Constructing Tensegrity Frameworks and Related Applications in Multi-Agent Formation Control

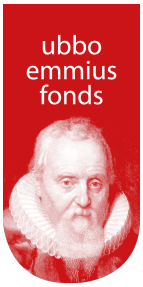
Qingkai Yang

university of groningen

The research described in this dissertation has been carried out at the Faculty of Science and Engineering, University of Groningen, The Netherlands, within collaboration between the Engineering and Technology Institute Groningen (ENTEG) and the School of Automation, Beijing Institute of Technology (BIT), China.

disc

This dissertation has been completed in partial fulfillment of the requirements of the Dutch Institute of Systems and Control (DISC) for graduate study.



This work was supported by the Ubbo Emmius Scholarship.

Cover design: Iliana Boshoven-Gkini || AgileColor.com
Printed by : ProefschriftMaken || www.proefschriftmaken.nl
The Netherlands

ISBN (book): 978-94-034-0694-7
ISBN (e-book): 978-94-034-0693-0

university of groningen

Constructing Tensegrity Frameworks and Related Applications in Multi-Agent Formation Control

PhD thesis

to obtain the degree of PhD at the
University of Groningen
on the authority of the
Rector Magnificus Prof. E. Sterken
and in accordance with
the decision by the College of Deans.

This thesis will be defended in public on

Friday 25 May 2018 at 11.00 hours

by

Qingkai Yang

born on 3 August 1988
in Hebei, China

Supervisors
Prof. M. Cao
Prof. J.M.A. Scherpen

Assessment Committee
Prof. H.L. Trentelman
Prof. Z. Lin
Prof. H.S. Ahn

To my wife, Yuhan Wang, and our family without whose supports this thesis would never have been written.

This thesis is also dedicated in memory of my beloved mother, Huayin Duanmu. May she rest in peace.

Acknowledgments

On March 15th 2014, I collected the first stamp on my passport from Beijing custom, which indeed opens an unforgettable journey in my life. Now, my Groningen time almost comes to an end even though I really do not want to stop the excellent journey so quickly because of the beautiful, quiet town as well as the lovely people I met here. I firmly believe that joining DTPA group was the best decision I have ever made when looking back at the four years' working experience.

First, I would like to express my sincerest gratitudes to my advisor, Prof. Ming Cao, for giving me this opportunity to conduct PhD research in Groningen. I really appreciate his patient guidance, strict criticism and career suggestions, without which I might have been struggling in the large amount of papers. I also wish to show my deepest thanks to him for the full freedom in research topics and numerous efforts for revising papers and this thesis. Under his supervision, I have learned not only how to establish critical thinking but also the philosophy of life. If I have to set a role model for my future life, it must be nobody but him. I also want to thank Mrs. Jing Cao for inviting me and my wife to their home. We enjoyed a lot chatting with them and playing with lovely Miumiu and Mimi. I sincerely wish them grow up healthily and be surrounded by happiness everyday. I also would like to thank my promoter Prof. Jacquelien Scherpen for providing me with constructive suggestions for the organization of this thesis. Really appreciate her careful reading and comments that make this thesis more charming. With her support, I am so grateful that I can present my research work on behalf of our DTPA group during annual ENTEG meeting in October 2016.

I deeply thank my reading committee members, Prof. Harry L. Trentelman, Prof. Zhiyun Lin, and Prof. Hyo-Sung Ahn, for spending time reading the manuscript and returning me valuable comments from different perspectives.

I also would like to express my deep thanks to Prof. Hao Fang, my advisor in Beijing Institute of Technology. I feel extremely lucky to be his first PhD student, and he spares no effort to train me from a real bachelor who knows nothing about

research to a qualified PhD student. I appreciate very much his countless help, support and encouragement during my whole PhD period. Both of his enthusiasm in working and keen insights on scientific research have influenced me a lot, which always inspire me to constantly explore more unknowns. Besides Prof. Hao Fang, I also would like to thank Prof. Jie Chen and Prof. Lihua Dou for providing me with the opportunity to study abroad and the help for academic position application in BIT. I have indeed benefited a lot from their valuable suggestions and the training program schemed for me.

I would like to give my special thanks to my collaborators. Prof. Brian D. O. Anderson from Australia National University, I truly cannot admire you more from the first time we met in Groningen. I still remember that I was so nervous to talk with such a celebrated scientist. However, your patience, rigorous and sound mathematical analysis, and energetic status have conquered me. All of these will be cherished in my whole life. Dr. Zhiyong Sun, thank you for your selfless sharing with me on many many aspects. I really enjoy discussing with you and appreciate your critical thinking. You are such a nice guy that I believe no one can refuse to be a friend with you.

My sincere thanks also go to my dear colleagues. I have learnt a lot from the discussions with Prof. Bayu Jayawardhana, and his rich knowledge in mathematics really impressed me. It is a pity that our collaboration work ends up with a failure. I am extremely grateful to Frederika Fokkens, Johanna M. Tinga, and Karen Meyer for countless help in many aspect. I also would like to thank Sietse Achterop and Martin Stokroos for your technical support on the robots configuration in our lab. Prof. Claudio De Persis, thanks for organizing colloquiums in various topics. You always bring a dozen of papers with you when attending every talk.

Special thanks to Nelson Chan for translating the summary of this thesis into Dutch and Alain Govaert for proofreading it. Hector Garcia de Marina Peinado and Desti Alkano, my officemates when I came to Groningen, you guys bring me so many enjoyable memories. Thank you Hector for your patient guidance and technical discussion on formation control problems and experiment setup. Desti, thank you for sharing your experience on research, family, and work etc. Also thanks go to my current officemates, Tony Xue, Muhammad Zaki Almuzakki, and Agung Prawira Negara, who give me a relaxed and warm working environment. Yuri Kapitanyuk, thank you for your critical comments in almost every presentation and vast efforts made for the lab session of Robotics. Hadi Taghvafard, really appreciate for your kindness and help whenever and wherever I need. Also I think you should be awarded the 'model worker' since you must be in the office every time I come to work even at weekends. Xiaoshan Bai, you are also counted. Yu Kawano, Alain Govaert, Hildeberto Jardón Kojakhmetov, Carlo Cenedese, James Riehl, Yuzhen Qin, and Shuai Feng, thank you guys for giving me a fantastic time in basketball courts. Alain, you are a good defender by your funny words since

nobody can focus playing with a joker. I guess this must be your defense strategy. Dinh Bao Nguyen, a Hungarian who always receives greeting words in Chinese, can speak only one sentence in Chinese, i.e., "I am not Chinese; I can not speak Chinese". I also had a great time playing football with you. Thank you for passing me so many balls in our football matches. Hongyu Zhang, I can not love your "Steamed Chicken with Chili Sauce" more. Also many thanks go to Liangming Chen, Yanjun Lin, Jing Guo, Matthijs de Jong, Weijia Yao, Luke Gong, Xiaodong Cheng, Martijn Dresscher, Rully Tri Cahyono, Michele Cucuzzella, Pablo Borja, Miao Guo, Ning Zhou, Mingming Shi, Tobias Van Damme, Erik Weitenberg, Tjardo Scholten, Sebastian Trip, Monica Rotulo, Alessandro Luppi, Henk van Waarde, Danial Senejohnny, Nima Monshizadeh, Jesus Barradas Berglind, Junjie Jiao, Jiajia Jia, Rodolfo Reyes-Báez, and Jieqiang Wei. Probably, I have not counted everybody.

Jie Huang, my academic brother from BIT to RUG for almost seven years, many thanks for your help, support, and sharing. Wish you success in your new position. Fan Zhang, thanks for sharing the interesting stories. Weiguo Xia, we get to know each other from a cycling to the south lake in Groningen. Thank you for giving me so many valuable suggestions whenever I have trouble. Chen Wang, I truly appreciate your sharing of the experience on double degree program, which motivates me quite a lot. Pouria Ramazi, thanks for your jokes and concerns on my health. I am sorry that I cannot be your paranymph due to the time conflict with my Chinese PhD promotion date. Anton Proskurnikov, thank you for helping me solving the stubborn problems, which you might think very easy though. Jianlei Zhang, I should say sorry that the bicycle you left me was stolen. I swear I locked it very well. I wish you all great success in your positions.

Yuzhen and Shuai, thank you for being my paranymphs. I really enjoy working with your guys. I guess you might think this is too official for you two. Ok, then let me tell you the truth. What makes me more exciting is playing basketball and billiards, and drinking alcohol with you guys since I always feel I am the world champion. To be serious, Yuzhen and Shanshan, I really enjoy traveling with you this spring in Italy. Thank you for your help and understanding during the whole trip. Wish you happiness and success in your future life. Shuai, I can recall that the room is filled with happy laughters and cheerful voices every time we have parties. Thank you for your funny jokes. You must be an excellent comedian if you took show business, which has been proved in Groningen Spring Festival Gala twice. Wish you good luck in everything.

I am very grateful to my Chinese friends I met in Groningen. Please allow me to express my sincere thanks to you in Chinese. 涛哥（袁涛），逗比的外表下其实是一枚纯金暖男，感谢涛哥无数的大餐和接济，你的大腿我是抱紧不松手了。也感谢周丽嫂子在法国的热情款待，美味的正宗法餐至今还让我念念不忘。祝涛哥和周丽嫂子回国发展一切顺利，小宝宝健康快乐成长，鱼塘早日建成。全哥，影帝兼格村的交际花，我刚来格村时的领路人，帮了我很多很多，真诚的感谢。也

祝你和忠洪，禾禾，还有即将出生的二宝在瑞典生活幸福，硕果累累。想想忠洪怀孕时，我们竟然还通宵升级，疯狂至极。全哥，期待下次打牌你能给力点啊，哈哈。老唐（建军），王丽，程振，非常感谢你们一家人一直以来无私的帮助和庇护，相见恨晚但不妨碍一见如故。你家真是魔力太大，萌萌一直喊着要搬去你们那住（此处有捂脸表情）。祝小糖果越来越美丽，懂事，希望我们回国后还是好邻居，你们懂得。程振，注定有缘啊，我答辩之日，你回国之时，期待早日收到你的喜讯。老王（真的就住隔壁啊！）和晶晶，我们多次畅谈家庭、人生、社会等等，各抒己见，非常过瘾。感谢你们的关心和帮助，祝你们这对好cp也恩爱到老。武静，很佩服你的勇气，遵从自己的内心，为自己而活，你肯定会幸福的。成勇，姜黎，未来的荷兰人，感谢这几年间我们经历的一切，让我们都成长了很多。祝你们以后在荷兰生活美满，小宝宝健康茁壮成长，记得在你们的大豪宅里留一间客房哈。段丛，耿直的妹子，这几年我也是见证了你的酒量从一杯就倒到一瓶起步的变化，我的酒吧行就靠你帮我实现了。花拉，善良的妹子，感谢你多次的热情招待。你的变化我们都有目共睹，学习能力不能说不强。

感谢身边的小伙伴们。义飞，肄辉，分分钟虐哭老外的名字组合，祝你们留守格村一切顺利。军哥（李军）和虹羽，两个享受超大豪宅的人，下次去科隆打卡就靠你们了。唐教授（名振辰），小明，华堂，P楼的扛把子们（虽然唐教授已经搬离去筑爱巢了），祝你们早日成为科研界的翘楚。洪燕，你的卡依然睡在我家，德国生活很快乐吧，祝早日完成学业。许瑾，感谢多次邀请我们去你家参加大party，祝你在荷兰生活一切顺遂。亚楠，格村的网络名人，祝一直美丽下去。力强和媛媛，才华和颜值兼具的一对组合，期待早日看到你们的小宝宝啊。滨琪&仕莉，王帆&邵闫，很开心和你们相识相知，多次被你们的激情感染到眩晕，祝你们一路高歌猛进，未来大展宏图。还要感谢刚来荷兰认识的“前辈们”。涛哥（张涛）、文君姐和暖暖一家（我格村足球生涯的领路人，也是荷兰大闸蟹的探索者）；其宏，足球场上的闪电侠；国伟，大蒜痴汉；刘院士（刘凯）娟娟和Esther、Grace一家（格村的传奇，把青千做为保底的科研牛人，祝你们一家在美国一切美满幸福）；飞哥（郑立飞）、飞嫂和鲁鲁一家（同样为学术巨擘，足球场上标致性动作，一扣一拉，过掉无数人）；硕哥（杨硕）、婧怡和团团一家（低调做人，高调做事的典范）；陈院士（家文）、蓓蓓和Robin及弟弟一家（诺奖团队的实际掌舵人，事业家庭双丰收的人生赢家）；谢艾（第一个带我去AH超市的人）。感谢环Planetenlaan的朋友们：石老板和佳盼（祝生意兴隆），张蕊姐、Robert和Rose、Roderick一家（祝小朋友们早日长大）；培亮和丽娟，延军和培培，晓明和徐萍一家。我还要感谢一票从足球队认识的风流少年们，让我曾经有过非常美好的时光。老曹（曹豪杰，屁股防守发挥到极致），老刘（刘博群，身体本身就是一堵墙），李牧（笔名，真名张彦曦，隐藏的很深，在厕所碰到次数最多的人），静静（顾家的好男人），石头，虎哥（王跃虎），司韵，黄刚，刘凌，卓华，老姜等。

在格村并肩作战朝夕相处的师弟师妹们（好像没有师妹）及家属们，本来准备了很长的篇幅给你们。但是限于论文结构原因，只得在这里再次说一句，感谢你们的陪伴还有理解。祝你们多出优秀成果，未来一切顺利，回国再见。在这里还要感谢过来访问的老师们，北航刘晓峰老师、沈航齐义文老师、北科刘会央老

师、燕大姚老师，给我工作方面的宝贵建议和帮助。

最后，我要特别感谢我的家人们。感谢父母亲对我的养育之恩，没有您们当年在极其贫苦的情况下依然选择牺牲自己供哥哥和我读书的伟大魄力和承受力，就没有我们的今天！感谢我的妻子，萌萌，多年来对我无私的支持、关心照顾和牺牲。我相信经过我们的共同努力，未来会无限美好。感谢我的哥哥、嫂子和大侄子皮皮，有你们这样的家人是我的福气。感谢岳父岳母一家，一直以来对我的认可和无条件的支持。还要感谢我的大姨、老舅一家人对我一如既往的帮助。

总之，我觉得自己是异常的幸运，在我目前短短三十年的人生中遇到了这么多可敬可爱的人。拥有你们，我真的非常幸福！

Qingkai Yang
Groningen
14-04-2018

Contents

[1] The majority of this section is taken from one of Prof. Brian D. O. Anderson's unpublished technical reports. For completeness of the strategy for growing locally rigid tensegrity frameworks, we put it in this thesis.

List of symbols

Chapter 1

Introduction

This thesis investigates the problem of constructing tensegrity frameworks and the related applications in formation control in the context of multi-agent systems. We will concentrate on growing tensegrity frameworks in several scenarios, including merging rigid/infinitesimally rigid/super stable tensegrity frameworks, and Henneberg constructions on super stable tensegrity frameworks. In order to make good use of the desirable features of tensegrity frameworks, such as flexible scalability and robustness, a control scheme based on virtual tensegrity frameworks is proposed to manipulate formations for carrying out different tasks. Before proceeding to the specific problems, I will briefly introduce the background, motivations, and structure of this thesis.

1.1 Background

In this chapter, the basic knowledge of tensegrity frameworks and formation control is provided. The detailed literature review will be provided at the beginning of each main chapter for specific problems.

1.1.1 Tensegrity frameworks

The English word "Tensegrity" was coined by Buckminster Fuller in the late 1950s by combining the words "tension" and "integrity" [46, 106]. This conjunction also literally implies that this class of structures is integrated by the inner tension in various geometric forms. In fact, one decade before the word tensegrity structures were named, a contemporary sculptor, Kenneth Snelson, had constructed the "X-Piece", shown in Fig. 1.1, which has been regarded as the first widely known piece of the tensegrity structure. It is made of two plywood X's placed in an interlaced manner and one stands over the other linked by nylon lines in tension. Apart from X-Piece, Snelson has also made a series of sculptures in the following decades, like "Needle Tower", created in 1968, now exhibiting in Hirshhorn Museum and Sculpture Garden, Washington and "Sleeping Dragon" created in 2002-03, now "sleeping" in Kirkpatrick Oil Company Building, Oklahoma City.

Figure 1.1: *The "X-Piece" made by Kenneth Snelson in 1948.*

Even though a variety of tensegrity structures has been built by researchers, engineers, artists, and sculptors, there exists no general form of the definition of tensegrity due to practitioners' different perspectives (see, for example, [38, 46, 97, 114]). A detailed introduction to the historical development and fundamental concepts of tensegrity frameworks can be found in [85]. In spite of the diverse definitions, one commonly-accepted statement is that a tensegrity structure consists of compression elements (i.e., struts) and tension elements (i.e., cables), with which the resulted pushing and pulling forces are balanced such that the whole structure is stable [64]. Because of the elements in compression or tension together with their carefully designed connections, tensegrity structures enjoy several remarkable features [113]: Efficiency in supporting loads; deployability to a large volume; easy adjustment; reliable modeling and control; and clear connections to many biological structures.

With these features, tensegrity structures have received extensive attention from different disciplines. Starting from explaining the molecular structure of the spider fiber [111, 121], researchers have applied tensegrity structures to model biologic organisms from cell cytoskeletons [60], cats' hind legs [37] to the spines of humans [74], and in fact, one can recognize a new research community "biotensegrity". The beauty of the tensegrity structures is admired not only by the biologists but also by the artists. As mentioned above, Snelson has created many sculptures, which are exhibited in museums and art galleries worldwide. At the same time, systematic analyses on the equilibrium conditions were initially reported in [67, 97], all of which laid theoretical foundations for the achievements afterwards. In architecture, the tensegrity concepts have also been adopted to construct shelters, bridges, roofs

and even whole buildings. These constructions embrace the advantages that, on one hand, they can deform their shapes to survive drastic movements caused by an earthquake or whatever disturbances without breakdown and on the other hand, they are much lighter yet bear higher stiffness in comparison with traditional structures [112]. Besides the extensive applications of tensegrity structures in civil engineering, people also seek to explore a new paradigm in the design and control of locomotor robots using tensegrity concepts [69, 94]. Recently, a project aiming for exploring the deeper space was launched by NASA (National Aeronautics and Space Administration), in which the core platform "Super Ball Bot" based on tensegrity structure can flexibly adjust its configuration to suit the complicated environment.

Due to the increasing applications of tensegrity structures in different fields, researchers conducted more rigorous mathematical analysis much beyond the geometrical interpretations in the early stage. One of the most important problems is to identify an equilibrium configuration, known as form-finding [64, 80, 122], which falls out of the scope of this thesis, and thus we skip the discussion on this problem. Instead, we will focus on the theoretical study of tensegrity structures in the context of rigidity graph theory. To keep consistent with the conventions in graph theory, we will refer to tensegrity structures as "tensegrity frameworks" in the rest of this thesis. The concepts related to rigidity in terms of bar frameworks were extended to tensegrity frameworks in [104]. Then Connelly investigated the local rigidity conditions in terms of the stress-based energy function [19]. Later, the concepts of second-order rigidity and pre-stress stability for tensegrity frameworks were established in [26], where the physical parameters and the stress were linked via the Hessian matrix. In addition to these, global rigidity [23], super stability [21], and iterative universal rigidity [24] of tensegrity frameworks were also studied based on the stress matrix.

1.1.2 Distributed formation control

The last two decades witnessed sustained considerable efforts on distributed formation control, which is one of the central topics in the context of cooperative control of multi-agent systems. Initially inspired by the collective behavior of groups of animals, such as birds and fish, people gain new insight into the control of complex systems. Then distributed control schemes were proposed based on only local interactions with neighbors in contrast to the all-to-all or all-to-one communications in centralized control. Consequently, the distributed control systems can obtain more benefits because of the facts that they have low operational cost, high robustness to disturbances and system failure, and flexible scalability [17]. Due to these advantages, formations of robots have been employed to carry out various tasks, such as satellites flying in a certain shape to explore the deep space, drones

flying in formation to transport goods, and wheeled robots moving in an organized pattern to map an area [10].

The objective of formation control is to achieve some prescribed formations normally specified by relative positions or pairwise distances among agents. According to the collective behavior of the whole group, formation control can be roughly categorized into two scenarios [102]: *formation producing (stabilization)* refers to the convergence of team agents to some pre-defined feasible geometric shapes by running control laws; *formation tracking* refers to formation stabilization, in the meantime, following a given leader or a given reference signal. This implies that formation tracking control can be regarded as the integration of formation stabilization and trajectory tracking control. It is worth noting that the realization of formation tracking is more than just simple addition between formation stabilization and trajectory tracking due to the coupling effect caused by the sub-controllers to the other.

As summarized in the recent survey paper [91], the approaches to solving formation control problems can be generally classified into position-based, displacement-based, and distance-based control according to the sensing and communication variables. By invoking the position-based control, agents can move towards their desired positions individually, which means that a global coordinate system is compulsory. To remove such a restrictive requirement, the displacement-based control measures the relative positions, and thus relies on local coordinate systems but with consistent orientations. Among these three strategies, the distance-based control is the most efficient approach in practice, since the inter-agent distances can be obtained in a fully local manner in the sense that neither a global coordinate system or the same orientation is required. However, in general, only local convergence can be expected from the gradient system using distance-based control. Many results have been reported on this issue, such as [13, 29, 30, 70].

Apart from these basic tasks discussed above, formations are also required to vary in size or even in shape to adapt to the changing environment in some situations. For example, a team of flying drones needs to shrink its formation size to pass through some restricted areas. We call this type of transformation, i.e., altering only the formation size without changing the geometric shape, *formatioin scaling*. This issue will be addressed in this thesis by investigating how to control a small number of agents to recast the size of the formation. In addition, it is also a common phenomenon that as a formation of robots moves, it might split into sub-formations and then merge the small portions to form a bigger whole after a certain period of time, for the purpose of obstacle avoidance, predator avoidance, or target enclosing [5]. The merging problem will also be discussed in the context of tensegrity frameworks in this thesis.

1.2 Outline and main contributions of this thesis

Chapter 2 first provides some notations employed in this thesis, followed by the basic concepts of graphs and bar frameworks together with rigidity theory. Parallel to the bar frameworks, Section 2.3 presents the definitions and preliminaries of rigidity theory associated with tensegrity frameworks.

Chapter 3 deals with the problem of how to preserve infinitesimal rigidity and rigidity of the tensegrity frameworks after the merging operation in $\mathbb{R}^2$. We show that in the case of merging separate infinitesimal rigid tensegrity frameworks, there exists a set of proper self-stresses for the post-merged tensegrity framework, where the type of a pre-existing member can be maintained by checking the sign of the new stress. Then based on this self-stress, it can be shown mathematically that the combined tensegrity framework is infinitesimally rigid. Furthermore, we also show that rigidity can be expected when merging two rigid tensegrity frameworks via analyzing the distance perturbation on the linking members. For appropriate assignment of the new members, a novel method is proposed by morphing the rigidity matrix. To the best of our knowledge, no results on analyzing infinitesimal rigidity or rigidity of the merged tensegrity framework have been reported in the existing literature. The results of this chapter can serve as the theoretical foundation for formation control strategies design in practical applications.

In Chapter 4, we focus on the problem of growing super stable tensegrity frameworks. We first investigate the *vertex addition* and *edge splitting* operations on a super stable tensegrity framework along the line of classic Henneberg construction for bar frameworks. It is shown that these operations can preserve the super stability with appropriate selection of struts or cables inserted during the growing process in $\mathbb{R}^2$ ($\mathbb{R}^3$). In addition, Chapter 4 studies the merging problem for two super stable tensegrity frameworks in $\mathbb{R}^d$ under the condition that they share at least $d+1$ vertices. We present one mild sufficient condition to ensure super stability by looking into the stress matrix. When the dimension of the working space is constrained to two or three, we give detailed procedures on how they may be merged to generate a super stable framework. In the last section of Chapter 4, comparisons are made with bar frameworks in terms of the quantity of the members to accomplish the growing procedure. The results of this chapter are important not only in enriching the rigidity graph theory with respect to super stable tensegrity frameworks, but also in constructing large-scale stable tensegrity frameworks in civil engineering.

Chapter 5 explores how to construct a universally rigid tensegrity framework given any configuration in general positions. We present a numerical algorithm to derive a stress matrix, based on which a universally rigid tensegrity framework can be built accordingly. We then consider the formation control problem with constraints on the inter-agent distances, in which the strict upper or lower bounds

for pairs of agents are imposed. By projecting the multi-agent system into a virtual tensegrity framework, we propose distributed control laws to stabilize prescribed formations. One can easily use our control strategies to tackle the distance constraints within the subject of formation control in applications. For example, the vehicles cannot diverge from each other too much due to the existence of cables in tethered robots or pipes in aerial refueling.

We study the formation scaling problem in Chapter 6, aiming at changing the formation size by controlling only a small portion of the agents. The virtual tensegrity framework is again employed to model the connecting relationships among the agents as well as the weight on each edge. We first show that the size of the formation in $\mathbb{R}^d$ can be altered by d pairs of agents whose position vectors span the whole space $\mathbb{R}^d$. To further reduce the number of informed agents, we design another class of stress-based formation scaling control laws involving orthogonal projections used to drive the agents to correct directions. In this circumstance, it is shown that one pair of informed agents is sufficient to determine the size of the whole formation. Moreover, we discuss the equilibria when the stress agrees with a generic universally rigid tensegrity framework. As an extreme case, when only one agent is informed of the size of the formation, we design a new type of distributed estimator-based control algorithms, with which the formation scaling problem can be solved. This greatly improves the feasibility of the control laws in the terms of communication and sensing requirements. Last but not least, by introducing the negative weights, more interaction models can be involved to comprehensively study the mechanism of coordination.

In Chapter 7, we address the formation tracking problem for multi-agent systems, where the centroid of the formation needs to be controlled to follow a given reference signal. In light of the fact that the centroid of the formation is a global variable that cannot be computed easily using only local information, we design a finite-time centroid estimator for each agent. In comparison with existing results, our proposed estimator can get rid of the explicit knowledge of the bound of the agents' speed. Using the centroid estimation, distance-based formation tracking control laws are designed and stability is proved by invoking rigidity graph theory. What also deserves to be highlighted in this chapter is that the proposed control scheme can be accomplished in local coordinate frames, which can definitely broaden its applications in practice.

Chapter 8 presents the conclusions of this thesis, and provides some possible directions of interest, from my point of view, for future research.

