

東南大學

畢業設計(論文)報告

題目:

17212

1 0 1 06

东南大学毕业（设计）论文独创性声明

东南大学毕业（设计）论文使用授权声明

摘 要

编队控制在区域监控搜索、军事对抗、物流运输等领域中有着广泛的应用，是当前研究的热点。无人车集群编队控制问题通常包含两个子问题。第一是编队形状控制，即控制无人车形成预期的几何构型。第二是编队行进控制，即实现无人车编队的持续性的行进仿射变换。在现实复杂环境下，无人车之间的协同面临外界噪音干扰的影响，此外，受外界环境中的各种障碍物的影响，编队要求能够自适应变换队形。本论文聚焦于复杂环境下分布式编队控制算法的设计与性能分析问题。

首先，对多智能体分布控制算法的研究现状进行总结介绍；其次，对近期提出的基于应力矩阵的固定领航者编队控制算法和移动领航者编队控制算法进行仿真复现，并使用这两种算法对编队形状控制和编队行进控制这两个子问题进行仿真分析，检验算法的效果；在复现的基础上，本文进一步通过仿真分析了分布式编队控制算法中的增益大小对于算法性能的影响，同时，分析了通信次数对算法性能的影响，为增益大小的选择和通信次数的优化提供一定的经验指导。

关键词：无向图，分布式算法，无人车控制算法，多智能体系统，仿射变换

ABSTRACT

Formation control is widely used in the fields of regional monitoring and search, military confrontation, logistics and transportation. It is a hot topic of current research. The formation control problem of unmanned vehicles usually consists of two sub-problems. The first is formation shape control, which is to control the unmanned vehicle to form the expected geometric configuration. The second is formation control, which is to realize continuous affine transformation of the formation. In the real complex environment, the synergy between unmanned vehicles faces the influence of external noise interference. In addition, due to the influence of various obstacles in the external environment, the formation requires adaptive formation change. This thesis focuses on the design and performance analysis of distributed formation control algorithms in complex environments.

Firstly, the research status of multi-agent distributed control algorithm is summarized and introduced. Secondly, the recently proposed fixed leader formation control algorithm and mobile leader formation control algorithm based on stress matrix are simulated and reimplemented, and these two algorithms are used to simulate the two sub-problems of formation shape control and formation travel control, to check the effectiveness of the algorithm. On the basis of the simulation, the influence of gain size on the performance of distributed formation control algorithm and the influence of communication times on the performance of algorithm are further analyzed, which provides some empirical guidance for the selection of gain size and the optimization of communication times.

KEY WORDS: Deep learning, Distributed optimization, Communication networks, Directed graphs, Accelerated algorithms

目 录

摘 要	I
ABSTRACT	II
目 录	III
第一章 绪论	1
1.1. 研究背景与意义	1
1.2. 研究现状	2
1.3. 本文研究内容	3
1.4. 本文组织结构	4
第二章 问题陈述	5
2.1. 编队控制问题的数学描述	5
2.2. 仿射变换和仿射独立	6
2.3. 应力矩阵和刚性	7
2.4. 编队仿射变换的问题陈述	9
2.4.1 时变目标编队	9
2.4.2 标准构型的仿射图像	10
2.5. 本章小结	10
第三章 行进编队的仿射变换控制算法	11
3.1. 领航者固定	11
3.1.1 控制算法	11
3.1.2 算法实现	11
3.2. 领航者移动	14
3.2.1 控制算法	14
3.2.2 算法实现	15
3.3. 本章小结	19
第四章 分布式编队控制算法性能分析及优化	20
4.1. 实验一 (编队形成的领航者固定控制算法)	20
4.1.1 抗噪声性能分析	20

4.1.2 通信优化	21
4.2. 仿真实验二 (编队形成的领航者移动控制算法)	21
4.2.1 参数对于收敛时间影响	21
4.2.2 抗噪声性能分析	22
4.2.3 通信优化	23
4.3. 仿真实验三 (编队的行进和仿射变换)	24
4.4. 本章小结	26
第五章 总结与展望	27
5.1. 总结	27
5.2. 展望	27
参考文献	29
致 谢	31

第一章 绪论

1.1 研究背景与意义

随着物联网的发展，人们逐渐将联网技术运用于车辆上，为无人车的编队发展奠定了基础，车辆编队的相关技术也逐渐开始发展。在军用方面使用无人车战车编队，在民用方面的物流产业使用无人车编队进行物流运输等。故而研究无人车编队控制对于国家的国防和人们平常的生活都有着重要的意义。车辆编队的控制问题是研究实现无人车编队的重要研究内容，无人车作为智能体的一种，可以使用多智能体的控制策略来进行无人车的编队控制研究。

多智能体系统^[1]是指在一个环境中交互的多个智能体组成的计算系统，其作为分布式人工智能的分支，在过去 15 年内一直是一个主要的研究领域。

相较于单智能体系统，多智能体系统拥有很多优势：

1. 多智能体系统具有更好的时间和空间分布性。多智能体系统在以某个固定编队执行任务时，通过合理协调和分配能够减少系统中因为个体机器人的自主运动而造成相互干扰或者妨碍而导致的时间或者空间上的浪费。
2. 多智能体中的个体获得的信息更加丰富和准确。由于各个机器人之间可以通过通信实现信息传输，以此来实现多个传感器信息的有效互补，使得信息内容更加准确。
3. 多智能体系统对个体机器人的功能要求相对较低。当单个机器人因功能有限或者能力不足无法完成任务时，可以通过其他机器人的协调处理将复杂任务分解。
4. 多智能体系统具有更好的鲁棒性。由于具有较大的冗余性，当某个个体机器人出现故障时，其他机器人可以快速新城新的编队。
5. 多智能体机器人可以完成较为复杂的任务。

而在多智能体编队控制中分为集中式控制和分布式控制两大主要策略。集中式控制：编队中所有个体要将自己的状态信息和运动目标等信息与编队中所有个体进行交互。分布式控制：编队中给所有个体仅与自己相邻的个体进行状态信息的共享与交互。相比之下，集中式控制策略的收敛速度更快，个体获得的信息量更大，但是分布式控制策略的结构较为简单且可靠，需要交互的信息量较少。同时分布式控制策略适应性强，具有较好的扩充与容错能力。

所以本文以无人车作为个体机器人研究对象，对多无人车系统分布式协同编队控制算法进行研究，重点关注队形保持和轨迹跟踪两个关键子问题。

1.2 研究现状

编队控制是指多个智能体组成的编队在向特定方向或者目标运动过程中，不仅要适应环境约束，并且整个智能体系统还要维持稳定的几何的控制问题。多智能体编队控制问题大方向上可以分为两个子问题。一个是编队的形状控制，另一个是编队的行进控制。具体可以分为以下四个问题：

1. 队形生成. 主要研究多智能体之间如何从初始构型形成预期的几何构型。其中不同的队形起到不同的作用。例如 2011 年英国机械工程师协会的报告指出，跨洋飞行如果采用 V 形编队最高可降低 12% 的能源。编队控制领域更多的研究队形生成问题。例如 2015 年英国学者 Lau[2] 等人针对多无人机任意队形设计问题提出基于张拉整体结构队形建模得到了多种稳定队形。2017 年，刘春 [3] 等利用势场函数来设计轮式机器人的控制量输出，生成了多机器人持久队形，但是其研究忽略了运动过程中存在的干扰问题。
2. 队形保持主要研究多智能体编队的稳定问题。目前学者的研究工作主要致力于对领航者，基于行为法和虚拟结构法等方法的融合上以提高系统的稳定性。例如 2005 年，Tanner[4] 等通过改进人工势场法构造了多机器人系统的导航函数实现了队形爆出与避障。
3. 队形切换主要研究编队如何从当前队形切换成另外一个不同的队形，其中包含被动变换和主动变换两种方法。被动变换指由于环境的约束（例如避障）使得队形不得不改变，而主动变换时按照任务还要求进行队形变换。
4. 自适应主要研究 MAS 在未知情况下，如何自动改变几何队形或维持几何队形来适应环境。其研究内容通常用于解决动力非线性、异构智能体、通信延迟干扰、不确定等问题。例如：2010 年，Duran 等 [5] 收到文献 [6] 的启发提出一种自适应模糊控制方法，用于解决具有非线性和不确定动力学的多智能体系统种的编队控制和目标跟踪稳定性的问题。

在过去的 20 年里，无人车编队问题已经被人们使用很多方法研究过。早年间基于行为的编队控制方法已经可以完成较为复杂的编队问题(具体见，[7][8][9][10])。但是这个方法系统的收敛性无法很好的用数学证明。从实际应用的角度来看，系统的收敛性是至关重要的，因为这保证了系统的行为会按照预期进行。

由于一致性理论在编队控制中成功的应用 [11][12]，许多学者致力于研究发展收敛保障的编队控制(近期研究见 [13][14])。受到这种启发，学者们提出一些方法以期望能实现编队的期望行进。已有的主流编队控制方法可以根据目标编队的定义方法分为三类，分别为基于位移，基于距离和基于方向。这三类方法都是使用常量限制无人车之间的位移，距离和方向。目标编队的常量约束的不变性对编队的机动性有至关重要的影响。因此，基于位

移的编队控制方法可以应用于时变平移 [11],[12]。但是缩放和方向很难控制，因为改变缩放和方向需要改变约束条件。作为比较，基于距离的控制方法可以应用于跟踪具有时变平移和方向的目标 [13],[14]，但是很难跟踪时变的编队的缩放。基于方向的控制方法可以跟踪时变的平移和缩放，但是很难跟踪时变的方向。

受到这三种方法限制的启发，学者们提出一些改进的方法以期望能完成预期的编队行进目标。例如 [15] 中对于基于位移的方法进行改进，加入了编队缩放的估计机制。在 [16] 中改进了基于距离的编队控制算法允许最终的编队可以有未指定的缩放比例。但是，这些改进通常产生了非常复杂的控制和估计问题，并且可能每个智能体都需要额外的传感器和通讯工具。近期在 [17] 中提出了一种可以跟踪时变编队的方法。但是这种方法中每一个无人车的期望的编队行进必须事先定义。

近期，学者们提出了基于局部方向 [16]，重心坐标系 [17]，复杂 Laplacians [18][19]，和应力矩阵 [20] 等方法。由于新约束的增强不变性，这些方法很有吸引力。例如，一个复杂的拉普拉斯算子对于一个队形的平移、旋转和缩放变化是不变的。因此，基于复杂拉普拉斯算子的方法可以同时实现平移、旋转和缩放编队机动。然而，这种方法仅适用于二维的编队控制。

在这些新方法中，基于应力矩阵的方法有望实现一般的编队机动。应力矩阵可以被看成是一个广义上的 Laplacian 图。它的结构是由底层图决定的，但是这些元素的值是由队形共同决定的。应力矩阵已经被用于静态目标编队的稳定问题 [20]，但是其关于解决行进编队问题的潜力还远远没有被发掘。事实上，应力矩阵对于任何仿射变换都是不变的。仿射变换是一种一般的线性变换，它可能对应于它们的平移、旋转、缩放、剪切或合成。因此，应力矩阵为实现各种编队机动行为提供了强大的工具。

