Deadlock Analysis and Resolution for Multi-Robot Systems

Jaskaran Singh Grover, Changliu Liu, and Katia Sycara *

The Robotics Institute, Carnegie Mellon University
{jaskarag,cliu6,sycara}@andrew.cmu.edu

Abstract. Collision avoidance for multirobot systems is a well studied problem. Recently, control barrier functions (CBFs) have been proposed for synthesizing controllers that guarantee collision avoidance and goal stabilization for multiple robots. However, it has been noted that reactive control synthesis methods (such as CBFs) are prone to *deadlock*, an equilibrium of system dynamics that causes robots to come to a standstill before reaching their goals. In this paper, we formally derive characteristics of deadlock in a multirobot system that uses CBFs. We propose a novel approach to analyze deadlocks resulting from optimization based controllers (CBFs) by borrowing tools from duality theory and graph enumeration. Our key insight is that system deadlock is characterized by a force-equilibrium on robots and we show how complexity of deadlock analysis increases approximately exponentially with the number of robots. This analysis allows us to interpret deadlock as a subset of the state space, and we prove that this set is non-empty, bounded and located on the boundary of the safety set. Finally, we use these properties to develop a provably correct decentralized algorithm for deadlock resolution which ensures that robots converge to their goals while avoiding collisions. We show simulation results of the resolution algorithm for two and three robots and experimentally validate this algorithm on Khepera-IV robots.

Keywords: Collision Avoidance, Optimization and Optimal Control

1 Introduction

Multirobot systems have been studied thoroughly for solving a variety of complex tasks such as search and rescue [1], sensor coverage [2] and environmental exploration [3]. Global coordinated behaviors result from executing local control laws on individual robots interacting with their neighbors [4], [5]. Typically, the local controllers running on these robots are a combination of a task-based controller responsible for completion of a primary objective and a reactive collision avoidance controller. However, including a hand-engineered safety control no longer guarantees that the original task will be satisfied [6]. This problem becomes all the more pronounced when the number of robots increases. Motivated by this bottleneck, our paper focuses on an algorithmic analysis of the performance-safety trade-offs that result from augmenting a task-based controller with collision avoidance constraints as done using CBF based quadratic programs (QPs) [7]. Although CBF-QPs mediate between safety and performance in a rigorous way, yet ultimately they are distributed local controllers. Such approaches exhibit a lack of look-ahead, which causes the robots to be trapped in *deadlocks* as noted in [8,9,10]. In deadlock, the robots stop while still being away from their goals and persist in this state unless intervened. This occurs because robots reach a state where conflict becomes inevitable, *i.e.*

* This research was supported by the DARPA Cooperative Agreement HR00111820051.

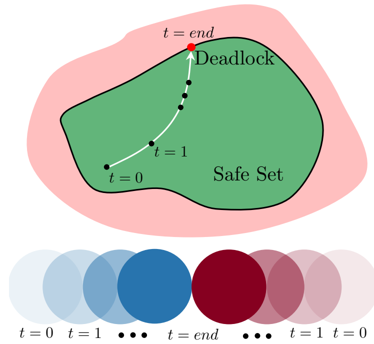


Fig. 1: Two robots moving towards each other fall in deadlock. System state converges to the boundary of safe set.

a control favoring goal stabilization will violate safety (see red dot in Fig. 1). Hence, the only feasible strategy is to remain static. Although small perturbations can steer the system away from deadlock, there is no guarantee that robots will not fall back in deadlock. To circumvent these issues, this work addresses the following technical questions:

1. What are the characteristics of a system in deadlock ?
2. What all geometric configurations of robots are admissible in deadlock ?
3. How can we leverage this information to provably exit deadlock using decentralized controllers?

To address these questions, we first review technical definitions for CBF based QPs [10] to synthesize controllers for collision avoidance and goal stabilization in section 3. In section 4, we recall the definition of deadlock and use KKT conditions to motivate a novel set theoretic interpretation of deadlock with an eye towards devising controllers that evade/exit this set. We use graph enumeration to highlight the combinatorial complexity of geometric configurations of robots admissible in deadlock. Following this development, in section 5 and section 6, we focus on the easier to analyze cases for two and three robots respectively and examine mathematical properties of the deadlock set for these cases. We show that this set is on the boundary of the safety set, is non-empty and bounded. In section 7, we show how to design a provably-correct decentralized controller to make the robots exit deadlock. We demonstrate this strategy on two and three robots in simulation, and experimentally on Khepera-IV robots. Finally, we conclude with directions for future work.

2 Prior Work

Several existing methods provide inspiration for the results presented here. Of these, two are especially relevant: in the first category, we describe prior methods for collision avoidance and in the second, we focus on deadlock resolution.

2.1 Prior Work on Avoidance Control

Avoidance control is a well-studied problem with immediate applications for planning collision-free motions for multirobot systems. Classical avoidance control assumes a worst case scenario with no cooperation between robots [11,12] Cooperative collision avoidance is explored in [13,14] where avoidance control laws are computed using value functions. Velocity obstacles have been proposed in [15] for motion planning in dynamic environments. They select avoidance maneuvers outside of robot's velocity obstacles to avoid static and moving obstacles by means of a tree-search. While this method is prone to undesirable oscillations, the authors in [16,17,18] propose reciprocal velocity obstacles that are immune to such oscillations. More recently, control barrier function based controllers have been used in [6,10] to mediate between safety and performance using QPs.

2.2 Prior Work on Deadlock Resolution

The importance of coordinating motions of multiple robots while simultaneously ensuring safety, performance and deadlock prevention has been acknowledged in works as early as in [9]. Here, authors proposed scheduling algorithms to asynchronously coordinate motions of two manipulators to ensure that their trajectories remain collision and deadlock free. In the context of mobile robots, [19] identified the presence of deadlocks in a cooperative scenario using mobile robot troops. To the best of our knowledge, [20] were the first to propose algorithms for deadlock resolution specifically for multiple mobile robots. Their strategy for collision avoidance modifies planned paths by inserting idle times and resolves deadlocks by asking the trajectory planners of each robot to plan an alternative trajectory until deadlock is resolved. Authors in [21] proposed coordination graphs to resolve deadlocks in robots navigating through narrow corridors. [10,22] added perturbation terms to their controllers for avoiding deadlock.

Differently from these, we characterize analytical properties of system states when in deadlock. We explicitly analyze controls from CBF based QPs and demonstrate that intuitive explanations for systems in deadlock are indeed recovered using duality. Our analysis can be extended to reveal bottlenecks of any optimization based controller synthesis method. Additionally, we use graph enumeration to highlight the complexity of this analysis. We do not consider additive perturbations for resolving deadlocks, since there are no formal guarantees. Instead, we use feedback linearization and the geometric properties recovered from duality to guide the design of a provably correct controller that ensures safety, performance and deadlock resolution.

3 Avoidance Control with CBFs: Review

In this section, we review CBF based QPs used for synthesizing controllers that mediate between safety (collision avoidance) and performance (goal-stabilization) for multirobot systems. We refer the reader to [10] for a comprehensive treatment on this subject, since our work builds on top of their approach. Assume that we have N mobile robots, each of which follows double-integrator dynamics:

$$\begin{bmatrix}\dot{\boldsymbol{p}}_i\\ \dot{\boldsymbol{v}}_i\end{bmatrix}=\begin{bmatrix}\dot{\boldsymbol{v}}_i\\ \mathbf{0}\end{bmatrix}+\begin{bmatrix}\mathrm{O}\\ \mathbf{I}\end{bmatrix}\boldsymbol{u}_i, \tag{1}$$

where $\boldsymbol{p}_i=(x_i,y_i)\in\mathbb{R}^2$ represents the position of robot i, $\boldsymbol{v}_i\in\mathbb{R}^2$ represents its velocity and $\boldsymbol{u}_i\in\mathbb{R}^2$ represents the acceleration (*i.e.* control). The collective state of robot i is denoted by $\boldsymbol{z}_i=(\boldsymbol{p}_i,\boldsymbol{v}_i)$ and the collective state of the multirobot system is denoted as $\boldsymbol{Z}=(\boldsymbol{z}_1,\boldsymbol{z}_2,...,\boldsymbol{z}_N)$. Assume that each robot has maximum allowable acceleration limits $|\boldsymbol{u}_i|\leq\alpha_i$ that represent actuator constraints. The problem of goal stabilization with avoidance control requires that each robot i must reach a goal $\boldsymbol{p}_{d_i}$ while avoiding collisions with every other robot $j\neq i$. For reaching a goal, assume that there is a prescribed PD controller

$\hat{\boldsymbol{u}}_i(\boldsymbol{z}_i) = -k_p(\boldsymbol{p}_i - \boldsymbol{p}_{d_i}) - k_v\boldsymbol{v}_i$ with $k_p,k_v>0$. This controller is chosen as a nominal reference controller because by itself, it ensures exponential stabilization of each robot to its goal. However, there is no guarantee that the resulting trajectories will be collision free.

