2024

Cooperative Rendezvous in Multi-vehicle systems

Rajeev Shobhit Voleti

Cooperative Rendezvous in Multi-vehicle systems

by

RAJEEV SHOBHIT VOLETI

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON
December 2023

Acknowledgements

I would like to express my deepest appreciation to my advisor, Professor Kamesh Subbarao, for his invaluable guidance, patience, and support throughout my research journey.

My sincere gratitude extends to the members of the exam committee, Prof. Animesh Chakravarthy, Prof. Linda Wang, Prof. Li Wang and Prof. Paul Davidson for their mentorship, time, effort, and insightful feedback that have greatly contributed to my academic development.

I wish to thank my father Seshagiri Rao Voleti for his unwavering belief in my capabilities, my mother Radha Voleti for her nurturing love, care and support, and my wife Olivia, my warmth and my joy. Thank you for being by my side, always.

I am grateful for the financial support received from the Office of Naval Research through Grant No. N00014-18-1-2215, without which this research wouldn't be possible.

Finally, a special thanks to Dr. Diganta Bhattacharjee for his technical assistance, support, and inspiration.

November 13, 2023

Abstract

Cooperative Rendezvous in Multi-vehicle systems

Rajeev Shobhit Voleti, Ph.D.
The University of Texas at Arlington, 2023

Supervising Professor: Kamesh Subbarao

The field of cooperative, multi-vehicle systems has witnessed a significant expansion and evolution, yielding considerable opportunities for improved autonomy, resilience, and robustness. Despite these promising developments, complex challenges persist in ensuring secure and efficient rendezvous among cooperative peers. The term "rendezvous," within the realm of cooperative control, refers to the simultaneous convergence of multi-vehicle systems to a designated target location. In missile guidance, the problem of multiple pursuer missiles achieving rendezvous with a target is termed as a salvo attack. Current methodologies often grapple with issues related to synchronization, high latency and network security, all of which can adversely impact system performance and reliability. Furthermore, traditional consensus protocols tend to fall short in mitigating threats within such complex environments, leaving the system susceptible to a range of potential attacks. For example, traditional flocking or cooperative rendezvous methods utilise a shared network of position and velocity measurements to synchronize and achieve rendezvous. Malicious agents with access to the information in this network would lead to compromised strategy and potentially interrupt the interception of the pursuers with the target. These predicaments highlight an urgent need for the development of robust and innovative solutions.

Our research aims to bridge these gaps by proposing a pioneering solution: the orchestration of multi-vehicle secure rendezvous within a finite time period through the use of a shared network time-to-go consensus protocol. The essence of this approach lies in its utilization of a finite time, time-dependent control input. For the case of the multiple pursuer salvo problem, the guidance law proposed effects same time position convergence of these multiple missiles to their target location. This unique methodology is brought to life through a variety of simulation scenarios, highlighting its potential in addressing the complexities of secure rendezvous.

The scope of the research further extends to incorporate a terminal phase guidance law designed to enhance target acquisition. This is achieved by enabling more practical position convergence through state estimation of the target vehicle. A critical comparison of this approach with the nonlinear estimators extensively used in current literature affirms the viability of the proposed framework.

Building on this, the research explores the application of this convergence framework to accommodate an array of protocols within multi-agent systems. An in-depth analysis of the decentralized, leaderless network of agents, focusing on the conditions necessary for rendezvous, further reinforces the versatility and applicability of this framework.

In pursuit of an optimal solution to the secure rendezvous problem, a novel collocation-based control optimization scheme has been proposed. This strategy exhibits significant potential to extend the rendezvous framework, enabling it to cater to higher-order, centralized, objective-based requirements. The framework developed has been shown to be effective for heterogeneous, networked teams of agents. The realization of this approach would mark a significant step forward in overcoming

the challenges inherent to secure rendezvous in cooperative, multi-vehicle systems, thereby bringing us closer to the ultimate goal of enhanced system autonomy, resilience, and robustness.

Table of Contents

Appendix

List of Illustrations

List of Tables

Table Page

Chapter 1

Introduction

In the landscape of modern cooperative systems, the task of effectively managing multi-vehicle networks has emerged as an area of profound interest and research. These systems, characterized by their cooperative nature and complexity, provide significant prospects for improving autonomy, resilience, and robustness. Nevertheless, several challenges remain, especially when it comes to the efficient and secure rendezvous of these cooperative entities. The term “rendezvous” in this context is described as the act of simultaneous convergence of multi-vehicle systems to a specific target location. The key here being that the approach towards the target by the vehicles must happen at the same time. This problem is also posed as the salvo problem in missile dynamics. In existing literature, the majority of methods addressing rendezvous or flocking primarily rely on shared position and velocity information. However, this approach has inherent vulnerabilities. It could, for instance, expose the system to a man-in-the-middle attack, whereby a malicious agent intercepting communications may gain access to sensitive information about the vehicles. Such a security breach could have potentially catastrophic consequences, especially in critical applications such as missile target engagement. As such, there is an urgent need to address these security vulnerabilities in the design and execution of rendezvous or flocking methods for multi-vehicle systems.

1.1 Goals and Objectives

Outlined below are the specific goals and objectives that will steer this research:

- Develop and refine a framework that enables finite-time, dynamic solutions for the rendezvous problem in multi-agent systems, with a focus on:
  - Establishing secure information sharing protocols between agents.
  - Utilizing leaderless communication systems to enhance security.
  - Analyzing the impact of different consensus protocols on security and efficiency during rendezvous.
  - Investigating the influence of sparse communication networks on successful rendezvous.
- Explore the practical application of the proposed framework to the complex scenario of multiple missile-target engagements, aiming to:
  - Extend the solution to include state estimation for greater resilience against uncertainties.
  - Assess the robustness of the proposed solutions in dynamic and uncertain environments.
- Propose an optimal solution to the secure rendezvous problem by:
  - Conducting a multifaceted analysis of various numerical implementation methods.
  - Incorporating heterogeneous multi-vehicle systems into the framework.
  - Adapting the results to centralized communication networks for broader applicability.
  - Applying the framework to specific objectives and practical scenarios in the field.

These goals and objectives are carefully crafted to ensure a thorough exploration of the secure rendezvous problem in multi-agent systems, offering a blend of theoretical innovation and practical application.

1.2 Background and Motivation

In order to provide a comprehensive foundation for our work, this section offers a detailed examination of the relevant background information, the existing literature in the field, and the key motivations guiding our research objectives.

Leaderless Protocols for Same-Time Consensus

The cooperative rendezvous of several agents to a target has been well documented as a solution to *the rendezvous problem* [1] and *the salvo problem* in missile dynamics. It is defined as the simultaneous or same-time position consensus of numerous agents to a target location. In modern combat with targets being more resilient and secure than ever before, a simultaneous attack has major advantages in pacifying defenses and ensuring a target is intercepted. Strategic targets may also be guarded by defending agents to subdue pursuer attacks in which case through salvo, even if some of the pursuing agents are neutralized, the remaining agents can intercept and overwhelm the target. Certain targets such as battleships may be resilient to interception by a single pursuer missile in which case it becomes necessary to execute a salvo attack to pacify the target. Rendezvous has also been studied from a game theoretic perspective [2]. Lindsay and Givigi [2] have proposed position consensus as a cooperative game with the players attempting to maximize their utility functions which is determined by the total number of players, the greatest possible distance between any two players in the game and their respective distances from each other. The location of consensus is determined based on a normalized remainder vector generated from the weighted difference in the players' utility functions. It is assumed that the players have information about the utility function and thereby locations of all the players in the game at all times and not just their neighbors and the rendezvous location is determined based on the state of each player.

Motivation: Rendezvous is effected in literature by sharing the location of each agent with their connected neighbors [3] which leaves the location of the agents and the target vulnerable to interception by an outsider tapping into communications. Secure rendezvous is a spin on the existing problem with an added element of resistance to information interception. The rendezvous for drones or unmanned aerial vehicles (UAVs) to a secure location can be subject to a man in the middle attack leaving the location of the secure target vulnerable. Second order consensus in position and velocity for multi-agent systems with double-integrator dynamics has been posed as a modified rendezvous problem by Yu et al. [4] and has been shown to achieve consensus in position and velocity asymptotically in strongly connected graph communication topologies leveraging shared position and velocity information of the agents. Qin et al. [5] have extended second order consensus for systems with separate graph communication topologies modeling position and velocity information exchange and proposed necessary and sufficient conditions on double-integrator systems achieving second order consensus asymptotically. Similarly, Liu et al. [6] have recently proposed consensus under a communication topology. Zhang et al. [7] have extended a secure rendezvous framework to be resilient to a DoS (Denial of Service) attack. However, neither of these proposed methods discuss the resiliency of the networked system if a malicious outsider hacks into the information being shared among the nodes. The hacker can supply spurious disturbance signals to the nodes with the highest out-degree and lead to a spurious consensus. The use of shared position and velocity leaves the agents in rendezvous doubly vulnerable to having their individual locations and the location of consensus along with their velocity information intercepted through their shared communication by an outsider. The use of shared time-to-go or expected time to arrival (ETA) prevents any location or velocity information about the agents or the target from being discovered even for an outsider intercepting communications.

In the existing literature, planar same-time consensus has been posed in the context of fixed terminal time or finite terminal time. Fixed terminal time solutions to the missile salvo problem [8] have been proposed to achieve simultaneous consensus to the location of a target with the convergence time specified a priori. This has been shown to work in a leader-follower framework by Tian et al. [9]. But, as with [3], these works also assume agent positions are shared among connected neighbors. Due to the time independent nature of fixed time consensus, a break in consensus of any agent can cause failure of consensus at the target. The salvo problem has also been solved by cost function minimization in a stationary target chase scenario [10] or maximizing the damage expected by minimizing the time to impact [11]. Kang et al. [12] have leveraged a model predictive controller to drive the position states to their target subject to acceleration and field-of-view constraints. Alternatively, a sliding mode controller effected by a super twisting algorithm has been proposed in [13]. Finite-time consensus is a more general approach to the fixed time consensus problem. Using a combination of artificial potential field and sliding mode control, a finite-time stabilizing controller for tracking or consensus of second-order leader-follower systems augmented by disturbance rejection is evidenced as robust in [14].

Hu et al. [15] have proposed consensus in linear multi-agent systems through the use of event triggered strategies. A consensus event is triggered based on an error function of the agent's position. This framework is shown to achieve consensus asymptotically through shared positional information. Dong and Xu [16] have extended this framework for consensus in leader-follower systems with linear systems modeled by single-integrator and double-integrator systems. The event trigger depends solely on the information gathered in the previous consensus evaluation as opposed to having the measurement errors being monitored and has been extended to leaderless consensus in multi-agent systems with nonlinear dynamics in [17]. However, convergence for these methods is still asymptotic and the corresponding control designs do not involve minimization of a suitable cost function to induce optimal performance. Leaderless consensus in finite time among multi-agent systems has been shown to be achieved by Du et al. [18] through the use of shared positional information. While the agents use local information, the agents are not stealthy and the target location is vulnerable to interception. The use of shared ETA implies unavailability of shared vehicle location or velocity which imparts a degree of resilience to the system. Since time-to-go information is the sole information shared, a malicious agent cannot estimate the consensus value, the positions of the vehicles, or their speeds without knowledge of the communication protocol despite having access to the shared communication data. Zadka et al. [19] have proposed geometric rules for simultaneous target interception through the use of shared time-to-go using a max-time consensus protocol in a leader-follower framework.

Finite Time Consensus Guidance Laws for Salvo Engagement

We further explore the rendezvous problem in context of missile salvo. Soren et al. [3] have proposed a controller based on the graph weighted difference in positions of the agents. However, the convergence is asymptotic in nature and does not ascertain consensus at the same time and position being the shared variable is insecure and vulnerable to an outside entity being aware of their position at times when information is shared between the agents. Planar same time consensus is also posed in the context of fixed terminal time as a solution to the missile salvo problem [8] and as with [3], this work also assumes agent positions are shared among connected neighbors. Tian et al. [9] have also proposed consensus through the use of shared positions under a leader-follower framework. However, being fixed time, an unexpected deviation of one of the agents can cause consensus to fail at the target because of the time independent nature of the solution. It is also important to examine a scenario where the target is maneuvering to account for deviations in the solution and rendezvous time.

In literature, the salvo problem (multiple agents released to 'overwhelm' a target) has been addressed by minimising a cost function comprising of the control effort and reciprocal of the time-to-go for a group of missiles chasing a stationary target [10] or minimising the time consumption and maximising the damage expected [11]. Kang et al. have proposed driving the position states to the target subject to field of view and acceleration constraints through the use of a model predictive controller [12]. Alternatively, a super twisting algorithm has been used to effect a sliding mode controller [13]. Obtaining consensus in finite time is a more general solution to the fixed time consensus problem i.e the time taken to reach consensus is defined ahead of time. Through the use of a combination of sliding mode control and artificial potential field, a modified finite time stabilizing controller for consensus or tracking of second order leader follower systems with disturbance rejection is shown to be robust [14]. The time to converge for these methods is still asymptotic and as is common with such consensus algorithms, the time to converge, even when accounting for the initial conditions is not optimal. All of these methods are applicable to a stationary target. Situations may arise when the target is maneuverable; hence, a need to account for the maneuverability of the target. Yucelen et al. have proposed a finite-time control law for cooperative pursuer target engagement utilizing time transformations to achieve finite-time consensus [20]. Though, the time to convergence is an apriori user defined parameter and the control law is time independent

Due to the unavailability of vehicle location or velocity measurements, the use of time-to-go, or expected time to arrival (ETA), on the other hand provides a degree of resilience to the system. Given the fact that the time-to-go information is exchanged, an outsider who is ignorant of the consensus protocol cannot estimate the locations of the vehicles, their velocities, or the consensus value.

Further, Zadka et al. have proposed geometric consensus based control laws for rendezvous utilizing the max time-to-go protocol in a leader follower framework [19]. Sen et al. have explored max-tracking in multi-agent systems with single integrator dynamics in a distributed network. However, the protocols developed are for consensus to a stationary target assuming perfect information about the states of the pursuers. There has been extensive research in estimation and rendezvous with a non-cooperating target. However, the dynamics of the pursuer is assumed to be linear continuous time [21]. In situations where non linear dynamics have been considered,the engagement is studied between a single pursuer and target [22] [23].

Optimal Finite Time Cooperative Rendezvous

The coordination and cooperation of multiple vehicles to perform collective tasks is a dynamic and complex problem that poses a myriad of challenges. The problem of multi-vehicle coordination has transitioned from a largely theoretical concern to a practical challenge of considerable complexity and urgency. This transition has been driven, in large part, by the surging demand for autonomous vehicles in a range of sectors and applications, including but not limited to defense, logistics, surveillance, and exploration missions [24]. In each of these domains, the ability to coordinate the movements of multiple vehicles – whether terrestrial, aerial, or marine – can yield significant improvements in efficiency, effectiveness, and safety. One particularly salient aspect of this challenge is the problem of cooperative rendezvous, whereby multiple vehicles are tasked with arriving at a common destination at the same time. The intrinsic difficulty of this task is compounded by its critical importance to a variety of mission scenarios, from landing and docking missions to precision salvo strikes in military contexts [19]. Given this confluence of challenge and importance, to address this problem, this paper presents an innovative approach that leverages advancements in finite-time optimal control. By harnessing the power of shared network information, specifically time-to-go values, our approach aims to coordinate multi-vehicle systems with precision and security. Through the use of time-to-go as the only shared network state, an intruder intercepting communications can neither locate the vehicles nor estimate their path or speed of motion. At a time when autonomous vehicles are becoming increasingly ubiquitous and indispensable, the development of such coordination mechanisms holds great promise.

Motivation: Solving the multi-vehicle rendezvous problem has spurred the development of many varied approaches. Cooper's research, for instance, has made use of artificial potential fields to accomplish multi-vehicle search and rescue [25]. However, it has been observed that the more recent approaches have gravitated towards strategies that deploy Model Predictive Control (MPC). Taner and Subbarao have, in this vein, proposed a model predictive framework for cooperative systems specifically tackling the quadrotor docking problem [26]. Furthermore, Persson et al. have developed an MPC solution for cooperative rendezvous involving a fixed-wing UAV and a ground vehicle [27]. Meanwhile, Earnhardt et al. have demonstrated the potential of a discrete optimal control in facilitating platooning using predicted fuel-optimal operation. While these aforementioned methods have indeed proven to be effective in many scenarios, they have also revealed their inherent limitations when applied to the multi-agent rendezvous problem. To elaborate, traditional MPC primarily focuses on minimizing a cost function over a control horizon, however, it gives no due consideration for the primary goal of rendezvous. Additionally, the optimality of MPC is constrained within a finite predictive horizon and is subject to the precision of the model and the optimization algorithm used to solve the control problem. Furthermore, the solution provided by MPC does not cater to finite time control, and the quality of the solution depends largely on how accurately the objective function and constraints depict the system's desired behavior. Turning our attention to the finite-time consensus problem, it is viewed as a broader interpretation of the fixed time consensus problem, with consensus being achieved in finite time as opposed to a pre-determined fixed time. Wang et al. have developed a finite time consensus algorithm that can be applied to nonlinear agents with switching communication topologies and constrained states [28]. In our previous works, we have drawn attention to the benefits of employing a leaderless, distributed framework of time-to-go consensus and simultaneous positional convergence [29] [30]. However, it is important to note that these methods did not guarantee optimality. Given the inherent challenges associated with cooperative rendezvous, particularly its time-dependent and gain-free nature, there is significant research being undertaken across various disciplines. Traditional methodologies such as Model Predictive Control (MPC), despite being useful tools, often lack the flexibility required to effectively manage a broad spectrum of vehicle capabilities and dynamic environments. To address these limitations, we propose an innovative approach. Our approach utilizes the power of collocation in generating and solving a finite-time optimal control problem, potentially offering a path to overcome the challenges of traditional methods.

1.3 Contributions

Leaderless Protocols for Same-Time Consensus

We develop a foundational framework for achieving consensus on time-to-go and positional convergence in leaderless multi-agent systems.

This has been initiated by establishing time-to-go consensus protocols for systems modeled by double integrators. The aim being to further extend these results to integrate with a communication framework characterized by sparsity yet ensuring connectivity among the agents. The study analyzes the behaviors manifested by the multi-agent systems when subject to various communication protocols, assessing their efficacy and potential compromises during rendezvous operations.

The publication based on this research is as follows:

- Voleti, R.S., Bhattacharjee, D. and Subbarao, K., 2023. Leaderless Time-To-Go Protocols for Same-Time Position Consensus. IEEE Transactions on Systems, Man, and Cybernetics: Systems, doi:10.1109/tsmc.2023.3234563

Finite Time Consensus Guidance Laws for Salvo Engagement

We introduce the 'max time protocol' as a strategic approach to address the challenge of multiple missile-target engagements. This protocol is designed to optimize the use of finite time and dynamic aspects, thus offering a robust solution to engagement scenarios. It incorporates state estimation techniques to predict the target's trajectory and expand the protocol's applicability to systems with sparsely connected communication graphs.

The valuable contributions of this research have led to the following academic publications:

- Voleti, R.S., Bhattacharjee, D. and Subbarao, K., 2022. Finite same-time consensus guidance laws for unmanned aerial systems. In AIAA SCITECH 2022 Forum (p. 1845), Jan. 03, 2022. doi: 10.2514/6.2022-1845.
- Voleti, R.S., Bhattacharjee, D. and Subbarao, K., 2022. Finite same-time consensus guidance laws for missile-target salvo engagement. currently under review in the AIAA Journal of Guidance, Control, and Dynamics, 2023.

Optimal Finite Time Cooperative Rendezvous

We scrutinize the established framework for its optimality. The intent is to transcend from feasible solutions to the most advantageous one for any given scenario. This will be achieved through the application of numerical collocation methods and a centralized approach towards optimal rendezvous problem-solving. This is particularly relevant in situations where the security of the agents' positions is not of paramount importance, thereby allowing the exploitation of established finite time optimal control methods. We implement a more generalized framework to incorporate wider range of systems with heterogeneous dynamics and make extensions to systems with sparser yet connected graph communication topologies.

The publication based on this research is as follows:

- Voleti, R.S. and Subbarao, K., Optimal Finite Time Cooperative Rendezvous for Multiple Vehicles., AIAA SCITECH 2024 Forum, Jan 8-12, Orlando, FL.

Chapter 2

Leaderless Time-to-go Protocols for Same-Time Position Consensus

2.1 Problem Statement

Let us consider a group of N agents and let $\mathcal{I} = \{1, 2, ..., N\}$. We consider planar motion of the agents, with a scenario wherein the agents are oriented such that their paths intersect at a point. The agents share solely their time-to-go information among themselves with the information exchange being modeled as a communication graph $\mathcal{G}$. The relative separation between this intersection point and an agent $i \in \mathcal{I}$ is denoted by $r_i(t)$. The dynamics of agent i is given by

$$\begin{aligned} \dot{r}_i(t) &= u_i(t) \\ \dot{u}_i(t) &= a_i \end{aligned} \tag{2.1}$$

where $u_i < 0$ is the radial speed and a_i is the corresponding acceleration input. The discrete-time equivalent of the above dynamics, assuming a sample time of ΔT, is given by

$$\begin{aligned} r_i(k+1) &= r_i(k) + u_i(k)\Delta T \\ u_i(k+1) &= u_i(k) + a_i(k)\Delta T \end{aligned}$$

where $k \in \mathbb{Z}_\star$ with $\mathbb{Z}_\star$ denoting the set of non-negative integers. The orientation or heading angle of an agent is given by θ_i. However, note that we do not consider any variation in θ_i. The agents share a dynamical estimate of time-to-go denoted by $t_{go,i}$. These agents, at regular intervals of ΔT_c follow a protocol dictated consensus value of time-to-go given by $t_{go,C}$ In this setting, we investigate protocols that would make the agents reach the intersection point at the same time. In other words, we are interested in formulating protocols for same-time position consensus defined as:

Definition 2.1.1. *[30] Same-time consensus is said to be achieved if the all the agents converge to within a specified tolerance of the target simultaneously, i.e., the agents satisfy* $r_i(t_f) \leq \epsilon, \forall i \in \mathcal{P}$ *at some* $t_f > 0$.

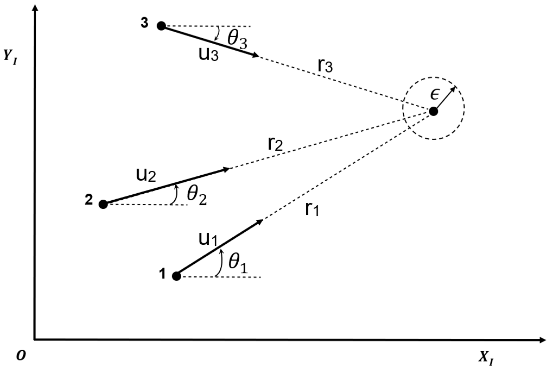


Figure 2.1. A schematic of the same-time consensus.

This is schematically depicted in Fig. 2.1. Our objective is to achieve these different types of consensus only through use of the *shared time-to-go values* following (a) min-time, (b) max-time, and (c) achievable-time. Utilizing a method similar to that stated by Kumar et al. [31], the time-to-go commanded is formulated as

$$t_{go,C}(t) = T_f(k_c) - t_e, \forall t \in [k_c \Delta T_c, (k_c + 1)\Delta T_c] \tag{2.2}$$

where $t_e \in [0, \Delta T_c]$ is the time elapsed after the consensus evaluation at k_c. The value of T_f is dependent on the choice of consensus protocol.We perform a comprehensive analysis of the different protocols. These protocols delineate the behavioral characteristics of the multi-vehicle system, more specifically, their modus operandi in achieving their operational objectives. However, the choice of one protocol over another is not arbitrary. It is influenced by a multitude of factors such as system constraints, operational requirements, and mission objectives. A judicious choice of protocol can contribute significantly to the efficiency of the operation, making it a critical area of study. Having established the crucial role of the protocol in the system's operation, we further explore the effect of the graph communication architecture adopted between the vehicles within the system. The intent here is to study its impact on the consensus framework and position convergence.

2.2 Mathematical Preliminaries

This section introduces the mathematical foundations that are crucial for the development of the control strategies discussed later in this chapter.

2.2.1 Finite-time control

We introduce the dynamical equations governing finite-time error convergence, a concept pivotal to the design of control laws that guarantee the state of a system reaches a desired value in a finite amount of time.

Lemma 2.2.1 (Finite-time Convergence [32]). *Consider a dynamical system governed by the following equation:*

$$\dot{x} = -cx(t)^{\frac{1}{q}} - dx(t)^{\frac{p}{q}} \tag{2.3}$$

where $x \in \mathbb{R}$ represents the state of the system, $\alpha = \frac{p}{q} \in (0,1)$ with p and q being positive odd integers such that $q > p$, and $c, d > 0$ are positive system parameters. Under these conditions, the finite time t_s required for the state x to converge to zero is given by:

$$t_s = \frac{1}{c(1-\alpha)} \ln \left[\frac{c(x(0))^{1-\alpha} + d}{d} \right] \tag{2.4}$$

In the above lemma, the term $\dot{x}$ denotes the time derivative of the state x, and $x(0)$ denotes the initial condition of the state. The significance of this lemma lies in its ability to predict the exact time at which the system's state will reach zero, which is of particular importance in control applications where timing is critical.

2.2.2 Graphs

We will extensively refer to graphs as a means of representing the network of agents sharing their time-to-go information. We will stick to the notation used in literature [33] with $\mathcal{G}$ representing the graph communication network of our agents and $\mathbf{A}$ representing the adjacency matrix of the graph.

Let $\mathcal{G} = (V, E)$ be a graph with n vertices. Then the adjacency matrix $\mathbf{A}$ of $\mathcal{G}$ is an $n \times n$ matrix where $\mathbf{A}[i, j] = 1$ if there is an edge from vertex i to vertex j, and $\mathbf{A}[i, j] = 0$ otherwise. Formally, the adjacency matrix $\mathbf{A}$ is defined as:

$$\mathbf{A}_{ij} = \begin{cases} 1 & \text{if } (i, j) \in E \\ 0 & \text{if } (i, j) \notin E \end{cases}$$

where $\mathbf{A}_{ij}$ denotes the element of $\mathbf{A}$ in the i^{th} row and j^{th} column.

Transitioning from the foundational concepts of graph theory to the more specialized domain of max-plus algebra, we acknowledge the complexity and necessity of robust mathematical tools to analyze and optimize the communication networks represented by $\mathcal{G}$ and $\mathbf{A}$. As we have established the notations and interpretations of graphs that will represent the intricate agent networks, it becomes imperative to introduce a suitable algebraic structure that can handle the operations within these networks efficiently. Max-plus algebra emerges as a fitting candidate, offering a unique computational framework that aligns well with the problems inherent in network communications and consensus algorithms. This algebraic system, detailed in the subsequent section, provides the operations and elements that enable a more profound analysis of the temporal dynamics and convergence properties within agent interactions, fundamental to achieving a unified consensus under the constraints of limited communication.

2.2.3 Max-plus algebra

To prove consensus under limited communication, max-plus algebra is utilized [34]. The two operations of particular interest to this work are $a \oplus b = \max(a, b)$ and $a \otimes b = a + b$. Neutral elements in max plus algebra are $-\infty$ for addition (denoted by ε) and 0 for multiplication (denoted by e). These are also referred to as zero and one elements of max plus algebra, respectively. Let us consider two matrices $\mathbf{B} = [b_{ij}] \in \mathbb{R}^{m \times n}$ and $\mathbf{C} = [c_{ij}] \in \mathbb{R}^{n \times q}$. Matrix multiplication in the max plus algebra framework is defined as $(\mathbf{C} \otimes \mathbf{B})_{i,j} = \bigoplus_{k=1}^{n}(c_{ik} \otimes b_{kj}) = \max_k\{c_{ik} + b_{kj}\} \in \mathbb{R}^{m \times q}$. We define the matrix $\mathbf{E}$ using the neutral elements of max-plus algebra as a matrix with all elements equal to e.

The adjacency matrix $\mathbf{A}$ is modified for max-plus algebra as

$$\bar{\mathbf{A}} = [\mathbf{\bar{a}_{ij}}] = \begin{cases} e & \text{for } i = j \text{ and } (i,j) \text{ have an edge} \\ \varepsilon & \text{otherwise} \end{cases} \tag{2.5}$$

This is assuming each agent has information of it's own states. With this we have the tools to transform the non-linear problem of max time consensus to a linear problem in max plus algebra.

2.2.4 Min-plus algebra

Similar to the max-time protocol, we begin by defining the elementary operations in min-plus algebra [35]. Addition in min-plus algebra is defined as $b \oplus c = \min(b, c)$ and multiplication as $b \otimes c = b + c$. It is inferred that the neutral elements in min plus algebra are $+\infty$ for addition and 0 for multiplication, denoted by ε_m and e_m respectively. To illustrate matrix multiplication under min-plus algebra, we define matrices $\mathbf{B} = [b_{ij}] \in \mathbb{R}^{m \times n}$ and $\mathbf{C} = [c_{ij}] \in \mathbb{R}^{n \times q}$. Matrix multiplication is defined as

$(\mathbf{C} \otimes \mathbf{B})_{i,j} = \bigoplus_{k=1}^{n}(c_{ik} \otimes b_{kj}) = \min\limits_{k}\{c_{ik} + b_{kj}\} \in \mathbb{R}^{m \times q}$. $\mathbf{E}_m$ is used to define a matrix of the neutral elements of min-plus algebra, with all its elements equal to e_m.

To perform equivalent operations in min-plus algebra, we define the modified adjacency matrix as

$$\bar{\mathbf{A}}_{\mathbf{min}} = [\bar{\mathbf{a}}_{\mathbf{ij}}] = \begin{cases} e_m & \text{for } i = j \text{ and } (i,j) \text{ connected} \\ \varepsilon_m & \text{otherwise} \end{cases} \tag{2.6}$$

Again, this is assuming each agent has self information of its own state. With this in place, the nonlinear problem of min-time consensus is transformed to a linear one in min-plus algebra.

2.3 Speed Modulation

We begin by examining consensus in time-to-go for single integrator agents. In this section, three different protocols for achieving same-time consensus in max-time, min-time, and achievable-time are established through speed modulation . It is assumed that the agents communicate over a complete graph for the results in this section.

Max-time protocol

We start our discussion with the max-time protocol for the single-integrator dynamics. The single-integrator dynamics of each agent is given as

$$\dot{r}_i(t) = u_i(t) \tag{2.7}$$

which can be expressed equivalently in discrete-time as

$$r_i(k+1) = r_i(k) + u_i(k)\Delta T \tag{2.8}$$

where $k \in \mathbb{Z}_\star$ (with $\mathbb{Z}_\star$ denoting the set of non-negative integers), ΔT is the discretization time step, and $u_i(k) < 0$ denotes the commanded control speed of agent i at time step k in the radial direction. The position disagreement error $\delta_i(k)$ for agent i at time step k is equal to $r_i(k)$, i.e., $\delta_i(k) = r_i(k)$. Thus, we have

$$\delta_i(k+1) = \delta_i(k) + u_i(k)\Delta T \tag{2.9}$$

We define the shared time-to-go estimate for agent i as

$$t_{go,i}(k) = \frac{-\delta_i(k)\Delta T}{\delta_i(k) - \delta_i(k-1)} \tag{2.10}$$

The control speeds, for all the agents $i \in \mathcal{P}$ and at each time step $k \in \mathbb{Z}_\star$, are assumed to be constrained as

$$u_{\min} \leq |u_i(k)| \leq u_{\max} \Rightarrow -u_{\max} \leq u_i(k) \leq -u_{\min}$$

where $u_{\min} > 0$ and $u_{\max} > 0$ are known bounds.