在这篇文章中我们基于应力矩阵使用领航者-跟随者的策略来解决编队行进控制问题。

1.3 本文研究内容

本文对无人车的分布式编队控制算法展开研究。首先，对于已有的编队控制原理和控制方法进行总结。控制原理包括分布式、集中式和分散式，主流控制方法有领航者跟随法、基于行为法、虚拟结构法和基于图论法。其次，本文将针对领航者跟随法实现编队形成、编队保持、编队切换的算法仿真，其中控制方法主要参考论文 [21]。通过复现该论文中的控制算法，实现无人车编队从随机初始状态完成预期编队的生成，并使队形完成预设的仿射变换行进路线。最后，对于算法的性能进行比较并对算法的通信次数进行优化。主要贡献如下：

(1) 从分布式算法切入，对已有的分布式编队控制方法算法进行文献调研，并分析不同控

制方法的优势与可行性。

- (2) 结合论文 [21]，实现了其中提出的领航者控制方法，完成了编队形成、编队保持和编队仿射变换的仿真。
- (3) 讨论了算法中参数以及噪声对于算法性能的影响。
- (4) 对于算法中的通信次数进行优化。

1.4 本文组织结构

本文将由以下五部分组成：

- (1) 绪论: 介绍了分布式编队控制算法的研究背景，即为什么要研究分布式编队控制算法。又对分布式编队控制算法的背景知识做了简单的介绍，从时间空间分布性、鲁棒性和任务复杂度三个角度解释了分布式编队控制算法的意义。最后又对分布式编队控制的研究现状进行了一个总结。
- (2) 问题陈述: 本章节将对控制编队仿射变换的问题及相关概念进行详细的定义和陈述。
- (3) 分布式编队控制算法: 给出分布式控制算法的控制律，并通过仿真复现其算法以验证算法的有效性。
- (4) 算法性能分析: 探讨算法中参数对于算法的影响，探讨噪声对于算法的影响，对无人车之间的通信次数进行优化。
- (5) 总结与展望: 对毕设进行的工作进行总结与未来的展望。

第二章 问题陈述

本章首先使用无向图对无人车编队进行数学描述，在此基础上介绍了编队仿射变换的相关概念和数学形式。在确定了编队仿射变换的数学描述之后，为了控制编队的构型（具体定义见 2.1 节）本文引入应力矩阵和刚性的概念。最后对无人车编队控制问题进行数学描述，将其转化为数学收敛问题。

2.1 编队控制问题的数学描述

考虑一组在 \mathbb{R}^d 上数量为 n 的无人车，由于无人车只能在 2 维平面上移动，所以其中 $d = 2$ 并且 $n \geq d + 1$ 。在 \mathbb{R}^2 平面上，我们使用无向图 \mathcal{G} 来定义无人车编队控制问题。将每个无人车看作是一个节点，令 $p_i \in \mathbb{R}^d$ 为无人车 i 的位置，由全局参考系中每个无人车 x 轴和 y 轴的坐标表示。我们称无人车编队由特定的相对位置形成的形状称为构型，最终预期完成的编队形状为期望构型。 $p = [p_1^T, \dots, p_n^T \in \mathbb{R}^{dn}]$ 为整个编队的构型（具体表示可以参考图 2.1）。值得注意的是这是一个 $1 \times n$ 的向量。

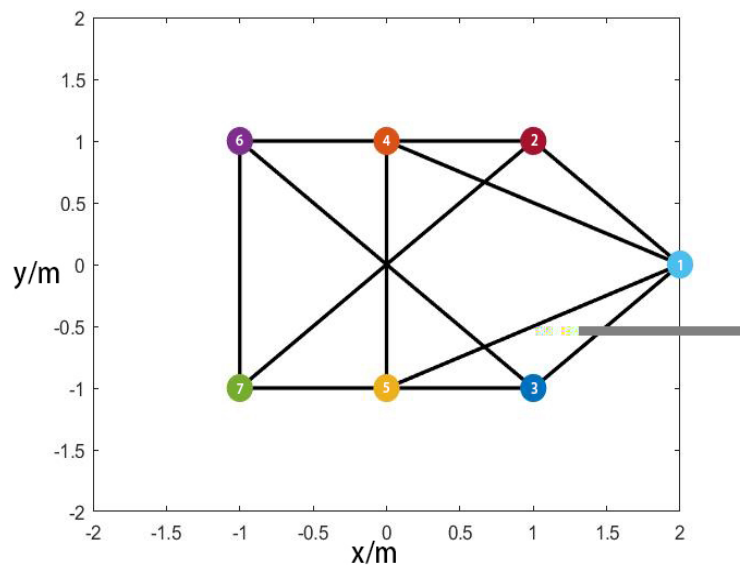


图 2.1: 编队的期望构型

该构型的矩阵表达形式为 $p = [2, 0, 1, 1, 1, 1, -1, 0, 1, 0, -1, -1, 1, -1, -1]$

另外，在无向图 \mathcal{G} 中记边的数量为 m 。如果无人车 p_i 和 p_j 之间存在边 (i, j) ，则表示无人车 i 与无人车 j 之间存在通信，且两个节点互为相邻节点。由于本文中只考虑无向图，即 $(i, j) \in \varepsilon \Leftrightarrow (j, i) \in \varepsilon$ ，所以边 (i, j) 所连接的两个节点之间的通信是双向的。下表给出了本文常用的符号及其含义：

符号	含义	备注
n	无人车数量	节点数量
m	边的数量	
p_i	无人车 i 的位置	
$p = [p_1^T, \dots, p_n^T \in \mathbb{R}_{dn}]$	整个编队的构型	
$\mathcal{G} = (\mathcal{V}, \varepsilon)$	无人车之间的互相关系	
$\mathcal{V} = 1, \dots, n$	顶点集	
$\varepsilon \subseteq \mathcal{V} \times \mathcal{V}$	边集	
$(i, j) \in \varepsilon$	边	$(i, j) \in \varepsilon \Leftrightarrow (j, i) \in \varepsilon$

2.2 仿射变换和仿射独立

给定 \mathbb{R}^d 上一系列节点 $\{p_i\}_{i=1}^n$ ，对于这些点的仿射空间记为 S ，

$$S = \left\{ \sum_{i=1}^n a_i p_i : a_i \in \mathbb{R} \text{ 且 } \sum_{i=1}^n a_i = 1 \right\}$$

给定任意的仿射空间，我们总是可以平移它来包含原点，从而得到一个线性空间。线性变换的维度即仿射变换的维度 d ，我们称其为在 \mathbb{R}^d 上的仿射变换。

一个点集 $\{p_i\}_{i=1}^n$ 如果存在变换比例 $\{a_i\}_{i=1}^n$ 非全零以致于 $\sum_{i=1}^n a_i p_i = 0$ 且 $\sum_{i=1}^n a_i = 0$ 则称为仿射依赖，反则为仿射独立。定义构型矩阵 $P \in \mathbb{R}^{n \times d}$ 和扩展矩阵 $\bar{P} \in \mathbb{R}^{n \times (d+1)}$ 为

$$P(p) = \begin{bmatrix} p_1^T \\ \vdots \\ p_n^T \end{bmatrix},$$

$$\bar{P}(p) = \begin{bmatrix} p_1^T & 1 \\ \vdots & \\ p_n^T & 1 \end{bmatrix} = [P(p), 1_n]$$

例如根据定义，图2.1的构型矩阵写为 $P(p) = [2, 0; 1, 1; 1, -1; 0, 1; 0, -1; -1, 1; -1, -1]$ 。

根据定义 $\{p_i\}_{i=1}^n$ 为仿射依赖当且仅当 $\bar{P}(p)$ 的行为线性依赖的， $\{p_i\}_{i=1}^n$ 为仿射独立当且仅当 $\bar{P}(p)$ 的行为线性独立的。因为 $\bar{P}(p)$ 有 $d+1$ 行，所以这里最多有 $d+1$ 个点在 \mathbb{R}^d 上是仿射独立的。由此可知， $\bar{P}(p)$ 有 $d+1$ 行，且每一行为线性独立，因此， $\text{rank}(\bar{P}(p)) = d+1$ 。这个结论可以作为一个引理。

引理 2.1. $\{p_i\}_{i=1}^n$ 仿射 \mathbb{R}^d 当且仅当 $n \geq d+1$ 且 $\text{rank}(\bar{P}(p)) = d+1$

由于仿射空间 \mathbb{R}^d 需要至少 $d + 1$ 个点，所以领航者的最少数量为 $d + 1$ 个。

2.3 应力矩阵和刚性

由于编队在行进过程中整体构型是在参考系中不断移动的，所以使用坐标形成的构型矩阵 P 来控制队形是不现实的。在此，我们引入应力矩阵来对编队的队形进行控制。

对于编队 (\mathcal{G}, p) ，应力为一个标量集合， $\{\omega_{ij}\}_{(i,j) \in \varepsilon}$ 对应图 \mathcal{G} 中的每条边。 $\{\omega_{ij}\}_{(i,j)}$ 的值表示了边 (i, j) 上的力，当 $\omega_{ij} \geq 0$ 时表示吸引力，当 $\omega_{ij} \leq 0$ 时表示排斥力。可以想象，当每条边上的排斥力或吸引力达到一种平衡的时候，编队处于稳定状态，各节点之间的距离为确定值。当 $\{\omega_{ij}\}_{(i,j)}$ 满足

$$\sum_{j \in \mathcal{N}_i} \omega_{ij}(p_j - p_i) = 0, i \in \mathcal{V} \quad (2.1)$$

时我们称之为均衡应力。可以将等式 (2.1) 看作是无人车 $j \in \mathcal{N}_i$ 施加在无人车 i 的应力达到平衡。记 $\omega = [\omega_1, \dots, \omega_m] \in \mathbb{R}^m$ 为应力向量，其中 ω_k 与第 k 条无向边 ($k = 1, \dots, m$) 相对应。注意，平衡应力只能由一个标量因子决定。这意味着如果 ω 是均衡应力，则 $k\omega$ 对于任意的 $k \in \mathbb{R}_{\neq 0}$ 也是均衡应力。公式 (2.1) 的矩阵形式表示为 $(\Omega \otimes I_d)p = 0$ 。其中 $\Omega \in \mathbb{R}^{n \times n}$ 为应力矩阵 (举例说明可见图 2.2)，满足

$$[\Omega]_{ij} = \begin{cases} 0, i \neq j, (i, j) \notin \varepsilon \\ -\omega_{ij}, i \neq j, (i, j) \in \varepsilon \\ \sum_{k \in \mathcal{N}_i} \omega_{ik}, i = j \end{cases}$$