Based on [10], a safety constraint is formulated for every pair of robots to ensure mutually collision free motions. This constraint is mathematically posed by defining a function that maps the joint state space of robots i and j to a real-valued safety index *i.e.* $h:\mathbb{R}^4\times\mathbb{R}^4\longrightarrow\mathbb{R}$. For a desired safety margin distance D_s, this index is defined as

$$h_{ij} = \sqrt{2(\alpha_i+\alpha_j)(\left\|\Delta\boldsymbol{p}_{ij}\right\| - D_s)} + \frac{\Delta\boldsymbol{p}_{ij}^T\Delta\boldsymbol{v}_{ij}}{\left\|\Delta\boldsymbol{p}_{ij}\right\|}. \tag{2}$$

Robots i and j are considered to be collision-free if their states $(\boldsymbol{z}_i,\boldsymbol{z}_j)$ are such that $h_{ij}(\boldsymbol{z}_i,\boldsymbol{z}_j)\geq 0$. We define "safe set" as the 0-level superset of h_{ij} *i.e.* $\mathcal{C}_{ij} := \{(\boldsymbol{z}_i,\boldsymbol{z}_j)\in\mathbb{R}^8 \,|\, h_{ij}(\boldsymbol{z}_i,\boldsymbol{z}_j)\geq 0\}$. The boundary of the safe set is

$$\partial\mathcal{C}_{ij} = \{(\boldsymbol{z}_i,\boldsymbol{z}_j)\in\mathbb{R}^4 | h(\boldsymbol{z}_i,\boldsymbol{z}_j)=0\} \tag{3}$$

Assuming that the initial positions of robots i and j are in the safe set $\mathcal{C}_{ij}$, we would like to synthesize controls $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ that ensure that future states of the robots i and j also stay in $\mathcal{C}_{ij}$. This can be achieved by ensuring that

$$\frac{dh_{ij}}{dt} \geq -\kappa(h_{ij}), \tag{4}$$

where we choose, $\kappa(h) := h^3$. For the given choice of h, (4) can be rewritten as

$$-\Delta\boldsymbol{p}_{ij}^T\Delta\boldsymbol{u}_{ij} \leq b_{ij}, \text{ where} \tag{5}$$

$$b_{ij} = \left\|\Delta\boldsymbol{p}_{ij}\right\| h_{ij}^3 + \frac{(\alpha_i+\alpha_j)\Delta\boldsymbol{p}_{ij}^T\Delta\boldsymbol{v}_{ij}}{\sqrt{2(\alpha_i+\alpha_j)(\left\|\Delta\boldsymbol{p}_{ij}\right\| - D_s)}} + \|\Delta\boldsymbol{v}_{ij}\|^2 - \frac{(\Delta\boldsymbol{p}_{ij}^T\Delta\boldsymbol{v}_{ij})^2}{\left\|\Delta\boldsymbol{p}_{ij}\right\|^2} \tag{6}$$

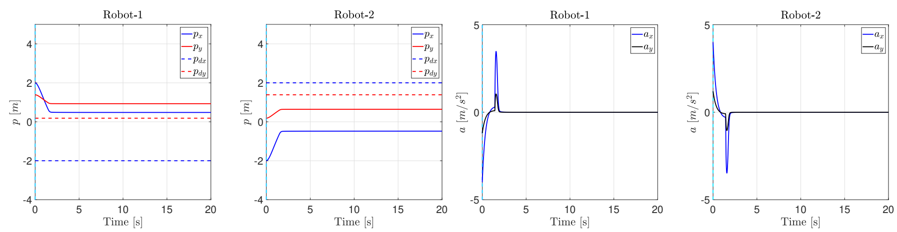
This constraint is distributed on robots i and j as:

$$-\Delta\boldsymbol{p}_{ij}^T\boldsymbol{u}_i \leq \frac{\alpha_i}{\alpha_i+\alpha_j}b_{ij} \text{ and } \Delta\boldsymbol{p}_{ij}^T\boldsymbol{u}_j \leq \frac{\alpha_j}{\alpha_i+\alpha_j}b_{ij} \tag{7}$$

Therefore, any $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ that satisfy (7) will ensure collision free trajectories for robots i and j in the multirobot system. Note that these constraints are linear in $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ for a given state $(\boldsymbol{z}_i,\boldsymbol{z}_j)$. Therefore, the feasible set of controls is convex. Assuming robot i wants to avoid collisions with its M neighbors, there will be M collision avoidance constraints. To mediate between safety and goal stabilization, a QP is posed that computes a controller closest to the PD control $\hat{\boldsymbol{u}}_i(\boldsymbol{z}_i)$ (in 2-norm) and satisfies the M collision avoidance constraints:

$$\begin{aligned} \underset{\boldsymbol{u}_i}{\text{minimize}} \quad & \|\boldsymbol{u}_i - \hat{\boldsymbol{u}}_i(\boldsymbol{z}_i)\|_2^2 \\ \text{subject to} \quad & -\Delta\boldsymbol{p}_{ij}^T\boldsymbol{u}_i \leq \frac{\alpha_i}{\alpha_i+\alpha_j}b_{ij} \;\; j\in\{1,...,M\} \\ & |\boldsymbol{u}_i| \leq \boldsymbol{\alpha}_i \end{aligned} \tag{8}$$

This QP has $(M+4)$ constraints (M from collision avoidance with M neighbors and four from acceleration limits). Each robot i executes a local version of this QP and computes its optimal $\boldsymbol{u}_i^*$ at every time step. As long as the QP remains feasible, the generated control $\boldsymbol{u}_i^*$ ensures collision avoidance of robot i with its neighbors. In the next section, we derive an analytical expression for $\boldsymbol{u}_i^*$ as a function of $(\boldsymbol{z}_1,...,\boldsymbol{z}_N)$ to analyze the closed-loop dynamics of the ego robot and use this to investigate the incidence of deadlocks resulting from this technique.



(a) Positions (b) Accelerations (control inputs)

Fig. 2: Positions and accelerations of robots falling in deadlock. Note that $\lim_{t\to\infty} p_{x,y} \neq p_{d_{x,y}}$ yet $\lim_{t\to\infty} a_{x,y} = 0$. Video: https://tinyurl.com/y4ylzwh8

4 Analysis of N Robot Deadlock

We reviewed the formulation of multirobot collision avoidance and goal stablization using the framework of CBF based QPs. In this section, we will show that this approach can result in deadlocks (depending on the initial conditions and goals of robots). We want to analyze qualitative properties of a robot in deadlock. Towards that end, we will investigate the KKT conditions [23] of the problem in (8). Our goal is to use these conditions to compute properties of geometric configurations of robots in deadlock and then exploit these properties to make the robots exit deadlock. Fig. 2 shows the states of two robots that have fallen in deadlock while executing controllers based on (8). Notice from Fig. 2(a) that the positions of robots have converged, but **not** to their respective goals. Therefore, the outputs from the prescribed PD controller will still be non-zero after convergence. However, the control inputs from (8) have already converged to zero Fig. 2(b). From these observations, deadlock is defined as follows [10]

Definition 1. *A robot i is in deadlock if $\boldsymbol{u}_i^* = 0$, $\boldsymbol{v}_i = 0$, $\hat{\boldsymbol{u}}_i \neq 0$ and $\boldsymbol{p}_i \neq \boldsymbol{p}_{d_i}$*

In simpler terms, for a robot to be in deadlock, it should be static *i.e.* its velocity should be zero, and the output from the QP based controller should also be zero, even though the reference PD controller reports non-zero acceleration since the robot is not at its intended goal. We now look at the KKT conditions for the optimization problem in (8).

