# Multi-Robot Control by Utilizing Distributed Model Predictive Controllers

KTH Royal Institute of Technology
School of Electrical Engineering
Department of Automatic Control

Student: Alexandros Filotheou (alefil@kth.se)
Supervisor: Alexandros Nikou (anikou@kth.se)
Examiner: Dimos Dimarogonas (dimos@kth.se)

February 1, 2017

**Abstract**

This paper addresses the problem of position- and orientation-based formation control of a class of second-order nonlinear multi-agent systems in 3D space, under static and undirected communication topologies. More specifically, we design a decentralized control protocol for each agent in the sense that each agent uses only local information from its neighbors to calculate its own control signal. Additionally, by introducing certain inter-agent distance constraints, we guarantee collision avoidance both between among the agents and between the agents and possible obstacles of the workspace. Connectivity maintenance between agents that are initially connected is also achieved by the proposed controller scheme. Finally, simulation results verify the performance of the proposed controllers.

# Contents

# 1 Introduction

formation of multi-agent systems, mpc intro etc.

motivation why we need mpc controllers...

In many control problems it is desired to design a stabilizing feedback such that a performance criterion is minimized while satisfying constraints on the controls and the states. Ideally one would look for a closed solution for the feedback law satisfying the constraints while optimizing the performance. However, typically the optimal feedback law cannot be found analytically, even in the unconstrained case, since it involves the solution of the corresponding Hamilton-Jacobi-Bellman partial differential equations. One approach to circumvent this problem is the repeated solution of an open-loop optimal control problem for a given state. The first part of the resulting open-loop input signal is implemented and the whole process is repeated. Control approaches using this strategy are referred to as Model Predictive Control (MPC).

# 2 Notation and Preliminaries

## 2.1 Notation

The set of positive integers is denoted by$\mathbb{N}$. The real $n$-coordinate space, with $n \in \mathbb{N}$, is denoted by $\mathbb{R}^n$; $\mathbb{R}^n_{\geq 0}$ and $\mathbb{R}^n_{>0}$ are the sets of real $n$-vectors with all elements nonnegative and positive, respectively. Given a set $S$, we denote as $|S|$ its cardinality. The notation $\|\mathbf{x}\|$ is used for the Euclidean norm of a vector $\mathbf{x} \in \mathbb{R}^n$. Given a symmetric matrix $\boldsymbol{A} = \boldsymbol{A}^T, \lambda_{\min}(\boldsymbol{A}) = \min\{|\lambda| : \lambda \in \sigma(\boldsymbol{A})\}$ denotes the minimum eigenvalue of $\boldsymbol{A}$, respectively, where $\sigma(\boldsymbol{A})$ is the set of all the eigenvalues of $\boldsymbol{A}$ and $\text{rank}(\boldsymbol{A})$ is its rank; $\boldsymbol{A} \otimes \boldsymbol{B}$ denotes the Kronecker product of matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m \times n}$, as was introduced in [?]. Define by $1_n \in \mathbb{R}^n, I_n \in \mathbb{R}^{n \times n}, 0_{m \times n} \in \mathbb{R}^{m \times n}$ the column vector with all entries 1, the unit matrix and the $m \times n$ matrix with all entries zeros, respectively. A matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ is called skew-symmetric if and only if $\boldsymbol{A}^\top = -\boldsymbol{A}$. $\mathcal{B}(\mathbf{c}, r) = \{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x} - \mathbf{c}\| \leq r\}$ is the 3D sphere of radius $r \in \mathbb{R}_{\geq 0}$ and center $\mathbf{c} \in \mathbb{R}^3$.