Remark 2.3.1. *Note that we only need to guarantee $\delta_i(k_f) = \epsilon$ for all i at some $k_f \in \mathbb{Z}_\star$. This would suffice for the same-time consensus as any $u_i(k_f) < 0$ will make $\delta_i(k) < \epsilon$ for any $k > k_f$.*

To achieve same-time consensus, the control input for agent i is proposed as

$$u_i(k) = \frac{-\delta_i(k)}{\max\{t_{go,i}(k) : i \in \mathcal{P}\}} \tag{2.11}$$

Next, we state a result that provides an estimate of time to convergence for a specified error tolerance of the final consensus position.

Lemma 2.3.2. *Under the max-time protocol and a given $\epsilon > 0$, there exists a $k_f \in \mathbb{Z}_\star$ such that $\delta_i(k_f) = \epsilon, \forall i \in \mathcal{P}$, with k_f satisfying*

$$\log_{(1-z)}\left(\frac{\epsilon}{\delta_i(0)}\right) < k_f$$

where

$$z = \frac{u_{min}\Delta T}{\delta_{max}(0)}$$

and $\delta_{max}(0) = \max\{\delta_i(0) : i \in \mathcal{P}\}$ *with* $\delta_{max}(0) > u_{min}\Delta T$*.*

Proof. Proof given in Appendix A □

Min-time protocol

The min-time protocol proposes the consensus time-to-go to be set as

$$t_{go,C}(k) = \min\{t_{go,i}(k) : i \in \mathcal{I}\}$$

Next, we state a result similar to the one in Lemma 2.3.2.

Lemma 2.3.3. *Under the min-time protocol and for a specified* $\epsilon > 0$ *and sampling time* ΔT*, there exists a* $k_f \in \mathbb{Z}_\star$ *such that* $\delta_i(k_f) = \epsilon, \forall i \in \mathcal{I}$*, with* k_f *satisfying*

$$\log_{(1-z)}\left(\frac{\epsilon}{\delta_i(0)}\right) > k_f$$

where

$$z = \frac{u_{max}\Delta T}{\delta_{min}(0)}$$

and $\delta_{min}(0) = \min\{\delta_i(0) : i \in \mathcal{I}\}$ *with* $\delta_{min}(0) > u_{min}\Delta T$*.*

Proof. Proof is given in Appendix A □

Remark 2.3.4. *Similar to the max-time protocol, a group of agents dictated by the min-time protocol might not achieve same-time consensus if all of the agents move at their maximum constrained velocities. To illustrate this, let us consider the scenario wherein all agents are traveling at* $u_i(k) = -u_{max}$*. Then, the time-to-go consensus value evaluated at time step* k *is given by*

$$t_{go,C}(k) = \min\{t_{go,i}(k) : i \in \mathcal{I}\} = \frac{\delta_{min}(k)}{u_{max}}$$

where $\delta_{min}(k)$ denotes the relative separation of the agent closest to the destination at k. Then, the commanded control (speed) for agent i at the time step k is given by

$$u_i(k) = -\frac{\delta_i(k)}{\delta_{min}(k)} u_{max}$$

where we have $\delta_i(k) \geq \delta_{min}(k), \forall i \in \mathcal{I}$. Clearly, the commanded control violates the constraint placed on it for any of the agents traveling at their constrained maximum speeds except the one closest to the destination. Thus, if the agents move at their maximum constrained speed, then same-time consensus can not be achieved.

Achievable-time Protocol

The analysis of the min-time protocol reveals a limitation: if agents operate at their maximum constrained speeds, the protocol may fail to achieve same-time consensus. This is due to the stringent synchronization requirements which may not be met under certain speed constraints. Conversely, an analogous issue arises with the max-time protocol when agents travel at their minimum constrained speeds. In such cases, the protocol's conservative approach can preclude the realization of same-time consensus. These observations underscore the necessity for a protocol that can adapt to the varying speed limitations of agents to ensure consensus is attainable.

This can be inferred from observing both protocols applied to identical agents initialized randomly. The max-time protocol when applied uninterruptedly converges to the estimate of time-to-go for the agent that takes longest to rendezvous whereas the min time-to-go protocol converges to the time-to-go estimate of the agent that has the shortest estimate of time-to-go.

To find a trade off between achieving consensus faster and ensure consensus to the target is achieved among the agents, we propose the achievable-time protocol in this paper. Towards this end, we first need to define critical time-to-go values for the agents. Critical time-to-go values are defined as the estimated time-to-go for an agent assuming they travel at their constrained speeds. For an agent $i \in \mathcal{I}$, the critical time-to-go value at time step k is defined as

$$t_{cr,i}^{min}(k) = \frac{\delta_i(k)}{u_{\max}}$$

To execute the protocol, the agents in $\mathcal{I}$ share not only their time-to-go estimates, but also a Boolean flag. When the time-to-go estimate for an agent $i \in \mathcal{I}$ is equal to $t_{cr,i}^{min}(k)$, the flag is set to 1 or 'True'. Based on the information being shared, the agents will follow the min-time protocol unless there is an agent or a group of agents with their shared flag equal to 'True', i.e., agents only appear in $\mathcal{M}$ iff their time-to-go is equal to the critical time-to-go value. Let $\mathcal{M} \subseteq \mathcal{I}$ be the subset of agents with their shared flag set to True. If or when $\mathcal{M} \neq \emptyset$, the consensus value for the entire group is set to the maximum time-to-go value of the agents in $\mathcal{M}$.

The protocol is initialized similar to the min-time protocol. For the latter part of the protocol, it can be shown that consensus can be achieved among the agents in $\mathcal{M}$. Then, the remaining task is to ensure that all the other agents in $\mathcal{I}$ achieve consensus with the agents in $\mathcal{M}$. Let $\mathcal{M} \neq \emptyset$ and let $\mathcal{L} = \mathcal{I} \backslash \mathcal{M}$.

Assumption 2.3.1. *The commanded speeds for the agents in $\mathcal{L}$ satisfy $|u_i(k)| \geq u_{min}, \forall i \in \mathcal{L}, k \in \mathbb{Z}_\star$ and$\delta_{min} > u_{min}\Delta T_c$.*

Theorem 2.3.5. *Achievable Time Consensus: Under Assumption 2.3.1, all the agents in $\mathcal{I}$ implementing the achievable-time protocol will achieve same-time consensus.*

Proof. See the Appendix. □

Remark 2.3.6. *It is important to note that same-time consensus may not be possible if the commanded speed $|u_{i_L}(k)| < u_{min}$ for some $k \in \mathbb{Z}_\star$. However, this can be mitigated by having the default protocol when $\mathcal{M} = \emptyset$ being the min-time protocol* *which speeds up the agents. Also, it should be noted that the achievable-time protocol will not work if the communication graph between the agents is incomplete.*

2.4 Main Results

To obtain the chief results pertaining to the same-time consensus problem posed for systems with double integrator dynamics, we utilize the following developments. We assume that speed can be changed at discrete instances in time, which we refer to as speed modulation in this research. In a manner similar to the backstepping control strategy [36], our solution methodology consists of following two parts:

1. Speed modulation: We use discrete-time protocols for evaluating the consensus time-to-go among the agents for speed modulation.
2. Continuous-time error convergence: Control is synthesized to ensure the errors between protocol-dependent consensus time-to-go and the dynamic local agent time-to-go estimate converge to zero. We also require that the error converges to zero between any two successive consensus evaluations. To achieve that, we use a finite-time approach for the control synthesis.

The speed modulation is obtained through formulating the problem using single integrator dynamics.

Let the successive evaluations of consensus occur at time intervals of ΔT_c.

Further, let $k_c \in \mathbb{Z}_\star$ denote the time steps at which consensus evaluation occurs.

We define the position disagreement error for an agent i as in Section 2.3, i.e., $\delta_i(t) = r_i(t)$. Additionally, a time-to-go estimate for an agent is chosen, similar to the one in (2.10), as

$$t_{go,i}(t) = -\frac{\delta_i(t)}{u_i(t)} \tag{2.12}$$

Now, the error in time-to-go for agent $i \in \mathcal{I}$ is defined as

$$\eta_i(t) = t_{go,i}(t) - t_{go,C}(t) \tag{2.13}$$

For a successful same-time consensus, we need to satisfy $\eta_i(t) = 0, \forall i \in \mathcal{I}$ in finite time, and we do so by designing the acceleration inputs appropriately. Moreover, along with the update in consensus time-to-go estimate at every $k_c \in \mathbb{Z}_\star$ (i.e., $t = k_c \Delta T_c$), we update the speeds of all the agents based on the results in Section 2.3, depending upon the particular protocol implemented.

Now, upon differentiating the estimate of time-to-go with respect to time, we derive

$$\dot{t}_{go,i}(t) = \left(\frac{r_i}{u_i^2}\right) a_i - 1$$

For successful same-time consensus, we need to satisfy $\eta_i(t) = 0, \forall i \in \mathcal{I}$ in finite time, and we do so by designing the acceleration inputs appropriately. Moreover, along with the update in consensus time-to-go estimate at every $k_c \in \mathbb{Z}_\star$ (i.e., $t = k_c \Delta T_c$), we update the speeds of all the agents based on the results in Section 2.3, depending upon the particular protocol implemented. We define the error in time-to-go as

$$\eta_i(t) = t_{go,i}(t) - t_{go,C}(t) \tag{2.14}$$

Hence, the error dynamics in time-to-go is observed to be [31]

$$\dot{\eta}_i(t) = 1 + \dot{t}_{go,i}(t) = \left(\frac{r_i}{u_i^2}\right) a_i \tag{2.15}$$

For synthesizing a_i such that η_i is driven to zero in finite time, let us consider the following positive definite candidate Lyapunov function for agent $i \in \mathcal{I}$

$$V_i = \frac{1}{2}\eta_i^2$$

First derivative of V_i with respect to time gives us

$$\dot{V}_i = \eta_i \dot{\eta}_i = \eta_i(t) \left(\frac{r_i}{u_i^2} \right) a_i \tag{2.16}$$

To cancel nonlinear terms and effect finite time convergence, control (the acceleration term) is chosen as

$$a_i = -\frac{u_i^2}{r_i} \left(\kappa_{i,1} \eta_i(t) + \kappa_{i,2} \left(\eta_i(t) \right)^{\alpha} \right) \tag{2.17}$$

where $\kappa_{i,1}, \kappa_{i,2} > 0$ are the control gains and $\alpha = \frac{p}{q} \in (0, 1)$ $(p, q(q > p)$ are positive odd integers) is a control parameter which induces finite time convergence [32]. Using the above expression of a_i, (2.16) can be expressed as

$$\dot{V}_i = -2\kappa_{i,1} V_i - 2^{(1+\alpha)/2} \kappa_{i,2} V_i^{(1+\alpha)/2} \tag{2.18}$$

Clearly, we have $\dot{V}_i < 0$. Hence, V_i is a Lyapunov function for the error dynamics of agent $i \in \mathcal{I}$, and the error dynamics of the closed loop system is globally asymptotically stable (since V_i is radially unbounded).

We notice that through the use of a_i, we obtain the error dynamics in time-to-go to be of the form described in 2.2.1. Hence, with a choice of ΔT_c we can ascertain that the time-to-go estimates will converge to the consensus value of time-to-go thereby modulating their speeds.

Substituting a_i in (2.15), we obtain

$$\dot{\eta}_i(t) = -\kappa_{i,1} \eta_i(t) - \kappa_{i,2} \left(\eta_i(t) \right)^{\alpha} \tag{2.19}$$

which is clearly in the form given in (2.3) and the above result can be utilized to conclude that η_i goes to zero in finite time. Now, we require the error in time-to-go estimate to be zero in at most ΔT_c. This would require $t_s \leq \Delta T_c$ which can be achieved by a proper selection of the parameters $\kappa_{i,1}$ and $\kappa_{i,2}$ as described next. For a desired maximum value of t_s ($\leq \Delta T_c$) and an arbitrary value of $\kappa_{i,1} > 0$, $\kappa_{i,2}$ can be determined using

$$\kappa_{i,2} = \kappa_{i,1}|\eta_{\max}(0)|^{(1-\alpha)}(e^{\kappa_{i,1}(1-\alpha)t_s} - 1)^{-1} \tag{2.20}$$

where $|\eta_{\max}(0)| = \max\{|\eta_i(0)| : i \in \mathcal{I}\}$ to account for the limiting value of the time of convergence among the agents. For a detailed discussion on how the parameters $\kappa_{i,1}$, $\kappa_{i,2}$, α affect convergence, see [32].

Remark 2.4.1. *Since the error in time-to-go converges to zero in at most ΔT_c, at every consensus evaluation time step k_c the agents' speeds are set equal to*

$$u_i(k_c\Delta T_c) = \frac{-\delta_i(k_c\Delta T_c)}{T_f(k_c)} = \frac{-\delta_i(k_c\Delta T_c)}{t_{go,C}(k_c\Delta T_c)} \tag{2.21}$$

which is the same as the expression in for the single integrator speed modulated system described in Section 2.3. Hence, the group of agents governed by the double-integrator dynamics would achieve same-time consensus using the proposed protocols.

Remark 2.4.2. *Under any specified protocol, consensus in time-to-go estimates is said to be achieved for a network of N double integrator agents if the error between time-to-go estimates ($t_{go,i}$) and the protocol dependent time-to-go consensus value ($t_{go,C}$) is zero for all agents , i.e*

$$\eta_i(t) = t_{go,i}(t) - t_{go,C}(t) = 0 \quad \forall i \in \{1, 2, ..N\}$$

Control synthesized using (2.17) *gives us that for a complete graph network of connected vehicles, $\eta_i \rightarrow$ in at most ΔT_c which can be verified using Lemma 2.2.1.*

Corollary 2.4.2.1. *For time-to-go consensus achieved at time step k_s, the time-to-go estimates($t_{go,i}$) have converged to the protocol dependent time-to-go consensus value($t_{go,C}$). Using the definition* (2.2)*,*

$$t_{go,i}[(k_s + 1)\Delta T_c] = t_{go,C}(k_s\Delta T_c) - \Delta T_c = t_{go,i}(k_s\Delta T_c) - \Delta T_c \tag{2.22}$$

The consensus in time-to-go does not break until position convergence with the intersection point is achieved.

Corollary 2.4.2.2. *On application of these protocols, it is evident that there must be at least two consensus evaluation time steps before any of the agents achieve position consensus with the intersection point. One interval for the consensus time to go values to be established and another for the time-to-go estimates to converge to the established consensus value of time-to-go. This gives us that*

$$\Delta T_c \leq \frac{1}{2} \min\{t_{go,i}(0) : i \in \mathcal{I}\}$$

We illustrate our framework of consensus and convergence in Fig. 2.2

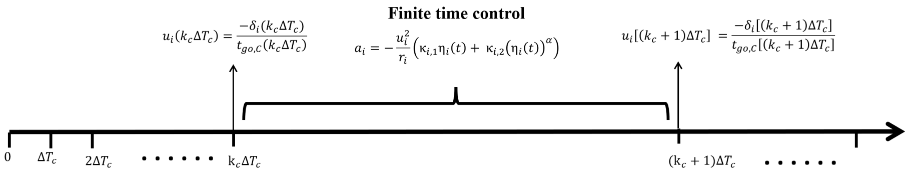

Finite time control
$u_i(k_c\Delta T_c) = \frac{-\delta_i(k_c\Delta T_c)}{t_{go,C}(k_c\Delta T_c)}$
$a_i = -\frac{u_i^2}{r_i}\left(\kappa_{i,1}\eta_i(t) + \kappa_{i,2}(\eta_i(t))^\alpha\right)$
$u_i[(k_c+1)\Delta T_c] = \frac{-\delta_i[(k_c+1)\Delta T_c]}{t_{go,C}[(k_c+1)\Delta T_c]}$
0
ΔT_c
$2\Delta T_c$
......
$k_c\Delta T_c$
$(k_c+1)\Delta T_c$
......

Figure 2.2. Continuous-discrete framework of consensus.

Algorithm 1 Max-Time Protocol

```
Initialization: $\kappa_{i,1}, \kappa_{i,2}$ are chosen based on (2.20)
while $r_i(k) > \epsilon, \forall i \in \mathcal{I}$ do
    Calculate $t_{go,i}(k)$ for all agents
    $t_{go,C}(k) = \max\{t_{go,i}(k)\}, \forall i \in \mathcal{I}$
    Control Term $a_i$ is calculated for each agent
    $k \leftarrow k + 1$
end while
```

2.4.1 Max-time protocol

Under the max-time protocol, the consensus value of time-to-go at any consensus time step $k_c \in \mathbb{Z}_\star$ (or at $t = k_c \Delta T_c$) is given by

$$T_f(k_c) = \max\{t_{go,i}(k_c \Delta T_c) : i \in \mathcal{I}\}$$

The number of consensus evaluations required may be calculated using Lemma 2.3.2. Implementation of the max-time protocol is summarized in Algorithm 1.

Algorithm 2 Min-Time Protocol

```
Initialization: $\kappa_{i,1}, \kappa_{i,2}$ are chosen based on (2.20).
while $r_i(k) > \epsilon, \forall i \in \mathcal{I}$ do
    Calculate $t_{go,i}(k)$ and $t_{cr,i}^{min}(k)$ for all agents
    if $t_{go,i}(k) = t_{cr,i}^{min}(k)$ then
        flag = 1                    ▷ convergence may not be achieved
    end if
    $t_{go,C}(k) = \min\{t_{go,i}(k)\}, \forall i \in \mathcal{I}$
    Control Term $a_i$ is calculated for each agent
    $k \leftarrow k + 1$
end while
```

2.4.2 Min-time Protocol

Under the max-time protocol, the consensus value of time-to-go at any consensus time step $k_c \in \mathbb{Z}_\star$ (or at $t = k_c \Delta T_c$) is given by

$$T_f(k_c) = \min\{t_{go,i}(k_c \Delta T_c) : i \in \mathcal{I}\}$$

The number of consensus evaluations required may be calculated using Lemma 2.3.3. Implementation of the min-time protocol is summarized in Algorithm 2.

2.4.3 Achievable-time Protocol

We have shown consensus to occur in both the min-time and max-time protocols. The achievable-time protocol can be shown to achieve consensus through Theorem 2.3.5. The algorithm to synthesize control a_i to achieve same-time consensus using the achievable-time protocol is shown in Algorithm 3.

Algorithm 3 Achievable-Time Protocol

```
**Initialization:** $\kappa_{i,1}, \kappa_{i,2}$ are chosen based on (2.20)
while $r_i(k) > \epsilon, \forall i \in \mathcal{I}$ do
    Calculate $t_{go,i}(k)$ and $t_{cr,i}^{min}(k)$ for all agents
    if $t_{go,i}(k) = t_{cr,i}^{min}(k)$ then
        $flag(i) \leftarrow 1$
    else
        $flag(i) \leftarrow 0$
    end if
    if $any(flag) = 1$ then
        if $flag(i) = 1$ then
            $\mathcal{M} \leftarrow (I)_i$
        end if
        $t_{go,C}(k) = \max\{t_{go,i}(k)\}, \forall i \in \mathcal{M}$
    else
        $t_{go,C}(k) = \min\{t_{go,i}(k)\}, \forall i \in \mathcal{I}$
    end if
    Control Term $a_i(k)$ is calculated for each agent
    $k \leftarrow k + 1$
end while
```

2.5 Consensus Under Limited Communication

2.5.1 Max time protocol

It has been assumed until now that time-to-go estimates are communicated directly between any two agents in the network. Similar to our approach in [29], let ΔT_c denote the interval of time between successive consensus evaluations. We utilize directed graphs to represent the communication in the network of agents. Let $\mathcal{G} = (\mathcal{I}, \mathcal{E})$ represent the communication graph for the N agents' network with $\mathcal{I} = \{1, 2, \ldots, N\}$ being the non empty set of nodes representing the N agents and $\mathcal{E}$ being the set of ordered pairs termed as edges. Their adjacency matrix is given by $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$. Under this assumption, the consensus value of time-to-go of each agent is contingent on the nodes connected to the agent. We define a matrix of time-to-go values of all the agents at time t as

$$\mathbf{T}_{go}(t) = \begin{cases} \text{diag}(t_{go,i}(t)) & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases}$$

The time-to-go information available to an agent i in the network is given by the ith row of the matrix product $\mathbf{AT}_{go}$. Hence, the consensus value of time-to-go will vary locally. From graph theory [37], we know that a connected graph topology is one where a path or sequence of edges in $\mathcal{E}$ exists between any two agents in the graph. We also define a simple path of a graph as a sequence of edges where no node repeats twice. It is evident from the definition of connected graphs that a simple path exists in a connected graph.

The vector of time-to-go consensus values, $\mathbf{t_{go,C}}(k\Delta T_c)$, for this protocol is given by

$$\mathbf{t_{go,C}}(k\Delta T_c) = \max_{i \,\in\, RS(\mathbf{AT}_{go})} [\mathbf{AT}_{go}(k\Delta T_c)]_{ij} \tag{2.23}$$

where $[\cdot]_{i,j}$ stands for the (i,j) element of a matrix and $RS(\cdot)$ denotes the row space of a matrix. The commanded speed of agent i for the system to achieve same-time consensus using the max-time protocol is given as

$$u_i(k\Delta T_c) = \frac{-\delta_i(k\Delta T_c)}{[\mathbf{t_{go,C}}(k\Delta T_c)]_i} \tag{2.24}$$

where $\delta_i(k\Delta T_c)$ is the disagreement error at consensus evaluation time step k for agent i and $[\mathbf{t_{go,C}}(k\Delta T_c)]_i$ is the ith element of the vector $\mathbf{t_{go,C}}(k\Delta T_c)$.

Theorem 2.5.1. *Consider a group of N agents with double integrator dynamics and a communication graph $\mathcal{G}$ with 'p' as the maximum length of a simple path of the graph. If the group executes the max-time protocol, the following holds:*

1. *Sufficient conditions for time-to-go consensus are:*
   - *(a) The graph $\mathcal{G}$ must be connected,*
   - *(b) $\max\limits_{i\in\mathcal{I}}([\mathbf{t_{go}}(0)]_i) \geq p\Delta T_c$, where $\mathbf{t_{go}}(\cdot)$ is the vector of time-to-go estimates of the agents .*
2. *Consensus values of time-to-go estimates are given by*

$$\begin{aligned}&[\mathbf{t_{go,C}}(k_c\Delta T_c)]_i \\ &= \max_{j\in\mathcal{I}}([\mathbf{t_{go}}(k_c\Delta T_c)]_j) \quad \forall i\in\mathcal{I},\ k_c\geq p\end{aligned} \tag{2.25}$$

where $\mathcal{I} = \{1, 2, \ldots, N\}$.

Proof is given in AppendixA.4.

With the error η_i between $t_{go,i}$ and $t_{go,C}$ being driven to zero through the control acceleration given by (2.17), Theorem 2.5.1 gives us that the max-time protocol applied to a connected graph commands that the speed modulated by an agent at the consensus evaluation time step $k \geq N$ is given as

$$u_i(k\Delta T_c) = \frac{-\delta_i(k\Delta T_c)}{\max\limits_{j\in\mathcal{I}}([\mathbf{t_{go}}(k_c\Delta T_c)]_j)} \tag{2.26}$$

Using the equivalence of (2.26), (2.21), and (2.24), we conclude that the system of double integrators employing the max-time protocol achieves same-time consensus. Thus, Lemma 2.3.2 can be used to calculate the number of time-to-go consensus time steps until position convergence. With this we extend the max time protocol to be applicable to a connected graph topology as opposed to a complete graph communication topology.

2.5.2 Min time protocol

The control law that leverages consensus to the minimum value of the time-to-go shared between agents is studied. To model the network connections, we re-define the adjacency matrix for the min-time protocol as $\mathbf{A_m} = [a_{ij}] \in \mathbb{R}^{N\times N}$ with a_{ij} denoting the weight of the edge from agent j to agent i. This value is ∞ if no such edge exists. For practical purposes, this is modeled as a very large positive number.

The consensus values of time-to-go can be expressed as a vector $\mathbf{t_{go,C}} \in \mathbb{R}^{N\times 1}$ given by

$$\mathbf{t_{go,C}}(k\Delta T_c) = \min_{i \,\in\, RS(\mathbf{A_m}\mathbf{T}_{go})} [\mathbf{A_m}\mathbf{T}_{go}(k\Delta T_c)]_{ij} \tag{2.27}$$

The commanded speed of agent i for the system to achieve same-time consensus through the min-time protocol is obtained using (2.24).

Theorem 2.5.2. *Consider a group of N agents with double integrator dynamics and a communication graph $\mathcal{G}$. If the group executes the min-time protocol, the following holds:*

1. *Sufficient conditions for time-to-go consensus are:*
   - *(a) The graph $\mathcal{G}$ must be connected,*
   - *(b) $\min_{i\in\mathcal{I}} ([\mathbf{t_{go}}(0)]_i) \geq N\Delta T_c$, where $\mathbf{t_{go}}(\cdot)$ is the vector of time-to-go estimates of the agents.*

2. Consensus values of time-to-go estimates are given by

$$\begin{aligned}&[\mathbf{t_{go,C}}(k_c\Delta T_c)]_i \\ &= \min_{j\in\mathcal{I}}\left([\mathbf{t_{go}}(k_c\Delta T_c)]_j\right) \quad \forall i\in\mathcal{I},\ k_c\geq N\end{aligned} \tag{2.28}$$

where $\mathcal{I}=\{1,2,\ldots,N\}$.

Proof is given in Appendix A.5.

Using Theorem 2.5.2 and (2.2) for a connected graph, we obtain

$$\begin{aligned}&[\mathbf{t_{go,C}}(k\Delta T_c+m)]_i \\ &= \min_{i\in\mathcal{I}}([\mathbf{t_{go}}(k\Delta T_c)]_i)-m \quad \forall k\geq N, m\in[0,\Delta T_c)\end{aligned}$$

The consensus value of time-to-go at the consensus evaluation time step k is the minimum of all the time-to-go estimates for the agents in $\mathcal{I}$. With the error η_i between $t_{go,i}$ and $t_{go,C}$ being driven to zero through the control acceleration given by (2.17), the commanded speed can now be written as

$$u_i(k\Delta T_c)=\frac{-\delta_i(k\Delta T_c)}{\min_{j\in\mathcal{I}}\left([\mathbf{t_{go}}(k_c\Delta T_c)]_j\right)} \tag{2.29}$$

Using the equivalence of (2.29), (2.21) and (2.24), we conclude that the system of double integrators employing the min-time protocol achieves same-time consensus and, therefore, we can use Lemma 2.3.3 to calculate the number of consensus time steps until position convergence. Next we look at a modification to the achievable time protocol.

2.5.3 Tokenized achievable-time protocol

The achievable-time protocol achieves same-time position consensus only if the graph communication between the agents is a complete graph. The protocol has been modified to achieve consensus for cyclic connected graphs through a passable distress token. The achievable-time protocol is characterized by a Boolean flag and $t_{cr,i}^{min}$, given by

$$t_{cr,i}^{min}(k\Delta T_c) = \frac{-\delta_i(k\Delta T_c)}{u_{\max}} \tag{2.30}$$

Instead of a Boolean flag, for cyclic graphs, we generate a passable distress '*token*'. If the time-to-go consensus value of an agent i is less than the critical time-to-go value for an agent, a distress token is generated and broadcast along the cycle. The value of the token is set to the critical time-to-go value of agent i. This token is then passed to the next agent in the cycle $i+1$. The agent $i+1$ now uses this token to determine an achievable consensus value of time-to-go as

$$[\mathbf{t_{go,C}}(k\Delta T_c + p)]_{i+1} = \max\{token_i, [\mathbf{t_{go,C}}(k\Delta T_c)]_{i+1}\} - p$$

where $p \in [0, \Delta T_c]$. The value of the token must now be determined based on which of the consensus time-to-go values between agent i and $i+1$ is achievable for both before being passed to the next agent in cycle. The value of the token is determined as

$$token_{i+1} = \max\{token_i, [\mathbf{t_{go,C}}(k\Delta T_c)]_i\}$$

It is evident that the token needs to be passed around the cycle once for the consensus value of time-to-go for an agent in $\mathcal{I}$ to be equal to the maximum of the time-to-go consensus values ($[\mathbf{t_{go,C}}]_j$) of the agents $j \in \mathcal{M}$. This would ensure that the agents set their consensus value of time-to-go to the fastest achievable time-to-go as opposed to the largest or a possibly unattainable smallest value of the consensus time-to-go.

Remark 2.5.3. *This protocol can be extended to any connected graph by passing the token from one end of its spanning tree to the other and cycling the token back to the initial node, thereby obtaining a sufficient condition as* $\min_{i\in\mathcal{I}} ([\mathbf{t_{go}}(0)]_i) \geq 2N\Delta T_c$.

Similar to the regular achievable-time protocol, when $\mathcal{M} = \emptyset$, the agents execute the min-time protocol which can be shown to achieve same-time consensus using Theorem 2.5.2.

Now, consider a time step k when $\mathcal{M} \neq \emptyset$, i.e., the min-time protocol has been applied till some arbitrary time step k at which point $\mathcal{M} \neq \emptyset$. Hence, at this time step k, a $token_i$ is circulated from agent i. The consensus value of time-to-go becomes

$$\max\{[\mathbf{t}_{go,C}(k\Delta T_c)]_i, token_i\} = \frac{-\delta_{i_M}(k\Delta T_c)}{u_{\max}}$$

where the subscript i_M denotes the agent in $\mathcal{M}$ with the token.

Now, for any arbitrary agent $i_L \in \mathcal{L}$, the commanded control is given by

$$u_{i_L}(k\Delta T_c) = \frac{-\delta_{i_L}(k\Delta T_c)}{\delta_{i_M}(k\Delta T_c)} u_{\max}$$

If the token is circulated throughout the network, it is evident that i_M denotes the agent in $\mathcal{M}$ with the largest value of estimated time-to-go in $\mathcal{M}$ (in other words, largest $\delta_i(k)$ among all the agents in $\mathcal{M}$). Since the group was initialized with the min-time protocol, we can conclude that $\delta_{iL}(k) < \delta_{iM}(k)$. This gives us that

$$|u_{i_L}(k)| = \frac{\delta_{i_L}(k)}{\delta_{i_M}(k)} u_{\max} < u_{\max}$$

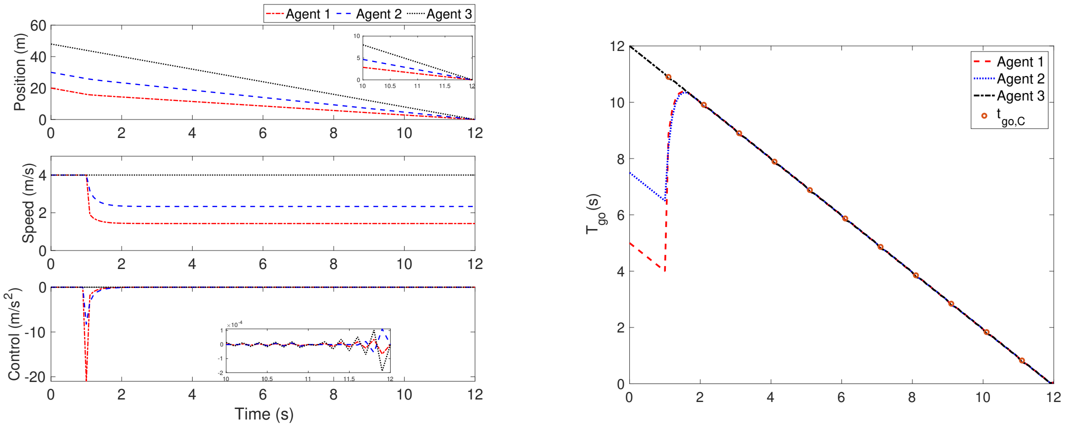
Also, under Assumption 2.3.1, we have $|u_{i_L}(k)| \geq u_{\min}$. This would imply that all agents in $\mathcal{L}$ can achieve convergence to the target through the tokenized achievable-time protocol. Further, using Lemma 2.3.2 it can be shown that all agents in $\mathcal{M}$ and hence all agents in $\mathcal{I}$ achieve same-time consensus through the tokenized achievable-time protocol. Implementation of the tokenized achievable-time protocol is summarized in Algorithm 4.