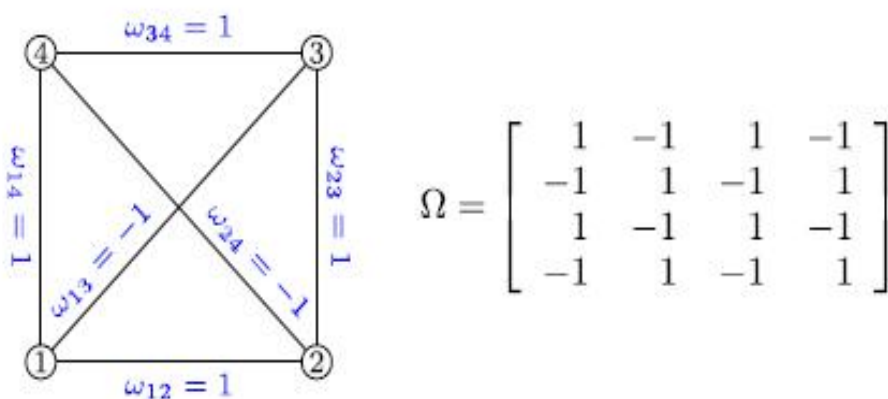


图 2.2: 应力矩阵及对应的 Ω 矩阵

一旦标准构型确定，接下来就可以计算应力矩阵。计算应力矩阵是非常复杂的，在论

文 [20] 中提到该问题可以表述为一个动态规划问题。这里，我们根据论文 [21] 给出的替代性算法来计算应力矩阵。

定义 ω 为标准构型的应力向量。考虑任意方向的无向图 G ，令 $H \in \mathbb{R}^{m \times n}$ 为其关联矩阵， $H = [h_1, \dots, h_n]$ 。定义

$$E = \begin{bmatrix} \bar{P}^T H^T \text{diag}(h_1) \\ \vdots \\ \bar{P}^T H^T \text{diag}(h_n) \end{bmatrix} \in \mathbb{R}^{n(d+1) \times m} \quad (2.2)$$

令 $z_1, \dots, z_q \in \mathbb{R}^m$ 为 $\text{Null}(E)$ 的基础。实际上，假设 $\bar{P}(r)$ 的 SVD 是 $\bar{P}(r) = U \Sigma V^T$ 。令 $U = [U_1, U_2]$ 其中 U_1 包含 U 中的前 $d+1$ 行。定义 $M_i = U_2^T H^T \text{diag}(z_i) H U_2, i = 1, \dots, q$ 。则均衡应力矩阵可以通过如下方法计算。

标准构型的均衡应力为

$$\omega = \sum_{i=1}^q c_i z_i$$

其中 c_1, \dots, c_q 满足线性矩阵不等式 (LMI)

$$\sum_{i=1}^q c_i M_i > 0 \quad (2.3)$$

可以使用 Matlab 中 LMI 工具包来解决这个线性矩阵不等式问题。

应力矩阵的性质与编队的刚性有着至关重要的联系。刚性是用于刻画基于保持成对节点间距离的无向编队控制结构，使得无向图中的边不仅描绘了信息的流向而且表明了节点间的距离限制。所谓刚性编队，就是在连续运动（平移和旋转）下保证相邻节点之间的距离保持不变。例如图 2.2 为刚性编队，图 2.3 为非刚性编队。非刚性编队虽然约束了节点之间的距离，但是同样的约束会产生不同的编队形状，这对于编队控制算法是不利的。如果编队是刚性的，通过约束节点上边的应力可以得到唯一确定的编队构型。如下图 2.3 所示，同样的距离约束会产生两种不同的编队构型。

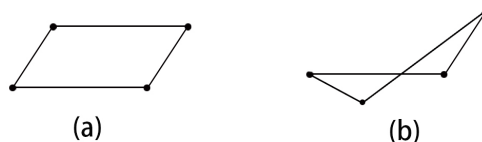


图 2.3: 非刚性编队

为了能够通过应力矩阵来控制编队，我们必须保证应力矩阵与编队构型之间有一一对

应的关系。由 [22][23][24] 可以得到引理 2。

引理 2.2. 给定一个无向图 \mathcal{G} 和一个通用构型 p ，编队 (\mathcal{G}, p) 时全局刚性的当且仅当存在一个应力矩阵 Ω 是半正定的并且 $\text{rank}(\Omega) = n - d - 1$

2.4 编队仿射变换的问题陈述

该章节定义了时变目标编队问题。

2.4.1 时变目标编队

仿射编队机动控制的目标是引导一组智能体跟踪时变的目标编队。

定义 2.1. 时变目标编队的构型用以下的形式表示：

$$p^*(t) = [I_n \otimes A(t)]r + 1_n \otimes b(t)$$

其中 $r = [r_1^T, \dots, r_n^T]^T = [r_l^T, r_f^T]^T \in \mathbb{R}^{dn}$ 是一个不变的常量构型，而 $A(t) \in \mathbb{R}^{d \times d}$ 和 $b(t) \in \mathbb{R}^d$ 是持续时变的。无人车 i 在目标编队中的期望位置表示为 $p_i^*(t) = A(t)r_i + b(t)$ 。

常量构型 r 表示一个编队会包含的经典的几何模型，在文中我们称 r 为标准的几何构型，称 (\mathcal{G}, r) 为标准编队。目标构型实际上是通过标准构型的时变仿射变换而来。仿射变换是一系列线性变换包括平移、旋转、放缩、剪切等的组合。仿射变换将会保留直线和平面，以及平行线。因此，如果变换前的编队是共线或者共平面的，变换后仍然将是共线或者共平面的。

定义过目标编队后，本文要解决的问题就转换为控制一组智能体去跟踪时变目标构型来达到当 $t \rightarrow \infty$ 时 $p(t) \rightarrow p^*(t)$ 。问题的关键是如何让每一个智能体知道 $A(t)$ ， $b(t)$ 和 r_i 这样每个智能体都能追踪自己对应的行进轨迹。但是这种方法的缺点在于这种方法要求对应所有时间 t 的 $A(t)$ ， $b(t)$ 都必须预先确定并存储在智能体中。这种要求很难被实现，因为这样就无法对于不确定的环境进行动态的响应。

为了在分布式下完成目标编队，我们采用领航者-跟随着策略。该策略中仅有少量智能体知道目标的行进编队，称为领航者，其他剩余智能体成为跟随着。跟随着不需要知道目标编队，仅需要跟随领航者的行动即可。在 [21] 中已经证明领航者的位置与仿射变换 (A, b) 有着一对一的确定关系，因此，整体的编队仿射变换可以通过控制领航者的位置来完成。由于领航者的数量一般而言非常少，所以并不需要特别地设计领航者的控制律，默认领航者可以被很好地控制。在实际情况下，领航者一般会通过人工控制或者智能的决策程序控制。假设领航者在所有时刻 t 都处于预期的目标编队位置，即 $p_l(t) = p_l^*(t)$ 。因此，控制问题变成了控制跟随着使得其在 $t \rightarrow \infty$ 时 $p_f(t) \rightarrow p_f^*(t)$

关于如何计算 $A(t)$, $b(t)$ 如果 $\{r_i\}_{i \in \mathcal{V}_l}$ 在 \mathbb{R}^d 上仿射变换, 对于任何 $p \in \mathcal{A}(r)$, 对应的 A 和 b 可以被唯一确定为

$$A = \left(\sum_{i \in \mathcal{V}_l} p_i \tilde{r}_i^T \right) \left(\sum_{i \in \mathcal{V}_l} \tilde{r}_i \tilde{r}_i^T \right)^{-1} \quad (2.4)$$

$$b = \frac{1}{n_l} \sum_{i \in \mathcal{V}_l} p_i - \left(\sum_{i \in \mathcal{V}_l} p_i \tilde{r}_i^T \right) \left(\sum_{i \in \mathcal{V}_l} \tilde{r}_i \tilde{r}_i^T \right)^{-1} \bar{r} \quad (2.5)$$

2.4.2 标准构型的仿射图像

标准构型的仿射图像被定义为

$$\mathcal{A}(r) = \{p \in \mathbb{R}^{dn} : p = (I_n \otimes A)r + 1_n \otimes b, A \in \mathbb{R}^{d \times d}, b \in \mathbb{R}^d\}$$

$$= \{p = [p_1^T, \dots, p_n^T]^T \in \mathbb{R}^{dn} : p_i = Ar_i + b, A \in \mathbb{R}^{d \times d}, b \in \mathbb{R}^d\}$$

仿射图像就是对于标准构型 r 的全部仿射变换几何, 对于所有 t , 时变目标构型 $p^*(t)$ 都在 $\mathcal{A}(r)$ 中。对于 $\mathcal{A}(r)$ 维度的分析如下:

定义 2.2. $\mathcal{A}(r)$ 的维度等于 $d^2 + d$ 当且仅当 $\{r_i\}_{i=1}^n$ 在仿射空间 \mathbb{R}^d 中。

具体的详细证明过程请参考 [21]。

根据以上的定理, 我们提出对于标准构型的假设如下:

假设 2.1. 对于标准编队构型 (\mathcal{G}, r) , 假设 $\{r_i\}_{i=1}^n$ 仿射空间为 \mathbb{R}^d

假设 2.2. 假设标准构型 (\mathcal{G}, r) 有半正定应力矩阵 Ω 满足 $\text{rank}(\Omega) = n - d - 1$

2.5 本章小结

本章节首先确定了编队控制问题的数学描述, 主要使用无向图对无人车和编队构型进行描述。其次介绍了仿射变换的相关概念, 其中仿射独立这个概念对于无人车编队问题有重要意义, 用于确定编队中领航者的最少数量。之后, 介绍了应力矩阵和刚性的概念。应力矩阵的作用在于控制编队的构型, 刚性的概念则是保证编队构型与应力矩阵之间存在一一对应的关系。在介绍完基本的数学描述和相关概念后, 本章最后一个小节将时变编队控制问题转化为控制跟随者使得其在 $t \rightarrow \infty$ 时 $p_f(t) \rightarrow p^*f(t)$ 。

第三章 行进编队的仿射变换控制算法

在这一章节中我们将给出单积分的仿射变换的分布式控制算法。

3.1 领航者固定

3.1.1 控制算法

首先考虑最简单的情况，即领航者固定的情况。在这种情况下，目标编队也是固定的，可以使用以下的控制律来解决仿射变换的控制问题

$$\dot{p}_i = - \sum_{j \in \mathcal{N}_i} w_{ij} (p_i - p_j), i \in \mathcal{V}_f \quad (3.1)$$

该公式的矩阵形式为

$$\dot{p}_f = -\bar{\Omega}_{ff} p_f - \bar{\Omega}_{fi} p_i^* \quad (3.2)$$

定义轨迹偏差为 $\delta_{pf}(t) = p_f(t) - p_f^*(t)$ ，则 (3.2) 可以写为 $\dot{p}_f = -\bar{\Omega}_{ff} p_f^*$ 。所以可以看作是一种 Lyapunov 函数 $V = 1/2 \delta_{pf}^T \bar{\Omega}_{ff} \delta_{pf}$ 的梯度下降算法。当没有领航者的时候将会退化成 $\dot{p} = -\bar{\Omega}_p$ 。