The vector expressing the coordinates of the origin of frame $\{j\}$ in frame $\{i\}$ is denoted by $\mathbf{p}_{j \triangleright i}$. When this vector is expressed in 3D space in a third frame, frame $\{k\}$, it is denoted by $\mathbf{p}^k_{j \triangleright i}$. The angular velocity of frame $\{j\}$ with respect to frame $\{i\}$, expressed in frame $\{k\}$ coordinates, is denoted by $\boldsymbol{\omega}^k_{j \triangleright i} \in \mathbb{R}^3$. We also use the notation $\mathbb{M} = \mathbb{R}^3 \times \mathbb{T}^3$. We further denote as $\mathbf{q}_{j \triangleright i} \in \mathbb{T}^3$ the Euler angles representing the orientation of frame $\{j\}$ with respect to frame $\{i\}$, where $\mathbb{T}^3$ is the 3D torus. For notational brevity, when a coordinate frame corresponds to the inertial frame of reference $\{\mathcal{O}\}$, we will omit its explicit notation (e.g., $\mathbf{p}_i = \mathbf{p}_{i \triangleright \mathcal{O}} = \mathbf{p}^{\mathcal{O}}_{i \triangleright \mathcal{O}}$, $\boldsymbol{\omega}_i = \boldsymbol{\omega}_{i \triangleright \mathcal{O}} = \boldsymbol{\omega}^{\mathcal{O}}_{i \triangleright \mathcal{O}}$). All vector and matrix differentiations are derived with respect to the inertial frame $\{\mathcal{O}\}$ unless stated otherwise.

## 2.2 Graph Theory

?? ISTORISOU // EXPAND An *undirected graph* $\mathcal{G}$ is a pair $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a finite set of nodes, representing a team of agents, and $\mathcal{E} \subseteq \big\{\{i, j\} : i, j \in \mathcal{V}, i \neq j\big\}$, with $M = |\mathcal{E}|$, is the set of edges that model the communication capability between neighboring agents. For each agent, its neighbors' set $\mathcal{N}_i$ is defined as $\mathcal{N}_i = \{i_1, \dots, i_{N_i}\} = \big\{j \in \mathcal{V} : \{i, j\} \in \mathcal{E}\big\}$, where $i_1, \dots, i_{N_i}$ is an enumeration of the neighbors of agent $i$ and $N_i = |\mathcal{N}_i|$. Moreover, the notation $\bar{\mathbf{x}}_i = (\mathbf{x}_{i_1}, \dots, \mathbf{x}_{i_{N_i}})$ is used to denote the vector of the neighbors of agent i, where $i_1, \dots, i_{N_i} \in \mathcal{N}_i$.

If there is an edge $\{i, j\} \in \mathcal{E}$, then $i, j$ are called *adjacent*. A *path* of length $r$ from vertex $i$ to vertex $j$ is a sequence of $r + 1$ distinct vertices, starting with $i$ and ending with $j$, such that consecutive vertices are adjacent. For $i = j$, the path is called a *cycle*. If there is a path between any two vertices of the graph $\mathcal{G}$, then $\mathcal{G}$ is called *connected*. A connected graph is called a *tree* if it contains no cycles.

## 2.3 Non-linear Model Predictive Control

# 3 Problem Formulation

## 3.1 System Model

Consider a set of $N$ rigid bodies, with $\mathcal{V} = \{1, 2, \ldots, N\}$, $N \geq 2$, operating in a workspace $W \subseteq \mathbb{R}^3$. A coordinate frame $\{i\}, i \in \mathcal{V}$ is attached to each body's center of mass. The workspace is assumed to be modeled as a bounded sphere $\mathcal{B}(\mathbf{0}, r_W)$ expressed in an inertial frame $\{\mathcal{O}\}$.

We consider that over time $t$ each agent $i$ occupies the space of a sphere $\mathcal{B}(\mathbf{p}_i(t), r_i)$, where $\mathbf{p}_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^3$ is the position of the agent's center of mass, and $r_i < r_W$ is the radius of the agent's body. We denote by $\mathbf{q}_i(t) : \mathbb{R}_{\geq 0} \to \mathbb{T}^3, i \in \mathcal{V}$, the Euler angles representing the agents' orientation, with $\mathbf{q}_i = [\phi_i, \theta_i, \psi_i]^\top$. We define $\mathbf{x}_i = [\mathbf{p}^\top, \mathbf{q}_i^\top]^\top, \mathbf{v}_i = [\dot{\mathbf{p}}_i^\top, \boldsymbol{\omega}_i^\top]^\top$, $\mathbf{x}_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^3 \times \mathbb{T}^3 = \mathbb{M}, \mathbf{v}_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^3 \times \mathbb{R}^3 = \mathbb{R}^6$, and model the motion of agent $i$ under second order dynamics:

$$\dot{\mathbf{x}}_i(t) = \boldsymbol{J}_i^{-1}(\mathbf{x}_i)\mathbf{v}_i(t), \tag{1a}$$
$$\mathbf{u}_i = \boldsymbol{M}_i(\mathbf{x}_i)\dot{\mathbf{v}}_i(t) + \boldsymbol{C}_i(\mathbf{x}_i, \dot{\mathbf{x}}_i)\mathbf{v}_i(t) + \mathbf{g}_i(\mathbf{x}_i), \tag{1b}$$

In equation (1a), $\boldsymbol{J}_i : \mathbb{T}^3 \to \mathbb{R}^{6\times 6}$ is a Jacobian matrix that maps the Euler angle rates to the orthogonal angular rates $\mathbf{v}_i$:

$$\boldsymbol{J}_i(\mathbf{x}_i) = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{0}_{3\times 3} \\ \boldsymbol{0}_{3\times 3} & \boldsymbol{J}_q(\mathbf{x}_i) \end{bmatrix}, \text{ with } \boldsymbol{J}_q(\mathbf{x}_i) = \begin{bmatrix} 1 & 0 & -\sin\theta_i \\ 0 & \cos\phi_i & \cos\theta_i \sin\phi_i \\ 0 & -\sin\phi_i & \cos\phi_i \cos\theta_i \end{bmatrix}$$

In order for $\boldsymbol{J}_i$ to be always well-defined (and hence invertible, since $\det(\boldsymbol{J}_i) = \cos\theta_i$) we need to make the following assumption:

?? EVENTUALLY NEEDED? CHECK

**Assumption 3.1.** The angle $\theta_i$ satisfies the inequality

$$-\frac{\pi}{2} < \theta_i(t) < \frac{\pi}{2}, \forall i \in \mathcal{V}, t \in \mathbb{R}_{\geq 0}$$

.

In equation (1b), $\boldsymbol{M}_i : \mathbb{M} \to \mathbb{R}^{6\times 6}$ is the symmetric and positive definite *inertia matrix*, $\boldsymbol{C}_i : \mathbb{M} \times \mathbb{R}^6 \to \mathbb{R}^{6\times 6}$ is the *Coriolis matrix* and $\mathbf{g}_i : \mathbb{M} \to \mathbb{R}^6$ is the *gravity vector*. The aforementioned vector fields are unknown, continuous, and expressed in the respective coordinate frame of each agent. Finally, $\mathbf{u}_i \in \mathbb{R}^6$ is the control input vector representing the 6D generalized *actuation force* acting on the agent.

**Remark 3.1.** According to [?], the matrices $\dot{\boldsymbol{M}}_i - 2\boldsymbol{C}_i, i \in \mathcal{V}$ are skew-symmetric. From [?], we have that a quadratic form of a skew-symmetric matrix is always equal to $\boldsymbol{0}$. Hence, for the matrices $\dot{\boldsymbol{M}}_i - 2\boldsymbol{C}_i$ it holds that:

$$\mathbf{y}^\top \left[\dot{\boldsymbol{M}}_i - 2\boldsymbol{C}_i\right] \mathbf{y} = 0, \forall \mathbf{y} \in \mathbb{R}^n, i \in \mathcal{V}. \tag{2}$$