Algorithm 4 Tokenized Achievable-Time Protocol

```
**Initialization:** Choose $\kappa_{i,1}, \kappa_{i,2}$ based on (2.20). Communication network is characterized by the adjacency matrix $\mathbf{A}$.
while $r_i(k) > \epsilon, \forall i \in \mathcal{I}$ do
    Calculate $t_{go,i}(k)$ and $t_{cr,i}^{min}(k)$ for all agents.
    if $t_{go,i}(k) = t_{cr,i}^{min}(k)$ then
        $token_i = t_{go,i}$
    else
        $token_i = 0$
    end if
    if $token > 0$ then
        $T_{f,i} = \max\{token_i, [\mathbf{t}_{go,C}]_i\}$
        $i \leftarrow i + 1, i \in \mathcal{I}$
    else
        $t_{go,C}(k) = \min_{i \,\in\, RS(\mathbf{AT}_{go})} [\mathbf{AT}_{go}(k)]_{ij}$
    end if
    Calculate the control term $a_i(k)$ for each agent.
    $k \leftarrow k + 1$
end while
```

2.6 Simulation Results

The simulation results for the proposed protocols are described in this section. Note that the acceleration is assumed to be 0 m/s^2 for the first time step for all the simulation results. Also, for the same-time consensus in the sense of Definition 2.1.1, we have selected $\epsilon = 1$ m. Unless otherwise specified, we have used sampling time $\Delta T_s = 0.1$ s and $\Delta T_c = 1$ s for all the results.



(a) Position, speed, and acceleration

(b) Time-to-go estimates

Figure 2.3. Max-time protocol results.

2.6.1 Agents with a complete communication graph

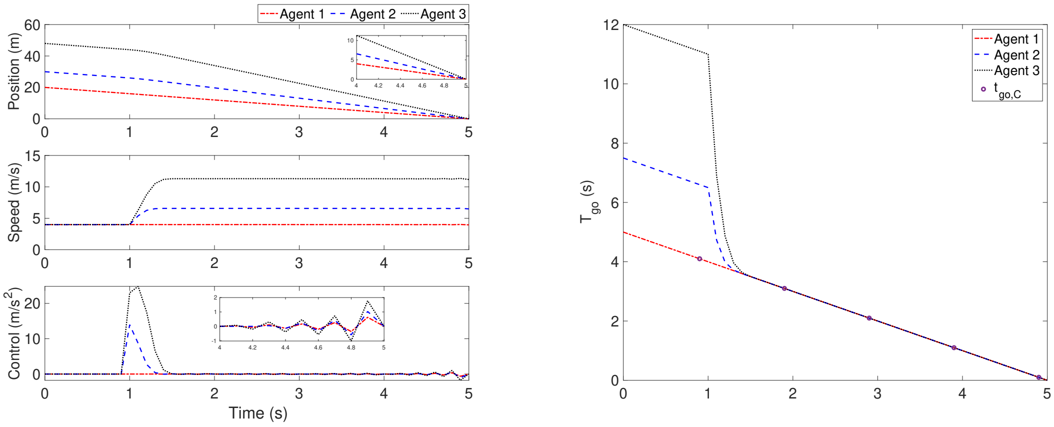
First, we illustrate application of the proposed protocols for the rendezvous of multiple UAVs and consider three agents having initial conditions as follows:

$$\boldsymbol{\delta}(0) = \begin{bmatrix} \delta_1(0) \\ \delta_2(0) \\ \delta_3(0) \end{bmatrix} = \begin{bmatrix} 20 \text{ m} \\ 30 \text{ m} \\ 48 \text{ m} \end{bmatrix}, \ \mathbf{u}(0) = \begin{bmatrix} u_1(0) \\ u_2(0) \\ u_3(0) \end{bmatrix} = \begin{bmatrix} 4 \text{ m/s} \\ 4 \text{ m/s} \\ 4 \text{ m/s} \end{bmatrix}$$

The time interval between successive consensus evaluations is $\Delta T_c = 1$ s. The control gains are chosen as: $\kappa_{i,1} = 5$ and $\kappa_{i,2} = 1.125$ (calculated using (2.20) with $t_s = \Delta T_c$).

2.6.1.1 Max-time protocol

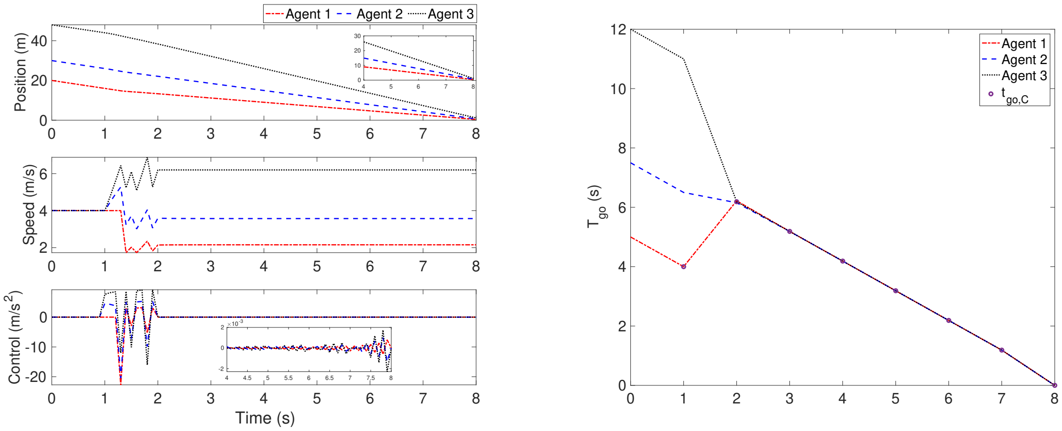
The simulation results for this scenario are shown in Figs. 2.3(a), 2.3(b). We notice that the agents converge their time-to-go to the slowest agent (Fig. 2.3(b)). The agents achieve convergence to within $\epsilon = 1$m of the target at the same time which is ascertained from Figs. 2.3(a). On inspecting Fig. 2.3(a) further, we notice that the agent furthest away does not show as much control activity as the other agents. The other agents slow down significantly to ensure consensus through the max-time protocol. This is due to the agent furthest away having the largest time-to-go estimate among the agents.



(a) Position, speed, and acceleration

(b) Time-to-go estimates

Figure 2.4. Min-time protocol results.



(a) Position, speed, and acceleration

(b) Time-to-go estimate

Figure 2.5. Achievable-time protocol results.

2.6.1.2 Min-time protocol

The simulation results are shown in Figs. 2.4(a), 2.4(b). The agents converge their time-to-go with the fastest agent as shown in Fig. 2.4(b). Consensus is ascertained as the states converge to within $\epsilon = 1$ m of the target at the same time as inferred from Figs. 2.4(a). We also notice that the agent closest to the target does not show much control activity, instead has the other agents speed up significantly to ensure consensus through the min-time protocol. This is due to the closest agent having the smallest value of time-to-go estimates among the agents.

2.6.1.3 Achievable-time protocol

From Remark 2.3.4 we infer that if the commanded speed from control is greater than the constrained speed, the min-time protocol might not be successful. To illustrate the advantage of the achievable-time protocol, we impose a constraint on the maximum commanded speed to be $u_{\max} = 6$ m/s. The agents converge to within $\epsilon = 1$ m of the target at the same time despite the constraint, as shown in Figs.

2.5(a), 2.5(b). Since the constraint specified is lesser than the speed commanded by the min-time protocol, the consensus value of time-to-go when using the achievable-time protocol converges to the agent with the most achievable time-to-go of the agents traveling at their constrained speeds as shown in Fig. 2.5(a),2.5(b).

Remark 2.6.1. *From the Lemmas 2.3.2 and 2.3.3, we notice that the time interval between successive consensus evaluations of time-to-go ΔT_c must be less than the minimum of the initial time-to-go values. In other words, there must be at least one consensus evaluation before any of the agents reach the target location.*

For the more general case of a connected graph as opposed to a complete graph discussed in Section 2.5, the lower bound on the time interval between consensus evaluations is increased by a factor of the number of agents (N). We must analyze the sensitivity of these algorithms to a change in ΔT_c. Immediately it is easy to see that due to the nature of our framework, since the errors in time-to-go require at least ΔT_c to settle to 0, this would imply

$$\Delta T_c \leq \frac{1}{2} \min\{t_{go,i}(0) : i \in \mathcal{I}\} \tag{2.31}$$

To illustrate this, we choose ΔT_c equal to 0.5, 1, 2.5 and 3 s, and simulate the various protocols for the rendezvous problem under consideration. The results of this study are summarized in Table 2.1. For a choice of $\Delta T_c > 2.5$ s, we notice that the protocols fail.

From initial conditions chosen with $\Delta T_c = 3$ s, agent 1 at the first evaluation would have a time-to-go estimate of 2 s. This would not be enough time for both a time-to-go consensus and speed modulation for same-time consensus, and hence the protocols fail at this choice of ΔT_c.

Table 2.1. Effect of choice of ΔT_c on same-time consensus

Protocol	ΔT_c	Consensus
Max-time	0.5	Achieved
	1	Achieved
	2.5	Achieved
	3	Failed
Min-time	0.5	Achieved
	1	Achieved
	2.5	Achieved
	3	Failed
Achievable-time	0.5	Achieved
	1	Achieved
	2.5	Achieved
	3	Failed

2.6.2 Three agents over a directed cycle

We simulate the max-time and the min-time protocols for three agents connected in a directed cycle. The graph communication topology is as shown in Fig. 2.6. We notice that the graph communication is indeed connected and a simple path exists between the three nodes. Here we consider the salvo problem, and the initial

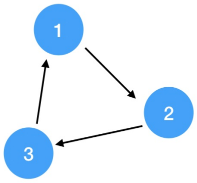
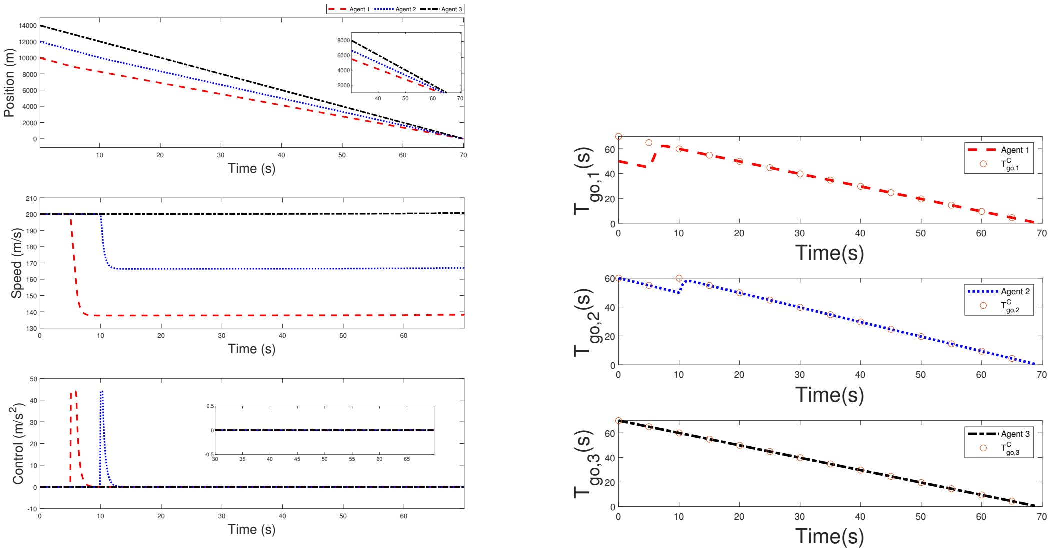


Figure 2.6. Graph connectivity (three agents).

conditions of the agents are given by

$$\boldsymbol{\delta}(0) = \begin{bmatrix} \delta_1(0) \\ \delta_2(0) \\ \delta_3(0) \end{bmatrix} = \begin{bmatrix} 10 \text{ km} \\ 12 \text{ km} \\ 14 \text{ km} \end{bmatrix}, \ \mathbf{u}(0) = \begin{bmatrix} u_1(0) \\ u_2(0) \\ u_3(0) \end{bmatrix} = \begin{bmatrix} 200 \text{ m/s} \\ 200 \text{ m/s} \\ 200 \text{ m/s} \end{bmatrix}$$

We choose $\kappa_{i,1} = 2$, and $\kappa_{i,2} = 0.61$ (again calculated using (2.20) with $t_s = \Delta T_c$). The time to go consensus interval(ΔT_c) is 5 sec.



(a) Position, speed, and acceleration

(b) Time-to-go estimate

Figure 2.7. Max-time protocol results (three agents).

2.6.2.1 Max-time protocol

The agents achieve position consensus with the max-time protocol, as evidenced by the results shown in Fig. 2.7(a). Despite the lack of a directed edge from agent 1 to agent 3, from Fig. 2.7(b) we notice that the consensus value of time-to-go for each agent eventually converges to the maximum of the time-to-go estimates of the agents and individually, each of these estimates converge to zero at the same time implying that same time position consensus with the target is achieved.

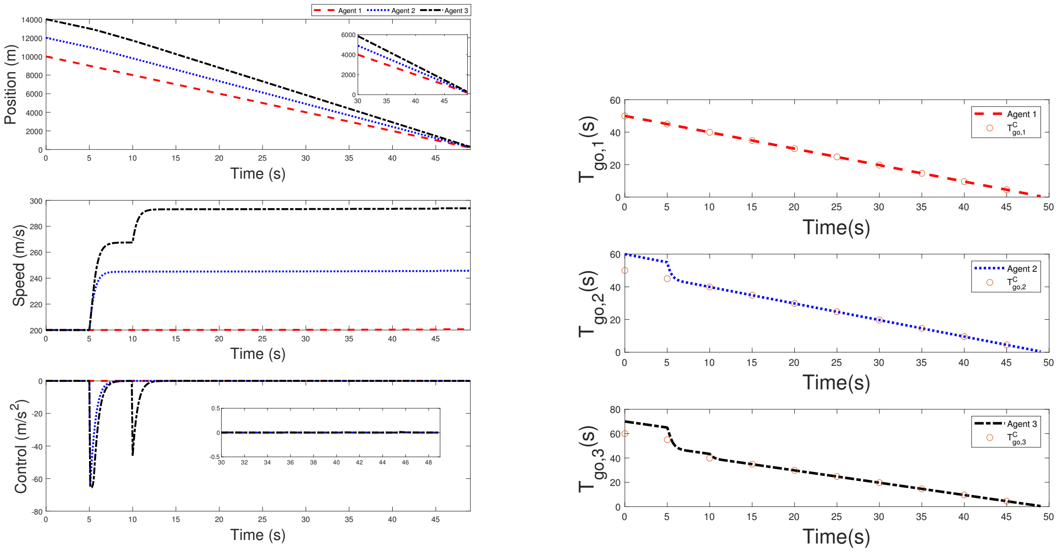
2.6.2.2 Min-time protocol

It is evident from Fig. 2.8(a) that the agents achieve position consensus with the target through the use of the min-time protocol. We notice that the agents move faster and achieve consensus in position quicker than the max-time protocol. From Fig. 2.8(b), we notice that the time-to-go consensus value for each agent converges to the minimum of the time-to-go estimates of all the agents which eventually converges to zero for every agent at the same time, thereby showing same-time position consensus for agents executing the min-time protocol. Figure 2.8(a) also illustrates how the min-time protocol commands large values of commanded speed and uses significantly larger control effort to make consensus happen which helps highlight the need for the 'slower' max-time protocol and more importantly the tokenized achievable-time protocol to achieve same-time consensus in constrained systems.

2.6.3 Six agents over a directed cycle

Next, we simulate six agents with their communication graph modeled as a directed six cycle as shown in Fig. 2.9. Similar to the graph communication topology shown previously, the graph is connected and a simple path between the six nodes exists. The initial conditions of the agents are given by

$$\boldsymbol{\delta}(0) = \Big[10 \text{ km}, 15 \text{ km}, 25 \text{ km}, 20 \text{ km}, 12 \text{ km}, 17 \text{ km}\Big]^T,$$
$$\mathbf{u}(0) = \Big[25 \text{ m/s}, 25 \text{ m/s}, 25 \text{ m/s}, 25 \text{ m/s}, 25 \text{ m/s}, 25 \text{ m/s}\Big]^T$$



(a) Position, speed, and acceleration

(b) Time-to-go estimate

Figure 2.8. Min-time protocol results (three agents).

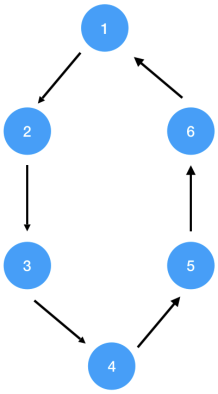
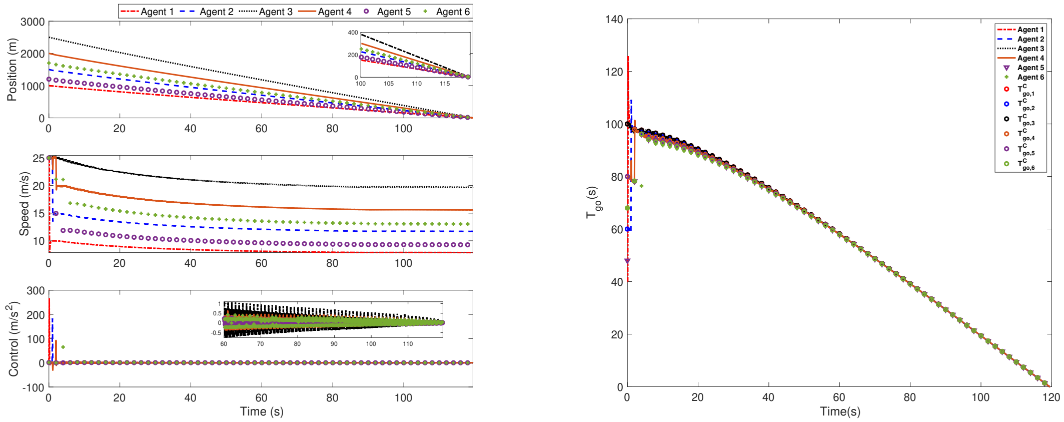


Figure 2.9. Graph connectivity (six agents).

which are chosen to represent a rendezvous problem in multiple fixed-wing UAVs. The proposed protocols are applied with $\kappa_{i,1} = 10$ and $\kappa_{i,2} = 0.2$.

2.6.3.1 Max-time protocol

The agents achieve position consensus with the max-time protocol as evidenced by Fig. 2.10(a). We notice that the consensus value of time-to-go for each agent eventually converges to the maximum of the time-to-go estimates of the agents and individually, each of these estimates converge to zero at the same time (Fig. 2.10(b)) implying that same-time position consensus with the target is achieved.



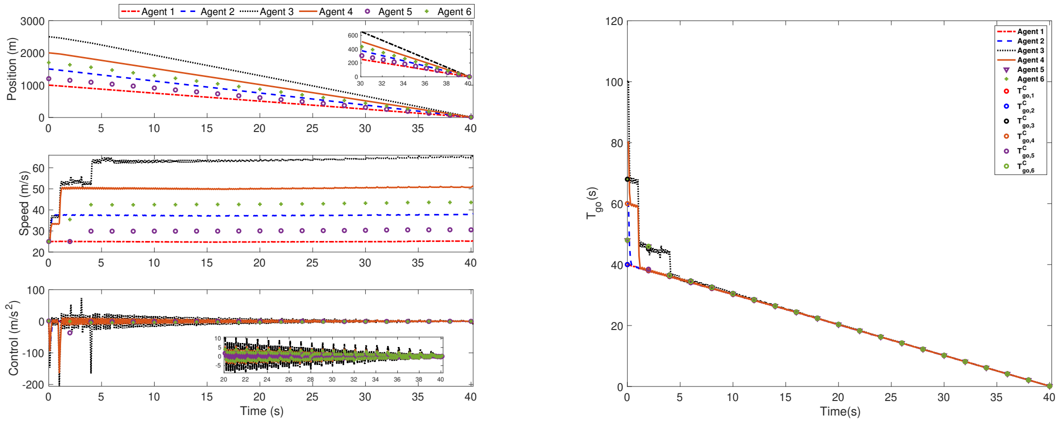
(a) Position, speed, and acceleration

(b) Time-to-go estimate

Figure 2.10. Max-time protocol results (six agents).

2.6.3.2 Min-time protocol

Through the use of the min-time protocol, it is evident from Fig. 2.11(a) that the agents achieve position consensus with the target. We notice that the agents move faster and achieve consensus in position quicker than the max-time protocol. From Fig. 2.11(b), we notice that the time-to-go consensus value for each agent converges to the minimum of the time-to-go estimates of all the agents, which eventually converges to zero for every agent at the same time. This shows that the same-time position consensus for the agents executing the min-time protocol has been achieved.



(a) Position, speed, and acceleration

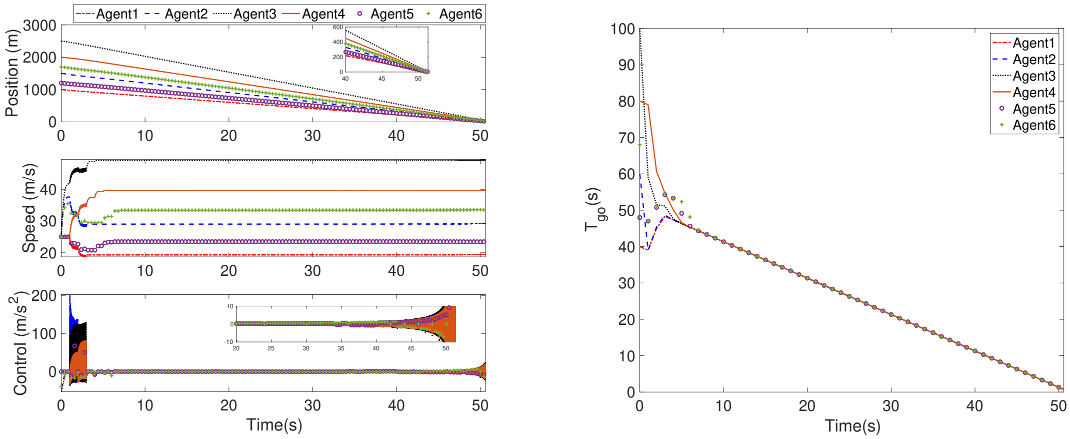
(b) Time-to-go estimate

Figure 2.11. Min-time protocol results (six agents).

2.6.3.3 Tokenized achievable-time protocol

The simulation results are as shown in Figs. 2.12(a), 2.12(b), and it is evident from Fig. 2.12(a) that the agents achieve position consensus with the target. The critical value of speed is chosen as $u_{\max} = 50$ m/s. We notice that the agents achieve consensus in position quicker than the max-time protocol (similar to the min-time protocol). From Fig. 2.12(b) we notice that the time-to-go consensus value for each agent converges to the most achievable time-to-go estimate of all the agents which eventually converges to zero for every agent at the same time thereby showing same time position consensus for the agents.

From Fig. 2.11(a), we notice that the commanded value of speed from the default min-time protocol is greater than the critical value ($u_{\max} = 50$ m/s) for



(a) Position, speed, and acceleration

(b) Time-to-go estimate

Figure 2.12. Tokenized achievable-time protocol results (six agents).

agents in the network. Based on the adjacency matrix, the value of the token is eventually set to the critical time-to-go of agent 3 (Fig. 2.13) which has the largest value of time-to-go of the agents that are traveling at their critical speeds similar to the complete graph achievable-time protocol.

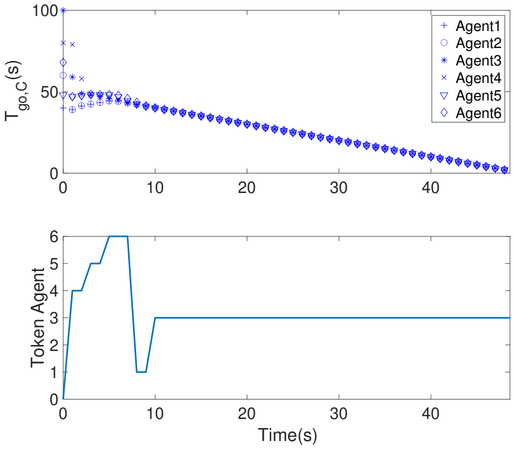


Figure 2.13. Tokenized achievable-time protocol consensus time-to-go and token agents vs time.

2.7 Summary

In this chapter, a consensus framework is developed that estimates the number of consensus evaluations required to achieve consensus on the time-to-go estimates and, ultimately, same-time position consensus with a target. The framework introduces and utilizes max-time, min-time, and achievable-time protocols to synchronize the positions of agents with a target at the same time. It addresses scenarios where min-time protocols may fail due to speed constraints, proposing the achievable-time protocol as an alternative to ensure consensus.

Simulations demonstrate the application of these protocols in secure rendezvous or salvo engagements involving multiple pursuers and a target. It is observed that the framework's applicability extends to any pursuer agents modeled within a connected communication graph, provided that the agents carry out at least N consensus evaluations and that the time between these evaluations is no less than N times the system's sampling interval.

Furthermore, the achievable-time protocol can be enhanced by introducing a distress token that circulates within the agent network, facilitating same-time position consensus even among agents with merely connected communication graphs. This foundational framework is also adaptable to an event-triggered consensus evaluation mechanism.

The chapter culminates by applying the discussed protocols to a planar multiple pursuer-target engagement scenario. By integrating state estimation, this framework is poised to offer a comprehensive solution to the challenges of simultaneous target tracking, state estimation, and same-time target position rendezvous.

Chapter 3

Same-time consensus for Missile Target Salvo Engagement

3.1 Prerequisites

In this section we introduce the estimation framework utilized further in this chapter. Chiefly we summarize the Extended Kalman Filter and the Unscented Kalman Filter.

3.1.1 Extended Kalman Filter(EKF)

The Extended Kalman Filter (EKF) is an advanced estimation method used for target state estimation in dynamic systems, as detailed in Table 3.1.

Table 3.1. Extended Kalman Filter for Target State Estimation

Model	$\dot{\mathbf{x}}_i = \mathbf{f}(\mathbf{x}_i, a_{m,i}) + \mathbf{G}w_i, \qquad w_i \sim N(0, Q)$ $\tilde{\mathbf{y}}_i = \mathbf{H}\mathbf{x}_i + \mathbf{v}_i, \qquad \mathbf{v}_i \sim N(\mathbf{0}, \mathbf{R}_i)$
Initialize	$\hat{\mathbf{x}}_i(0) = \hat{\mathbf{x}}_{i,0}$ $\mathbf{P}_{i,0} = \mathbb{E}\{\tilde{\mathbf{x}}_{i,0}\ \tilde{\mathbf{x}}_{i,0}^T\}$
Gain	$\mathbf{K}_i = \mathbf{P}_i^- \mathbf{H}^T \left[\mathbf{H}\mathbf{P}_i\mathbf{H}^T + \mathbf{R}_i\right]^{-1}$
Update	$\hat{\mathbf{x}}_i^+ = \hat{\mathbf{x}}_i^- + \mathbf{K}_i \left[\tilde{\mathbf{y}}_i - \mathbf{H}\hat{\mathbf{x}}_i^-\right]$ $\mathbf{P}_i^+ = \left[\mathbf{I} - \mathbf{K}_i\mathbf{H}\right]\mathbf{P}_i^-$
Propagation	$\dot{\hat{\mathbf{x}}}_i = \mathbf{f}(\hat{\mathbf{x}}_i, a_{m,i})$ $\dot{\mathbf{P}}_i = \mathbf{F}(t)\mathbf{P}_i + \mathbf{P}_i\mathbf{F}^T(t) + \mathbf{G}Q\mathbf{G}^T$

For an extensive discussion on the EKF refer to [38], [39]. In summary, the EKF operates through the following steps:

1. **Modeling:** The state-space model is defined by

$$\dot{\mathbf{x}}_i = \mathbf{f}(\mathbf{x}_i, a_{m,i}) + \mathbf{G}w_i, \quad w_i \sim N(0, Q),$$

   representing the system dynamics with process noise. The observation model is

$$\tilde{\mathbf{y}}_i = \mathbf{H}\mathbf{x}_i + \mathbf{v}_i, \quad \mathbf{v}_i \sim N(\mathbf{0}, \mathbf{R}_i),$$

   representing how the system states are observed with measurement noise.

2. **Initialization:** The initial state estimate $\hat{\mathbf{x}}_i(0)$ and the initial error covariance $\mathbf{P}_{i,0}$ are established, representing the initial knowledge about the system.

3. **Kalman Gain Computation:** The Kalman gain $\mathbf{K}_i$ is calculated as

$$\mathbf{K}_i = \mathbf{P}_i^- \mathbf{H}^T \left[\mathbf{H}\mathbf{P}_i^- \mathbf{H}^T + \mathbf{R}_i\right]^{-1},$$

   balancing the estimates from the model and the measurements.

4. **State Update:** The state estimate is updated using the Kalman gain:

$$\hat{\mathbf{x}}_i^+ = \hat{\mathbf{x}}_i^- + \mathbf{K}_i \left[\tilde{\mathbf{y}}_i - \mathbf{H}\hat{\mathbf{x}}_i^-\right],$$

   integrating new measurement information.

5. **Error Covariance Update:** The error covariance is updated to reflect the reduced uncertainty:

$$\mathbf{P}_i^+ = \left[\mathbf{I} - \mathbf{K}_i\mathbf{H}\right]\mathbf{P}_i^-.$$

6. **Propagation:** Between measurements, the EKF propagates the state and error covariance:

$$\dot{\hat{\mathbf{x}}}_i = \mathbf{f}(\hat{\mathbf{x}}_i, a_{m,i}), \quad \dot{\mathbf{P}}_i = \mathbf{F}(t)\mathbf{P}_i + \mathbf{P}_i\mathbf{F}^T(t) + \mathbf{G}Q\mathbf{G}^T,$$

   predicting future states and uncertainties.

The EKF provides an effective means to track and predict the state of a dynamic system by continuously updating its estimates based on new measurements and the system's inherent uncertainties.

3.1.2 Unscented Kalman Filter(UKF)

The Unscented Kalman Filter (UKF) is an estimation technique used for target state estimation in nonlinear systems. The UKF approach is summarized in Table 3.2,

Table 3.2. Unscented Kalman Filter for Target State Estimation

Model	$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \qquad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ $\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \qquad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$
Initialize	$\hat{\mathbf{x}}_0 = \hat{\mathbf{x}}_{0\vert 0}$ $\mathbf{P}_0 = \mathbf{P}_{0\vert 0}$
Sigma Points	$\mathcal{X}_k = \text{SigmaPoints}(\hat{\mathbf{x}}_{k\vert k}, \mathbf{P}_{k\vert k}, \kappa)$
Predict	$\mathcal{X}_{k\vert k-1} = \mathbf{f}(\mathcal{X}_k)$ $\hat{\mathbf{x}}_{k+1\vert k} = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{X}_{k\vert k-1}^{(i)}$ $\mathbf{P}_{k+1\vert k} = \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{X}_{k\vert k-1}^{(i)} - \hat{\mathbf{x}}_{k+1\vert k})(\mathcal{X}_{k\vert k-1}^{(i)} - \hat{\mathbf{x}}_{k+1\vert k})^T + \mathbf{Q}_k$
Update	$\mathcal{Y}_k = \mathbf{h}(\mathcal{X}_{k\vert k-1})$ $\hat{\mathbf{z}}_k = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}_k^{(i)}$ $\mathbf{S}_k = \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{Y}_k^{(i)} - \hat{\mathbf{z}}_k)(\mathcal{Y}_k^{(i)} - \hat{\mathbf{z}}_k)^T + \mathbf{R}_k$ $\mathbf{C}_k = \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{X}_{k\vert k-1}^{(i)} - \hat{\mathbf{x}}_{k+1\vert k})(\mathcal{Y}_k^{(i)} - \hat{\mathbf{z}}_k)^T$ $\mathbf{K}_k = \mathbf{C}_k \mathbf{S}_k^{-1}$ $\hat{\mathbf{x}}_{k+1\vert k+1} = \hat{\mathbf{x}}_{k+1\vert k} + \mathbf{K}_k(\mathbf{z}_k - \hat{\mathbf{z}}_k)$ $\mathbf{P}_{k+1\vert k+1} = \mathbf{P}_{k+1\vert k} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T$

Further details on the UKF may be found in the work done by Julier et al. [40]. In summary, the UKF involves the following key steps:

1. **Model:** The system and observation models are defined as

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k), \tag{3.1}$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k). \tag{3.2}$$

2. **Initialization:** The initial state estimate $\hat{\mathbf{x}}_0$ and the initial error covariance $\mathbf{P}_0$ are set based on prior knowledge.

