 0). \tag{8}$$

In the DTMPC scheme, only $u_i[k|k]$ will be used to generate the inputs at step k. An auxiliary controller $K_i(x_i[k], x_i[k|k])$ is introduced to ensure that the deviation of the actual state from the predicted nominal state remains within an invariant set. We define $u_i[k]$ as

$$u_i[k] \coloneqq u_i[k|k] + K_i(x_i[k], x_i[k|k]), \tag{9}$$

Let $h_i[k]$ denote the deviation between the actual and nominal state, i.e.,

$$h_i[k+1] \coloneqq f(x_i[k], u_i[k], d_i[k]) - f(x_i[k|k], u_i[k|k], 0), \tag{10}$$

with $h_i[0] = 0$. The following robust positively invariant (RPI) set $\mathcal{H}_i$ could be proposed:

Assumption 2 (RPI): [9] There exists a neighborhood of the origin $\mathcal{H}_i \in \mathbb{R}^3$ such that $h_i[k+1] \in \mathcal{H}_i$ if $h_i[k] \in \mathcal{H}_i$, $u_i[k|k] \in \mathcal{U}_i, d_i[k] \in \mathcal{D}_i$.

According to Assumption 2, it follows that

$$x_i[k] \in x_i[k|k] \oplus \mathcal{H}_i. \tag{11}$$

In practice, $\mathcal{H}_i$ is computed by linearizing the unicycle dynamics about the equilibrium, making the discrete error map Schur stable within the minimal RPI set by a box $\mathcal{H}_i$ with given half-widths. The admissible set for auxiliary term $K_i(x_i[\kappa], x_i[\kappa|k])$ is defined as

$$\Delta\mathcal{U}_i \coloneqq \bigcup_{k\geq 0} \Big\{ K_i(x_i[k], x_i[k|k]) \in \mathcal{U}_i \Big| x_i[k], x_i[k|k] \in \mathcal{X}_i, x_i[k] - x_i[k|k] \in \mathcal{H}_i \Big\}. \tag{12}$$

Consequently, we can define the following set for nominal state and controller: $\hat{\mathcal{X}}_i \coloneqq \mathcal{X}_i \ominus \mathcal{H}_i$, $\hat{\mathcal{U}}_i \coloneqq \mathcal{U}_i \ominus \Delta\mathcal{U}_i$ and $\hat{\mathcal{X}}_{i,c} \coloneqq \bigcup_{k\geq 0} \Big\{ x_i[k|k] \in \mathbb{R}^3 \Big| c_{i,\alpha}(x_i[k|k], x_{j\in\mathcal{N}_i}[k|k]) \leq 0,\ \forall x_i[k] \in x_i[k|k] \oplus \mathcal{H}_i,\ \forall x_j[k] \in x_j[k|k] \oplus \mathcal{H}_j,\ j \in \mathcal{N}_i \Big\}$.

The control objective is to robustly and asymptotically stabilize the desired states x_i^g, $i \in \mathcal{V}$, where

$$x_i^g \coloneqq \left\{ x_i^g \;\middle|\; \begin{array}{l} x_i^g = f_i(x_i^g, u_{x_i^g}, 0), \\ x_i^g \in \hat{\mathcal{X}}_i, \quad x_i^g \in \hat{\mathcal{X}}_{i,c}(x^g_{j\in\mathcal{N}_i}) \\ u_{x_i^g} \in \hat{\mathcal{U}}_i, \end{array} \right\} \tag{13}$$