1.3 List of publications

Journal papers:

- Q. Yang, M. Cao, H. G. de Marina, H. Fang, and J. Chen. Distributed formation tracking using local coordinate systems. *Systems & Control Letters*, 111:70-78, 2018. (Chapter 7)
- Q. Yang, M. Cao, H. Fang, and J. Chen. Constructing universally rigid tensegrity frameworks with application in multi-agent formation control. *IEEE Transactions on Automatic Control*, March 2019. (Chapter 5)
- Q. Yang, M. Cao, and B. D. O. Anderson. Growing super stable tensegrity frameworks. *IEEE Transactions on Cybernetics*, DOI: 10.1109/TCYB.2018.2826049. (Chapter 4)
- Q. Yang, M. Cao, Z. Sun, H. Fang, and J. Chen. Stress matrix-based formation scaling control. *Automatica*, under review, 2018. (Chapter 6)
- Q. Yang, B. D. O. Anderson, and M. Cao. Merging rigid tensegrity frameworks. 2018. In preparation. (Chapter 3)

Conference papers

- Q. Yang, M Cao, Z. Sun, H. Fang, and J. Chen. Formation scaling control using the stress matrix. *In Proc. of the 56th IEEE Conference on Decision and Control*, pp. 3449-3554, Melbourne, Australia, 2017.
- J. Huang, M. Cao, N. Zhou, Q. Yang, and X. Bai. Distributed behavioral control for second-order nonlinear multi-agent systems. *IFAC-PapersOnline*, 50(1), 2445-2450, 2017.
- Q. Yang, Z. Sun, M. Cao, H. Fang, and J. Chen. Construction of universally rigid tensegrity frameworks and their applications in formation scaling control. *In Proc. of the 36th Chinese Control Conference*, pp. 8177-8182, Dalian, China, 2017.
- Q. Yang, M. Cao, H. Fang, J. Chen, and J. Huang. Distributed formation stabilization for mobile agents using virtual tensegrity structures. *In Proc. of the 34th Chinese Control Conference*, pp. 447-452, Hangzhou, China. 2015.

Chapter 2

Theoretical preliminaries

In this chapter, we first introduce some general notation and definitions that will be used throughout the thesis. To make this thesis self-contained and its definitions consistent, we then mainly follow [25] to review some basic concepts on tensegrity frameworks. Among those, the key concepts from graph rigidity theory associated with tensegrity frameworks and bar frameworks will be highlighted respectively. This lays theoretical foundations in stability analysis of distributed formation control, which in turn serves as one application of tensegrity frameworks in this thesis.

2.1 Notations

In this section, we introduce some standard notations. Let $\mathbb{R}^n$ be the n-dimensional Euclidean space. $\mathbb{R}^{m\times n}$ is used to represent the set of real matrices with dimension $m \times n$. Denote by I_n the identity matrix with dimension n. We use $\mathbf{1}_n$ and $\mathbf{0}_n$ to denote the n-dimensional column vector with all ones and zeros, respectively. The subscripts will be omitted when there is no confusion in the context.

For a given matrix $X \in \mathbb{R}^{m\times n}$, $X^\top$ denotes its transpose. The rank, column space (i.e., image) and null space of a matrix X are represented by $\mathrm{rank}(X)$, $\mathrm{col}(X)$ and $\mathrm{null}(X)$, respectively. Let $\det(X)$ denote the determinant of a real square matrix X. For $x \in \mathbb{R}$, $\mathrm{sign}(x)$ is the signum function. For a vector x, $\mathrm{sign}(x)$ is defined in a component-wise manner. For a vector $x = [x_1, \cdots, x_n]^T \in \mathbb{R}^n$, $\|x\|$ represents the Euclidean norm of x, and $\mathrm{diag}(x) = \mathrm{diag}(x_1, \cdots, x_n)$ is a diagonal matrix with the vector x on its diagonal. For a matrix $V = [v_1, \cdots, v_n] \in \mathbb{R}^{m\times n}$, we use $\mathrm{span(V)}$ to denote the linear span of the elements $\{v_1, \cdots, v_n\}$. For a set $\mathcal{S}$, $|\mathcal{S}|$ denotes the cardinality of $\mathcal{S}$.

Given two matrices $A = [a_{ij}]_{m\times n}$ and $B = [b_{ij}]_{p\times q}$, the Kronecker product $A \otimes B$ is defined by

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}_{mp\times nq}.$$

2.2 Preliminary on frameworks and rigidity theory

2.2.1 Graph theory

A *graph* comprises a set of vertices and edges, in which the edges specify how the vertices are connected. We assume that the graph studied in this thesis is finite and simple, i.e., without loops or multiple edges. Let $\mathcal{V} = \{1, 2, \cdots, n\}$ and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ be, respectively, the vertex set and the edge set of a graph $\mathcal{G}$ representing the neighboring relationships between n vertices. The graph $\mathcal{G}$ is defined as the pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. A graph is said to be *directed* if the pairs of nodes are ordered, namely, a directed edge (i, j) means that the information flows along the direction from i to j, but not necessarily vice versa. In contrast, a graph is *undirected* if $(i, j) \in \mathcal{E}$ implies $(j, i) \in \mathcal{E}$ [101]. In this thesis, we mainly focus on undirected graphs, where vertices i and j are neighbors if and only if there exists an edge (i, j). The set of vertices that are adjacent to i is denoted by $\mathcal{N}_i = \{j | (i, j) \in \mathcal{E}\}$. The adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ associated with the graph $\mathcal{G}$ is defined in such a way that $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. For an undirected graph, A is a symmetric matrix.

Define the *Laplacian matrix* $L = [l_{ij}] \in \mathbb{R}^{n \times n}$ by

$$l_{ii} = \sum_{j=1, j \neq i}^{n} a_{ij},\ l_{ij} = -a_{ij},\ i \neq j. \tag{2.1}$$

It can be checked that the row sums of the Laplacian matrix equal zero, which implies that $\mathbf{1}_n$ is always an eigenvector associated with the zero eigenvalue. This property plays a key role in controller design for achieving consensus of multi-agent systems.

Under the assumption that we have assigned an arbitrary orientation to $\mathcal{G}$, the incidence matrix $H = [h_{ij}] \in \mathbb{R}^{n \times |\mathcal{E}|}$ encoding the relationships between edges and nodes is defined by

$$h_{ij} = \begin{cases} 1, & \text{if node } i \text{ is the head of edge } j, \\ -1, & \text{if node } i \text{ is the tail of edge } j, \\ 0, & \text{otherwise,} \end{cases}$$

where i and j are the indices running over the node and edge sets, respectively. With incidence matrix, the Laplacian matrix can be shown to be equal to

$$L = HH^{\top}.$$

From this property, we can observe that Laplacian matrix is always symmetric and

positive semidefinite [83].

Lemma 2.1. *[101, Lemma 2.10] Suppose that $z = [z_1^\top, \cdots, z_n^\top]^\top$ with $z_i \in \mathbb{R}^d$ and L defined in* (2.1)*. Then the following statements are equivalent.*

(a) L has a simple zero eigenvalue with an associated eigenvector $\mathbf{1}_n$ and all other eigenvalues are positive,

(b) The undirected graph of L is connected,

(c) $(L \otimes I_d)z = \mathbf{0}$ implies that $z_1 = \cdots = z_n$,

(d) Consensus is reached asymptotically for the system $\dot{z} = -(L \otimes I_d)z$,

(e) The rank of L is $n - 1$.

Remark 2.2. The graph in Lemma 2.1 is assumed to be undirected, which is a special case of the directed graph discussed in [101, Lemma 2.10].

2.2.2 Frameworks and rigidity

A *configuration* is a finite collection of n labeled points in the d-dimensional Euclidean space $\mathbb{R}^d$, denoted by $q = [q_1, \cdots, q_n]$. We say a configuration q is *generic* if the elements of q are algebraically independent over the rational numbers, namely, there is no non-zero polynomial with rational coefficients that vanishes at the elements of q [23].

Also, to avoid certain special cases, for a framework in a Euclidean d-dimensional space, an assumption is often made that the framework is at a *general position*, i.e., no k points of $q_1, \cdots, q_n$ lie in a $(k-1)$-dimensional affine space for $1 \leqslant k \leqslant d$. We introduce here a class of transformation of q, called *affine transformation*, which is determined by a matrix $M \in \mathbb{R}^{d\times d}$ and a vector $b \in \mathbb{R}^d$. Then given a configuration q, an affine image is given by

$$\begin{aligned}\mathcal{A}(q) \triangleq \{p = [p_1, \cdots, p_n] | p_i = Mq_i + b,\\ M \in \mathbb{R}^{d\times d} \text{ and } b \in \mathbb{R}^d, i = 1, \cdots, n\},\end{aligned}$$

or equivalently

$$\mathcal{A}(q) \triangleq \{p = (I_n \otimes M)q + \mathbf{1}_n \otimes b | M \in \mathbb{R}^{d\times d} \text{ and } b \in \mathbb{R}^d\}.$$

A graph $\mathcal{G}$ together with its configuration q in $\mathbb{R}^d$ is called a *framework*, denoted by $(\mathcal{G}, q)$. The edges of the underlying graph $\mathcal{G}$ in $(\mathcal{G}, q)$ are called *members*. If a member has fixed length constraint, we call this a *bar*. A framework is said to be a *bar framework* if all its members are bars. In the rest of this section, we will use 'framework' for 'bar framework' unless otherwise stated.

Graph rigidity theory is for identifying whether partial edge lengths can determine the graph shape uniquely up to translations, rotations, and reflections. Some basic concepts are given as follows.

Given a framework $(\mathcal{G}, q)$ in $\mathbb{R}^d$, if there exists another framework $(\mathcal{G}, p)$ in $\mathbb{R}^d$ such that $\|p_i - p_j\| = \|q_i - q_j\|, \forall (i,j) \in \mathcal{E}$, then we say that $(\mathcal{G}, p)$ is *equivalent* to $(\mathcal{G}, q)$. Furthermore, they are *congruent* if $\|p_i - p_j\| = \|q_i - q_j\|, \forall i, j \in \mathcal{V}$. With these concepts, we say that a framework $(\mathcal{G}, q)$ in $\mathbb{R}^d$ is

- *(locally) rigid,* if all frameworks $(\mathcal{G}, p)$ in $\mathbb{R}^d$ equivalent to $(\mathcal{G}, q)$ and sufficiently close to $(\mathcal{G}, q)$ are congruent to $(\mathcal{G}, q)$;
- *globally rigid*, if all frameworks $(\mathcal{G}, p)$ in $\mathbb{R}^d$ equivalent to $(\mathcal{G}, q)$ are congruent to $(\mathcal{G}, q)$;
- *universally rigid*, if all frameworks $(\mathcal{G}, p)$ in any $\mathbb{R}^D \supset \mathbb{R}^d$ equivalent to $(\mathcal{G}, q)$ are congruent to $(\mathcal{G}, q)$.

In addition to the intuitive geometric definitions of rigidity, we also use *rigidity matrix* to justify the rigidity property of a framework by checking its rank. Before introducing the definition, we need the *distance function* given by

$$f_{\mathcal{G}}(q_1, \cdots, q_n) = \frac{1}{2}[\cdots, \|q_i - q_j\|^2, \cdots]^\top,$$

where $(i,j) \in \mathcal{E}$.

Definition 2.3. *[9] A framework $(\mathcal{G}, q)$ is rigid in $\mathbb{R}^d$ if there exist a neighborhood $\mathcal{P}$ of q such that $f_{\mathcal{G}}^{-1}(f_{\mathcal{G}}(q)) \cap \mathcal{P} = f_{\mathcal{K}}^{-1}(f_{\mathcal{K}}(q)) \cap \mathcal{P}$, where $\mathcal{K}$ is the complete graph with the same vertex set $\mathcal{V}$ of $\mathcal{G}$.*

For a rigid framework, it means if one node moves, the rest also moves as a whole in order to satisfy the distance constraints. One illustrative example is shown in Fig. 2.1. The rectangle framework presented in Fig. 2.1(a) is not rigid since the top two nodes can smoothly move in the horizontal direction without breaking other distance constraints. This framework becomes rigid if we insert a crossing bar between nodes 1 and 3 (or 2 and 4) shown in Fig. 2.1(b).

To characterize the rigidity of a framework, another useful tool is the *rigidity matrix* $R(q) \in \mathbb{R}^{|\mathcal{E}| \times nd}$, which is defined by

$$R(q) = \frac{\partial f_{\mathcal{G}}(q)}{\partial q}. \tag{2.2}$$

As an example, if we assign the orders of members in Fig. 2.1(b) shown as the

circled numbers, the rigidity matrix of the framework can be written as

$$R(q) = \begin{bmatrix} (q_1 - q_2)^\top & (q_2 - q_1)^\top & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (q_2 - q_3)^\top & (q_3 - q_2)^\top & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & (q_3 - q_4)^\top & (q_4 - q_3)^\top \\ (q_1 - q_4)^\top & \mathbf{0} & \mathbf{0} & (q_4 - q_1)^\top \\ (q_1 - q_3)^\top & \mathbf{0} & (q_3 - q_1)^\top & \mathbf{0} \end{bmatrix}.$$

Before moving forward, we introduce some concepts related to *infinitesimal rigidity*. Given a framework $(\mathcal{G}, q)$, $\dot{q} = [\dot{q}_1, \cdots, \dot{q}_n]$ is called an *infinitesimal motion* if for each edge (i, j), there holds

$$(q_i - q_j)^T(\dot{q}_i - \dot{q}_j) = 0.$$

It is easy to check that rotations, translations, and their combinations always satisfy the above equation. These motions are said to be *trivial*. Then we say that a framework is *infinitesimally rigid* if the infinitesimal motions are trivial. This can also be validated through the following lemma.

Lemma 2.4. *[55] A framework $(\mathcal{G}, q)$ is infinitesimally rigid in a d-dimensional space if*

$$rank(R(q)) = nd - d(d+1)/2.$$

In general, infinitesimal rigidity implies rigidity, but the converse is not true. Infinitesimal rigidity only allows the motions as combinations of translation and rotation. It has been discussed that the framework in Fig. 2.1(a) is not rigid, and not infinitesimally rigid. One set of non-trivial infinitesimal motions is presented in Fig. 2.1(c). The framework in Fig. 2.1(b) is not only rigid, but also infinitesimally rigid, which can be verified by the fact that $\mathrm{rank}(R) = 5$.

Definition 2.5. *[5] A framework is minimally rigid if it is rigid and no edge can be removed without losing rigidity.*

To be specific, a rigid framework $(\mathcal{G}, q)$ with n vertices in 2D or 3D is minimally rigid, if it has exactly $2n - 3$ or $3n - 6$ edges, respectively. It can be checked that the framework shown in Fig. 2.1(b) is infinitesimally minimally rigid.

2.3 Tensegrity frameworks and rigidity

A *tensegrity framework* $(\mathcal{G}, q)$ is obtained by embedding an undirected graph $\mathcal{G}$ in $\mathbb{R}^d$ and replacing the edges of $\mathcal{G}$ by three types of *members*: *cables*, *struts* and *bars*, where cables and struts can only carry tensions and compressions respectively, while bars can carry both tensions and compressions. For the physical interpretation,

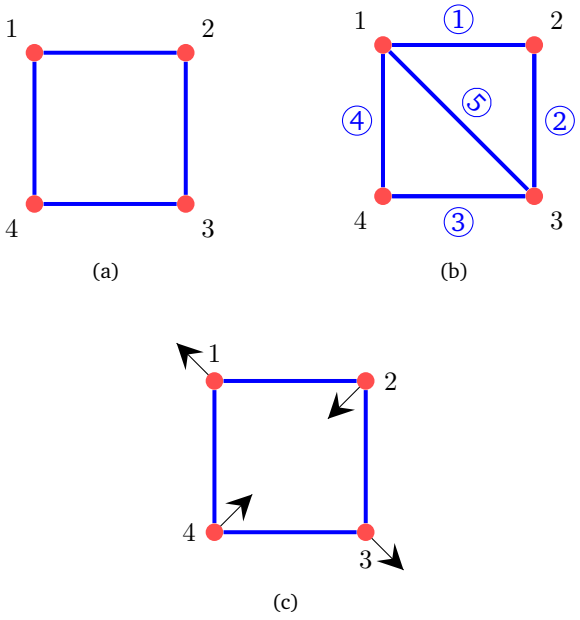


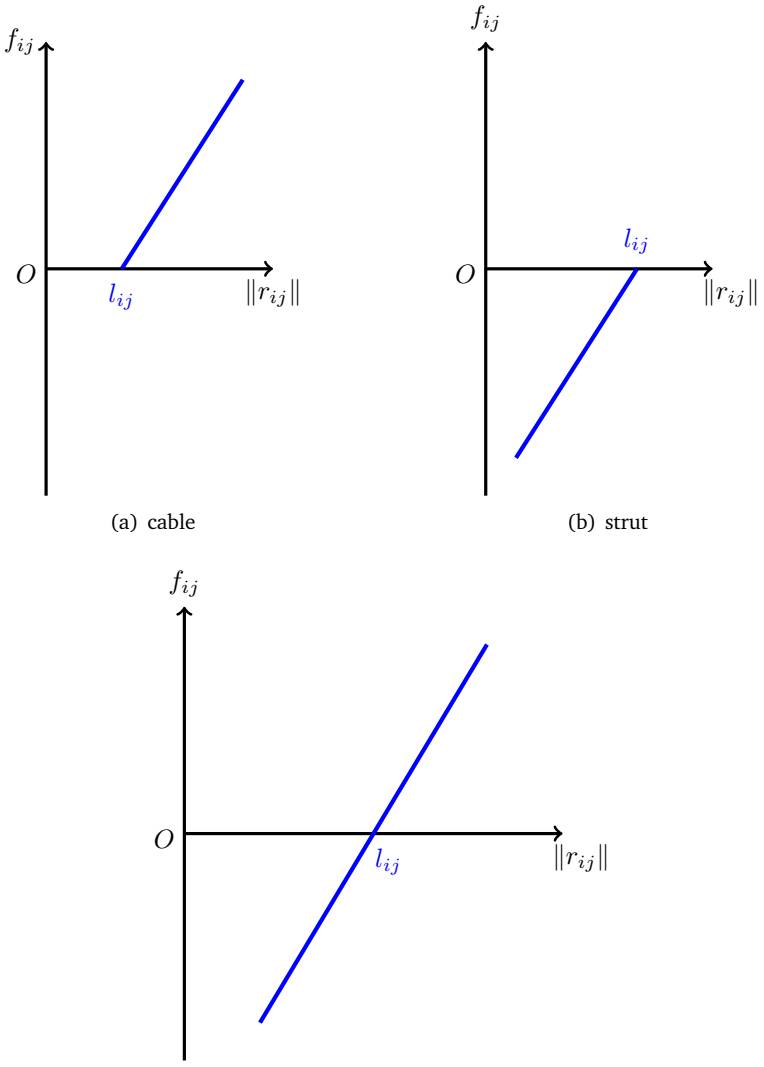
Figure 2.1: *Examples of $2D$ frameworks. (a) non-rigid framework; (b) infinitesimally (and minimally) rigid framework; (c) one set of non-trivial infinitesimal motions.*

we know that the member (i, j) carries different internal forces f_{ij} depending on its rest length l_{ij}, current length $\|r_{ij}\|$ and stiffness k_{ij}. Their relationship can be formulated by

$$f_{ij} = \begin{cases} k_{ij}(\|r_{ij}\| - l_{ij}), \text{ if } (i,j) \text{ is a cable and } \|r_{ij}\| > l_{ij}, \\ -k_{ij}(l_{ij} - \|r_{ij}\|), \text{ if } (i,j) \text{ is a strut and } \|r_{ij}\| < l_{ij}, \\ k_{ij}(\|r_{ij}\| - l_{ij}), \text{ if } (i,j) \text{ is a bar,} \\ \mathbf{0}, \text{ otherwise.} \end{cases} \tag{2.3}$$

As one can see from (2.3), cables and struts sustain positive and negative internal forces, respectively. However, bars can carry both positive and negative forces, which implies that bars can act as springs generating both attractive and repulsive forces. We illustrate these properties graphically in Fig. 2.2.

For a tensegrity framework $(\mathcal{G}, q)$ in $\mathbb{R}^d$ with the fixed configuration q, we are interested in its associated configurations p that satisfy the following *tensegrity*



(a) cable (b) strut

(c) bar

Figure 2.2: *Relationships between internal forces and lengths with respect to different type of members.*

constraints

$$\begin{cases} \|p_i - p_j\| \leqslant \|q_i - q_j\|, & \text{when } (i,j) \text{ is a cable,} \\ \|p_i - p_j\| \geqslant \|q_i - q_j\|, & \text{when } (i,j) \text{ is a strut, and} \\ \|p_i - p_j\| = \|q_i - q_j\|, & \text{when } (i,j) \text{ is a bar.} \end{cases} \tag{2.4}$$

We say that the tensegrity framework $(\mathcal{G}, q)$ whose shape is determined by the configuration q is *rigid* if any other configuration p is congruent to q whenever p is sufficiently close to q and satisfies the tensegrity constraints (2.4); furthermore, if the congruent relationship between p and q holds for all p in $\mathbb{R}^d$, then we say $(\mathcal{G}, q)$ is *globally rigid*; and if this congruent relationship still holds for all p living in any higher-dimensional space than $\mathbb{R}^d$, we say $(\mathcal{G}, q)$ is *universally rigid* [21, 104]. For the rest of this thesis, we only consider cable-strut tensegrity frameworks unless otherwise stated.

To distinguish different members in a tensegrity framework $(\mathcal{G}, q)$, we employ the concept of *stress*. For each member (i, j) of $(\mathcal{G}, q)$, we assign a scalar $\omega_{ij} = \omega_{ji}$, and use $\omega \in \mathbb{R}^{|\mathcal{E}|}$ to denote the vector $\omega = (\cdots, \omega_{ij}, \cdots)^T$. Then ω is called a *stress* of $(\mathcal{G}, q)$; if further, each ω_{ij} satisfies $\omega_{ij} \geqslant 0$ whenever (i, j) is a cable and $\omega_{ij} \leqslant 0$ whenever (i, j) is a strut, then ω is said to be a *proper stress*. Note that for a stress to be *proper*, there is no restriction on a bar. In physics, ω_{ij} is interpreted as the axial force per unit length along the member (i, j). It is called *strict* if the stress in each cable and strut is nonzero. We say that ω is a *self-stress* for the configuration p in $\mathbb{R}^d$ of the framework $(\mathcal{G}, p)$ if for each node i, there holds

$$\sum_{j \in \mathcal{N}_i} \omega_{ij}(q_j - q_i) = \mathbf{0}. \tag{2.5}$$

We also call the stress *an equilibrium stress* if equation (2.5) holds.

Note that for an affine transformation, we have

$$\sum_{j \in \mathcal{N}_i} \omega_{ij}(Mq_i - Mq_j) = M \sum_{j \in \mathcal{N}_i} \omega_{ij}(q_i - q_j) = \mathbf{0}, \quad \forall i,$$

which implies that the affine transformations donot change the equilibrium stress. The corresponding *stress matrix* $\Omega = [\Omega_{ij}] \in \mathbb{R}^{n \times n}$ is defined by

$$\Omega_{ij} = \begin{cases} -\omega_{ij}, & i \neq j, \\ \sum_{j \in \mathcal{N}_i} \omega_{ij}, & i = j. \end{cases} \tag{2.6}$$

For the rigidity of tensegrity frameworks, we introduce the following lemmas.

Lemma 2.6. *[25, Theorem 4.3.1] If a tensegrity $(\mathcal{G}, p)$ is infinitesimally rigid, then it is rigid.*

Lemma 2.7. *[104] Let $(\mathcal{G}, p)$ be a tensegrity framework in $\mathbb{R}^d$, and $(\bar{\mathcal{G}}, p)$ the corresponding bar framework, where all the members of the tensegrity framework have been replaced by bars. Then $(\mathcal{G}, p)$ is infinitesimally rigid (and equivalently statically rigid) if and only if the following two conditions are satisfied:*

1). $(\bar{\mathcal{G}}, p)$ is infinitesimally rigid in $\mathbb{R}^d$, and

2). there is a proper self-stress ω for $(\mathcal{G}, p)$, where for each cable and strut (i, j) of $\mathcal{G}$, $\omega_{ij} \neq 0$.

Lemma 2.8. *[25, Corollary 4.8.2] If a tensegrity $(\mathcal{G}, p)$, with n nodes in $\mathbb{R}^d$, m members and at least one strut or cable, is infinitesimally rigid, then $m \geqslant nd - d(d+1)/2 + 1$.*

To define the universal rigidity for a class of tensegrity frameworks, we present the following lemma.

Lemma 2.9. *[1] Let $(\mathcal{G}, q)$ be a generic tensegrity framework on n vertices in $\mathbb{R}^d$, $d \leqslant n-2$. Then $(\mathcal{G}, q)$ is universally rigid if and only if there exists a positive semi-definite stress matrix Ω such that its rank is $n - d - 1$.*

Next we present conditions to guarantee super-stability of a tensegrity framework. We first introduce some basic concepts.

Definition 2.10. *[23] If ω is a proper equilibrium stress for the tensegrity framework $(\mathcal{G}, q)$, then the relative position $q_i - q_j$ is called a* stressed direction *if $\omega_{ij} \neq 0$.*

Definition 2.11. *[43] We say a function $A : R^m \to R^n$ is* affine *if there is a linear function $L : R^m \to R^n$ and a vector $b \in R^n$ such that*

$$A(x) = L(x) + b \tag{2.7}$$

for all x in $\mathbb{R}^m$.

Definition 2.12. *[24] A* flex *of a framework $(\mathcal{G}, q)$ is a continuous motion $q(s), 0 \leqslant s \leqslant 1$, $q(0) = q$, where $q(s)$ is equivalent to q. It is nontrivial if $q(s)$ is not congruent to q for all $s > 0$. If $q(s) = A(s)p(0)$, where $A(s)$ is an affine function of Euclidean space, then we say $q(s)$ is an* affine flex*.*

Definition 2.13. *Given a collection of vectors $V = \{v_i\}_{i \in \mathbb{N}}$,*[1] affine span *of V is defined as the collection of all finite linear combinations, i.e.,*

$$\sum_{1}^{k} a_i v_{s_i}, \quad k \in \mathbb{N}^+,^2\, v_{s_i} \in V,$$

where a_i are all scalars satisfying $\sum a_i = 1$.

Lemma 2.14. *[25] Let $(\mathcal{G}, q)$ be a tensegrity framework whose affine span of q is $\mathbb{R}^d$, with an equilibrium stress ω and stress matrix Ω. Suppose further that*

1. *Ω is positive semi-definite,*

[1] $\mathbb{N}$ denotes the set of natural numbers.
[2] $\mathbb{N}^+$ denotes the set of positive natural numbers.

2. *the rank of* Ω *is* $n - d - 1$*,*

3. *and* $(\mathcal{G}, q)$ *has no affine flex in* $\mathbb{R}^d$*,*

then $(\mathcal{G}, q)$ *is super stable.*

Remark 2.15. Lemma 2.14 is known as the fundamental theorem for super-stability. When ω is a proper equilibrium stress for $(\mathcal{G}, q)$, a *stressed direction* is the relative position of two connected nodes i and j with $\omega_{ij} \neq 0$, i.e., $q_i - q_j$. Note that condition (3) of Lemma 2.14 can be replaced by "the framework $(\mathcal{G}, q)$ is rigid in $\mathbb{R}^d$"[23], and "the configuration q is in general position" [24]. We also have the conclusion that super stable tensegrity frameworks are universally rigid, but not vice versa.

Another lemma describing translation movements of a generic configuration is introduced as follows.

Lemma 2.16. *[78] Suppose* $q = [q_1^T, \cdots, q_n^T]^T$ *is a generic configuration in* $\mathbb{R}^d$*. A configuration* $p \in \mathcal{A}(q)$ *is a translation of* q *if and only if there exist at least* d *pairs of vertices such that the dimension of the convex hull of that* d *pairs of vertices is* d *and*

$$p_k - p_j = q_k - q_j.$$

The following lemma will be used in the sequel in different places for the discussion of positive semi-definite stress matrices.