4.1 KKT Conditions

Recall that each robot i computes a control by solving a local QP as in (8). Define $\boldsymbol{a}_j := -\Delta \boldsymbol{p}_{ij}$ and $\hat{b}_j = \frac{\alpha_i}{\alpha_i + \alpha_j} b_{ij}$.We will drop subscript i and implicitly assume that the QP is being solved for the ego robot. Hence, we rewrite (8) as:

$$\begin{aligned} \underset{\boldsymbol{u}}{\text{minimize}} \quad & \|\boldsymbol{u} - \hat{\boldsymbol{u}}\|_2^2 \\ \text{subject to} \quad & \widetilde{A}\boldsymbol{u} \leq \widetilde{\boldsymbol{b}} \end{aligned} \tag{9}$$

where $\tilde{A} := (\boldsymbol{a}_1^T;...;\boldsymbol{a}_M^T;\boldsymbol{e}_1^T;...;-\boldsymbol{e}_2^T)$ and $\tilde{\boldsymbol{b}} := (\hat{b}_1;...;\hat{b}_M;\alpha;...;\alpha)$. Let $\tilde{\boldsymbol{a}}_k$ denote the k'th row of $\tilde{A}$ and $\tilde{b}_k$ denote the k'th element of $\tilde{\boldsymbol{b}}$. The Lagrangian for (9) is

$$L(\boldsymbol{u}, \boldsymbol{\mu}) = \|\boldsymbol{u} - \hat{\boldsymbol{u}}\|_2^2 + \sum_{k=1}^{M+4} \mu_k (\tilde{\boldsymbol{a}}_k^T \boldsymbol{u} - \tilde{b}_k) \tag{10}$$

Let $(\boldsymbol{u}^*, \boldsymbol{\mu}^*)$ be the optimal primal-dual solution to (9). The KKT conditions are

1. Stationarity: $\nabla_{\boldsymbol{u}} L(\boldsymbol{u},\boldsymbol{\mu})|_{(\boldsymbol{u}^*,\boldsymbol{\mu}^*)} = \boldsymbol{0}$

$$\implies \boldsymbol{u}^* = \hat{\boldsymbol{u}} - \frac{1}{2}\sum_{k=1}^{M+4} \mu_k^* \tilde{\boldsymbol{a}}_k^T. \tag{11}$$

2. Primal Feasibility

$$\tilde{\boldsymbol{a}}_k^T \boldsymbol{u}^* \leq \tilde{b}_k \ \forall k \in \{1,2,...,M+4\} \tag{12}$$

3. Dual Feasibility

$$\mu_k^* \geq 0 \ \forall k \in \{1,2,...,M+4\} \tag{13}$$

4. Complementary Slackness

$$\mu_k^* \cdot (\tilde{\boldsymbol{a}}_k^T \boldsymbol{u}^* - \tilde{b}_k) = 0 \ \forall k \in \{1,2,...,M+4\} \tag{14}$$

Define the set of active and inactive constraints as follows:

$$\mathcal{A}(\boldsymbol{u}^*) = \{k \in \{1,2,...,M+4\} \,|\, \tilde{\boldsymbol{a}}_k^T \boldsymbol{u}^* = \tilde{b}_k\} \tag{15}$$

$$\mathcal{IA}(\boldsymbol{u}^*) = \{k \in \{1,2,...,M+4\} \,|\, \tilde{\boldsymbol{a}}_k^T \boldsymbol{u}^* < \tilde{b}_k\} \tag{16}$$

Using complementary slackness from (14), we deduce

$$\mu_k^* = 0 \ \forall k \in \mathcal{IA}(\boldsymbol{u}^*) \tag{17}$$

Therefore, we can restrict the sum in (11) to only the set of active constraints

$$\boldsymbol{u}^* = \hat{\boldsymbol{u}} - \frac{1}{2}\sum_{k \in \mathcal{A}(\boldsymbol{u}^*)} \mu_k^* \tilde{\boldsymbol{a}}_k^T \tag{18}$$

4.2 KKT Conditions for the deadlock case

From Def. 1, we know that in deadlock, $\boldsymbol{u}^* = \boldsymbol{0}$, $\hat{\boldsymbol{u}} \neq 0$ and $\boldsymbol{v} = \boldsymbol{0}$. We conclude:

1. In deadlock, $\boldsymbol{u}^* \neq \hat{\boldsymbol{u}}$ *i.e.* the solution to the QP is not equal to the prescribed PD controller (which means that $\hat{\boldsymbol{u}}$ is infeasible in deadlock *i.e.* $\boldsymbol{a}^T \hat{\boldsymbol{u}} \nleq \hat{b}$).
2. $\boldsymbol{u}^* = \boldsymbol{0} \implies \boldsymbol{u}^* \neq \pm\boldsymbol{\alpha}$. This implies that at least the last four constraints in $\tilde{A}\boldsymbol{u} \leq \tilde{b}$ are inactive *i.e.* $\{M+1, M+2, M+3, M+4\} \in \mathcal{IA}(\boldsymbol{u}^*)$ in deadlock.

Using these observations, we rewrite the KKT conditions for the deadlock case:

1. Stationarity: $\nabla_{\boldsymbol{u}} L(\boldsymbol{u},\boldsymbol{\mu})|_{(\boldsymbol{0},\boldsymbol{\mu}^*)} = \boldsymbol{0}$

$$\implies \hat{\boldsymbol{u}} = \frac{1}{2}\sum_{k \in \mathcal{A}(\boldsymbol{u}^*)} \mu_k^* \tilde{\boldsymbol{a}}_k^T \tag{19}$$

2. Primal Feasibility

$$\tilde{b}_k \geq 0 \ \forall k \in \{1,2,...,M+4\} \tag{20}$$

3. Dual Feasibility

$$\mu_k^* \geq 0 \ \forall k \in \{1,2,...,M+4\} \tag{21}$$

4. Complementary Slackness

$$\mu_k^* \cdot (\tilde{\boldsymbol{a}}_k^T \boldsymbol{u}^* - \tilde{b}_k) = 0$$
$$\implies \mu_k^* \cdot \tilde{b}_k = 0 \ \forall j \in \{1,2,...,M+4\} \tag{22}$$

Based on these conditions, we will now motivate a set-theoretic interpretation of deadlock. Assume that the state of the ego robot is $\boldsymbol{z} = (\boldsymbol{p},\boldsymbol{v})$ and it has M neighbors denoted as $\boldsymbol{Z}_{nb.}$. Define $P \in \mathbb{R}^{2\times 4}$ and $V \in \mathbb{R}^{2\times 4}$ appropriately to extract the position and velocity components from $\boldsymbol{z}$ *i.e.* $\boldsymbol{p} = P\boldsymbol{z}$ and $\boldsymbol{v} = V\boldsymbol{z}$. Finally, define $\mathcal{D}$ as:

$$\mathcal{D}(\boldsymbol{z} \,|\, \boldsymbol{Z}_{nb.}) = \{\boldsymbol{z} \in \mathbb{R}^4 \,|\, \boldsymbol{u}^*(\boldsymbol{z}) = 0, \hat{\boldsymbol{u}}(\boldsymbol{z}) \neq 0, V\boldsymbol{z} = 0, \mu_k^*(\boldsymbol{Z}) > 0 \ \forall \ k \in \mathcal{A}(\boldsymbol{u^*})\} \tag{23}$$

The set $\mathcal{D}$ is defined as the set of all states of the ego robot which satisfy the criteria of being in deadlock. We have combined the conditions of deadlock into a set theoretic definition. Note that for each robot, its set of deadlock states depends on the states of its neighboring robots. This is because the Lagrange multipliers depend on the states of all robots. The motivation behind stating this definition is to interpret deadlock as a bonafide set in the state space of the ego robot and derive a control strategy that makes the robot evade/exit this set. We now rewrite this definition in more easily interpretable conditions. From (18) and (19), note that

$$\boldsymbol{u}^*(\boldsymbol{z})=0 \Longleftrightarrow \hat{\boldsymbol{u}}(\boldsymbol{z})=\frac{1}{2}\sum_{k\in\mathcal{A}(\boldsymbol{u}^*(\boldsymbol{z}))}\mu_k^*\tilde{\boldsymbol{a}}_k \tag{24}$$

Since $\tilde{\boldsymbol{a}}_k=-\Delta\boldsymbol{p}_{ik}=-P(\boldsymbol{z}-\boldsymbol{z}_k)$, we rewrite (24) as:

$$\boldsymbol{u}^*(\boldsymbol{z})=0 \Longleftrightarrow \hat{\boldsymbol{u}}(\boldsymbol{z})=-\frac{1}{2}\sum_{k\in\mathcal{A}(\boldsymbol{u}^*(\boldsymbol{z}))}\mu_k^*P(\boldsymbol{z}-\boldsymbol{z}_k) \tag{25}$$