假设 3.1. 假设标准构型 (\mathcal{G}, r) 可以根据领航者仿射定位的。

回顾目前为止的 3 个假设，假设 2.1 要求标准构型的仿射空间为 \mathbb{R}^d 使得 $\dim(A(r)) = d^2 + d$ ；假设 2.2 要求标准队形满足一些刚性限制使得应力矩阵 Ω 是半正定的，且满足 $\text{rank}(\Omega) = n - d - 1$ ；假设 3.1 要求被选中的领航者在标准编队中仿射空间为 \mathbb{R}^d 。这三个假设有重要的数学目的，即保证了 $\bar{\Omega}$ 是正的。而 $\bar{\Omega}$ 是正的则有 $p_f^*(t) = -\bar{\Omega}_{ff}^{-1} \bar{\Omega}_{fi} p_i^*(t)$ 。则定义的追踪误差：

$$\begin{aligned} \delta_{pf}(t) &= p_f(t) - p_f^*(t) \\ &= p_f(t) + \bar{\Omega}_{ff}^{-1} \bar{\Omega}_{fi} p_i^*(t) \end{aligned} \quad (3.3)$$

因此控制目标变为控制跟随者使得 $t \rightarrow 0$ 时 $\delta_{pf}(t) \rightarrow 0$ 。

3.1.2 算法实现

该仿真实验实现了智能体任意初始化位置下对于预期编队的形成。使用的控制规则为 (3.1)，即领航者固定的控制规则。

为了实现控制算法，首先需要设计一个满足假设 1-3 的标准型编队。为了满足假设 2.1，

标准构型必须在 \mathbb{R}^d 上仿射变换。为了满足假设2.2，标准构型必须是全局刚性的。为了满足假设3.1，至少选择 $d + 1$ 个标准构型中在 \mathbb{R}^d 上进行仿射变换的智能体为领航者。

编队的标准构型如下图所示：

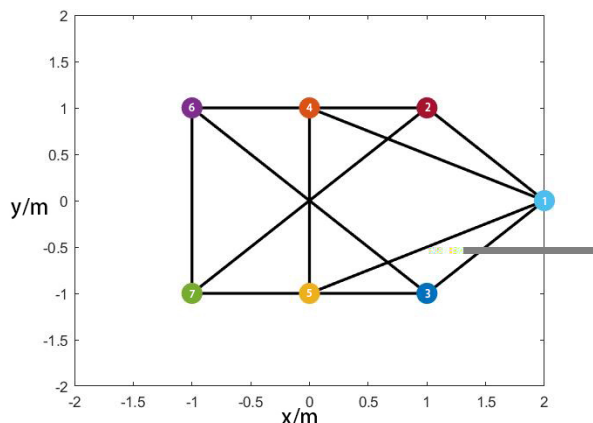


图 3.1: 编队的期望队形，其中节点 1, 2, 3 为领航者

该构型通过 2.3 节中描述的计算方法计算后得到的应力矩阵为：

$$\Omega = \begin{bmatrix} 0.2741 & -0.2741 & -0.2741 & 0.1370 & 0.1370 & 0 & 0 \\ -0.2741 & 0.6852 & 0 & -0.5482 & 0 & 0 & 0.1370 \\ -0.2741 & 0 & 0.6852 & 0 & -0.5482 & 0.1370 & 0 \\ 0.1370 & -0.5482 & 0 & 0.7537 & -0.0685 & -0.2741 & 0 \\ 0.1370 & 0 & -0.5482 & -0.0685 & 0.7537 & 0 & -0.2741 \\ 0 & 0 & 0.1370 & -0.2741 & 0 & 0.2741 & -0.1370 \\ 0 & 0.1370 & 0 & 0 & -0.2741 & -0.1370 & 0.2741 \end{bmatrix}$$

Algorithm 1 编队生成算法，使用控制方法 (3.1)

Require: 给出标准构型的关联矩阵 D 和标准构型的坐标矩阵 r

// 节点 $i \in 1, 2, 3 \dots n$, $edge \in 1, 2, 3, \dots, m$

- 1: $H = D'$
 - 2: 对 r 进行全 1 扩展生成扩展矩阵 P
//计算应力矩阵
 - 3: **for** 迭代次数 $i = 0, 1, 2, \dots, n$ **do**
 - 4: 根据公式 (4.1) 生成矩阵 E
 - 5: **end for**
 - 6: 对矩阵 P 进行 SVD 分解
 - 7: 使用 LMI 工具包根据 (4.2) 解得应力矩阵 Ω
//初始化所有的智能体初始位置
 - 8: 使用随机函数生成所有智能体的初始位置
//对控制方法进行仿真
 - 9: **while** $x - r < 1 \times -2$ **do**
 - 10: $dx = -\Omega x$
 - 11: $x = x + dt * dx$
 - 12: **end while**
-

实验过程图:

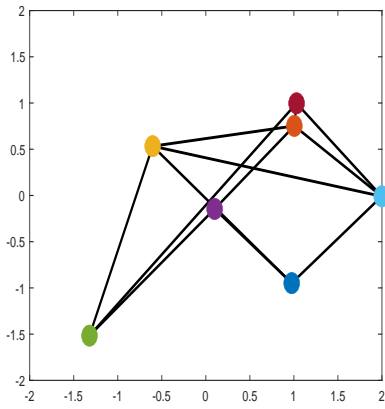


图 3.2: 实验过程图 $t=0s$

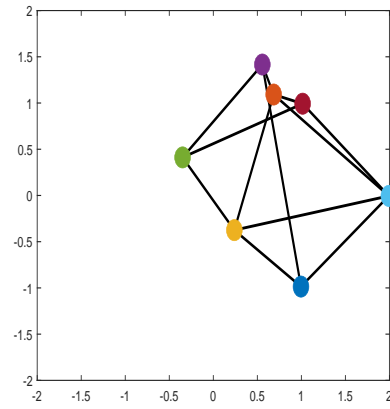


图 3.3: 实验过程图 $t=500s$

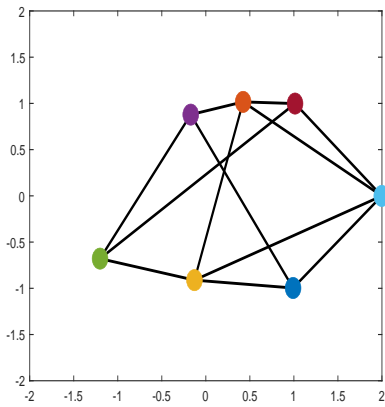


图 3.4: 实验过程图 $t=1000s$

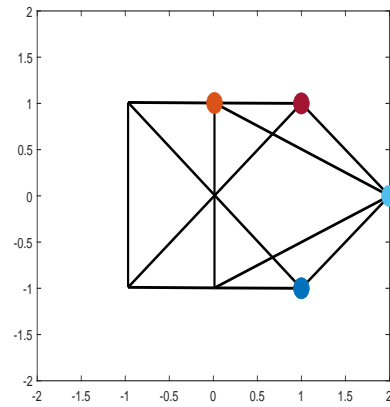


图 3.5: 实验过程图 $t=1500s$

将 4 个跟随者的位置初始化为 $P(r) = \begin{bmatrix} -2 & 0 \\ -2 & -0.5 \\ -2 & -1 \\ -2 & -1.5 \end{bmatrix}$

根据公式3.3可以计算出该控制算法的追踪误差如下图：

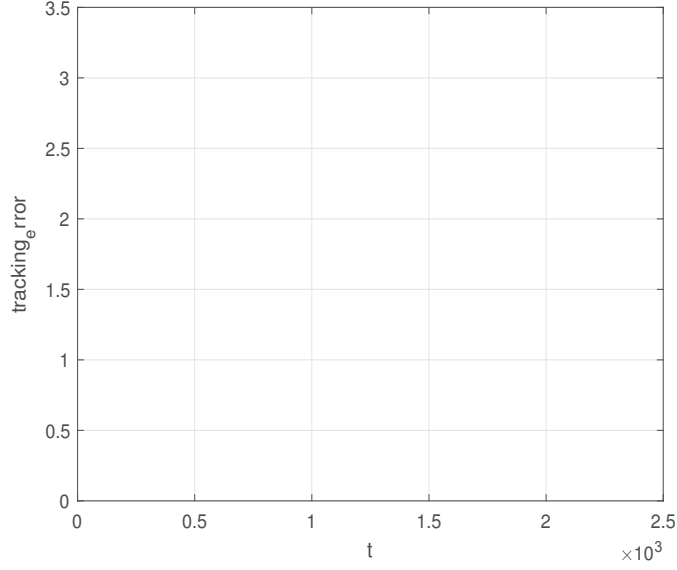


图 3.6: 算法无噪声情况下的跟踪误差

从实验结果可以看出，控制规则 (3.1) 可以有效控制编队生成。

3.2 领航者移动

3.2.1 控制算法

如果领航者以固定不变的非零速率进行移动，那么控制规则 (3.1) 将无法保证零跟踪误差。为了解决这个问题，我们引入另外的积分项并提出以下比例积分控制规则：

$$\dot{p}_i = -\alpha \sum_{j \in \mathcal{N}_i} w_{ij}(p_i - p_j) - \beta \int_0^t \sum_{j \in \mathcal{N}_i} w_{ij}(p_i(\tau) - p_j(\tau)) d\tau \quad (3.4)$$

其中 α 和 β 是正的常量控制增益。值得注意的是，控制规则 (3.4) 相对于 (3.1) 并没有额外增加新的测量量。通过引入新的变量，可以将 (3.4) 的积分项写成：

$$\begin{aligned} \dot{p}_i &= -\alpha \sum_{j \in \mathcal{N}_i} w_{ij}(p_i - p_j) - \beta \xi \\ \dot{\xi} &= \sum_{j \in \mathcal{N}_i} w_{ij}(p_i - p_j), i \in \mathcal{V}_f \end{aligned} \quad (3.5)$$

其中 $\xi = [\dots \xi_i^T \dots]^T \in \mathcal{R}^{dn_f}$ 。(3.4) 的矩阵形式为：

$$\begin{aligned} \dot{p}_i &= -\alpha \bar{\Omega}_{ff} p_f - \alpha \bar{\Omega}_{fl} p_l^* - \beta \xi \\ \dot{\xi} &= \bar{\Omega}_{ff} p_f + \bar{\Omega}_{fl} p_l^* \end{aligned} \quad (3.6)$$

3.2.2 算法实现

3.2.2.1 编队形成

该仿真实验实现了智能体任意初始化位置下对于预期编队的形成。使用的控制规则为(3.4)，即领航者匀速移动的控制规则。期望队形仍使用图3.1

Algorithm 2 编队生成算法，使用控制方法 (3.4)