To ensure the nominal state $x[\kappa|k]$ stays in a neighborhood of the warm trajectory in the first stage, one introduce the following consistency constraint:

$$x_i[\kappa|k] \in x_i^{\text{warm}}[\kappa|k] \oplus \hat{\mathcal{C}}_i, \tag{14}$$

where $\hat{\mathcal{C}}_i$ is some compact neighborhood of the origin, named as consistency constraint set. Note that both $x_i^{\text{warm}}[\kappa|k]$ and $\hat{\mathcal{C}}_i$ are known to neighbors. Besides, according to (11), the initial nominal state satisfies

$$x_i[k|k] \in x_i[k] \oplus (-\mathcal{H}_i), \tag{15}$$

where $-\mathcal{H}_i = -\mathbf{1}\mathcal{H}_i$. Define $\mathcal{C}_i \coloneqq \hat{\mathcal{C}}_i \oplus \mathcal{H}_i$. (14) and (15) imply

$$x_i[k] \in x_i^{\text{warm}}[k|k] \oplus \mathcal{C}_i, \tag{16}$$

Based on the above discussion, the local optimality criterion (17) can be defined by:

$$J_i^*(x_i[k]) = \min_{x_i[k|k], u_i[\cdot|k]} \quad J_i(x_i[\cdot|k], u_i[\cdot|k]) \tag{17}$$

where

$$J_i(\cdot) = \sum_{\kappa=k}^{k+N-1} \ell_i(x_i[\kappa|k], u_i[\kappa|k]) + J_i^f(x_i[k+N|k]).$$

with a positive-definite terminal cost-function $J_i^f(\cdot)$ and

$$\ell_i(\cdot) = (\|x_i[\kappa|k] - x_i^g\|^2_{Q_i} + \left\|u_i[\kappa|k] - u_{x_i^g}\right\|^2_{R_i})$$

Besides, Q_i, R_i are positive semi-definite weighting matrices. The local optimality criterion (17) is subject to

$$x_i[k|k] \in x_i[k] \oplus (-\mathcal{H}_i) \tag{18a}$$

$$x_i[\kappa+1|k] = f_i(x_i[\kappa|k], u_i[\kappa|k], 0) \tag{18b}$$

$$u_i[\kappa|k] \in \hat{\mathcal{U}}_i, \quad \forall\kappa \in \{k, \ldots, k+N-1\} \tag{18c}$$

$$x_i[\kappa|k] \in x_i^{\text{warm}}[\kappa|k] \oplus \hat{\mathcal{C}}_i, \quad \forall\kappa \in \{k, \ldots, k+N-1\} \tag{18d}$$

$$x_i[k+N|k] \in \mathcal{X}_i^f. \tag{18e}$$

The terminal set $\mathcal{X}_i^f$ will be specified later in Assumption 3. The obtained optimal nominal states and control inputs are denoted by $x_i^*[\cdot|k],\ u_i^*[\cdot|k]$. The MPC control law for the i-th agent at time $k \geq 0$ is

$$u_i^{\mathrm{MPC}}(x_i[k]) := u_i^*[k|k] + K_i\big(x_i[k],\, x_i^*[k|k]\big), \tag{19}$$

where $u_i^*[k|k]$ is the first element of the optimal input sequence, and $K_i(\cdot,\cdot)$ denotes an auxiliary controller ensuring Assumption 2. To ensure the recursive feasibility and closed-loop stability, the following basic assumption regarding the terminal set is proposed:

Assumption 3: Suppose that for terminal set $\mathcal{X}_i^f$ and the terminal cost function $J_i^f(\cdot)$, there exist some state feedback controllers $\kappa_i^f(\cdot)$,$i \in \mathcal{V}$ such that:

1) for all $i \in \mathcal{V}$, $\mathcal{X}_i^f \subseteq \mathcal{X}_i \ominus \mathcal{C}_i$ and for all $x_i \in \mathcal{X}_i^f \oplus \mathcal{C}_i, x_j \in \mathcal{X}_j^f \oplus \mathcal{C}_j, j \in \mathcal{N}_i$, the coupled constraints $c_{i,r}(x_i, x_j) \leq 0$ hold;
2) $k_i^f(x_i) \in \hat{\mathcal{U}}_i$, for all $x_i \in \mathcal{X}_i^f$;
3) $f(x_i, \kappa_i^f(x_i), 0) \in \mathcal{X}_i^f$, for all $x_i \in \mathcal{X}_i^f$;
4) $J_i^f(f(x_i, \kappa_i^f(x_i), 0)) - J_i^f(x_i) \leq -\ell_i(x_i, \kappa_i^f(x_i))$, for all $x_i \in \mathcal{X}_i^f$.