We will use this condition to replace the $\boldsymbol{u}^*(\boldsymbol{z})=0$ criterion in the def. of $\mathcal{D}$ in (23). Secondly, we know that prescribed controller $\hat{\boldsymbol{u}}(\boldsymbol{z})$ is a PD controller. Define the goal state as $\boldsymbol{z}_d=(\boldsymbol{p}_d,\boldsymbol{0})$. Noting that $\hat{\boldsymbol{u}}(\boldsymbol{z})\neq 0$ and $\boldsymbol{v}=\boldsymbol{0}$,

$$\hat{\boldsymbol{u}}(\boldsymbol{z})=-k_p(\boldsymbol{p}-\boldsymbol{p}_d)-k_v\boldsymbol{v}\neq\boldsymbol{0} \Longleftrightarrow P(\boldsymbol{z}-\boldsymbol{z}_d)\neq\boldsymbol{0} \tag{26}$$

This criterion is restating that in deadlock the ego robot is not at its goal. The final condition is that the velocity of the ego robot is zero *i.e.* $\boldsymbol{v}=\boldsymbol{0}\Longleftrightarrow V\boldsymbol{z}=\boldsymbol{0}$. Combining these conditions, we rewrite the definition of the deadlock from (23) as follows:

$$\mathcal{D}(\boldsymbol{z}\,|\,\boldsymbol{Z}_{nb.})=\{\boldsymbol{z}\in\mathbb{R}^4|\hat{\boldsymbol{u}}(\boldsymbol{z})=-\frac{1}{2}\sum_{k\in\mathcal{A}(\boldsymbol{u}^*(\boldsymbol{z}))}\mu_k^*P(\boldsymbol{z}-\boldsymbol{z}_k),P(\boldsymbol{z}-\boldsymbol{z}_d)\neq 0,V\boldsymbol{z}=0,$$
$$\mu_k^*>0\ \forall\ k\in\mathcal{A}(\boldsymbol{u}^*(\boldsymbol{z}))\} \tag{27}$$

Building on the definition of one robot deadlock, we motivate *system deadlock* to be the set of states where all robots are in deadlock and is defined as

$$\mathcal{D}_{system}=\{(\boldsymbol{z}_1,\boldsymbol{z}_2,\cdots,\boldsymbol{z}_N)\in\mathbb{R}^{4N}\,|\,\boldsymbol{z}_i\in\mathcal{D}(\boldsymbol{z}_i\,|\,\boldsymbol{Z}^i_{nb.})\ \forall i\in\{1,2,\cdots,N\}\} \tag{28}$$

For the rest of the paper, we will focus our analysis on system deadlock. This is because the case where only a subset of robots are in deadlock can be decomposed into subproblems where a subset is in *system deadlock* and the remaining robots free to move. The next section focuses on the geometric complexity analysis of *system deadlock*.

4.3 Graph Enumeration based Complexity Analysis of Deadlock

The Lagrange multipliers μ_k^* are in general, a nonlinear function of the state of robots $\boldsymbol{z}$. Their values depend on which constraints are active/inactive (an example calculation is shown in (32)). An active constraint will in-turn determine the set of possible geometric configurations that the robots can take when they are in deadlock (sections 5 and 6) and this in turn will guide the design of our deadlock resolution algorithm (section 7). Therefore, we are interested in deriving all possible combinations of active/inactive constraints that the robots can assume once in deadlock. But first we derive upper and lower bounds for the number of valid configurations in *system deadlock*.

We can interpret an active collision avoidance constraint between robots i and j as an undirected edge between vertices i and j in a graph formed by N labeled vertices, where each vertex represents a robot. The following property (which follows from symmetry) allows the edges to be undirected.

Lemma 1. *If robot i and j are both in deadlock and i's constraint with j is active (inactive), then j's constraint with i is also active (inactive).*

Upper Bound Given N vertices, there are ${}^{N}C_2$ distinct pairs of edges possible. The overall system can have any subset of those edges. Since a set with ${}^{N}C_2$ members has $2^{{}^{N}C_2}$ subsets, we conclude that there are $2^{{}^{N}C_2}$ possible graphs. In other words, given N robots, the number of configurations that are admissible in deadlock is $2^{{}^{N}C_2}$. However, this number is an upper bound because it includes cases where a given vertex can be disconnected from all other vertices, which is not valid in *system deadlock* as shown next.

Lower Bound We further impose the restriction that each vertex in the graph have at-least one edge *i.e.* each robot have at-least one constraint active with some other robot. This is because if a robot has no active constraints *i.e.* $\mu_k^*=0\ \forall k$ then from (24), we will get $\boldsymbol{u}^*(\boldsymbol{z})=\hat{\boldsymbol{u}}(\boldsymbol{z})=\mathbf{0}$ which would contradict the definition of deadlock for that robot and hence contradict *system deadlock*. From this observation, it follows that the set of graphs that are valid in *system deadlock* is a superset of connected simple graphs that can be formed by N labeled vertices. This is because there could be graphs that are not simply connected yet admissible in deadlock. While this argument is based on algebraic qualifiers resulting from the 'edge' interpretation of collision avoidance constraints, it is possible that some simply connected graphs may not be geometrically feasible due to restrictions imposed by Euclidean geometry. Graphs that are (a) simply-connected (to enforce deadlock for each robot), (b) have N labelled vertices (since each robot has an ID), (c) are embedded in $\mathbb{R}^2$ (since the robots/environment are planar), (d) have Euclidean distance between connected vertices equal to D_s, (e) that between unconnected vertices greater than D_s, and (f) have exactly one or two edges for each vertex, necessarily represent admissible geometric configurations of robots in *system deadlock*. The reason for qualifiers (d) and (e) is explained in the proof of theorem 1. (f) is needed because the decision variables in (9) are in $\mathbb{R}^2$, so there can be either one or two active constraints per ego robot. The number of graphs meeting qualifiers (a) and (b) can be obtained using the following recurrence relation [24]

$$d_N=2^{{}^{N}C_2}-\frac{1}{N}\sum_{k=1}^{N-1}k\ {}^{N}C_k 2^{{}^{N-k}C_2}d_k \tag{29}$$

For $N=\{1, 2, 3, 4\}$, this number is $\{1, 1, 4, 38\}$. The number of graphs meeting qualifiers (a), (c) and (d) can be obtained by calculating the number of connected matchstick graphs on N nodes [25]. The number of graphs meeting (b) and (d) was obtained in [26] and is exponential in N^2 (for unit distance graphs). A lower bound for graphs satisfying all qualifiers (a)-(f) can be shown to be $0.5(N+1)(N-1)!$ as follows ($N\geq 3$). Consider a cyclic graph whose each node is the vertex of an N regular polygon with side D_s. Such a graph necessarily satisfies (a)-(f). Re-arrangements of its vertices gives rise to $0.5(N-1)!$ graphs. Likewise, a graph with nodes along an open chain also satisfies (a)-(f), and gives $0.5N!$ rearrangements. Thus, the total is $0.5(N-1)!+0.5N!=0.5(N+1)(N-1)!$ It is well known that factorial overtakes exponential, thus highlighting the increase in the number of geometric configurations. Our MATLAB simulations show that the exact number of configurations for $N=\{1, 2, 3, 4\}$ are $\{1, 1, 4, 18\}$ whereas our bound gives $\{1, 1, 4, 15\}$. This simulation demonstrates the explosion in the number of possible geometric configurations that are admissible in *system deadlock* with increasing number of robots. Therefore for further analysis, we will restrict to the case of two and three robots.

5 Two-Robot Deadlock

In section 4, we proposed a set-theoretic definition of deadlock for a specific robot in an N robot system. In this section, we will refine the KKT conditions derived in section 4.2 for the case of two robots in the system. This setting reveals several important underlying characteristics of the system that are extendable to the N robot case, as will be shown for $N=3$. One key feature of a two-robot system is that a single robot by itself cannot be in deadlock *i.e.* either both robots are in deadlock or neither. This is because the sole collision avoidance constraint is symmetric due to Lemma 1. Hence, a two-robot system can only exhibit *system deadlock*. Additionally since the ego robot avoids collision only with the one other robot, there is no sum in (24) *i.e.*

$$\boldsymbol{u}^*(\boldsymbol{z})=0 \Longleftrightarrow \hat{\boldsymbol{u}}(\boldsymbol{z})=\frac{1}{2}\mu^*\boldsymbol{a} \tag{30}$$

