

Feasible Region-based Identification Using Duality (Extended Version)

Jaskaran Grover¹, Changliu Liu¹, Katia Sycara^{1*}

Abstract

We consider the problem of estimating bounds on parameters representing tasks being performed by individual robots in a multirobot system. In our previous work, we derived necessary conditions based on persistency of excitation analysis for exact identification of these parameters. We concluded that depending on the robot’s task, the dynamics of individual robots may fail to satisfy these conditions, thereby preventing exact inference. As an extension to that work, this paper focuses on estimating bounds on task parameters when such conditions are not satisfied. Each robot in the team uses optimization-based controllers for mediating between task satisfaction and collision avoidance. We use KKT conditions of this optimization and SVD of active collision avoidance constraints to derive explicit relations between Lagrange multipliers, robot dynamics and task parameters. Using these relations, we are able to derive bounds on each robot’s task parameters. Through numerical simulations we show how our proposed region based identification approach generates feasible regions for parameters when a conventional estimator such as a UKF fails. Additionally, empirical evidence shows that this approach generates contracting sets which converge to the true parameters much faster than the rate at which a UKF based estimate converges. Videos of these results are available at <https://bit.ly/2JDMgeJ>

1 Introduction

Task-based motion planning and control for multiple robots has several applications including search and rescue [1], sensor coverage [2] and environmental exploration [3]. The resiliency and robustness that can be achieved by multiple robots is superior to that achievable by one [4], [5]. While controlling these robots can be treated as a forward problem, in this paper we consider the inverse problem of task inference [6], which has several applications of its own. For example, analyzing how easy it is to infer tasks being performed by a team of robots by simply observing them can reveal vulnerabilities of that robot team to adversarial attacks. This analysis can be used to design privacy preserving controllers to make the team resilient to attacks. In this paper, we take the view of an observer monitoring a team of robots in which each robot is performing some task. The question we want to address is how can the observer compute *reasonable bounds* on the parameters representing the task of each robot by measuring their positions over time.

The approach to infer parameters by estimating bounds is different from inferring them using point-wise estimation algorithms such as an Unscented Kalman Filter (UKF) [7] or adaptive observers [8],[9],[10]. These algorithms are point-wise estimators because they generate a single estimate of the parameter that converges to the true parameter. Their convergence, however, is only guaranteed when the system dynamics satisfy the *persistency of excitation* condition [11]. When an observer has the freedom to command inputs to the system, he can design *sufficiently rich* inputs that will ensure convergence of these estimators to the true parameters of the system. This open-loop identification has been used for identifying physical parameters of manipulators [12] and quadrotors [13].

By contrast, our scenario is different from these systems because of two reasons. Firstly, our observer can only observe the robots but not intervene physically. Hence the parameter identification must occur in a closed-loop way. Secondly, while manipulators and quadrotors are monolithic, our “plant” is comprised of multiple robots. In our previous work [14], we derived necessary identifiability conditions for this system and showed that multiple active interactions between robots leads to violation of the persistency of excitation condition. Since interactions between robots are inevitable, point-wise parameter estimation algorithms would fail to converge owing to the violation of this condition.

This takeaway from our previous work forms the motivation for this paper. That is, while it may not be possible to get an exact estimate of the task parameter of a robot, we want to determine if we can estimate any bounds for it. To address this problem, we propose a *feasible-region based parameter identification* algorithm that generates a bounded set where the true task parameter of each robot must belong. Further, we design this algorithm in a way such that the “measure” of this set (*i.e.* its size) is non-increasing with time. There are several advantages of our proposed identification approach over point-wise estimation. Firstly, this algorithm is anytime *i.e.* the set computed at a given time is comprised of parameters that are all valid candidates for explaining the robot measurements observed until that time. Secondly, this algorithm does not have any gains to tune unlike Kalman filters or adaptive observers [10] where tuning relies on user’s experience. Thirdly, this algorithm can also work symbiotically with point-wise estimation algorithms and can expedite their convergence by continually projecting their estimates to the feasible regions generated by this algorithm. Finally, our empirical evidence suggests that even in cases where point-wise estimation is possible, region-based estimation converges much faster than a point-wise estimator (we have used a UKF for comparison).

^{*1}J. Grover, C. Liu, K. Sycara are with the Robotics Institute at Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA. {jaskarag, cliu6, sycara}@andrew.cmu.edu

The outline of this paper is as follows. In section 2, we give a brief review of point-wise parameter estimation following the development in [14, 15] and establish notation for our proposed *feasible region-based* identification approach. In section 3, we discuss how prior work [16] has addressed the multirobot task satisfaction problem using optimization-based controllers. Although prior work has mostly considered the task of making robots reach some goals, we make the formulation abstract to allow for arbitrary tasks. We formalize the *feasible region-based* task identification problem in section 4. The main contributions begin from section 5 where we derive the KKT conditions of the control-synthesis optimization. These conditions allow us to pose an equality constrained optimization problem (EQP) formulated using the set of active interactions (*i.e.* active constraints) of the ego robot. In section 6, we classify this robot’s dynamics based on the number of constraints in this EQP, and linear independence relations amongst these constraints. Taking the SVD of these constraints allows us to derive explicit relations between robot dynamics, Lagrange multipliers and task parameters. Finally, by using these relations along with the KKT conditions, we derive analytical descriptions of the sets where the task parameters must belong as we wanted. We demonstrate the power of this identification approach through numerical simulations in section 7. In these simulations, we consider geometric settings that span all combinations of number of active constraints and linear independence relations we theorized in section 6. For these cases, we show through simulations how the bounds on task parameters computed by our approach converge much faster to the true parameter than an Unscented Kalman Filter. We conclude in 8 by summarizing and provide directions for future work.

2 Approaches for Parameter Identification

We describe two different approaches for parameter identification. These include (a) Point-wise identification and (b) Feasible-region based identification. The need for distinguishing between these two approaches is necessitated by our application *i.e.* task inference for multirobot systems. For these systems, we will demonstrate that approach (a) which is the de-facto standard for parameter identification, can often fail to converge owing to unavoidable interactions between robots. In such circumstances, approach (b) on the other hand, can provide reasonable bounds on task parameters, if not their exact values.

2.1 Point-wise identification

The *point-wise identification* approach focuses on designing an online parameter update law which ensures that the estimate of the parameter converges to the true parameter of the underlying system. Several known estimation algorithms fall in this category, including RLS [17], UKF, EKF [7] and the ones based on adaptive observers [15, 9, 8]. Adaptive observer based estimators provide conditions based on the *persistence of excitation* criterion [18, 19] under which convergence to the true parameters is achieved. To formalize point-wise identification, consider a nonlinear system as follows

$$\dot{\mathbf{x}} = G(\mathbf{x})\boldsymbol{\theta} + f(\mathbf{x}), \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^n$ is the measurable state, $\boldsymbol{\theta} \in \mathbb{R}^p$ is the unknown parameter and $G(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times p}$, $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ are known functions. For example, in our context, $\mathbf{x}(t)$ will correspond to the position of the ego robot under observation and $\boldsymbol{\theta}$ denotes the task parameters of its controller that the observer wishes to infer. We can safely assume that the observer runs several parallel estimators, one for estimating the parameters of each robot, so the observer’s focus is on the ego robot. Adhering to the *pointwise identification* paradigm, the observer’s problem is to design an estimation law $\dot{\hat{\boldsymbol{\theta}}} = \psi(\hat{\boldsymbol{\theta}}, \mathbf{x})$ that guarantees convergence of $\hat{\boldsymbol{\theta}} \rightarrow \boldsymbol{\theta}$ by using $\mathbf{x}(t)$ over some $t \in [0, T]$ where T is large enough.

2.2 Feasible region-based identification

This approach computes a set in the parameter space where the parameter of the underlying system must belong. The observer may use some measurements $\mathbf{y}(t)$ and compute a set $\Theta(t) \subset \mathbb{R}^p$ such that

1. $\Theta(t)$ is bounded.
2. $\boldsymbol{\theta} \in \Theta(t) \forall t$.
3. The “measure” of this set, denoted by $\text{Vol}(\Theta(t))$ is non-increasing with time.

Ideally, if condition (2) is satisfied and condition (3) is replaced with a strict decrease in $\text{Vol}(\Theta(t))$, it is guaranteed that $\lim_{t \rightarrow \infty} \Theta(t) = \boldsymbol{\theta}$. However, the relaxed condition (3) *i.e.* $\text{Vol}(\Theta(t))$ is non-increasing is easier to ensure in practice and will be our focus. One way to define a set that satisfies these three conditions is as follows. Suppose at a given time t , we deduce that $\boldsymbol{\theta} \in \Omega(t)$ where

$$\Omega(t) := \{\hat{\boldsymbol{\theta}} \in \mathbb{R}^p | \mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}}) \prec \mathbf{0}\}, \quad (2)$$

for some function $\mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}})$ that depends on the observer’s measurements $\mathbf{y}(t)$. Note that $\Omega(t)$ need not be bounded, that would depend on how $\mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}})$ is defined. Further, suppose $\Theta_0 \subset \mathbb{R}^p$ is a known time-invariant compact set in which $\boldsymbol{\theta}$ is known to belong a-priori. Then defining

$$\Theta(t) := \bigcap_{0 \leq \tau \leq t} \Omega(\tau) \cap \Theta_0 \quad (3)$$

will ensure satisfaction of the three conditions proposed above. Indeed, let t_1 and t_2 be two time instants such that $t_1 < t_2$, then from (3) it is evident that $\Theta(t_2) \subseteq \Theta(t_1)$. This would in-turn imply that $\text{Vol}(\Theta(t_2)) \leq$

$\text{Vol}(\Theta(t_1))$ or in other words, $\text{Vol}(\Theta(t))$ non-increasing. Since $\Theta(t)$ is derived from the set $\Omega(t)$, it suffices to focus on constructing the *instantaneous feasible set* $\Omega(t)$ *i.e.* deriving $\mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}})$.