The initialization of the second stage DTMPC is ensured by the following assumption on the initialization of warm-start with respect to the terminal constraints :

Assumption 4 (Terminal Constraints with warm-start): For each subsystem $i \in \mathcal{V}$, the initial warm- start trajectories satisfies

$$x_i^{\mathrm{warm}}[N|0] \in \mathcal{X}_i^f. \tag{20}$$

The warm-start trajectories $x_i^{\mathrm{warm}}[\kappa|0], \kappa \in \{0, \ldots, N\}$ and corresponding input $u_i^{\mathrm{warm}}[\kappa|0], \kappa \in \{0, \ldots, N-1\}$, are said to be initially feasible if (20) and (21a)-(21e) hold:

$$x_i^{\mathrm{warm}}[0|0] = x_{i,0}, \tag{21a}$$
$$x_i^{\mathrm{warm}}[\kappa+1|0] = f_i\big(x_i^{\mathrm{warm}}[\kappa|0],\, u_i^{\mathrm{warm}}[\kappa|0], 0\big), \tag{21b}$$
$$h_i(x_i) \leq 0, \qquad \forall x_i \in x_i^{\mathrm{warm}}[\kappa|0] \oplus \mathcal{C}_i, \tag{21c}$$
$$c_{i,r}(x_i, x_{j\in\mathcal{N}_i}) \leq 0, \forall x_i \in x_i^{\mathrm{warm}}[\kappa|0] \oplus \mathcal{C}_i, \quad \forall x_j \in x_j^{\mathrm{warm}}[\kappa|0] \oplus \mathcal{C}_j, j \in \mathcal{N}_i \tag{21d}$$
$$u_i^{\mathrm{warm}}[\kappa|0] \in \hat{\mathcal{U}}_i. \tag{21e}$$

Observe that (21a) and (21b) naturally hold in terms of $x_i^{\mathrm{warm}}[0|0] = x_{i,0}$ and (6b). According to the continuity of the nonlinear dynamics $f(x_i, u_i, 0)$ and the boundedness of control input, we can naturally verify the continuity of the warm-start trajectories $x_i^{\mathrm{warm}}[\kappa|k], i \in \mathcal{V}$. By selecting suitable safety margins δ_i in (6c) and ϵ_i in Assumption 1, (21c) and (21d) hold in terms of the continuity of uncoupled constraint $h_i(\cdot)$ and coupled constraint $c_{i,r}(\cdots)$. Moreover, since the warm-start trajectories and controllers are also updated according to the disturbance-free dynamics in (6b), one can similarly impose an RPI assumption on them. Therefore, based on (6d), it is reasonable to assume (21e) for the initialization of the warm-start control input. At each time step k, the initial value of the warm-start DMPC is defined as

$$x_i^{\mathrm{warm}}[k|k] = x_i^*[k|k-1] \tag{22}$$

to preserve continuity between successive optimizations. This intuitively implies

$$x_i^*[k|k-1] \in x_i^{\mathrm{warm}}[k|k] \oplus \hat{\mathcal{C}}_i. \tag{23}$$

In terms of the continuity of dynamics $f_i(\cdot,\cdot,0)$ and bounded inputs $u_i^*[\kappa|k-1]$, $u_i^{\mathrm{warm}}[\kappa|k]$, the deviation $f_i(x_i^*[\kappa|k-1], u_i^*[\kappa|k-1], 0) - f_i(x_i^{\mathrm{warm}}[\kappa|k], u_i^{\mathrm{warm}}[\kappa|k], 0)$ is bounded, for $\kappa = k, \cdots, k+N-2$. Hence, we can propose the following assumption for the warm-start trajectories:

Assumption 5: For $i \in \mathcal{V}$, the initial warm- start trajectories satisfies

$$x_i^*[\kappa|k-1] \in x_i^{\mathrm{warm}}[\kappa|k] \oplus \hat{\mathcal{C}}_i, \kappa = k, \cdots, k+N-1. \tag{24}$$

Remark 3.2: DI-DTMPC enforces consistency only at the initial prediction index $\kappa = k$ through an anchor condition ($x_i^{\mathrm{warm}}[k|k] = x_i^*[k|k-1]$); horizon-wide coherence follows automatically from dynamics propagation under bounded inputs. In contrast, for [9], the reference-trajectory approach imposes a cross-time, horizon-wise consistency condition that the previously optimal prediction $x_i^*[\kappa|k-1]$ must lie in the tube around the selected reference $x_i^{\mathrm{ref}}[\kappa|k]$ for all prediction indices κ.