Require: 给出标准构型的关联矩阵 D 和标准构型的坐标矩阵 r

```

// 节点  $i \in 1, 2, 3 \dots n$ ,  $edge \in 1, 2, 3, \dots, m$ 
1:  $H = D'$ 
2: 对  $r$  进行全 1 扩展生成扩展矩阵  $P$ 
   // 计算应力矩阵
3: for 迭代次数  $i = 0, 1, 2, \dots, n$  do
4:   根据公式 (4.1) 生成矩阵  $E$ 
5: end for
6: 对矩阵  $P$  进行 SVD 分解
7: 使用 LMI 工具包根据 (4.2) 解得应力矩阵  $\Omega$ 
   // 初始化所有的智能体初始位置
8: 使用随机函数生成所有智能体的初始位置
   // 对控制方法进行仿真
9: while  $x - r < 1 \times 10^{-2}$  do
10:   $dx = -\beta\Omega y - \beta\Omega x$ 
11:   $dy = \beta\Omega(x + dx)$ 
12:   $dx(1:3,:) = dx(1:3,:) - \alpha * (x(1:3,:) - r(1:3,:))$ 
13:   $x = x + dt * dx$ 
14:   $y = y + dt * dy$ 
15: end while

```

实验过程图:

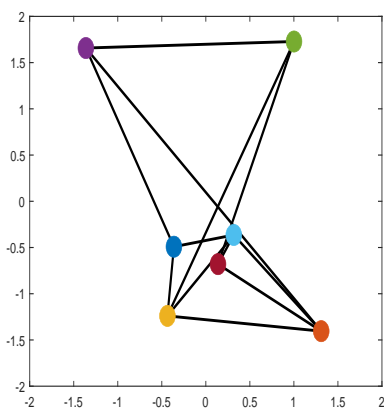


图 3.7: 实验过程图 $t=0s$

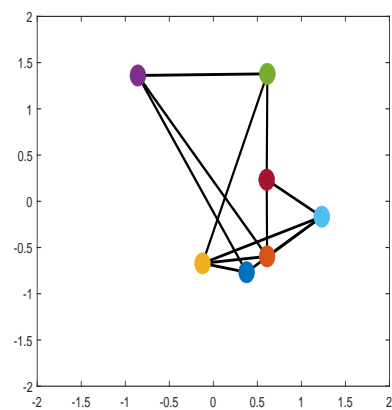


图 3.8: 实验过程图 $t=6000s$

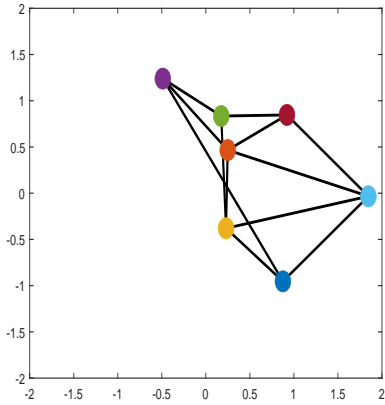


图 3.9: 实验过程图 $t=12000s$

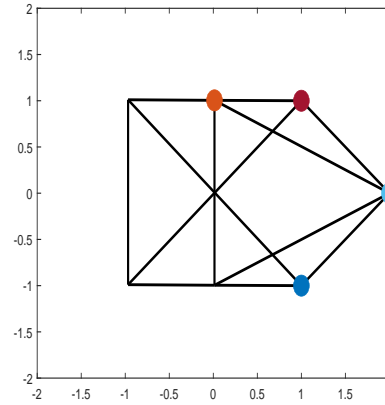


图 3.10: 实验过程图 $t=14000s$

根据公式3.3可以计算出该控制算法的追踪误差如下图：

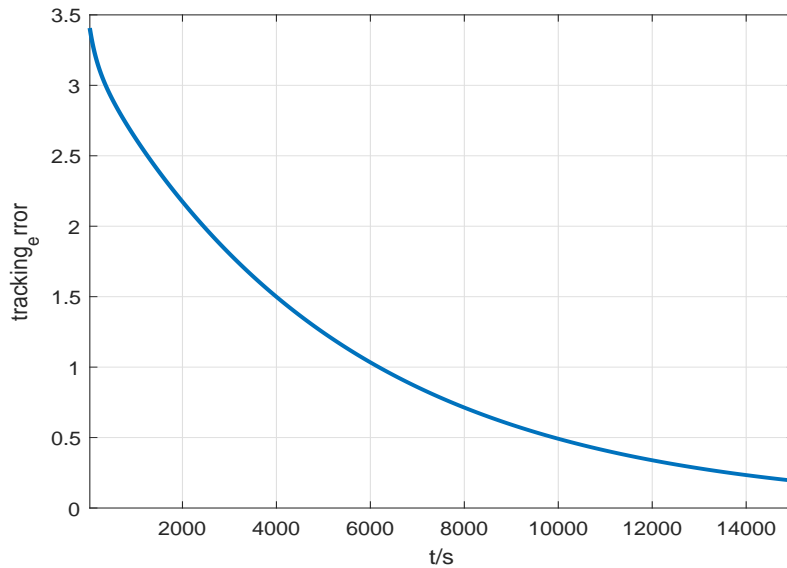


图 3.11: 算法无噪声情况下的跟踪误差

从实验结果可以看出，控制规则 (3.4) 可以有效控制编队生成。

3.2.2.2 编队行进

该算法仿真编队在行进过程中对于编队的保持和仿射变换。通过控制三个领航者的轨迹使得智能体编队完成预定的车队行进轨迹。编队整体要进行两次仿射变换和两次旋转，整个轨迹图如下：

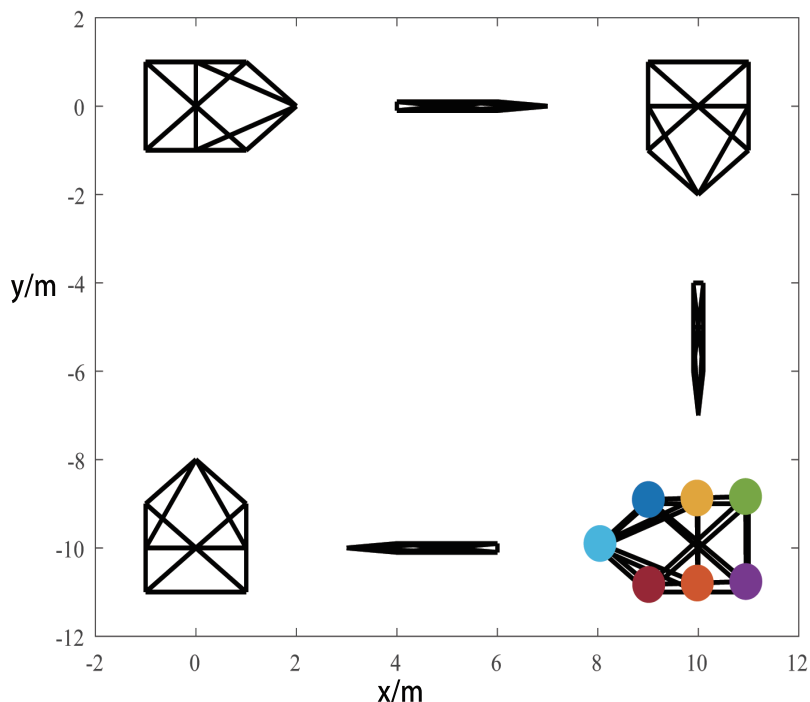


图 3.12: 编队的期望队形轨迹

Algorithm 3 编队行进算法，使用控制方法 (3.4)

Require: 给出标准构型的关联矩阵 D 和标准构型的坐标矩阵 r

// 节点 $i \in 1, 2, 3 \dots n$, $edge \in 1, 2, 3, \dots, m$

- 1: $H = D'$
- 2: 对 r 进行全 1 扩展生成扩展矩阵 P
// 计算应力矩阵
- 3: **for** 迭代次数 $i = 0, 1, 2, \dots, n$ **do**
- 4: 根据公式 (4.1) 生成矩阵 E
- 5: **end for**
- 6: 对矩阵 P 进行 SVD 分解
- 7: 使用 LMI 工具包根据 (4.2) 解得应力矩阵 Ω
// 设定领航者的行进路线
- 8: 生成领航者的平移矩阵 b
- 9: **for** $j=1:b$ 的行数 **do**
- 10: **if** 无人车达到预定仿射变换的位置 **then**
- 11: 生成矩阵 $T1$ 为形变压缩矩阵
- 12: 生成矩阵 $T2$ 为旋转矩阵
- 13: **end if**
- 14: 无人车编队的队形使用 $r * T2' * T1' + b$ 来表示
- 15: **end for**
// 初始化所有的智能体初始位置
- 16: 使用随机函数生成所有智能体的初始位置
// 对控制方法进行仿真
- 17: **for** 以 0.05 为步长，迭代直至编队达到目的地 **do**
- 18: 使用 (3.4) 控制方法
- 19: $x = x + dt \times dx$
- 20: **end for**

实验过程图:

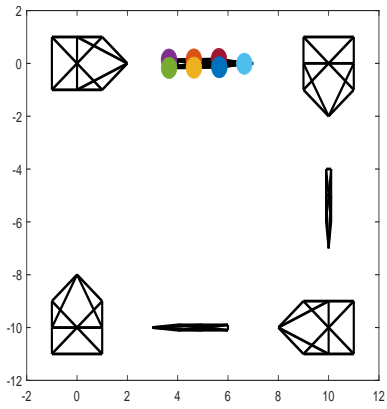


图 3.13: 实验过程图 $t=12.5s$

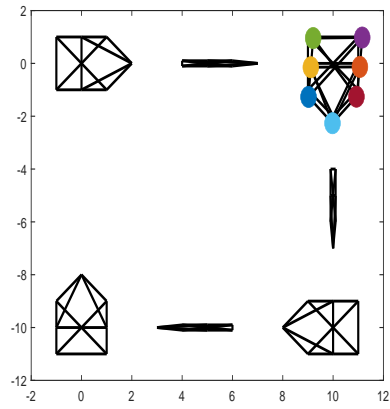


图 3.14: 实验过程图 $t=30.6s$

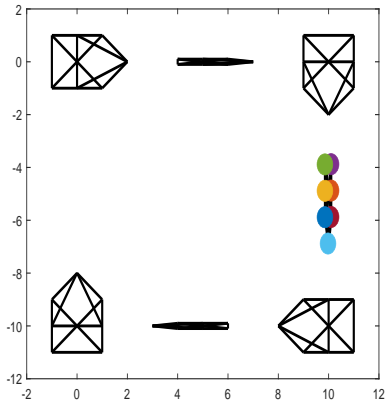


图 3.15: 实验过程图 $t=48.1s$

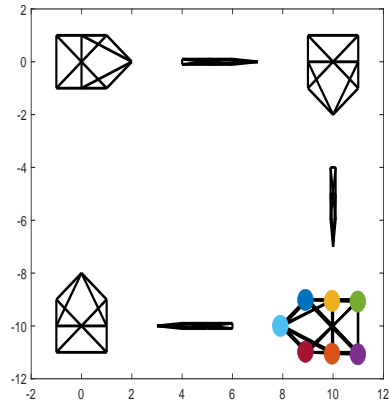


图 3.16: 实验过程图 $t=67.8s$

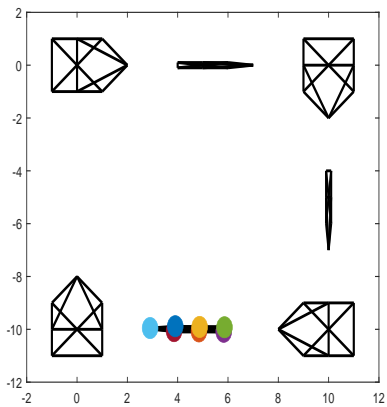


图 3.17: 实验过程图 $t=78.9s$

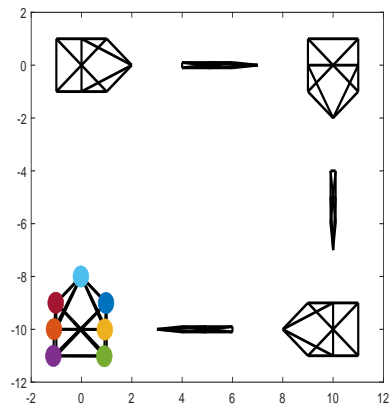


图 3.18: 实验过程图 $t=97.8s$

实验中跟踪误差如下图：

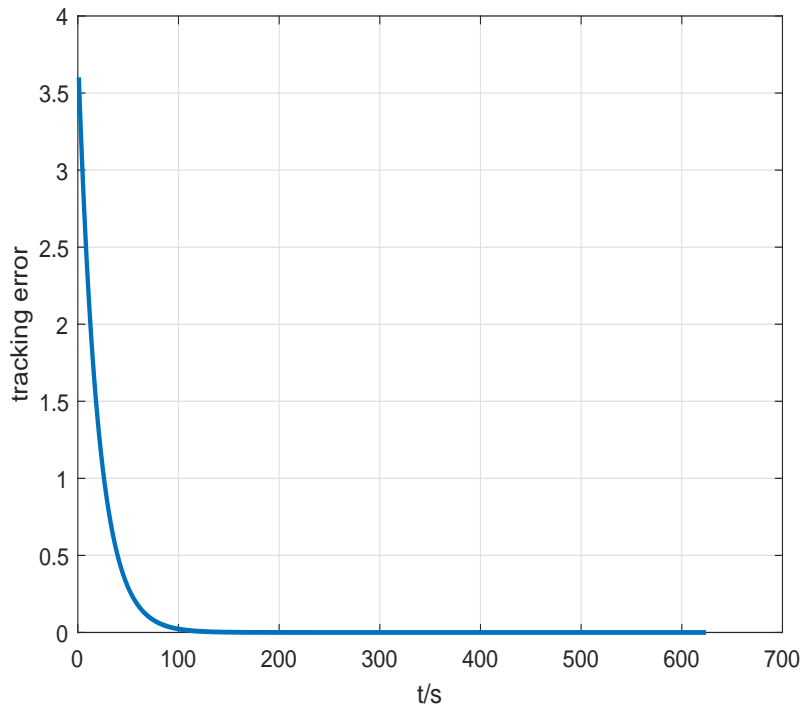


图 3.19: 无噪声情况下的跟踪误差

从实验结果可以看出，控制算法 (3.4) 可以很好地控制无人车编队完成预定行进轨迹。

3.3 本章小结

本章节给出了两种单节分的领航者-跟随者分布式编队控制算法。第一个是领航者固定的控制算法，第二个是领航者以固定速率移动的控制算法。在介绍完控制算法的公式之后对于相应的控制算法进行了复现仿真实验以证明算法的有效性并对算法的性能进行计算。对于控制算法 (3.1) 仿真了编队生成实验，对于控制算法 (3.4) 仿真了编队生成和编队行进两个实验。算法的性能通过绘制不同跟踪误差的曲线来体现。(跟踪误差的计算公式见公式3.3) 跟踪误差曲线体现了控制算法的收敛速度和收敛时间，对算法的性能有较好的体现。

第四章 分布式编队控制算法性能分析及优化

本章将会对于文章前面提到的控制算法进行性能分析，并对算法的通信次数进行优化。因为跟踪误差能很好地体现算法的收敛速度变化和收敛时间，所以算法的性能使用跟踪误差来体现。

4.1 实验一（编队形成的领航者固定控制算法）

4.1.1 抗噪声性能分析

该实验的目的为观察无人车个体间通信噪声对于编队控制算法效果的影响，检测算法的抗噪声的性能。

向控制算法 (3.1) 中增加大小为 ± 0.01 的噪声，使得跟随者观察领航者的过程中观察测量的数据有大小为 ± 0.01 误差。实验过程中的跟踪误差如下图：

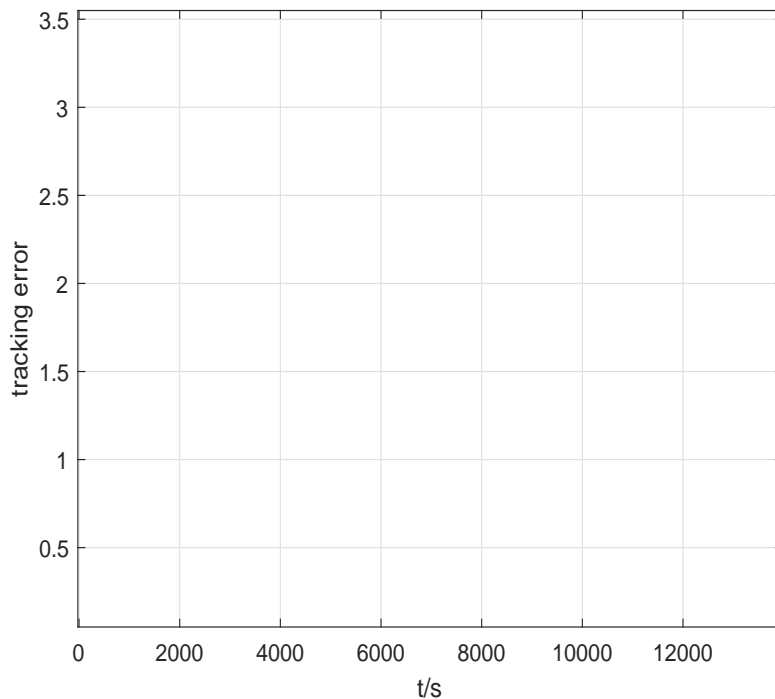


图 4.1: 有 ± 0.01 噪声情况下的跟踪误差

实验结果可以看出，算法仍然收敛，但是对比无噪声情况下（见图3.6）的收敛速度大大减慢，从 2000s 左右变为 12000s 左右。说明该算法对于噪声有一定的抵抗能力，在实际应用中即便存在测量误差也可以较好地完成编队控制任务。

4.1.2 通信优化

该实验的目的是对算法的通信次数进行优化。在实际应用中通信会产生时延并且会受到干扰，减少通信的次数和时间对于算法的实际应用有重要意义。

控制算法 (3.1) 中相邻各节点之间的通信是不间断的，各节点之间时刻通信交互位置信息。现在，对算法进行优化，令各节点之间的通信每隔 0.3s 通信一次，以减少通信次数。算法性能使用跟踪误差进行衡量，实验过程中跟踪误差如下图所示：

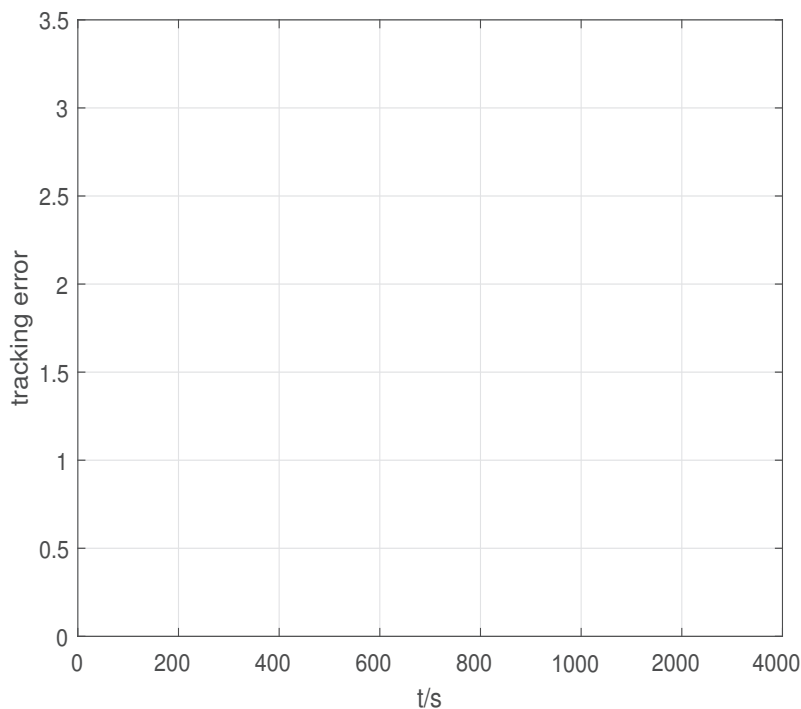


图 4.2: 以 0.3s 为周期进行通信的跟踪误差

根据实验结果可以看出优化之后的算法的收敛速度和收敛时间相较于图3.6没有较大区别，说明改进后的算法可以在对编队控制效果影响不大的情况下减少通信的次数和时间，使算法得到优化。

4.2 仿真实验二 (编队形成的领航者移动控制算法)