Lemma 2.17. *Given positive semi-definite matrices* $X \in \mathbb{R}^{n\times n}$ *and* $Y \in \mathbb{R}^{n\times n}$*, let* $Z = X + Y$*. Then for any nonzero vector* $\xi \in \mathbb{R}^n$*,* $\xi \in \mathrm{null}(Z)$ *if and only if* $\xi \in \mathrm{null}(X)$ *and* $\xi \in \mathrm{null}(Y)$*.*

Chapter 3

Merging rigid tensegrity frameworks

This chapter is to analyze the existence of a strictly proper self-stress for the merged tensegrity framework obtained by connecting two separate ones. It discusses what kind of condition allows the combined framework to be in equilibrium with the new stress without altering any existing member's type, viz. cable or strut, in the previously existing frameworks. In addition, this chapter also studies the problem of merging two rigid tensegrity frameworks. It is shown that there exists an insertion of four new members with which the combined tensegrity framework is still rigid. By injecting distance perturbations into the combined framework, we propose a method with which the type of the fourth member can be determined.

3.1 Introduction

Rigidity graph theory serves as a fundamental mathematical tool to solve a wide range of problems in different fields, such as formation control of teams of mobile robots [6, 15, 35, 54, 64, 75, 98], localization of sensor networks [41, 110], molecular structural analysis in bio-chemistry [61, 128] and construction of stable frameworks in [107]. Of particular theoretical and practical interest are tensegrity frameworks, which are able to support large loads because of the use of cables and struts in comparison with bar frameworks. This property has been well employed in the design and control of tensegrity robots, see e.g. [94, 105].

In many, if not most, applications, the framework is expected to be rigid. This means the formation shape of the framework can be maintained as long as the distance constraints associated with all the edges are maintained, i.e. for a bar, an exact distance is maintained, for a cable, an upper bound is maintained, and for a strut, a lower bound is maintained. As stated in Chapter 2, global rigidity and universal rigidity can be accordingly defined if rigidity still holds in the whole given space and any higher dimensional space, respectively.

From an engineering point of view, a framework may be required to be augmented by adding one or more vertices, or even merging or becoming connected with another framework. More precisely, by *merging* we mean, given two frameworks, the operations of superimposing some of their vertices and adding additional

members joining a vertex pair with the vertices drawn from the two different frameworks. Normally, rigidity of frameworks is aimed to be preserved after adding vertices or merging.

In the plane, it is well known that the *Henneberg construction (HC)* [120] is an efficient technique to grow *minimally rigid* graphs. Recall that a rigid graph is said to be *minimally rigid* if no single edge can be removed without losing rigidity. The constructions of [120] propose two techniques, termed *vertex addition* and *edge splitting*. Due originally to Henneberg [58], a minimally rigid framework (in an ambient two or three-dimensional space) can acquire an additional vertex (in the process that additional members are introduced). Henneberg also proposed a merging procedure for two (minimally) rigid graphs in an ambient two-dimensional space, whereby three members (bars in a normal framework) were inserted to link the two frameworks.

However, the HC did not impose any geometric constraints, such as the length and angle, on the newly added edges, which might result in contradictions with the practical use. To solve this problem, a new construction method using Delaunay triangulation is proposed such that the angle measurements and the number of links can be optimized in [39]. As a follow-up, [40] addressed several topics relevant to rigid frameworks, including minimal cover problem, splitting and merging problem, and closing ranks problem. By *minimal cover problem*, we mean finding a new set of edges to be inserted into a given framework, such that the resulting framework is minimally rigid. The merging problem was regarded as one special case of minimal cover problem therein, and a type of new strategy based on reduction procedure was designed. Later, the conditions on how to generate globally rigid frameworks through merging in two- and three- dimensional space are provided, respectively. To fully cover all the possible cases of merging frameworks, where it is permitted to have one or more of the vertices of one merging framework made coincident with the same number of the other framework, three principles to conduct *optimal merging* of minimally or globally rigid frameworks were proposed in [133] for $\mathbb{R}^2$ and $\mathbb{R}^3$ frameworks. The merging is said to be *optimal* if the number of newly added member for a given number of shared vertices is minimized. At the same time, merging of multiple (more than two) rigid frameworks was investigated in [134], where each framework is regarded as a *meta-vertex*, and thus the problem can be solved from the meta*meta-formation* prospective. In this way, the proposed strategies in [133] can be extended to the case involving multiple frameworks. Relying on HC operations, [136] investigated optimal growing of rigid frameworks in the sense of $\mathcal{H}_2$ performance. Motivated by the implications of rigid networks in formation control and localizability, [18] identified the conditions for rigidity-preserving splitting as opposed to merging, under which the corresponding algorithms to perform the partition were also proposed therein. In addition to these work on growing rigid frameworks with undirected underlying graphs, some

efforts were also made based on directed graphs [53, 57]. In parallel to rigidity of undirected graphs, *persistence* was introduced for directed graphs in [57], where the conditions to ensure persistence after merging was given in both two- and three-dimensional space. Recently in [53], the dynamic merging problem under switching topology has been solved under the condition that each follower is jointly reachable from a leader over any time interval of certain length.

Even though some strategies have been proposed for growing rigid bar frameworks, to the best of our knowledge, the systematic analysis on how to create rigid tensegrity frameworks by merging has been rarely reported due to the complexity caused by the inequality constraints (2.4). One relevant work was presented in [119], where it has been shown that *labeled 1-extension* operation on a rigid *tensegrity graph* does not change its rigidity. A *Tensegrity graph* $\mathcal{G}(\mathcal{V}, C, S)$ is obtained through replacing the edges of a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ by cables and struts. Denote by $\mathcal{C}$ and $\mathcal{S}$ the cable and strut set, respectively. The *labeled 1-extension* is defined in such a way that the original member (u, w) is removed and a new vertex v is added together with three new members $(v, u), (u, w)$ and (v, t) under the constraint that the type of (u, w) is the same as at least one of (v, u) and (v, w).

Motivated by this circumstance, in this chapter, we will first prove that by merging two isolated infinitesimally rigid tensegrity framework with four members, there exist a proper self-stress such that the resulting tensegrity framework is still infinitesimally rigid and the type of the members can be preserved. We then explore the existence of the assignment of cables and struts to the four linking members when merging two rigid tensegrity frameworks, under which the combined tensegrity framework is rigid. In this chapter, all of the results are constrained to $\mathbb{R}^2$ unless otherwise indicated.

The rest of this chapter is organized as follows. In Section 3.2, we prove that the infinitesimal rigidity can be preserved by linking two originally infinitesimally rigid tensegrity frameworks with four appropriate members. In addition to infinitesimal rigid tensegrity frameworks, we also considered the problem of merging rigid tensegrity frameworks in Section 3.3. Concluding remarks are given in Section 3.4.

3.2 Merging infinitesimally rigid tensegrity frameworks

In this section, we investigate whether infinitesimal rigidity can be maintained after merging two pre-existing infinitesimally rigid tensegrity frameworks by inserting four members of the different type.

To be specific, the two given planar separate tensegrity frameworks $\mathcal{T}_A$ and $\mathcal{T}_B$, shown in Fig. 3.1, are infinitesimally rigid with underlying graphs $\mathcal{G}_A(\mathcal{V}_A, \mathcal{E}_A)$ and $\mathcal{G}_B(\mathcal{V}_B, \mathcal{E}_B)$, respectively. It is assumed that the underlying graphs $\mathcal{G}_A$ and $\mathcal{G}_B$ respectively consist of n_A and n_B nodes, which are linked via m_A and m_B members,

i.e., $|\mathcal{V}_A| = n_A$, $|\mathcal{V}_B| = n_B$, $|\mathcal{E}_A| = m_A$, and $|\mathcal{E}_B| = m_B$. For infinitesimally rigid tensegrity frameworks in $\mathbb{R}^2$, it follows from Lemma 2.8 that

$$m_A \geqslant 2n_A - 2 \quad \text{and} \quad m_B \geqslant 2n_B - 2. \tag{3.1}$$

Note that to obtain a rigid tensegrity framework from interconnecting two separate rigid tensegrity frameworks, it is necessary to have at least four connecting members. In addition, the absolute position (centroid location and orientation) of the frameworks $\mathcal{T}_A$ and $\mathcal{T}_B$ is irrelevant (though after insertion of the connections some freedom is of course lost).

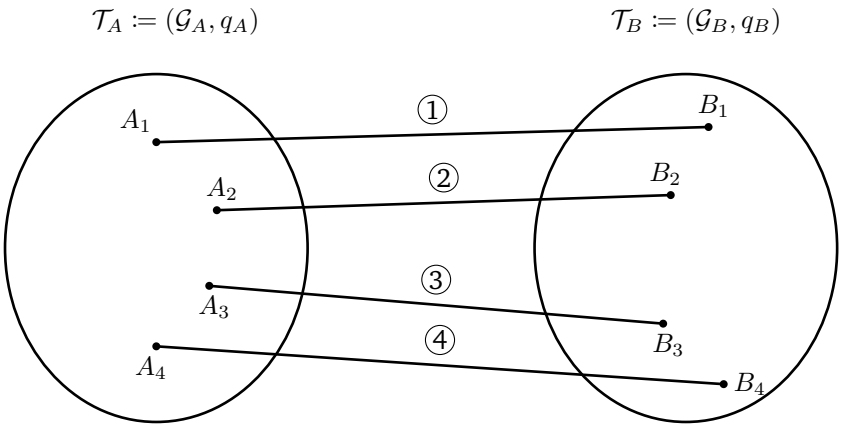


Figure 3.1: *Growing rigid tensegrity framework by inserting four new members in ambient space* $\mathbb{R}^2$.

In the sequel, denote by $\omega_i^A, i = 1, \cdots, m_A$, the original stress of member i in $\mathcal{T}_A$, and denote by $\hat{\omega}_i^A, i = 1, \cdots, n_A$, the new stress after inserting the four members $(A_i, B_i), i = 1, \cdots, 4$. Analogously, we can define the stress ω_i^B and $\hat{\omega}_i^B, i = 1, \cdots, m_B$ associated with tensegrity framework $\mathcal{T}_B$. Those four members are labeled as member $\textcircled{i}, i = 1, \cdots, 4$, shown in Fig. 3.1. All the points of A_i are distinct and so are all the points of B_i. It is also assumed that at least three of the members, without loss of generality $(A_i, B_i), i = 1, 2, 3$, are not concurrent or parallel. If all four members are concurrent or parallel, it is not hard to see that there is an infinitesimal displacement at right angles to each of the A_i exists for which the corresponding infinitesimal length changes are zero, i.e. infinitesimal rigidity cannot hold.

Theorem 3.1. *Consider two infinitesimally rigid tensegrity frameworks $\mathcal{T}_A$ and $\mathcal{T}_B$ in $\mathbb{R}^2$ and assume they are connected by 4 new members that are not concurrent or parallel. Then there exist an assignment of cables and struts to the new members such that the combined tensegrity framework is infinitesimally rigid as well. Furthermore,*

the sign of the new proper self-stress associated with pre-existing members can be maintained, namely, the members' type, viz. cable or strut, can be preserved in both $\mathcal{T}_A$ and $\mathcal{T}_B$.

Proof. We first prove that there exists a new strictly proper self-stress such that the signs of the stress associated with the previously existing members do not change. Then by invoking Lemma 2.7, the infinitesimal rigidity of the combined tensegrity framework can be ensured.

For an infinitesimally rigid tensegrity framework $\mathcal{T}_A$, the self-stress satisfies

$$R_A^\top(q)\omega^A = \mathbf{0}, \tag{3.2}$$

where $R_A(q) \in \mathbb{R}^{m_A \times 2n_A}$ is the rigidity matrix of $\mathcal{T}_A$, and $\omega^A = [\omega_1^A, \cdots, \omega_{m_A}^A]^\top \in \mathbb{R}^{m_A}$ is the stress with respect to $\mathcal{T}_A$. With the four members $(A_i, !\text{'}!\text{`}B_i), i = 1, \cdots, 4$, joining the two separate tensegirty frameworks, the combined self-stress satisfies

$$R_A^T(q)\hat{\omega}^A + r_A^T w^N = \mathbf{0}, \tag{3.3}$$

where ω^N is the stress associated with new edges in the form of $\omega^N = [\omega_1, \cdots, \omega_4]^T \in \mathbb{R}^4$. The matrix $r_A \in \mathbb{R}^{4\times 2n_A}$ is defined by

$$\begin{aligned} r_A &= \left[r_{A1}^\top, r_{A2}^\top, r_{A3}^\top, r_{A4}^\top\right]^\top \\ &= \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & (q_{A1}-q_{B1})^\top & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & (q_{A2}-q_{B2})^\top & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & (q_{A3}-q_{B3})^\top & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & (q_{A4}-q_{B4})^\top \end{bmatrix}. \end{aligned} \tag{3.4}$$

Suppose the types of pre-existing members are maintained after interconnection. A sufficient condition meeting this requirement is given by

$$\hat{\omega}^A = \omega^A + \Psi^A \operatorname{sign}(\omega^A), \tag{3.5}$$

where $\Psi^A = \operatorname{diag}(\psi_1^A, \cdots, \psi_{m_A}^A) \in \mathbb{R}^{m_A \times m_A}$ is a diagonal matrix with $\psi_i^A > 0, i = 1, \cdots, m_A$, an arbitrary scalar. $\operatorname{sign}(\omega^A) = [\operatorname{sign}(\omega_1^A), \cdots, \operatorname{sign}(\omega_{m_A}^A)]^\top \in \mathbb{R}^{m_A}$. The setting $\hat{\omega}^A$ as in (3.5) ensures that $\operatorname{sign}(\hat{\omega}^A) = \operatorname{sign}(\omega^A)$. Note that (3.5) can be equivalently written as

$$\hat{\omega}_i^A = \begin{cases} \omega_i^A + \psi_i^A, \text{ if } \omega_i^A > 0, \\ \omega_i^A - \psi_i^A, \text{ if } \omega_i^A < 0. \end{cases} \tag{3.6}$$

Substituting (3.5) into (3.3), we get

$$R_A^\top(q)\left(\omega^A + \Psi^A \operatorname{sign}(\omega^A)\right) + r_A^\top w^N = \mathbf{0}. \tag{3.7}$$

In light of (3.2), we have

$$R_A^\top(q)\left(\Psi^A \operatorname{sign}(\omega^A)\right) + r_A^\top w^N = \mathbf{0}. \tag{3.8}$$

Analogously, for tensegrity framework $\mathcal{T}_B$, we also have

$$R_B^\top(q)\left(\Psi^B \operatorname{sign}(\omega^B)\right) + r_B^\top w^N = \mathbf{0}, \tag{3.9}$$

where variables $R_B^T(q), \Psi^B, \omega^B$ and r_B are defined in the same way as the corresponding variables associated with $\mathcal{T}_A$. Denote by $\psi^A \in \mathbb{R}^{m_A}$ the vector consisting of the diagonal entries of Ψ^A, i.e., $\psi^A = [\psi_1^A, \cdots, \psi_{m_A}^A]^T$. Note that for diagonal matrices Ψ^A, there holds

$$\Psi^A \operatorname{sign}(\omega^A) = \operatorname{diag}(\operatorname{sign}(\omega^A))\psi^A, \tag{3.10}$$

and

$$\Psi^B \operatorname{sign}(\omega^B) = \operatorname{diag}(\operatorname{sign}(\omega^B))\psi^B. \tag{3.11}$$

Then, in view of (3.8)-(3.11), we have

$$\begin{bmatrix} R_A^\top(q)\operatorname{diag}(\operatorname{sign}(\omega^A)) & 0 \\ 0 & R_B^\top(q)\operatorname{diag}(\operatorname{sign}(\omega^B)) \end{bmatrix} \begin{bmatrix} \psi^A \\ \psi^B \end{bmatrix} + r^\top \omega^N = \mathbf{0}, \tag{3.12}$$

where $r = [r_A, r_B] \in \mathbb{R}^{4\times 2(n_A+n_B)}$. Summarizing, to show the existence of a strictly proper self-stress associated with the combined tensegrity framework is equivalent to showing that the linear algebraic equation (3.12) has nontrivial solutions to the unknowns ψ^A, ψ^B and ω^N subject to $\psi^A > 0$ and $\psi^B > 0$. [Here by denoting $x > 0, x \in \mathbb{R}^n$, we mean that all of its entries are positive.]

This equation can be further written as

$$\left[\begin{pmatrix} R_A^\top(q)\operatorname{diag}(\operatorname{sign}(\omega^A)) & 0 \\ 0 & R_B^\top(q)\operatorname{diag}(\operatorname{sign}(\omega^B)) \end{pmatrix}, r^\top\right] \begin{bmatrix} \psi^A \\ \psi^B \\ \omega^N \end{bmatrix} = \mathbf{0}. \tag{3.13}$$

Note that the rigidity matrix of the combined tensegrity framework is in the form of

$$R(q) = \begin{bmatrix} R_A(q) & 0 \\ 0 & R_B(q) \\ & r \end{bmatrix}. \tag{3.14}$$

Therefore, (3.13) can be transformed to

$$R^{\top}(q)\underbrace{\begin{bmatrix} \mathrm{diag}(\mathrm{sign}(\omega^A)) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathrm{diag}(\mathrm{sign}(\omega^B)) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_4 \end{bmatrix}}_{:=D}\begin{bmatrix} \psi^A \\ \psi^B \\ \omega^N \end{bmatrix} = \mathbf{0}. \tag{3.15}$$

From the definition of $R(q)$ in (3.14), it is easy to see that the dimension of $R(q)$ is $(m_A + m_B + 4) \times 2(n_A + n_B)$, and thus $R^{\top}(q) \in \mathbb{R}^{2(n_A+n_B)\times(m_A+m_B+4)}$. This rigidity matrix $R(q)$ corresponds to the underlying bar framework as well. Given two infinitesimally rigid bar frameworks in the plane, it has been established by Henneberg [58] that if one joins the two frameworks by bars $(A_i, B_i), i = 1, 2, 3$, that are neither concurrent (when prolonged if necessary) nor parallel, then the framework with the three bars inserted is also infinitesimally rigid. Therefore, the combined new bar framework is infinitesimally rigid. Thus, there holds

$$\mathrm{rank}(R(q)) = \mathrm{rank}(R^{\top}(q)) = \mathrm{rank}(R^{\top}(q)D) = 2(n_A + n_B) - 3, \tag{3.16}$$

which implies

$$\begin{aligned} \dim\left(\mathrm{null}(R^{\top}(q))\right) &= (m_A + m_B + 4) - 2(n_A + n_B) + 3 \\ &\geqslant (2n_A - 2 + 2n_B - 2 + 4) - 2(n_A + n_B) + 3 \\ &= 3 \end{aligned} \tag{3.17}$$

Hence there always exist nontrivial solutions with respect to $[\psi^A, \psi^B, \omega^N]^{\top}$ of equation (3.15), disregarding for the moment sign constraints. We now choose one set of specified solutions with nonzero entries as η^A, η^B and η^N, satisfying

$$R^{\top}(q)\begin{bmatrix} \mathrm{diag}(\mathrm{sign}(\omega^A)) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathrm{diag}(\mathrm{sign}(\omega^B)) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_4 \end{bmatrix}\begin{bmatrix} \eta^A \\ \eta^B \\ \eta^N \end{bmatrix} = \mathbf{0}, \tag{3.18}$$

where one or more entries of η^A and η^B might be negative. Hence to show the existence of a strictly proper self-stress, we still need to find the positive solutions with respect to ψ^A and ψ^B satisfying (3.15).

Before moving on, note that (3.18) can be equivalently rewritten as

$$\begin{bmatrix} R_A^{\top}(q)\mathrm{diag}(\mathrm{sign}(\omega^A)) & 0 \\ 0 & R_B^{\top}(q)\mathrm{diag}(\mathrm{sign}(\omega^B)) \end{bmatrix}\begin{bmatrix} \eta^A \\ \eta^B \end{bmatrix} + r^{\top}\eta^N = \mathbf{0}. \tag{3.19}$$

Now, we consider the homogeneous part of (3.12), i.e.,

$$\begin{bmatrix} R_A^\top(q) & \mathbf{0} \\ \mathbf{0} & R_B^\top(q) \end{bmatrix} \begin{bmatrix} \mathrm{diag}(\mathrm{sign}(\omega^A)) & 0 \\ 0 & \mathrm{diag}(\mathrm{sign}(\omega^B)) \end{bmatrix} \begin{bmatrix} \psi^A \\ \psi^B \end{bmatrix} = \mathbf{0}. \quad (3.20)$$

Then any solution of (3.20), denoted by ξ^A and ξ^B, satisfies

$$\begin{bmatrix} R_A^\top(q)\mathrm{diag}(\mathrm{sign}(\omega^A)) & 0 \\ 0 & R_B^\top(q)\mathrm{diag}(\mathrm{sign}(\omega^B)) \end{bmatrix} \begin{bmatrix} \xi^A + \eta^A \\ \xi^B + \eta^B \end{bmatrix} + r^\top \eta^N = \mathbf{0}. \quad (3.21)$$

Therefore, $\left[\xi^A + \eta^A, \xi^B + \eta^B, \eta^N\right]^\top$ can be regarded as one set of solutions to the unknowns $\left[\psi^A, \psi^B, \omega^N\right]^\top$ in (3.12). Here note that ξ^A and ξ^B are general solutions to the homogeneous linear algebraic equation (3.20) rather then specific vectors. However, to satisfy the constraints that $\psi^A > 0$ and $\psi^B > 0$ in (3.12) under the chosen $\omega^N = \eta^N$, we need to find at least one set of solutions among ξ^A and ξ^B, denoted by ζ^A and ζ^B, such that $\zeta^A + \eta^A > 0$ and $\zeta^B + \eta^B > 0$.

From (3.2), it is evident that

$$\begin{cases} \mathrm{span}(\omega^A) \subseteq \mathrm{null}(R_A^\top(q)), \\ \mathrm{span}(\omega^B) \subseteq \mathrm{null}(R_B^\top(q)). \end{cases} \quad (3.22)$$

Note that the dimension of $R_A^\top(q)$ is $2n_A \times m_A$, and $\mathrm{rank}\left(R_A^\top(q)\right) = 2n_A - 3$ for a infinitesimally rigid framework $\mathcal{T}_A$. We denote by $\mathrm{col}(X)$ and $\mathrm{null}(X)$ the column space and the null space of a matrix X. Since

$$\dim\left(\mathrm{col}\left(R_A^\top(q)\right)\right) + \dim\left(\mathrm{null}\left(R_A^\top(q)\right)\right) = m_A,$$

and

$$\mathrm{rank}\left(R_A^\top(q)\right) = \dim\left(\mathrm{col}\left(R_A^\top(q)\right)\right),$$

we have

$$\dim\left(\mathrm{null}\left(R_A^\top(q)\right)\right) = m_A - 2n_A + 3. \quad (3.23)$$

In view of (3.1), i.e., $m_A \geqslant 2n_A - 2$, (3.23) satisfies

$$\dim\left(\mathrm{null}\left(R_A^\top(q)\right)\right) \geqslant 1. \quad (3.24)$$

By taking (3.22) into account, (3.24) implies that though $\mathrm{span}(\omega^A)$ lies in the null space of $R_A^\top(q)$, it might not fully span the null space. Only when the equality sign holds in (3.24), $\mathrm{span}(\omega^A) = \mathrm{null}\left(R_A^\top(q)\right)$. For framework $\mathcal{T}_B$, we have the similar

results. So the specific solution ζ^A and ζ^B to (3.20) can be set to satisfy

$$\begin{cases} \mathrm{diag}(\mathrm{sign}(\omega^A))\zeta^A \in \mathrm{span}(\omega^A), \\ \mathrm{diag}(\mathrm{sign}(\omega^B))\zeta^B \in \mathrm{span}(\omega^B). \end{cases} \tag{3.25}$$

To be specific, ζ^A and ζ^B can be chosen as

$$\zeta^A = k^A|\omega^A| \quad \text{and} \quad \zeta^B = k^B|\omega^B|, \tag{3.26}$$

where k^A and k^B are positive scalars. This equation can be written in the componentwise form as

$$\begin{bmatrix} \zeta_1^A \\ \vdots \\ \zeta_{m_A}^A \end{bmatrix} = k^A \begin{bmatrix} |\omega_1^A| \\ \vdots \\ |\omega_{m_A}^A| \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \zeta_1^B \\ \vdots \\ \zeta_{m_B}^B \end{bmatrix} = k^B \begin{bmatrix} |\omega_1^B| \\ \vdots \\ |\omega_{m_B}^B| \end{bmatrix}. \tag{3.27}$$

Recall the requirement that $\zeta^A + \eta^A > 0$, which is equivalent to

$$\begin{cases} \zeta_1^A + \eta_1^A > 0, \\ \quad \vdots \\ \zeta_{m_A}^A + \eta_{m_A}^A > 0. \end{cases} \tag{3.28}$$

Substituting (3.27) into (3.28), it yields

$$\begin{cases} k^A|\omega_1^A| + \eta_1^A > 0, \\ \quad \vdots \\ k^A|\omega_{m_A}^A| + \eta_{m_A}^A > 0. \end{cases} \tag{3.29}$$

Then, the parameter k^A can be derived from (3.29) that

$$k^A > \max\left\{ -\frac{\eta_1^A}{|\omega_1^A|}, \cdots, -\frac{\eta_{m_A}^A}{|\omega_{m_A}^A|} \right\}. \tag{3.30}$$

Analogously, in terms of parameter k^B, we can also have

$$k^B > \max\left\{ -\frac{\eta_1^B}{|\omega_1^B|}, \cdots, -\frac{\eta_{m_B}^B}{|\omega_{m_B}^B|} \right\}. \tag{3.31}$$

Therefore, the existence of a strictly proper self-stress associated with the combined framework is proved. In addition, the type of pre-existing members can be maintained. Then it follows from Lemma 2.7 that the overall tensegrity frame-

work after interconnection is also infinitesimally rigid. This completes the proof of Theorem 3.1.

□

3.3 Merging rigid tensegrity frameworks [1]

In this section we explain how to choose the type, viz. strut or cable, of the four elements in a tensegrity framework generated by joining two separate tensegrity frameworks.

More precisely, we shall consider two tensegrity frameworks, call them $\mathcal{T}_A$ and $\mathcal{T}_B$, with four identified points in each framework, viz, $A_1, \cdots, A_4, B_1, \cdots, B_4$. We assume that these are in general position and that connections are made, with either a strut or a cable, between each A_i and the corresponding B_i, $i = 1, \cdots, 4$.

3.3.1 General approach to the problem

We first assume that the two separate tensegrity frameworks are infinitesimally rigid. In addition, there are (for the moment) *bars* between three points A_1, A_2, A_3 say of framework $\mathcal{T}_A$ and the corresponding three points B_1, B_2, B_3 of framework $\mathcal{T}_B$. We can measure the distance between A_4 and B_4 but there is no bar.

It was established by Henneberg [58] that if one extends the lines joining $A_i, B_i, i = 1, 2, 3$ and they are not concurrent or parallel, then the single framework with the three bars inserted to join the separate frameworks $\mathcal{T}_A$ and $\mathcal{T}_B$ is also infinitesimally rigid (the converse also holds).

This means that there is a finite number of noncongruent frameworks (defined up to inessential translation, reflection and rotation) realizing the associated set of lengths, with the frameworks $\mathcal{T}_A$ and $\mathcal{T}_B$ being held invariant apart from possible translation, reflection or rotation. Consequently, there is a finite number of possibility for the squared distance $\bar{d}_4 := ||q_{A4} - q_{B4}||^2$, one value of squared distance being associated with each of the noncongruent frameworks. (Of course, in some special cases, the distance for two noncongruent frameworks might coincide, but in general this cannot be expected).