Theorem 1: Suppose Assumptions 1-5 hold. For i-th agent, $i \in \mathcal{V}$, the related optimization problem (17) subject to (18) are recursively feasible with respect to the closed-loop dynamics (2) for any $d_i[k] \in \mathcal{D}_i$. Moreover, the closed-loop dynamics (2) are robustly asymptotically stable with respect to x_i^g, $i \in \mathcal{V}$.

Based on Theorem 1, (18d) holds in the optimization for all k, which indicates

$$x_i^*[k|k-1] \in x_i^{\mathrm{warm}}[k|k-1] \oplus \hat{\mathcal{C}}_i. \tag{25}$$

It then follows with (22) that

$$\begin{aligned} x_i^{\mathrm{warm}}[k|k] &\in x_i^{\mathrm{warm}}[k|k-1] \oplus \hat{\mathcal{C}}_i \\ &\subseteq x_i^{\mathrm{warm}}[k|k-1] \oplus \mathcal{C}_i. \end{aligned} \tag{26}$$

Observe that (21d) holds for the initialization. If we suppose (21d) exists for step $k-1$ for all $i \in \mathcal{V}$ (replacing $x_i^{\mathrm{warm}}[\kappa|0], \kappa = 0, \cdots, N$ with $x_i^{\mathrm{warm}}[\kappa|k-1], \kappa = k-1, \cdots, k+N-1$), then at step k,

$$c_{i,r}(x_i^{\mathrm{warm}}[k|k], x_{j\in\mathcal{N}_i}^{\mathrm{warm}}[k|k]) \leq 0. \tag{27}$$

in terms of (26). One can extend (27) to $x_i^{\mathrm{warm}}[\kappa|k]$ and its neighbors $x_{j\in\mathcal{N}_i}^{\mathrm{warm}}[\kappa|k]$ for $\kappa = k, \cdots, k+N$. Together with (6c), we can propose the following assumption:

Assumption 6: For $i \in \mathcal{V}$, the initial warm- start trajectories satisfies for $\kappa = k, \cdots, k+N$,

$$h_i(x_i) \leq 0, \qquad \forall x_i \in x_i^{\mathrm{warm}}[\kappa|k] \oplus \mathcal{C}_i \tag{28}$$
$$c_{i,r}(x_i, x_{j\in\mathcal{N}_i}) \leq 0, \quad \forall x_i \in x_i^{\mathrm{warm}}[\kappa|k] \oplus \mathcal{C}_i, \quad \forall x_j \in x_j^{\mathrm{warm}}[\kappa|k] \oplus \mathcal{C}_j, j \in \mathcal{N}_i \tag{29}$$

With assumption 6, one can verify that the real state $x_i[k], k \geq 0$ and control input $u_i^{\mathrm{MPC}}(x_i[k]), k \geq 0$ satisfy the state constraints and input constraint $u_i \in \mathcal{U}_i$.

Proposition 1: Suppose Assumptions 1-6 hold, let there be a set of initially feasible warm start trajectories $x_i^{\text{warm}}[\kappa|k]$, then for all $i \in \mathcal{V}$, the real state $x_i[k], k \geq 0$ and control input $u_i^{\text{MPC}}(x_i[k]), k \geq 0$ satisfy the state constraints (3)-(5) and input constraint $u_i \in \mathcal{U}_i$.

IV. Results and Discussion

This section describes and presents the results obtained from the proposed DI-DTMPC scheme through both numerical simulations and hardware experiments.