In the rest of the paper, our goal will be to design a *feasible-region based identification* method for inferring task parameters of a multirobot system. To construct the *instantaneous feasible set* $\Omega(t)$, we need to define functions $\mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}})$ needed in (2). This requires some assumptions on the controllers that the robots use to perform their tasks (parametrized by $\boldsymbol{\theta}$) and the resulting dynamics of the robots. The most basic of these assumptions is safety *i.e.* we assume that all robots have an underlying collision avoidance mechanism to ensure collision-free motions while they perform their tasks. To mediate between task satisfaction and safety, a common approach is to use reactive optimization-based controllers as has been demonstrated in [16, 20]. Throughout the rest of the paper, we will assume that the robots use such controllers. The identification framework we develop assumes *Control Barrier Function based QPs* (CBF-QPs) as the underlying controllers, just to illustrate the mechanics of constructing $\Omega(t)$. However, our approach can be used to construct this set for any other type of optimization-based controller as well [21].

3 Task Satisfaction and Collision Avoidance

We give a brief summary of CBF-QPs here, referring the reader to [16] for a detailed treatment. Let there be a total of $M + 1$ robots in the system. From the perspective of an ego robot, the remaining M robots are “cooperative obstacles” who share the responsibility of avoiding collisions with the ego robot, while performing their own respective tasks. In the following discussion, the focus is on the ego robot. This robot follows single-integrator dynamics *i.e.*

$$\dot{\mathbf{x}} = \mathbf{u}, \quad (4)$$

where $\mathbf{x} = (p_x, p_y) \in \mathbb{R}^2$ is its position and $\mathbf{u} \in \mathbb{R}^2$ is its velocity (*i.e.* the control input). Assume that this robot can use a nominal controller $\hat{\mathbf{u}}(\mathbf{x}) = C(\mathbf{x})\boldsymbol{\theta} + d(\mathbf{x})$ that guarantees satisfaction of the task when there is no other robot in the system. Here, $\boldsymbol{\theta}$ is the task-parameter the observer wishes to infer, and $C(\mathbf{x}), d(\mathbf{x})$ are appropriate functions. As an example, if the ego robot’s task is to reach a goal at \mathbf{x}_d , the ego robot can use $\hat{\mathbf{u}}(\mathbf{x}) = -k_p(\mathbf{x} - \mathbf{x}_d)$, a proportional controller that guarantees $\mathbf{x} \rightarrow \mathbf{x}_d$. If the observer’s problem is to infer the goal \mathbf{x}_d , then defining $C(\mathbf{x}) = k_p$, $d(\mathbf{x}) = -k_p\mathbf{x}$ and $\boldsymbol{\theta} = \mathbf{x}_d$ gives $C(\mathbf{x})\boldsymbol{\theta} + d(\mathbf{x}) = -k_p(\mathbf{x} - \mathbf{x}_d)$.

Now although $\hat{\mathbf{u}}(\mathbf{x})$ by itself ensures task completion, the robots must also avoid collisions amongst themselves. To that end, the ego robot must maintain a distance of at-least D_s with all the other robots located at $\{\mathbf{x}_j^o\}_{j=1}^M$. That is the their positions $(\mathbf{x}, \mathbf{x}_j^o)$ must satisfy $\|\Delta\mathbf{x}_j\|^2 \geq D_s^2$ where $\Delta\mathbf{x}_j := \mathbf{x} - \mathbf{x}_j^o$. To combine the collision avoidance requirement with the task-satisfaction objective, the ego robot solves a QP that computes a controller closest to the nominal task control $\hat{\mathbf{u}}(\mathbf{x})$ and satisfies M collision avoidance constraints as follows:

$$\begin{aligned} \mathbf{u}^* = \arg \min_{\mathbf{u}} \quad & \|\mathbf{u} - \hat{\mathbf{u}}(\mathbf{x})\|^2 \\ \text{subject to} \quad & A(\mathbf{x})\mathbf{u} \leq \mathbf{b}(\mathbf{x}) \end{aligned} \quad (5)$$

Here $A(\mathbf{x}) \in \mathbb{R}^{M \times 2}$, $\mathbf{b}(\mathbf{x}) \in \mathbb{R}^M$ are defined such that the j^{th} row of A is \mathbf{a}_j^T and the j^{th} element of \mathbf{b} is b_j :

$$\begin{aligned} \mathbf{a}_j^T(\mathbf{x}) &:= -\Delta\mathbf{x}_j^T = -(\mathbf{x} - \mathbf{x}_j^o)^T \\ b_j(\mathbf{x}) &:= \frac{\gamma}{2}(\|\Delta\mathbf{x}_j\|^2 - D_s^2) \quad \forall j \in \{1, 2, \dots, M\} \end{aligned} \quad (6)$$

The ego robot solves this QP at every time step to determine its control \mathbf{u}^* , which ensures collision avoidance while encouraging satisfaction of the task parametrized by $\boldsymbol{\theta}$.

Aside from its position \mathbf{x} , the ego robot’s control \mathbf{u}^* depends on task parameter $\boldsymbol{\theta}$. This dependence is implicitly encoded through the cost function of (5) (recall $\hat{\mathbf{u}}(\mathbf{x}) = C(\mathbf{x})\boldsymbol{\theta} + d(\mathbf{x})$). To highlight this dependence, we denote the control as $\mathbf{u}_{\boldsymbol{\theta}}^*(\mathbf{x})$ where $\boldsymbol{\theta}$ is the unknown parameter the observer aims to identify. Next, we formulate the problem that the observer seeks to solve.

4 Region-based Identification Problem Formulation

The observer’s problem is to identify a region in the parameter space consistent with conditions (1) and (2) of the *feasible-region based identification* approach proposed in Sec. 2. That is, the observer must identify a set $\Theta(t) \subset \mathbb{R}^p$ where $\boldsymbol{\theta}$ must lie $\forall t \in [0, T]$ such that its volume is non-increasing over this duration. Following the analysis in subsec. 2.2, it suffices to estimate the *instantaneous feasible set* $\Omega(t)$ (2) as the 0-sublevel set of a function $\mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}})$. The measurements $\mathbf{y}(t)$ (2) that the observer can use to compute this set include (a) the positions of the ego robot $\mathbf{x}(t)$ and (b) the positions of the “cooperative obstacles” $\mathbf{x}_j^o(t) \forall j \in \{1, 2, \dots, M\}$ over $t \in [0, T]$ *i.e.* $\mathbf{y}(t) = (\mathbf{x}(t), \mathbf{x}_1^o(t), \dots, \mathbf{x}_M^o(t))$. The observer will run $M + 1$ parallel set-estimators to compute $\{\Theta^i(t)\}_{i=1}^{M+1}$ for each robot in the team since each robot has its unique task parameter $\boldsymbol{\theta}_i$.

Focusing on the ego robot, the observer needs to derive functions $\mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}})$ needed in (2) to compute $\Omega(t)$. For that, the observer must know an explicit relation between the dynamics of the ego robot and the parameters $\boldsymbol{\theta}$ of the form $\dot{\mathbf{x}} = G(\mathbf{x})\boldsymbol{\theta} + f(\mathbf{x})$ similar to (1). That is, it must know $G(\mathbf{x})$ and $f(\mathbf{x})$. However, owing to the fact that $\dot{\mathbf{x}} = \mathbf{u}_{\boldsymbol{\theta}}^*(\mathbf{x})$ is optimization-based (5), such explicit relations are not known. It is possible to derive these relations by focusing on the KKT conditions of (5). We derive these conditions in the next section using which the observer will be able to compute the set $\Omega(t)$. Before proceeding, let’s state all the assumptions on the observer’s knowledge:

Assumption 1. The observer can measure both the position $\mathbf{x}(t)$ and velocity (i.e. the control $\mathbf{u}_\theta^*(\mathbf{x}(t))$) of the ego robot.

Assumption 2. The observer knows the form of safety constraints $A(\mathbf{x}), \mathbf{b}(\mathbf{x})$ in (5) and that the cost function is of the form $\|\mathbf{u} - \hat{\mathbf{u}}(\mathbf{x})\|^2$.

Assumption 3. The observer knows the task functions $C(\mathbf{x}), d(\mathbf{x})$ of $\hat{\mathbf{u}}(\mathbf{x}) = C(\mathbf{x})\boldsymbol{\theta} + d(\mathbf{x})$

Assumption 1 is not restrictive because positions are easily measurable and velocities can be obtained through numerical differentiation of positions. Assumption 2 is needed since we are interested in deriving expressions of $\mathbf{g}(\mathbf{y}(t), \hat{\boldsymbol{\theta}})$ and that requires knowledge of the underlying control synthesis approach. Assumption 3 is not restrictive in practice because the observer can hypothesize on the high-level task by observing the robots and query a library of task-to-function mappings for guessing $C(\mathbf{x}), d(\mathbf{x})$.