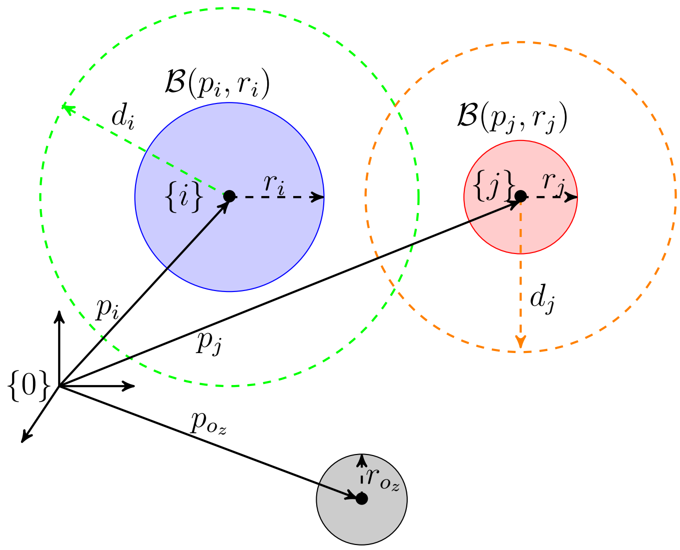


Figure 1: Illustration of two agents $i, j \in \mathcal{V}$ and an static obstacle $o_z$ in the workspace; $\{\mathcal{O}\}$ is the inertial frame, $\{i\}, \{j\}$ are the frames attached to the agents' center of mass, $\mathbf{p}_i, \mathbf{p}_j, \mathbf{p}_{o_z} \in \mathbb{R}^3$ are the positions of the center of mass of the agents $i, j$ and the obstacle $o_z$ respectively, expressed in frame $\{\mathcal{O}\}$. $r_i, r_j, r_{o_z}$ are the radii of the agents $i, j$ and the obstacle $o_z$ respectively. $d_i, d_j$ with $d_i > d_j$ are the agents' sensing ranges. In this figure, agents $i$ and $j$ are not neighbours, since $\mathbf{p}_j \notin \mathcal{B}(\mathbf{p}_i(t), d_i)$.

**Assumption 3.2.** (Measurements Assumption) Each agent $i$ has access to the measurements $\mathbf{p}_i, \mathbf{q}_i, \dot{\mathbf{p}}_i, \boldsymbol{\omega}_i, i \in \mathcal{V}$, pertaining to himself, and a (upper-bounded) sensing range of $d_i$ such that

$$d_i > \max\{r_i + r_j : \forall i, j \in \mathcal{V}, i \neq j\}$$

Therefore, by defining the neighboring set $\mathcal{N}_i(t) = \{j \in \mathcal{V} : \mathbf{p}_j(t) \in \mathcal{B}(\mathbf{p}_i(t), d_i)\}$, agent $i$ also knows at each time instant $t$ all $\mathbf{p}_{j \triangleright i}(t), \mathbf{q}_{j \triangleright i}(t)$ and, since it knows its own $\mathbf{p}_i(t), \mathbf{q}_i(t)$, it can compute all $\mathbf{p}_j(t), \mathbf{q}_j(t), \forall j \in \mathcal{N}_i(t), t \in \mathbb{R}_{\geq 0}$.

In the workspace there are $|Z|$ *static obstacles*, modeled as spheres with centers at positions $\mathbf{p}_{o_z}$ with radii $r_{o_z} \in \mathbb{R}^3, z \in \mathcal{Z} = \{1, \ldots, |Z|\}$. Thus, the obstacles are modeled by the spheres $\mathcal{B}(\mathbf{p}_{o_z}, r_{o_z}), z \in \{1, \ldots, |Z|\}$. The geometry in workspace $W$ of two agents $i$ and $j$ as well as an obstacle $z$ is depicted in Fig. 1.

Let us also define the distance between agents $i, j$ as $d_{ij,a} : \mathbb{R}^6 \to \mathbb{R}_{\geq 0}$, and that between agent $i$ and obstacle $z$ as $d_{iz,o} : \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$:

$$d_{ij,a}(t) = \|\mathbf{p}_i(t) - \mathbf{p}_j(t)\|, \tag{3a}$$

$$d_{iz,o}(t) = \|\mathbf{p}_i(t) - \mathbf{p}_z(t)\| \tag{3b}$$

$\forall i, j \in \mathcal{V}, i \neq j, z \in \mathcal{Z}$, as well as constants $\underline{d}_{ij,a} = r_i + r_j, \underline{d}_{iz,o} = r_i + r_{o_z}$. The latter stand for the minimum distance between two agents, and between an agent and an obstacle. These arise spatially as physical limitations and will be utilized as collision-avoidance constraints.