A. Simulations

1) Simulation setup: In the numerical simulation, a fleet of four (4) mobile robots navigating in a 2D environment. The main reason for selecting this fleet size is to be within the range of our physical fleet of mobile robots. The states of the simulated mobile robots are initialized as follows: $x_1[0] = [0.20, 0.20, \frac{\pi}{4}]^\top$ m, $x_2[0] = [1.80, 0.20, \frac{3\pi}{4}]^\top$ m, $x_3[0] = [0.20, 1.80, -\frac{\pi}{4}]^\top$ m, $x_4[0] = [1.80, 1.40, -\frac{3\pi}{4}]^\top$ m, with their corresponding final goals designed as: $x_1^g = [1.80, 1.80, \frac{\pi}{4}]^\top$ m, $x_2^g = [0.40, 1.80, \frac{3\pi}{4}]^\top$ m, $x_3^g = [1.60, 0.20, -\frac{\pi}{4}]^\top$ m, $x_4^g = [0.40, 0.55, -\frac{3\pi}{4}]^\top$ m. To challenge the performance of the proposed scheme, three static obstacles are introduced as circular regions with radii and centers given by: $((0.90, 0.90), 0.10), ((0.45, 1.35), 0.08), ((1.50, 0.50), 0.08)\}$ m. Numerical values of the parameters are as follows. The sampling period is $h = 0.20$ s, with a prediction horizon of $N = 30$ and total simulation time $T_{\text{sim}} = 10.0$ s for $N_r = 4$ robots operating in the workspace $\mathcal{X} = [-0.5, 2.5]^2$ m. The input constraints are defined as $v \in [0, 0.5]$ and $\omega \in [-2\pi, 2\pi]$. The stage cost weights are $Q = \text{diag}(1, 5, 0.5)$ and $R = \text{diag}(0.5, 0.05)$, with disturbance bound $\bar{w} = [10^{-3}, 10^{-3}, 5 \times 10^{-4}]^\top$. The minimum inter-robot safety distance is $g_{\min} = 0.25$ m, and the robot–obstacle safety margin is $d_{\text{safe}} = 0.05$ m. The Stage 1 margin parameter is $\delta_i = 0.06$ m, and the consistency set half-widths are $\hat{C} = [0.35, 0.35, 0.6]^\top$. The cost weights and consistency-set widths follow standard MPC tuning practice in which position weights in Q dominates, R is small relative to Q, and $\hat{C}$ is designed large enough to allow Stage 2 to deviate from the Stage 1 reference without losing its guidance role. The communication radius is $r_{\text{comm}} = 1.0$ m. Finally, the terminal set parameters are defined by a shrink rate $\rho_\gamma = 0.97$ and a minimum size $\gamma_{\min} = 10^{-8}$. The final control inputs are computed in MATLAB R2025b. The NLP solver used throughout is IPOPT via the CasADi framework on a Dell laptop (Intel Core Ultra i7 processor, 16 GB RAM).

2) Comparative study: (i) *Case 1 (comparison with the single-stage DMPC [3]).* The implementation of the proposed DI-DTMPC scheme has been demonstrated through the autonomous navigation of the robots from the previously set initial positions to their final goals, while satisfying all imposed constraints as described in subsection IV-A. The generation of the warm-start trajectories, as the first stage of the proposed DI-DTMPC scheme, is depicted in

TABLE I: Travel time comparison of the proposed DI-DTMPC against single-stage DMPC [3].

Robot	Single-stage DMPC [3] (s)	Proposed DI-DTMPC (s)
Robot 1	11.00	6.40 (41.82 % ↓)
Robot 2	8.60	7.60 (11.63 % ↓)
Robot 3	5.80	5.40 (6.90 % ↓)
Robot 4	9.00	5.40 (40.00 % ↓)
Mean	8.60	6.20 (27.91 % ↓)

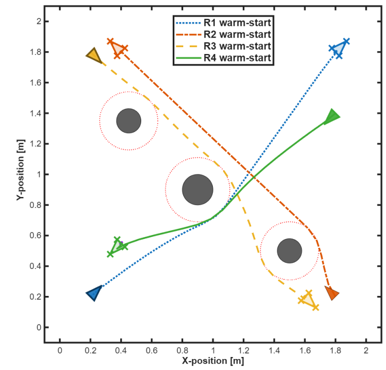

Fig. 1: Stage 1 (proposed DI-DTMPC) warm-start trajectories for four unicycle robots.