4.2.1 参数对于收敛时间影响

该实验的目的是研究不同的参数组合对于算法的收敛时间的影响，以期在运用算法的过程中达到最好的收敛效果。

取 1-50 之间不同的 α 和 β 组合，观察不同的参数对于算法 (3.4) 收敛时间的影响。收敛时间如下图：

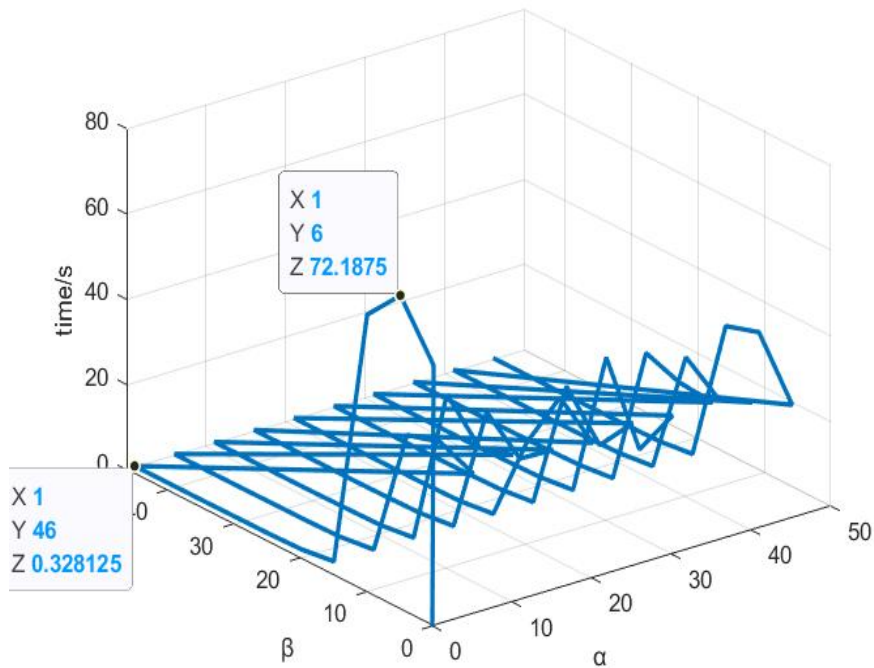


图 4.3: 参数对于算法收敛时间影响

可以看出参数对于收敛时间有较大影响。收敛时间最短的是当 $\alpha = 1, \beta = 46$ 时，收敛时间为 0.32s。当 $\alpha = 1, \beta = 6$ 时收敛时间最长，为 72s。

4.2.2 抗噪声性能分析

该实验的目的为观察无人车个体间通信噪声对于编队控制算法效果的影响，检测算法的抗噪声的性能。

向控制算法 (3.4) 中增加大小为 ± 0.01 的噪声，使得跟随者观察领航者的过程中观察测量的数据有误差。实验过程中的跟踪误差如下图：

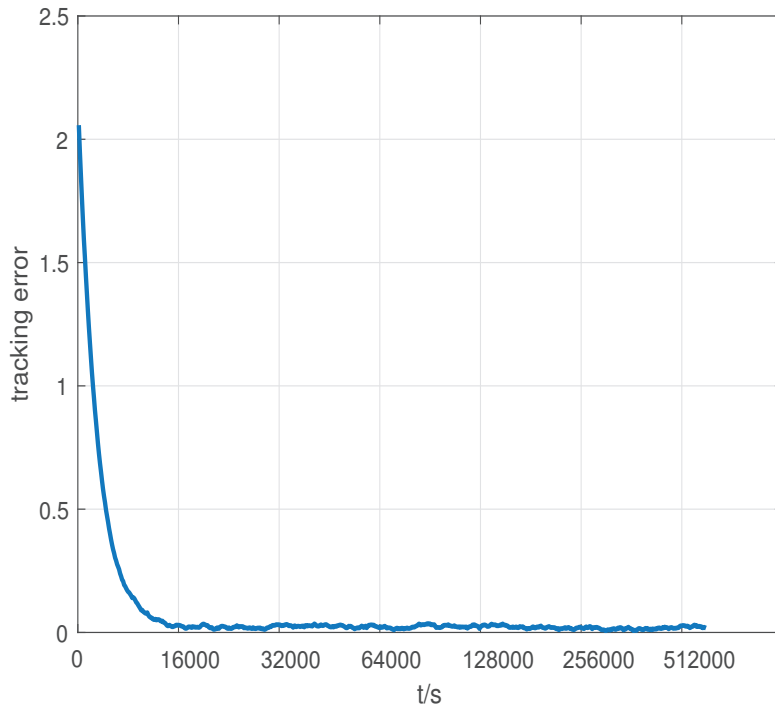


图 4.4: 有 ± 0.01 噪声情况下的跟踪误差

实验结果可以看出，算法仍然收敛，但是对比无噪声情况下（见图3.11）的收敛速度有所减慢。

4.2.3 通信优化

该实验的目的是对算法的通信次数进行优化。在实际应用中通信会产生时延并且会受到干扰，减少通信的次数和时间对于算法的实际应用有重要意义。

控制算法 (3.4) 中相邻各节点之间的通信是不间断的，各节点之间时刻通信交互位置信息。现在，对算法进行优化，令各节点之间的通信每隔 0.3s 通信一次，以减少通信次数。算法性能使用跟踪误差进行衡量，实验过程中跟踪误差如下图所示：

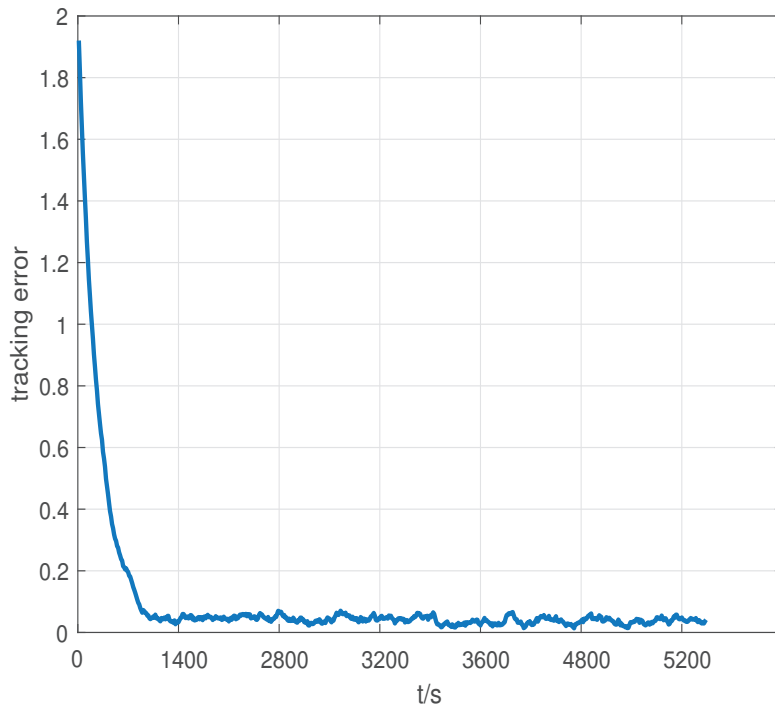


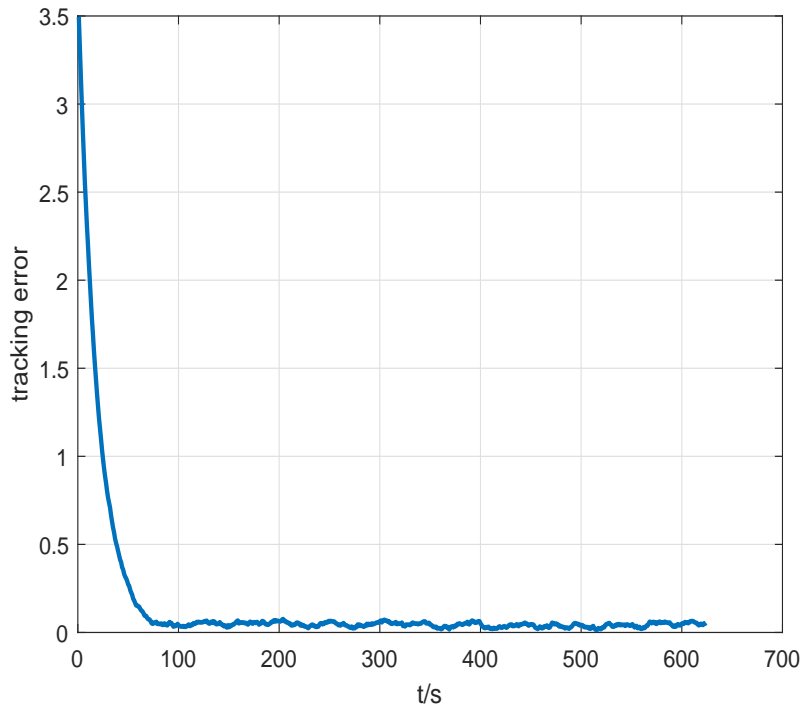
图 4.5: 以 0.3s 为周期进行通信的跟踪误差

通过实验可以看到，相较于时刻通信的算法（见图3.11），减少通信次数对算法的收敛时间影响不大。

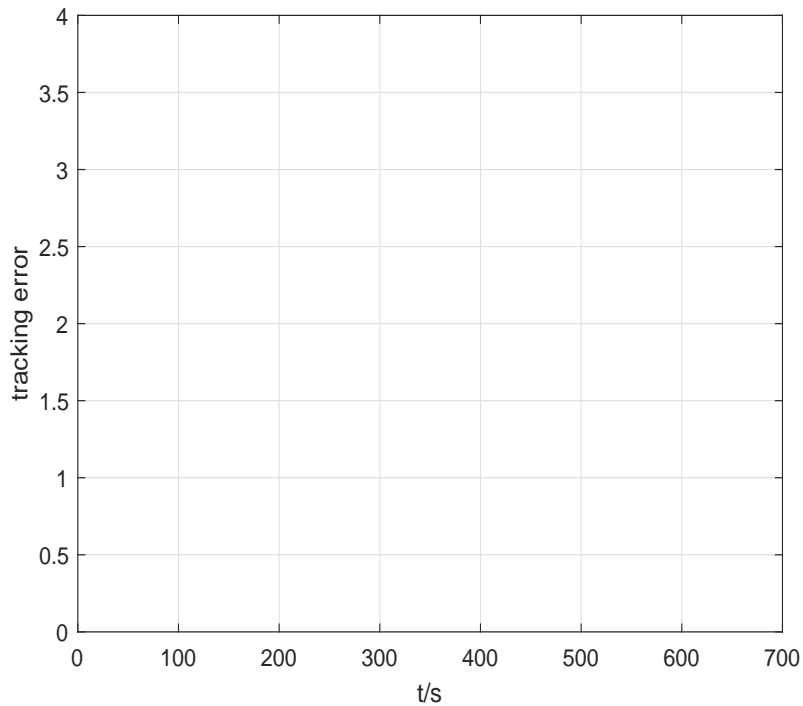
4.3 仿真实验三（编队的行进和仿射变换）

该实验的目的为观察无人车个体间通信噪声对于编队控制算法效果的影响，检测算法的抗噪声的性能。

在实验中我们尝试对跟随者的位置增加大小为 ± 0.01 的噪声，使得该分布式算法中跟随者观察领航者的位置信息为不完全准确的位置信息。在实验过程中记录跟踪误差的结果如下：

图 4.6: 有为 ± 0.01 噪声情况下的跟踪误差

可以看出，算法可以仍能收敛到目标构型，编队仍然可以完成预定的行进轨迹，但是队形会受到噪声的影响，跟踪误差在 0 附近有微小波动。当噪声增加到 ± 0.1 时，跟踪误差如下：