3. **Sigma Points:** Sigma points $\mathcal{X}_k$ are generated to capture the mean and covariance of the state estimate accurately.

4. **Predict:** The UKF predicts the state at the next time step and updates the error covariance using the sigma points, as given by

$$\mathcal{X}_{k|k-1} = \mathbf{f}(\mathcal{X}_k), \tag{3.3}$$

$$\hat{\mathbf{x}}_{k+1|k} = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{X}_{k|k-1}^{(i)}, \tag{3.4}$$

$$\mathbf{P}_{k+1|k} = \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{X}_{k|k-1}^{(i)} - \hat{\mathbf{x}}_{k+1|k})(\mathcal{X}_{k|k-1}^{(i)} - \hat{\mathbf{x}}_{k+1|k})^T + \mathbf{Q}_k. \tag{3.5}$$

5. **Update:** The filter updates the state estimate and error covariance based on the new measurement:

$$\mathcal{Y}_k = \mathbf{h}(\mathcal{X}_{k|k-1}), \tag{3.6}$$

$$\hat{\mathbf{z}}_k = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}_k^{(i)}, \tag{3.7}$$

$$\mathbf{S}_k = \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{Y}_k^{(i)} - \hat{\mathbf{z}}_k)(\mathcal{Y}_k^{(i)} - \hat{\mathbf{z}}_k)^T + \mathbf{R}_k, \tag{3.8}$$

$$\mathbf{C}_k = \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{X}_{k|k-1}^{(i)} - \hat{\mathbf{x}}_{k+1|k})(\mathcal{Y}_k^{(i)} - \hat{\mathbf{z}}_k)^T, \tag{3.9}$$

$$\mathbf{K}_k = \mathbf{C}_k \mathbf{S}_k^{-1}, \tag{3.10}$$

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_k(\mathbf{z}_k - \hat{\mathbf{z}}_k), \tag{3.11}$$

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T. \tag{3.12}$$

These steps enable the UKF to effectively estimate the state of a nonlinear system, accurately capturing its uncertainty and adapting to new measurements.

3.2 Problem Setup

Let us consider a group of N pursuers and let $\mathcal{P} = \{1, 2, \ldots N\}$ denote the set of pursuer agents. These pursuers are in planar engagement with a target and the engagement is assumed to continue for the entire duration of the same-time consensus scenario. We are interested in formulating lateral acceleration commands for the pursuers to achieve same-time consensus with a target in finite time, using only a shared time-to-go estimate.

The engagement between a pursuer agent $i \in \mathcal{P}$ and the target (T) is schematically shown in Fig. 3.1 where $x - O - y$ is an inertial frame of reference and LOS stands for line of sight. Also in Fig. 3.1, the target velocity and heading angle are denoted by V_T and α_T, respectively, and the target accelerates with a lateral acceleration a_T (which is perpendicular to V_T). Note that agent i applies its acceleration $(a_{m,i})$ at an angle η_i w.r.t its velocity vector $(V_{m,i})$, as shown in Fig. 3.1.

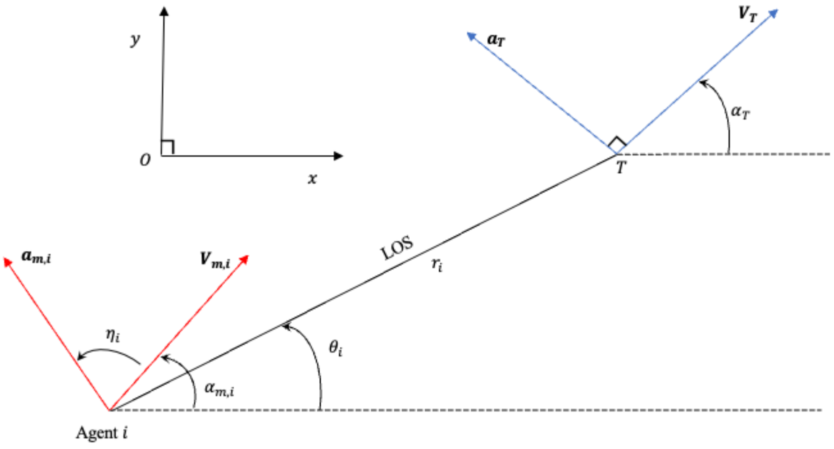


Figure 3.1. The planar engagement geometry.

The engagement kinematics between a pursuer agent $i \in \mathcal{P}$ and the target can be expressed as

$$\begin{bmatrix} \dot{r}_i \\ \dot{\theta}_i \\ \dot{V}_{i,r} \\ \dot{V}_{i,\theta} \\ \dot{\alpha}_{m,i} \\ \dot{V}_{m,i} \\ \dot{\alpha_T} \end{bmatrix} = \begin{bmatrix} V_{i,r} \\ \frac{V_{i,\theta}}{r_i} \\ \frac{V_{i,\theta}^2}{r_i} \\ -\frac{V_{i,\theta}V_{i,r}}{r_i} \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -\cos(\eta_i + \alpha_{m,i} - \theta_i) \\ -\sin(\eta_i + \alpha_{m,i} - \theta_i) \\ \frac{\sin(\eta_i)}{V_{m,i}} \\ \cos(\eta) \\ 0 \end{bmatrix} a_{m,i} + \begin{bmatrix} 0 \\ 0 \\ -\sin(\alpha_T - \theta_i) \\ \cos(\alpha_T - \theta_i) \\ 0 \\ 0 \\ \frac{1}{V_T} \end{bmatrix} a_T \quad (3.13)$$

where r_i is the relative separation between agent i and the target, θ_i is the LOS angle (or bearing angle to the target), $V_{i,r}$ is the component of relative velocity along the radial direction, and $V_{i,\theta}$ is the component of relative velocity along the tangential direction. Additionally, $\alpha_{m,i}$ is the heading angle of agent i. We define the position disagreement error for an agent i as $\delta_i(t) = r_i(t)$. An estimate of the time-to-go for each agent is selected as

$$t_{go,i}(t) = -\frac{\delta_i(t)}{V_{i,r}(t)} \quad (3.14)$$

Let the consensus value of time-to-go at any consensus time step $k_c \in \mathbb{Z}_\star$ (or at $t = k_c\Delta T_c$) be denoted by $T_f(k_c)$ (which, under the max-time protocol, is equal to $\max\{t_{go,i}(k_c\Delta T_c) : i \in \mathcal{P}\}$).

Remark 3.2.1. *In our other work [30], we showcase the utility of various time-to-go consensus protocols. In our application to the pursuer target engagement problem, requiring the pursuer agent to increase their speed is often not possible especially if the control input to the pursuer is a lateral acceleration. However, a pursuer can take a longer path to achieve same time consensus with another pursuer with a larger* *time-to-go estimate. Hence, the max time protocol has been chosen to be implemented in the solution framework*

Note that $T_f(k_c)$ is updated every ΔT_c and remains constant between two successive consensus evaluations. Adopting an approach similar to the one outlined in Kumar et al. [31], we specify the commanded value of time-to-go as

$$t_{go,C}(t) = T_f(k_c) - t_e, \forall t \in [k_c \Delta T_c, (k_c + 1)\Delta T_c] \tag{3.15}$$

where $t_e \in [0, \Delta T_c]$ is the time elapsed after the consensus evaluation at k_c. In this setting, we are interested in the same-time consensus defined previously in Section 2.1.1.

3.3 Main Result: Finite-time Engagement Kinematics

In this section, we examine same-time consensus of the pursuers in planar engagement with their common target as shown in Fig. 3.1. This happens in 2 stages: Time-to-go convergence and Terminal interception. Let us assume that the sampling time of the system is ΔT_s and the time interval between successive consensus evaluations is ΔT_c. Without loss of generality, we assume $\Delta T_c > \Delta T_s$. In our current setting, let $k_c \in Z$ be the time steps of consensus evaluations.

The error in time-to-go for agent $i \in \mathcal{P}$ is defined as previously in Eq. (2.13)

3.3.1 Time-to-go Convergence

It is crucial to have $\varepsilon_i(t) = 0, \forall i \in \mathcal{P}$ in finite time for the same-time consensus to occur. In a manner similar to Section 2.4, we synthesize the acceleration input in the following such that this is achieved.

The dynamics of error in estimate of time-to-go is obtained by differentiating the above as

$$\dot{\varepsilon}_i = \frac{r_i \dot{V}_{i,r}}{V_{i,r}^2} = \frac{V_{i,\theta}^2}{V_{i,r}^2} - \frac{r_i a_{m,i} \cos(\eta_i + \alpha_{m,i} - \theta_i)}{V_{i,r}^2} - \frac{r_i a_T \sin(\alpha_T - \theta_i)}{V_{i,r}^2} \tag{3.16}$$

Consider the positive definite candidate Lyapunov function

$$L_i = \frac{1}{2}\varepsilon_i^2 \tag{3.17}$$

The derivative of L_i with respect to time along the dynamics given in Eq. (3.16) yields

$$\dot{L}_i = \varepsilon_i \dot{\varepsilon}_i = \varepsilon_i(t) \left(\frac{V_{i,\theta}^2}{V_{i,r}^2} - \frac{r_i a_{m,i} \cos(\eta_i + \alpha_{m,i} - \theta_i)}{V_{i,r}^2} - \frac{r_i a_T \sin(\alpha_T - \theta_i)}{V_{i,r}^2} \right)$$

To effect finite time convergence, the control input (acceleration) is chosen as

$$a_{m,i} = \frac{[V_{i,\theta}^2 + V_{i,r}^2(\kappa_{i,1}\varepsilon_i + \kappa_{i,2}\varepsilon_i^{\alpha})]}{r_i \cos(\eta_i + \alpha_{m,i} - \theta_i)} - \frac{a_T \sin(\alpha_T - \theta_i)}{\cos(\eta_i + \alpha_{m,i} - \theta_i)} \tag{3.18}$$

The derivative of the candidate Lyapunov function after substituting the chosen control input with the max-time protocol is given by

$$\dot{L}_i = -2\kappa_{i,1} L_i - 2^{(1+\alpha)/2} \kappa_{i,2} L_i^{(1+\alpha)/2} \tag{3.19}$$

If $\alpha = \frac{p}{q}$ is such that p and q are both are odd numbers and $p \leq q$, $\dot{L}_i < 0$ and hence, L_i is a Lyapunov function for the error dynamics of agent $i \in \mathcal{P}$, and it is globally asymptotically stable (since L_i is radially unbounded). Next, we recall an important result from the Prerequisites Section 2.2.1

Substituting $a_{m,i}$ from Eq. (3.18) in Eq. (3.16), we observe that the error dynamics is in the form shown in Eq. (2.3). Thus, from Lemma 2.2.1, the error terms ε_i for all agents can be driven to zero between any two consecutive consensus evaluations by selecting the time to drive ε_i to zero to be $t_{\varepsilon_i} \leq \Delta T_c$. Also, based on the estimate of time-to-go, the agents' velocity state at the consensus evaluation time step is given as

$$V_{i,r}(k_c\Delta T_c) = \frac{-\delta_i(k_c\Delta T_c)}{T_f(k_c\Delta T_c)} \tag{3.20}$$

The control gain $\kappa_{i,2}$ can be chosen for an arbitrary value of $\kappa_{i,1}$ as

$$\kappa_{i,2} = \frac{\kappa_{i,1}\left[|\varepsilon_{max}(0)|^{(1-\alpha)}\right]}{\left(e^{\kappa_{i,1}(1-\alpha)\Delta T_c} - 1\right)} \tag{3.21}$$

where $|\varepsilon_{max}(0)| = \max\{|\varepsilon_i(0)| : i \in \mathcal{P}\}$ to account for the limiting value of the time of convergence among the agents. For a detailed discussion on how the parameters $\kappa_{i,1}$, $\kappa_{i,2}$, α affect convergence, see [32].

Remark 3.3.1. *Since these equations have been stated to be true at every ΔT_c s, on discretizing the states at the consensus evaluation time step we may write the closed loop modulated radial speed as*

$$V_{i,r}(k_c\Delta T_c) = \frac{-\delta_i(k_c\Delta T_c)}{\max\{t_{go,i}(k_c\Delta T_c) : i \in \mathcal{P}\}} \tag{3.22}$$

which is the same as the expression in Eq. (2.11)*.*

Since the speed has been modulated for the agents, The convergence in position of the pursuer agents to their target can now be analyzed equivalently with the results shown for the single integrator system shown in Appendix 2.3 in order to ascertain same time position convergence.

3.3.2 Terminal Approach

In traditional PN guidance laws, the impact is head on or a tail-chase. For terminal modulation of V_θ, we consider the following modified error function definitions. Error in time to go is denoted as $\varepsilon_{i,t} = t_{go,i} - t_{go,C}$ similar to ε_i above and error in component of relative velocity along the tangential direction as $\varepsilon_{i,\theta} = V_{i,\theta} - V_{i,\theta,d}$, where $V_{i,\theta,d}$ is some desired V_θ. Since our analysis is reliant on time-to-go estimates on a communication graph, the terminal approach is set in when the time to go consensus value of the network of agents is within $N\Delta T_c$ of position convergence i.e $\Delta T_c \leq N\Delta T_c$. With this we propose a modified candidate Lyapunov function

$$V(\varepsilon_{i,t}, \varepsilon_{i,\theta}) = \frac{1}{2}\left(\varepsilon_{i,t} + \lambda\varepsilon_{i,\theta}\right)^2 = \frac{1}{2}\varepsilon_{i,v}^2 \tag{3.23}$$

where $\varepsilon_{i,v} = \varepsilon_{i,t} + \lambda\varepsilon_{i,\theta}$ and $0 < \lambda < 1$. The time derivative of V in Eq.3.23 along the trajectories given by engagement dynamics can be obtained as,

$$\dot{V}(\varepsilon_t, \varepsilon_\theta) = \varepsilon_{i,v}\left[\left(\frac{V_{i,\theta}^2}{V_{i,r}^2} - \lambda\frac{V_{i,\theta}V_{i,r}}{r_i}\right) - a_{m,i}\ R_i\cos(\xi_i) - a_T\ R_i\sin(\alpha_T - \theta_i - \Psi_i)\right] \tag{3.24}$$

where, $\Psi_i = \tan^{-1}\left(\frac{\lambda V_{i,r}^2}{r_i}\right)$, $\xi_i = \eta_i + \alpha_{m,i} - \theta_i - \Psi_i$. Let $R_i = \sqrt{\left(\frac{r_i}{V_{i,r}^2}\right)^2 + \lambda^2}$. Similar to the analysis used to obtain Eq. 3.18, we obtain an expression for the control acceleration required as

$$a_{m,i} = \frac{1}{\cos(\xi_i)}\left[-a_T\ \sin(\alpha_T - \theta_i - \Psi_i) + \frac{1}{R_i}\left(\frac{V_{i,\theta}^2}{V_{i,r}^2} - \frac{V_{i,\theta}V_{i,r}}{r_i}\right) + \frac{(\kappa_{i,1}\varepsilon_{i,v} + \kappa_{i,2}\varepsilon_{i,v}^\alpha)}{R_i}\right] \tag{3.25}$$

On utilizing this control acceleration, the derivative of the candidate Lyapunov function is given as

$$\dot{V}_i = -2\kappa_{i,1}V_i - 2^{(1+\alpha)/2}\kappa_{i,2}V_i^{(1+\alpha)/2}$$

This is similar to the result obtained in Eq. (3.19). Through this result we find that the Lyapunov function decays to zero in finite time.(t_s) For our framework of same time consensus, we choose $t_s = \Delta T_c$. The gains for the terminal interception acceleration i.e $\kappa_{i,1}$ and $\kappa_{i,2}$ can be obtained using Lemma 2.2.1 and our choice of t_s.

On examining the error described in Eq. (3.23), we can see that on having the error $\varepsilon_{i,v} \to 0$ causes some of the agents to arrive at the target quicker. This effect causes a scheduled interception with the target with one of the agents intercepting the target within a small duration of one another. The scheduled interception is a direct consequence of the lock-in terminal target approach and depends on the consensus time interval(ΔT_c) and the number of pursuer agents.

3.4 Extension to Connected Communication Graphs

In the presented framework, the communication graph for the multi-agent system is assumed to be complete, meaning that every agent has access to the time-to-go information of all other agents. While this assumption simplifies the analysis, it may not be practical in some applications due to communication constraints, security concerns or the need to reduce information sharing overhead. Therefore, it is crucial to consider the extension of the results to scenarios where the communication between agents is modeled by a connected graph instead of a complete graph.

Utilizing max-plus algebra in a manner similar to Theorem 2.5.1, we can extend the results to connected graphs. To make such an extension, we analyze the path length of connected graphs, which is an essential property to consider in the context of multi-agent consensus algorithms. As described in [33], the path length in a connected graph refers to the number of edges between any two vertices. In a connected graph with N_s pursuer nodes, the maximum path length is at most $N_s - 1$.

Taking into account the maximum path length, we can establish the initial consensus time-to-go for the pursuers to ensure proper information dissemination in a connected communication graph. Specifically, the initial consensus time-to-go should be greater than or equal to $p\Delta T_c$ seconds, where p represents the maximum length of a simple path in the connected graph and ΔT_c is the time interval between consensus updates.

By adhering to this condition, we can guarantee that the agents will reach consensus on the time-to-go information even in the presence of a connected communication graph, thus ensuring the applicability of our proposed algorithms to more scenarios.

3.5 Performance

In this section, we evaluate the performance of the proposed guidance law.Unlike traditional missile guidance systems, the unique nature of our problem setup means that conventional performance metrics, such as miss distance, are not applicable. The derived control law ensures rendezvous of all pursuers with the target in a finite time frame.

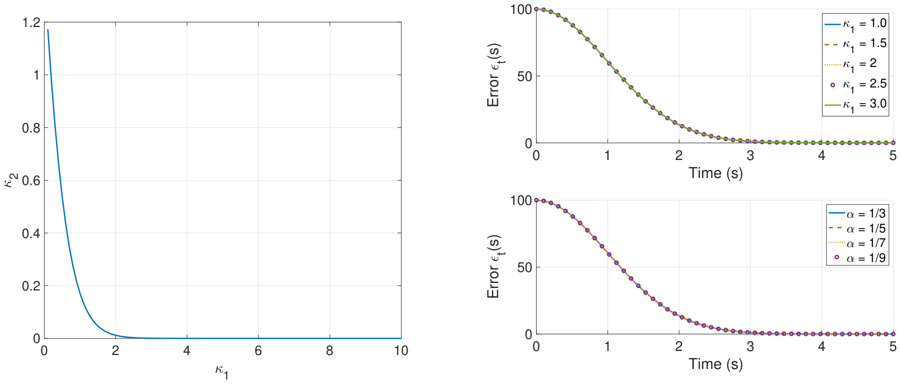
To effectively characterize the performance of our guidance law, we primarily focus on pivotal criteria:

1. The average latax of each pursuer.
2. The distance between the nearest pursuer and the target when the estimated time-to-go approaches 1 second.
3. The overall acceleration profiles of both pursuers.
4. Robustness to disturbances

With the use of the Extended Kalman Filter or the Unscented Kalman filter, we establish the robustness of our framework to disturbances. A salient observation from our results is the behavior of the guidance law with respect to time. The law demonstrates finite-time convergence of the dynamic time-to-go estimates towards the protocol maximum values. This tendency is further exemplified by the decay of

$t_{go,C}$ to zero. We break down this behavior for a more granular understanding in two primary stages.

Convergence of $t_{go,i}$ to $t_{go,C}$: The consensus of the time-to-go estimates, converging to their protocol value, is primarily influenced by several parameters. These include ΔT_C, which represents the time interval between successive consensus evaluations, and α and $\kappa_{i,1}$. The influence of ΔT_c has been discussed in detail. We begin by examining the role of $\kappa_{i,1}$ on the decay of error between time to go estimate and consensus value of time to go within the chosen ΔT_C.



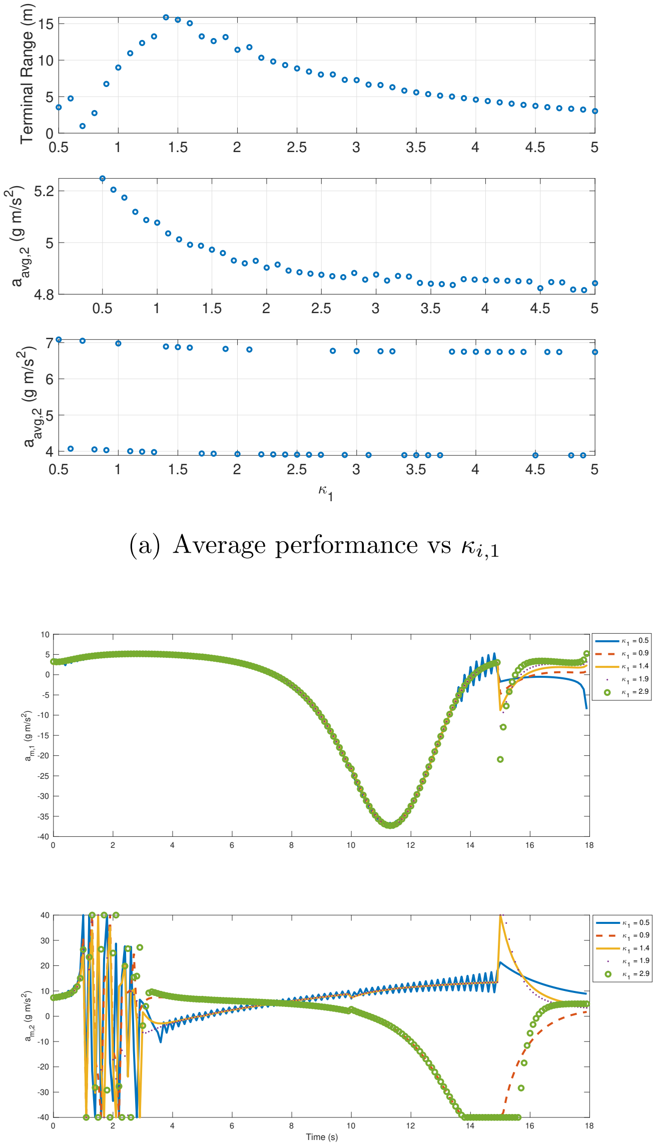
(a) $\kappa_{i,1}$ vs $\kappa_{i,2}$ for $\Delta T_c = 5sec$

(b) ϵ_t sensitivity to $\kappa_{i,1}$ and α

Figure 3.2. $t_{go,C}$ convergence plots.

To have the time-to-go estimate coverage, we establish a relationship between $\kappa_{i,1}$ and $\kappa_{i,2}$ which is visualised in Fig.3.2(a). On choosing the gains depicted on the curve, we examine the effect of tuning these gains on the performance of the guidance law.

In Fig. 3.2(b) we notice that as long as the gains lie on the curve described in Fig. 3.2(a), changing α or $\kappa_{i,1}$ does not change the nature of convergence of time-to-



(a) Average performance vs $\kappa_{i,1}$

(b) $a_{m,i}$ sensitivity to $\kappa_{i,1}$

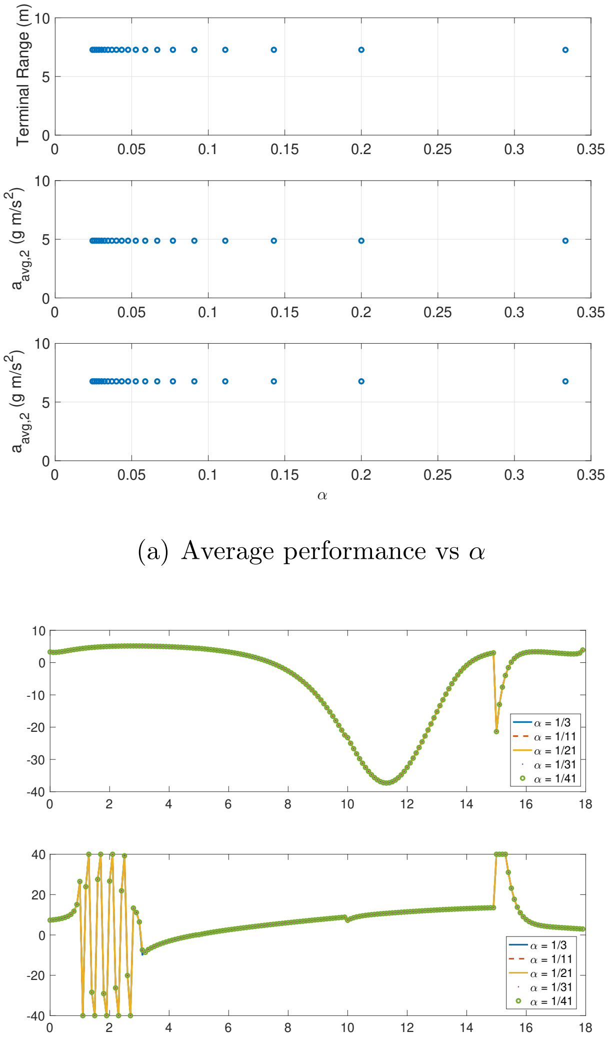
Figure 3.3. Performance sensitivity to $\kappa_{i,1}$.

go estimates to their consensus value. With this observation we discuss the effect on the performance of the guidance law

Performance sensitivity: To investigate the sensitivity of the guidance law performance with respect to parameter tuning, we simulate a scenario where the target travels at an elevated speed of $V_T = 200\ m/s$, executing a maneuver with an acceleration of $a_T = 5g\ m/s^2$. The effects of varying the gain $\kappa_{i,1}$ can be observed in Fig. 3.3(a) and Fig. 3.3(b). From the simulations, it becomes evident that choosing a value of $\kappa_{i,1} < 1$ induces high-frequency oscillatory peaks in the acceleration profile, a behavior deemed undesirable for our application. While the gain does not appear to directly influence the average acceleration, a higher gain facilitates the missile in achieving closer proximity to the target in a shorter timeframe.

From Fig. 3.4(a) and 3.4(b), it can be observed that variations in α exert a minimal impact on the selected performance parameters. Given the analyses conducted, we opt for $\kappa_{i,1} = 3$ and $\alpha = 1/3$ as the appropriate gains for our control law.

To demonstrate robustness of our control law to disturbances, we estimate the state of the target and use this estimate to implement our control law.



(a) Average performance vs α

(b) $a_{m,i}$ sensitivity to α

Figure 3.4. Performance sensitivity to α.

3.6 Target Estimation and Consensus

For same time consensus with a maneuvering target, we notice that the control term in Eq. (3.18) requires knowledge of the target acceleration, heading and velocity which must be estimated. For this purpose, an extended Kalman filter (EKF) has been proposed to effect the proposed control law.

We define the state vector for the filter (of agent i) as $\mathbf{x}_i = [r_i,\ \theta_i,\ V_{i,r},\ V_{i,\theta},\ \alpha_{m,i},\ V_{i,m}\ \alpha_T,\ V_T,\ a_T]^T$ which includes the target states (heading, velocity and acceleration). Thus, the truth model is given by

$$\dot{\mathbf{x}}_i = \begin{bmatrix} V_{i,r} \\ \frac{V_{i,\theta}}{r_i} \\ \frac{V_{i,\theta}^2}{r_i} - a_T \sin(\alpha_T - \theta_i) \\ -\frac{V_{i,\theta} V_{i,r}}{r_i} + a_T \cos(\alpha_T - \theta_i) \\ 0 \\ 0 \\ \frac{a_T}{V_T} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -\cos(\eta_i + \alpha_{m,i} - \theta_i) \\ -\sin(\eta_i + \alpha_{m,i} - \theta_i) \\ \frac{\sin(\eta_i)}{V_{m,i}} \\ \cos(\eta_i) \\ 0 \\ 0 \\ 0 \end{bmatrix} a_{m,i} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} w_i \tag{3.26}$$

where w_i is zero mean Gaussian noise with variance Q. The measurement equation is given as

$$\tilde{\mathbf{y}}_i = \mathbf{H}\mathbf{x}_i + \mathbf{v}_i \tag{3.27}$$

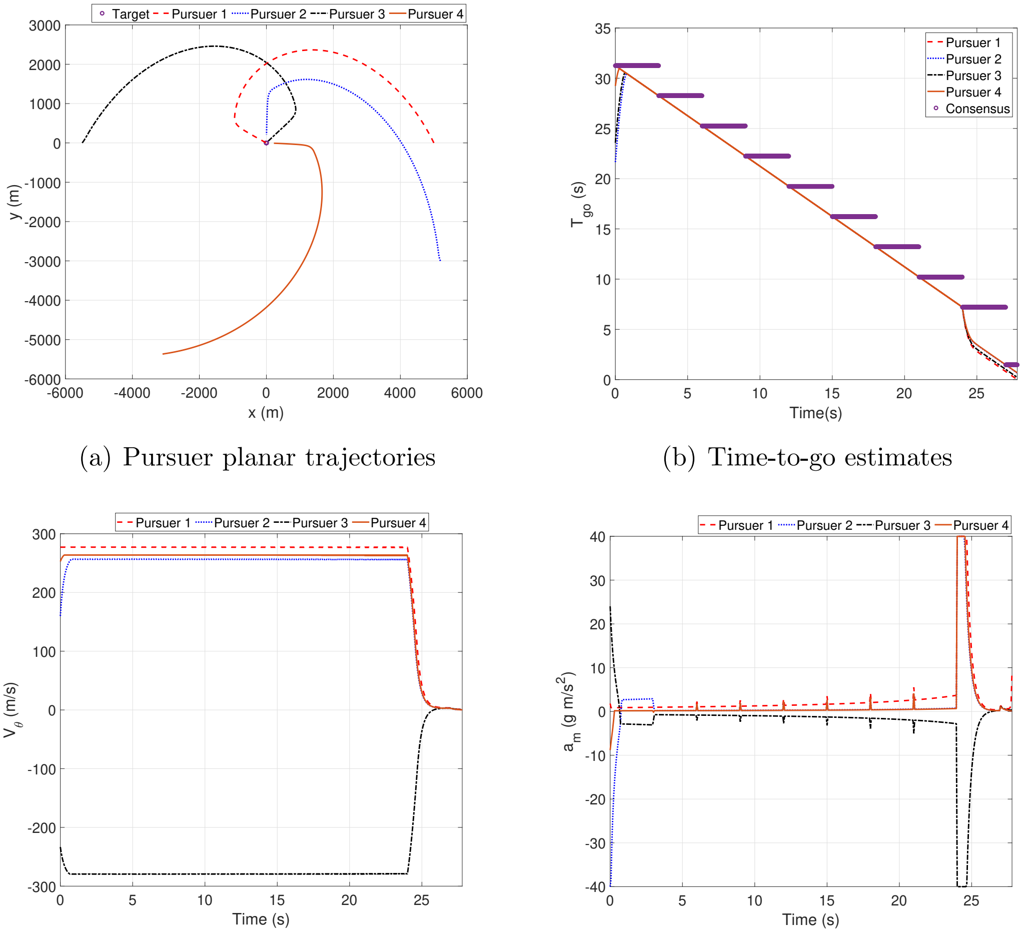
where $\mathbf{v}_i$ is assumed to be zero mean Gaussian white noise with variance given by $\mathbf{R}_i$.