5 Analysis using KKT conditions

To analyze the relation between the optimizer of (5) i.e. $\mathbf{u}_\theta^*(\mathbf{x})$ and parameters $\boldsymbol{\theta}$, we look at the KKT conditions of this QP. These are necessary and sufficient conditions satisfied by $\mathbf{u}_\theta^*(\mathbf{x})$. The Lagrangian for (5) is

$$L(\mathbf{u}, \boldsymbol{\mu}) = \|\mathbf{u} - \hat{\mathbf{u}}\|_2^2 + \boldsymbol{\mu}^T (A\mathbf{u} - \mathbf{b}).$$

Let $(\mathbf{u}_\theta^*, \boldsymbol{\mu}_\theta^*)$ be the optimal primal-dual solution to (5). The KKT conditions are [22]:

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

$$\begin{aligned} \implies \mathbf{u}_\theta^* &= \hat{\mathbf{u}} - \frac{1}{2} \sum_{j \in \{1, \dots, M\}} \mu_{j\theta}^* \mathbf{a}_j \\ &= \hat{\mathbf{u}} - \frac{1}{2} A^T \boldsymbol{\mu}_\theta^*. \end{aligned} \quad (7)$$

2. Primal Feasibility

$$A\mathbf{u}_\theta^* \leq \mathbf{b} \iff \mathbf{a}_j^T \mathbf{u}_\theta^* \leq b_j \quad \forall j \in \{1, \dots, M\}. \quad (8)$$

3. Dual Feasibility

$$\mu_{j\theta}^* \geq 0 \quad \forall j \in \{1, 2, \dots, M\}. \quad (9)$$

4. Complementary Slackness

$$\mu_{j\theta}^* \cdot (\mathbf{a}_j^T \mathbf{u}_\theta^* - b_j) = 0 \quad \forall j \in \{1, 2, \dots, M\}. \quad (10)$$

We define the set of active and inactive constraints as

$$\mathcal{A}(\mathbf{u}_\theta^*) := \{j \in \{1, 2, \dots, M\} \mid \mathbf{a}_j^T \mathbf{u}_\theta^* = b_j\}, \quad (11)$$

$$\mathcal{IA}(\mathbf{u}_\theta^*) := \{j \in \{1, 2, \dots, M\} \mid \mathbf{a}_j^T \mathbf{u}_\theta^* < b_j\}. \quad (12)$$

The set of active constraints qualitatively represents those robots that the ego robot ‘‘worries’’ about for collisions. From the perspective of the ego robot, we simply refer to the ‘‘other robots’’ as *obstacles*.

5.1 Using KKT conditions for defining the feasible region:

Now we briefly describe how these conditions are useful for the computing the *instantaneous feasible set* where the parameter $\boldsymbol{\theta}$ can lie i.e. the set $\Omega(t)$ defined in (2).

1. **Inactive constraints:** From the definition of inactive constraints (11), recall $\mathbf{a}_j^T \mathbf{u}_\theta^* < b_j \quad \forall j \in \mathcal{IA}(\mathbf{u}_\theta^*)$. Thus, deriving an explicit expression for $\mathbf{u}_\theta^* = G(\mathbf{x})\boldsymbol{\theta} + f(\mathbf{x})$ will allow us to prune regions in the parameter where $\boldsymbol{\theta}$ belongs.
2. **Non-negativity of Lagrange multipliers:** Likewise from dual feasibility (9), recall that $\mu_{j\theta}^* \geq 0 \quad \forall j \in \{1, 2, \dots, M\}$. Since $\mu_{j\theta}^*$ depends on $\boldsymbol{\theta}$, deriving an explicit expression for $\mu_{j\theta}^*$ as a function of $\boldsymbol{\theta}$ will allow us to further prune the region where $\boldsymbol{\theta}$ belongs.

Thus we define the *instantaneous feasible set* $\Omega(t)$ as

$$\Omega(t) := \{\hat{\boldsymbol{\theta}} \in \mathbb{R}^p \mid \mathbf{a}_j^T(t) \mathbf{u}_\theta^*(t) - b_j(t) < 0 \quad \forall j \in \mathcal{IA}(\mathbf{u}_\theta^*(t)), -\mu_{j\theta}^*(t) \leq 0 \quad \forall j \in \{1, \dots, M\}\} \quad (13)$$

Table 1: Enumerating cases where either point-wise identification or region-based identification or both are possible

Case	Number of active constraints	$\text{rank}(A_{ac})$	Point-wise identification	Region-based identification
A	None	-	Possible	Possible
B	1	1	Possible	Possible
C	≥ 2	1	Possible	Possible
D	2	2	Not possible	Possible
E	≥ 3	2	Not possible	Possible

5.2 Using KKT conditions for deriving \mathbf{u}_θ^* and $\mu_{j\theta}^*$

The following discussion summarizes how we can simplify problem (5) to a simpler problem so as to derive explicit expressions for both \mathbf{u}_θ^* and $\mu_{j\theta}^*$. Let there be a total of K active constraints *i.e.* $\text{card}(\mathcal{A}(\mathbf{u}_\theta^*)) = K$ where $K \in \{0, 1, \dots, M\}$. Using (9) and (10), we deduce

$$\mu_{j\theta}^* = 0 \quad \forall j \in \mathcal{I}\mathcal{A}(\mathbf{u}_\theta^*). \quad (14)$$

Therefore, we can restrict the summation in (7) only to the set of active constraints *i.e.*

$$\begin{aligned} \mathbf{u}_\theta^* &= \hat{\mathbf{u}} - \frac{1}{2} \sum_{j \in \mathcal{A}(\mathbf{u}_\theta^*)} \mu_{j\theta}^* \mathbf{a}_j \\ &= \hat{\mathbf{u}} - \frac{1}{2} A_{ac}^T \boldsymbol{\mu}_\theta^{ac}. \end{aligned} \quad (15)$$

where $A_{ac}(\mathbf{x}) \in \mathbb{R}^{K \times 2}$ is the matrix formed using the rows of A that are indexed by the active set $\mathcal{A}(\mathbf{u}_\theta^*)$, and likewise $\boldsymbol{\mu}_\theta^{ac} := \{\mu_{j\theta}^*\}_{j \in \mathcal{A}(\mathbf{u}_\theta^*)}$. Similarly, let $\mathbf{b}_{ac}(\mathbf{x}) \in \mathbb{R}^K$ denote the vector formed from the elements of \mathbf{b} indexed by $\mathcal{A}(\mathbf{u}_\theta^*)$. Likewise, we can define $A_{inac}(\mathbf{x})$ and $\mathbf{b}_{inac}(\mathbf{x})$ corresponding to the inactive set. By deleting all inactive constraints and retaining only the active constraints from (5), we can pose another QP that consists only of active constraints, whose solution is the same as that of (5). This equality-constrained program (EQP) is given by

$$\begin{aligned} \mathbf{u}^* &= \arg \min_{\mathbf{u}} \quad \|\mathbf{u} - \hat{\mathbf{u}}(\mathbf{x})\|^2 \\ \text{subject to} \quad & A_{ac}(\mathbf{x})\mathbf{u} = \mathbf{b}_{ac}(\mathbf{x}) \end{aligned} \quad (16)$$

The system $A_{ac}(\mathbf{x})\mathbf{u} = \mathbf{b}_{ac}(\mathbf{x})$ is always consistent by construction (11), as long as a solution \mathbf{u}_θ^* to (5) exists. This EQP is useful because it is easier to derive an expression $\mathbf{u}_\theta^*(\mathbf{x}) = G(\mathbf{x})\boldsymbol{\theta} + f(\mathbf{x})$ for (16) than the inequality constrained problem (5). The question is how can the observer estimate the active set $\mathcal{A}(\mathbf{u}_\theta^*)$ to determine $A_{ac}(\mathbf{x}), \mathbf{b}_{ac}(\mathbf{x})$ for (16). This can be done as follows. From Assumption 1, recall that the observer can measure both the position $\mathbf{x}(t)$ and velocity $\mathbf{u}_\theta^*(\mathbf{x}(t))$ of the robot. Using these, the observer can determine $\mathcal{A}(\mathbf{u}_\theta^*)$ by comparing residuals $|\mathbf{a}_j^T(\mathbf{x})\mathbf{u}_\theta^* - b_j(\mathbf{x})|$ against a small threshold $\epsilon > 0$ consistent with (11):

$$\mathcal{A}^{obs.} := \{j \in \{1, \dots, M\} \mid |\mathbf{a}_j^T(\mathbf{x})\mathbf{u}_\theta^* - b_j(\mathbf{x})| < \epsilon\} \quad (17)$$

For a small threshold ϵ , it holds true that $\mathcal{A}^{obs.} = \mathcal{A}(\mathbf{u}_\theta^*)$ consistent with (11). This allows the observer to determine the active set. Next, we work with (16) to derive an explicit expression for control *i.e.* $\mathbf{u}_\theta^*(\mathbf{x}) = G(\mathbf{x})\boldsymbol{\theta} + f(\mathbf{x})$ and $\mu_{j\theta}^*$ for various combinations of $\text{card}(\mathcal{A}(\mathbf{u}_\theta^*)) = K$ and $\text{rank}(A_{ac}(\mathbf{x}))$.