图 4.7: 有为 ± 0.1 噪声情况下的跟踪误差

此时编队算法已经无法收敛到目标构型，编队无法完成预定仿射变换轨迹。所以该控制算法具有一定的抗噪声能力，但是当噪声过大时也将无法收敛。

4.4 本章小结

本章主要是对于第三章中提出的两种控制算法的参数对于收敛速度的影响以及通信噪声对于性能的影响进行测试，并且对于算法中各个无人车之间通信的次数进行优化。

根据实验结果可以看出，控制算法 (3.1) 和控制算法 (3.4) 对于噪声都有一定的抗干扰能力，但是当噪声的大小增加到 ± 0.1 时算法 (3.4) 将无法收敛。

对于通信优化后的算法，实验表明算法的收敛速度和收敛时间相较于未改进的算法没有较大区别，证明优化较为成功。

第五章 总结与展望

5.1 总结

本文从队形生成、队形保持和队形行进三个问题入手对分布式多智能体控制算法进行了充分调研，总结了目前分布式控制算法的研究现状以及各种类型算法的优劣性。之后对于编队控制算法进行了详细的数学描述，并介绍了本文使用到的控制算法的相关概念。最后详细描述了参考论 [21] 中提出的控制算法对这三个问题进行了仿真，证明了算法的有效性并对比了不同参数对于算法的影响。

控制算法 (3.1) 在无噪声条件下，收敛时间大约为 2000s，在大小为 ± 0.01 通信噪声下仍可收敛，收敛时间大约为 12000s。

控制算法 (3.4) 的编队生成仿真中参数 α 和 β 对于算法的收敛速度影响较大。当 $\alpha = 1, \beta = 46$ 时收敛速度最快，为 0.32s。 $\alpha = 1, \beta = 6$ 时收敛速度最慢，为 72s。在编队行进仿真中大约在 100s 收敛至目标构型。

此外本文还对 [21] 中提出的控制算法进行了通信次数的改进。仿真显示：当以 0.3s 为周期各个节点间进行通信交互位置信息时，控制算法 (3.1) 和控制算法 (3.4) 的收敛时间相对于时刻通信都只是略有增加，影响不大。

由此可以看出，改进的算法不仅减少了通信的次数，还能很好地保持收敛速度。

具体的仿真结果可以参考第四章节。

5.2 展望

多智能体的编队控制算法属于一个较新的领域，而相较于目前使用较为广泛的集中式控制算法，分布式的控制算法能够很好地解决集中式控制方法中鲁棒性较差，对于集中控制器计算能力要求较高，不利于扩展等缺点。但是目前的控制算法仍然面临着对不确定环境的智能适应能力不足，通信延迟和噪声等对于算法影响较大等问题。

并且本文对于无人车编队控制在不确定环境中的适应性没有很好的讨论，这个方向仍然是无人车编队控制的难题。目前绝大多数的控制算法都是在默认领航者控制算法较为简单或者编队的领航者受到人为控制情况下进行讨论的。想要解决不确定环境的适应性就必须让编队不断检测周围的环境情况以及环境变化，目前计算机视觉算法比较有望能够解决这个问题。但是计算机视觉算法，例如 ORB-SLAM 算法通常对于无人车个体的计算能力有较高要求，而且鲁棒性较差。而传统的惯导，里程碑，激光测距等感知环境的测量方法对于不确定的陌生环境的感知能力较为有限且测量误差较大。所以我觉得可能使用多种测

量方式进行定位融合来消除误差可能时未来的发展方向。

无人车编队问题对于人们的生产生活有着重要的意义，解决无人车编队问题不仅能解决物流运输以及搜爆等危险作业中的问题，还可以将解决方法进行维度上面的拓展，有望解决无人机的编队控制问题。

参考文献

- [1] Hewitt C. Viewing control structures as patterns of passing messages[J]. *Artificial Intelligence*, 1977, 8(3):323–364. <https://www.sciencedirect.com/science/article/pii/0004370277900339>.
- [2] Lau S Y, Naeem W. Co-operative tensegrity-based formation control algorithm for a multi-aircraft system[J]. *Proceedings of the American Control Conference*, 2015, 2015:750–756.
- [3] 刘春, 宗群, 龚立谦, et al. 基于持久图的双轮机器人编队生成与控制 [J]. *控制工程*, 2017, 03(v.24;No.147):43–48.
- [4] Tanner H, Kumar A. Towards Decentralization of Multit-robot Navigation Functions[C]. In: *IEEE International Conference on Robotics & Automation*. 2005.
- [5] Duran S, Gazi V. Adaptive formation control and target tracking in a class of multi-agent systems[C]. In: *International Conference on Control*. 2014.
- [6] Spooner J T, Passino K M. Stable Adaptive Control Using Fuzzy Systems and Neural Networks[J]. *Fuzzy Systems, IEEE Transactions on*, 1996, 4(3):339–359.
- [7] Balch, T., Arkin, et al. Behavior-based formation control for multirobot teams[J]. *Robotics and Automation, IEEE Transactions on*, 1998.
- [8] Lawton J, Beard R W, Young B J. A decentralized approach to formation maneuvers[J]. *IEEE Transactions on Robotics & Automation*, 2004, 19(6):933–941.
- [9] Das, Aneek, K., et al. A Vision-Based Formation Control Framework.[J]. *IEEE Transactions on Robotics & Automation*, 2002.
- [10] Desai, Jaydev, P., et al. Modeling and Control of Formations of Nonholonomic Mobile Robots.[J]. *IEEE Transactions on Robotics & Automation*, 2001.
- [11] Runge J. Necessary and sufficient graphical conditions for optimal adjustment sets in causal graphical models with hidden variables[J]. *arXiv e-prints*, 2021:arXiv:2102.10324.
- [12] Ren W, Beard R W, Atkins E M. Information consensus in multivehicle cooperative control[J]. *IEEE Control Systems Magazine*, 2007, 27(2):71–82.
- [13] Cao, Yongcan, Yu, et al. An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination[J]. *Industrial Informatics IEEE Transactions on*, 2013.
- [14] A K K O, B M C P, B H S A. A survey of multi-agent formation control[J]. *Automatica*, 2015, 53(C):424–440.

- [15] Coogan, S., Arcak, et al. Scaling the size of a formation using relative position feedback[J]. Automatica Oxford, 2012.
- [16] Zelazo D, Giordano P R, Franchi A. Bearing-Only Formation Control Using an SE(2) Rigidity Theory[C]. In: IEEE Conference on Decision & Control. 2015.
- [17] Han T, Lin Z, Zheng R, et al. A Barycentric Coordinate-Based Approach to Formation Control Under Directed and Switching Sensing Graphs[J]. IEEE Transactions on Cybernetics, 2017:1–14.
- [18] Lin Z, Wang L, Han Z, et al. Distributed Formation Control of Multi-Agent Systems Using Complex Laplacian[J]. IEEE Transactions on Automatic Control, 2014, 59(7):1765–1777.
- [19] Han Z, Wang L, Lin Z, et al. Formation Control With Size Scaling Via a Complex Laplacian-Based Approach[J]. IEEE Transactions on Cybernetics, 2016, 46(10):2348–2359.
- [20] Wang, Lili, Fu, et al. Necessary and Sufficient Graphical Conditions for Affine Formation Control[J]. IEEE Transactions on Automatic Control, 2016.
- [21] Zhao S. Affine Formation Maneuver Control of Multiagent Systems[J]. IEEE Transactions on Automatic Control, 2018, 63(12):4140–4155.
- [22] A R C, b T J, C W W. Generic global rigidity of body–bar frameworks[J]. Journal of Combinatorial Theory, Series B, 2013, 103(6):689–705.
- [23] Gortler S J, Thurston D P. Characterizing the Universal Rigidity of Generic Frameworks[J]. Discrete & Computational Geometry, 2014, 51(4):1017–1036. <http://dx.doi.org/10.1007/s00454-014-9590-9>.
- [24] Alfakih A Y, Ye Y. On affine motions and bar frameworks in general position[J]. Linear Algebra & Its Applications, 2013, 438(1).

致 谢

在这里，我最要感谢的是我的导师杨绍富教授。我两年前跟随他进行科研工作，参与军科委无人车无人机的项目进行计算机视觉方面和人工智能的学习。大四又在他的指导下完成了我关于多智能体体系的分布式控制算法的毕业设计。他不论在学习还是生活上都都很关心我，不仅在科研方面教我如何找论文，阅读论文以及解决一个陌生的问题应该使用什么思路，而且也与我讨论了许多我毕业之后的学习和工作的发展问题，谈了很多他自己的见解，对我而言收益颇丰。本科期间跟随他进行科研工作让我懂得，科研工作就是踏着前任的脚步继续向前，积跬步以至千里，集川流以成江海。也许最后解决这个问题的路并不是你所探索的这一条，但是你的工作对于这个问题的解决也是有意义的。每个人向前的脚步最后会成为量变到质变飞跃的一块基石。本科期间的实验室工作将会是我一生中珍贵的回忆。

虽然我毕业之后要进入深圳计算机科学研究院工作，并没有继续读研深造。但是不论工作还是读研，最重要的事还是提升自己解决问题的能力，对行业和计算机知识体系的思考是无处不在的，只是形式略有不同罢了。工作不应当只专注于赚钱，读研也不应当只专注于拿到那个文凭。

其次感谢我的父母在大四一年中给予我很多帮助，不论我做出什么决定他们都表示了支持。同时自己对于成年之后还让他们担心自己，为自己的未来操心而感到很愧疚。

最后感谢我的朋友马瑞、霍然、胡艺凡等，感谢你们在我做毕设期间一直督促和鼓励我，一起在图书馆的研讨室中度过了许多时光，感谢你们的一路陪伴。感谢吴林婧和杨临风在使用 latex 排版方面帮助我解决了许多问题。

至此我的大学生涯即将结束，回顾自己的大学四年，觉得自己也算是不负时光。大学期间努力学习，参加了科研项目。虽然那段时间很艰难，但是所幸自己坚持了过来。除了学习和科研，我大一加入学生会，到大二成为部长，看着自己的小孩们不断成长。看着自己制作的宣传品被大家看到，被许多人收藏，感叹自己在学生会中也度过了许多快乐的时光。大三的时候前往内蒙古进行支教，第一次登上讲台，第一次感受他们的生活。在离开的时候跟所有的孩子们约定在东南大学等他们，在回南京的火车上读他们写给我的信，听他们说想我。当然，贯穿我大学生涯的除了学习就是打篮球了。从新生杯到院系杯，到加入校队之后打 CUBA，打第二次院系杯，打省赛，最后在毕业前完成第三次院系杯的比赛。每一场球赛，即便是一场必输的战争，我们也未曾放弃。就在一周前我完成了大学期间最后一次半决赛，陪伴我的朋友走完了最后一程。我自己也完成了从大一不会运球到首发控卫的蜕变，也算是一种圆满。

感谢所有在大学期间陪伴我的朋友们，也许只是短暂相处，但是我将永远记得十八岁那年见到你们的那个初夏。