The topology of the multi-agent network is modeled through the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V} = \{1, \ldots, N\}$ and $\mathcal{E} = \big\{\{i, j\} \in \mathcal{V} \times \mathcal{V} : i \neq j, j \in \mathcal{N}_i(0) \text{ and } i \in \mathcal{N}_j(0)\big\}$. The latter implies that at $t = 0$ the graph is undirected, i.e.,

$$d_{ij,a}(0) < d_i, \forall i \in \mathcal{V}, j \in \mathcal{N}_i(0) \tag{4}$$

We also consider that $\mathcal{G}$ is static in the sense that no edges are added to the graph. We do not exclude, however, edge removal through connectivity loss between initially neighboring agents, which we guarantee to avoid, as presented in the sequel. It is also assumed that at $t = 0$ the neighboring agents are in a *collision-free configuration*, i.e.,

$$\underline{d}_{ij,a} < d_{ij,a}(0), \forall i, j \in \mathcal{V}, i \neq j \tag{5}$$

To capitulate, we assume that at time $t = 0$

$$\underline{d}_{ij,a} < d_{ij,a}(0) < d_i$$

for agent $i$ and all of his adjacent agents $j$ at time $t = 0$.

We will now introduce the assumption we make on the feasibility of the problem. But first, let us denote by $d_o$

$$d_o = \min\{\|\mathbf{p}_{o_z} - \mathbf{p}_{o_{z'}}\| : z, z' \in \mathcal{Z}, z \neq z'\},$$

the distance between the two least distant obstacles in the workspace, by $d_{o,W}$

$$d_{o,W} = \min\{r_W - \big(r_{o_z} + \|\mathbf{p}_{o_z}\|\big) : z \in \mathcal{Z}\},$$

the distance between the least distant obstacle from the boundary of the workspace and the boundary itself, by $D$

$$D = \min\{d_o, d_{o,W}\}$$

the least of these two distances, and by $\Delta$

$$\begin{aligned}\Delta = \max\Big\{ & d_{ij,a} + r_i + r_j : i, j \in \mathcal{V}, i \neq j,\\ & \|\mathbf{p}_k - \mathbf{p}_\ell\| = \mathbf{p}_{k\ell,\mathrm{des}}, \|\mathbf{q}_k - \mathbf{q}_\ell\| = \mathbf{q}_{k\ell,\mathrm{des}}, k \in \mathcal{V}, \ell \in \mathcal{N}_k(0)\Big\}\end{aligned}$$

the *diameter of formation* $\Delta$, which is the distance between the two most distant agents when formation is achieved.

**Assumption 3.3.** (Problem Feasibility Assumption) In order for the problem to be feasible, we make the following natural geometric assumptions:

- All the agents should be able to pass between any two obstacles and between all obstacles and the boundary of the workspace, and simultaneously, without any of them colliding to each other or with the obstacles or the boundary of the workspace. Thus, it is required $D > \sum_{i \in \mathcal{V}} 2r_i$.
- When the multi-agent system reach the desired formation, it should be able to pass between two of the obstacles and between an obstacle and the boundary of the workspace. Thus, it is required $D > \Delta$.