The left hand side of this equation is $\hat{\boldsymbol{u}}(\boldsymbol{z})=-k_p(\boldsymbol{p}_{ego}-\boldsymbol{p}_d)$. The right hand side is $\frac{1}{2}\mu^*\boldsymbol{a}=-\frac{1}{2}\mu^*(\boldsymbol{p}_{ego}-\boldsymbol{p}_{neighbor})$. Writing this another way, we have $-k_p(\boldsymbol{p}_{ego}-\boldsymbol{p}_d)+\frac{1}{2}\mu^*(\boldsymbol{p}_{ego}-\boldsymbol{p}_{neighbor})=\boldsymbol{0}$. The first term as represents an attractive force that pulls the ego robot towards goal $\boldsymbol{p}_d$. Since $\mu^*>0$, the second term represents a repulsive force pushing the ego robot away from its neighbor. Thus, *system deadlock* occurs when the net force due to attraction and repulsion on each robot vanishes (see Fig. 3(a)). We now define the *system deadlock* set $\mathcal{D}_{system}$ using (27) and (28):

$$\begin{aligned}\mathcal{D}_{system}=\{(\boldsymbol{z}_1,\boldsymbol{z}_2)\in\mathbb{R}^8 \,|\, \hat{\boldsymbol{u}}_1=\frac{1}{2}\mu_1^*\boldsymbol{a}_1,\ \hat{\boldsymbol{u}}_2=\frac{1}{2}\mu_2^*\boldsymbol{a}_2,\ \mu_1^*>0,\ \mu_2^*>0,\\ (P(\boldsymbol{z}_1-\boldsymbol{z}_{d_1}),P(\boldsymbol{z}_2-\boldsymbol{z}_{d_2}))\neq(\boldsymbol{0},\boldsymbol{0}),\ (V\boldsymbol{z}_1,V\boldsymbol{z}_2)=(\boldsymbol{0},\boldsymbol{0})\}.\end{aligned} \tag{31}$$

where $\boldsymbol{a}_1=-\boldsymbol{a}_2=-(\boldsymbol{p}_1-\boldsymbol{p}_2)$. Next, we derive analytical expressions for the Lagrange multipliers μ_1^*,μ_2^*. Depending on whether the collision avoidance constraint is active/inactive at the optimum, there are two cases:

Case 1: The constraint $\boldsymbol{a}^T\boldsymbol{u}\leq\hat{b}$ is active at $\boldsymbol{u}=\boldsymbol{u}^*$ *i.e.* $\boldsymbol{a}^T\boldsymbol{u}^*=\hat{b}$

$$\begin{aligned}&\Longrightarrow \boldsymbol{a}^T\left(\hat{\boldsymbol{u}}-\frac{1}{2}\mu^*\boldsymbol{a}\right)=\hat{b}\\ &\Longrightarrow \mu^*=2\frac{\boldsymbol{a}^T\hat{\boldsymbol{u}}-\hat{b}}{\|\boldsymbol{a}\|_2^2}\end{aligned} \tag{32}$$

Case 2: The constraint $\boldsymbol{a}^T\boldsymbol{u}\leq\hat{b}$ is inactive at $\boldsymbol{u}=\boldsymbol{u}^*$ *i.e.* $\boldsymbol{a}^T\boldsymbol{u}^*<\hat{b}$. From complementary slackness, it follows $\mu^*=0$ and hence $\boldsymbol{u}^*=\hat{\boldsymbol{u}}$. However, this contradicts the definition of deadlock. Hence, case 2 can never arise in deadlock.

5.1 Characteristics of two-robot deadlock

We now analyze qualitative properties of the system deadlock set towards synthesizing a controller that will enable the robots to exit this set. We will show that when deadlock occurs, (1) the two robots are separated by the safety distance, (2) deadlock set is non-empty and (3) bounded and of measure zero.

Theorem 1 (Safety Margin Apart). *In deadlock, the two robots are separated by the safety distance and the robots are on the verge of violating safety (see Fig. 1, 3(a))*

Proof. In (32), we proved that the collision avoidance constraint is active in deadlock. Since both robots are in deadlock, we know that both of their collision avoidance constraints are active *i.e.* $\boldsymbol{a}_1^T\boldsymbol{u}_1^*=\hat{b}_{12}$, $\boldsymbol{a}_2^T\boldsymbol{u}_2^*=\hat{b}_{21}$ and $\boldsymbol{u}_1^*=\boldsymbol{0}$ and $\boldsymbol{u}_2^*=\boldsymbol{0}$. This implies $\hat{b}_{12}=\hat{b}_{21}=0$. Using (6) and (7) and that in deadlock, $(\boldsymbol{v}_1,\boldsymbol{v}_2)=(\boldsymbol{0},\boldsymbol{0})$ we get

$$\hat{b}_{12}=\frac{\alpha_1}{\alpha_1+\alpha_2}\|\Delta\boldsymbol{p}_{12}\|h_{12}^3=0\Longrightarrow h_{12}=0 \tag{33}$$

Recall h_{12} from (2) and using that $(\boldsymbol{v}_1,\boldsymbol{v}_2)=(\mathbf{0},\mathbf{0})$, we get

$$h_{12}(\boldsymbol{z}_1,\boldsymbol{z}_2)=\sqrt{2(\alpha_1+\alpha_2)(\|\Delta\boldsymbol{p}_{12}\|-D_s)} \tag{34}$$

Therefore, $h_{12}=0 \iff \|\Delta\boldsymbol{p}_{12}\|=D_s$. Assuming QP is feasible, we disregard $\|\Delta\boldsymbol{p}_{12}\|=0$. Therefore, $\|\Delta\boldsymbol{p}_{12}\|=D_s$. Additionally, recalling the definition from $\partial\mathcal{C}$ from (3) we deduce that, in deadlock, $(\boldsymbol{z}_1,\boldsymbol{z}_2)\in\partial\mathcal{C}$ *i.e.* $\mathcal{D}_{system}\subset\partial\mathcal{C}$. □

This result confirms our intuition, because if the robots are separated by more than the safety distance, then they will have wiggle room to move because they are not at their goals and $\hat{\boldsymbol{u}}\neq\mathbf{0}$. However, the ability to move, albeit with small velocity would contradict the definition of deadlock. We now propose a family of states that are always in the system deadlock set $\mathcal{D}_{system}$.

Theorem 2 ($\mathcal{D}_{system}$ is Non-Empty). $\forall\ k_p, k_v, D_s > 0, \exists$ *a family of states* $(\boldsymbol{z}_1^*,\boldsymbol{z}_2^*)\in\mathcal{D}_{system}$. *These states are such that the robots and their goals are all collinear.*

Proof. To prove this theorem, we propose a set of candidate states $(\boldsymbol{z}_1^*,\boldsymbol{z}_2^*)$ and show that they satisfy the definition of deadlock (31). See Fig. 3(a) for an illustration of geometric quantities referred to in this proof.

Let $\boldsymbol{p}_1^*=\alpha\boldsymbol{p}_{d_1}+(1-\alpha)\boldsymbol{p}_{d_2}$ and $\boldsymbol{p}_2^*=\boldsymbol{p}_1^*-D_s\hat{e}_\beta$ where $\beta=\tan^{-1}(\frac{y_{d_2}-y_{d_1}}{x_{d_2}-x_{d_1}})$ and $\alpha\in(0,1)$. Note that $\boldsymbol{p}_1^*,\boldsymbol{p}_2^*,\boldsymbol{p}_{d_1},\boldsymbol{p}_{d_2}$ are collinear by construction. Let $\boldsymbol{z}_1^*=(\boldsymbol{p}_1^*,\mathbf{0})$ and $\boldsymbol{z}_2^*=(\boldsymbol{p}_2^*,\mathbf{0})$. Then we will show that $\boldsymbol{Z^*}=(\boldsymbol{z}_1^*,\boldsymbol{z}_2^*)\in\mathcal{D}_{system}$. Note that

$$\begin{aligned}\boldsymbol{a}_1&=-(\boldsymbol{p}_1^*-\boldsymbol{p}_2^*)=-D_s\hat{e}_\beta\\ \hat{\boldsymbol{u}}_1&=-k_p(\boldsymbol{p}_1^*-\boldsymbol{p}_{d_1})\end{aligned} \tag{35}$$

From definition, $\hat{e}_\beta=\frac{1}{D_G}(x_{d_2}-x_{d_1},y_{d_2}-y_{d_1})$ where $D_G=\left\|\boldsymbol{p}_{d_2}-\boldsymbol{p}_{d_1}\right\|$ is the distance between the goals. Therefore, we have

$$\begin{aligned}\boldsymbol{p}_1^*-\boldsymbol{p}_{d_1}&=-(1-\alpha)\boldsymbol{p}_{d_1}+(1-\alpha)\boldsymbol{p}_{d_2}\\ &=(1-\alpha)D_G\hat{e}_\beta\end{aligned} \tag{36}$$