Now if the squared distances $\bar{d}_i := ||q_{Ai} - q_{Bi}||^2$ for $i = 1, 2, 3$ are varied by an infinitesimally small amount to $d_i = \bar{d}_i + \delta d_i$, by for example holding framework A fixed and rotating and/or translating framework B infinitesimally, then there will be necessarily be a corresponding infinitesimal change replacing $\bar{d}_4$ by $d_4 = \bar{d}_4 + \delta d_4$.

It is evident that having fixed a particular framework from the finite set realizing the joining of $\mathcal{T}_A$ and $\mathcal{T}_B$ using bars of squared lengths $\bar{d}_1, \bar{d}_2, \bar{d}_3$ we can find a

[1] The majority of this section is taken from one of Prof. Brian D. O. Anderson's unpublished technical reports. For completeness of the strategy for growing locally rigid tensegrity frameworks, we put it in this thesis.

smooth function $f : \mathbb{R}^3 \to \mathbb{R}$ for which

$$f(d_1, d_2, d_3) = d_4 \tag{3.32}$$

with in particular

$$f(\bar{d}_1, \bar{d}_2, \bar{d}_3) = \bar{d}_4. \tag{3.33}$$

Moreover, to first order,

$$\delta d_4 = \frac{\partial f}{\partial d_1}\delta d_1 + \frac{\partial f}{\partial d_2}\delta d_2 + \frac{\partial f}{\partial d_3}\delta d_3. \tag{3.34}$$

It is assumed that for generic $\bar{d}_1, \bar{d}_2, \bar{d}_3$, the function f will not have a critical point, i.e. there will not hold $\frac{\partial f}{\partial d_i} = 0, i = 1, 2, 3$ at $\bar{d}_1, \bar{d}_2, \bar{d}_3$. Now we make a change of viewpoint. We suppose that the three bars linking A_i to B_i for $i = 1, 2, 3$ are replaced by either a strut or a cable, *and* a strut or cable is placed between A_4 and B_4. We claim the following theorem.

Theorem 3.2. *Consider the arrangement described above. There exists a choice of strut or cable for each of the four linkages between q_{Ai} and q_{Bi} to ensure rigidity.*

Proof. When a desired distance is $\bar{d}_i$, a strut of this length allows a positive value of δd_i and a chain a negative value. For $i = 1, 2, 3$, choose link i to be a strut or a cable according as $\frac{\partial f}{\partial d_i}$ is positive or negative. For $i = 4$, choose a cable.

Observe that the choices for $i = 1, 2, 3$ ensure that any allowed changes of length in three links will always cause the right side in (3.34) to be positive. The choice for $i = 4$ however means that any allowed change in d_4 must be negative. This means there is no set of nonzero changes δd_i consistent with the assignation of struts and cables. Equivalently, the framework is rigid. This proves the theorem. □

Obviously, the same conclusion follows if we reverse the assignment of cables and struts in the proof.

While the above theorem asserts the existence of a mixture of struts and cables assuring rigidity, the actual determination of the link type appears to involve knowledge of the functions f_i, or their derivatives. The functions themselves in general may be very difficult to find. The derivatives on the other hand are easier to find, and we now show how this can be done using the rigidity matrix.

3.3.2 Determining δd_4 using the rigidity matrix

Suppose that the frameworks $\mathcal{T}_A$ and $\mathcal{T}_B$ referred to above are described by stacked vectors of vertex positions q_A, q_B and stacked vectors of squared edge lengths realized in the frameworks using tensegrity elements given by $\bar{d}_A, \bar{d}_B$. In obvious

notation, the rigidity matrix of the overall framework with four tensegrity elements connecting the two frameworks is given by

$$R = \begin{bmatrix} R_A & 0 \\ 0 & R_B \\ r_1^\top & \\ r_2^\top & \\ r_3^\top & \\ r_4^\top & \end{bmatrix}. \tag{3.35}$$

Now let us consider infinitesimal perturbations of the squared distances $\bar{d}_1, \bar{d}_2, \bar{d}_3$ by amounts $\delta d_1, \delta d_2$ and δd_3. There will be a consequential perturbation in d_4 of δd_4, and consequential perturbations $\delta q_A, \delta q_B$ of the vertex positions q_A, q_B. The fact that R is the Jacobian of the mapping from vertex positions to squared distances means we can write

$$\begin{bmatrix} 0 \\ \vdots \\ 0 \\ \delta d_1 \\ \delta d_2 \\ \delta d_e \\ \delta d_4 \end{bmatrix} = R \begin{bmatrix} \delta q_A \\ \delta q_B \end{bmatrix}. \tag{3.36}$$

Note that for given $\delta d_1, \delta d_2$ and δd_3, the perturbations $\delta q_A, \delta q_B$ are not unique; the adjustment of any vertex perturbation vector by the same infinitesimal translation and/or rotation will leave the right side of the above equation invariant.

In pursuit of our ultimate goal of explaining how the derivatives $\frac{\partial d_4}{\partial \bar{d}_i}, i = 1, 2, 3$ can be computed using the rigidity matrix, it is convenient to eliminate this nonuniquenessss. This is done as follows. Suppose that a new framework is formed by making four changes:

1. Bars replace all tensegrity elements within framework $\mathcal{T}_A$ and framework $\mathcal{T}_B$.
2. Bars are removed from framework $\mathcal{T}_A$ and framework $\mathcal{T}_B$ to make the frameworks both minimally rigid.
3. The tensegrity elements of squared lengths $\bar{d}_1, \bar{d}_2, \bar{d}_3$ and $\bar{d}_4$ are replaced by bars.
4. One vertex of framework $\mathcal{T}_A$ is pinned at the origin and a neighbor vertex in the framework is pinned on the x axis.

The resulting framework is pinned and is a rigid joint-bar framework. It is not minimally rigid but has the property that removal of the bar corresponding

to length $\bar{d}_4$ would make it minimally rigid. Call the associated rigidity matrix $\hat{R}$; this is a 'reduced' rigidity matrix, of size $(2n-2)\times(2n-3)$ where n is the total vertex count for frameworks $\mathcal{T}_A$ and $\mathcal{T}_B$. The reduced rigidity matrix is obtained from R defined in (3.35) through deletion of rows within the blocks $[R_A\ 0]$ and $[0\ R_B]$, corresponding to step 2 of the reduction procedure and deletion of three columns of R, corresponding to step 4. The rank of $\hat{R}$ is the same as the rank of R. viz $2n-3$. Let $\hat{r}^\top$ denote the last row of $\hat{R}$, and note that it is obtainable from $r^\top$ through deletion of three entries. It is also a linear combination of the rows first $2n-3$ rows of $\hat{R}$.

Let us also define one further matrix, $\hat{\hat{R}}$, which we term a doubly reduced rigidity matrix, obtained from $\hat{R}$ by deleting its last row.

Define δq_A^{red} to be the vector obtained from δq_A by deletion of those entries of δq_A associated with the pinning process of step 4 above. Define $\delta\hat{q}_A^{red}$ and $\delta\hat{q}_B$ to be the infinitesimal perturbations in vertex positions (disregarding pinned coordinates) given infinitesimal perturbations $\delta d_1, \delta d_2, \delta d_3$ for the transformed framework.

Then these perturbations still satisfy (3.36) but they are now unique. To see this, observe that for the transformed framework, there will hold (with $\mathbf{0}_{mA}, \mathbf{0}_{mB}$ denoting vectors of zeros with $2n_A-3, 2n_B-3$ entries, n_A, n_B denoting the number of nodes in frameworks $\mathcal{T}_A$, $\mathcal{T}_B$)

$$\begin{bmatrix} 0_{mA} \\ 0_{mB} \\ \delta d_1 \\ \delta d_2 \\ \delta d_3 \end{bmatrix} = \hat{\hat{R}} \begin{bmatrix} \delta\hat{q}_A^{red} \\ \delta\hat{q}_B \end{bmatrix}, \tag{3.37}$$

and

$$\begin{bmatrix} 0_{mA} \\ 0_{mB} \\ \delta d_1 \\ \delta d_2 \\ \delta d_3 \\ \delta d_4 \end{bmatrix} = \hat{R} \begin{bmatrix} \delta\hat{q}_A^{red} \\ \delta\hat{q}_B \end{bmatrix} = \begin{bmatrix} \hat{\hat{R}} \\ \hat{r}^\top \end{bmatrix} \begin{bmatrix} \delta\hat{q}_A^{red} \\ \delta\hat{q}_B \end{bmatrix}. \tag{3.38}$$

Now suppose that infinitesimal values of $\delta d_1, \delta d_2$ and δd_3 are specified. It follows from the invertibility of $\hat{\hat{R}}$ that $\delta\hat{q}_A^{red}, \delta\hat{q}_B$ are expressible uniquely in terms

of $\delta d_1, \delta d_2, \delta d_3$ and entries of the inverse of $\hat{\hat{R}}$, i.e.

$$\begin{bmatrix} \hat{\delta} q_A \\ \delta q_B \end{bmatrix} = \hat{\hat{R}}^{-1} \begin{bmatrix} 0_{mA} \\ 0_{mB} \\ \delta d_1 \\ \delta d_2 \\ \delta d_3 \end{bmatrix}. \tag{3.39}$$

Then from (3.38) we have simply

$$\delta d_4 = \hat{r}_4^\top \hat{\hat{R}}^{-1} \begin{bmatrix} 0_{mA} \\ 0_{mB} \\ \delta d_1 \\ \delta d_2 \\ \delta d_3 \end{bmatrix}, \tag{3.40}$$

and now we see how the partial derivatives of the function $\bar{d}_4 = f(\bar{d}_1, \bar{d}_2, \bar{d}_3)$ are identified in terms of entries of the rigidity matrix. The derivatives are in fact the last three entries of the row vector $\hat{r}_4^\top \hat{\hat{R}}^{-1}$.

3.4 Concluding remarks

In this chapter, we considered two scenarios of merging rigid tensegrity frameworks. First, we have shown that by interconnecting infinitesimally rigid tensegrity frameworks with four new members, there exists a distribution of cables and struts to the new members such that the merged tensegrity framework is still infinitesimally rigid. Furthermore, we also proved that the infinitesimal rigidity of the combined tensegrity framework can be obtained without changing the type of pre-existing members. In the case of merging rigid tensegrity frameworks, it has been shown that the rigidity can also be preserved by properly choosing the joining members. To efficiently determine the type of the fourth member once others are fixed, one method has been proposed by invoking the rigidity matrix.

Chapter 4
Growing super stable tensegrity frameworks

This chapter discusses methods for growing tensegrity frameworks akin to what is now known as Henneberg constructions, which apply to bar-joint frameworks. In particular, this chapter presents tensegrity framework versions of the three key Henneberg constructions of vertex addition, edge splitting and framework merging (whereby separate frameworks are combined into a larger framework). This is done for super stable tensegrity frameworks in a Euclidean two or three-dimensional space. We start with the operation of adding a new vertex to an original super stable tensegrity framework, named *vertex addition*. We prove that the new tensegrity framework can be super stable as well if the new vertex is attached to the original framework by an appropriate number of members, which include struts or cables, with suitably assigned stresses. *Edge splitting* can be secured in $\mathbb{R}^2$ ($\mathbb{R}^3$) by adding a vertex joined to three (four) existing vertices, two of which are connected by a member, and then removing that member. This procedure, with appropriate selection of struts or cables, preserves super-stability. In d dimensional Euclidean space, merging two super stable frameworks sharing at least $d+1$ vertices that are in general positions, we show that the resulting tensegrity framework is still super stable. Based on these results, we further investigate the strategies of merging two super stable tensegrity frameworks in $\mathbb{R}^d, (d \in \{2, 3\})$ that share fewer than $d + 1$ vertices, and show how they may be merged through the insertion of struts or cables as appropriate between the two structures, with a super stable structure resulting from the merge.

4.0.1 Introduction

In addition to rigidity and infinitesimal rigidity discussed in Chapter 2, much attention, especially but not exclusively in the tensegrity literature, has been given to super-stability due to its superior properties in robustness. One surprising fact is that a globally rigid tensegrity framework can be drastically deformed under mild perturbation even at an equilibrium configuration [21]. It turns out that it is generally easier to analyze super stable tensegrity structures as opposed to tensegrity structures that are not super stable, due to the availability of more

relevant theoretical foundations. Universally rigid tensegrity structures are often intuitively and easily understandable, for example, we note the concept of *Cauchy polygon* [19]. It is a class of tensegrity frameworks in the plane, where the vertices $1, \cdots, n$ in order form a convex polygon, and the edges $(i, i+1), i = 1, \cdots, n$, are cables and $(i, i+2), i = 1, \cdots, n-2$, are struts with the indices modulo n. In [19], it was shown that any Cauchy polygon is super stable. In addition, sufficient conditions were given for general convex polygons to be super stable, and these conditions are cast in terms of scalar variables termed *stresses*, one of which is associated with each member of the framework. Later, the results were extended in [23] for general tensegrity frameworks. This makes it possible to infer super-stability using the stress concept tool.

Providing foundations to study universal rigidity, [22] and [50] investigated global rigidity for tensegrity frameworks that are generic. These results were further extended to universal rigidity in [49]. In addition, [3] presented conditions for frameworks in general position to be universally rigid. In [2], it was demonstrated that universal rigidity can be maintained even under the weaker condition that each vertex and its neighbors affinely span $\mathbb{R}^d$. In [41], it has been proved that the extended framework is still generically globally rigid if the new vertex is linked to $d+1$ existing vertices in general positions of a generically globally rigid framework.

All these results mentioned above on merging/splitting were for bar frameworks; in contrast, the merging of tensegrity frameworks was first reported in [21], where only two special examples were discussed as illustrations. More recently, it has been shown that a necessary and sufficient condition for a framework obtained by merging two super stable frameworks that are in general positions in $\mathbb{R}^d$ to be super stable, and without the introduction of new members, is that the number of their shared vertices is at least $d+1$ [99]. This has implications for tensegrity frameworks.

In spite of the aforementioned efforts made to study merging of tensegrity frameworks, there exists no systematic strategy for augmenting super stable tensegrity frameworks by adding new vertices in sequence. It is also desirable to design strategies for merging super stable tensegrity frameworks if they share fewer than $d+1$ vertices, indeed possibly no vertices; this requires the introduction of new members. Motivated by these considerations, the aim of this chapter is to first extend the various Henneberg construction steps to super stable tensegrity frameworks in $\mathbb{R}^d$, $(d \in \{2, 3\})$, such that the tensegrity frameworks after the *vertex addition* or *edge splitting* operation are still super stable. We then show that if two super stable tensegrity frameworks in $\mathbb{R}^d$ share at least $d+1$ vertices, super-stability of the merged tensegrity framework can be guaranteed under the weaker condition that only the shared vertices are in general positions. We further develop strategies to merge super stable frameworks in the case of sharing fewer than $d+1$ vertices by introducing new elements in $\mathbb{R}^d$, $(d \in \{2, 3\})$, to bridge the theoretical gap.

Our constructions also are underpinned by algorithms for determining whether an introduced member should be a cable or a strut.

The rest of this chapter is organized as follows. In Section 4.1, we propose a Henneberg construction on super stable frameworks, including vertex addition and edge splitting operations. The strategies of merging super stable frameworks are presented in Section 4.2. We finally give concluding remarks in Section 4.3.

4.1 Henneberg construction on super stable tensegrity frameworks

In this section, we aim at extending the classical *Henneberg constructions (HC)* operating on graphs associated with bar-joint frameworks to super stable tensegrity frameworks in $\mathbb{R}^d, (d \in \{2, 3\})$. Two types of operations to grow minimally rigid graphs are reviewed as follows.

1. Vertex addition: Adding a new vertex u to the existing graph $\mathcal{G}$ via d new edges between u and d vertices in $\mathcal{G}$.

2. Edge splitting: Removing an edge (j, k), then adding a new vertex u and $d+1$ new edges between u and $d+1$ vertices to $\mathcal{G}$, two of which are (u, j) and (u, k).

It can be checked that for both operations in the plane, the increase in the number of edges at each step to form a new minimally rigid graph is two. Correspondingly, for the spatial graphs, the number will increase by three. We first consider the growing of super stable tensegrity frameworks in the plane. Under this scenario, vertex addition requires three new members; any notion of minimality is destroyed. However, if the three new members are linked to vertices for which a pair already have a member between them, that member can be removed without loss of super-stability by properly adjusting the remaining members' stresses, known as edge splitting, and each additional vertex involves adding d new members. Thus this is a cheaper approach in terms of members than vertex addition.

The tensegrity framework $(\mathcal{G}, q)$ to be operated on is assumed to be super stable with $n \geqslant 3$ vertices, three arbitrary vertices of which are denoted by i, j and k. The resulting tensegrity framework after adding the new vertex u and new members of cables and struts, is denoted by $(\bar{\mathcal{G}}, \bar{q})$, where $\bar{q} = [q_1, \cdots, q_n, q_u] \in \mathbb{R}^{2\times(n+1)}$. Now, we first consider the vertex addition operation to generate a super stable framework $(\bar{\mathcal{G}}, \bar{q})$.

4.1.1 Vertex addition in $\mathbb{R}^2$

The position of the new vertex u to be connected to $(\mathcal{G}, q)$ can fall into the following three situations:

(a) not collinear with any two of i, j and k;

(b) collinear with two of i, j and k;

(c) collinear with all of i, j, k. (This situation can be reduced to (b).)

For situation (a), under the assumption that i, j and k are not collinear, there are seven possible regions to place the new vertex u, shown in Fig. 4.1, denoted by region $A, B \cdots, F$, and H. Note that the members (cables or struts) need to be inserted between the new vertex u and the vertices in the original tenserity framework $(\mathcal{G}, q)$ vary as the position of vertex u changes. But, the necessary condition of the equilibrium stress with respect to vertex u is always

$$\omega_{ui}(q_u - q_i) + \omega_{uj}(q_u - q_j) + \omega_{uk}(q_u - q_k) = \mathbf{0}, \tag{4.1}$$

where ω_{ui}, ω_{uj} and ω_{uk} are the stresses of members $(u, i), (u, j)$ and (u, k), respectively. Here, we associate the new vertex u with three vertices i, j and k rather than only two, since in scenario (a), any two of the three vectors, $(q_u - q_i), (q_u - q_j)$ and $(q_u - q_k)$, are linearly independent, which implies that there is no solution to (4.1) if we remove any single term on its left-hand side; equivalently, the three stresses must all be nonzero. This immediately means that in the plane, any one of the three vectors can be represented as a linear combination of the other two. Without loss of generality, we assume

$$q_u - q_k = \kappa_1(q_u - q_i) + \kappa_2(q_u - q_j), \tag{4.2}$$

where κ_1 and κ_2 are nonzero scalars. Using the fact that any two vectors in the vector set $\{(q_u - q_i), (q_u - q_j), (q_u - q_k)\}$ are linearly independent, we have

$$\omega_{ui} + \kappa_1\omega_{uk} = 0, \tag{4.3a}$$

$$\omega_{uj} + \kappa_2\omega_{uk} = 0. \tag{4.3b}$$

Now, we record the member assignations (cable/strut) required to meet the equilibrium stress condition with respect to u in different regions.

1. The new vertex u lies in regions outside of H, i.e., $A, \cdots, F$, shown in Fig. 4.1.

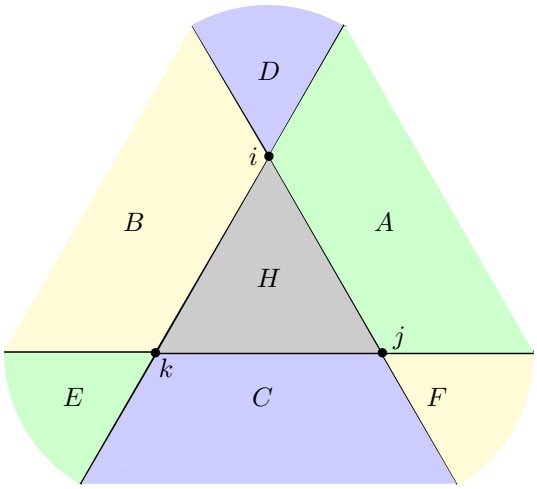


Figure 4.1: *Possible regions for placing u in scenario (a).*

First, consider the case when u lies in region A or E. In this case, the two scalars κ_1 and κ_2 in (4.2) are both positive, i.e., $\kappa_1 > 0$ and $\kappa_2 > 0$. Then, (4.3) implies

$$\begin{cases} \omega_{ui}\omega_{uk} < 0 \\ \omega_{uj}\omega_{uk} < 0 \\ \omega_{ui}\omega_{uj} > 0 \end{cases}, \tag{4.4}$$

which in turn implies

$$\begin{cases} \omega_{ui} > 0 \\ \omega_{uk} < 0 \\ \omega_{uj} > 0 \end{cases}, \quad \text{or} \quad \begin{cases} \omega_{ui} < 0 \\ \omega_{uk} > 0 \\ \omega_{uj} < 0 \end{cases}. \tag{4.5}$$

Equivalently, members (u, i) and (u, j) are cables with (u, k) being a strut, or members (u, i) and (u, j) are struts with (u, k) being a cable.

Analogously, when vertex u is located in region B or F, we know (u, i) and (u, k) are the same type of members, either cable or strut, while (u, j) should be different from them; when vertex u is located in region C or D, the two members that are of the same type are (u, j) and (u, k), which differ from member (u, i).

2. The new vertex u lies in region H.

   In this case, from the geometric relationship, we know both κ_1 and κ_2 in

(4.2) are negative, and consequently solutions to (4.3) satisfy

$$\begin{cases} \omega_{ui}\omega_{uk} > 0, \\ \omega_{uj}\omega_{uk} > 0, \\ \omega_{ui}\omega_{uj} > 0, \end{cases} \tag{4.6}$$

which implies all the three stresses have the same sign. In other words, when the newly added vertex u lies within the convex hull spanned by the three existing vertices i, j and k, the three new members connecting u and i, j, k are of the same type, which are either cables or struts.

We then consider situation (b) for which the newly added vertex u is collinear with two of the existing vertices, say i and j, and thus the new members to be inserted are (u, i) and (u, j). In view of the collinearity between i, j and u, we have

$$q_u - q_i = \lambda(q_u - q_j), \tag{4.7}$$

where $\lambda > 0$ if u lies outside of the line segment with two endpoints i and j; $\lambda < 0$, otherwise. Hence, the equilibrium stress condition (4.1) reduces to

$$\omega_{ui}(q_u - q_i) + \omega_{uj}(q_u - q_j) = \mathbf{0}, \tag{4.8}$$

where ω_{ui} and ω_{uj} are stresses of the new members (u, i) and (u, j), respectively. Consequently, $\omega_{ui}\omega_{uj} < 0$ if $\lambda > 0$; $\omega_{ui}\omega_{uj} > 0$, if $\lambda < 0$. In other words, when the new vertex u is not between i and j, the two new members (u, i) and (u, j) are of different types. In contrast, when the new vertex u is between i and j, the two new members are of the same type. At the same time, it should be noted that to stabilize three vertices in $\mathbb{R}^1$, the two members incident to the middle vertex should be of the same type, and the other member connecting the two endpoints is of the other type. A sketch will rapidly show these conclusions are intuitively reasonable, if not obvious.

Situation (c) can be reduced to situation (b) by only considering the new vertex u and any two of the three collinear vertices i, j, k in $(\mathcal{G}, q)$. Actually, both (b) and (c) can be regarded as operations in $\mathbb{R}^1$.

The main theorem on vertex addition for super stable tensegrity frameworks in the plane is given as follows.

Theorem 4.1. *Given a super stable tensegrity framework $(\mathcal{G}, q)$ in $\mathbb{R}^2$, consider two growing strategies in terms of the position of the new vertex u. One is adding a new vertex u and three members between u and three distinct noncollinear vertices i, j and k to $(\mathcal{G}, q)$ when u is not collinear with any two of i, j, k. The other one is adding u and two members between u and two distinct vertices i, j when u is collinear with two*

vertices of the original framework. Then there always exist stresses of the new members, such that the newly obtained tensegrity framework $(\bar{\mathcal{G}}, \bar{q})$ is also super stable.

Proof. First, we consider the scenario when the new vertex u is not collinear with any two of the three distinct noncollinear vertices i, j and k in $(\mathcal{G}, q)$. Note that the equilibrium condition (4.1) can be written as

$$\underbrace{[q_u - q_i, q_u - q_j, q_u - q_k]}_{\triangleq q_r} \begin{bmatrix} \omega_{ui} \\ \omega_{uj} \\ \omega_{uk} \end{bmatrix} = \mathbf{0}, \tag{4.9}$$

where $q_r \in \mathbb{R}^{2\times 3}$. Since $rank(q_r) = 2$, the solution to (4.9) with respect to ω cannot be uniquely determined. However, for a fixed but arbitrary vector $[a_1, a_2, a_3]^T$ satisfying $a_1 + a_2 + a_3 \neq 0$ in the null space of q_r, the solution to (4.9) is

$$\omega_{ui} = a_1 s, \quad \omega_{uj} = a_2 s, \quad \omega_{uk} = a_3 s, \tag{4.10}$$

for $s \in \mathbb{R}$ and $s \neq 0$. In view of the non-collinearity of the three vertices, there holds $q_k - q_u = c_1(q_k - q_i) + c_2(q_k - q_j)$ for some nonzero c_1, c_2. It follows that $c_1(q_u - q_i) + c_2(q_u - q_j) - (c_1 + c_2 - 1)(q_u - q_k) = 0$. Then one can observe that there always exist vectors satisfying (4.10).

Assume the stress matrix of the original framework $(\mathcal{G}, q)$ is $\Omega \in \mathbb{R}^{n\times n}$, which is positive semi-definite with rank $n - 3$. Then, to derive the new stress matrix $\bar{\Omega} \in \mathbb{R}^{(n+1)\times(n+1)}$ for the framework $(\bar{\mathcal{G}}, \bar{q})$, one seeks to directly augment Ω by adding a new row and column to Ω in the form of

$$\hat{\Omega} = \left(\begin{array}{ccc:c} & & & 0 \\ & & & \vdots \\ & \Omega & & 0 \\ & & & -\omega_{ui} \\ & & & -\omega_{uj} \\ & & & -\omega_{uk} \\ \hdashline 0 & \cdots & 0 & -\omega_{ui} \quad -\omega_{uj} \quad -\omega_{uk} \quad \hat{\Omega}_{uu} \end{array}\right). \tag{4.11}$$

However, this $\hat{\Omega}$ is not a stress matrix, since the $(n-2)$th to nth row/column sum is not zero. Therefore, to obtain a valid stress matrix based on $\hat{\Omega}$, the values of some entries in the original stress matrix Ω need to be changed correspondingly. Further, to ensure the new tensegrity framework $(\bar{\mathcal{G}}, \bar{q})$ is super stable, the new stress matrix should be positive semi-definite with rank $n - 2$.