These geometrical assumptions can be summarized in the following inequality:

$$D > \max\left\{\Delta, \sum_{i \in \mathcal{V}} 2r_i\right\} \tag{6}$$

## 3.2 Problem Statement

Due to the fact that the agents are not dimensionless and their communication capabilities are limited, the control protocol, except from achieving desired position formation $\mathbf{p}_{ij,\text{des}}$ and desired formation angles $\mathbf{q}_{ij,\text{des}}$ for all neighboring agents $i \in \mathcal{V}, j \in \mathcal{N}_i(0)$, it should also guarantee for all $t \in \mathbb{R}_{\geq 0}$ that (i) all the agents avoid collision with every other agent, (ii) all the agents avoid collision between all the obstacles; (iii) all the agents avoid collision between the workspace boundary, (iv) all the initial edges are maintained, i.e., connectivity is maintained; (v) the singularity of the Jacobian matrices $J_i$ is avoided. Therefore, all the neighboring agents of agent $i$ must remain within distance less than $d_i$, for all $i \in \mathcal{V}$ and all the agents $i, j \in \mathcal{V}, i \neq j$ must remain within distance greater than $\underline{d}_{ij,a}$. We also make the following assumption that is required on the initial graph topology:

**Assumption 3.4.** The communication graph $\mathcal{G}$ is connected at time $t = 0$ and the agents are in collision- and singularity-free configuration, i.e., (4) and (5) hold, and $\theta_i(0) \neq \pm\frac{\pi}{2}, \forall i \in \mathbb{V}$.

Formally, the control problem under the aforementioned constraints is formulated as follows:

**Problem 3.1.** Given $N$ agents performing in workspace $W$ modeled as bounded sphere $\mathcal{B}(0, r_W)$, with spherical obstacles $\mathcal{B}(p_{o_z}, r_{o_z}), z \in \mathcal{Z}$, governed by the dynamics (1), under the Assumptions 1-3, under the geometric feasibility constraint (6) and given the desired inter-agent distances and angles $p_{ij,\text{des}}, q_{ij,\text{des}}$, with $\underline{d}_{ij,a} < p_{ij,\text{des}} < \bar{d}_i$, $\forall i \in \mathcal{V}, j \in \mathcal{N}_i(0)$, design decentralized control laws $u_i \in \mathbb{R}^6, i \in \mathcal{V}$ such that:

- $\forall i \in \mathcal{V}, j \in \mathcal{N}_i(0)$, the following hold:
  1) $\lim\limits_{t \to \infty} [p_i(t) - p_j(t) - p_{ij,\text{des}}] = 0_{3\times 1}$,
  2) $\lim\limits_{t \to \infty} [q_i(t) - q_j(t) - q_{ij,\text{des}}] = 0_{3\times 1}$.
- $\forall i, j \in \mathcal{V}, i \neq j$ the following holds:
  3) $\|p_i(t) - p_j(t)\| > \underline{d}_{ij,a}, \forall\, t \in \mathbb{R}_{\geq 0}$.
- $\forall\, i \in \mathcal{V}, z \in \mathcal{Z}$ the following holds:
  4) $\mathcal{B}(p_i(t), r_i) \cap \mathcal{B}(p_{o_z}(t), r_{o_z}) = \emptyset, \forall\, t \in \mathbb{R}_{\geq 0}$.
- $\forall i \in \mathcal{V}, j \in \mathcal{N}_i(0)$ the following holds:
  5) $\|p_i(t) - p_j(t)\| < \bar{d}_i, \forall\, t \in \mathbb{R}_{\geq 0}$.

The aforementioned specifications imply the following:

- 1 stands for formation control;
- 2 stands for orientation alignment;
- 3 stands for inter-agent collision avoidance;
- 4 stands for collision avoidance between the agents and the obstacles;
- 5 stands for connectivity maintenance of the initial graph;

# 4 Problem Solution

In this section, a systematic solution to Problem 3.1 is introduced. Our overall approach builds on formulating a model predictive control optimization problem for each agent such that solution this captures all the desired control specifications. The following analysis is performed:

1. The form of the proposed optimizaton problem is described in Section 4.1.
2. The feasibility of the problem is given in 4.2.
3. The stability analysis is given in 4.3.

## 4.1 Distributed MPC

## 4.2 Feasibility Analysis

## 4.3 Stability Analysis

# 5 Simulation Results

# 6 Conclusions and Future Work