We can utilize the Extended Kalman Filter(EKF) or the Unscented Kalman Filter(UKF) to estimate the states of the dynamic planar engagement.

- The EKF leverages the gradient of the dynamics to estimate the state of the system based on measurements. The Jacobian needed to compute estimates using the EKF has been provided in Appendix B.1.

- In certain situations where computation of the gradient is infeasible or tedious, we utilize the Unscented Kalman Filter(UKF) which leverages the Unscented Transform. [41].

Remark 3.6.1. *The control input is computed at intervals of* $\Delta T_u = m\Delta T_s$*. For effective convergence of the estimates,* $m \approx 10$*. The EKF or UKF can be utilized for a connected graph communication topology if the time interval between consensus evaluations* $\Delta T_C \geq p\Delta T_u$ *where 'p ' is the maximum length of a simple path in the graph communication network of the pursuer agents.*



(a) Pursuer planar trajectories

(b) Time-to-go estimates

(c) V_θ vs Time

(d) Pursuer control inputs

Figure 3.5. Stationary target: Trajectory, time-to-go estimates, vs time and control inputs.

3.7 Simulation Results

The illustrative simulation results are described in this section. First, we describe pursuer engagement with a stationary target. Subsequently, the scenarios wherein the agents communicate using a connected graph and utilize EKF estimates have been included.

Stationary Target

The engagement has been simulated for the stationary target scenario. The communication graph of information sharing $\mathcal{G}$ is assumed to be complete, i.e., any agent in the network can exchange information about time-to-go with every other agent. The initial positions of the agents are given by

$$\mathbf{r_0} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix} = \begin{bmatrix} 5000 \text{ m} \\ 6000 \text{ m} \\ 5500 \text{ m} \\ 6200 \text{ m} \end{bmatrix}, \qquad \theta_{\mathbf{0}} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} = \begin{bmatrix} 180^\circ \\ 150^\circ \\ 0^\circ \\ 60^\circ \end{bmatrix}$$

Two of the agents are traveling with identical initial velocities $V_{m,i} = 320$ m/s and the other two agents travel with $V_{m,i} = 330$ m/s the acceleration is applied at an angle $\eta_i = \pi/3$ rad w.r.t the velocity vector. The control acceleration is applied perpendicular to the velocity of the missile($\eta_i = 90^\circ$). In order to minimize control effort, the consensus evaluations are done every $\Delta T_c = 3$ s.

The agents achieve consensus to a stationary destination (the origin, as shown in Fig. 3.5(a)). We notice that the time-to-go estimates converge to the consensus value in finite time, and, in fact, they approach zero as they reach the target (Fig. 3.5(b)). Figures 3.5(a), 3.5(b) imply that the agents achieve rendezvous with their target at the same time. The control accelerations for each agent are shown in Fig. 3.5(d) and are observed to be bounded. We notice that during the terminal interception phase, the vehicles speed up and due to the decay of the V_θ component of velocity observed in Fig. 3.5(c), the pursuers are 'locked' in and achieve same time consensus at a time faster than the anticipated initial time-to-go due to the velocity being directed along the LOS radially towards the target

Constant Velocity Target

The communication graph of information sharing $\mathcal{G}$ is assumed to be connected which means every agent need not share time-to-go information with every other agent but there must be a path between any two agent nodes in the graph. We consider a $N = 4$ pursuer system chasing a single evader target. The communication between these pursuers is modeled as shown in Fig.4.3. The initial conditions of the pursuers are identical to the stationary target case discussed previously. The target moves at $V_T = 60m/s$

Complete graph

To begin with, we analyze the results for the case of a complete graph which is a connected graph where all agents can exchange communication with one another. In this scenario, every agent has the time-to-go information of the entire network.

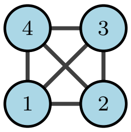
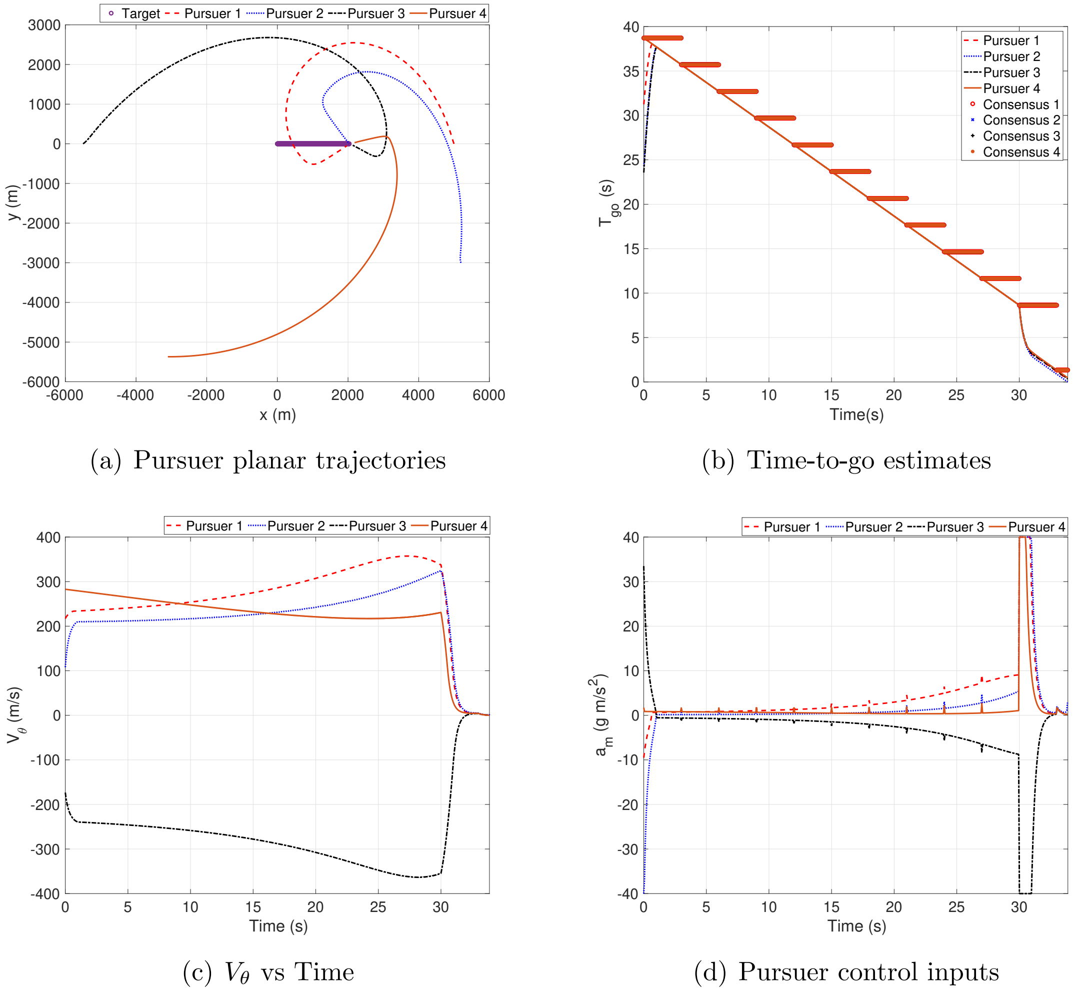


Figure 3.6. Graph Connectivity: Complete.

As is evident in Fig. 3.7(b), all the agents have the same time-to-go consensus value which is the maximum time-to-go of all the agents in the network. Same time consensus here is straightforward and is evidenced by the position of the agents in Fig. 3.7(a). The lateral acceleration of the pursuers is bounded as shown in Fig.3.7(d). In agreement with our framework, the pursuers lock in to the moving target as their consensus time-to-go approaches 10 sec. which results in a drastic lowering of the relative velocity in the tangential direction as evidenced by Fig.3.7(c).



(a) Pursuer planar trajectories

(b) Time-to-go estimates

(c) V_θ vs Time

(d) Pursuer control inputs

Figure 3.7. Directed cycle Constant Velocity target: Trajectory, time-to-go estimates, V_θ vs time and control inputs.

Directed cycle

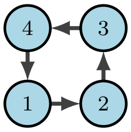


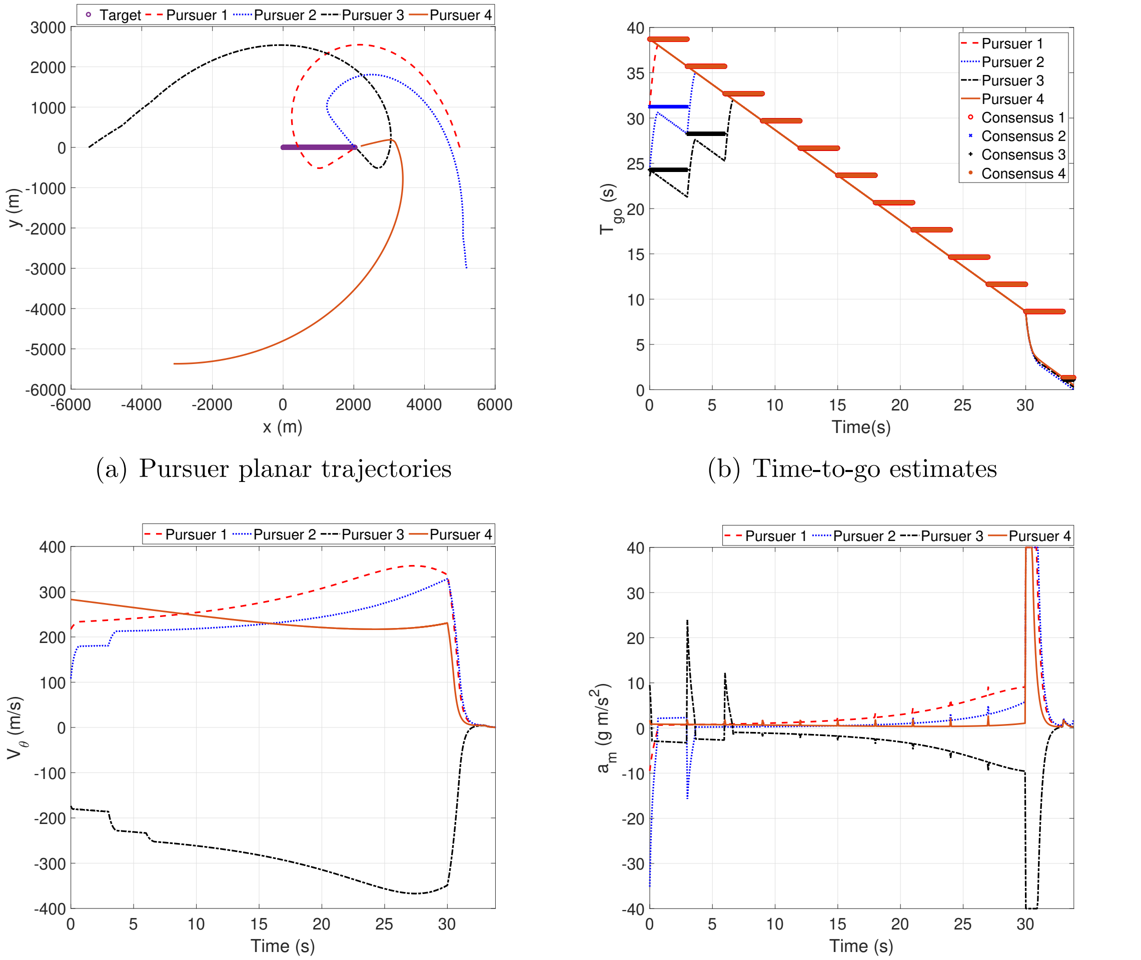
Figure 3.8. Graph Connectivity: Directed Cycle.

During terminal interception, pursuer agents successfully lock on to the moving target despite its motion. A decay in the V_θ component of the velocity is observed, as illustrated in Fig. 3.9(c). Same time consensus with the target is ultimately achieved, as depicted in Fig. 3.9(a). The impact of scarce communication among the pursuers is significant. In line with findings from our prior research [29], agents require 4 consensus evaluation intervals to achieve time-to-go consensus, consistent with the max time protocol. This can be inferred from Fig. 3.9(b).

Reducing communication between pursuers has both advantages and drawbacks. While it offers increased security, it also affects the initial consensus of their time-to-go values. It is crucial to consider that all the pursuers are at least $N_s + 1$ consensus evaluation intervals away from the target initially, where N_s denotes the maximum path length of the connected graph communication network. In the case of the directed cycle presented, this value is 3.

Undirected cycle

In Fig. 3.10, it can be observed that for an undirected cycle, the maximum path length is reduced, leading to a faster time-to-go consensus among the pursuers, as suggested by Fig. 3.11(b). Nonetheless, the terminal convergence bears resemblance to the directed cycle, as illustrated in Fig. 3.11(a), considering that the time-to-go consensus has already been reached and the initial conditions remain unchanged. The control law transition towards the end effectively locks in the pursuers, as demonstrated in Fig. 3.11(c), by providing bounded lateral acceleration inputs, as shown in Fig. 3.11(d). This observation highlights the impact of a denser communication graph structure on the consensus process and the subsequent performance of the proposed multi-agent control algorithm.



(a) Pursuer planar trajectories

(b) Time-to-go estimates

(c) V_θ vs Time

(d) Pursuer control inputs

Figure 3.9. Directed cycle Constant Velocity target: Trajectory, time-to-go estimates, V_θ vs time and control inputs.

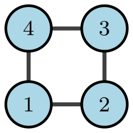
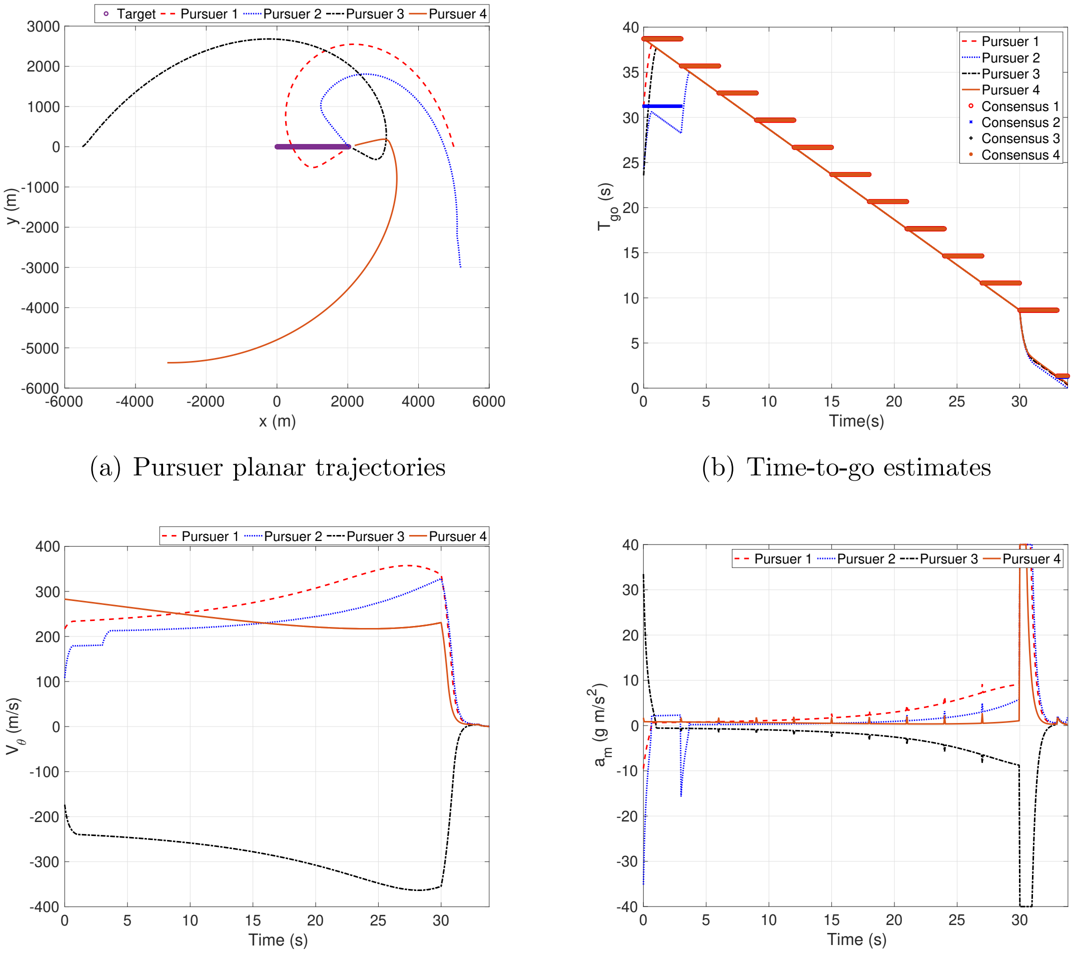


Figure 3.10. Graph Connectivity: Undirected cycle.

Maneuvering Target: No lock in

The consequence of using the lock-in terminal approach can be studied by observing same-time consensus without using the terminal approach control law. The



(a) Pursuer planar trajectories

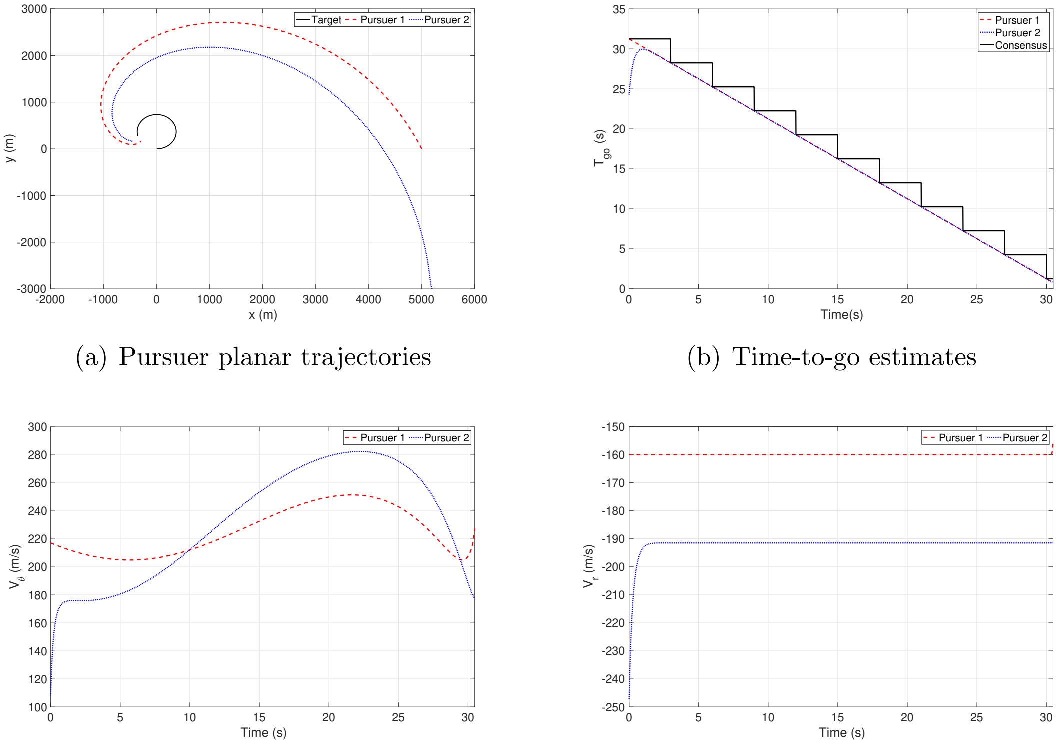
(b) Time-to-go estimates

(c) V_θ vs Time

(d) Pursuer control inputs

Figure 3.11. Undirected cycle Constant Velocity target: Trajectory, time-to-go estimates, V_θ vs time and control inputs.

results are shown in Fig. 3.12(a) - 3.12(d).As noted in the V_r plot shown in Fig. 3.12(d), the speeds are perfectly modulated however the terminal V_θ is quite large. On requiring the agents to lock-in for the terminal approach, we notice in Fig. 3.13(c) compared to 3.12(c) that the terminal V_θ is much lower which is essential for interception. As an artifact of this and due to the fact that we are utilizing a leaderless protocol, on observing the pursuer agent that does not set the value of $t_{go,C}$ i.e the agents with $t_{go,i}(t_{sw}) \neq t_{go,C}(t_{sw})$, where t_{sw} is the time at which we switch to the



(a) Pursuer planar trajectories

(b) Time-to-go estimates

(c) V_θ vs Time

(d) V_r vs Time

Figure 3.12. No lock in: Trajectory, time-to-go estimates, V_θ and V_r vs time.

terminal approach control law, the agents intercept the target at a time-to-go lower than the consensus value of time-to-go. This causes a “scheduling” effect for the intercepting agents. When the terminal approach is engaged, the $V_\theta \rightarrow \Delta T_C$ for the agents because of how V_θ appears in Eq. (3.23). This causes the time-to-go of the agents to be lower than the $t_{go,C}$ hence causing interception quicker than anticipated. The additional change in the $\varepsilon_{i,v}$ for the pursuers requires additional time to converge to zero compared to the agent that sets $t_{go,C}$ and since this occurs as the agents approach the target, the time-to-go values of the agents are comparable to the value of t. An estimate for the scheduled decrease in time-to-go can be obtained using Lemma 2.2.1.

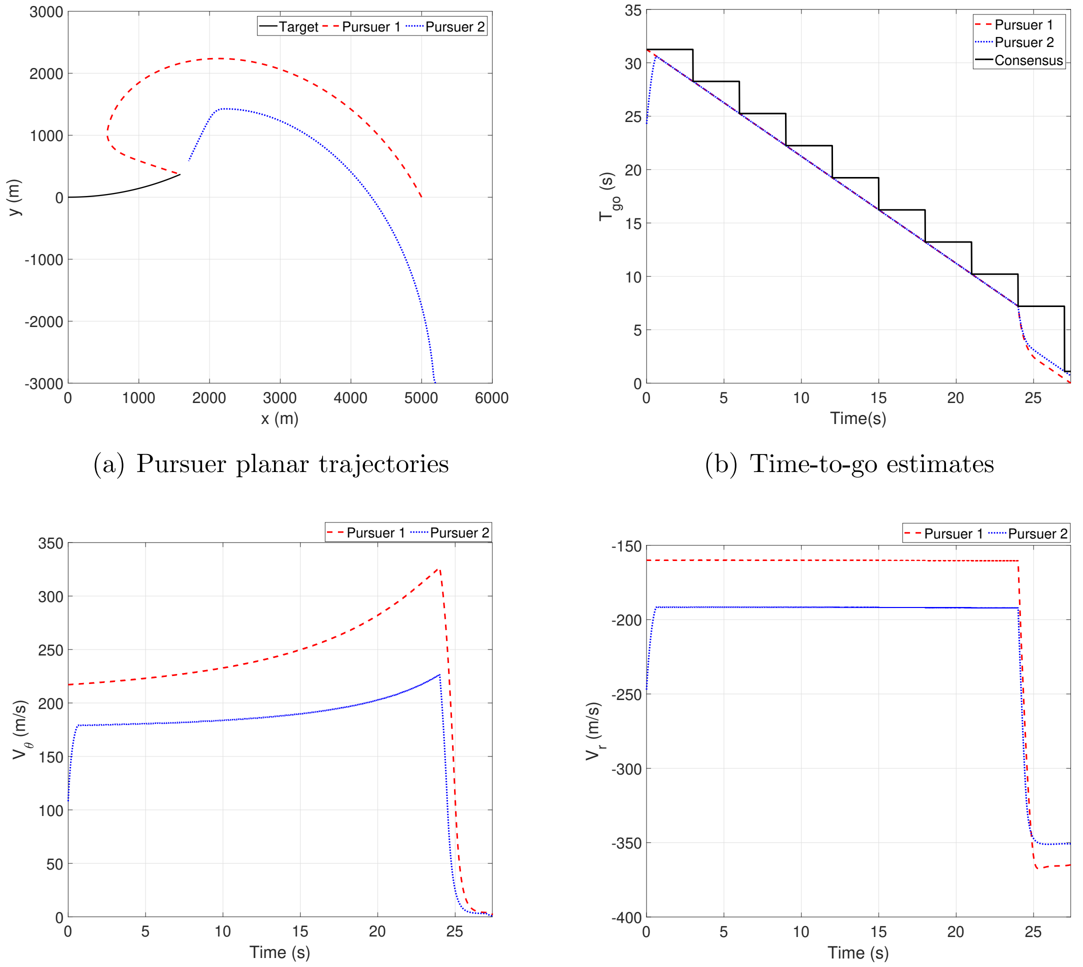
Maneuvering Target: Perfect information

The engagement has been simulated for a maneuvering target and $N = 2$ agents assuming information about all the states is known. The target is moving at $V_T = 60$ m/s with a constant lateral acceleration $a_T = 0.1g$ ($g = 9.8$ m/s^2). The initial positions of the agents are chosen as

$$\mathbf{r_0} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} 5 \text{ km} \\ 6 \text{ km} \end{bmatrix}, \qquad \theta_\mathbf{0} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 180^\circ \\ 150^\circ \end{bmatrix}$$

Agents are traveling at $V_{m,i} = 320$ m/s. The control acceleration is acting at $\eta_i = 90^\circ$ for all the agents. In order to minimize control effort, consensus is evaluated every $\Delta T_c = 3$ s. The agents achieve consensus to a target initially at the origin, as shown in Fig. 3.13(a)).

From Fig. 3.13(a) and Fig. 3.13(b), we can conclude that the agents intercept the maneuvering target at the same time. Fig. 3.13(d) is illustrative of the framework of consensus through speed modulation, it is important to note that the agents slow down to accommodate the agent with the largest time-to-go to rendezvous with the target. The 'slowest' agent or agent with the largest value of time-to-go (which is Pursuer 1 in this case) dictates the consensus time-to-go value. The control accelerations for each agent remain bounded. The upward trend in commanded acceleration as the agents arrive close to the destination is due to the definition of time-to-go. The $V_{i,r}$ term increases for a decreasing value of r_i which leads to an increase in the commanded control compounded by the appearance of r_i in the denominator of Eq. (3.18). During the terminal phase, due to the modulation of the V_θ, the V_r term increases causes faster interception. However, this occurs with a tradeoff. Due to residuals in the modulation of V_θ, the speed modulation for time to go consensus suffers causing same time consensus with a greater ϵ radius of convergence.



(a) Pursuer planar trajectories

(b) Time-to-go estimates

(c) V_θ vs Time

(d) V_r vs Time

Figure 3.13. Maneuvering target: Trajectory, time-to-go estimates, V_θ and V_r vs time.

EKF for Maneuvering Target

To show consensus, two agents and a target have been simulated. The initial conditions for the agents are

$$\mathbf{r_0} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} 5 \text{ km} \\ 6 \text{ km} \end{bmatrix}, \qquad \theta_\mathbf{0} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 180^\circ \\ 150^\circ \end{bmatrix}$$

The pursuer agents are initially traveling at velocities equal to 320m/s and the accelerations are applied laterally, i.e., $\eta_1 = \eta_2 = 90^\circ$. The Target is traveling at an initial velocity of $V_T = 60\text{m/s}$ and applies acceleration $a_T = 0.1g$ laterally, where g is the acceleration due to gravity. The estimates are initialized arbitrarily close to the actual values. The process noise is modeled with process noise covariance matrix given by

$$Q = diag([0, (\frac{\pi}{6})^2, 1, 1, (\frac{\pi}{6})^2, (\frac{\pi}{6})^2, 1, 1]) \tag{3.28}$$

We consider the following measurement model

$$\tilde{\mathbf{y}}_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_i + \mathbf{v}_i = \mathbf{H_1}\mathbf{x}_i + \mathbf{v}_i$$

where $\mathbf{v}_i$ is assumed to be zero mean Gaussian white noise with variance given by $\mathbf{R}_i$.

The variance $\mathbf{R}_i$ is given as

$$\mathbf{R}_i = \begin{bmatrix} \sigma_r^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_\theta^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\alpha_m}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\alpha_T}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{V_T}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{a_T}^2 \end{bmatrix}$$

Here σ_r is the standard deviation in the range measurements, σ_θ is the standard deviation in the measurements of the LOS angle, σ_{α_m} is the standard deviation in the measurements of the pursuer heading, σ_{α_T} is the standard deviation in the measurements of the target heading, σ_{V_T} is the standard deviation in the measurements of the target velocity and σ_{a_T} is the standard deviation in the measurements of the target acceleration.

The radar has a sensor measurement standard deviation of $\sigma_r = 0.1$ km and the seeker has a sensor measurement standard deviation of $\sigma_\theta = 0.01$ rad. The target heading, speed and lateral acceleration is obtained through measurements with covariances of $\sigma_{\alpha_T} = 0.01, \sigma_{V_T} = 0.1$ and $\sigma_{a_T} = 0.1$ respectively. The initial covariance matrix is chosen as $\mathbf{P}_{i,0} = \mathbf{I}$ for each agent. The communication between the agents is undirected i.e both the agents can exchange information with each other. The sampling time of the system is 10 ms. The time between control computation is 100 ms and consensus evaluation occurs every 2 seconds. 100 Monte Carlo simulations have been run and the averaged results are shown in Fig. 3.14. Due to the magnitude of velocities and control being computed every 100 ms, the agents arrive to within a distance of 100 m for same time consensus to be achieved. The error in the pursuer states converge to within bounds and same time consensus is indeed achieved.

EKF vs UKF for Maneuvering Target

The initial conditions for the agents are

$$\mathbf{r_0} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} 2 \text{ km} \\ 2.5 \text{ km} \end{bmatrix}, \qquad \theta_{\mathbf{0}} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0^\circ \\ 40^\circ \end{bmatrix}$$

The pursuer agents are initially traveling at velocities equal to 320m/s and the accelerations are applied laterally, i.e., $\eta_1 = \eta_2 = 90^\circ$. The Target is traveling at an initial velocity of $V_T = 200$m/s and applies acceleration $a_T = 5\ g$ laterally, where g is the acceleration due to gravity. The estimates are initialized arbitrarily close to the actual values. The process noise is modeled with process noise covariance matrix given by

$$Q = diag([0, (\frac{\pi}{6})^2, 1, 1, (\frac{\pi}{6})^2, (\frac{\pi}{6})^2, 1, 1]) \tag{3.29}$$

Measurement model is given by

$$\tilde{\mathbf{y}}_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}_i + \mathbf{v}_i$$

with the covariance of $\mathbf{v}_i$ as follows

$$\mathbf{R}_i = \text{diag}\left(\sigma_r^2, \sigma_\theta^2, \sigma_{V_r}^2, \sigma_{V_\theta}^2, \sigma_{\alpha_m}^2\right)$$

where $\sigma_{(\cdot)}$s denote standard deviation in the respective measurements.

The measurements are obtained with standard deviations of $\sigma_r = 1\text{m}, \sigma_\theta = 0.01^\circ$, $\sigma_{V_r} = 0.1$ m/s, $\sigma_{V_\theta} = 0.1$ m/s and $\sigma_{\alpha_m} = 0.01^\circ$. The initial covariance matrices for the filters are chosen as $\mathbf{P}_{i,0} = \mathbf{I}$ for each agent. For the modified problem we notice the pursuers intercept the target and due to the nature of the terminal guidance law, lock into the target as is evident from Fig. 3.16. 100 Monte Carlo simulations have been run and the averaged results are shown in Fig. 3.16

We notice in Fig. 3.15 that same time consensus with the target is achieved with both estimation methods. Time-to-go consensus is achieved among the pursuer agents and the estimates of time-to-go decay to zero.