Since the new edges might affect the stresses of the edges between vertices i, j

and k, we look for the new stress matrix $\hat{\Omega}$ with the following form

$$\hat{\Omega} = \underbrace{\left(\begin{array}{c:c} \Omega & \mathbf{0}_{n\times 1} \\ \hdashline \mathbf{0}_{1\times n} & 0 \end{array}\right)}_{\triangleq \Omega_a} + \underbrace{\left(\begin{array}{c:c} \mathbf{0}_{(n-3)\times(n-3)} & \mathbf{0}_{(n-3)\times 4} \\ \hdashline \mathbf{0}_{4\times(n-3)} & \Omega_u \end{array}\right)}_{\triangleq \Omega_b}, \tag{4.12}$$

where $\Omega_u \in \mathbb{R}^{4\times 4}$ is a positive semi-definite stress matrix of rank 1 associated with the vertices i, j, k and u. Existence and construction of Ω_u will be demonstrated later. Further, we seek to ensure that $\hat{\Omega}$ satisfies

a) $\hat{\Omega}$ is positive semi-definite.

b) $\hat{\Omega}$ is a stress matrix associated with vertices $1, \cdots, n, u$, whose stresses are in equilibrium with the configuration $\bar{q} = [q, q_u] \in \mathbb{R}^{2\times(n+1)}$.

c) $rank(\hat{\Omega}) = n - 2$.

For statement a), it is straightforward to check Ω_a and Ω_b are both positive semi-definite from (4.12). So obviously, $\hat{\Omega} = \Omega_a + \Omega_b$ is also positive semi-definite.

For statement b), consider the facts that

$$\sum_{j=1,\cdots,n,(n+1)} \omega_{ij}^a (q_j - q_i) = \mathbf{0}, \quad \forall i, \tag{4.13}$$

and

$$\sum_{j=(1,\cdots,n-3),n-2,\cdots,n+1} \omega_{ij}^b (q_j - q_i) = \mathbf{0}, \quad \forall i, \tag{4.14}$$

where ω_{ij}^a and ω_{ij}^b are respectively the entries associated with matrices Ω_a and Ω_b, vertices i, j and k are assigned with the indexes as $(n-2)$, $(n-1)$ and n, repectively, and the new vertex u is labeled as $n+1$ for consistence. Summing up (4.13) and (7.19), we get the equilibrium equation

$$\sum_{j=1,\cdots,n+1} \hat{\omega}_{ij} (q_j - q_i) = \mathbf{0}, \quad \forall i, \tag{4.15}$$

where $\hat{\omega}_{ij} = \omega_{ij}^a + \omega_{ij}^b$.

Furthermore, it can be concluded from Lemma A.1 in the Appendix that statement c) also holds.

Hence, the augmented stress matrix $\hat{\Omega}$ through operation (4.12) is positive semi-definite with the maximal rank $n-2$, and the stresses are in equilibrium with $\bar{q}$. Note that for a general framework $(\mathcal{G}, q)$ that is rigid, through the typical Henneberg operation, the resulted new framework is still rigid. Hence, it can be concluded from Lemma 2.14 that the new framework $(\bar{\mathcal{G}}, \bar{q})$ is super stable. In the

construction, the type of the new members, strut or cable, is determined by the signs of the stresses, which satisfy (4.9) and (4.10).

As for the scenario that the newly added vertex u is collinear with two existing vertices in the original framework, the dimension of the stress matrix Ω_u in (4.12) will decrease to 3-by-3 since three vertices are sufficient to determine a super stable tensegrity framework in $\mathbb{R}^1$. Moreover, it should be noted that in this case only two new members are required to make the new tensegrity framework super stable. The proof can be conducted following the same argument as above, which is omitted here.

To sum up, we have shown that for a super stable framework in the plane, by vertex addition, the newly obtained tensegrity framework is still super stable. □

Remark 4.2. When vertices i, j and k in $(\mathcal{G}, q)$ are collinear, one can always find another vertex k' in the original framework such that i, j and k' are not collinear; otherwise the tensegrity framework will be reduced to $1D$. Then the new vertex u will be connected to vertices i, j and k'. Following the same analysis, we know there exist proper stresses of the new members such that the augmented framework $(\bar{\mathcal{G}}, \bar{q})$ is super stable.

4.1.2 Vertex addition in $\mathbb{R}^3$

For the vertex addition in $\mathbb{R}^3$, the type of new members are also determined by the position of the new vertex u with respect to the four vertices, denoted by i, j, k and l, to be connected in $(\mathcal{G}, q)$. In view of their geometric relationship in the space, three cases might arise, namely

(a) The new vertex u is collinear with two of the four vertices;

(b) The new vertex u is coplanar with three of the four vertices;

(c) u and the four vertices are neither collinear nor coplanar.

Cases (a) and (b) can be reduced to $\mathbb{R}^1$ and $\mathbb{R}^2$ respectively, which have been addressed above. For case (c), analogously, the equilibrium stress condition with respect to u implies

$$\omega_{ui}(q_u - q_i) + \omega_{uj}(q_u - q_j) + \omega_{uk}(q_u - q_k) + \omega_{ul}(q_u - q_l) = \mathbf{0}, \tag{4.16}$$

where $\omega_{ui}, \omega_{uj}, \omega_{uk}$ and ω_{ul} are the stresses of members $(u, i), (u, j), (u, k)$ and (u, l), respectively. Again from the linear independence relationship, we have

$$q_u - q_l = \kappa_1'(q_u - q_i) + \kappa_2'(q_u - q_j) + \kappa_3'(q_u - q_k), \tag{4.17}$$

where κ_1', κ_2' and κ_3' are nonzero scalars. Combining (4.16) and (4.17), we know

$$\begin{cases} \omega_{ui} + \kappa_1' \omega_{ul} = 0, \\ \omega_{uj} + \kappa_2' \omega_{ul} = 0, \\ \omega_{uk} + \kappa_3' \omega_{ul} = 0. \end{cases} \tag{4.18}$$

Then, following the same analysis in $\mathbb{R}^2$, one can determine the type of new members by looking at the signs of the stresses, derived from (4.18). To avoid repetition, we omit the details here. Correspondingly, for case (c), we have the following main result on vertex addition for super stable tensegrity frameworks in $\mathbb{R}^3$.

Corollary 4.3. *For a given super stable tensegrity framework $(\mathcal{G}, q)$ in $\mathbb{R}^3$, adding a new vertex u and four members between u and four distinct vertices in $(\mathcal{G}, q)$, where there exists no collinear or coplanar relationship between u and the four vertices, there always exist stresses of the members incident to the chosen vertices, such that the extended tensegrity framework is also super stable.*

The same strategy employed in the proof of Theorem 4.1 can be used for proving Corollary 4.3. We omit it here, again to avoid repetition.

4.1.3 Computation of the stress matrix Ω_u

In this subsection, for completeness, we present the specific form of the matrix Ω_u. Since the techniques used in the computation of the matrix Ω_u in $\mathbb{R}^2$ and $\mathbb{R}^3$ are the same, we only focus on the scenario of $\mathbb{R}^2$. For the case when u is not collinear with any two of the existing vertices i, j and k, the stresses of the newly added members are represented in (4.10), based on which we will come up with a numerical method to derive the stress matrix Ω_u. Before moving on, we define the sub-configuration matrix with respect to vertices i, j, k and u as

$$Q_u \triangleq \begin{pmatrix} q_i & q_j & q_k & q_u \\ 1 & 1 & 1 & 1 \end{pmatrix} \in \mathbb{R}^{3\times 4}, \tag{4.19}$$

and note it satisfies

$$Q_u \Omega_u = \mathbf{0}_{3\times 4}. \tag{4.20}$$

Since $rank(Q_u) = 3$, there exists a nonzero vector $\phi = [\phi_1, \phi_2, \phi_3, \phi_4]^T \in \mathbb{R}^4$ satisfying

$$Q_u \phi = \mathbf{0}. \tag{4.21}$$

Then matrix Ω_u can be determined up to scaling through

$$\Omega_u = \phi\phi^T = \begin{pmatrix} \phi_1^2 & \phi_1\phi_2 & \phi_1\phi_3 & \phi_1\phi_4 \\ \phi_2\phi_1 & \phi_2^2 & \phi_2\phi_3 & \phi_2\phi_4 \\ \phi_3\phi_1 & \phi_3\phi_2 & \phi_3^2 & \phi_3\phi_4 \\ \phi_4\phi_1 & \phi_4\phi_2 & \phi_4\phi_3 & \phi_4^2 \end{pmatrix}. \tag{4.22}$$

Combining (4.22) and (4.10), we have

$$\begin{cases} \phi_1\phi_4 = -\omega_{ui} = -a_1 s \\ \phi_2\phi_4 = -\omega_{uj} = -a_2 s \\ \phi_3\phi_4 = -\omega_{uk} = -a_3 s \end{cases}. \tag{4.23}$$

Furthermore, in light of the fact that the row/column sum of Ω_u in (4.22) is zero, we know

$$\phi_4^2 = (a_1 + a_2 + a_3)s. \tag{4.24}$$

Then, by setting s so that $(a_1 + a_2 + a_3)s > 0$, it follows from (4.23) and (4.24) that ϕ can be represented in terms of s as follows

$$\begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{pmatrix} = \frac{1}{\sqrt{(a_1 + a_2 + a_3)s}} \begin{pmatrix} -a_1 s \\ -a_2 s \\ -a_3 s \\ (a_1 + a_2 + a_3)s \end{pmatrix}. \tag{4.25}$$

Therefore, as long as s is determined, the specific form of Ω_u can be obtained as well by substituting (4.25) into (4.21).

Based on (4.25), Ω_u is in the form of

$$\Omega_u = \frac{1}{\Omega_{uu}} \begin{pmatrix} \omega_{ui}^2 & \omega_{ui}\omega_{uj} & \omega_{ui}\omega_{uk} & -\omega_{ui}\Omega_{uu} \\ \omega_{ui}\omega_{uj} & \omega_{uj}^2 & \omega_{uj}\omega_{uk} & -\omega_{uj}\Omega_{uu} \\ \omega_{ui}\omega_{uk} & \omega_{uj}\omega_{uk} & \omega_{uk}^2 & -\omega_{uk}\Omega_{uu} \\ -\omega_{ui}\Omega_{uu} & -\omega_{uj}\Omega_{uu} & -\omega_{uk}\Omega_{uu} & \Omega_{uu}^2 \end{pmatrix}. \tag{4.26}$$

For the case when vertex u is collinear with at least two vertices, we omit the calculation procedure here due to space limitations. It is similar to the computations above.

Remark 4.4. If the configuration of vertices i, j, k and u is fixed, the values of Ω_u is unique up to the *affine transformation* of $[q_i, q_j, q_k, q_u]$. We define the affine

transformation of q by

$$\begin{aligned}\mathcal{A}(q) \triangleq \{p = [p_1, \cdots, p_n] | p_i = Aq_i + b,\\ A \in \mathbb{R}^{d\times d} \text{ and } b \in \mathbb{R}^d, i = 1, \cdots, n\}.\end{aligned} \tag{4.27}$$

4.1.4 Edge splitting

In this subsection, the edge splitting strategy on super stable tensegrity frameworks is designed based on the vertex addition of a degree 3 or degree 4 vertex in $\mathbb{R}^2$ or $\mathbb{R}^3$ respectively, together with the removal of a member (j,k) of the original tensegrity framework. To be consistent with the discussions above, the matrix $\hat{\Omega}$ will denote the stress matrix of the new super stable tensegrity framework after the operation of vertex addition. Note that from the perspective of stress, removing a member (following the vertex addition) is equivalent to altering the stress of the corresponding member to be zero without changing the positive semi-definiteness and the rank of $\hat{\Omega}$, as well as the self-equilibrium condition for $\bar{q}$. As mentioned before, the new vertex u can lie in several possible regions. We first consider the case when u is not collinear (coplanar) with any two (three) of the existing vertices i, j and k (i, j, k and l) in $\mathbb{R}^2$ ($\mathbb{R}^3$). The main result is given as follows.

Theorem 4.5. *Assume we remove a member (j,k) in the original super stable tensegrity framework $(\mathcal{G}, q)$ in $\mathbb{R}^2$ ($\mathbb{R}^3$), and then add to $(\mathcal{G}, q)$ a new vertex u together with three (four) members incident on u, two of which are (u,j) and (u,k). Then, there exist appropriate stresses of the three (four) members such that the new tensegrity framework $(\mathcal{G}', \bar{q})$ is super stable.*

Proof. We present the proof only for $\mathbb{R}^2$ for simplicity; it can be straightforwardly extended to the analysis in $\mathbb{R}^3$. The stress matrix after a vertex addition operation is presented in (4.39).

$$\hat{\Omega} = \left(\begin{array}{ccc:ccc:c} \Omega_{1,1} & \cdots & \Omega_{1,n-3} & \Omega_{1,n-2} & \Omega_{1,n-1} & \Omega_{1,n} & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Omega_{n-3,1} & \cdots & \Omega_{n-3,n-3} & \Omega_{n-3,n-2} & \Omega_{n-3,n-1} & \Omega_{n-3,n} & 0 \\ \hdashline \Omega_{i,1} & \cdots & \Omega_{i,n-3} & \Omega_{ii} + \frac{\omega_{ui}^2}{\Omega_{uu}} & \Omega_{ij} + \frac{\omega_{ui}\omega_{uj}}{\Omega_{uu}} & \Omega_{ik} + \frac{\omega_{ui}\omega_{uk}}{\Omega_{uu}} & -\omega_{ui} \\ \Omega_{j,1} & \cdots & \Omega_{j,n-3} & \Omega_{ji} + \frac{\omega_{uj}\omega_{ui}}{\Omega_{uu}} & \Omega_{jj} + \frac{\omega_{uj}^2}{\Omega_{uu}} & \Omega_{jk} + \frac{\omega_{uj}\omega_{uk}}{\Omega_{uu}} & -\omega_{uj} \\ \Omega_{k,1} & \cdots & \Omega_{k,n-3} & \Omega_{ki} + \frac{\omega_{uk}\omega_{ui}}{\Omega_{uu}} & \Omega_{kj} + \frac{\omega_{uk}\omega_{uj}}{\Omega_{uu}} & \Omega_{kk} + \frac{\omega_{uk}^2}{\Omega_{uu}} & -\omega_{uk} \\ 0 & \cdots & 0 & -\omega_{ui} & -\omega_{uj} & -\omega_{uk} & \Omega_{uu} \end{array}\right). \tag{4.39}$$

Notice that in light of (4.25), the values of the entries of the matrix Ω_u in (4.26) is uniquely determined up to the scaling variable s. This implies that we have one degree of freedom to set the values of ω_{ui}, ω_{uj} and ω_{uk}. The observation motivates us to seek to zero out $\hat{\Omega}_{jk}$ through properly setting ω_{uk} such that

$$\Omega_{jk} + \frac{\omega_{uj}\omega_{uk}}{\Omega_{uu}} = 0.$$

Then by simple calculation, it follows

$$\omega_{uk} = -\frac{\Omega_{jk}\Omega_{uu}}{\omega_{uj}}. \tag{4.40}$$

Replacing ω_{uk} in (4.39) with (4.40), we have the matrix $\hat{\Omega}'$ given as follows.

$$\hat{\Omega}' = \left(\begin{array}{ccc:ccccc} \Omega_{1,1} & \cdots & \Omega_{1,n-3} & \Omega_{1,n-2} & \Omega_{1,n-1} & \Omega_{1,n} & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Omega_{n-3,1} & \cdots & \Omega_{n-3,n-3} & \Omega_{n-3,n-2} & \Omega_{n-3,n-1} & \Omega_{n-3,n} & 0 \\ \hdashline \Omega_{i,1} & \cdots & \Omega_{i,n-3} & \Omega_{ii} + \frac{\omega_{ui}^2}{\Omega_{uu}} & \Omega_{ij} + \frac{\omega_{ui}\omega_{uj}}{\Omega_{uu}} & \Omega_{ik} - \frac{\omega_{ui}}{\omega_{uj}}\Omega_{jk} & -\omega_{ui} \\ \Omega_{j,1} & \cdots & \Omega_{j,n-3} & \Omega_{ji} + \frac{\omega_{uj}\omega_{ui}}{\Omega_{uu}} & \Omega_{jj} + \frac{\omega_{uj}^2}{\Omega_{uu}} & 0 & -\omega_{uj} \\ \Omega_{k,1} & \cdots & \Omega_{k,n-3} & \Omega_{ik} - \frac{\omega_{ui}}{\omega_{uj}}\Omega_{jk} & 0 & \Omega_{kk} + \frac{\Omega_{jk}^2\Omega_{uu}}{\omega_{uj}^2} & \frac{\Omega_{jk}\Omega uu}{\omega_{uj}} \\ 0 & \cdots & 0 & -\omega_{ui} & -\omega_{uj} & \frac{\Omega_{jk}\Omega uu}{\omega_{uj}} & \Omega_{uu} \end{array}\right). \tag{4.43}$$

It is obvious that $rank(\hat{\Omega}') = rank(\hat{\Omega})$. Moreover, the positive semi-definiteness, as well as the null space, of the matrix $\hat{\Omega}$ is not altered. Therefore, the new stress matrix $\hat{\Omega}'$ is still positive semi-definite with rank $n-2$, and at equilibrium with the configuration $\bar{q}$. Recalling that rigidity of a framework can be maintained through typical Henneberg operation, so the new tensegrity framework $(\mathcal{G}', \bar{q})$ is still super stable with the corresponding stress matrix $\hat{\Omega}'$.

□

Note that if u is coplanar with some of the vertices in $\mathbb{R}^3$, then one can fall back on the analysis in $\mathbb{R}^2$. Hence, as for the location of the new vertex u, we only need to consider another possible scenario that u is collinear with two vertices in $\mathbb{R}^2$. In this case, only three vertices together with three members are involved to construct the stress matrix Ω_u, and the dimension of their configuration has reduced to one. It can be further checked that no one of the three members can be removed without losing super-stability. Hence, for the collinear situation, only when the newly added vertex u is collinear with at least three vertices in the original tensegrity framework

$(\mathcal{G}, q)$, can an edge splitting operation be conducted. We have the following result.

Corollary 4.6. *Given a super stable tensegrity framework $(\mathcal{G}, q)$ with three collinear vertices i, j and k, add a new vertex u on some member (j, k) and thus replace the member (j, k) by two new members (j, u) and (u, k). Then, there exist appropriate members (j, u), (u, k) and (u, i) to be inserted to $(\mathcal{G}, q)$ such that the new tensegrity framework is still super stable.*

Remark 4.7. The idea of Corollary 4.6 is the same as that of Theorem 4.5, namely, remove some member by altering its stress to be zero through properly setting one of the stresses associated with the new members. Hence, the proof of Corollary 4.6 is omitted here. For the case when the new vertex u is collinear with four or more vertices, only three of them together with the new vertex u are needed to conduct the edge splitting operation.

4.2 Merging two super stable tensegrity frameworks

In this section, we aim to investigate the strategies of merging two super stable tensegrity frameworks $(\mathcal{G}_A, q_A)$ and $(\mathcal{G}_B, q_B)$. According to the number of shared vertices between the two tensegrity frameworks before merging, denoted by $|\mathcal{V}_C|$, we consider two sub-scenarios: $|\mathcal{V}_C| \geqslant d+1$, and $|\mathcal{V}_C| < d+1$. When $(\mathcal{G}_A, q_A)$ and $(\mathcal{G}_B, q_B)$ share no fewer than $d+1$ vertices, we show that the merged tensegrity framework is still super stable if the shared vertices are in general position. This result relaxes the stringent condition that both of the two frameworks need to be in general positions in [99]. For the case when $|\mathcal{V}_C| < d+1$, we summarize the results recording the minimum number of new members required in a table by constraining d to be 2 and 3. The type of these members, i.e. strut or cable, depends on the specific location of the various vertices, and so cannot be recorded.

In the following, we denote the positive semi-definite (PSD) stress matrices associated with $(\mathcal{G}_A, q_A)$ and $(\mathcal{G}_B, q_B)$ as Ω_A and Ω_B, respectively, each of which has nullity $d+1$. The cardinalities of the vertex sets satisfy $|\mathcal{V}_A| = n_A$, $|\mathcal{V}_B| = n_B$, and $|\mathcal{V}_C| = n_C$.

4.2.1 The number of shared vertices is no fewer than $d+1$

To be consistent with the merging of two tensegrity frameworks, we assume that the last (resp. first) n_C rows and columns of Ω_A (resp. Ω_B) correspond to the stresses incident on the shared vertices. The merged tensegrity framework is denoted by $(\tilde{\mathcal{G}}, \tilde{q})$ with the stress matrix $\tilde{\Omega} \in \mathbb{R}^{n \times n}$, where $\tilde{n} = n_A + n_B - n_C$. Accordingly, we argument the stress matrices Ω_A and Ω_B to form matrices $\tilde{\Omega}_A$ and $\tilde{\Omega}_B$ of size $\tilde{n} \times \tilde{n}$

by adding zeros as follows:

$$\begin{aligned}\tilde{\Omega}_A &= \left(\begin{array}{c:c} \Omega_A & \mathbf{0}_{n_A\times(\tilde{n}-n_A)} \\ \hdashline \mathbf{0}_{(\tilde{n}-n_A)\times n_A} & \mathbf{0}_{(\tilde{n}-n_A)\times(\tilde{n}-n_A)} \end{array}\right), \\ \tilde{\Omega}_B &= \left(\begin{array}{c:c} \mathbf{0}_{(n_A-n_C)\times(n_A-n_C)} & \mathbf{0}_{(n_A-n_C)\times n_B} \\ \hdashline \mathbf{0}_{n_B\times(n_A-n_C)} & \Omega_B \end{array}\right).\end{aligned} \tag{4.44}$$

Note that the stress matrices Ω_A and Ω_B can also be partitioned as

$$\Omega_A = \left(\begin{array}{c:c} \Omega_{A1} & \Omega_{A2} \\ \hdashline \Omega_{A3} & \Omega_{A4} \end{array}\right), \quad \text{and} \quad \Omega_B = \left(\begin{array}{c:c} \Omega_{B4} & \Omega_{B2} \\ \hdashline \Omega_{B3} & \Omega_{B1} \end{array}\right), \tag{4.45}$$

where $\Omega_{A1} \in \mathbb{R}^{(n_A-n_C)\times(n_A-n_C)}$, $\Omega_{A2} \in \mathbb{R}^{(n_A-n_C)\times n_C}$, $\Omega_{A3} \in \mathbb{R}^{n_C\times(n_A-n_C)}$, $\Omega_{A4} \in \mathbb{R}^{n_C\times n_C}$, $\Omega_{B1} \in \mathbb{R}^{(n_B-n_C)\times(n_B-n_C)}$, $\Omega_{B2} \in \mathbb{R}^{n_C\times(n_B-n_C)}$, $\Omega_{B3} \in \mathbb{R}^{(n_B-n_C)\times n_C}$, and $\Omega_{B4} \in \mathbb{R}^{n_C\times n_C}$. Then, the stress matrix of the post-merged tensegrity framework $(\tilde{\mathcal{G}}, \tilde{q})$ can be written as

$$\begin{aligned}\tilde{\Omega} &= \tilde{\Omega}_A + \tilde{\Omega}_B \\ &= \begin{pmatrix} \Omega_{A1} & \Omega_{A2} & \mathbf{0}_{(n_A-n_C)\times(n_B-n_C)} \\ \Omega_{A3} & \Omega_{A4}+\Omega_{B4} & \Omega_{B2} \\ \mathbf{0}_{(n_B-n_C)\times(n_A-n_C)} & \Omega_{B3} & \Omega_{B1} \end{pmatrix}.\end{aligned} \tag{4.46}$$

Now, we are ready to give another main result.

Theorem 4.8. *Given two super stable tensegrity frameworks in $\mathbb{R}^d$ with the corresponding PSD stress matrices of nullity $d+1$, if they share at least $d+1$ vertices that are in general position, then the merged tensegrity framework $(\tilde{\mathcal{G}}, \tilde{q})$ is still super stable. Moreover, one of the PSD stress matrices of nullity $d+1$ associated with the new framework is in the form of* (4.46).

Proof. We first consider the case when the two tensegrity frameworks share exactly $d+1$ vertices, i.e., $n_C = d+1$. Then, by denoting the configuration of shared $d+1$ vertices as $q_{C1}, \cdots, q_{C(d+1)}$, one has

$$\tilde{q} = [q_{A1}, \cdots, q_{A(n_A-d-1)}, q_{C1}, \cdots, q_{C(d+1)}, q_{B(d+2)}, \cdots, q_{Bn_B}]. \tag{4.47}$$

From Lemma 2.14, to show that $(\tilde{\mathcal{G}}, \tilde{q})$ is super stable, it is sufficient to prove the synthetic stress matrix $\tilde{\Omega}$ in (4.46) satisfies the three conditions therein. It is obvious that $\tilde{\Omega}$ is PSD, as $\tilde{\Omega}_A$ and $\tilde{\Omega}_B$ are both PSD from their definitions in (4.44). In addition, for two rigid frameworks in $\mathbb{R}^d$, if they share no fewer than d vertices, then the framework after merging is rigid [133], which implies that the third condition in Lemma 2.14 is satisfied. Hence, what is left to show is that the

rank of $\tilde{\Omega}$ is $\tilde{n}-d-1$, namely, the nullity of $\tilde{\Omega}$ is $d+1$.

Similar to the analysis in the proof of Theorem 4.1, we consider the solution space of the following equations,

$$\tilde{\Omega}_A x_A = \mathbf{0}, \tag{4.48a}$$

$$\tilde{\Omega}_B x_B = \mathbf{0}. \tag{4.48b}$$

Then the solution spaces of (4.48a) and (4.48b) are respectively given by

$$\mathbb{S}_A = \left\{ \begin{pmatrix} q^A_{11} \\ \vdots \\ q^A_{(n_A-d-1)1} \\ q^C_{11} \\ \vdots \\ q^C_{(d+1)1} \\ \xi_{11} \\ \vdots \\ \xi_{(n_B-d-1)1} \end{pmatrix}, \cdots, \begin{pmatrix} q^A_{1d} \\ \vdots \\ q^A_{(n_A-d-1)d} \\ q^C_{1d} \\ \vdots \\ q^C_{(d+1)d} \\ \xi_{1d} \\ \vdots \\ \xi_{(n_B-d-1)d} \end{pmatrix}, \begin{pmatrix} 1 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 1 \\ c_{A1} \\ \vdots \\ c_{A(n_B-d-1)} \end{pmatrix} \right\}, \tag{4.49}$$

and

$$\mathbb{S}_B = \left\{ \begin{pmatrix} \zeta_{11} \\ \vdots \\ \zeta_{(n_A-d-1)1} \\ q^C_{11} \\ \vdots \\ q^C_{(d+1)1} \\ q^B_{(d+2)1} \\ \vdots \\ q^B_{n_B 1} \end{pmatrix}, \cdots, \begin{pmatrix} \zeta_{1d} \\ \vdots \\ \zeta_{(n_A-d-1)d} \\ q^C_{1d} \\ \vdots \\ q^C_{(d+1)d} \\ q^B_{(d+2)d} \\ \vdots \\ q^B_{n_B d} \end{pmatrix}, \begin{pmatrix} c_{B1} \\ \vdots \\ c_{B(n_A-d-1)} \\ 1 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \right\}, \tag{4.50}$$

where for configuration q the superscript denotes the configuration set, and the subscripts, say (ij) in q^A_{ij}, represent the jth component of vector q_{Ai}. $\xi_i \in \mathbb{R}^d, i = 1, \cdots, n_B-d-1$, $\zeta_j \in \mathbb{R}^d, j = 1, \cdots, n_A-d-1$, $c_A \in \mathbb{R}^{n_B-d-1}$, and $c_B \in \mathbb{R}^{n_A-d-1}$ are arbitrary real vectors. Following the same line of the proof of Theorem 4.1, we get

$$null(\tilde{\Omega}) = \mathbb{S}_A \cap \mathbb{S}_B = span\left(\tilde{q}^T, \mathbf{1}_{\tilde{n}}\right), \tag{4.51}$$

which implies $nul(\tilde{\Omega}) = d+1$. Therefore, it follows from the relationship between nullity and rank of $\tilde{\Omega}$, $nul(\tilde{\Omega}) + rank(\tilde{\Omega}) = \tilde{n}$, that $rank(\tilde{\Omega}) = \tilde{n} - d - 1$.