6 Computing $\Omega(t)$ using SVD of $A_{ac}(\mathbf{x})\mathbf{u} = \mathbf{b}_{ac}(\mathbf{x})$

The aim of this section is to derive explicit expressions for the optimal control \mathbf{u}_θ^* and Lagrange multipliers $\mu_{j\theta}^*$ for region based estimation of $\boldsymbol{\theta}$. To derive these, we will bank on the EQP proposed in (16) as a surrogate to the original problem in (5). We can further simplify the analysis of (16) by considering different cases that can arise based on the number of active constraints (K) and linear independence relations among the constraints governed by $\text{rank}(A_{ac}(\mathbf{x}))$. Table 1 summarizes these cases along with whether point-wise estimation and feasible region-based estimation is possible when either of these arise in practice. Note that the last two rows where the number of active constraints are ≥ 2 correspond to cases where region-based identification is the only method to infer bounds on the $\boldsymbol{\theta}$. As we will show, these in fact correspond to situations that are most likely to arise in multirobot systems applications as well, highlighting the importance of our proposed technique.

6.1 No active constraints *i.e.* $K = 0$

When no constraint is active, we have $\mu_{j\theta} = 0 \quad \forall j \in \{1, 2, \dots, M\}$. This means that $\mathcal{A}(\mathbf{u}_\theta^*) = \emptyset$ and $\mathcal{I}\mathcal{A}(\mathbf{u}_\theta^*) = \{1, 2, \dots, M\}$. In other words, the ego robot does not “worry” about collisions with any other robot. From (7) we get $\mathbf{u}_\theta^* = \hat{\mathbf{u}}(\mathbf{x}) = C(\mathbf{x})\boldsymbol{\theta} + \mathbf{d}(\mathbf{x})$. From this expression, it is evident that the control \mathbf{u}_θ^* and the robot dynamics $\dot{\mathbf{x}}$, exhibit a well-defined dependence on $\boldsymbol{\theta}$. Therefore, it is possible to do point-wise identification of $\boldsymbol{\theta}$ using RLS, UKF or AO type of estimators (provided that $C(\mathbf{x})$ is persistently exciting [19]). However, since point-wise identification is not under the purview of this paper, we focus on estimating regions where $\boldsymbol{\theta}$ can lie. Referring to 5.1, note that the condition for nonnegativity of Lagrange multipliers is

satisfied trivially because $\mu_{j\theta} = 0 \forall j \in \{1, 2, \dots, M\}$, so they do not give any information about θ . However, the condition for inactive constraints gives $\forall j \in \mathcal{IA}(\mathbf{u}_\theta^*) = \{1, 2, \dots, M\}$

$$\begin{aligned} & \mathbf{a}_j^T \mathbf{u}_\theta^* - b_j < 0 \\ \implies & \mathbf{a}_j^T (C\theta + \mathbf{d}) - b_j < 0 \\ \implies & (\mathbf{a}_j^T C)\theta < b_j - \mathbf{a}_j^T \mathbf{d} \end{aligned} \quad (18)$$

which represents a set of M halfspace constraints on θ . Recall from (6) that \mathbf{a}_j and b_j depend on the position of the ego robot *i.e.* \mathbf{x} and the positions of other robots *i.e.* $\mathbf{x}_j^o \forall j \in \{1, 2, \dots, M\}$. In (18), we have omitted these dependencies to keep the notation light. Thus, the desired set Ω is defined using (6) and (13) as follows

$$\Omega := \{\hat{\theta} \in \mathbb{R}^p | (AC)\hat{\theta} < \mathbf{b} - A\mathbf{d}\} \quad (19)$$

6.2 Exactly one active constraint *i.e.* $K = 1$

When exactly constraint is active *i.e.* $K = 1$, there is one obstacle that the ego robot “worries” about for collision. Since there are two degrees of freedom in the control input, and one obstacle to avoid, the ego robot can avoid this obstacle and additionally minimize $\|\mathbf{u} - \hat{\mathbf{u}}(\mathbf{x})\|^2$ with the remaining degree of freedom. This causes \mathbf{u}_θ^* to exhibit a well-defined dependence on $\hat{\mathbf{u}}(\mathbf{x})$ and by extension, on θ . This allows for point-wise identification of θ .

Region-based identification is also possible and can be used to expedite the convergence of point-wise identification. To derive the feasible region where θ belongs, we need the expression for control \mathbf{u}_θ^* and the Lagrange multiplier corresponding to the active constraint. Let $i \in \{1, 2, \dots, M\}$ denote the index of the active constraint. Thus, from (11), we have $A_{ac}(\mathbf{x})\mathbf{u}_\theta^* = \mathbf{a}_i^T(\mathbf{x})\mathbf{u}_\theta^* = b_i(\mathbf{x})$ where $A_{ac}(\mathbf{x}) := \mathbf{a}_i^T(\mathbf{x})$ and $\mathbf{a}_i^T(\mathbf{x}), b_i(\mathbf{x})$ are defined in (6). We use the SVD of $A_{ac} = U\Sigma V^T$. Defining

$$\begin{aligned} U &= 1 \\ \Sigma &= [\Sigma_m, 0] \text{ where } \Sigma_m = \|\mathbf{a}_i\| \\ V &= [V_1, V_2] \quad V_1 = \frac{\mathbf{a}_i}{\|\mathbf{a}_i\|}, V_2 = R_{\frac{\pi}{2}} \frac{\mathbf{a}_i}{\|\mathbf{a}_i\|}, \end{aligned} \quad (20)$$

Since V forms a basis for \mathbb{R}^2 , any \mathbf{u} can be expressed as

$$\begin{aligned} \mathbf{u} &= [V_1, V_2] \begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{bmatrix}. \\ \implies \mathbf{a}_i^T \mathbf{u} - b_i &= U [\Sigma_m, 0] \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} [V_1, V_2] \begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{bmatrix} - b_i \\ &= U\Sigma_m \tilde{u}_1 + 0 \cdot \tilde{u}_2 - b_i = 0 \end{aligned} \quad (21)$$

Choosing $\tilde{u}_1 = \Sigma_m^{-1} U^T b_i$ and $\tilde{u}_2 = \psi \in \mathbb{R}$, we find that

$$\mathbf{u} = V_1 \Sigma_m^{-1} U^T b_i + V_2 \psi \quad (22)$$

satisfies $\mathbf{a}_i^T(\mathbf{x})\mathbf{u} = b_i(\mathbf{x}) \forall \psi \in \mathbb{R}$. Recall from the properties of SVD that V_2 forms a basis for $\mathcal{N}(\mathbf{a}_i^T(\mathbf{x}))$. We tune ψ to minimize $\|\mathbf{u} - \hat{\mathbf{u}}\|^2$ by solving the following unconstrained minimization problem

$$\begin{aligned} \psi^* &= \arg \min_{\psi} \|\mathbf{u} - \hat{\mathbf{u}}\|^2 \\ &= \arg \min_{\psi} \|V_1 \Sigma_m^{-1} U^T b_i + V_2 \psi - \hat{\mathbf{u}}\|^2, \end{aligned} \quad (23)$$

which gives $\psi^* = V_2^T \hat{\mathbf{u}}$. Substituting this in (22), gives

$$\mathbf{u}_\theta^* = V_1 \Sigma_m^{-1} U^T b_i + V_2 V_2^T \hat{\mathbf{u}}. \quad (24)$$

This equation is the solution to (16) and by extension, to (5). Substituting $\hat{\mathbf{u}}(\mathbf{x}) = C(\mathbf{x})\theta + \mathbf{d}(\mathbf{x})$ in (24) gives

$$\begin{aligned} \mathbf{u}_\theta^* &= V_1 \Sigma_m^{-1} U^T b_i + V_2 V_2^T (C\theta + \mathbf{d}) \\ &= \underbrace{V_2 V_2^T C}_{\mathbf{G}} \theta + \underbrace{V_1 \Sigma_m^{-1} U^T b_i + V_2 V_2^T \mathbf{d}}_{\mathbf{f}} \\ &= \mathbf{G}\theta + \mathbf{f} \end{aligned} \quad (25)$$

From inactive constraints, we get $\forall j \in \mathcal{IA}(\mathbf{u}_\theta^*)$:

$$\begin{aligned} & \mathbf{a}_j^T \mathbf{u}_\theta^* - b_j < 0 \\ \implies & \mathbf{a}_j^T (\mathbf{G}\theta + \mathbf{f}) - b_j < 0 \\ \implies & (\mathbf{a}_j^T \mathbf{G})\theta < b_j - \mathbf{a}_j^T \mathbf{f} \\ \implies & (A_{inac} \mathbf{G})\theta < \mathbf{b}_{inac} - A_{inac} \mathbf{f}, \end{aligned} \quad (26)$$

where $A_{inac}, \mathbf{b}_{inac}$ are the rows of A, \mathbf{b} indexed by the inactive constraints $\mathcal{IA}(\mathbf{u}_\theta^*)$. To get the Lagrange multiplier $\mu_{i_\theta}^*$, we use (15) and (25),

$$\begin{aligned} \mathbf{u}_\theta^* &= \hat{\mathbf{u}} - \frac{1}{2} A_{ac}^T \mu_{i_\theta}^* \\ \implies \frac{1}{2} A_{ac}^T \mu_{i_\theta}^* &= \underbrace{(C - G)}_{\tilde{G}} \boldsymbol{\theta} + \underbrace{(\mathbf{d} - \mathbf{f})}_{\tilde{\mathbf{f}}} \\ &= \tilde{G} \boldsymbol{\theta} + \tilde{\mathbf{f}} \\ \implies \mu_{i_\theta}^* &= 2 \frac{A_{ac}(\tilde{G} \boldsymbol{\theta} + \tilde{\mathbf{f}})}{\|A_{ac}^T\|^2} \end{aligned} \quad (27)$$