The target state errors are within the 3 σ bounds for both estimation frameworks as evident in Fig. 3.16(a),3.16(b) and Fig. 3.16(c),3.16(d).

Estimation errors, specifically mean squared error (MSE) and root mean squared error (RMSE), are evaluated under varying measurement noise magnitudes, as illustrated in Fig. 3.17. If $\mathbf{e_{res,i}} = \mathbf{x_i} - \tilde{\mathbf{x}}$ denotes the residual error between true state ($\mathbf{x_i}$) and estimated state ($\tilde{\mathbf{x}}_\mathbf{i}$) for the i^{th} Monte Carlo run, we define MSE as

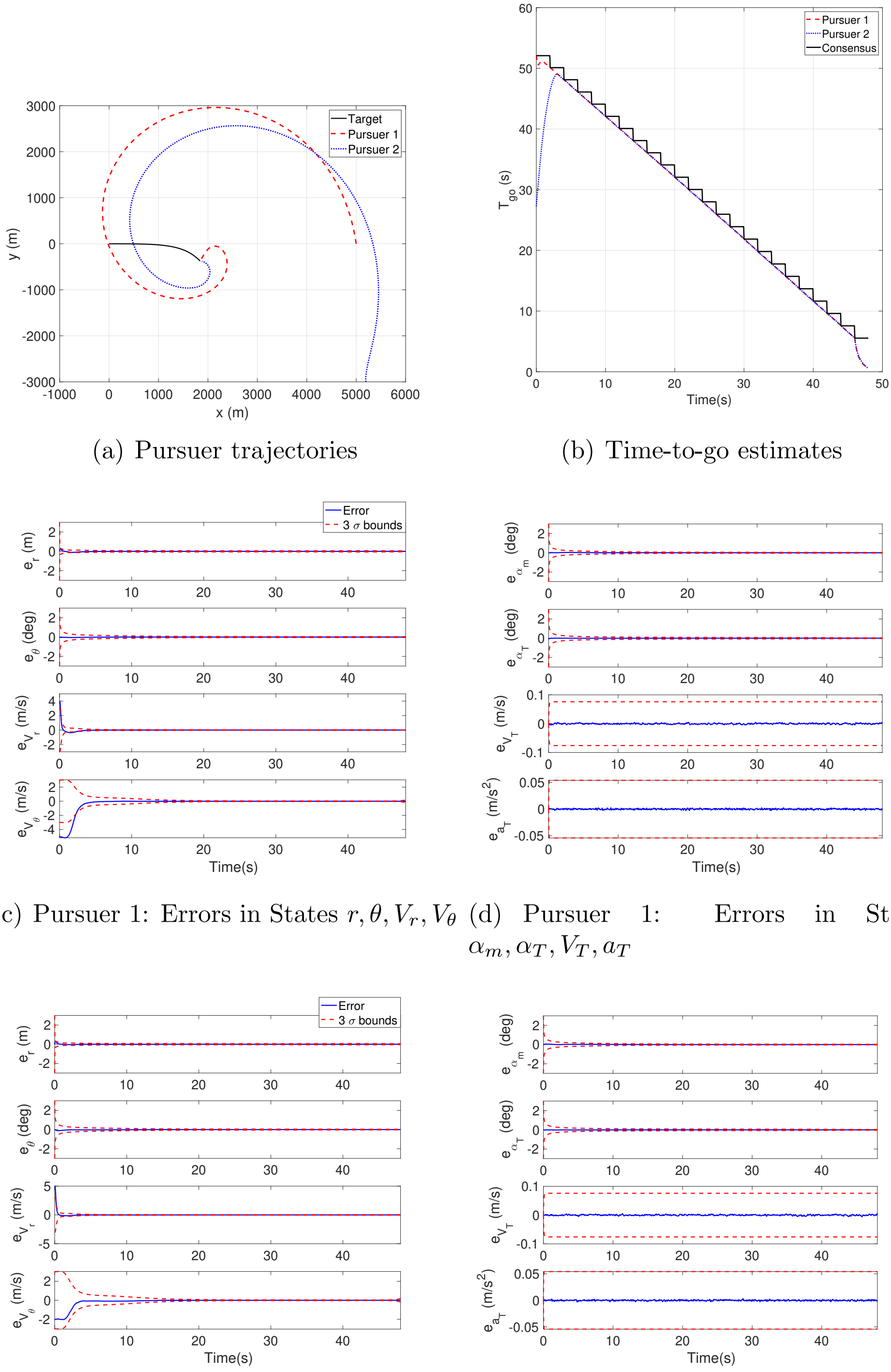
$$\mathbf{e_{res,mc}} = \sum_{i=1}^{100} \frac{\mathbf{e_{res,i}}}{100}$$
$$MSE = \sum \frac{\|(\mathbf{e_{res,i}} - \mathbf{e_{res,mc}})\|^2}{100}$$

Both EKF and UKF estimates exhibit similar errors. The EKF slightly outperforms, likely due to the accessible gradient of the dynamics. Conversely, the engagement's nonlinear characteristics and control contribute to increased UKF estimate errors.

The estimation errors – namely, mean squared error (MSE) and root mean squared error (RSME) – are computed for varying order of magnitudes of measurement noise.

The results are shown in Fig. 3.17. We notice that the errors in the EKF and UKF estimates are comparable. The EKF overall does a little better at estimating the states presumably due to the availability of the gradient of the dynamics while the nonlinear nature of the engagement geometry and the control, causes greater errors in the UKF estimates. However, as the magnitude of measurement error increases, the UKf performance starts to become more comparable and in some cases better than the EKF which is presumably due to the measurement Jacobian generated in the EKF update process being more inaccurate.

Remark 3.7.1. *It should be noted that discrepancies in estimation accuracy between the pursuers are observed. One pursuer achieves more accurate estimations by setting the protocol value of time-to-go, whereas the other, which aligns its time-to-go to this protocol value, exhibits higher estimation errors.*



(a) Pursuer trajectories

(b) Time-to-go estimates

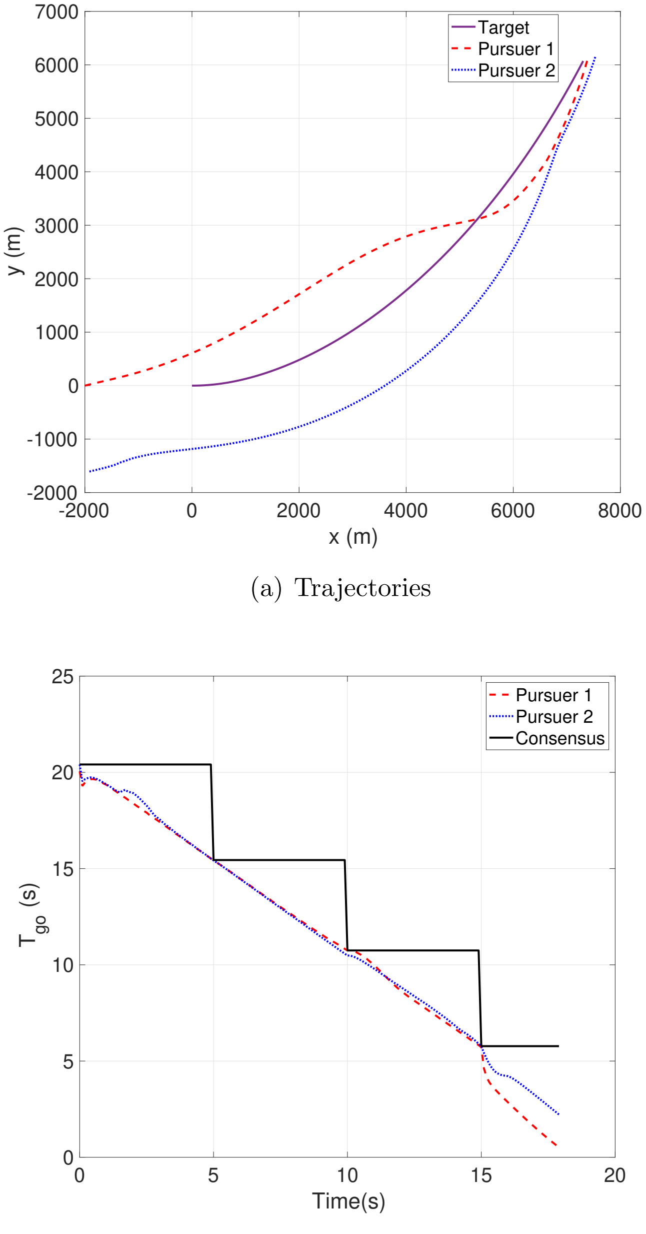
(c) Pursuer 1: Errors in States r, θ, V_r, V_θ

(d) Pursuer 1: Errors in States $\alpha_m, \alpha_T, V_T, a_T$

(e) Pursuer 2: Errors in States r, θ, V_r, V_θ

(f) Pursuer 2: Errors in States $\alpha_m, \alpha_T, V_T, a_T$

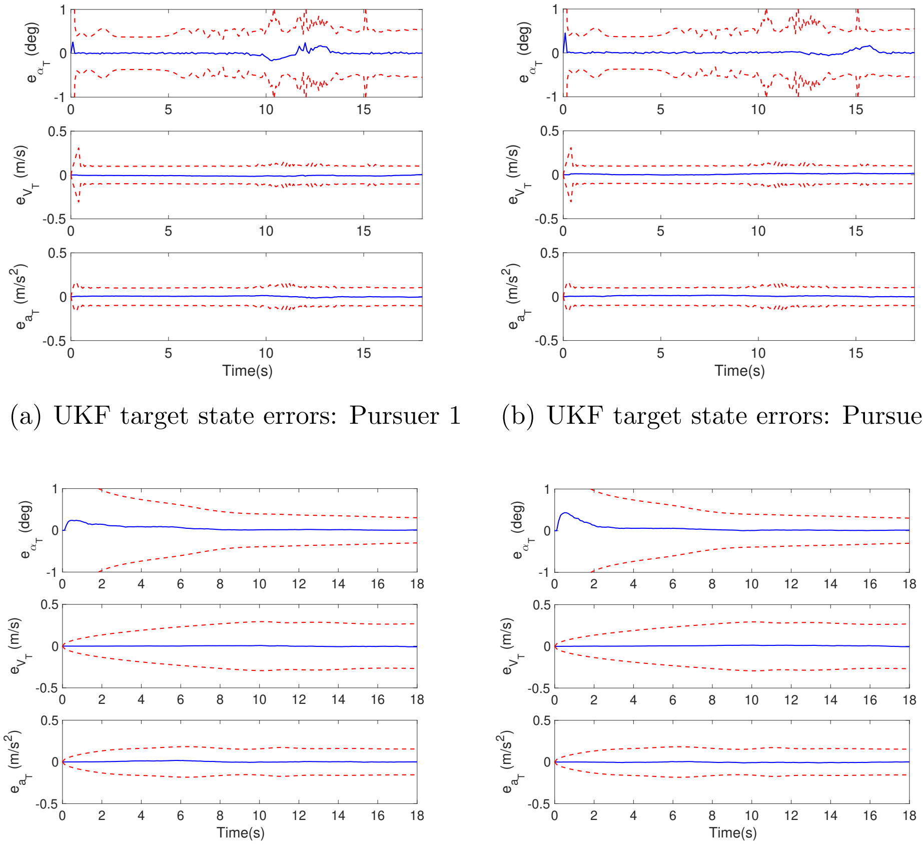
Figure 3.14. EKF for maneuvering target: Trajectory, errors in state estimation.



(a) Trajectories

(b) Time-to-go estimates

Figure 3.15. Estimation performance: Trajectories and time-to-go.



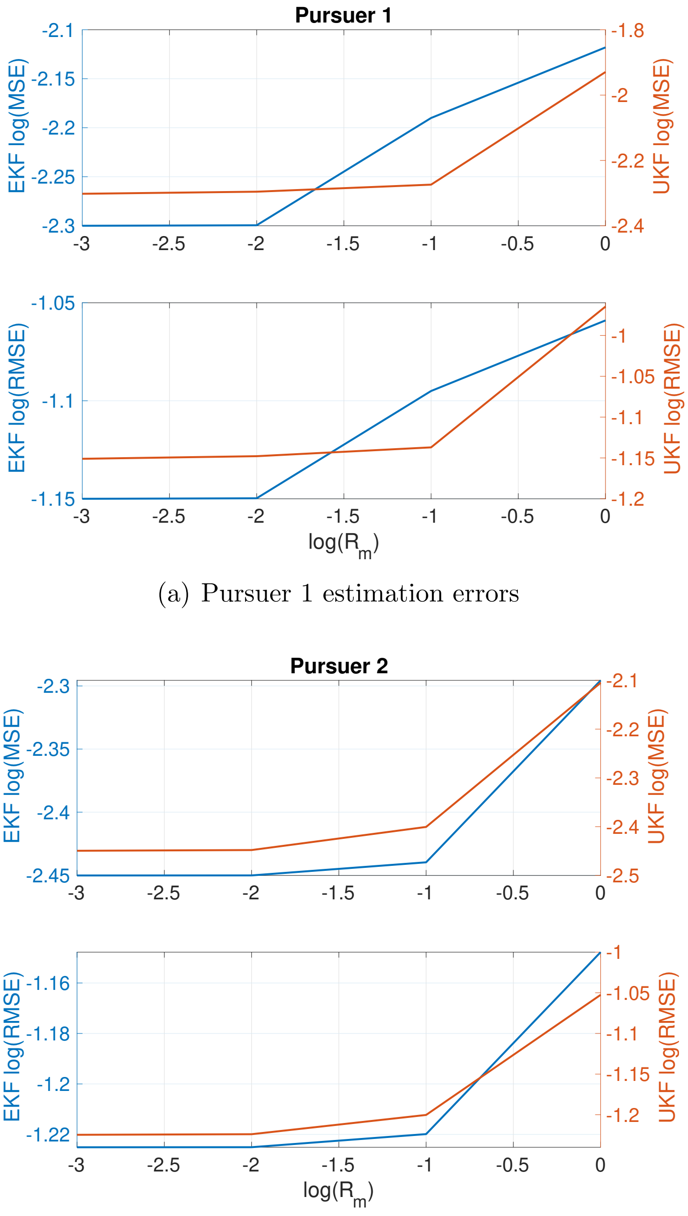
(a) UKF target state errors: Pursuer 1

(b) UKF target state errors: Pursuer 2

(c) EKF target state errors: Pursuer 1

(d) EKF target state errors: Pursuer 2

Figure 3.16. State Estimation performance.



(a) Pursuer 1 estimation errors

(b) Pursuer 2 estimation errors

Figure 3.17. Error in estimation vs measurement noise.

3.8 Summary

The control acceleration proposed effects consensus of multiple pursuer vehicles to a target at the same time, through the use of a shared estimate to each of the vehicles' time-to-go. This consensus is reached by integrating a shared estimate of each vehicle's time-to-go. The time-to-go estimation method provides a synchronized solution, ensuring the effective coordination of multiple pursuers to the target, a task that is especially challenging in dynamic environments.

As the group of pursuer vehicles nears the target during the terminal phase of the operation, we introduce a refined control law. This law is meticulously designed to achieve same-time consensus while concurrently minimizing the rotational component of relative velocity. By managing both these aspects, our approach ensures precision in target acquisition while retaining same time consensus within the multi-agent system.

The versatility of our consensus and estimation framework is highlighted by its compatibility with agent communication models represented by any connected graph. Given its time-dependent nature, our methodology is resilient to course alterations by individual agents, preserving the consensus among the pursuer vehicles. This adaptability brings about a level of robustness that is highly advantageous in complex, real-world scenarios where system parameters may vary over time.

Our research demonstrates that the proposed framework facilitates the achievement of consensus within a finite time frame, regardless of whether the targets are stationary or maneuvering. This represents a significant leap forward in the field, providing a viable solution to one of the most challenging aspects of multi-agent systems.

Additionally, our work includes a comparative analysis on the selection of non-linear filtering techniques. Specifically, we assess these techniques for their efficacy in simultaneously tracking and achieving consensus to a maneuvering target. This comparison provides insights into the optimal filtering techniques for ensuring reliable tracking and target acquisition in dynamic contexts.

Chapter 4

Optimal Finite Time Cooperative Rendezvous for Multiple Vehicles

4.1 Problem Statement

Consider 'N' vehicles and let $\mathcal{P}$ denote the set of agents that achieve same-time consensus with a target. Let $\mathcal{G}$ denote the communication network among the vehicles. The agents exchange their time-to-go information every 'ΔT_c sec. The rendezvous is presented for a network of agents. The dynamics of each agent is given as a modified form of the Dubin's car model with speed as an additional control variable.

$$\dot{\boldsymbol{x}}_i = \begin{bmatrix} \dot{p}_{x,i} \\ \dot{p}_{y,i} \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} V_i \cos\theta_i \\ V_i \sin\theta_i \\ \omega_i \end{bmatrix} \tag{4.1}$$

here, subscript 'i' denotes the i^{th} agent, (p_x, p_y) denote the (x,y) Cartesian coordinates of the agent, θ denotes the steering angle of the agent and V, ω denote the velocity control input and the angular velocity control input respectively.

We also consider an additional differential drive model for the vehicles with dynamics given as

$$\dot{\boldsymbol{x}}_i = \begin{bmatrix} \dot{p}_{x,i} \\ \dot{p}_{y,i} \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \frac{(u_{l,i}+u_{r,i})}{2} \cos\theta_i \\ \frac{(u_{l,i}+u_{r,i})}{2} \sin\theta_i \\ \frac{(u_{l,i}-u_{r,i})}{b} \end{bmatrix} \tag{4.2}$$

here u_l and u_r denote the speeds of the left and right wheels respectively and b denotes the wheelbase. The goal is to compute the optimal control commands to minimize the performance index J

$$J = \int_{t_0}^{T_f} L(\boldsymbol{x}, u, t)dt = \frac{1}{2}\int_{t_0}^{T_f} (u(t)^T u(t))dt \tag{4.3}$$

By minimizing the above performance index, we find the control inputs required to achieve same time consensus with the target location requiring least amount of control effort. If r_i denotes the distance between agent 'i' and the target, we define same time consensus as Section 2.1.1

4.2 Solution Methodology

The solution methodology consists of two parts:

- Utilising the exchange in time-to-go information to formulate a protocol based consensus time-to-go.
- Leverage this consensus value of time-to-go obtained to synthesize finite-time optimal control commands calculated for a control horizon based on computational restrictions.

Each agent calculates the time necessary for position convergence with the target. We choose the max time protocol from [30] as our choice for the time-dependent optimal control parameter $T_f(t)$ for each agent. Based on this consensus time-to-go value, optimal control commands $u(T_f, t, \boldsymbol{x}_{i,0})$ are synthesized. Among the computed optimal control commands, those for $t = \Delta T_{hor}$ are utilized, where ΔT_{hor} represents the local optimal controller's horizon.

Since ΔT_{hor} is related to the agents' hardware and computational capabilities, it is reasonable to evaluate consensus in time-to-go at intervals of $\Delta T_c = \Delta T_{hor}$ for optimal adaptation of the time-dependent trajectories. If consensus is assessed at a rate much slower than the controller horizon, the agents' hardware capabilities would be underutilized. On the other hand, if time-to-go consensus is evaluated more quickly, the agents' computational capabilities might struggle to keep up with the trajectory adaptation demands.

As vehicles approach the target and their time-to-go estimates become comparable to the controller horizon's value, trajectory adaptation may introduce errors. Consequently, agents cease sharing their time-to-go estimates and rely on the optimal controller's commands to achieve positional consensus with the target.

An important detail of our framework is the local T_f for each agent. In a manner similar to Kumar et al. [31], between successive consensus evaluations $t \in [k\Delta T_C(k+1)\Delta T_c]$, we define the protocol consensus time-to-go as

$$T_f(t) = T_f(k\Delta T_c) - t_e \tag{4.4}$$

where t_e denotes the time elapsed since the last consensus evaluation. This causes the eventual decay of the time-to-go to effect same-time consensus. In order to illustrate the framework of time-to-go consensus and optimization, we consider a motivating example of same-time consensus of multiple vehicles with double integrator dynamics to a target.

4.2.1 Motivating Example

Let us consider 'N' double integrator agents. The dynamics are described as

$$\dot{\boldsymbol{x}}_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \boldsymbol{x}_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i \tag{4.5}$$

where $\boldsymbol{x}_i = [x_1 \quad x_2]^T$ denotes the state of the agent comprising of the position and velocity of the i^{th} agent. For convenience, we assume that the state is relative to the stationary target at the origin. The gain free optimal control solution is well documented in literature. We utilize the finite-time analytical solution from Lewis et al. [42]. The optimal control input obtained for minimizing the control effort subject to having the agent arrive at the target location in finite time T_f is given as

$$u(T_f, t, \boldsymbol{x_i}) = \begin{bmatrix} \frac{(6T_f - 12t)}{T_f^3} & \frac{(-2T_f + 6t)}{T_f^2} \end{bmatrix} (\boldsymbol{x}_{target} - \begin{bmatrix} 1 & T_f \\ 0 & 1 \end{bmatrix} \boldsymbol{x_i}) \tag{4.6}$$

Since we can obtain an analytical expression for the optimal control in this case, it is easy to see that the solution can be parameterized in terms of the target position and the finite time required to reach the target. On doing this, for a given target position and finite time-to-go, it is possible to obtain an optimal set of control inputs. Leveraging this we observe the same-time consensus problem of multiple ground vehicle agents to a target location while may not have an analytical solution to the optimal control problem, can be parameterized similarly to obtain the optimal control inputs to achieve same-time consensus.

4.2.2 Boundary value problem

The optimal rendezvous of agents to a target location can also be posed as a boundary value problem. For Dubin's car dynamics presented in Eq.(4.1), the Hamiltonian would be defined as

$$H = \frac{1}{2}(\omega_i^2 + V_i^2) + \lambda_1 V_i \cos(\theta_i) + \lambda_2 V_i \sin(\theta_i) + \lambda_3 \omega_i \tag{4.7}$$

where λ_j $j = 1, 2, 3$ denote the costates for an agent 'i'. The costate equations when using Pontryagin's Maximum Principle are given as

$$\dot{\lambda}_1 = -\frac{\partial H}{\partial p_{x,i}} = 0 \tag{4.8}$$

$$\dot{\lambda}_2 = -\frac{\partial H}{\partial p_{y,i}} = 0 \tag{4.9}$$

$$\dot{\lambda}_3 = -\frac{\partial H}{\partial \theta_i} = -\lambda_1 V_i \sin(\theta_i) + \lambda_2 V_i \cos(\theta_i) \tag{4.10}$$

The optimal control inputs are derived in terms of the costate as

$$\omega_i^* = -\lambda_3 \tag{4.11}$$

$$V_i^* = -(\lambda_1 \cos(\theta_i) + \lambda_2 \sin(\theta_i)) \tag{4.12}$$

We now substitute the optimal control into the dynamics and pose a boundary value problem over the combined state and costate equations with boundary conditions

$$\mathbf{p_i}(t = 0) = \mathbf{p_{i,0}} \tag{4.13}$$

$$\mathbf{p_i}(t = t_f) = \mathbf{p_{i,f}} \tag{4.14}$$

where $\mathbf{p_i} = [p_{i,x} \quad p_{i,y}]^T$ denotes the position vector of an agent. For the Dubin's car, we notice that the solution to the BVP problem is highly sensitive to the initial guess required. In order to obtain an initial guess sufficiently close to the required optimal solution, we leverage the collocation method. We rollout a trajectory based on collocation and use the rolled out trajectory as an initial guess to find a BVP solution. Even this method of solving the optimal rendezvous is highly sensitive to the initial conditions, the mesh points for the initial collocation problem, the mesh points for the BVP solver. There needs to be better solvers and advances in solving boundary value problems to make the framework more robust and usable. The boundary value problem also requires a much larger number of mesh points for an accurate solution and higher computational resources. For these reasons we choose to implement the cooperative framework using collocation alone.

4.3 Main Results

4.3.1 Optimal control problem

For our network of vehicles at an instance of time t, the optimal control inputs are obtained as a solution to the cost function minimization problem subject to dynamical and end point constraints. The optimal control problem can be formulated as

$$\begin{aligned}
&\min_{u} && J = \int_{t_0}^{T_f} L(\boldsymbol{x},,u,t)dt && \text{(4.15a)}\\
&\text{Hamiltonian} && H(\boldsymbol{x},u,t) = L(\boldsymbol{x},u,t) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x},u,t), && \text{(4.15b)}\\
&\text{State Equation} && \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x},u,t) && \text{(4.15c)}\\
&\text{Costate Equation} && -\dot{\boldsymbol{\lambda}} = \frac{\partial H}{\partial \boldsymbol{x}} && \text{(4.15d)}\\
&\text{Stationarity} && \frac{\partial H}{\partial u} = 0 && \text{(4.15e)}\\
&\text{Boundary Conditions} && \boldsymbol{x}(t_0) = \text{given}, \boldsymbol{x}(T_f) = \text{target} && \text{(4.15f)}
\end{aligned}$$

Although the optimization problem does not permit the derivation of an analytical solution, optimal control inputs can be acquired by employing a direct collocation method to solve Eq. (4.15a)-(4.15f). Subsequently, this solution can be parameterized for a required T_f using the knowledge of the current state vector. From our previous work in decentralized leaderless protocols [30], in a manner similar to Theorem 5.1 we stipulate that the communication graph $\mathcal{G}$ must be connected, the consensus evaluation time interval, initial separation between the agents must be such that at least p consensus evaluations must occur where $(p-1)$ is the largest path length between agents in $\mathcal{G}$. This can be done through the use of max-plus algebra.

Theorem 4.3.1. *For 'N' agents connected by a graph $\mathcal{G}$ with maximum path length $(p-1)$, at time $t > p\Delta T_c$, the policy $u_i^*(t, T_f, \boldsymbol{x})$ is optimal $\forall t > p\Delta T_c$ for same time position consensus.*

Proof. Proof given in Appendix C.1 □

Remark 4.3.2. *It is important to note that if the communication graph $\mathcal{G}$ of the agents is connected but not complete, T_f is better understood as a local consensus parameter. This can be more accurately represented as $[\mathbf{t_f}]_i$, where the i^{th} element of $\mathbf{t_f}$ denotes the local consensus value of the protocol, based on the limited information shared among agents.*

4.3.2 Collocation method

In order to obtain a solution to the optimal control problem, we employ a collocation method. This is a direct method because it transcribes the continuous optimal control problem into a finite-dimensional nonlinear programming (NLP) problem. We begin by choosing an appropriate discretization time interval T. For our framework, we utilize the current time-to-go information of an agent 'i' to calculate the number of discrete time intervals till same time position consensus is achieved given by $N_{t,i}$. This is given as

$$N_{t,i} = round\left(\frac{t_{go,i}}{T}\right) \tag{4.16}$$

here $round(.)$ denotes the rounding operation, which adjusts its argument to the nearest integer. Due to the nature of our framework we introduce two new parameters: the discrete controller horizon N_{hor} and the number of samples between successive consensus evaluations N_c. From our previous analysis, these two quantities are set equal by design and will be used interchangeably in the analysis that follows. We obtain the value of these quantities as

$$N_{hor,i} = N_{c,i} = round\left(\frac{\Delta T_{hor}}{T}\right) \tag{4.17}$$

To proceed, we leverage the aforementioned discretization values to derive discrete states and controls, based on a piecewise polynomial assumption over each time interval. An initial estimation of the state trajectory and control input sequence is rolled out. Subsequently, for each time interval, a set of collocation points is selected. These are strategically positioned within the interval to enforce the adherence of system dynamics. The selection of these points is strongly dependent on the specific attributes of the problem at hand. In cases where systems exhibit high nonlinearity, these points may be determined using Legendre or Radau collocation roots [43], both of which are frequently employed in the literature on optimal control due to their effectiveness in such scenarios.

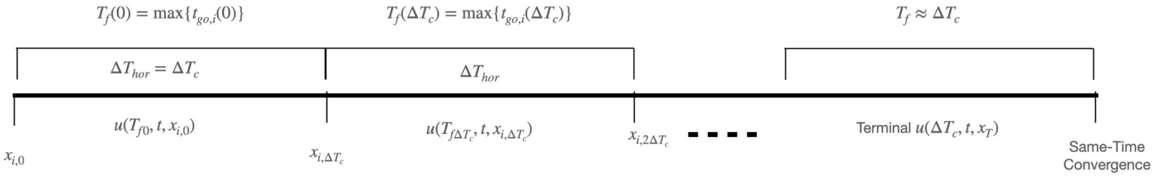

$T_f(0) = \max\{t_{go,i}(0)\}$
$T_f(\Delta T_c) = \max\{t_{go,i}(\Delta T_c)\}$
$T_f \approx \Delta T_c$
$\Delta T_{hor} = \Delta T_c$
ΔT_{hor}
$u(T_{f0}, t, x_{i,0})$
$u(T_{f\Delta T_c}, t, x_{i,\Delta T_c})$
Terminal $u(\Delta T_c, t, x_T)$
$x_{i,0}$
$x_{i,\Delta T_c}$
$x_{i,2\Delta T_c}$
Same-Time Convergence

Figure 4.1. Framework timeline.

We illustrate the timeline of our framework in Fig. 4.1. The effectiveness and equivalence of solving NLPs using collocation as a solution to an optimal control problem is well documented in literature [44].

Theorem 4.3.3. *The solution to the optimal control problem described in Eq.* (4.15a)-(4.15f) *and the optimal control inputs obtained through the method of collocation are equivalent.*

Proof. Proof is given in Appendix C.2 □

The objective function and constraints of the optimal control problem are then transcribed into the form required for an NLP problem. The constraints are the algebraic equations obtained from enforcing the system dynamics, as well as any other constraints on the state or control variables. In our case we enforce constraints on the initial and final position of the vehicles. Finally, we solve the resulting NLP problem using an NLP solver. The solution to the NLP problem gives us the optimal control inputs. In a manner similar to model predictive control, to preserve the time dependent nature of our consensus, we utilize the optimal control solution for N_{hor} discrete time intervals. Once the controller horizon is reached, we recompute the optimal trajectory based on the protocol time-to-go consensus value and the state of the vehicle at that time for the next N_{hor} discrete time steps. This approach ensures an optimal, gain free, finite time, time dependent solution to the same time position consensus problem.

Remark 4.3.4. *As the $N_{t,i}$ approaches the value of $N_{hor,i}$, we initiate a terminal approach. we use the state information and final $N_{t,i}$ value to synthesize the optimal control inputs putting a pause to our time-to-go evaluation and using the computed optimal controls to achieve same time position consensus with a target*

We present our framework in the form of an algorithm5 for an agent vehicle in a complete graph communication network where information can be exchanged between any 2 agents in the network

Algorithm 5 Optimal same-time position consensus

```
Require: $\boldsymbol{x}(0) = \text{known}, \boldsymbol{x}_T = \text{known}, [T, N_{hor}, N_c] = \text{chosen}$
  $T_f \leftarrow protocol(t_{go}(0))$
  i = 1, j = 1
  while $r(t) \leq \epsilon$ do                                    ▷ From Definition2.1.1
      $U_c(1 : N_t) = \text{Solve Optimization Problem}(T_f, N_t, \boldsymbol{x}, t)$
      while $j < N_{hor}$ do
          $U = U_c(j)$
          $\boldsymbol{x}(j+1) \leftarrow \boldsymbol{f}(\boldsymbol{x}(j), U, t)$
          $j \leftarrow j + 1$
      end while
      $i \leftarrow i + 1$
      $T_f \leftarrow protocol(t_{go,i}(N_c T))$
      $N_t \leftarrow round(t_{go,i}(N_c)/T)$
  end while
```

4.3.3 Choice of Protocol

The max time protocol is defined by setting the T_f of the framework to the maximum of all time-to-go values of agents in the network i.e

$$T_f(k\Delta T_c) = \max(t_{go,i}(k\Delta T_c)) \quad \forall i \in \mathcal{I} \tag{4.18}$$

where $\mathcal{I}$ denotes the set of all agents.