The analysis for the scenario when two super stable tensegrity frameworks share more than $d+1$ vertices is similar to the aforementioned scenario. We omit it to avoid redundancy. This completes the proof of Theorem 4.8.

□

4.2.2 The number of shared vertices is less than $d+1$ in $\mathbb{R}^d$ $(d \in \{2,3\})$

The aim of this sub-section is to determine the minimum number of both new members and vertices incident to them when merging two super stable tensegrity frameworks in $\mathbb{R}^d$ $(d \in \{2,3\})$. We refer to this operation as *optimal merging*. Based on Theorem 4.8 and the HC discussed in Section 4.1, we present iterative procedures to merge two separate tensegrity frameworks.

Before describing the results, let us define $\mathcal{V}_{new}$ to denote a set of vertices satisfying $\mathcal{V}_{new} \subseteq \mathcal{V}_B \backslash \mathcal{V}_A$ and $|\mathcal{V}_{new}| = d + 1 - |\mathcal{V}_C| = n_{new}$. Let $\mathcal{E}_{new}$ be the set of members connecting the vertices in $\mathcal{V}_{new}$ to $(\mathcal{G}_A, q_A)$. We will indicate how $\mathcal{E}_{new}$ is obtained and determine $|\mathcal{E}_{new}|$ in the process. The situation is akin to linking to globally rigid formations with further edges to ensure the combined formation is globally rigid (see [133]). Then, as a direct extension of Theorem 4.8, we have the following Corollary.

Corollary 4.9. *Given two super stable tensegrity frameworks $(\mathcal{G}_A, q_A)$ and $(\mathcal{G}_B, q_B)$ in $\mathbb{R}^d$ $(d \in \{2,3\})$, satisfying $|\mathcal{V}_C| \leqslant d$, if the tensegrity framework $(\mathcal{G}'_A, q'_A)$ with $\mathcal{V}'_A = \mathcal{V}_A \cup \mathcal{V}_{new}$ and $\mathcal{E}'_A = \mathcal{E}_A \cup \mathcal{E}_{new}$ is super stable, in which vertices in $\mathcal{V}_{new}$ are in general position, then the tensegrity framework $(\tilde{\mathcal{G}}, \tilde{q})$ is super stable, where $\tilde{\mathcal{V}} = \mathcal{V}_A \cup \mathcal{V}_B$ and $\tilde{\mathcal{E}} = \mathcal{E}'_A \cup \mathcal{E}_B$.*

Illustrations of Corollary 4.9 are given in Figs. 4.2-4.4, where the merging operation is carried out in $\mathbb{R}^2$. In the plane, three scenarios are considered in terms of $|\mathcal{V}_c|$ as follows.

1. $|\mathcal{V}_C| = 0$.

   In this case, $n_{new} = 3 - |\mathcal{V}_C| = 3$.

   As Fig. 4.2 shows, to construct $(\mathcal{G}'_A, q'_A)$, we first add a new vertex u from $\mathcal{V}_B$ to $\mathcal{V}_A$ and three new members $(u,i), (u,j)$ and (u,k) by employing Theorem 4.1. Then applying Theorem 4.5, one adds the second new vertex v together with the corresponding members (v,i) and (v,j), noting there is already an explicit or implicit member (v,u). Consequently, the member (u,j) can be removed. Analogously, w and the member (w,i) are added in the last step, in which two explicit or implicit members (w,u) and (w,v) are considered.

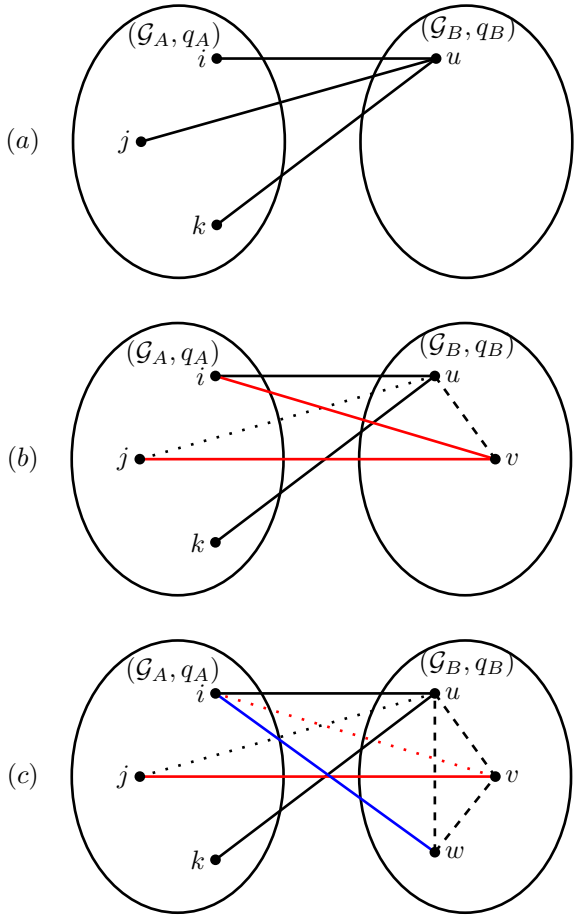


Figure 4.2: *Three steps of merging two super stable frameworks when $|\mathcal{V}_C| = 0$, where dashed lines and loosely dotted lines represent explicit or implicit members and removed members, respectively.*

Again from Theorem 4.5, the member (v, i) can be removed without losing super-stability. Hence, $\mathcal{E}_{new} = \{(u, i), (u, k), (v, j), (w, i)\}$, and thus $|\mathcal{E}_{new}| = 4$.

2. $|\mathcal{V}_C| = 1$.

   In this case, $n_{new} = 3 - |\mathcal{V}_C| = 2$.

   Vertex k is assumed to be common to $\mathcal{V}_A$ and $\mathcal{V}_B$. Based on Theorem 4.1 and 4.5, Fig. 4.3 shows that two new members, (u, i) and (v, j), are required to construct a super stable tensegrity framework. Hence, we know $|\mathcal{E}_{new}| = 2$.

3. $|\mathcal{V}_C| = 2$.

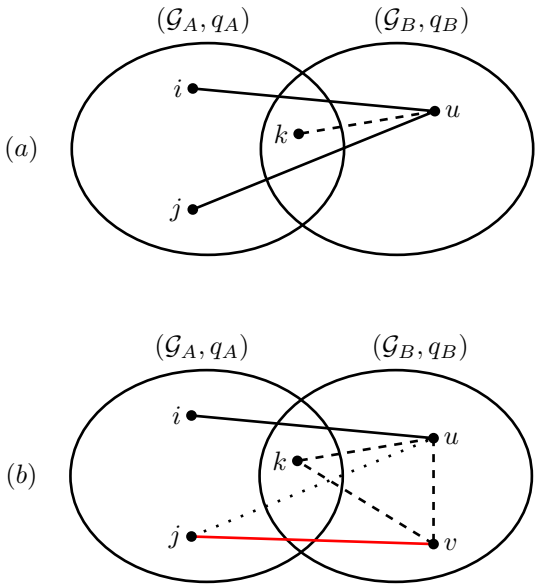


Figure 4.3: *Procedures of merging two super stable frameworks when $|\mathcal{V}_C| = 1$, where dashed lines and loosely dotted lines represent explicit or implicit members and removed members, respectively.*

In this case, $n_{new} = 3 - |\mathcal{V}_C| = 1$.

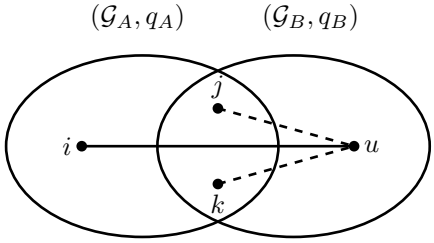


Figure 4.4: *Merging two super stable frameworks when $|\mathcal{V}_C| = 2$, where dashed lines represent explicit or implicit members.*

The common vertices are j and k. From Theorem 4.1, it can be checked that only one member is required to construct a super stable tensegrity framework as shown in Fig. 4.4, and thus $|\mathcal{E}_{new}| = 1$.

The results for structures defined in $\mathbb{R}^3$ are obtained similarly. Note that whether a new member is a cable or a strut is determined at each step of the addition process in accord with the procedure set out in the earlier section treating vertex addition and edge splitting. To sum up, the optimal merging of two super stable frameworks

is listed in Table 4.1 and 4.2.

Table 4.1: *Optimal merging of two super stable tensegrity frameworks in* $\mathbb{R}^2$.

$\lvert\mathcal{V}_C\rvert$	$\lvert\mathcal{E}_{new}\rvert$	$\lvert\mathcal{V}_{new}\rvert$
0	4	3
1	2	2
2	1	1
3 or more	0	0

Table 4.2: *Optimal merging of two super stable tensegrity frameworks in* $\mathbb{R}^3$.

$\lvert\mathcal{V}_C\rvert$	$\lvert\mathcal{E}_{new}\rvert$	$\lvert\mathcal{V}_{new}\rvert$
0	7	4
1	3	3
2	2	2
3	1	1
4 or more	0	0

The numbers contained in these tables are partially identical with those to be found in [133] for global rigidity. This is not completely surprising, given that super-stability is a specialized form of global rigidity.

4.3 Concluding remarks

In this chapter, we have addressed the problem of how to grow super stable tensegrity frameworks by adding a vertex or a super stable framework in $\mathbb{R}^d$, $(d \in \{2,3\})$. We have systematically developed the HC on tensegrity frameworks and a numerical method of calculating stress matrices associated with resultant tensegrity frameworks. In addition, in the case of merging two super stable tensegrity frameworks in $\mathbb{R}^d$, we have shown that super-stability can be maintained if the frameworks share no fewer than $d+1$ vertices in general positions. Finally, to cover all the possible scenarios of merging in $\mathbb{R}^d$, $(d \in \{2,3\})$, we have presented the detailed steps of optimal merging. The results have been summarized in two tables.

Chapter 5

Constructing universally rigid tensegrity frameworks with application in multi-agent formation control

Rigidity graph theory has found broad applications in engineering, architecture, biology, and chemistry, while systematic and computationally tractable construction of rigid frameworks is still a challenging task. In this chapter, starting from any given configuration in general positions, we show how to construct a universally rigid tensegrity framework by looking into the kernel of the tensegrity framework's stress matrix. As one application, we show how to stabilize a formation of mobile agents by assigning a universally rigid virtual tensegrity framework for the formation and then design distributed controllers based on the forces determined by the stresses of the edges. Such formation controllers are especially useful when one needs to satisfy formation constraints in the form of strict upper or lower bounds on inter-agent distances arising from tethered robots

5.1 Introduction

Rigidity graph theory has always been playing a key role in solving topology related problems in various fields, e.g. the formation control problem of multi-agent systems [15, 35, 78], the geometric analysis of molecular models in bio-chemistry [128], and the localization of wireless sensor networks [139]. A challenging problem concerned with rigidity is to determine whether a given framework is rigid (resp. globally rigid and universally rigid). It is already well known that the local rigidity can be guaranteed if the rank of the corresponding rigidity matrix exceeds some bound. However, for global rigidity and universal rigidity, the problem is shown to be NP-hard even in the lower dimensional space $\mathbb{E}^1$ [108]. Then, to make the problem more tractable, researchers concentrate on the frameworks with *a generic configuration*, where the nodal coordinates are algebraically independent over the rationals. Some exploring efforts along this line have been made in [1, 20, 22], where the sufficient conditions for a framework of generic configurations to be globally/universally rigid have been given, and the necessity of these conditions is later proved in [49]. Recently, a new kind of universal rigidity called 'iterative

universal rigidity' has been addressed in [24], where the analysis of the rigidity of the framework is decomposed into a sequence of affine sets in the configuration space. One weaker condition compared with being generic for the configuration of the framework is *general*, which is preferable in practice since it is checkable in polynomial time, while not for the generic configuration [4]. It has been verified that the genericity assumption of the configuration to ensure universal rigidity can be relaxed to the situation of general positions in [3]. In addition, the universal rigidity of one-dimensional framework in general positions with a complete bipartite underlying graph is fully discussed in [63].

Of particular theoretical and practical interest is tensegrity frameworks. In recent years, they have found broad applications in engineering, architecture, biology, and arts because of their superior features, such as deployability, deformability and robustness [76, 123]. However, little research has been conducted to construct universally rigid tensegrity frameworks from some given geometric shape described by the overall configuration. One of the well-established findings is that for a framework in the shape of convex polygons in the plane, the universal rigidity can be obtained if the boundary and the interior members are respectively set to be cables and struts [19]. For a class of so-called Grünbaum frameworks, the construction for universally rigid tensegrity frameworks in two and three dimensions is studied in [66], in which the approach strongly relies on the computation of the convex hull. If the underlying graph of the framework is given beforehand, a purification based algorithm is designed to compute the corresponding stresses for the $(d+1)$-lateration frameworks in general positions [4]. In the case that only the configuration is known, to create a universally rigid tensegrity framework, the underlying graph together with the type of members are determined through the algorithm in [88], while it is highly likely to result in a complete graph.

For control engineers, the issue of identifying and designing rigid frameworks are particularly relevant for the formation control problem of teams of mobile agents. The stable tensegrity frameworks, due to their strong robustness and clear physical interpretation, have been used as the virtual multi-agent framework to understand better the behavior of the agents around the equilibrium corresponding to the prescribed formation shape. Tensegrity frameworks, due to their robustness and scalability, have been employed as the virtual framework to solve the formation control problem of multi-agent systems. Starting from one-dimensional space, i.e., a line, [93] introduced a tensegrity-based control law that can exponentially stabilize the agents with prescribed distances. Then the same idea was used to deal with the problem in higher-dimensional space by collinear projections. In [72], the model of an unmanned aerial vehicle was integrated with a virtual cross-tensegrity framework, based on which a decentralized control strategy was designed such that a scalable formation was achieved. As a direct application, a stress-based formation control scheme is proposed to stabilize "affine" formations in [78]. In contrast

to a rigid formation, an affine formation allows more transformations besides translation and rotation, such as scaling, shearing and reflection. But most of the existing results are restricted to special class of frameworks, e.g., one-dimensional tensegrity framework [93] or *cross-tensgrities* that are rectangles with four boarder cables and two crossing struts [71, 72].

Motivated by the recent advances in rigidity graph theory, it is the goal of this chapter to first design an algorithm to construct a universally rigid tensegrity framework for any given configurations in general positions. Taking into account the engineering and theoretical concerns that sparse frameworks are more desirable because of extendibility, complexity and computation, we will focus on building tensegrity frameworks with fewer members through constructing stress matrices with close to maximal allowed number of zeros. We then use the constructed universally rigid tensegrity framework for multi-agent formation control. Distributed control laws are designed to stabilize formations when their topologies are implemented using the virtual tensegrity framework. So the main contribution of this chapter is two-fold. First, we develop a systematic algorithm to construct universally rigid tensegrity frameworks. Such algorithms have not been reported in the literature. Second, we apply the virtual tensegrity frameworks to distributed formation control, which therefore enables us to have a clear intuitive estimate of the domain of attraction around the system's equilibrium. This application is particularly useful for the emerging cooperative control of tethered robots, where the challenging formation constraints, such as the strict maximum or minimum inter-agent distances, could be incorporated in the virtual tensegrity framework.

The rest of this chapter is organized as follows. In Section 5.2, we propose our algorithm to construct universally rigid tensegrity frameworks for any given configuration in general position. The application for formation control is discussed in Section 5.3. Some illustrative examples are presented in Section 5.4. Section 5.5 concludes this chapter.

5.2 Constructing universally rigid tensegrity frameworks

We first provide the steps of the algorithm in detail and then show the constructed tensegrity frameworks are in general close to minimal by giving an upper bound of the numbers of their members.

5.2.1 Algorithm

We assume the given configuration $q^* \in \mathbb{R}^{d\times n}$ is in general position. Define the extended configuration matrix $Q^* \triangleq [(q^*)^T, \mathbf{1}_n]^T \in \mathbb{R}^{(d+1)\times n}$. Then, it follows that

every $(d+1)\times(d+1)$ submatrix of Q^* has full rank. From the definition of the stress matrix Ω and (2.5), one can check that matrix Ω always lives in the null space of Q^*, i.e.

$$Q^*\Omega = \mathbf{0}_{(d+1)\times n}. \tag{5.1}$$

Given q^*, the key component of the algorithm is to determine matrix Ω, which determines in turn which two nodes are connected together with their nonzero stress. Obviously, such Ω is in general not unique and naturally we want to obtain an Ω with more zeros which leads to fewer members and thus lower complexity. Towards this end, we convert our problem into the *sparse null space problem* first considered in [96], namely, given a matrix A, to find a sparse matrix B such that B is full rank and its column span is null(A) [51].

In view of Lemmas 2.9, to obtain a universally rigid tensegrity framework, matrix Ω is required to be positive semi-definite with rank $n-d-1$. However, since Ω in (5.1) is not full rank, we cannot directly solve the sparse null space problem. Instead, we try to construct a column full-rank matrix $D\in\mathbb{R}^{n\times(n-d-1)}$ such that

$$Q^*D = \mathbf{0}_{(d+1)\times(n-d-1)}, \tag{5.2}$$

where D is a *Gale matrix* of q^* [3]. If indeed such a D can be constructed, it must be true that

$$Q^*DD^T = \mathbf{0}_{(d+1)\times(n-d-1)}D^T = \mathbf{0}_{(d+1)\times n}, \tag{5.3}$$

and hence the matrix DD^T can serve as the stress matrix Ω. So the construction of an Ω is equivalent to the design of such a sparse D. In addition, for computational efficiency, we make an even stronger requirement that Ω is in its *band form*, whose non-zero entries are confined to be in a diagonal band containing the main diagonal. Now we present our 5-step algorithm to construct the universally rigid framework with the stress matrix Ω, which is inspired by the classical "turning back" method for computing the sparse null space basis [48].

Step 1: Arrange matrix

$$Q^* = [(q^*)^T, \mathbf{1}_n]^T \in \mathbb{R}^{(d+1)\times n}.$$

Step 2: Locate the nonzero elements of D. We first find the smallest $k_1>0$ such that Q^*'s columns with the indices $d+2, d+1, \ldots, d+2-k_1$ are linearly dependent. We then set the nonzero elements of D's first column to be located at the positions $d+2-k_1$ through $d+2$. Then, in order to record the positions of the nonzero elements of the second column of D, we find the smallest $k_2>0$ such that those columns with the indices $d+3, d+2, \ldots, d+3-k_2$, excluding $d+2-k_1$, of Q^* are linearly dependent. Again, the indices correspond to the nonzero elements' positions of D's second column. We repeat this procedure until we have determined the positions of the nonzero elements of the last column of D. Note that the configuration q^* is in general position, there are

exactly $d+2$ nonzero elements in each column of D, yielding

$$D = \begin{array}{c} \begin{array}{cccc} 1 & 2 & \cdots & n-d-1 \end{array} \\ \begin{pmatrix} * & 0 & \cdots & 0 \\ * & * & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ * & * & \cdots & 0 \\ 0 & * & \cdots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & * \end{pmatrix} \end{array} \begin{array}{c} 1 \\ 2 \\ \vdots \\ d+2 \\ d+3 \\ \vdots \\ n \end{array} \tag{5.4}$$

where '$*$' indicates a nonzero element.

Step 3: Now we compute the values of the nonzero elements of D by solving the following equation

$$Q^* D = \mathbf{0}_{(d+1)\times(n-d-1)}, \tag{5.5}$$

which is underdetermined since it is a set of $d+1$ linear equations with $d+2$ unknowns. Hence, we can always find a set of nonzero elements of D and thus fully determine D. In addition, it is easy to check that the constructed D is always column full-rank.

Step 4: The stress matrix is then the positive semidefinite matrix $\Omega = DD^T$ whose rank is $n-d-1$. The obtained Ω is a square, symmetric, band matrix with one upper triangular and one lower triangular $(n-d-2)$-dimensional submatrix in its upper right and lower left corners respectively. We write Ω in the notation below where all the determined zero elements are denoted by zero and all the other elements that may or may not be zero are denoted by '$\times$':

$$\Omega = \begin{bmatrix} \times & \cdots & \times & 0 & \cdots & \cdots & 0 \\ \times & \cdots & \cdots & \times & \ddots & \ddots & \vdots \\ \vdots & & & \vdots & & \ddots & \vdots \\ \times & \cdots & \cdots & \times & \cdots & \times & 0 \\ \times & \cdots & \cdots & \times & \cdots & \cdots & \times \\ \vdots & & & \vdots & & & \vdots \\ 0 & \times & \cdots & \times & \cdots & \cdots & \times \\ \vdots & \ddots & & \vdots & & & \vdots \\ \vdots & \ddots & \ddots & \times & \cdots & \cdots & \times \\ 0 & \cdots & \cdots & 0 & \times & \cdots & \times \end{bmatrix}_{n\times n} \tag{5.6}$$

Step 5: According to Ω, assign cables and struts to the n-node framework: For each $\Omega_{ij} < 0$, we assign a cable between nodes i and j and for each $\Omega_{ii} > 0$, we assign a strut between nodes i and j. The stresses of the assigned cables and struts are $\omega_{ij} = -\Omega_{ii}$. The desired tensegrity framework is then obtained.

Now we prove that the constructed tensegrity framework is indeed universally rigid.

Theorem 5.1. *Given a configuration q^* in general position, the proposed 5-step algorithm returns a universally rigid tensegrity framework with the geometric shape prescribed by q^*.*

Proof. The constructed tensegrity framework has the stress matrix Ω which, as shown in step 4, is positive semi-definite and has rank $n-d-1$. So the conditions (1) in Lemma 2.9 is satisfied. Recalling that the given configuration q^* is in general position, so the obtained tensegrity framework is again universally rigid because of Lemma 2.9. □

Remark 5.2. In step 4, the stress matrix Ω can be more generally determined by $\Omega = D\Psi D^T$, where Ψ is a nonsingular, $(n-d-1)\times(n-d-1)$, symmetric matrix, used to adjust the magnitudes of the stresses of the members of the constructed tensegrity framework. In this chapter, we set $\Psi = I_{n-d-1}$ for simplicity.

In practice, one usually prefers fewer edges to reduce the complexity, if possible, in a rigid framework. Hence, a natural question to ask is whether the tensegrity framework obtained by the proposed algorithm is indeed structurally simple. To address this question, we construct an upper bound of the number of members of the obtained tensegrity framework, namely $|\mathcal{E}|$ of $(\mathcal{G}, q^*)$, which is roughly nd when n is big.

5.2.2 Upper bound of $|\mathcal{E}|$

We look into the number of nonzero elements of Ω in (5.6). The densest Ω, namely that contains the largest possible number of nonzero elements, appears when all the elements denoted by '$\times$' are nonzero. Then the number of off-diagonal zero elements in $\Omega = DD^T$ is

$$2((n-d-2)+(n-d-3)+\cdots+1) = (n-d-1)(n-d-2),$$

or equivalently the number of off-diagonal nonzero elements in Ω is

$$n(n-1)-(n-d-1)(n-d-2) = (d+1)(2n-d-2),$$

which, when divided by 2, is exactly the number of members to be inserted into the tensegrity framework in view of Step 5 of the algorithm. Therefore, this densest Ω gives an upper bound $(d+1)\left(n-\frac{d+2}{2}\right)$.

So we have proved the following theorem.

Theorem 5.3. *The number of members of the constructed tensegrity framework is bounded from above by*

$$|\mathcal{E}| \leqslant (d+1)\left(n-\frac{d+2}{2}\right). \tag{5.7}$$

Remark 5.4. One necessary condition for a tensegrity framework in generic configurations (with n larger than $d+1$) to be globally rigid or universally rigid is that it has to be rigid with at least one self-stress state, which in turn implies that it needs to have at least $nd-d(d+1)/2+1$ members. This is the lower bound of the number of members required to construct a globally rigid or universally rigid tensegrity framework in generic configurations. The lower bound differs from the constructed upper bounded roughly by n when n is big.

Remark 5.5. The upper bound of the number of members presented in (5.7) corresponds to the number of members of universally rigid tensegrity frameworks constructed on $(d+1)$-tree graphs [4]. The construction of universally rigid Grünbaum frameworks in generic configurations with the minimal number of edges in 2D and 3D are investigated in [66], where it is shown that the two-dimensional Grünbaum frameworks in nongeneric configurations are also universally rigid. The problem on how to compute the stress matrix has also been considered in [88], which, however, most likely yields stress matrices without any zero elements. In comparison, our algorithm in general always returns a stress matrix with close to maximal allowed number of zeros.

In the next section, we show how universally rigid tensegrity frameworks can be used as virtual structures to help the design of distributed formation controllers for teams of mobile agents.

5.3 Formation stabilization

5.3.1 Formation Control Problem

We consider a group of n mobile agents, each of which is modeled by a kinematic point

$$\dot{q}_i = u_i, \quad i = 1, 2, \cdots, n, \tag{5.8}$$

where $q_i \in \mathbb{R}^d$ represents agent i's position and $u_i \in \mathbb{R}^d$ is its control input. The neighbor relationships will be designed and characterized by an undirected graph

$\mathcal{G}$ with the vertex set $\mathcal{V}$ and the edge set $\mathcal{E}$.

The *formation stabilization problem* is that given a desired configuration $q^* = [q_1^*, \cdots, q_n^*] \in \mathbb{R}^{d\times n}$ for this team of n agents (5.8), design the neighbor relationship graph $\mathcal{G}$ for the team and correspondingly, for each agent $i = 1, \ldots, n$, design distributed control laws $u_i(q_i - q_j, q_i^* - q_j^*)$, $j \in \mathcal{N}_i$, such that the agents' positions are driven to the target set

$$\mathcal{T} = \{q \in \mathbb{R}^{dn} |\, q_i - q_j = q_i^* - q_j^*, \quad \forall (i,j) \in \mathcal{E}\}. \tag{5.9}$$

Note that here q^* is given in an arbitrary coordinate system of choice. When each agent has its own coordinate system that may differ from each other, q^* is then given to each agent in its own coordinate system, in which case different agents' given q^* differ up to some congruent transformation of translation and rotation.

Ample previous work has discussed how to solve this problem locally when $(\mathcal{G}, q^*)$ is rigid and the controllers are derived by attaching virtual springs to the agents. To make the controllers simpler, $\mathcal{G}$ is usually required to contain as few edges as possible. In what follows, we use the virtual tensegrity framework constructed in Section 5.2 to describe the necessary sensings between the agents and the resulted distributed controllers whose gains are derived from the stresses.

5.3.2 Controller design and stability analysis

We use the universally rigid tensegrity framework constructed previously to determine which agents need to sense which other agents. To be specific, from the given q^*, we run the 5-step algorithm and obtain a framework $(\mathcal{G}, q^*)$. Then the underlying graph $\mathcal{G}$ is used to represent the sensing graph of the n-agent team.