Fig. 1. Similarly, the final trajectories of the robots, after executing the second stage of the proposed DI-DTMPC scheme, are shown in Fig. 2. From the same figure and Fig. 3, we can observe that the robots navigate through the environment with respect to the constraints detailed in (3)-(5). For evaluation purposes, the proposed DI-DTMPC is compared with a single-stage DMPC in [3]). Note that in a conventional single-stage DMPC scheme, each agent solves a single optimization problem that simultaneously accounts for both local and coupling constraints over the entire prediction horizon. The average travel time per robot is used as a metric to evaluate the performance of the schemes, as shown in Table I. From Table I, the results show that DI-DTMPC reduces the average travel time by around 27.91 % compared with the conventional single-stage DMPC scheme in [3]. The travel-time reduction is attributed to the quality of initialization rather than to a faster solver time.

(ii) *Case 2 (comparison with a predefined trajectory-based scheme [9]).* To further challenge the proposed DI-DTMPC, we conduct a rigorous comparison with a recently proposed predefined trajectory-based scheme in [9]. Both schemes are implemented on three-wheel omnidirectional robot dynamics using identical kinematics and parameters, and evaluated under two different scenarios (i.e., connectivity and collision).

For the connectivity scenario, three (3) omnidirectional robots must maintain $\|q_i - q_j\| \leq \xi_{11}$ for all neighbors, with $\xi_{11} \in \{1.5, 2.0\}$ m, while in the collision scenario, four (4) robots start from antipodal positions and must cross to the opposite side without collision. Tables II and III summarize the per-robot total cost J_i and mean solve time per input step

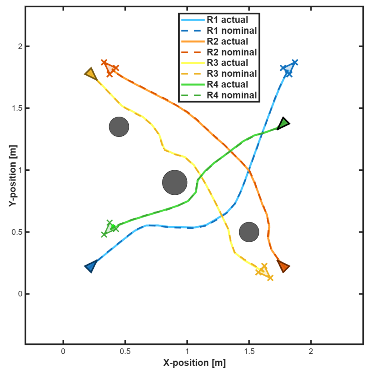

Fig. 2: Stage 2 (proposed DI-DTMPC) closed-loop trajectories.

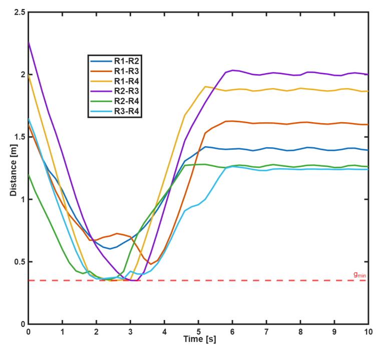

Fig. 3: Inter-robot distances during the simulation.

for both techniques.

On the connectivity scenario, the proposed DI-DTMPC shows elevated cost for Robot 1 while Robots 2 and 3 achieve lower cost than PDMPC [9]. On the collision scenario, DI-DTMPC achieves substantially lower cost on every robot at every iteration count. However, the proposed scheme requires more computation time for two NLPs per step but each NLP is easier to solve due to fewer active constraints and better initial guess. Thus, the offline pre-computation is reduced which resulted in low overall system-level computational budget of the proposed scheme.

B. Experiments

The validation of the proposed DI-DTMPC scheme for real-world applications has been shown through hardware experiments using a fleet of three mobile robots. Similar to the previous numerical simulations, in these real-time experiments, we set the initial states of the robots as: $x_1[0] = [0.0, 1.0, 0.0]^\top$ m, $x_2[0] = [1.0, 0.0, 0.0]^\top$ m, and $x_3[0] = [3.5, 0.0, 0.0]^\top$ m and their corresponding goal states as: $x_1^g = [3.0, 1.0, 0.0]^\top$ m, $x_2^g = [2.5, 2.0, 0.0]^\top$ m, and $x_3^g = [2.0, 2.5, 0.0]^\top$ m. It is worth noting that the tuning parameters remain the same as those presented in the simulation validation. To summarize this section, a complete video of the experimental tests and simulations can be found in https://shorturl.at/lNhmW.