From non-negativity of $\mu_{i_\theta}^*$ we get

$$\mu_{i_\theta}^* \geq 0 \iff -A_{ac} \tilde{G} \boldsymbol{\theta} \leq A_{ac} \tilde{\mathbf{f}} \quad (28)$$

Thus the desired set Ω where $\boldsymbol{\theta}$ belongs is defined using (26) and (28) as follows

$$\Omega := \{\hat{\boldsymbol{\theta}} \in \mathbb{R}^p \mid (A_{inac} G) \hat{\boldsymbol{\theta}} \prec \mathbf{b}_{inac} - A_{inac} \mathbf{f}, -A_{ac} \tilde{G} \hat{\boldsymbol{\theta}} \leq A_{ac} \tilde{\mathbf{f}}\} \quad (29)$$

6.3 $2 \leq K \leq M$ and $\text{rank}(A_{ac}(\mathbf{x})) = 1$

Let's consider the more general case in which there is more than one constraint active, but all these constraints are linearly dependent on one constraint among them. That is to say there is effectively only one "representative constraint". Consequently, this case is similar to the case with just one active obstacle. We now derive expressions for both \mathbf{u}_θ^* and $\boldsymbol{\mu}_\theta^{ac}$. Let $i_1, i_2, \dots, i_K \in \{1, \dots, M\}$ be the indices of the K active constraints which satisfy

$$\begin{aligned} \begin{bmatrix} \mathbf{a}_{i_1}^T(\mathbf{x}) \\ \mathbf{a}_{i_2}^T(\mathbf{x}) \\ \vdots \\ \mathbf{a}_{i_K}^T(\mathbf{x}) \end{bmatrix} \mathbf{u}_\theta^* &= \begin{bmatrix} b_{i_1}(\mathbf{x}) \\ b_{i_2}(\mathbf{x}) \\ \vdots \\ b_{i_K}(\mathbf{x}) \end{bmatrix} \quad \text{or} \\ A_{ac}(\mathbf{x}) \mathbf{u}_\theta^* &= \mathbf{b}_{ac}(\mathbf{x}), \end{aligned} \quad (30)$$

where $\mathbf{a}_{i_j}^T(\mathbf{x})$ and $b_{i_j}(\mathbf{x})$ are defined using (6). Using the SVD of A_{ac} (30), we can derive an expression for $\mathbf{u}_\theta^*(\mathbf{x})$. Let $A_{ac} = U \Sigma V^T$ where

$$\begin{aligned} U &:= [U_1, U_2] \quad \text{where } U_1 \in \mathbb{R}^{K \times 1}, U_2 \in \mathbb{R}^{K \times K-1} \\ \Sigma &:= \left[\begin{array}{c|c} \Sigma_r & 0^{1 \times 1} \\ \hline 0^{K-1 \times 1} & 0^{K-1 \times 1} \end{array} \right] \\ V &:= [V_1, V_2] \quad \text{where } V_1, V_2 \in \mathbb{R}^2. \end{aligned} \quad (31)$$

Choosing $\mathbf{u} = V_1 \tilde{u}_1 + V_2 \tilde{u}_2$, from (30) we get

$$\begin{aligned} A_{ac} \mathbf{u} - \mathbf{b}_{ac} &= [U_1, U_2] \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} [V_1, V_2] \begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{bmatrix} - \mathbf{b}_{ac} \\ &= [U_1, U_2] \left(\begin{bmatrix} \Sigma_r \tilde{u}_1 \\ 0 \end{bmatrix} - \begin{bmatrix} U_1^T \mathbf{b}_{ac} \\ U_2^T \mathbf{b}_{ac} \end{bmatrix} \right). \end{aligned} \quad (32)$$

Since $\|\cdot\|^2$ is unitary invariant, from (30)

$$\begin{aligned} \|A_{ac} \mathbf{u} - \mathbf{b}_{ac}\|^2 &= \left\| \begin{bmatrix} \Sigma_r \tilde{u}_1 \\ 0 \end{bmatrix} - \begin{bmatrix} U_1^T \mathbf{b}_{ac} \\ U_2^T \mathbf{b}_{ac} \end{bmatrix} \right\|^2 \\ &= \|\Sigma_r \tilde{u}_1 - U_1^T \mathbf{b}_{ac}\|^2 + \|U_2^T \mathbf{b}_{ac}\|^2. \end{aligned} \quad (33)$$

The minimum norm is achieved for $\tilde{u}_1 = \Sigma_r^{-1} U_1^T \mathbf{b}_{ac}$. Choosing $\tilde{u}_2 = \psi \in \mathbb{R}$, the "least-squares" solutions are

$$\mathbf{u} = V_1 \Sigma_r^{-1} U_1^T \mathbf{b}_{ac} + V_2 \psi. \quad (34)$$

Computing ψ by minimizing $\|\mathbf{u} - \hat{\mathbf{u}}\|^2$, we get $\psi^* = V_2^T \hat{\mathbf{u}}$ which gives

$$\mathbf{u}_\theta^* = V_1 \Sigma_r^{-1} U_1^T \mathbf{b}_{ac} + V_2 V_2^T \hat{\mathbf{u}}. \quad (35)$$

This control is the solution to (16) and by extension, to (5). Substituting $\hat{\mathbf{u}}(\mathbf{x}) = C(\mathbf{x}) \boldsymbol{\theta} + \mathbf{d}(\mathbf{x})$ in (35) gives

$$\begin{aligned} \mathbf{u}_\theta^* &= V_1 \Sigma_r^{-1} U_1^T \mathbf{b}_{ac} + V_2 V_2^T (C \boldsymbol{\theta} + \mathbf{d}) \\ &= \underbrace{V_2 V_2^T C}_{\tilde{G}} \boldsymbol{\theta} + \underbrace{V_1 \Sigma_r^{-1} U_1^T \mathbf{b}_{ac} + V_2 V_2^T \mathbf{d}}_{\tilde{\mathbf{f}}} \\ &= \tilde{G} \boldsymbol{\theta} + \tilde{\mathbf{f}} \end{aligned} \quad (36)$$

Following the approach in (26), for the inactive constraints, we get:

$$(A_{inac} G) \boldsymbol{\theta} \prec \mathbf{b}_{inac} - A_{inac} \mathbf{f} \quad (37)$$

To get the Lagrange multipliers $\boldsymbol{\mu}_\theta^{ac}$, we use (15) and (36),

$$\begin{aligned}
\mathbf{u}_\theta^* &= \hat{\mathbf{u}} - \frac{1}{2} A_{ac}^T \boldsymbol{\mu}_\theta^{ac} \\
\implies \frac{1}{2} A_{ac}^T \boldsymbol{\mu}_\theta^{ac} &= (C - G)\boldsymbol{\theta} + (\mathbf{d} - \mathbf{f}) \\
&= \tilde{G}\boldsymbol{\theta} + \tilde{\mathbf{f}} \\
\implies A_{ac}^T \boldsymbol{\mu}_\theta^{ac} &= 2(\tilde{G}\boldsymbol{\theta} + \tilde{\mathbf{f}}) \\
\implies A_{ac}^T \boldsymbol{\mu}_\theta^{ac} &= \mathbf{w}
\end{aligned} \tag{38}$$

We will call the right handside $2(\tilde{G}\boldsymbol{\theta} + \tilde{\mathbf{f}}) = \mathbf{w}$ for convenience of notation. Here, given the fact that $A_{ac}^T \in \mathbb{R}^{2 \times K}$ and $\text{rank}(A_{ac}^T) = 1$, the Lagrange multipliers are underdetermined. We use the SVD of $A_{ac}^T = \tilde{U}\tilde{\Sigma}\tilde{V}^T$ to deduce an expression for $\boldsymbol{\mu}_\theta^{ac}$. Here

$$\begin{aligned}
\tilde{U} &= [\tilde{U}_1, \tilde{U}_2] \text{ where, } \tilde{U}_1, \tilde{U}_2 \in \mathbb{R}^2 \\
\tilde{\Sigma} &= \begin{bmatrix} \tilde{\Sigma}_r & | & 0^{1 \times K-1} \\ \hline 0^{1 \times 1} & | & 0^{1 \times K-1} \end{bmatrix} \\
\tilde{V} &= [\tilde{V}_1, \tilde{V}_2] \text{ where, } \tilde{V}_1 \in \mathbb{R}^{K \times 1}, \tilde{V}_2 \in \mathbb{R}^{K \times K-1}.
\end{aligned} \tag{39}$$