To calculate the virtual forces utilizing the stresses, we set the rest length of each edge to be

$$l_{ij} = \gamma_{ij}\|r_{ij}^*\| = \begin{cases} \gamma_{ij}^c \|r_{ij}^*\| & \text{if } \omega_{ij} > 0, \\ \gamma_{ij}^s \|r_{ij}^*\| & \text{if } \omega_{ij} < 0, \end{cases} \tag{5.10}$$

where $\gamma_{ij}^c \in (0,1)$ and $\gamma_{ij}^s \in (1, +\infty)$ are constants as design parameters, and r_{ij}^* is the prescribed relative position of agent j with respect to agent i, i.e., $r_{ij}^* = q_j^* - q_i^*$. Then the formation control gain k_{ij} for agent i with respect to its neighbor j is given by

$$k_{ij} = \frac{\omega_{ij}}{1 - \gamma_{ij}}, \quad \forall (i,j) \in \mathcal{E}. \tag{5.11}$$

For convenience in notation, we define the auxiliary variable

$$z_i = q_i - q_i^*. \tag{5.12}$$

Let $z \in \mathbb{R}^{dn}$ be the vector obtained by stacking all the z_i together.

Now, we define the set $\mathcal{D}(0)$ as

$$\mathcal{D}(0) \triangleq \{z(0) \in \mathbb{R}^{nd} | \|z_i(0) - z_j(0)\| < \text{sign}(\omega_{ij})(\|r_{ij}^*\| - l_{ij}), \\ \forall (i,j) \in \mathcal{E}\}. \tag{5.13}$$

To proceed, we assume all the agents are initially located in the set $\mathcal{D}^\epsilon(0)$ defined as

$$\mathcal{D}^\epsilon(0) \triangleq \{z(0) \in \mathbb{R}^{nd} | \|z_i(0) - z_j(0)\| \leqslant \text{sign}(\omega_{ij})(\|r_{ij}^*\| - l_{ij}) - \epsilon, \\ \forall (i,j) \in \mathcal{E}\}, \tag{5.14}$$

where ϵ is a small positive number.

The potential function for agent i is defined as

$$P_i(z(t)) \triangleq \sum_{\omega_{ij}<0} \frac{k_{ij}\left[(l_{ij} - \|r_{ij}^*\|)^\alpha - \rho_{ij}^\alpha\right]^\beta}{\rho_{ij}} \\ + \sum_{\omega_{ij}>0} \frac{k_{ij}\left[(\|r_{ij}^*\| - l_{ij})^\alpha - \varrho_{ij}^\alpha\right]^\beta}{\varrho_{ij}}, \tag{5.15}$$

where

$$\rho_{ij} = l_{ij} - \|r_{ij}^*\| - \|r_{ij}(t) - r_{ij}^*\|, \\ \varrho_{ij} = \|r_{ij}^*\| - l_{ij} - \|r_{ij}(t) - r_{ij}^*\|, \tag{5.16}$$

with $r_{ij} = q_j - q_i$ being the relative position between agents i and j. $\alpha > 2$ and $0 < \beta < 1$ are the positive exponents. Note that in the set $\mathcal{D}^\epsilon(0)$, ρ_{ij} and ϱ_{ij} are both small positive numbers. The parameters k_{ij} are chosen such that

$$P_0 = \begin{cases} k_{ij}(\|r_{ij}^*\| - l_{ij})^{\alpha\beta}, & \text{if } \omega_{ij} > 0, \\ k_{ij}(l_{ij} - \|r_{ij}^*\|)^{\alpha\beta}, & \text{if } \omega_{ij} < 0, \end{cases} \tag{5.17}$$

where P_0 is an arbitrary positive pre-defined constant. All the parameters k_{ij} would be determined if any one of them is decided with given α and β.

Correspondingly, the potential function for the whole system is given by

$$P(z(t)) = \frac{1}{2}\sum_{i=1}^{n} P_i(z(t)) \tag{5.18}$$

The control input of agent i is

$$\dot{q}_i = \dot{z}_i = u_i = -\nabla_{z_i} P_i(z(t)) \tag{5.19}$$

Proposition 5.6. *For any given initial position $q(0) \in \mathcal{D}^{\epsilon}(0)$, the set $\mathcal{D}(0)$ is invariant for the system* (5.8) *under the control law* (5.19).

Before presenting the proof for Proposition 5.6, we first analyse the properties of the potential function (5.15). Consider the numerators of (5.15), where δ_s (resp. δ_c) are regarded as independent variables. Define function $f(x)$ as

$$f(x) = k_f(c_f^{\alpha} - x^{\alpha})^{\beta} > 0, \quad k_f > 0, x \in (0, c_f], \tag{5.20}$$

which coincides with the form of the numerators of the potential function. The first order derivative of $f(x)$ satisfies

$$f'(x) = -\alpha\beta k_f x^{\alpha-1}(c_f^{\alpha} - x^{\alpha})^{\beta} < 0,\ x \in (0, c_f], \tag{5.21}$$

which implies that the function $f(x)$ is monotonically decreasing with respect to x. Besides this, for sufficiently small x_s, $f'(x_s) \to 0$, due to $x_s^{\alpha-1} \to 0$, when $\alpha > 2$.

Look at the second order derivative of $f(x)$,

$$\begin{aligned} f''(x) = &-\alpha(\alpha-1)\beta k_f x^{\alpha-2}(c_f^{\alpha} - x^{\alpha})^{\beta-1} \\ &+ \alpha^2\beta(\beta-1)k_f x^{2(\alpha-1)}(c_f^{\alpha} - x^{\alpha})^{\beta-2}. \end{aligned} \tag{5.22}$$

It follows from $\alpha > 2$ and $0 < \beta < 1$ that $f'' < 0$, $\forall x \in (0, c_f]$. Together with the fact that $f'(x) < 0$, it implies that the decreasing rate of $f(x)$ increases as x grows.

Consider

$$f(x) = f(0) + \int_0^x f'(s)ds = f(0) - \int_0^x |f'(s)|ds, \quad x \in (0, c_f]. \tag{5.23}$$

It can be seen that $f'(x)$ drops rapidly, when the value of x increases.

Based on $f(x)$, together with (5.15), we introduce the function $P_f(x)$ as

$$P_f(x) = \frac{f(x)}{x}, \quad x \in (0, c_f]. \tag{5.24}$$

The first order derivatives of $P_f(x)$ satisfies

$$P_f'(x) < 0. \tag{5.25}$$

It can be seen that $P_f(x)$ has the same monotonicity with the potential function in (5.15) for a single edge. Hence, the potential energy (5.15) is monotonically decreasing with respect to δ_s and δ_c. This property implies that the closer the edges approach their rest lengths, the higher their potential energy will become.

Now, we give the proof of Proposition 5.6, which is partially motivated by [62].

Proof of Proposition 5.6. Recall the control input (5.19) for agent i

$$u_i = -\nabla_{z_i} P_i(z(t)). \tag{5.26}$$

Hence the time derivative of $P_i(t)$ satisfies,

$$\begin{aligned} \dot{P}_i(z(t)) &= \nabla_{z_i} P_i(z(t))^T \dot{z}_i(t) \\ &= -\left[\nabla_{z_i} P_i(z(t))\right]^T \left[\nabla_{z_i} P_i(z(t))\right] \leqslant 0, \end{aligned} \tag{5.27}$$

which implies that the potential energy associated with agent i is not increasing. However, the potential energy stored in some edge, say (i,ν) with negative stress, might increase, even though $P_i(z(t))$ is not increasing. This means that the length of (i,ν) tends to its rest length $l_{i\nu}$ and the other edges are driven far away from their rest lengths. Since the potential function for every edge defined in (5.15) is monotonically decreasing, the potential energy of edge (i,ν) will increase when $\|r_{i\nu}\|$ approaches to its rest length, i.e., $l_{i\nu}$. Meanwhile, since the energy generated by the single edge (i,ν) is less than that generated by all the edges that connecting agent i, by letting $P_i(z(t)) = P_{i\nu}(t)$, we can get the maximum length for edge (i,ν). Then, it can be checked whether $\|r_{i\nu}\|$ reaches its rest length $l_{i\nu}$.

Now, we consider the length changes of the edges. Since the potential energy $P_i(z(t))$ is not increasing, we have

$$P_{i\nu}(t) + \left[P_i(z(t)) - P_{i\nu}(t)\right] \leqslant P_i(z(0)), \quad t \geqslant 0, \tag{5.28}$$

where $P_{i\nu}(t)$ denotes the potential energy of edge (i,ν) stored at time t. $P_i(z(t))$ is the potential energy of all the neighbor edges of agent i. Obviously, we have $P_i(z(t)) > P_{i\nu}(t)$. Then, from (5.28), it is straightforward to check

$$P_{i\nu}(t) \leqslant P_i(z(0)) - \left[P_i(z(t)) - P_{i\nu}(t)\right] < P_i(z(0)). \tag{5.29}$$

Hence, to obtain the upper bound of $\|r_{i\nu}\|$, it is assumed that

$$P_{i\nu}(t) = P_i(z(0)). \tag{5.30}$$

According to (5.25), the potential energy $P_i(z(t))$ would increase if the length of the edges reach closer to their rest lengths. Hence, the maximum of $P_i(z(0))$ is obtained when the initial positions of the edges satisfy

$$\|z_i(0) - z_j(0)\| = \text{sign}(\omega_{ij})(\|r_{ij}^*\| - l_{ij}) - \epsilon, \quad \forall (i,j) \in \mathcal{E}. \tag{5.31}$$

Consequently,

$$P_i(z(0))_{max} = \sum_{\omega_{ij}<0} \frac{k_{ij}\left[(l_{ij} - \|r_{ij}^*\|)^\alpha - \epsilon^\alpha\right]^\beta}{\epsilon} + \sum_{\omega_{ij}>0} \frac{k_{ij}\left[(\|r_{ij}^*\| - l_{ij})^\alpha - \epsilon^\alpha\right]^\beta}{\epsilon}. \tag{5.32}$$

Then, we let

$$P_{i\nu} = \frac{k_{i\nu}\left[(l_{i\nu} - \|r_{i\nu}^*\|)^\alpha - \rho_{i\nu}^\alpha\right]^\beta}{l_{i\nu} - \|r_{i\nu}^*\| - \|r_{i\nu} - r_{i\nu}^*\|} = P_i(z(0))_{max}. \tag{5.33}$$

So far, the $\|r_{i\nu}\|$ can be derived from (5.32)-(5.33) theoretically. However, it can be seen that the resulting $\|r_{i\nu}\|$ depends on the other edges' rest lengths and desired lengths, which is difficult to determine whether edge (i,ν) reaches its rest length or not. Therefore, we further look for another relationship on the potential energy between the edge (i,ν) and the other edges.

According to the statement above, to check whether $\|r_{i\nu}\|$ reaches the rest length, only the potential energy of edge (i,ν) increases, i.e., only the edge (i,ν) changes towards its rest length, and the other edges move away from their rest lengths. This implies

$$0 < \rho_{i\nu}(t) \triangleq \epsilon_1 < \epsilon, \tag{5.34}$$

where ϵ_1 is a sufficiently small number. Now, we only take the numerators of (5.15) into consideration.

For two different edges, consider two functions

$$f_1(x) = k_{f1}(c_{f1}^\alpha - x^\alpha)^\beta, \quad k_{f1} > 0,\ x \in (0,\epsilon], \tag{5.35}$$

and

$$f_2(x) = k_{f2}(c_{f2}^\alpha - x^\alpha)^\beta, \quad k_{f2} > 0,\ x \in (0,\epsilon]. \tag{5.36}$$

Given sufficiently small number ϵ_1, we have

$$f_1(\epsilon_1) = f_1(0) + \int_0^{\epsilon_1} f_1'(x)dx. \tag{5.37}$$

Since the function $f_1(x)$ is continuous and differentiable in $(0,\epsilon)$, it follows from mean value theorem that

$$f_1(\epsilon_1) = f_1(0) + \epsilon_1 f_1'(s_1), \quad s_1 \in (0,\epsilon_1). \tag{5.38}$$

Similarly, for $\epsilon > \epsilon_1$, it follows

$$\begin{aligned} f_2(\epsilon) &= f_2(0) + \int_0^{\epsilon} f_2'(x)dx \\ &= f_2(0) + \int_0^{\epsilon_1} f_2'(x)dx + \int_{\epsilon_1}^{\epsilon} f_2'(x)dx \\ &= f_2(0) + \epsilon_1 f_2'(s_2) + (\epsilon - \epsilon_1) f_2'(s_3), \end{aligned} \tag{5.39}$$

where $s_2 \in (0, \epsilon_1)$ and $s_3 \in (\epsilon_1, \epsilon)$.

From (5.11), the initial values satisfy

$$f_1(0) = f_2(0) = P_0. \tag{5.40}$$

And, from (5.21), for a sufficiently small ϵ_1, under the condition that $\alpha > 2$, we have

$$f_1'(s_1) = f_2'(s_2) \to 0. \tag{5.41}$$

Considering the fact that $f''(x) < 0$ in (5.22), it yields

$$f_2'(s_3) < f_2'(s_2) < 0. \tag{5.42}$$

Combining (5.38)-(5.42), we obtain

$$f_2(\epsilon) < f_1(\epsilon_1), \tag{5.43}$$

which implies

$$f_{ij}(\epsilon) < f_{i\nu}(\rho_{i\nu}), \quad j \in \mathcal{N}_i. \tag{5.44}$$

In view of (5.32)-(5.34), we know that the edge (i, ν) satisfies

$$\begin{aligned} & k_{ij} \left[\left(\text{sign}(\omega_{ij})(\|r_{ij}^*\| - l_{ij}) \right)^{\alpha} - \epsilon^{\alpha} \right]^{\beta} \\ < & k_{i\nu} \left[(l_{i\nu} - \|r_{i\nu}^*\|)^{\alpha} - \rho_{i\nu}(t)^{\alpha} \right]^{\beta}, \quad j \in \mathcal{N}_i. \end{aligned} \tag{5.45}$$

Taking (5.32)-(5.33) and (5.45) into consideration, we have

$$\frac{1}{l_{i\nu} - \|r_{i\nu}^*\| - \|r_{i\nu}(t) - r_{i\nu}^*\|} < \frac{|\mathcal{N}_i|}{\epsilon}, \tag{5.46}$$

where $|\mathcal{N}_i|$ denotes the cardinality of the set $\mathcal{N}_i$, i.e., the number of agent i's neighbors. Then, through simple calculation, we get

$$\|r_{i\nu}(t) - r_{i\nu}^*\| \leqslant l_{i\nu} - \|r_{i\nu}^*\| - \frac{\epsilon}{|\mathcal{N}_i|}. \tag{5.47}$$

Note that $z_{i\nu}(t)$ satisfies

$$\|z_{i\nu}(t)\| = \|r_{i\nu}(t) - r^*_{i\nu}\| \geqslant \|r_{i\nu}(t)\| - \|r^*_{i\nu}\|. \tag{5.48}$$

Then, the upper bound for $\|r_{i\nu}(t)\|$ can be derived from (5.47)-(5.48)

$$\|r_{i\nu}(t)\| \leqslant l_{i\nu} - \frac{\epsilon}{|\mathcal{N}_i|}, \tag{5.49}$$

which implies the length of edge (i,ν) will not reach its rest length if it starts from $\mathcal{D}^\epsilon(0)$.

For the edge (i,ζ) with positive stress, we can prove similarly that

$$\|r_{i\zeta}(t)\| \geqslant l_{i\zeta} + \frac{\epsilon}{|\mathcal{N}_i|}. \tag{5.50}$$

Therefore, we can draw the conclusion from (5.49) and (5.50) that the edges will never escape from the set $\mathcal{D}(0)$ during the evolution, if the edges are initially located in $\mathcal{D}^\epsilon(0)$, namely, the set $\mathcal{D}(0)$ is an invariant set. This completes the proof. □

Remark 5.7. Proposition 5.6 indicates that using the control law (5.19) the agents will not escape the set $\mathcal{D}(0)$ if they start from $\mathcal{D}^\epsilon(0)$, which implies there will always be virtual forces along the edges. This further implies no sensing breakdown will happen since the virtual edges represent the sensing relationships between the agents.

Now, we are ready to present our main theorem.

Theorem 5.8. *For any given initial position $q(0) \in \mathcal{D}^\epsilon(0)$, the formation stabilization of the networked single-integrator systems modeled by* (5.8) *is achieved by the controller* (5.19).

Proof. From the control law (5.19) and potential function (5.15), we have

$$u_i = -\left(\sum_{\omega_{ij}<0} \frac{\phi_{ij}}{\rho_{ij}^2\|r_{ij} - r^*_{ij}\|} + \sum_{\omega_{ij}>0} \frac{\varphi_{ij}}{\varrho_{ij}^2\|r_{ij} - r^*_{ij}\|}\right)(z_i - z_j), \tag{5.51}$$

where

$$\phi_{ij} = k_{ij}\Big(\alpha\beta\rho_{ij}^\alpha\left[(l_{ij} - \|r^*_{ij}\|)^\alpha - \rho_{ij}^\alpha\right]^{\beta-1} + \left[(l_{ij} - \|r^*_{ij}\|)^\alpha - \rho_{ij}^\alpha\right]^\beta\Big),$$
$$\varphi_{ij} = k_{ij}\Big(\alpha\beta\varrho_{ij}^\alpha\left[(\|r^*_{ij}\| - l_{ij})^\alpha - \varrho_{ij}^\alpha\right]^{\beta-1} + \left[(\|r^*_{ij}\| - l_{ij})^\alpha - \varrho_{ij}^\alpha\right]^\beta\Big),$$

with ρ_{ij} and ϱ_{ij} being defined in (6.4). We further write the closed-loop system

(5.8) and (5.15) into their compact form

$$\dot{\bar{q}} = \dot{z} = -(\mathcal{L}_w(l, \bar{q}, \bar{q}^*) \otimes I_d)z, \tag{5.52}$$

where $\bar{q} = [q_1^T, \cdots, q_n^T]^T \in \mathbb{R}^{dn}$, $\bar{q}^* = [(q_1^*)^T, \cdots, (q_n^*)^T]^T \in \mathbb{R}^{dn}$, and $\mathcal{L}_w = H^T W(l, \bar{q}, \bar{q}^*)H$ is $\mathcal{G}$'s *weighted Laplacian matrix*. H is $\mathcal{G}$'s incidence matrix defined in (2.2.1), and the diagonal weight matrix $W(l, \bar{q}, \bar{q}^*)$ is

$$\begin{aligned} W(l, \bar{q}, \bar{q}^*) &= \mathrm{diag}(\{)w_\iota\}, \quad \iota = 1, 2, \cdots, |\mathcal{E}|, \text{ with} \\ w_\iota &= \begin{cases} \dfrac{\phi_\iota}{\rho_\iota^2 \|r_\iota - r_\iota^*\|}, & \text{if } \omega_\iota < 0, \\ \dfrac{\varphi_\iota}{\varrho_\iota^2 \|r_\iota - r_\iota^*\|}, & \text{if } \omega_\iota > 0. \end{cases} \end{aligned} \tag{5.53}$$

Given $q(0) \in \mathcal{D}^\epsilon(0)$, it follows from Proposition 5.6 that $\mathcal{L}_w$ in (5.52) is well defined and positive semi-definite [62]. Also, the interaction graph of the agents is connected due to the fact that the edges are always in tension or in compression. Then, it can be concluded from Lemma 2.1 that

$$\lim_{t\to\infty} z_1(t) = \lim_{t\to\infty} z_2(t) = \cdots = \lim_{t\to\infty} z_n(t), \tag{5.54}$$

which implies that

$$\lim_{t\to\infty} r_{ij}(t) = r_{ij}^*, \quad \forall (i,j) \in \mathcal{E}. \tag{5.55}$$

Thus, the prescribed formation is achieved. This completes the proof. □

Remark 5.9. Even though the result is in the sense of local stability, we can enlarge the stability region by choosing small γ_{ij}^c for edges with positive stress and large γ_{ij}^s for those with negative stress. The definition of $\mathcal{D}^\epsilon(0)$ also describes clearly an estimate of the domain of attraction of the desired equilibrium of the closed-loop system.

Remark 5.10. Note that the configuration of the desired formation coincides with the given configuration q^* in (5.1) for designing universally rigid tensegrity framework, we thus can stabilize the proper formation consisting of any arbitrary number of agents theoretically. When one or more agents encounter mechanical failures, to stabilize the rest, the weights k_{ij} in control law (5.51) are required to be updated by recalculating the stresses ω_{ij} based on the proposed algorithm, where the matrix Q^* needs to be altered via removing the failure agents' configuration. The resultant formation will remain the same as the original one associated with the agents without failure.

Remark 5.11. From (5.55), we know the equilibrium configuration is

$$\lim_{t\to\infty} q(t) = q^* + \gamma\, span(\mathbf{1}_d),$$

where γ is an arbitrary real number. Indeed, the stabilized configuration q is a class of *affine transformation* of q^* [23], which is still in equilibrium with the stress ω, i.e.,

$$\sum_{j\in\mathcal{N}_i} \omega_{ij}(q_i - q_j) = \sum_{j\in\mathcal{N}_i} \omega_{ij}(q_i^* - q_j^*) = \mathbf{0}, \quad i = 1, \cdots, n.$$

Remark 5.12. In this chapter, the single integrator model is taken into account under the assumption that the velocities of the agents can be controlled directly. In practice, the signals generated by the system can serve as the commanded 3D-velocity for the tracking controllers of the quadrotors [59, 81] or vessels [44]. This technique has been employed in experimental setups in formation and motion control [42].

5.4 Simulation results

This section gives the simulation results to validate the effectiveness of the theoretical results derived in the preceding sections. For the construction of universally rigid tensegrity frameworks, we consider a general as well as generic configuration. Then, the formation stabilization algorithm is simulated on the resultant universally rigid tensegrity framework.

5.4.1 Construction of universally rigid tensegrity frameworks

The generic configuration is given as follows

$$q^* = \begin{bmatrix} -\sqrt{3}/2 & \sqrt{3}/2 & 0 & -1/2 & 1 & -2/3 \\ -1/2 & -1/2 & 1 & -1/2 & 0 & 8/3-\sqrt{3} \\ 0 & 0 & 0 & 3 & 3 & 3\sqrt{3}-2 \end{bmatrix}. \tag{5.56}$$

With this configuration, the corresponding geometric shape of the six nodes is shown in Fig. 5.1. Based on our proposed algorithm, one solution of the Gale matrix D of the Gale matrix D derived from (5.5) is given as follows

$$D = \frac{1}{3}\begin{bmatrix} 2\sqrt{3} & -\sqrt{3} & -\sqrt{3} & -3 & 3 & 0 \\ 0 & 3 & 3\sqrt{3}-8 & -\sqrt{3} & 2-2\sqrt{3} & 3 \end{bmatrix}^T.$$

Then the stress matrix Ω can be calculated via $\Omega = DD^T$:

$$\Omega = \frac{1}{9}\begin{bmatrix} 12 & -6 & -6 & -6\sqrt{3} & 6\sqrt{3} & 0 \\ -6 & 12 & 9\sqrt{3}-21 & 0 & 6-9\sqrt{3} & 9 \\ -6 & 9\sqrt{3}-21 & 94-48\sqrt{3} & 11\sqrt{3}-9 & 19\sqrt{3}-34 & 9\sqrt{3}-24 \\ -6\sqrt{3} & 0 & 11\sqrt{3}-9 & 12 & -3-2\sqrt{3} & -3\sqrt{3} \\ 6\sqrt{3} & 6-9\sqrt{3} & 19\sqrt{3}-34 & -3-2\sqrt{3} & 25-8\sqrt{3} & 6-6\sqrt{3} \\ 0 & 9 & 9\sqrt{3}-24 & -3\sqrt{3} & 6-6\sqrt{3} & 9 \end{bmatrix}. \tag{5.57}$$

It can be seen from Ω that 13 members are required to construct a universally rigid tensegrity framework, 10 of which are cables and 3 struts. The corresponding tensegrity framework is shown in Fig. 5.2, where the thin black and thick blue lines are cables and struts, respectively. It is worth noting that 13 is exactly the minimal number of members required to build a universally rigid tensegrity framework in generic configurations with 6 nodes in 3D.

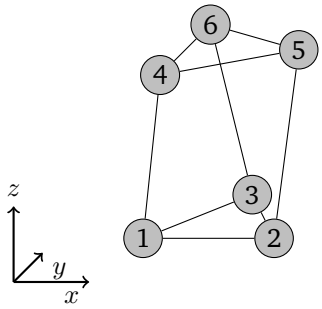


Figure 5.1: *Desired geometric shape associated with* (5.56).

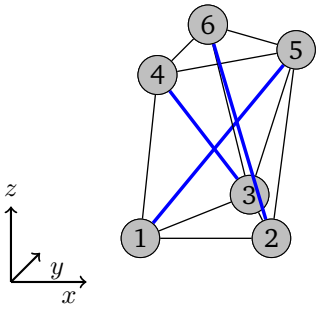


Figure 5.2: *Resultant framework based on* Ω *in* (5.57).

5.4.2 Formation stabilization

Based on the obtained tensegrity framework, where the nodes represent the agents modeled by (5.8), the control input (5.51) for each agent can be specified. For simplification, we set the parameters γ_{ij}^s to be the same value $\gamma^s = 2$, and γ_{ij}^c is chosen to be $\gamma^c = 0.5$. The initial states of $q(t)$ are chosen as $q_i(0) = q_i^* + 0.3 * \text{rands}(3,1)$, $i = 1, 2, \cdots, 6$.

It can be seen from Fig. 5.3 that the length of each member converges to its desired one. The upper panel of Fig. 5.3 shows the stabilization errors of the cables, and the corresponding stabilization errors of the struts are shown in the lower panel. The length evolution intervals of the cables and struts together with their rest lengths are presented respectively in Table 5.1 and 5.2, from which we can observe that the length of any cable (resp. strut) is always longer (resp. shorter) than its rest length during the evolution, verifying Proposition 5.6 numerically. As

can be seen from Fig. 5.3, all the stabilization errors converge to zero within 0.1s. Then the formation shape variations during $[0, 0.1]$s are depicted sequentially in Fig. 5.4, where the motion of the straight line of the whole formation results from an additional control input $u_e = [140, 0, 0]^T$ for each agent. This input is independent of the system state, and thus can be separated from the control input (5.51). The intention of designing u_e is to clearly show the variation of the geometric shape of the formation during the system evolution. Overall, these numerical results indicate that the prescribed formation can be achieved using the virtual framework and our proposed control algorithm (5.51).