Similarly the min time protocol is defined by setting the T_f of the framework to the minimum of all time-to-go values of agents in the network i.e

$$T_f(k\Delta T_c) = \min(t_{go,i}(k\Delta T_c)) \quad \forall i \in \mathcal{I} \tag{4.19}$$

Remark 4.3.5. *As discussed in Section 2.5, for a connected but not complete graph communication network, the max time protocol time-to-go* $\mathbf{t_f}$ *is defined as:*

$$\mathbf{t_f}(k\Delta T_c) = \max_{i \in RS(\mathbf{AT}_{go})} [\mathbf{AT}_{go}(k\Delta T_c)]_{ij} \tag{4.20}$$

Similarly, the min time protocol time-to-go is given by:

$$\mathbf{t_f}(k\Delta T_c) = \min_{i \in RS(\mathbf{AT}_{go})} [\mathbf{AT}_{go}(k\Delta T_c)]_{ij} \tag{4.21}$$

An extensive discussion on transforming the nonlinear problem of protocol time-to-go consensus into a linear one, along with its subsequent analysis, is presented in Section 2.5. A key result we utilize is that for a maximum path length p *in graph* $\mathcal{G}$*, the* $[\mathbf{t_f}]_i$ *equals the maximum time-to-go of all agents in* $\mathcal{I}$ *when using the max time protocol and the minimum of all the agents in* $\mathcal{I}$ *when using the min time protocol after* $(p-1)$ *consensus evaluation intervals or* $p\Delta T_c$ *seconds. This insight is crucial for the results derived in Theorem 4.3.1.*

The consensus and positional convergence frameworks may exhibit variations within an optimal context; however, our interest lies in discerning the emergent behaviors predicated on the chosen protocols. From our findings in prior works, the max time protocol, albeit slower, demonstrates a more reliable consensus process under velocity constraints. In contrast, the min time protocol facilitates a quicker consensus in position but is susceptible to breaking down when faced with speed limitations. We primarily advocate for the implementation of the min time to go and max time-to-go protocols within an optimal, leaderless, decentralized consensus framework.

Remark 4.3.6. *Traditional dynamical time-to-go estimates, based on linear motion assumptions, can yield inaccuracies and singularities within optimal control frameworks. Hence, we employ 'objective time' or 'mission time' in complex dynamical systems, which is derived from multiple factors, including the vehicle's state.*

4.3.4 Objective based optimal rendezvous

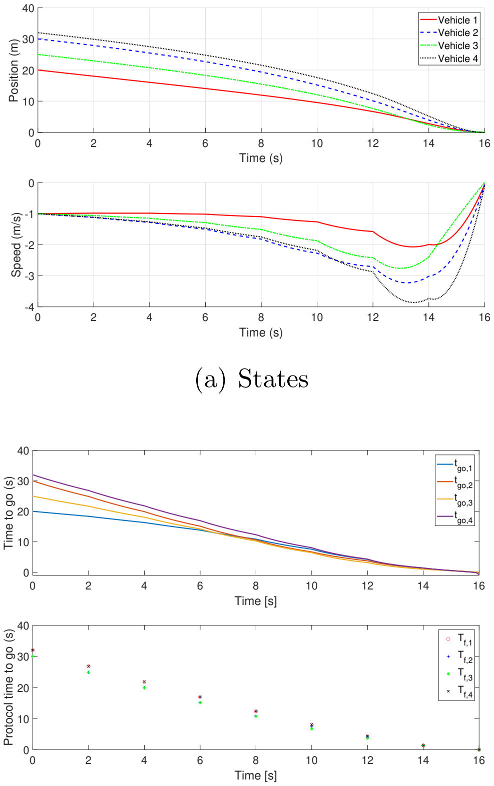
In our analysis thus far, we have primarily focused on decentralized, leaderless approaches to the problem of achieving simultaneous position consensus. However, the proposed framework of time-to-go consensus and position convergence is versatile, permitting extension to encompass objective-based requirements. To illustrate this augmentation to our framework, let's consider a scenario involving the rendezvous of multiple ground vehicles with a descending quadrotor. In this particular situation, the protocol time-to-go consensus value is not dictated by the states of the ground vehicles within the network. Instead, it is determined by the quadrotor, an external entity that remains unaffected by the vehicles' approach to the target location.

Given this context, we initially delineate the feasible set of vehicles. This set is defined based on the intended landing location of the quadrotor and its time-to-go value. Specifically, the feasible set comprises those vehicles that have the capability to reach the target location in time for the rendezvous. This feasibility is influenced by a variety of factors, such as available fuel or battery power, speed limits, and other physical constraints. Our analysis proceeds by assuming that simultaneous rendezvous is performed with vehicles from this feasible set. This scenario demonstrates the adaptability of our framework to cater to specific mission objectives. We utilize a quadrotor model using a nonlinear geometric controller that tracks an optimal polynomial trajectory similar to the design presented by Mellinger et al. [45]. These quadrotors are used for survey and reconnaissance and designed with a geometric controller to make aggressive maneuvers to get into difficult to reach spots. For our objective based optimal rendezvous, we set the value of T_f based on the descent of the quadrotor and use this to generate optimal control for the ground vehicles to rendezvous.

4.4 Simulation Results

4.4.1 Double Integrator Agents

We consider optimal rendezvous of multiple double integrator agents to a target location to illustrate our framework.



(a) States

(b) Time-to-go

Figure 4.2. Double integrator network.

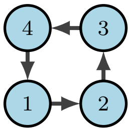


Figure 4.3. Graph Connectivity: Directed Cycle.

The vehicles are initialised with positions and velocities given in Eq. 4.22 and connected to each other through a directed cycle shown in Fig. 4.3. They exchange time to go information every 2 sec.

$$\begin{bmatrix} x_1^T \\ x_2^T \\ x_3^T \\ x_4^T \end{bmatrix} = \begin{bmatrix} 20 & 1 \\ 30 & 1 \\ 25 & 1 \\ 32 & 1 \end{bmatrix} \tag{4.22}$$

The results are consolidated in Fig. 4.2. As Fig. 4.2(a) indicates, same time position consensus is successfully attained. Nevertheless, an analysis of Fig. 4.2(b) reveals inaccuracies in the time-to-go estimates, which are derived from the vehicles' states.

In generating these estimates, the assumption made was that the agents would persist at the same velocity until consensus was reached. This premise, within an optimal control framework, results in agents reaching their destination considerably faster than anticipated. This phenomenon becomes apparent upon scrutinizing the analytical expression derived in Eq. (4.6). Although agents achieve $x_i(T_f) = x_{\text{target}}$, they exhibit non-zero acceleration at $t = 0$. Consequently, the nature of our optimal control solution inherently accelerates or decelerates the vehicles. During subsequent time-to-go consensus evaluations, this exerts a compounding influence, adjusting the time-to-go estimates either below or above the protocol's designated value for all agents.

This discrepancy underscores the necessity for objective time or mission time in such frameworks. The topology of the network graph clearly indicates a maximum path length of three. In concert with this observation, Fig. 4.2(b) illustrates that four consensus time intervals are required to achieve a uniform protocol-based T_f value across the network vehicles. This outcome aligns with our established theory regarding connected graph communication topologies in the context of simultaneous consensus problems.

4.4.2 Dubin's car rendezvous

We consider a homogeneous system of 3 agents with modified Dubin's car dynamics given by Eq.(4.1) executing the max time protocol. The agents use a dynamic time-to-go estimate given as

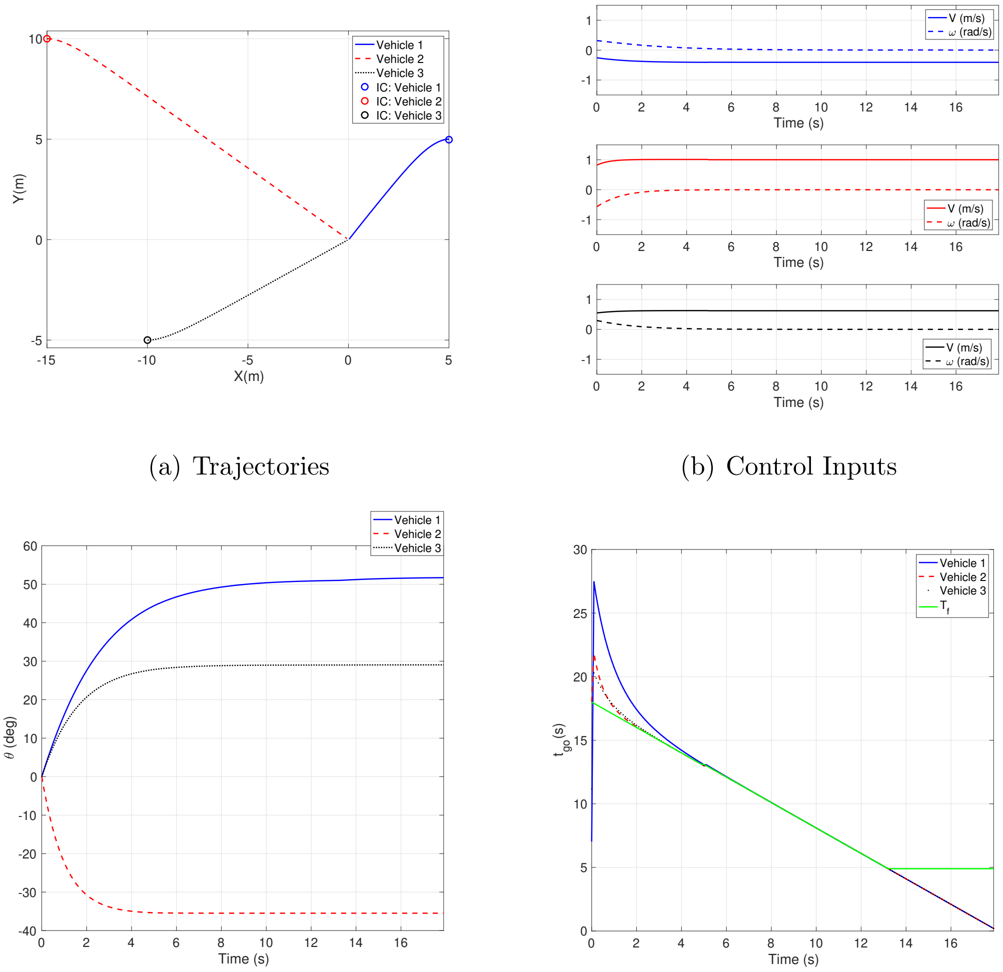
$$t_{go,i} = -\frac{r_i}{V_{i,r}}$$

here r_i is the distance of agent 'i' to the target and $V_{i,r}$ denotes the closing velocity. The horizon of the controller is 5 sec and the time-to-go consensus occurs at this interval i.e $\Delta T_c = 5sec$.

The initial states are given in Eq.4.23

$$\begin{bmatrix} x_1^T \\ x_2^T \\ x_3^T \end{bmatrix} = \begin{bmatrix} 5 & 5 & 0 \\ -15 & 10 & 0 \\ -10 & -5 & 0 \end{bmatrix} \tag{4.23}$$

As observed in Fig. 4.4(d), the agents require only a single consensus interval, approximately 5 seconds, to establish a time-to-go consensus. Following this period, as corroborated by Fig. 4.4(a), the agents reach the target location at the same time at the origin. This coordinated arrival is facilitated by the optimization of control inputs, as illustrated in Fig. 4.4(b). Furthermore, Fig. 4.4(d) reveals that our terminal approach is initiated when the time-to-go for the agents approximates ΔT_C. At this juncture, the consensus evaluations are temporarily halted, as indicated by a constant value of T_f, and the computed optimal control is employed to ensure same-time consensus. However, constructing a finite dynamic time-to-go estimate may not



(a) Trajectories

(b) Control Inputs

(c) Orientation

(d) Time-to-go

Figure 4.4. Dubin's car multi vehicle optimal rendezvous.

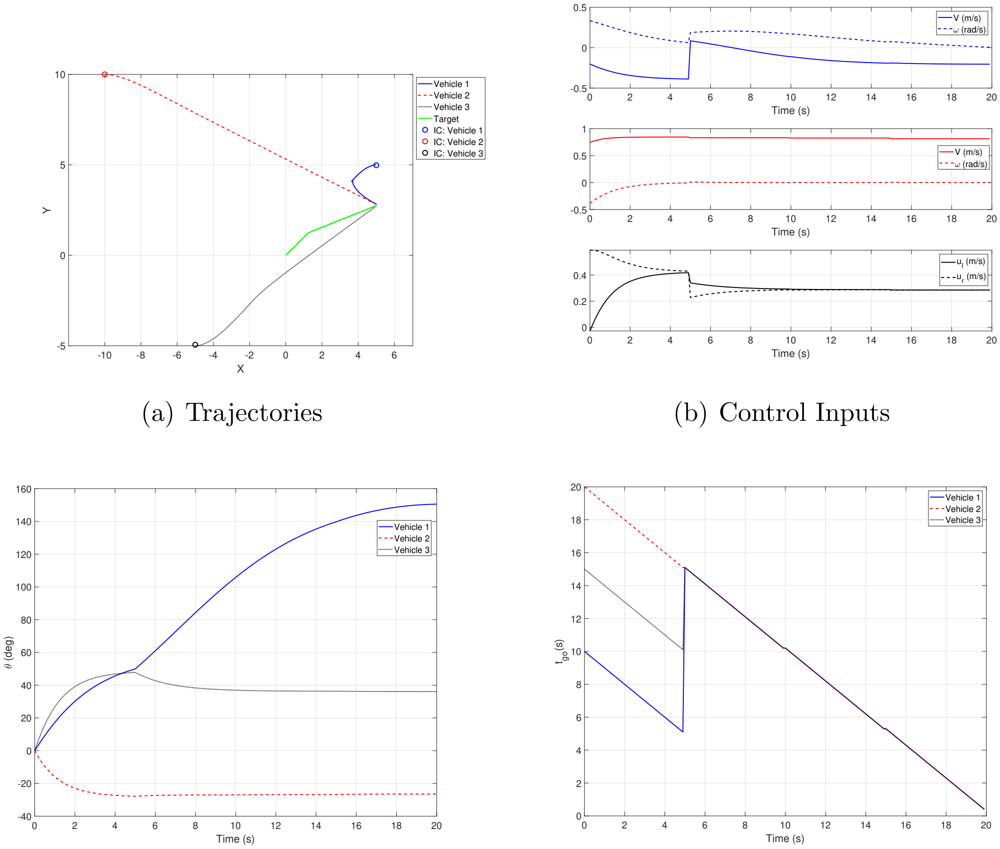
always be feasible, particularly with a moving target. In such scenarios, we pivot to using mission time, which will be illustrated in the subsequent section.

4.4.3 Heterogeneous moving target rendezvous

Three vehicles with complete graph communication are simulated. Two of these vehicles have modified Dubin's car dynamics given by Eq.(4.1). Vehicle 3 has differential drive dynamics given by Eq. (4.2). The time-to-go estimates are determined based on requirements and are hence classified as mission time rather than a dynamic time-to-go estimate.

The initial states are given in Eq.(4.24)

$$\begin{bmatrix} x_1^T \\ x_2^T \\ x_3^T \end{bmatrix} = \begin{bmatrix} 5 & 5 & 0 \\ 10 & -10 & 0 \\ -5 & -5 & 0 \end{bmatrix} \tag{4.24}$$



(a) Trajectories

(b) Control Inputs

(c) Orientation

(d) Mission time

Figure 4.5. Max time: Heterogeneous multi vehicle optimal rendezvous.

Max time protocol

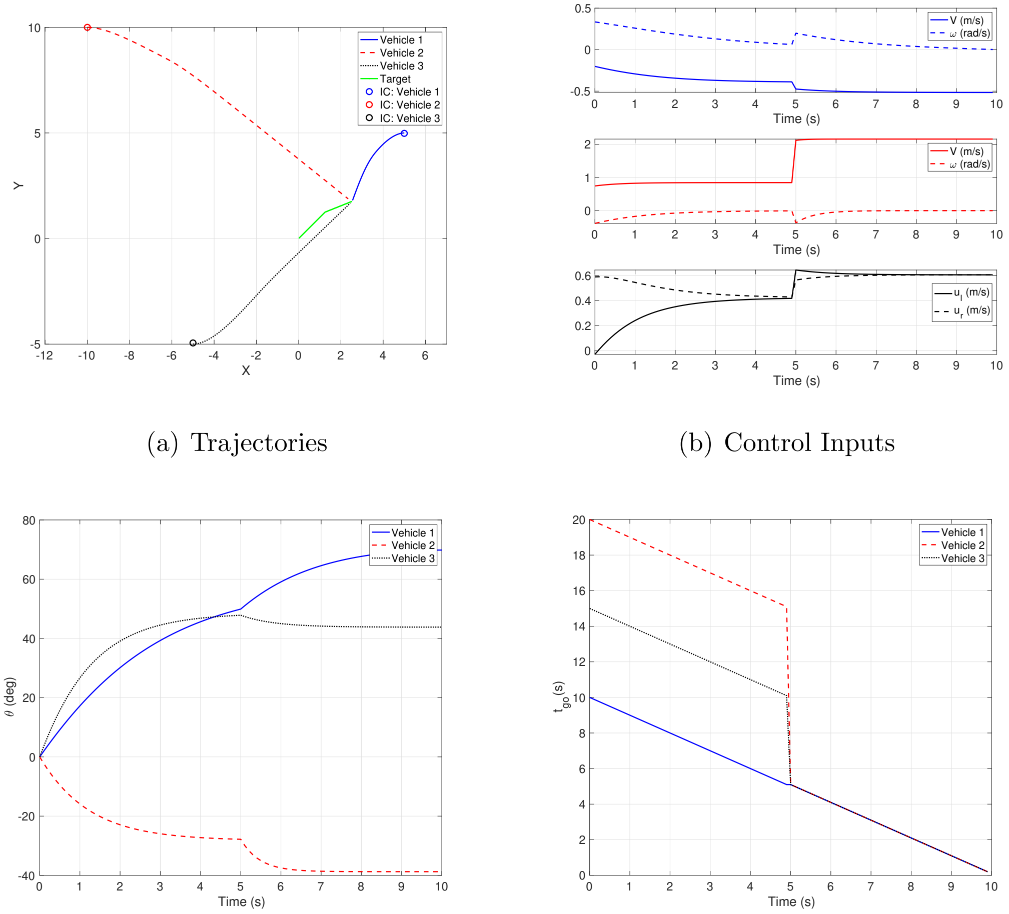
The simulation results are summarized in Fig.4.5. The framework is applied after the first consensus time interval ($\Delta T_c = 5sec$) We notice in Fig. 4.5(a) that the vehicles achieve same time position consensus with a moving target. The target moves at velocity $(v_x, v_y) = (0.25, 0.25)$ for the first consensus time interval, then switches to a velocity of $(v_x, v_y) = (0.1, 0.25)$. Despite the heterogeneous nature of consensus, the time-to go, while initially in disagreement, converge to the protocol based maximum time to go in one consensus time interval as shown in Fig.4.5(d) while all the control inputs are bounded and minimized.

Min time protocol

For this example we next, look at the results using the min time protocol. We notice that the control effort required in Fig. 4.6(b) is higher compared to the max time protocol. Similar to the max time protocol the dynamic estimate of time-to-go is unreliable but the agents arrive at the target location simultaneously as seen in Fig. 4.6(d) and Fig. 4.6(a).

4.4.4 Objective based optimal rendezvous

We illustrate the objective based rendezvous by looking at the same time position consensus between a quadrotor, a ground vehicle with differential drive dynamics and another ground vehicle with modified Dubin's car dynamics. The quadrotor executes an optimal minimum snap descent as described by Liu et al. [46] to a target location at $(10m, 10m, 0m)$. The ground vehicles utilise the time-to-go information from the quadrotor to calculate the optimal control required to achieve rendezvous with the target. We observe the trajectory plot shown in Fig. 4.7(a). Heterogeneous,



(a) Trajectories

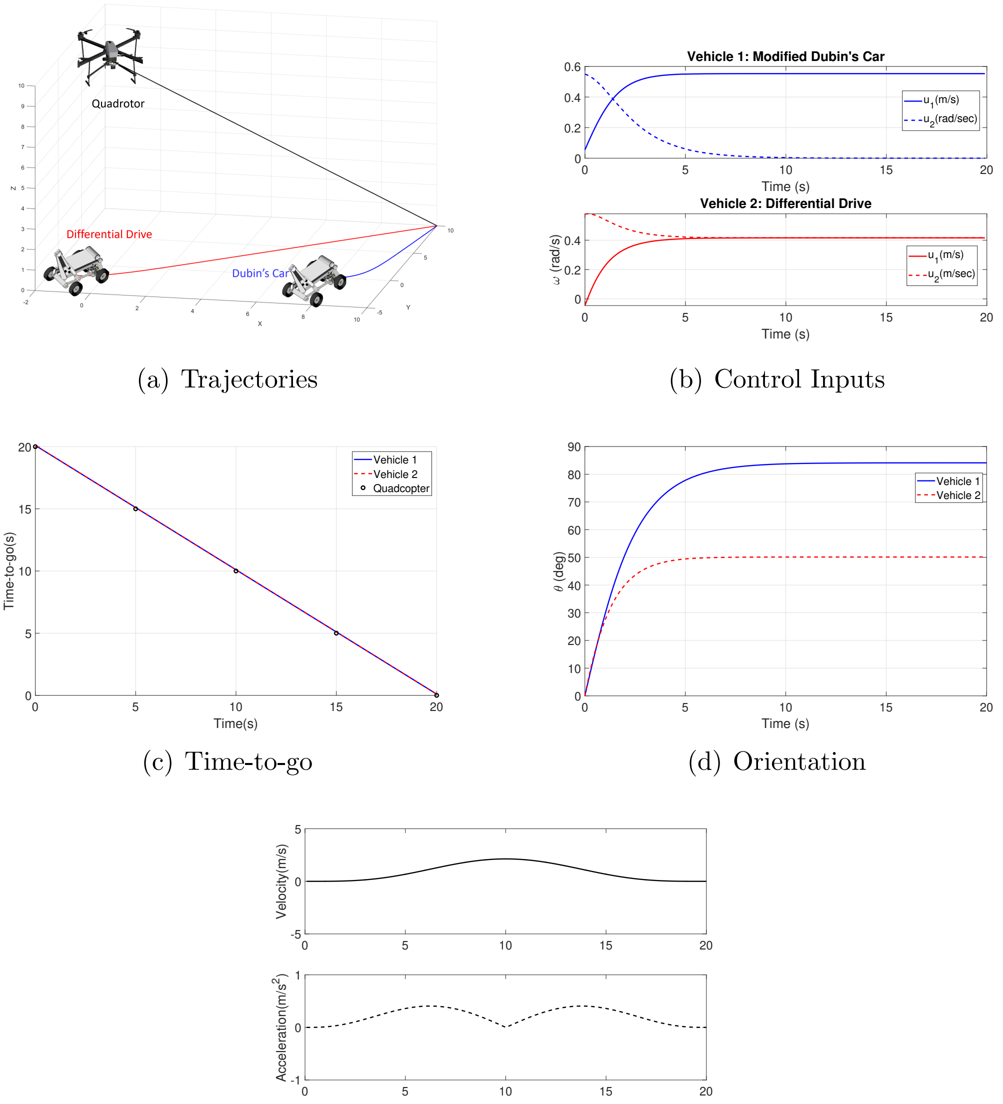
(b) Control Inputs

(c) Orientation

(d) Mission time

Figure 4.6. Min time: Heterogeneous multi vehicle optimal rendezvous.

same time position consensus is achieved among the two vehicles to the landing location of the quadrotor.



(a) Trajectories

(b) Control Inputs

(c) Time-to-go

(d) Orientation

(e) Quadrotor Approach

Figure 4.7. Objective based optimal rendezvous.

4.5 Summary

The primary contributions of this chapter are outlined as follows:

1. We have developed a novel framework for generating and solving finite-time optimal control problems using collocation, specifically designed for cooperative

rendezvous scenarios. This framework effectively addresses the complexities inherent in these situations.

2. A key feature of our approach is the implementation of a reevaluation process activated at the end of each time-to-go consensus interval. This process allows for a dynamic and adaptable control strategy, incorporating real-time updates from agents, enhancing the resilience and flexibility of our framework in response to rapid state changes and evolving conditions.
3. Our research presents an extensive analysis of an optimization problem with endpoint boundary conditions. This approach, diverging from traditional Model Predictive Control methods, ensures simultaneous position convergence and unites optimal control principles with consensus protocols, yielding a more versatile and efficient solution.
4. We extend our framework to include objective-based optimal rendezvous scenarios, adopting a leader-follower strategy focused on achieving finite-time objectives. This adaptability enables the application of our model to a wide range of scenarios, further enhancing its operational versatility.

Together, these innovations highlight the adaptability and real-time response capabilities of our solution. The model's precision and efficiency markedly improve its performance in managing diverse vehicle capabilities and dynamic operational environments, paving the way for robust and efficient cooperative multi-vehicle systems.

Chapter 5

Summary and Closing Remarks

5.1 Summary

This dissertation has successfully developed and implemented a cooperative framework for secure rendezvous, leveraging time-to-go consensus to achieve positional rendezvous within finite time frames. The approach is designed to be secure against malicious external entities, preventing them from deducing the agents' locations or speeds from shared communications. We have thoroughly explored and implemented max time, min time, and achievable time protocols, analyzing consensus and emergent behavior in leaderless multi-agent systems across various connected graph communication topologies.

Furthermore, the application of the max time protocol has been extended to address the secure missile target salvo problem for both stationary and maneuvering targets. This includes a novel terminal approach mechanism to modulate relative velocities during engagement. The methodology also caters to sparser connected graph communications among the pursuers.

A comprehensive performance evaluation of this framework was conducted, incorporating target estimation methods like the Extended Kalman Filter and the Unscented Kalman Filter. Through extensive comparative studies, the most effective estimation framework was identified.

Additionally, an optimal solution for secure, finite-time rendezvous was achieved, utilizing max time and min time protocols within a leaderless framework, implemented using a collocation method. This was extended to include connected graph communication topologies, and a centralized, objective-based rendezvous method was developed, aligning with existing literature and leader-follower methods.

5.2 Closing Remarks

As this dissertation reaches its conclusion, it's crucial to recognize the significant progress achieved in the realm of cooperative secure finite-time rendezvous in multi-agent systems, along with the numerous challenges and prospects for future exploration.

The research journey has been both demanding and fulfilling, fostering the hope that the insights gleaned will contribute significantly to future advancements in the cooperative control of multi-agent systems.

The findings from this study highlight the critical role of finite-time, time-dependent methods in cooperative control, especially when employing leaderless, sparse communication networks for optimal secure rendezvous. Future research is expected to build upon these findings, exploring the reachability aspects of the proposed optimal cooperative framework, examining scenarios with non-connected graph communication to identify agent subsets capable of achieving rendezvous, and extending these results to even sparser communication networks. Additionally, the potential for tokenizing other protocols and analyzing the impact of weighted communication on consensus protocols presents exciting avenues for further study.

In closing, my heartfelt gratitude goes to my advisor, Dr. Subbarao, for his relentless guidance and support throughout my research journey. I am also thankful to the Office of Naval Research (ONR) for their financial support under Grant No. N00014-18-1-2215. My sincere appreciation extends to Dr. Diganta Bhattacharjee, my dissertation committee members, and the faculty of the Mechanical and Aerospace

Engineering department at The University of Texas at Arlington for their invaluable guidance and support.

APPENDIX A

Appendix A

A.1 Proof of Lemma 2.3.2

Proof. Under the max-time protocol, we have

$$\delta_i(k+1) = \delta_i(k)\left(1 - \frac{\Delta T}{t_{go,C}(k)}\right), \forall i \in \mathcal{P}$$

where $t_{go,C}(k) = \max\{t_{go,i}(k)\}$. At time step k, let

$$t_{go,C}(k) = t_{go,i_k}(k) = \frac{-\delta_{i_k}(k)\Delta T}{\delta_{i_k}(k) - \delta_{i_k}(k-1)}.$$

for some $i_k \in \mathcal{P}$. Hence,

$$\delta_i(k+1) = \delta_i(k)\left(1 + \frac{\delta_{i_k}(k) - \delta_{i_k}(k-1)}{\delta_{i_k}(k)}\right)$$

However, we have

$$\delta_{i_k}(k) = \delta_{i_k}(k-1) + u_{i_k}(k-1)\Delta T \leq \delta_{i_k}(k-1) - u_{\min}\Delta T$$

due to the constraint on the control speed. This leads to

$$\delta_i(k+1) \leq \delta_i(k)\left(1 - \frac{u_{\min}\Delta T}{\delta_{i_k}(k)}\right)$$

Proceeding in this manner, we derive

$$\delta_i(k+1) \leq \delta_i(0)\left(1 - \frac{u_{\min}\Delta T}{\delta_{i_0}(0)}\right)\left(1 - \frac{u_{\min}\Delta T}{\delta_{i_1}(1)}\right)\cdots\left(1 - \frac{u_{\min}\Delta T}{\delta_{i_k}(k)}\right)$$

The above inequality can be further simplified by using the following fact:

$$\left(1 - \frac{u_{\min}\Delta T}{\delta_{i_0}(0)}\right)\left(1 - \frac{u_{\min}\Delta T}{\delta_{i_1}(1)}\right)\cdots\left(1 - \frac{u_{\min}\Delta T}{\delta_{i_k}(k)}\right) < \left(1 - \frac{u_{\min}\Delta T}{\delta_{\max}(0)}\right)^{k+1}$$

where $\delta_{\max}(0)$ is the maximum initial separation among the agents, satisfying $\delta_{\max}(0) \geq \delta_{i_0}(0)$ and $\delta_{\max}(0) > \delta_i(k), \forall i \in \mathcal{P}, k \in \mathbb{Z}_\star\backslash\{0\}$. Combining these, we have

$$\frac{\delta_i(k+1)}{\delta_i(0)} < \left(1 - \frac{u_{\min}\Delta T}{\delta_{\max}(0)}\right)^{k+1}$$

which can be equivalently expressed as

$$\frac{\delta_i(k)}{\delta_i(0)} < \left(1 - \frac{u_{\min}\Delta T}{\delta_{\max}(0)}\right)^k$$

Now, substituting $k = k_f$ and $\delta_i(k_f) = \epsilon$, we have

$$\frac{\epsilon}{\delta_i(0)} < \left(1 - \frac{u_{\min}\Delta T}{\delta_{\max}(0)}\right)^k$$

Now, setting $z = \frac{u_{\min}\Delta T}{\delta_{\max}(0)}$, we can rewrite the above inequality as

$$\log_{(1-z)}\left(\frac{\epsilon}{\delta_i(0)}\right) < k_f$$

Also, we need $z < 1$ for the above to hold. It leads to

$$\delta_{\max}(0) > u_{\min}\Delta T$$

□

A.2 Proof of Lemma 2.3.3

Proof. Under the min-time protocol, we have

$$\delta_i(k+1) = \delta_i(k)\left(1 - \frac{\Delta T}{t_{go,C}(k)}\right), \forall i \in \mathcal{I}$$

where $t_{go,C}(k) = \min\{t_{go,i}(k)\}$. At time step k, let

$$t_{go,C}(k) = t_{go,i_k}(k) = \frac{-\delta_{i_k}(k)\Delta T}{\delta_{i_k}(k) - \delta_{i_k}(k-1)}.$$

for some $i_k \in \mathcal{I}$. Hence,

$$\delta_i(k+1) = \delta_i(k)\left(1 + \frac{\delta_{i_k}(k) - \delta_{i_k}(k-1)}{\delta_{i_k}(k)}\right).$$

However, we have

$$\delta_{i_k}(k) = \delta_{i_k}(k-1) + u_{i_k}(k-1)\Delta T \geq \delta_{i_k}(k-1) - u_{\max}\Delta T$$

due to the constraint on the control speed. This leads to

$$\delta_i(k+1) \geq \delta_i(k) \left(1 - \frac{u_{\max}\Delta T}{\delta_{i_k}(k)}\right).$$

Proceeding in this manner, we derive

$$\delta_i(k+1) \geq \delta_i(0) \left(1 - \frac{u_{\min}\Delta T}{\delta_{i_0}(0)}\right) \left(1 - \frac{u_{\min}\Delta T}{\delta_{i_1}(1)}\right) \\ \cdots \left(1 - \frac{u_{\min}\Delta T}{\delta_{i_k}(k)}\right).$$

The above inequality can be further simplified by using the following fact:

$$\left(1 - \frac{u_{\max}\Delta T}{\delta_{i_0}(0)}\right) \left(1 - \frac{u_{\max}\Delta T}{\delta_{i_1}(1)}\right) \cdots \left(1 - \frac{u_{\max}\Delta T}{\delta_{i_k}(k)}\right) \\ > \left(1 - \frac{u_{\max}\Delta T}{\delta_{\min}(0)}\right)^{k+1}$$

where $\delta_{\min}(0)$ is the minimum initial separation among the agents, satisfying $\delta_{\min}(0) \leq \delta_{i_0}(0)$ and $\delta_{\min}(0) < \delta_i(k), \forall i \in \mathcal{I}, k \in \mathbb{Z}_\star \backslash \{0\}$. Combining these, we have

$$\frac{\delta_i(k+1)}{\delta_i(0)} > \left(1 - \frac{u_{\max}\Delta T}{\delta_{\min}(0)}\right)^{k+1}$$

which can be equivalently expressed as

$$\frac{\delta_i(k)}{\delta_i(0)} > \left(1 - \frac{u_{\max}\Delta T}{\delta_{\min}(0)}\right)^{k}.$$

Now, substituting $k = k_f$ and $\delta_i(k_f) = \epsilon$, we have

$$\frac{\epsilon}{\delta_i(0)} > \left(1 - \frac{u_{\max}\Delta T}{\delta_{\min}(0)}\right)^{k}.$$

Now, setting $z = \frac{u_{\max}\Delta T}{\delta_{\min}(0)}$, we can rewrite the above inequality as

$$\log_{(1-z)}\left(\frac{\epsilon}{\delta_i(0)}\right) > k_f.$$

Also, we need $z < 1$ for the above to hold. It leads to

$$\delta_{\min}(0) > u_{\max}\Delta T$$

which completes the proof. □

A.3 Proof of Theorem 2.3.5

Proof. If $\mathcal{M} = \emptyset$ for the entire duration of consensus, the min-time protocol has been employed and none of the agents are traveling with their time-to-go equal to their corresponding critical time-to-go ($t_{cr,i}^{min}(k)$) value. Thus, the agents achieve same-time consensus (cf. Lemma 2.3.3).