As before we represent, $\boldsymbol{\mu} = \tilde{V}_1 \tilde{\boldsymbol{\mu}}_1 + \tilde{V}_2 \tilde{\boldsymbol{\mu}}_2$. Since $\|\cdot\|^2$ is unitary invariant, from (38)

$$\begin{aligned}
\|A_{ac}^T \boldsymbol{\mu} - \mathbf{w}\|^2 &= \left\| \begin{bmatrix} \Sigma_r \tilde{\boldsymbol{\mu}}_1 \\ 0 \end{bmatrix} - \begin{bmatrix} \tilde{U}_1^T \mathbf{w} \\ \tilde{U}_2^T \mathbf{w} \end{bmatrix} \right\|^2 \\
&= \left\| \Sigma_r \tilde{\boldsymbol{\mu}}_1 - \tilde{U}_1^T \mathbf{w} \right\|^2 + \left\| \tilde{U}_2^T \mathbf{w} \right\|^2.
\end{aligned} \tag{40}$$

The minimum norm is achieved for $\tilde{\boldsymbol{\mu}}_1 = \Sigma_r^{-1} \tilde{U}_1^T \mathbf{w}$. Choosing $\tilde{\boldsymbol{\mu}}_2 = \boldsymbol{\eta} \in \mathbb{R}$, the ‘‘least-squares’’ solutions are

$$\boldsymbol{\mu} = V_1 \Sigma_r^{-1} \tilde{U}_1^T \mathbf{w} + V_2 \boldsymbol{\eta}. \tag{41}$$

Then using (38), we get

$$\boldsymbol{\mu}_\theta^{ac} = 2\tilde{V}_1 \tilde{\Sigma}_r^{-1} \tilde{U}_1^T (\tilde{G}\boldsymbol{\theta} + \tilde{\mathbf{f}}) + \tilde{V}_2 \boldsymbol{\eta}, \tag{42}$$

where $\boldsymbol{\eta} \in \mathbb{R}^{K-1}$ are floating variables which can assume any value. From non-negativity of $\boldsymbol{\mu}_\theta^{ac}$ we get

$$\boldsymbol{\mu}_\theta^{ac} \succeq \mathbf{0} \iff -2\tilde{V}_1 \tilde{\Sigma}_r^{-1} \tilde{U}_1^T \tilde{G}\boldsymbol{\theta} - \tilde{V}_2 \boldsymbol{\eta} \preceq 2\tilde{V}_1 \tilde{\Sigma}_r^{-1} \tilde{U}_1^T \tilde{\mathbf{f}} \tag{43}$$

Thus, the desired set $\boldsymbol{\Omega}$ where $\boldsymbol{\theta}$ belongs is defined using (37) and (43) as follows

$$\begin{aligned}
\boldsymbol{\Omega} := \{ \hat{\boldsymbol{\theta}} \in \mathbb{R}^p \mid & (A_{inac} G) \hat{\boldsymbol{\theta}} < \mathbf{b}_{inac} - A_{inac} \mathbf{f}, \\
& -2\tilde{V}_1 \tilde{\Sigma}_r^{-1} \tilde{U}_1^T \tilde{G} \hat{\boldsymbol{\theta}} - \tilde{V}_2 \boldsymbol{\eta} \preceq 2\tilde{V}_1 \tilde{\Sigma}_r^{-1} \tilde{U}_1^T \tilde{\mathbf{f}} \text{ if } \exists \boldsymbol{\eta} \in \mathbb{R}^{K-1} \}
\end{aligned} \tag{44}$$

6.4 $K = 2$ and $\text{rank}(A_{ac}(\mathbf{x})) = 2$

Consider the case where there are exactly *two* constraints that are active and linearly independent. In this case, there are as many degrees of freedom in control as the number of independent active obstacles to avoid. Consequently, \mathbf{u}_θ^* is completely determined by the active constraints and hence does not depend on $\hat{\mathbf{u}}$. We formally demonstrate this claim as follows. Let $i_1, i_2 \in \{1, 2, \dots, M\}$ be the indices of the two constraints that are active. These constraints satisfy (30) except that here $A_{ac}(\mathbf{x}) \in \mathbb{R}^{2 \times 2}$ and $\text{rank}(A_{ac}(\mathbf{x})) = 2$. This problem is well-posed, its solution is given by

$$\begin{aligned}
A_{ac}(\mathbf{x}) \mathbf{u}_\theta^* &= \mathbf{b}_{ac}(\mathbf{x}) \\
\implies \mathbf{u}_\theta^* &= A_{ac}^{-1}(\mathbf{x}) \mathbf{b}_{ac}(\mathbf{x})
\end{aligned} \tag{45}$$

where the inverse exists because $\text{rank}(A_{ac}(\mathbf{x})) = 2$. Since neither A_{ac}^{-1} nor \mathbf{b}_{ac} depend on $\boldsymbol{\theta}$ (6), \mathbf{u}_θ^* also does not depend on $\boldsymbol{\theta}$. Hence, point-wise identification is not possible. For feasible region-based identification, we cannot get information from inactive constraints because that is also contingent upon on \mathbf{u}_θ^* depending on $\boldsymbol{\theta}$ (subsec. 5.1). Nevertheless, non-negativity of Lagrange multipliers is still useful. Since constraints i_1 and i_2 are active, it follows from dual feasibility (9) and subsec. 5.1 that $\mu_{i_1 \theta}^* \geq 0$ and $\mu_{i_2 \theta}^* \geq 0$. Define $\boldsymbol{\mu}_\theta^{ac} = (\mu_{i_1 \theta}^*, \mu_{i_2 \theta}^*)$, then from (15) and (45), we have

$$\begin{aligned}
\mathbf{u}_\theta^* &= \hat{\mathbf{u}} - \frac{1}{2} A_{ac}^T \boldsymbol{\mu}_\theta^{ac} \\
\implies A_{ac}^{-1} \mathbf{b}_{ac} &= \hat{\mathbf{u}} - \frac{1}{2} A_{ac}^T \boldsymbol{\mu}_\theta^{ac} \\
\implies \boldsymbol{\mu}_\theta^{ac} &= 2A_{ac}^{-T} (\hat{\mathbf{u}} - A_{ac}^{-1} \mathbf{b}_{ac}) \\
\implies \boldsymbol{\mu}_\theta^{ac} &= 2A_{ac}^{-T} (C\boldsymbol{\theta} + \mathbf{d} - A_{ac}^{-1} \mathbf{b}_{ac}) \\
\implies \boldsymbol{\mu}^{ac} \succeq \mathbf{0} &\iff -A_{ac}^{-T} C\boldsymbol{\theta} \preceq A_{ac}^{-T} (\mathbf{d} - A_{ac}^{-1} \mathbf{b}_{ac})
\end{aligned} \tag{46}$$

Thus the desired set $\boldsymbol{\Omega}$ is defined using (13) as follows

$$\boldsymbol{\Omega} := \{ \hat{\boldsymbol{\theta}} \in \mathbb{R}^p \mid -A_{ac}^{-T} C \hat{\boldsymbol{\theta}} \preceq A_{ac}^{-T} (\mathbf{d} - A_{ac}^{-1} \mathbf{b}_{ac}) \} \tag{47}$$

where we have omitted the dependence on \mathbf{x} and \mathbf{x}_j^o to keep the notation light.

6.5 $2 < K \leq M$ and $\text{rank}(A_{ac}(\mathbf{x})) = 2$

Finally, let's consider the case where there are $K > 2$ constraints that are active and two of them are linearly independent. The remaining active constraints are linear combinations of these two. In this case, like in case D, there are fewer degrees of freedom in control than the number of independent active obstacles to avoid. Consequently, neither \mathbf{u}_θ^* nor the robot dynamics depend on θ making point-wise identification infeasible. Let $i_1, i_2, \dots, i_K \in \{1, 2, \dots, M\}$ be the indices of the active constraints. These constraints satisfy (30) except that here $\text{rank}(A_{ac}(\mathbf{x})) = 2$. This problem is well-posed albeit overdetermined, its solution is given by

$$\mathbf{u}_\theta^* = A_{ac}^\dagger(\mathbf{x})\mathbf{b}_{ac}(\mathbf{x}) \quad (48)$$

where A_{ac}^\dagger denotes the Moore-Penrose pseudoinverse which exists because $\text{rank}(A_{ac}(\mathbf{x})) = 2$. Since \mathbf{u}_θ^* is independent of θ , point-wise identification is not possible. Likewise, for feasible region-based identification, we cannot get information from inactive constraints. Nevertheless, non-negativity of Lagrange multipliers is still useful. Since constraints i_1, i_2, \dots, i_K are active, it follows from dual feasibility (9) and subsec. 5.1 that $\mu_{i_1\theta}^*, \dots, \mu_{i_K\theta}^* \geq 0$. From (15) and (48), we have

$$\begin{aligned} A_{ac}^\dagger \mathbf{b}_{ac} &= \hat{\mathbf{u}} - \frac{1}{2} A_{ac}^T \boldsymbol{\mu}_\theta^{ac} \\ \implies A_{ac}^T \boldsymbol{\mu}_\theta^{ac} &= 2(C\theta + \mathbf{d} - A_{ac}^\dagger \mathbf{b}_{ac}) \\ \implies A_{ac}^T \boldsymbol{\mu}_\theta^{ac} &= \mathbf{w} \end{aligned} \quad (49)$$