TABLE II: Connectivity scenario (3 robots).

Method / ξ_{11}	Robot 1	Robot 2	Robot 3	Time [s]
PDMPC [9] (0 inner iterations)				
$\xi_{11} = 1.5$	104	252	187	0.563
$\xi_{11} = 2.0$	156	256	190	0.568
Proposed DI-DTMPC				
$\xi_{11} = 1.5$	264	101	157	0.999
$\xi_{11} = 2.0$	397	101	160	1.065
Offline precomputation				
PDMPC [9]	338–345 s per scenario			–
Proposed DI-DTMPC	<10 s			–

TABLE III: Collision scenario (4 robots).

Method / Iterations	R1	R2	R3	R4
PDMPC [9] (0 Iter.)	562	379	342	289
PDMPC [9] (1 Iter.)	518	369	332	280
Proposed DI-DTMPC	**398**	**177**	**123**	**117**
Offline precomp.	PDMPC [9]			1454 - 2475 s
	Proposed DI-DTMPC			< **10** s

V. Conclusion

This paper proposes a scheme named, Decoupled Initialization-Based Distributed Tube MPC (DI-DTMPC), a novel hierarchical two-stage MPC architecture for collision-free navigation of nonholonomic multi-robot systems under bounded disturbances. Future directions will focus on extension to dynamic obstacle environments and analysis under dynamic communication topologies.

References

[1] R. Ghasemi, "Cooperative intelligent robust adaptive methodology for uncertain non-affine nonlinear multi-agent systems: Application to multi-robot formation flight," *Eng. Appl. Artif. Intell.*, vol. 158, p. 111433, 2025.

[2] M. Labbadi, M. Defoort, A. S. Tijjani, T. Berger, and Y. Sallez, "Decentralized receding horizon motion planner for multi-robot with risk management," in *Proc. Int. Workshop Service Orientation Holonic Multi-Agent Manufacturing*, 2023, pp. 371–381.

[3] K. Worthmann, M. W. Mehrez, M. Zanon, G. K. I. Mann, R. G. Gosine, and M. Diehl, "Model predictive control of nonholonomic mobile robots without stabilizing constraints and costs," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1394–1406, 2015.

[4] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[5] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI: Nob Hill Publishing, 2009.

[6] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[7] A. Richards and J. P. How, "Robust distributed model predictive control," *Int. J. Control*, vol. 80, no. 9, pp. 1517–1531, 2007.

[8] A. Nikou and D. V. Dimarogonas, "Decentralized tube-based model predictive control of uncertain nonlinear multi-agent systems," *Int. J. Robust Nonlinear Control*, vol. 29, no. 10, pp. 3024–3047, 2019.

[9] A. Wiltz, F. Chen, and D. V. Dimarogonas, "Parallelized robust distributed model predictive control in the presence of coupled state constraints," *Automatica*, vol. 171, p. 111952, 2025.

[10] R. Scattolini and P. Colaneri, "Hierarchical model predictive control," in *Proc. IEEE Conf. Decision and Control*, 2007, pp. 4803–4808.

[11] D. Benders, J. Köhler, T. Niesten, R. Babuška, J. Alonso-Mora, and L. Ferranti, "Embedded hierarchical MPC for autonomous navigation," IEEE Trans. Robot., vol. 41, pp. 3556–3573, 2025.

[12] D. Wang, J. Yu, S. Wu, Z. Li, C. Li, R. Xiong, S. Qu, and Y. Wang, "A hierarchical MPC for end-effector tracking control of legged mobile manipulators," *IEEE Trans. Autom. Sci. Eng.*, vol. 22, pp. 4855–4867, 2025.