Now, consider a time step k when $\mathcal{M} \neq \emptyset$, i.e., the min-time protocol has been applied till some arbitrary time step k at which point $\mathcal{M} \neq \emptyset$. Hence, at this time step k, the consensus value of time-to-go becomes

$$t_{go,C}(k\Delta T_c) = \frac{\delta_{i_M}(k\Delta T_c)}{u_{\max}}$$

where the subscript i_M denotes the agent in $\mathcal{M}$ with the largest value of estimated time-to-go in $\mathcal{M}$ (in other words, largest $\delta_i(k)$ among all the agents in $\mathcal{M}$). Now, for any arbitrary agent $i_L \in \mathcal{L}$, the commanded control is given by

$$u_{i_L}(k\Delta T_c) = \frac{-\delta_{i_L}(k\Delta T_c)}{\delta_{i_M}(k\Delta T_c)} u_{\max}$$

Since the group was initialized with the min-time protocol, we can conclude that $\delta_{i_L}(k\Delta T_c) < \delta_{i_M}(k\Delta T_c)$. Therefore, the commanded control for any agent $i_L \in \mathcal{L}$ satisfies the following:

$$|u_{i_L}(k\Delta T_c)| = \frac{\delta_{i_L}(k\Delta T_c)}{\delta_{i_M}(k\Delta T_c)} u_{\max} < u_{\max}$$

Also, under Assumption 2.3.1, we have $|u_{i_L}(k\Delta T_c)| \geq u_{\min}$. This would imply that all agents in $\mathcal{L}$ can achieve convergence to the target through the achievable-time protocol. Further, using Lemma 2.3.2 it can be shown that all agents in $\mathcal{M}$ and hence all agents in $\mathcal{I}$ achieve same-time consensus through the achievable-time protocol. □

A.4 Proof of Theorem 2.5.1

Proof. On utilizing the definition of matrix multiplication in max-plus algebra from Section 2.2.3, we recall an important property of the modified adjacency matrix. If a path of length l exists between two nodes i and j in the graph $\mathcal{G}(\bar{\mathbf{A}})$,

$$[\bar{\mathbf{A}}^l]_{i,j} = e \tag{A.1}$$

where $[\cdot]_{i,j}$ denotes the (i, j) element of the matrix. It is also noted that for $m > 1$, $[\bar{\mathbf{A}}^{l+m}]_{i,j} = e$ considering the addition and multiplication operations in max-plus algebra. This implies that for a simple path of length p in graph $\mathcal{G}$, using (A.1), we have

$$\bar{\mathbf{A}}^{[p+b]} = \mathbf{E}\ \forall b > 0 \tag{A.2}$$

where $\mathbf{E}$ is the neutral element for matrix multiplication in max-plus algebra. The minimum length of a path between nodes i and j is denoted by $|i, j|_{\min}$. Using Corollary 4.2 from the work done by Nejad et al. [47], a proof by contradiction yields

$$p = \max_{i,j \in \mathcal{I}} |i, j|_{\min} \tag{A.3}$$

The value p is the maximum of all lengths of simple paths in the graph $\mathcal{G}$.

The time-to-go consensus vector $\mathbf{t_{go,C}}$ for some time step $k_c > 0$ given by (2.23) can be written in max-plus algebra as

$$\mathbf{t_{go,C}}(k_c \Delta T_c) = \bar{\mathbf{A}} \otimes \mathbf{t_{go}}(k_c \Delta T_c) \tag{A.4}$$

Let $\mathbf{1} \in \mathbb{R}^{N \times 1}$ denote a vector of ones. Using the control acceleration (2.17), we know that the error in time-to-go $\eta_i(t)$ converges to zero in at most ΔT_c seconds, leading to a consensus in time-to-go estimates. This results in

$$\mathbf{t_{go,C}}(k\Delta T_c) = \bar{\mathbf{A}} \otimes \{\bar{\mathbf{A}} \otimes \mathbf{t_{go}}[(k-1)\Delta T_c] - \mathbf{1}\Delta T_c\}$$

Applying this recursively for some $k \geq p$ we obtain

$$\mathbf{t_{go,C}}(k\Delta T_c) = \bar{\mathbf{A}} \otimes \{\bar{\mathbf{A}}^{[k-1]} \otimes \mathbf{t_{go}}(0) - \mathbf{1}(k)\Delta T_c\} \tag{A.5}$$

In order to use (A.2), a simple path must exist within the graph.

Part-1.a)

Since a simple path must exist, the graph $\mathcal{G}$ must be connected to ensure that a path exists between any two nodes. Under this condition, for max-time consensus to occur, there must be at least q consensus evaluations where q is the length of the simple path. From the properties of connected graphs, if the maximum length of a simple path in the graph is known, we have $q = p$. This gives us the first sufficient condition.

Part-1.b)

For consensus to occur among all agents, we require p consensus evaluations. This is sufficient to state that consensus will occur in the time-to-go estimates of the agents, leading to our second condition:

$$\max\{\mathbf{t_{go}}(0)\} \geq p\Delta T_c$$

Part-2)

For a time step $k \geq p$, using (A.2), (A.5) modifies to

$$\mathbf{t_{go,C}}(k\Delta T_c) = \mathbf{E} \otimes \mathbf{t_{go}}(k\Delta T_c) \tag{A.6}$$

This equation can also be expressed as

$$[\mathbf{t}_{\mathbf{go},\mathbf{C}}(k_c\Delta T_c)]_i = \max_{j\in\mathcal{I}}([\mathbf{t}_{go}(k_c\Delta T_c)]_j) \quad \forall i \in \mathcal{I}, k_c \geq p \tag{A.7}$$

where $\mathcal{I}$ denotes the set of all N agents. □

A.5 Proof of Theorem 2.5.2

Proof. Utilizing the definition of matrix multiplication in min-plus algebra, as detailed in Section 2.2.4, we recall a critical property of the modified adjacency matrix $\bar{\mathbf{A}}_{\mathbf{min}}$ in the graph $\mathcal{G}(\bar{\mathbf{A}}_{\mathbf{min}})$. If a path of length l exists between two nodes i and j,

$$[\bar{\mathbf{A}}^{l}_{\mathbf{min}}]_{i,j} = e_m \tag{A.8}$$

where $[\cdot]_{i,j}$ represents the (i, j) element of the matrix. For a simple path of length p in $\mathcal{G}$, we have:

$$\bar{\mathbf{A}}^{[p+b]}_{\mathbf{min}} = \mathbf{E}_{\mathbf{m}} \; \forall b > 0 \tag{A.9}$$

where $\mathbf{E}_{\mathbf{m}}$ is the neutral element in min-plus algebra.

Similar to the max-time case in Theorem 2.5.1, the time-to-go consensus vector $\mathbf{t}_{\mathbf{go},\mathbf{C}}$ for a time step $k_c > 0$ can be expressed in min-plus algebra as:

$$\mathbf{t}_{\mathbf{go},\mathbf{C}}(k_c\Delta T_c) = \bar{\mathbf{A}}_{\mathbf{min}} \otimes \mathbf{t}_{\mathbf{go}}(k_c\Delta T_c) \tag{A.10}$$

Let $\mathbf{1} \in \mathbb{R}^{N\times 1}$ denote a vector of ones. From the control acceleration (2.17), the error in time-to-go $\eta_i(t)$ converges to zero in at most ΔT_c seconds. Hence,

$$\mathbf{t}_{\mathbf{go},\mathbf{C}}(k\Delta T_c) = \bar{\mathbf{A}}_{\mathbf{min}} \otimes \{\bar{\mathbf{A}}_{\mathbf{min}} \otimes \mathbf{t}_{\mathbf{go}}[(k-1)\Delta T_c] - \mathbf{1}\Delta T_c\}$$

Applying this recursively for $k \geq p$, we get:

$$\mathbf{t_{go,C}}(k\Delta T_c) = \mathbf{\bar{A}_{min}} \otimes \{\mathbf{\bar{A}_{min}^{[k-1]}}\mathbf{t_{go}}(0) - \mathbf{1}(k)\Delta T_c\} \tag{A.11}$$

Part-1.a)

For the min-time consensus, given that $\mathcal{G}$ is connected, a path exists between any two nodes. This necessitates at least q consensus evaluations, but as we know from the properties of connected graphs, $q = p - 1$ where p is the maximum path length.

Part-1.b)

For consensus among all agents, p consensus evaluations are required, ensuring that consensus occurs in the time-to-go estimates. Thus, we have:

$$\min\{\mathbf{t_{go}}(0)\} \geq p\Delta T_c$$

Part-2)

For $k_c > p - 1$, using (A.9), (A.11) simplifies to:

$$\mathbf{t_{go,C}}(k_c\Delta T_c) = \mathbf{E_m} \otimes \mathbf{t_{go}}(k_c\Delta T_c) \tag{A.12}$$

This can be equivalently expressed as:

$$[\mathbf{t_{go,C}}(k_c\Delta T_c)]_i = \min_{j\in\mathcal{I}}\{[\mathbf{t}_{go}(k_c\Delta T_c)]_j\} \quad \forall i \in \mathcal{I}, k_c \geq p$$

where $\mathcal{I}$ denotes the set of all agents. □

APPENDIX B

Appendix B

B.1 Jacobian for Engagement Dynamics

The Jacobian of the nonlinear system with respect to the state vector is given as follows

$$\boldsymbol{F} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ f_{21} & 0 & 0 & f_{24} & 0 & 0 & 0 & 0 & 0 \\ f_{31} & f_{32} & 0 & f_{34} & f_{35} & 0 & f_{37} & 0 & f_{39} \\ f_{41} & f_{42} & f_{43} & f_{44} & f_{45} & 0 & f_{47} & 0 & f_{49} \\ 0 & f_{52} & f_{53} & f_{54} & 0 & 0 & f_{57} & f_{58} & 0 \\ 0 & 0 & 0 & 0 & 0 & f_{66} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & f_{78} & f_{79} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{B.1}$$

where

$$\begin{aligned}
f_{21} &= \frac{-V_{i,\theta}}{r_i^2} \\
f_{24} &= \frac{1}{r_i} \\
f_{31} &= \frac{-V_{i,\theta}^2}{r_i^2} \\
f_{32} &= a_{m,i}\sin(\eta_i + \gamma_i - \theta_i) - a_T\cos(\alpha_T - \theta_i) \\
f_{34} &= 2\frac{V_{i,\theta}}{r_i} \\
f_{35} &= -a_{m,i}\sin(\eta_i + \gamma_i - \theta_i) \\
f_{37} &= -a_T\cos(\alpha_T - \theta_i) \\
f_{39} &= \sin(\alpha_T - \theta_i)
\end{aligned}$$

$$f_{41} = \frac{V_{i,r}V_{i,\theta}}{r_i^2}$$

$$f_{42} = a_{m,i}\cos(\eta_i + \gamma_i - \theta_i) + a_T\sin(\alpha_T - \theta_i)$$

$$f_{43} = -\frac{V_{i,\theta}}{r_i}$$

$$f_{44} = -\frac{V_{i,r}}{r_i}$$

$$f_{45} = -a_{m,i}\cos(\eta_i + \gamma_i - \theta_i)$$

$$f_{47} = -a_T\sin(\alpha_T - \theta_i)$$

$$f_{49} = \cos(\alpha_T - \theta_i)$$

$$f_{521} = [V_T\cos(\alpha_T - \theta_i) - V_{i,r}]^2 + [V_T\sin(\alpha_T - \theta_i) - V_{i,\theta}]^2$$

$$f_{522} = -\frac{a_{m,i}\sin(\eta_i)}{f_{521}^{3/2}}$$

$$\begin{aligned} f_{52} &= f_{522}\{[V_T\cos(\alpha_T - \theta_i) - V_{i,r}]V_T\sin(\alpha_T - \theta_i) \\ &\quad - [V_T\sin(\alpha_T - \theta_i) - V_{i,\theta}]V_T\cos(\alpha_T - \theta_i)\} \end{aligned}$$

$$f_{53} = -f_{522}[V_T\cos(\alpha_T - \theta_i) - V_{i,r}]$$

$$f_{54} = f_{522}[V_T\sin(\alpha_T - \theta_i) - V_{i,\theta}]$$

$$f_{57} = -f_{52}$$

$$\begin{aligned} f_{58} &= f_{522}\{[V_T\cos(\alpha_T - \theta_i) - V_{i,r}]\cos(\alpha_T - \theta_i) \\ &\quad + [V_T\sin(\alpha_T - \theta_i) - V_{i,\theta}]\sin(\alpha_T - \theta_i)\} \end{aligned}$$

$$f_{66} = cos(\eta_i)$$

$$f_{77} = -\frac{a_T}{V_T^2}$$

$$f_{78} = \frac{1}{V_T}$$

APPENDIX C

Appendix C

C.1 Proof of Theorem 4.3.1

Proof. We initiate our analysis with the assumption that the agents in the network have reached a consensus on time-to-go, correlated to a protocol-determined value. As demonstrated in our previous work, this event is expected to occur at $t = N\Delta T_c$ [30]. Using results from [47], we can further state that for $\mathcal{G}$ having a maximum path length of $(p - 1)$, the event occurs at $t = p\Delta T_c$ Following this, for the interval $t \in [k\Delta T_c, (k+1)\Delta T_c]$, the control $u_i^*(t, T_f, \boldsymbol{x})$ is ascertained via the resolution of Eq.(4.15a)-(4.15f). The obtained solution maintains its optimality throughout the horizon of the controller ΔT_{hor}. Given our decision to set $\Delta T_c = \Delta T_{hor}$, the solution retains its optimality for the duration of this interval.

We then proceed to apply Bellman's optimality principle [48]. According to this principle, an optimal policy possesses the characteristic that, irrespective of the initial state and decision, the subsequent decisions must comprise an optimal policy concerning the state resulting from the first decision.

Assuming $u_i^*(t, T_f, \boldsymbol{x})$ is optimal for each ΔT_c interval implies that, even subsequent to the initial decision at $t = k\Delta T_c$, the residual control policy $u_i^*(t, T_f, \boldsymbol{x})$ for $t \in [(k+1)\Delta T_c, (k+2)\Delta T_c]$ remains optimal. By recurrent application of Bellman's optimality principle, the control $u_i^*(t, T_f, \boldsymbol{x})$ continues to be optimal for every ensuing ΔT_c interval. Hence, we infer that if $u_i^*(t, T_f, \boldsymbol{x})$ is optimal for each discrete ΔT_c interval, it will be optimal for the entire operational framework until same time position consensus is achieved.

This analysis concludes the proof under the defined assumptions and conditions.

□

C.2 Proof of Theorem 4.3.3

Proof. The proof is well documented in literature. The equivalent Nonlinear Programming(NLP) problem is formulated as

$$\begin{aligned}
\min_{\mathbf{X},\mathbf{U}} \quad & J = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{m_u} u_{ij}^2 \\
\text{s.t.} \quad & \mathbf{X}_{k+1} = \mathbf{X}_k + T \cdot f(\mathbf{X}_k, \mathbf{U}_k), \quad \forall k \in \{1, \dots, N\} \\
& \mathbf{X}_1 = \mathbf{x}_0 \\
& \mathbf{X}_{N+1} = \mathbf{x}_{des}
\end{aligned}$$

Here, $\mathbf{X} \in \mathbb{R}^{m_s \times (N+1)}$ represents the states and $\mathbf{U} \in \mathbb{R}^{m_u \times N}$ the control inputs for a system with m_s states and m_u control inputs. The function f denotes the dynamics of the Dubin's car. The indices i and j iterate over the control inputs and time steps, respectively. The states at the first and last time steps must comply with $\mathbf{X}_1 = \mathbf{x}_0$ and $\mathbf{X}_{N+1} = \mathbf{x}_{des}$, where $\mathbf{x}_0$ is the initial state and $\mathbf{x}_{des}$ is the desired final state. The equivalence of this NLP with the original optimal control problem can be examined through the Pontryagin's Maximum Principle [49].

□

References

[1] W. Ren, "Collective motion from consensus with cartesian coordinate coupling - part i: Single-integrator kinematics," in 2008 47th IEEE Conference on Decision and Control. IEEE, 2008, pp. 1006–1011. [Online]. Available: https://doi.org/10.1109/cdc.2008.4738708

[2] J. Lindsay and S. N. Givigi, "A game theoretical representation for the rendezvous problem," in 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2012, pp. 2738–2743. [Online]. Available: http://dx.doi.org/10.1109/ICSMC.2012.6378162

[3] N. Sorensen and W. Ren, "Rendezvous problem in multi-vehicle systems: Information relay and local information based strategies," in 2006 IEEE Mountain Workshop on Adaptive and Learning Systems. IEEE, Jul. 2006, pp. 183–188. [Online]. Available: https://doi.org/10.1109/smcals.2006.250713

[4] W. Yu, G. Chen, M. Cao, and J. Kurths, "Second-order consensus for multiagent systems with directed topologies and nonlinear dynamics," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 40, no. 3, p. 881–891, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1109/TSMCB.2009.2031624

[5] J. Qin, W. X. Zheng, and H. Gao, "Coordination of multiple agents with double-integrator dynamics under generalized interaction topologies," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 42, no. 1, p. 44–57, Feb. 2012. [Online]. Available: http://dx.doi.org/10.1109/TSMCB.2011.2164523

[6] D. Liu and G.-H. Yang, “A dynamic event-triggered control approach to leader-following consensus for linear multiagent systems,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, no. 10, pp. 6271–6279, Oct. 2021.

[7] D. Zhang, Y.-P. Shen, S.-Q. Zhou, X.-W. Dong, and L. Yu, “Distributed secure platoon control of connected vehicles subject to DoS attack: Theory and application,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, no. 11, pp. 7269–7278, Nov. 2021.

[8] Z. Shiyu and Z. Rui, “Cooperative guidance for multimissile salvo attack,” Chinese Journal of Aeronautics, vol. 21, no. 6, pp. 533–539, Dec. 2008. [Online]. Available: https://doi.org/10.1016/s1000-9361(08)60171-5

[9] B. Tian, Z. Zuo, and H. Wang, “Leader–follower fixed-time consensus of multi-agent systems with high-order integrator dynamics,” International Journal of Control, vol. 90, no. 7, pp. 1420–1427, Jul. 2016. [Online]. Available: https://doi.org/10.1080/00207179.2016.1207101

[10] I.-S. Jeon, J.-I. Lee, and M.-J. Tahk, “Homing guidance law for cooperative attack of multiple missiles,” Journal of Guidance, Control, and Dynamics, vol. 33, no. 1, pp. 275–280, Jan. 2010. [Online]. Available: https://doi.org/10.2514/1.40136

[11] J. Zeng, L. Dou, and B. Xin, “Multi-objective cooperative salvo attack against group target,” Journal of Systems Science and Complexity, vol. 31, no. 1, pp. 244–261, Feb. 2018. [Online]. Available: https://doi.org/10.1007/s11424-018-7437-9

[12] S. Kang, J. Wang, G. Li, J. Shan, and I. R. Petersen, “Optimal cooperative guidance law for salvo attack: An MPC-based consensus perspective,” IEEE

Transactions on Aerospace and Electronic Systems, vol. 54, no. 5, pp. 2397–2410, Oct. 2018. [Online]. Available: https://doi.org/10.1109/taes.2018.2816880

[13] A. Sinha and S. R. Kumar, "Supertwisting control-based cooperative salvo guidance using leader–follower approach," IEEE Transactions on Aerospace and Electronic Systems, vol. 56, no. 5, pp. 3556–3565, Oct. 2020. [Online]. Available: https://doi.org/10.1109/taes.2020.2974044

[14] C. Sun, G. Hu, L. Xie, and M. Egerstedt, "Robust finite-time connectivity preserving coordination of second-order multi-agent systems," Automatica, vol. 89, pp. 21–27, Mar. 2018. [Online]. Available: https://doi.org/10.1016/j.automatica.2017.11.020

[15] W. Hu, L. Liu, and G. Feng, "Consensus of linear multi-agent systems by distributed event-triggered strategy," IEEE Transactions on Cybernetics, vol. 46, no. 1, p. 148–157, Jan. 2016. [Online]. Available: http://dx.doi.org/10.1109/TCYB.2015.2398892

[16] Y. Dong and S. Xu, "Rendezvous with connectivity preservation problem of linear multiagent systems via parallel event-triggered control strategies," IEEE Transactions on Cybernetics, vol. 52, no. 5, pp. 2725–2734, 2022.

[17] Z. Li, J. Yan, W. Yu, and J. Qiu, "Event-triggered control for a class of nonlinear multiagent systems with directed graph," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, no. 11, p. 6986–6993, Nov. 2021. [Online]. Available: http://dx.doi.org/10.1109/TSMC.2019.2962827

[18] H. Du, G. Wen, G. Chen, J. Cao, and F. E. Alsaadi, "A distributed finite-time consensus algorithm for higher-order leaderless and leader-following multi-agent systems," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 47, no. 7, pp. 1625–1634, Jul. 2017.

[19] B. Zadka, T. Tripathy, R. Tsalik, and T. Shima, "Consensus-based cooperative geometrical rules for simultaneous target interception," Journal of Guidance, Control, and Dynamics, vol. 43, no. 12, pp. 2425–2432, Dec. 2020. [Online]. Available: https://doi.org/10.2514/1.g005065

[20] T. Yucelen, Z. Kan, and E. Pasiliao, "Finite-time cooperative engagement," IEEE Transactions on Automatic Control, vol. 64, no. 8, pp. 3521–3526, Aug. 2019. [Online]. Available: https://doi.org/10.1109/tac.2018.2881132

[21] C. K. Peterson and D. A. Paley, "Distributed estimation for motion coordination in an unknown spatially varying flowfield," Journal of Guidance, Control, and Dynamics, vol. 36, no. 3, pp. 894–898, May 2013. [Online]. Available: https://doi.org/10.2514/1.59453

[22] J. Thienel, J. V. Eepoel, and R. Sanner, "Accurate state estimation and tracking of a non-cooperative target vehicle," in AIAA Guidance, Navigation, and Control Conference and Exhibit. American Institute of Aeronautics and Astronautics, Jun. 2006. [Online]. Available: https://doi.org/10.2514/6.2006-6802

[23] Q. Feng, Y. Liu, Z. H. Zhu, Y. Hu, Q. Pan, and Y. Lyu, "Vision-based relative state estimation for a non-cooperative target," in 2018 AIAA Guidance, Navigation, and Control Conference. American Institute of Aeronautics and Astronautics, Jan. 2018. [Online]. Available: https://doi.org/10.2514/6.2018-2101

[24] Y.-Z. Luo, Y.-J. Lei, and G.-J. Tang, "Optimal multi-objective nonlinear impulsive rendezvous," Journal of guidance, control, and dynamics, vol. 30, no. 4, pp. 994–1002, 2007. [Online]. Available: https://doi.org/10.2514/1.27910

[25] J. R. Cooper, "Optimal multi-agent search and rescue using potential field theory," in AIAA Scitech 2020 Forum. American Institute of

Aeronautics and Astronautics, Jan. 2020. [Online]. Available: https://doi.org/10.2514/6.2020-0879

[26] B. Taner and K. Subbarao, "Model predictive control for cooperative systems with task prioritization applied to vehicle rendezvous and docking," in AIAA SCITECH 2023 Forum. American Institute of Aeronautics and Astronautics, Jan. 2023. [Online]. Available: https://doi.org/10.2514/6.2023-0486

[27] L. Persson, T. Muskardin, and B. Wahlberg, "Cooperative rendezvous of ground vehicle and aerial vehicle using model predictive control," in 2017 IEEE 56th Annual Conference on Decision and Control (CDC), 2017, pp. 2819–2824.

[28] J. Wang, S. Yang, Q. Wang, and L. Ji, "Finite-time consensus of nonlinear delayed multi-agent system via state-constraint impulsive control under switching topologies," Nonlinear Dynamics, May 2023. [Online]. Available: https://doi.org/10.1007/s11071-023-08493-9

[29] R. S. Voleti, D. Bhattacharjee, and K. Subbarao, "Finite same-time consensus guidance laws for unmanned aerial systems," in AIAA SCITECH 2022 Forum. American Institute of Aeronautics and Astronautics, Jan. 2022, p. 1845. [Online]. Available: https://doi.org/10.2514/6.2022-1845

[30] ——, "Leaderless time-to-go protocols for same-time position consensus," IEEE Transactions on Systems, Man, and Cybernetics: Systems, pp. 1–15, 2023.

[31] S. R. Kumar and D. Mukherjee, "Finite-time impact time guidance using deviated pursuit against maneuvering targets," in 2020 American Control Conference (ACC). IEEE, Jul. 2020, pp. 3776–3781. [Online]. Available: https://doi.org/10.23919/acc45564.2020.9147789

[32] X. Yu and M. Zhihong, "Fast terminal sliding-mode control design for nonlinear dynamical systems," IEEE Transactions on Circuits and Systems

I: Fundamental Theory and Applications, vol. 49, no. 2, pp. 261–264, 2002. [Online]. Available: https://doi.org/10.1109/81.983876

[33] D. B. West, Introduction to Graph Theory, 2nd ed. Prentice Hall, 2001.

[34] A. Marianne, R. Bapat, and S. Gaubert, “Max-plus algebra,” Handbook of linear algebra 39, 2006.

[35] A. E. Spalding, Min-plus algebra and graph domination. University of Colorado at Denver, 1998.

[36] H. K. Khalil, Nonlinear systems third edition. Patience Hall, 1996.

[37] M. Van Steen, “Graph theory and complex networks,” An introduction, vol. 144, pp. 1–287, 2010.

[38] J. L. Crassidis and J. L. Junkins, Optimal estimation of dynamic systems. Chapman and Hall/CRC, 2004.

[39] J. L. Crassidis, “Introduction to the special issue on the kalman filter and its aerospace applications,” Journal of Guidance, Control, and Dynamics, vol. 40, no. 9, pp. 2137–2137, 2017.

[40] S. J. Julier and J. K. Uhlmann, “Unscented filtering and nonlinear estimation,” Proc. IEEE Inst. Electr. Electron. Eng., vol. 92, no. 3, pp. 401–422, Mar. 2004.

[41] J. L. Crassidis and F. L. Markley, “Unscented filtering for spacecraft attitude estimation,” Journal of guidance, control, and dynamics, vol. 26, no. 4, pp. 536–542, 2003.

[42] F. L. Lewis, D. Vrabie, and V. L. Syrmos, Optimal control. John Wiley & Sons, 2012. [Online]. Available: https://www.wiley.com/en-ae/Optimal+Control,+3rd+Edition-p-9781118122723

[43] E. Hairer, S. P. Nørsett, and G. Wanner, Solving Ordinary Differential Equations I: Nonstiff Problems. Springer-Verlag, 1993, vol. 1.

[44] C. R. Hargraves and S. W. Paris, “Direct trajectory optimization using nonlinear programming and collocation,” Journal of Guidance, Control, and Dynamics, vol. 10, no. 4, pp. 338–342, 1987.

[45] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 2520–2525.

[46] T.-L. Liu and K. Subbarao, “Optimal aggressive constrained trajectory synthesis and control for multi-copters,” Aerospace, vol. 9, no. 6, p. 281, May 2022. [Online]. Available: http://dx.doi.org/10.3390/aerospace9060281

[47] B. M. Nejad, S. A. Attia, and J. Raisch, “Max-consensus in a max-plus algebraic setting: The case of fixed communication topologies,” in 2009 XXII International Symposium on Information, Communication and Automation Technologies, 2009, pp. 1–7.

[48] R. Bellman, Dynamic Programming. American Association for the Advancement of Science (AAAS), Jul. 1966, vol. 153. [Online]. Available: http://dx.doi.org/10.1126/science.153.3731.34

[49] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, The Mathematical Theory of Optimal Processes. New York: Interscience, 1962.

Biographical Statement

Rajeev Shobhit Voleti was born in 1995 in Jangareddygudem, a town situated in the Eluru district of Andhra Pradesh, India. He embarked on his academic journey in aerospace engineering at the Indian Institute of Technology Bombay, where he earned his Bachelor of Technology degree in 2017, with a minor in Systems and Control Engineering.

In Spring 2019, Shobhit joined the University of Texas at Arlington (UTA) as a Ph.D. student in Aerospace Engineering. During his doctoral studies at UTA, he engaged in research at the Aerospace Systems Laboratory, contributing as a Graduate Research Assistant. His research interests are broad and interdisciplinary, encompassing nonlinear system analysis, nonlinear and optimal control designs, nonlinear estimation, and the distributed cooperative control of networked multi-agent systems.

Shobhit's academic and research pursuits have been driven by a passion for advancing the field of aerospace engineering, particularly in the areas of control systems and their applications in complex, networked environments.