We defined $\mathbf{w} = 2(C\theta + \mathbf{d} - A_{ac}^\dagger \mathbf{b}_{ac})$ to simplify notation. The above linear system for determining $\boldsymbol{\mu}_\theta^{ac}$ is full rank but underdetermined because $\text{rank}(A_{ac}^T) = 2$. The SVD of A_{ac}^T is $A_{ac}^T = \tilde{U}\tilde{\Sigma}\tilde{V}^T$ where

$$\begin{aligned} \tilde{U} &\in \mathbb{R}^{2 \times 2} \\ \tilde{\Sigma} &= [\tilde{\Sigma}_m \mid 0^{2 \times K-2}] \text{ where, } \tilde{\Sigma}_m \in \mathbb{R}^{2 \times 2} \\ \tilde{V} &= [\tilde{V}_1, \tilde{V}_2] \text{ where, } \tilde{V}_1 \in \mathbb{R}^{K \times 2}, \tilde{V}_2 \in \mathbb{R}^{K \times K-2}. \end{aligned} \quad (50)$$

Since \tilde{V} forms a basis for \mathbb{R}^K , any $\boldsymbol{\mu}$ can be expressed as

$$\begin{aligned} \boldsymbol{\mu} &= [\tilde{V}_1, \tilde{V}_2] \begin{bmatrix} \tilde{\boldsymbol{\mu}}_1 \\ \tilde{\boldsymbol{\mu}}_2 \end{bmatrix} = \tilde{V}_1 \tilde{\boldsymbol{\mu}}_1 + \tilde{V}_2 \tilde{\boldsymbol{\mu}}_2. \\ \implies A_{ac}^T \boldsymbol{\mu} - \mathbf{w} &= \tilde{U} [\Sigma_m, 0] \begin{bmatrix} \tilde{V}_1^T \\ \tilde{V}_2^T \end{bmatrix} [\tilde{V}_1, \tilde{V}_2] \begin{bmatrix} \tilde{\boldsymbol{\mu}}_1 \\ \tilde{\boldsymbol{\mu}}_2 \end{bmatrix} - \mathbf{w} \\ &= \tilde{U} \Sigma_m \tilde{\boldsymbol{\mu}}_1 + 0 \cdot \tilde{\boldsymbol{\mu}}_2 - \mathbf{w} = 0 \end{aligned} \quad (51)$$

Choosing $\tilde{\boldsymbol{\mu}}_1 = \Sigma_m^{-1} \tilde{U}^T \mathbf{w}$ and $\tilde{\boldsymbol{\mu}}_2 = \boldsymbol{\eta} \in \mathbb{R}^{K-2}$, we find that

$$\begin{aligned} \boldsymbol{\mu}_\theta^{ac} &= \tilde{V}_1 \tilde{\Sigma}_m^{-1} \tilde{U}^T \mathbf{w} + \tilde{V}_2 \boldsymbol{\eta} \\ &= 2\tilde{V}_1 \tilde{\Sigma}_m^{-1} \tilde{U}^T (C\theta + \mathbf{d} - A_{ac}^\dagger \mathbf{b}_{ac}) + \tilde{V}_2 \boldsymbol{\eta} \end{aligned} \quad (52)$$

Here $\boldsymbol{\eta} \in \mathbb{R}^{K-2}$ are floating variables which can assume any value. From non-negativity of $\boldsymbol{\mu}_\theta^{ac}$ we get

$$\boldsymbol{\mu}_\theta^{ac} \succeq 0 \iff -2\tilde{V}_1 \tilde{\Sigma}_m^{-1} \tilde{U}^T C\theta - \tilde{V}_2 \boldsymbol{\eta} \preceq 2\tilde{V}_1 \tilde{\Sigma}_m^{-1} \tilde{U}^T (\mathbf{d} - A_{ac}^\dagger \mathbf{b}_{ac}) \quad (53)$$

Thus the desired set Ω where θ belongs is defined using (53) as follows

$$\Omega := \{\hat{\theta} \in \mathbb{R}^p \mid -2\tilde{V}_1 \tilde{\Sigma}_m^{-1} \tilde{U}^T C\hat{\theta} - \tilde{V}_2 \boldsymbol{\eta} \preceq 2\tilde{V}_1 \tilde{\Sigma}_m^{-1} \tilde{U}^T (\mathbf{d} - A_{ac}^\dagger \mathbf{b}_{ac}) \text{ if } \exists \boldsymbol{\eta} \in \mathbb{R}^{K-2}\} \quad (54)$$

7 Simulation Results

In this section, we present simulations to demonstrate the effectiveness of our proposed region-based parameter identification. We consider a multirobot system in which the task of each robot is to reach a goal position while avoiding collisions with other robots. The observer's problem is to infer the goal of each robot using its positions and velocities as measurements. We use an Unscented Kalman Filter (UKF) as a baseline for goal estimation, to show how our region-based estimator outperforms a UKF.

To infer robot goals using the region-based estimator, the observer must estimate the instantaneous feasible region $\Omega(t)$ using which it will compute the *cumulative* feasible region $\Theta(t)$. Recall from (3), that $\Theta(t)$ is computed by taking intersections of all $\Omega(t)$ over time, and then its intersection with a compact set Θ_0 where the θ is known to belong. For the goal inference problem, the observer can compute the instantaneous set $\Omega(t)$ using the results we derived in subsections 6.1-6.5 by checking how many obstacles are active at a given time (17). As for Θ_0 , we assume that the observer knows some reasonable upper and lower bounds on the location of the robot goals. In subsec. 6.4, we showed that when two or more obstacles are active at a given time, point-wise parameter identification is not possible because robot dynamics are independent of the underlying parameters. The feasible region based estimator, however, can give produce a bounded set where the underlying parameter belongs. To substantiate this, we first show results for a single robot navigating in an environment consisting of static obstacles. Subsequently, we will show results for multirobot inference as well. In Figs. 1(a)-1(c), an ego robot (red) is trying to reach its goal \mathbf{x}_d shown in black. The green regions correspond to $\Theta(t)$ computed using the instantaneous feasible regions $\Omega(t)$. As the robot moves to the right, obstacles one and two remain active until $t = 1.1s$. After this, obstacles three and

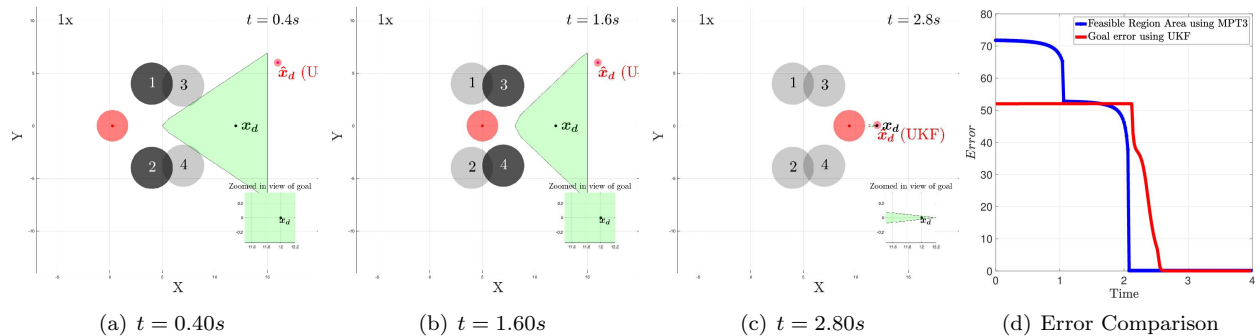


Figure 1: (a)-(c) Goal identification for a robot navigating among static obstacles. Dark discs represent active obstacles. (d) Comparison of region-based estimation (blue) with a UKF. Video at <https://youtu.be/jH3mxZhX2mA>

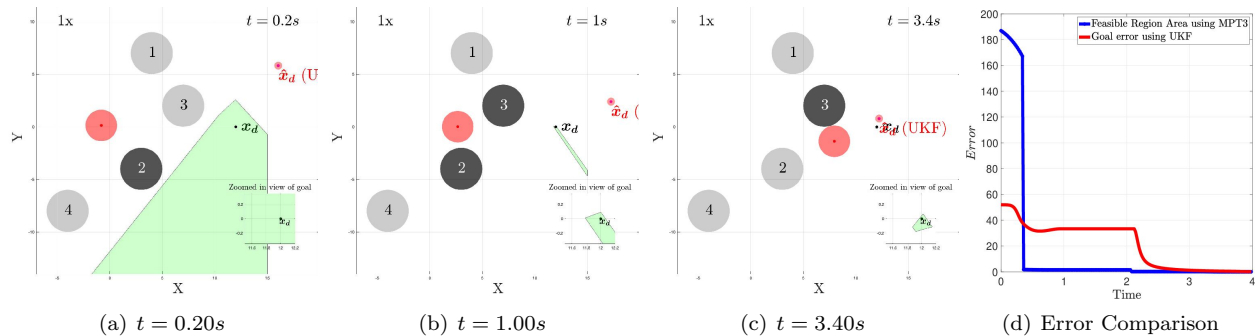


Figure 2: (a)-(c) Goal identification for a robot navigating among static obstacles. Dark discs represent active obstacles. (d) Comparison of region-based estimation (blue) with a UKF (red). Video at <https://youtu.be/VthIXiBvjfU>

four stay active until $t = 2.1s$. Therefore, a point-wise estimator will not result in reduction in estimation error. Thus, until $t = 2.1s$, the UKF based estimator (red) does not get updated as is evident from Figs. 1(a)-1(b). On the contrary, the green regions continue to shrink. The inset in these figures shows a zoomed in view around the goal to empirically demonstrate that the feasible region always remains non-empty and always encompasses the goal. In Fig. 1(c), all the obstacles are inactive so the UKF estimator will converge. Additionally, the feasible region based estimator will also provide feasible regions based on the set defined in subsec 6.1. We measure the $\text{Vol}(\Theta(t))$ of the feasible regions by computing the area of the green regions using the MPT3 toolbox in MATLAB. Fig. 1(d) shows this area as a function of time and demonstrates the fast convergence as the robot moves, in comparison to the UKF estimator based error shown in red.