Substituting (36) in (35) gives

$$\hat{\boldsymbol{u}}_1=-k_p(1-\alpha)D_G\hat{e}_\beta \tag{37}$$

From (35) and (37), we deduce that Lagrange multiplier μ_1

$$\begin{aligned}\mu_1&=2\frac{\boldsymbol{a}_1^T\hat{\boldsymbol{u}}_1}{\|\boldsymbol{a}_1\|_2^2}=2k_p(1-\alpha)\frac{D_G}{D_s}>0\ \forall\alpha\in(0,1)\\ \implies\frac{1}{2}\mu_1\boldsymbol{a}_1&=-\frac{1}{2}2k_p(1-\alpha)\frac{D_G}{D_s}D_s\hat{e}_\beta=\hat{\boldsymbol{u}}_1\end{aligned} \tag{38}$$

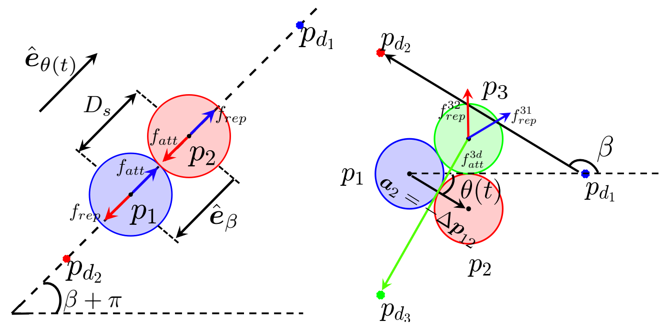
Hence, in (38), we have shown that $\hat{\boldsymbol{u}}_1=\frac{1}{2}\mu_1\boldsymbol{a}_1$ which is one condition in the definition of the deadlock set. Similarly, we can show that $\hat{\boldsymbol{u}}_2=\frac{1}{2}\mu_2\boldsymbol{a}_2$. Also note that in (38) we have shown that the Lagrange multiplier μ_1 is positive, which is another condition in (31). We can similarly show that $\mu_2>0$. Finally, note that for our choice of states, $\boldsymbol{v}_1^*=\boldsymbol{v}_2^*=\mathbf{0}$ and we have restricted $\alpha\in(0,1)$ so we can ensure that $\boldsymbol{p}_i^*\neq\boldsymbol{p}_{d_i}$ Hence, the proposed states are always in deadlock. □

Theorem 3 ($\mathcal{D}_{system}$ is bounded). *The system deadlock set is measure zero.*

Proof. Following the definition of $\boldsymbol{p}_1^*$ and $\boldsymbol{p}_2^*$ from theorem 1 and theorem 2, we can show that when two robots are in deadlock, their positions satisfy

$$\left\|(\boldsymbol{p}_1-\boldsymbol{p}_{d_1})\right\|+\left\|(\boldsymbol{p}_2-\boldsymbol{p}_{d_2})\right\|=D_s+D_G$$

This can be verified by straightforward substitution. From this constraint it is evident, that the deadlock set is not "large", bounded and of measure zero. That is why, random perturbations are one feasible way to resolve deadlock. □



(a) Two Robot Equilibrium (b) Three Robot Equilibrium

Fig. 3: Force Equilibrium in Deadlock

6 Three Robot Deadlock

Following the ideas developed for two robot deadlock, we now describe the three robot case. We will demonstrate that properties such as robots being on the verge of safety violation (theorem 4) and non-emptiness (theorem 5) are retained in this case as well. We are interested in analyzing *system deadlock*, which occurs when $\boldsymbol{u}_i^* = \boldsymbol{0}$, $\boldsymbol{v}_i = \boldsymbol{0}$ and $\hat{\boldsymbol{u}}_i \neq \boldsymbol{0}$ $\forall i \in \{1,2,3\}$. Since we are studying *system deadlock*, each robot will have at-least one active collision avoidance constraint (each robot has two constraints in total). Note that the *system deadlock* set $\mathcal{D}_{system}$ for three robots is defined analogously to (31).

Theorem 4 (Safety Margin Apart). *In system deadlock, either all three robots are separated by the safety margin or exactly two pairs of robots are separated by the safety margin.*

Proof. The proof is kept brief because it is similar to the proof of theorem 1. Based on the number of constraints that are allowed to be active per robot, all geometric configurations can be clubbed in two categories :

Category A - This arises when all collision avoidance constraints of each robot are active *i.e.* $\boldsymbol{a}_{ij}^T \boldsymbol{u}_i^* = \hat{b}_{ij} = 0 \iff \left\| \Delta \boldsymbol{p}_{ij} \right\| = D_s \ \forall j \in \{1,2,3\} \backslash i \ \forall i \in \{1,2,3\}$. As a result, each robot is separated by D_s from every other robot (Fig. 4(a)).

Category B - This arises when there is exactly one robot with both its constraints active (robot i in Fig. 4(b)), and the remaining two robots (j and k) have exactly one constraint active each. Hence, robot i is separated by D_s from the other two. After relabeling of indices, category B results in three rearrangements. □

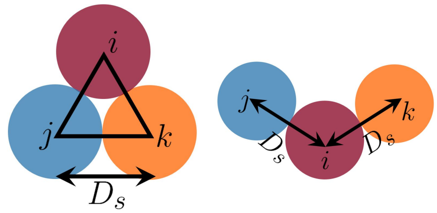
Theorem 5 (Non-emptiness). *$\forall\ k_p, k_v, D_s, R > 0$ and $\boldsymbol{p}_{d_i} = R\hat{\mathbf{e}}_{2\pi(i-1)/3}$ where $i = \{1,2,3\}$, $\exists\ (\boldsymbol{z}_1^*, \boldsymbol{z}_2^*, \boldsymbol{z}_3^*) \in \mathcal{D}_{system}$ where $\boldsymbol{z}_i^* = (\boldsymbol{p}_i^*, \boldsymbol{0})$ and $\boldsymbol{p}_i^*$ is proposed as follows:*

Configuration A: $\boldsymbol{p}_i^* = \frac{D_s}{\sqrt{3}} \hat{\mathbf{e}}_{\frac{2\pi(i-1)}{3}+\pi}$ *where* $i = \{1,2,3\}$

Configuration B: $\boldsymbol{p}_1^* = D_s \hat{\mathbf{e}}_\pi$, $\boldsymbol{p}_2^* = \boldsymbol{0}$, $\boldsymbol{p}_3^* = D_s \hat{\mathbf{e}}_{\frac{\pi}{3}}$ *if robot 2 has both constraints active.*

Proof. This proof is similar to the proof of theorem 2 so it is skipped. Some remarks:

1. In the statement of this theorem, we have predefined the desired goal positions unlike the statement of theorem 2. The candidate positions of the robots that we propose are in $\mathcal{D}_{system}$ are valid with respect to these given goals. We have derived a similar non-emptiness result for arbitrary goals but are not including it here for the sake of brevity.



(a) Category A (b) Category B

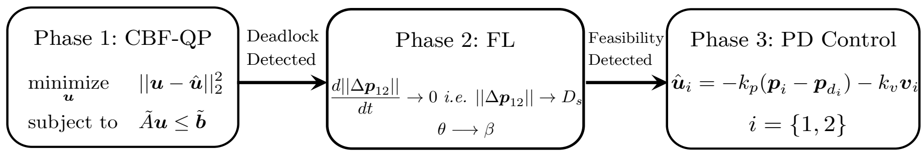
Fig. 4: Categories of geometric configurations in system deadlock of three robots

2. For configuration B, we proposed one set of positions that is valid in deadlock, however there is continuous family of positions that can be valid in configuration B. The representation of this family can be found in the supplementary material. □

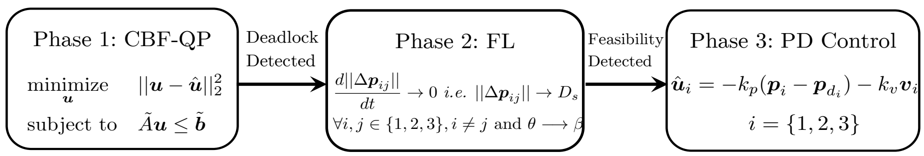
7 Deadlock Resolution

We now use the properties of geometric configurations derived in section 5 and section 6 to synthesize a strategy that (1) gets the robots out of deadlock, (2) ensures their safety and (3) makes them converge to their goals. One approach to achieve these objectives is to detect the incidence of deadlock while the CBF-QP controller is running on the robots and once detected, any small non-zero perturbation to the control will instantaneously give a non-zero velocity to the robots. Thereafter, CBF-QPs can take charge again and we can hope that using this controller the system state will come out of deadlock at-least for a short time. This has two limitations however; firstly, since it was the CBF-QP controller that led to deadlock, there is no guarantee that the system will not fall back in deadlock again. Secondly, perturbations can violate safety and even lead to degraded performance. Therefore, we propose a controller which ensures that goal stabilization, safety and deadlock resolution are met with guarantees. We demonstrate this algorithm for the two and three robot cases. Extension to $N \geq 4$ is left for future work since $N \geq 4$ admits a large number of geometric configurations that are valid in deadlock. Refer to Fig. 5 for a schematic of our approach. This algorithm is described here:

1. The algorithm starts by executing controls derived from CBF-QP in Phase 1. This ensures movement of robots to the goals and safety by construction. To detect the incidence of deadlock, we continuously compare $\|\boldsymbol{u}^*\|, \|\boldsymbol{v}\|, \|\boldsymbol{p}-\boldsymbol{p}_d\|$ against small thresholds. If satisfied, we switch control to phase 2, otherwise, phase 1 continues to operate.
2. In this phase, we rotate the robots around each other to swap positions while maintaining the safe distance.
   (a) For the two-robot case, we calculate $\boldsymbol{u}^1_{fl}(t)$ and $\boldsymbol{u}^2_{fl}(t)$ using feedback linearization (*see supplementary material*) to ensure that $\|\Delta\boldsymbol{p}_{12}\| = D_s$ and rotation ($\ddot{\theta} = -k_p(\theta-\beta) - k_v\dot{\theta} \implies \Delta\boldsymbol{p}_{12}^T\Delta\boldsymbol{v}_{12} = 0$) (See Fig. 3(b) for θ,β). This rotation and distance invariance guarantees safety *i.e.* $h_{12} = 0$. Adding an extra constraint $\boldsymbol{u}^1_{fl} + \boldsymbol{u}^2_{fl} = \boldsymbol{0}$ still ensures that the problem is well posed and additionally makes the centroid static.
   (b) For the three-robot case, we calculate $\boldsymbol{u}^1_{fl}(t), \boldsymbol{u}^3_{fl}(t), \boldsymbol{u}^3_{fl}(t)$ to ensure that $\|\Delta\boldsymbol{p}_{12}\| = \|\Delta\boldsymbol{p}_{23}\| = \|\Delta\boldsymbol{p}_{31}\| = D_s$ and rotation ($\ddot{\theta} = -k_p(\theta-\beta) - k_v\dot{\theta} \implies$



(a) Deadlock resolution for two robots



(b) Deadlock resolution for three robots

Fig. 5: Deadlock Resolution Algorithm Schematic

$\Delta\boldsymbol{p}_{12}^T\Delta\boldsymbol{v}_{12}=0$) (See Fig. 3(b) for θ,β). This guarantees safety *i.e.* $h_{12}=h_{23}=h_{31}=0$. Similarly, we impose $\boldsymbol{u}_{fl}^1+\boldsymbol{u}_{fl}^2+\boldsymbol{u}_{fl}^3=\boldsymbol{0}$ to make the centroid static.

3. Once the robots swap their positions, their new positions will ensure that prescribed PD controllers will be feasible in the future. Thus, after convergence of Phase 2 (which happens in finite time), control switches to Phase 3, which simply uses the prescribed PD controllers. This phase guarantees that the distance between robots is non-decreasing and safety is maintained as we prove in theorem 6.

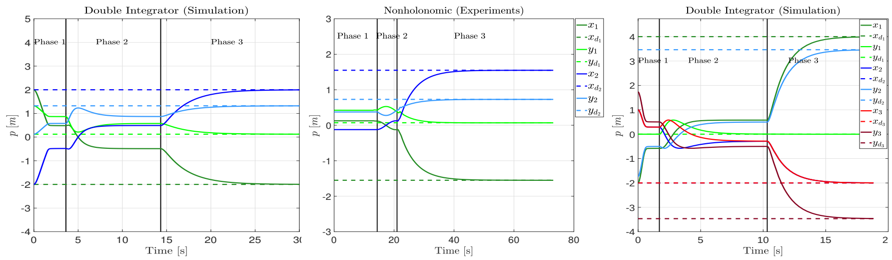
Fig. 6 shows simulation and experimental results from running this strategy on two (6(a)) and three (Fig. 6(c)) robots. Experiments were conducted using Khepera 4 nonholonomic robots (6(b)). Note that for nonholnomic robots, we noticed from experiments and simulations that deadlock only occurs if the body frames of both robots are perfectly aligned with one another at $t=0$. Since this alignment is difficult to establish in experiments, we simulated a virtual deadlock at $t=0$ *i.e.* assumed that the initial position of robots are ones that are in deadlock.

We next prove that this strategy ensures resolution of deadlock and convergence of robots to their goals. The proof of this theorem will exploit the geometric properties of deadlock we derived in theorem 1 and theorem 2. We prove this theorem for $N=2$ since the proof for $N=3$ is a trivial extension of $N=2$.

Theorem 6. *Assuming that PD controllers are overdamped and $D_G > D_s$, this strategy ensures that (1) the robots will never fall in deadlock and (2) converge to their goals.*

Proof. We would like to show that once phase three control begins, the robots will never fall back in deadlock. We will do this by showing that the distance between the robots is non-decreasing, once phase three control starts. We break this proof into three parts consistent with the three phases:

Phase 1 $\rightarrow$ Phase 2: Let $t=t_1$ be the time at which phase 1 ends (and phase 2 starts) *i.e.* when robots fall in deadlock. In theorem 1 we showed that in deadlock $\|\Delta\boldsymbol{p}_{21}\| = D_s$, and in theorem 2 we showed that the positions of robots and their goals are collinear. So at the end of phase 1, $\Delta\boldsymbol{p}_{21}(t_1) = D_s\hat{\boldsymbol{e}}_{\beta+\pi}$. The goal vector



(a) Two Robots Sims. (b) Expmts. with Kheperas. (c) Three Robots Sims.

Fig. 6: Positions of robots from deadlock resolution algorithm. In all figures, final positions converge to desired positions in Phase 3. Videos at https://tinyurl.com/y4ylzwh8

$\Delta \boldsymbol{p}_{d_{21}}(t) \coloneqq \boldsymbol{p}_{d_2} - \boldsymbol{p}_{d_1} = D_G \hat{\boldsymbol{e}}_\beta \ \forall t > 0$. Moreover, since the robots are static in deadlock, $\Delta \boldsymbol{v}_{21}(t_1) = \boldsymbol{0}$

Phase 2 $\longrightarrow$ Phase 3: The initial condition of phase two is the final condition of phase one *i.e.* $\Delta \boldsymbol{p}_{21}(t_1) = D_s \hat{e}_{\beta+\pi}$ and $\Delta \boldsymbol{v}_{21}(t_1) = \boldsymbol{0}$. In phase two, we use feedback linearization to rotate the assembly of robots making sure that the distance between them stays at D_s, until the orientation of the vector $\Delta \boldsymbol{p}_{21}(t) = D_s \hat{\boldsymbol{e}}_{\theta(t)}$ aligns with $\Delta \boldsymbol{p}_{d_{21}} = D_G \hat{\boldsymbol{e}}_\beta$ (*see supplementary for controller derivation*). Once done, $\exists$ a time t_2 at which $\theta(t_2) = \beta$. Moreover, at $t = t_2$, the robots are no longer moving, hence their velocities are zero, hence, $\Delta \boldsymbol{p}_{21}(t_2) = D_s \hat{\boldsymbol{e}}_\beta$, $\Delta \boldsymbol{v}_{21}(t_2) = \boldsymbol{0}$. These states are the final condition for phase 2 and initial for phase 3.