In Figs. 2(a)-2(c), we consider a different arrangement of obstacles. Here, until $t = 0.38s$, only one obstacle is active; then until $t = 2.08s$ exactly two obstacles are active, then until $t = 3.68s$ exactly one is active, following which all obstacles are inactive. As the robot moves, the region-based estimator converges quickly to the true goal as is evident in the insets in Figs. 2(a)-2(c). Further, the errors shown in 2(d) illustrate how the UKF estimator takes some time to converge while the region based estimator converges much faster to the true goal. Furthermore, the UKF estimator does not respect the feasibility of the goal, it frequently remains outside the feasible region. This is to be expected because the derivation for UKF estimator does not consider non-negativity of Lagrange multipliers. 2(d).

Finally, we consider a multirobot system in Fig. 3 in which we run parallel region-based estimators synchronously. There are four robots shown in different colors, their goals are shown in the same colors as are their feasible regions $\Theta_i(t) \forall i \in \{1, 2, 3, 4\}$. Fig. 4 compares the convergence of region based estimator (left) to the UKF estimator (right). As is evident, the region-based estimators converge to the true goal much faster than the UKF based estimators.

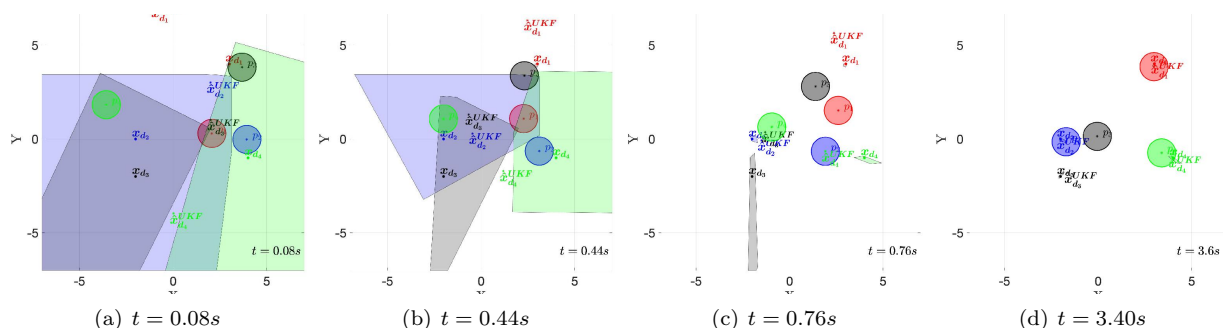


Figure 3: (a)-(d) Region-based goal identification for multirobot system. Video at <https://youtu.be/eh3nApDVUws>

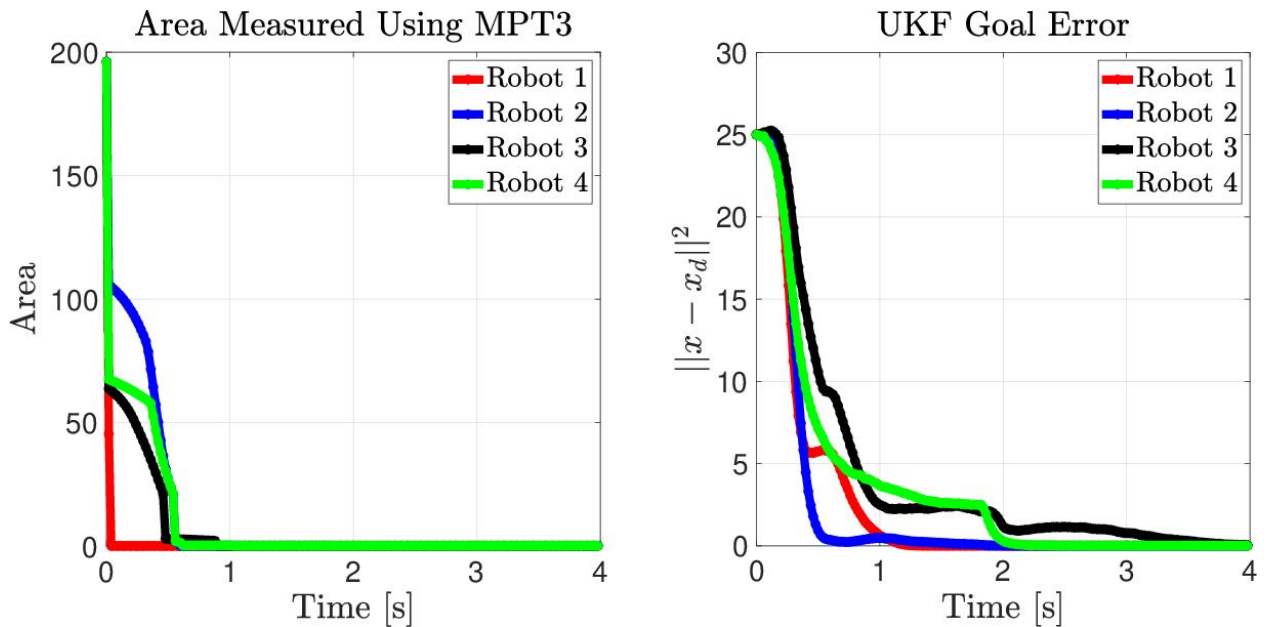


Figure 4: Estimation errors for our proposed region-based estimator(left) and UKF (right). The areas shrink around the goal much faster than UKF errors converge to zero.

8 Conclusions

In this paper, we proposed a region-based parameter identification method to compute bounds on parameters of tasks being performed by a multirobot system. This approach works even when exact estimation using point-wise estimators is likely to fail. We used KKT conditions to describe the instantaneous feasible set of the underlying parameters using which we computed contracting sets where the parameters must belong. To demonstrate the effectiveness of our method, we showed numerical simulations for inferring goals of a single robot moving amongst static obstacles as well as for each robot in a multirobot system. These scenarios show how our region based estimator outperforms a UKF in terms of speed of convergence. There are several directions that we would like to pursue further. Firstly, we want to see how physical intervention by the observer be designed to further expedite the convergence of the region-based estimator. Additionally, we also want to explore connections between our work and prior work on inference using inverse convex optimization.

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*, pp. 529–538, Springer.
- [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] A. Byravan, M. Monfort, B. Ziebart, B. Boots, and D. Fox, “Graph-based inverse optimal control for robot manipulation,” in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.
- [7] S. Thrun, “Probabilistic robotics,” *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.
- [8] J. Na, M. N. Mahyuddin, G. Herrmann, X. Ren, and P. Barber, “Robust adaptive finite-time parameter estimation and control for robotic systems,” *International Journal of Robust and Nonlinear Control*, vol. 25, no. 16, pp. 3045–3071, 2015.
- [9] V. Adetola and M. Guay, “Performance improvement in adaptive control of linearly parameterized nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2182–2186, 2010.
- [10] M. Hartman, N. Bauer, and A. R. Teel, “Robust finite-time parameter estimation using a hybrid systems framework,” *IEEE transactions on automatic control*, vol. 57, no. 11, pp. 2956–2962, 2012.
- [11] J. C. Willems, P. Rapisarda, I. Markovskiy, and B. L. De Moor, “A note on persistency of excitation,” *Systems & Control Letters*, vol. 54, no. 4, pp. 325–329, 2005.
- [12] C. Yang, Y. Jiang, W. He, J. Na, Z. Li, and B. Xu, “Adaptive parameter estimation and control design for robot manipulators with finite-time convergence,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8112–8123, 2018.

- [13] J. Zhao, X. Wang, G. Gao, J. Na, H. Liu, and F. Luan, “Online adaptive parameter estimation for quadrotors,” *Algorithms*, vol. 11, no. 11, p. 167, 2018.
- [14] J. S. Grover, C. Liu, and K. Sycara, “Parameter identification for multirobot systems using optimization based controllers (extended version).” <https://arxiv.org/abs/2009.13817>, 2020.
- [15] V. Adetola and M. Guay, “Finite-time parameter estimation in adaptive control of nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 53, no. 3, pp. 807–811, 2008.
- [16] 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.
- [17] T. F. Edgar, “Recursive least squares parameter estimation for linear steady state and dynamic models.”
- [18] P. A. Ioannou and J. Sun, *Robust adaptive control*. Courier Corporation, 2012.
- [19] K. S. Narendra and A. M. Annaswamy, *Stable adaptive systems*. Courier Corporation, 2012.
- [20] J. Grover, C. Liu, and K. Sycara, “Deadlock analysis and resolution in multi-robot systems (extended version),” 2019.
- [21] T. Wei and C. Liu, “Safe control algorithms using energy functions: A unified framework, benchmark, and new directions,” in *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 238–243, IEEE, 2019.
- [22] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.