Phase 3 $\longrightarrow \infty$: In this phase, the initial conditions are $\Delta \boldsymbol{p}_{21}(t_2) = D_s \hat{\boldsymbol{e}}_\beta$ and $\Delta \boldsymbol{v}_{21}(t_2) = \boldsymbol{0}$. Also, note that the dynamics of phase 3 control are specified by the prescribed PD controllers. The dynamics of relative positions and velocities are:

$$\begin{aligned} \Delta \dot{\boldsymbol{p}}_{21} &= \Delta \boldsymbol{v}_{21} \\ \Delta \dot{\boldsymbol{v}}_{21} &= -k_p(\Delta \boldsymbol{p}_{21} - \Delta \boldsymbol{p}_{d_{21}}) - k_v \Delta \boldsymbol{v}_{21}, \end{aligned} \tag{39}$$

where $\Delta \boldsymbol{p}_{d_{21}} = D_G \hat{\boldsymbol{e}}_\beta$. Now, we will do a coordinate change as described next. Let $\Delta \tilde{\boldsymbol{p}}_{21} \coloneqq R_{-\beta} \Delta \boldsymbol{p}_{21}$ and $\Delta \tilde{\boldsymbol{v}}_{21} \coloneqq R_{-\beta} \Delta \boldsymbol{v}_{21}$. The initial conditions in these coordinates are $\Delta \tilde{\boldsymbol{p}}_{21}(t_2) = R_{-\beta} D_s \hat{\boldsymbol{e}}_\beta = (D_s, 0)$ and $\Delta \tilde{\boldsymbol{v}}_{21}(t_2) = \boldsymbol{0}$ *i.e.* $\Delta \tilde{p}^x_{21}(t_2) = D_s, \Delta \tilde{p}^y_{21}(t_2) = 0, \Delta \tilde{v}^x_{21}(t_2) = 0$ and $\Delta \tilde{v}^y_{21}(t_2) = 0$. The dynamics in new coordinates are:

$$\begin{aligned} \Delta \dot{\tilde{\boldsymbol{p}}}_{21} &= \Delta \tilde{\boldsymbol{v}}_{21} \\ \Delta \dot{\tilde{\boldsymbol{v}}}_{21} &= -k_p(\Delta \tilde{\boldsymbol{p}}_{21} - R_{-\beta} \Delta \boldsymbol{p}_{d_{21}}) - k_v \Delta \tilde{\boldsymbol{v}}_{21}. \end{aligned} \tag{40}$$

Using these coordinates, note that $R_{-\beta} \Delta \boldsymbol{p}_{d_{21}} = (D_G, 0)$. Note from the dynamics and the initial conditions for the y components of relative position and velocities that the only solution is the zero solution *i.e.* $\Delta \tilde{p}^y_{21}(t) \equiv 0$ and $\Delta \tilde{v}^y_{21}(t) \equiv 0 \ \forall \ t \geq t_2$. As for the x component, we can compute the solution to be $\Delta \tilde{p}^x_{21}(t) = c_1 e^{\omega_1 (t-t_2)} + c_2 e^{\omega_2 (t-t_2)} + D_G$ and $\Delta \tilde{v}^x_{21}(t) = c_1 \omega_1 e^{\omega_1 (t-t_2)} + c_2 \omega_2 e^{\omega_2 (t-t_2)}$. Here

$$\omega_{1,2} = \frac{1}{2}\left(-k_v \pm \sqrt{k_v^2 - 4k_p}\right), \ c_1 = \frac{\omega_2 (D_G - D_s)}{\omega_1 - \omega_2}, \ c_2 = -\frac{\omega_1 (D_G - D_s)}{\omega_1 - \omega_2},$$

and $\omega_1 - \omega_2 = -\sqrt{k_v^2 - 4k_p}$ and $\omega_1 \omega_2 = k_p$. After substituting these values, we get, $\Delta \tilde{v}^x_{21}(t) = \frac{k_p (D_s - D_G)(e^{\omega_1 (t-t_2)} - e^{\omega_2 (t-t_2)})}{\sqrt{k_v^2 - 4k_p}}$. Now, from the assumptions that PD controllers are overdamped *i.e* $k_v, k_v^2 - 4k_p > 0$ and $D_G > D_s$, it follows that $\Delta\tilde{v}_{21}^x(t) = \frac{k_p(D_s - D_G)(e^{\omega_1(t-t_2)} - e^{\omega_2(t-t_2)})}{\sqrt{k_v^2 - 4k_p}} \geq 0$ and $\Delta\tilde{p}_{21}^x(t) = \frac{D_G - D_s}{\sqrt{(k_v^2 - 4k_p)}}\Big(\omega_1 e^{\omega_2(t-t_2)} - \omega_2 e^{\omega_1(t-t_2)}\Big) + D_G \geq D_s \geq 0$. Finally, note that $\frac{d\|\Delta\boldsymbol{p}_{12}(t)\|}{dt} = \frac{\Delta\tilde{\boldsymbol{p}}_{21}^T(t)\Delta\tilde{\boldsymbol{v}}_{21}(t)}{\|\Delta\tilde{\boldsymbol{p}}_{21}(t)\|} \geq 0$. Hence, the distance between the robots is non-decreasing *i.e.* the robots never fall in deadlock. Additionally, since the robots use a PD-type controller, their positions exponentially stabilize to their goals. □

8 Conclusions

In this paper, we analyzed the characteristic properties of deadlock that results from using CBF based QPs for avoidance control in multirobot systems. We demonstrated how to interpret deadlock as a subset of the state space and proved that in deadlock, the robots are on the verge of violating safety. Additionally, we showed that this set is non-empty and bounded. Using these properties, we devised corrective control algorithm to force the robots out of deadlock and ensure task completion. We also demonstrated that the number of valid geometric configurations in deadlock increases approximately exponentially with the number of robots which makes the analysis and resolution for $N \geq 4$ complex. There are several directions we would like to explore in future. Firstly, we want to extend this to $N \geq 4$ case. In the $N = 4$ case, we determined a large number of admissible geometric configurations. We find that there exist bijections among some of these configurations depending on the number of total active constraints. We believe this property can be exploited to reduce the complexity down to the equivalence classes of these bijections. We will exploit this line of approach to simplify analysis for $N \geq 4$ cases. Secondly, we are interested in identifying the basin of attraction of the deadlock set to formally characterize all initial conditions of robots that lead to deadlock. Tools from backwards reachability set calculation can be used to compute the basin of attraction. Finally, although we focused on CBF based QPs for analysis, we will extend this to other reactive methods such as velocity obstacles and tools using value functions, and explore the properties that make a particular algorithm immune to deadlock.

References

1. G. Kantor, S. Singh, R. Peterson, D. Rus, A. Das, V. Kumar, G. Pereira, and J. Spletzer, "Distributed search and rescue with robot and sensor teams," in *Field and Service Robotics*. Springer, 2003, pp. 529–538.
2. J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.
3. W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on robotics*, vol. 21, no. 3, pp. 376–386, 2005.
4. P. Ogren, M. Egerstedt, and X. Hu, "A control lyapunov function approach to multiagent coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 847–851, 2002.
5. R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
6. U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt, "Control barrier certificates for safe swarm behavior," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 68–73, 2015.

7. A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.
8. S. Petti and T. Fraichard, "Safe motion planning in dynamic environments," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 2210–2215.
9. P. O'DONNELL, "Deadlockfree and collision-free coordination of two robot manipulators." in *Proc. IEEE International Conference on Robotics and Automation*, 1989, pp. 484–489.
10. L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.
11. G. Leitmann and J. Skowronski, "Avoidance control," *Journal of optimization theory and applications*, vol. 23, no. 4, pp. 581–591, 1977.
12. ——, "A note on avoidance control," *Optimal Control Applications and Methods*, vol. 4, no. 4, pp. 335–342, 1983.
13. D. M. Stipanović, P. F. Hokayem, M. W. Spong, and D. D. Šiljak, "Cooperative avoidance control for multiagent systems," *Journal of dynamic systems, measurement, and control*, vol. 129, no. 5, pp. 699–707, 2007.
14. P. F. Hokayem, D. M. Stipanović, and M. W. Spong, "Coordination and collision avoidance for lagrangian systems with disturbances," *Applied Mathematics and Computation*, vol. 217, no. 3, pp. 1085–1094, 2010.
15. P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.
16. J. Van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *2008 IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 1928–1935.
17. J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics research*. Springer, 2011, pp. 3–19.
18. D. Wilkie, J. Van Den Berg, and D. Manocha, "Generalized velocity obstacles," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 5573–5578.
19. H. Yamaguchi, "A cooperative hunting behavior by mobile-robot troops," *the International Journal of robotics Research*, vol. 18, no. 9, pp. 931–940, 1999.
20. M. Jager and B. Nebel, "Decentralized collision avoidance, deadlock detection, and deadlock resolution for multiple mobile robots," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, vol. 3. IEEE, 2001, pp. 1213–1219.
21. Y. Li, K. Gupta, and S. Payandeh, "Motion planning of multiple agents in virtual environments using coordination graphs," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. IEEE, 2005, pp. 378–383.
22. E. J. Rodríguez-Seda, D. M. Stipanović, and M. W. Spong, "Guaranteed collision avoidance for autonomous systems with acceleration constraints and sensing uncertainties," *Journal of Optimization Theory and Applications*, vol. 168, no. 3, pp. 1014–1038, 2016.
23. S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
24. H. S. Wilf, *generatingfunctionology*. AK Peters/CRC Press, 2005.
25. E. Weisstein. [Online]. Available: https://oeis.org/A303792
26. N. Alon and A. Kupavskii, "Two notions of unit distance graphs," *Journal of Combinatorial Theory, Series A*, vol. 125, pp. 1–17, 2014.