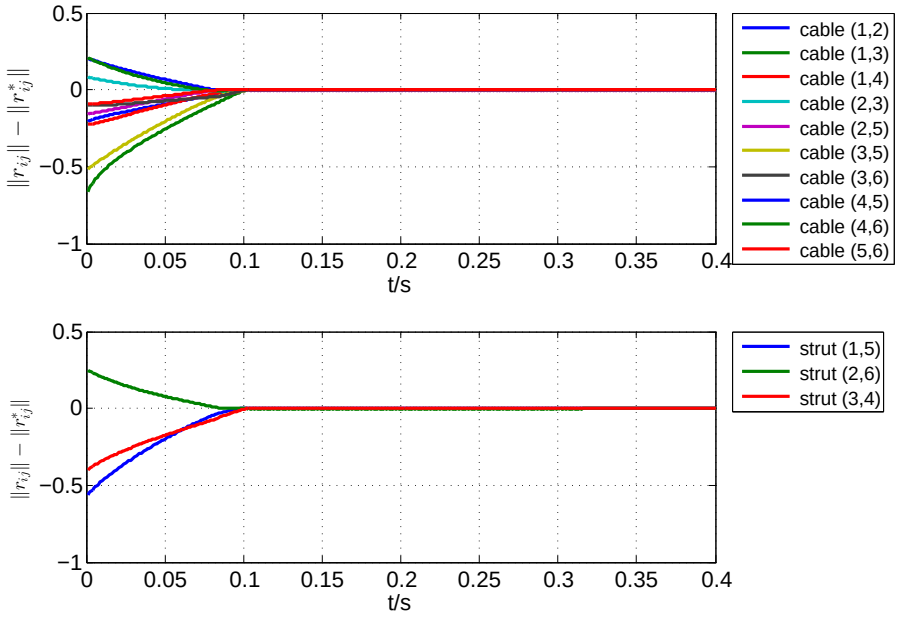


Figure 5.3: *The stabilization errors of the members.*

5.5 Concluding remarks

In this chapter, we have proposed a numerical algorithm, based on which universally rigid tensegrity frameworks can be built for any given configuration in general position. Furthermore, the upper bound of the members inserted in the framework has also been given. Then, we have investigated the formation stabilization problem as one of the applications, where distributed control strategies have been designed, such that the prescribed formation can be realized. During the stabilization evolution, the lengths of the members are shown to vary over or below some bounds, which is of great interest for tethered robot control.

Table 5.1: *Length evolution intervals of the cables and their rest lengths.*

Cable	Length interval	Rest length
(1, 2)	[1.5238, 1.7328]	0.8660
(1, 3)	[1.0683, 1.7340]	0.8660
(1, 4)	[2.8182, 3.0474]	1.5222
(2, 3)	[1.7211, 1.8152]	0.8600
(2, 5)	[2.8860, 3.0471]	1.5222
(3, 5)	[2.8011, 3.3172]	1.6583
(3, 6)	[3.1644, 3.2659]	1.6328
(4, 5)	[1.7305, 1.9374]	0.8600
(4, 6)	[1.8143, 2.0299]	0.9095
(5, 6)	[1.8263, 1.9219]	0.9604

Table 5.2: *Length evolution intervals of the struts and their rest lengths.*

Strut	Length interval	Rest length
(1, 5)	[3.0050, 3.5694]	7.1364
(2, 6)	[3.8210, 4.0727]	7.6479
(3, 4)	[3.1661, 3.5706]	7.1364

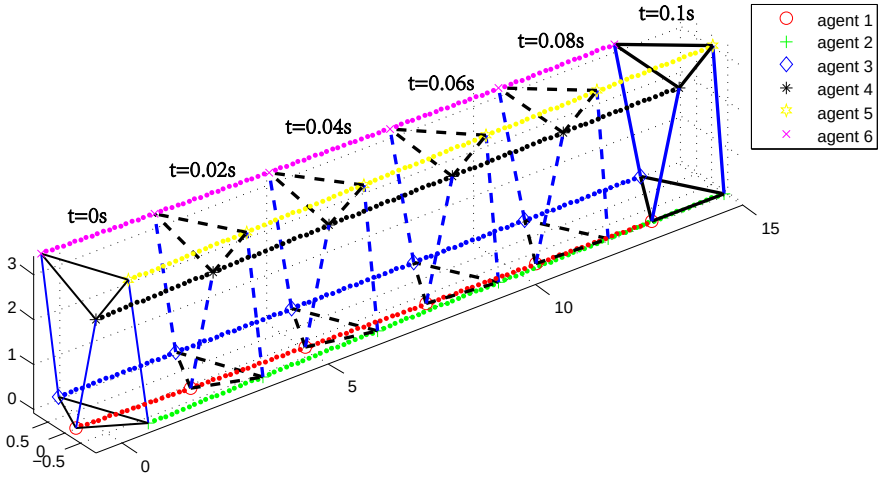


Figure 5.4: *Formation evolution between* $t = 0 - 0.1s$.

Chapter 6

Stress matrix-based formation scaling control

This chapter investigates the formation scaling control problem for multi-agent systems by mapping the formation into a universally rigid tensegrity framework with the underlying graph representing the agents and their interaction relationship. We first propose distributed formation scaling control laws by utilizing the stress of the universally rigid tensegrity framework. It is shown that global exponential convergence to the prescribed formation in $\mathbb{R}^d$ can be achieved by only controlling d pairs of agents whose position vectors span $\mathbb{R}^d$, under the assumption that each of the d pairs of agents has the knowledge of the desired formation size. Then by employing the technique of orthogonal projection, we design a new class of distributed control laws under which the agents are steered to form the desired formation under the relaxed assumption that only one pair of agents knows the scaling size; it is further proved that if the stress in the developed control law admits a generic universally rigid tensegrity framework, the equilibria correspond only to the translation and scaling of the given configuration among all the possible affine transformations. Finally, we propose a class of estimator-based control strategies, which can solve the formation scaling problem under the stricter condition that only one agent knows the prescribed size of the formation. Numerical simulations are carried out to validate the theoretical results.

6.1 Introduction

There has been a significant increase in the research on cooperative control of multi-agent systems. A fundamental task for cooperative control is formation control, which has found a wide range of applications, including networked mobile sensors performing ocean sampling tasks, a group of mobile robots enclosing a target, and unmanned aircrafts imaging in space [11, 73]. The main objective of distributed formation control is to design control laws using only local information to realize a given prescribed formation shape.

In general, the shape of a formation can be specified by various types of variables: absolute position, relative position (or displacement), distance, bearing

[138], and complex Laplacian [77]. In position-based control, each agent is informed of its absolute position and the desired position with respect to a global coordinate system, where agents can be controlled individually without any interaction with their neighbors. Therefore, network interaction among agents is not required but a global coordinate system for all agents is needed [91]. When the relative position becomes the sensed and controlled variable, the desired realizable formation can be achieved based on consensus algorithms using only measurements from local coordinate systems. However, the orientations of the local coordinate systems are required to be the same as that of a global coordinate system. In recent years, researchers have thoroughly studied the relative-position-based formation control from various aspects: linear and nonholonomic agent dynamics [79, 127]; undirected and directed switching interaction graphs [62, 92]; continuous- and discrete-time models [28, 130], to name a few.

In comparison, it is allowed in distance-based formation control that the sensed variable, i.e., relative position, can be measured in an arbitrary local coordinate system for each agent [91, 115]. However, using the gradient control protocols, only local stability is guaranteed for distance-based control systems under general graphs. In this scenario, rigidity graph theory has been shown to be an effective tool for analyzing the equilibrium formations up to translations and rotations. In [70], infinitesimal rigidity is shown to be a sufficient condition for locally asymptotically stabilizing an equilibrium formation under gradient control laws. To investigate global stability for triangular formations in the plane, it is shown in [14] that properly initialized formations can be controlled to exponentially converge to the desired formation with proper orientation. Note that to implement gradient control laws, relative positions are measured. The paper [15] proposes a stop-and-go cyclic strategy, which can stabilize a generically minimally rigid formation using only inter-agent distances. More recently, researchers have investigated the formation robustness issues, and have established formation movements in the presence of measurement mismatches [87, 116].

Investigating formation scaling is a growing major concern within formation control since the formation with varying size can dynamically adapt to changing environments in practice, such as obstacle avoidance for a group of vehicles. In [27], via a projection operator approach, two strategies are designed for the case when the scaling parameter is known to some of the agents. However, for the single-link method developed in [27], the monitoring graph needs to be chosen to contain all the vertices in the sensing graph. Later, the projection operator is also employed in [138], where bearing-based control frameworks are established. In addition, [54] addresses the formation scaling problem for both single- and double-integrator agent dynamics in the context of complex Laplacians.

In this chapter, we adopt a stress matrix-based approach to control a formation with the desired scaling, where the stress may contain negative values. It is worth

noting that most of the interacting weights in consensus-based protocols are positive. However, in some complex networks, e.g., social networks, the weights of the links cannot always be guaranteed to be positive. It is also shown in [129] that negative weights could contribute to faster convergence speed. Therefore, it is meaningful to incorporate negative weights in cooperative control. The stress matrix, defined in the same structure as a typical Laplacian matrix, is widely used to represent the stresses of edges and their connection relationships in a framework. Stress can be interpreted physically as the force per unit length, whose sign indicates the direction of the force. Hence, the stress matrix implicitly captures the features of a framework, e.g., rigidity, stability, and robustness [25]. Recently, a new type of formation pattern called affine formation has been investigated in [78], in which necessary and sufficient graphical conditions to achieve an affine formation are presented by employing the concept and properties of universal rigidity theory. It has also been revealed that an affine transformation of a given configuration is invariant to translation and scaling.

Motivated by these results, the goal of the current chapter is to first design distributed formation scaling control algorithms using the stress associated with a universally rigid tensegrity framework, such that the desired formation with predefined size in $\mathbb{R}^d$ is achieved. In the control algorithm, d pairs of agents whose position vectors span $\mathbb{R}^d$ are assumed to know the desired formation size, which renders the global exponential stability of the closed-loop system. Then to relax the condition that the chosen d pairs of the agents need to know the size, we propose orthogonal-projection-based control laws, where only *two* neighboring agents are required to be aware of the desired formation size. We show that the affine formation can be constrained to only translation and scaling even though only two of them have access to the desired size of the formation. Furthermore, under the more restrictive condition that only *one* agent knows the prescribed size, we design a class of estimator-based control laws, which successfully stabilize the agents to a predefined pattern from disordered initial formations. As a consequence, the feasibility of the proposed control law is highly improved in practice.

The rest of this chapter is organized as follows. Section 6.2 introduces the formation scaling control problem. In Section 6.3, we present basic stress matrix-based cooperative control laws for controlling formation scaling, followed by the stability analysis of the closed-loop system. Section 6.4 provides a new type of control laws by combining the stress and orthogonal projections. In Section 6.5, we introduce another type of estimator-based control strategies to further reduce the number of agents knowing the scaling parameter. Simulation results are presented in Section 6.6. Finally, we draw the conclusion in Section 6.7.

6.2 Problem formulation

Consider a group of $n \geqslant d+2$ mobile agents, each of which is modeled by single integrator dynamics

$$\dot{q}_i = u_i, \quad i = 1, \cdots, n, \tag{6.1}$$

where $q_i \in \mathbb{R}^d$ is the position of agent i and $u_i \in \mathbb{R}^d$ is the control input. Given a generic universally rigid tensegrity framework $(\mathcal{G}, q^*)$ with an equilibrium stress ω, the objective of formation scaling is, by using the stress ω, to design distributed control laws $u_i(q_i^* - q_j^*, q_i - q_j), j \in \mathcal{N}_i$, such that

$$\lim_{t\to\infty} (q_i(t) - q_j(t)) = \kappa(q_i^* - q_j^*), \forall (i,j) \in \mathcal{E}, \tag{6.2}$$

where κ is a positive constant indicating the size of the formation. Here, by mapping the multi-agent system to the universally rigid tensegrity framework, we assigned each edge of the formation with a weight (or stress), which can be either positive or negative.

Remark 6.1. The formation scaling problem becomes trivial if each agent knows the scaling parameter κ. However, in this chapter, we show the formation scaling can still be achieved using the proposed algorithms even only a small number of agents knows κ.

6.3 Formation scaling control using the stress matrix

In this section, we consider the formation scaling control problem, in which the formation can expand or shrink according to the parameter κ defined in (6.2). Distributed control laws are proposed by employing the stress of a universally rigid tensegrity framework.

Before moving on, we select d pairs of nodes in the given universally rigid tensegrity framework $(\mathcal{G}, q^*)$, such that the dimension of the convex hull of the selected nodes is d. Denote the set of edges corresponding to the d pairs of chosen nodes as $\mathcal{E}_l$. All the nodes involved in $\mathcal{E}_l$ are assembled in the node set $\mathcal{V}_l = \{1, \cdots, n_l\}$, and the set of the remaining nodes in $\mathcal{V}$ is denoted by $\mathcal{V}_f = \{n_l + 1, \cdots, n\}$. Here, the d pairs of nodes are chosen such that the resultant subgraph $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$ is connected. It is worth noting that the chosen d pairs of nodes can involve less than $2d$ nodes, due to the common endpoint shared by distinct edges. Fig. 6.1 shows an example of the setup for the subgraph $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$, where the dashed lines and solid lines represent the cables and struts, respectively.

(a) A tensegrity framework $(\mathcal{G}, q^*)$ in R^2.

(b) $\mathcal{E}_l = \{(1,2),(2,3)\}$, $\mathcal{V}_l = \{1,2,3\}$.

Figure 6.1: *An example of setting* $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$.

Then the control input for each agent i is designed as

$$u_i = -\sum_{(i,j)\in\mathcal{E}} \omega_{ij}(q_i - q_j) - \sum_{(i,j)\in\mathcal{E}_l} a_{ij}\left[(q_i - q_j) - \kappa(q_i^* - q_j^*)\right], \tag{6.3}$$

where ω_{ij} is the stress of member (i,j). It can be seen that the control input includes two parts:

$$u_i^F = -\sum_{(i,j)\in\mathcal{E}} \omega_{ij}(q_i - q_j), \tag{6.4}$$

and

$$u_i^S = -\sum_{(i,j)\in\mathcal{E}_l} a_{ij}\left[(q_i - q_j) - \kappa(q_i^* - q_j^*)\right], \tag{6.5}$$

where the internal force u_i^F generated from the virtual tensegrity framework is used to stabilize the formation shape, and the input u_i^S is to realize formation scaling. Equivalently, the control input (6.3) can be written as

$$u_i = \begin{cases} u_i^F + u_i^S, & \text{if } i \in \mathcal{V}_l, \\ u_i^F, & \text{if } i \in \mathcal{V}_f. \end{cases}$$

One of the main results concerning the formation scaling is presented as follows.

Theorem 6.2. *For system* (6.1)*, by employing the virtual tensegrity-framework-based control law* (6.3) *for each agent, the target formation with the prescribed size is globally exponentially stabilized.*

Proof. The control input u_i^F in (6.3) can be written in the compact form as

$$u^F = -(\Omega \otimes I_d)\bar{q}, \tag{6.6}$$

where $\bar{q} = [q_1^T, \cdots, q_n^T]^T \in \mathbb{R}^{dn}$ and $u^F = \left[(u_1^F)^T, \cdots, (u_n^F)^T\right]^T$ are the vector form of q_i and u_i^F, respectively. Similarly, consider the scaling control part of (6.3),

i.e., u_i^S, which can be written in the vector form as

$$u^S = -(L_s \otimes I_d)\tilde{q}, \tag{6.7}$$

where u^S is the concatenated form of u_i^S, and $\tilde{q} = [\tilde{q}_1^T, \cdots, \tilde{q}_n^T]^T \in \mathbb{R}^{dn}$, with $\tilde{q}_i \in \mathbb{R}^d$ being defined by

$$\tilde{q}_i = q_i - \kappa q_i^*. \tag{6.8}$$

The matrix L_s is given by

$$L_s = \begin{bmatrix} L_l & \mathbf{0}_{n_l \times (n-n_l)} \\ \mathbf{0}_{(n-n_l)\times n_l} & \mathbf{0}_{(n-n_l)\times(n-n_l)} \end{bmatrix}, \tag{6.9}$$

where L_l is the Laplacian matrix associated with the agents in the set $\mathcal{N}_l$, defined by

$$[L_l]_{ij} = \begin{cases} \sum\limits_{j\in\mathcal{N}_i} a_{ij}, & i = j, \\ -a_{ij}, & i \neq j. \end{cases}$$

By combining (6.6) and (6.7), it follows

$$u = -\left((\Omega + L_s) \otimes I_d\right)\tilde{q}, \tag{6.10}$$

where we have used the equilibrium stress condition that $(\Omega \otimes I_d)\bar{q}^* = \mathbf{0}$, and $\bar{q}^*$ is defined as $\bar{q}^* = \left[(q_1^*)^T, \cdots, (q_n^*)^T\right]^T$. Then the dynamics of $\tilde{q}$ is given by

$$\dot{\tilde{q}} = -\left((\Omega + L_s) \otimes I_d\right)\tilde{q} \triangleq -\bar{\Omega}\tilde{q}. \tag{6.11}$$

Note that the stress matrix Ω is positive semi-definite, so is L_s. Therefore, the matrix $\tilde{A}$ is positive semi-definite. Hence, the equilibrium of the closed-loop system (6.11) is globally stable. Furthermore, the equilibrium points of system (6.11), denoted by q^e, satisfy

$$\dot{\tilde{q}}^e = -\left((\Omega + L_s) \otimes I_d\right)\tilde{q}^e \triangleq -\bar{\Omega}\tilde{q}^e = \mathbf{0}_{nd\times 1},$$

where $\tilde{q}^e$ is the stacked vector of $\tilde{q}_i^e = q_i^e - \kappa q_i^*, i = 1, \cdots, n$. It follows from Lemma 2.17 that

$$\tilde{q}^e \in \mathrm{null}(\Omega \otimes I_d), \quad \text{and} \quad \tilde{q}^e \in \mathrm{null}(L_s \otimes I_d). \tag{6.12}$$

Therefore, we have

$$\left[u_1^F, \cdots, u_n^F\right] = -\left(\sum_{(1,j)\in\mathcal{E}} \omega_{1j}(q_1^e - q_j^e), \cdots, \sum_{(n,j)\in\mathcal{E}} \omega_{nj}(q_n^e - q_j^e)\right) = \mathbf{0}, \tag{6.13}$$

and similarly,

$$\left[u_1^S, \cdots, u_n^S\right] = -(q^e - \kappa q^*)L_s = \mathbf{0}_{d\times n}. \tag{6.14}$$

Equivalently, we consider the reduced form of (6.14) as follows

$$\left[u_1^S, \cdots, u_{n_l}^S\right] = -(q_s^e - \kappa q_s^*)L_l = \mathbf{0}_{d\times n_l}, \tag{6.15}$$

where $q_s^e = [q_1^e, \cdots, q_{n_l}^e] \in \mathbb{R}^{d\times n_l}$, $n_l > d$, and $q_s^* = [q_1^*, \cdots, q_{n_l}^*]$.

Combining (2.5) and (6.13), we know q^e is the affine transformation of q^* with respect to Ω, i.e.,

$$q_i^e = Mq_i^* + b, \quad i = 1, \cdots, n, \tag{6.16}$$

where $M \in \mathbb{R}^{d\times d}$ and $b \in \mathbb{R}^d$. Substituting (6.16) into (6.15), yields

$$\left[(M - \kappa I_d)q_1^* + b, \cdots, (M - \kappa I_d)q_{n_l}^* + b\right] L_l = \mathbf{0}_{d\times n_l}. \tag{6.17}$$

Note that the Laplacian matrix L_l satisfies

$$\mathrm{null}(L_l) = \mathrm{span}(\mathbf{1}_{n_l}). \tag{6.18}$$

Therefore, it follows from (6.17) and (6.18) that

$$\mathrm{span}\left[(M - \kappa I_d)q_1^* + b, \cdots, (M - \kappa I_d)q_{n_l}^* + b\right] = \mathrm{span}(\mathbf{1}_{n_l}),$$

i.e.,

$$\mathrm{span}[(M - \kappa I_d)q_s^* + (b \otimes \mathbf{1}_{n_l}^T)] = \mathrm{span}(\mathbf{1}_{n_l}). \tag{6.19}$$

In view of $n_l > d$, to make (6.19) hold, it requires

$$(M - \kappa I_d)q_s^* = [\xi, \cdots, \xi],$$

where $\xi \in \mathbb{R}^d$ is any arbitrary real vector. Then we obtain

$$(M - \kappa I_d)(q_i^* - q_j^*) = \mathbf{0}, \quad i, j \in \mathcal{V}_l. \tag{6.20}$$

By recalling that the dimension of the convex hull of $(q_i^* - q_j^*), i, j \in \mathcal{V}_l$, is d, it follows from (6.20) that $M = \kappa I_d$. Then, we can draw the conclusion that formation scaling is achieved.

Note that

$$\mathrm{null}(\Omega) = \mathrm{span}\left((q^*)^T, \mathbf{1}_n\right). \tag{6.21}$$

Since q converge to κq^*, only the freedom of translation is left for the stabilized formation, which results from the basis $\mathbf{1}_n$ in the null space of Ω. Note that

$$\mathrm{span}(\mathbf{1}_n) \in \mathrm{null}(L_s).$$

Consequently, again from Lemma 2.17, we have

$$\text{null}\left((\Omega + L_s)\right) = \text{span}(\mathbf{1}_n). \tag{6.22}$$

Now, we show the convergence is achieved globally exponentially and derive the guaranteed exponential rate.

Define the formation centroid by

$$q_c = \frac{1}{n}\sum_{i=1}^{n} q_i = \frac{1}{n}(\mathbf{1}_n \otimes I_d)^T \bar{q}.$$

Then the dynamics of the centroid satisfy

$$\dot{q}_c = \frac{1}{n}(\mathbf{1}_n \otimes I_d)^T \dot{\tilde{q}} = \frac{1}{n}(\mathbf{1}_n \otimes I_d)^T \left((\Omega + L_s) \otimes I_d\right) \tilde{q} = 0,$$

which implies that the centroid of the formation keeps static. Following the same line of the proof in [118, Theorem 3], we construct an orthogonal matrix $S \in \mathbb{R}^{dn\times dn}$ as

$$S = \begin{pmatrix} \frac{1}{\sqrt{n}}(\mathbf{1}_n \otimes I_d)^T \\ S_r \end{pmatrix},$$

where $S_r \in \mathbb{R}^{d(n-1)\times dn}$. Then consider the coordinate transformation

$$p = S\tilde{q} = \begin{pmatrix} p_c \\ p_r \end{pmatrix}, \tag{6.23}$$

where $p_c = \frac{1}{\sqrt{n}}(\mathbf{1}_n \otimes I_d)^T(\bar{q} - \kappa\bar{q}^*) = \sqrt{n}q_c - \kappa\sqrt{n}q_c^*$, with q_c^* defined by $q_c^* = 1/n\sum_{i=1}^{n} q_i^*$. From (6.23), one has

$$\tilde{q} = S^{-1}p = S^T p. \tag{6.24}$$

Taking the derivative of both sides of (6.23), we have

$$\dot{p} = S\dot{\tilde{q}} = -S\bar{\Omega}\tilde{q} = -S\bar{\Omega}S^T p.$$

Equivalently,

$$\begin{aligned}\begin{bmatrix} \dot{p}_c \\ \dot{p}_r \end{bmatrix} &= -S\bar{\Omega}S^T \begin{bmatrix} p_c \\ p_r \end{bmatrix} \\ &= -\begin{bmatrix} \frac{1}{\sqrt{n}}(\mathbf{1}_n \otimes I_d)^T \\ S_r \end{bmatrix} \bar{\Omega} \begin{bmatrix} \frac{1}{\sqrt{n}}(\mathbf{1}_n \otimes I_d)^T \\ S_r \end{bmatrix}^T \begin{bmatrix} p_c \\ p_r \end{bmatrix}\end{aligned}$$

$$= \begin{bmatrix} \mathbf{0}_{d\times d} & \mathbf{0}_{d\times d(n-1)} \\ \mathbf{0}_{d(n-1)\times d} & S_r\tilde{A}S_r^T \end{bmatrix} \begin{bmatrix} p_c \\ p_r \end{bmatrix}.$$

Consequently, the transformed system dynamics become

$$\begin{cases} \dot{p}_c = \mathbf{0} \\ \dot{p}_r = -S_r\bar{\Omega}S_r^T p_r \end{cases}. \tag{6.25}$$

In view of (6.22), we know the matrix $S_r\bar{\Omega}S_r^T$ is positive definite. Therefore, the state p_r will globally exponentially converges to the equilibrium $p_r = \mathbf{0}$. Recalling (6.24) with orthogonal matrix S and the fact that p_c keeps constant, we draw the conclusion that $\tilde{q}$ globally exponentially converges to zero, which implies q converges to κq^* globally exponentially from (6.8). This implies that the formation scaling is achieved in the sense of globally exponential stability. In addition, it can be seen from (6.25) that the convergence rate depends on the eigenvalues of matrix $\bar{\Omega}$, or equivalently, matrices Ω and L_s. □

Remark 6.3. From (6.21), it is clear that if one only uses the control law u_i^F in (6.4), then there is no constraint for the size of the formation. Therefore, the idea of designing the u_i^S in (6.5) is to reduce the dimension of the null space of Ω, namely, to restrict the null space of $(\Omega \otimes I_d)$ to $span(\mathbf{1}_n \otimes I_d)$. To achieve this goal, at least d pairs of agents are required to construct the sub-Laplacian matrix L_l in (6.9).

6.4 Formation scaling control via the stress matrix and orthogonal projections

In Section 6.3, we have shown that the formation scaling problem can be solved using the proposed control law (6.3) if d pairs of agents have accesses to the formation scaling parameter κ. Aiming to further reduce the number of the agents knowing κ, in this section, we present a new class of distributed control laws by utilizing the orthogonal projections.

To facilitate the design of control laws, we choose $d+1$ members in $(\mathcal{G}, q^*)$, automatically yielding $d+1$ pairs of nodes corresponding to the chosen $d+1$ members, such that the dimension of the convex hull spanned by any d pairs of the chosen nodes is d. The subgraph associated with those $d+1$ pairs of agents is denoted by $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$, with $|\mathcal{V}_l| = n_l$, $|\mathcal{E}_l| = d+1$. Here, the nodes are also properly chosen to make subgraph $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$ connected. Correspondingly, we have $\mathcal{V}_f$ and $\mathcal{E}_f$, such that $\mathcal{V}_l \cup \mathcal{V}_f = \mathcal{V}$ and $\mathcal{E}_l \cup \mathcal{E}_f = \mathcal{E}$. As illustrated before, there must be fewer

than $2(d+1)$ nodes involved in the chosen $d+1$ members due to the connectivity constraint of the subgraph $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$. Fig. 6.2 shows an example of determining the sub-graph $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$, where the dashed lines and solid lines represent cables and struts, respectively.

(a) A tensegrity framework $(\mathcal{G}, q^*)$ in $\mathbb{R}^2$.

(b) $\mathcal{V}_l = \{1,2,3\}$,$\mathcal{E}_l = \{(1,2),(2,3),(1,3)\}$.

Figure 6.2: *An example of determining* $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$.[1]

Then the incidence matrix H can be partitioned as

$$H = \left(\begin{array}{c:c} H_{ll} & H_{lf} \\ \hdashline H_{fl} & H_{ff} \end{array}\right), \tag{6.26}$$

where $H_{ll} \in \mathbb{R}^{n_l \times (d+1)}$, $H_{lf} \in \mathbb{R}^{n_l \times (m-d-1)}$, $H_{fl} \in \mathbb{R}^{(n-n_l)\times(d+1)}$, and $H_{ff} \in \mathbb{R}^{(n-n_l)\times(m-d-1)}$. Furthermore, from the definition of the sets $\mathcal{V}_l$ and $\mathcal{E}_l$, we know that no vertex in $\mathcal{V}_f$ is adjacent to the edges in $\mathcal{E}_l$, which implies $H_{fl} = \mathbf{0}$.

Suppose none of the agents has the knowledge of κ. However, the information of κ is implicitly contained in one specific edge. Without loss of generality, we assume this edge is adjacent to agents 1 and 2. This means $\kappa(q_1^* - q_2^*)$ is known by agents 1 and 2 as a whole piece of information. For other edges in the edge set $\mathcal{E}_l$, only the information of $q_i^* - q_j^*, (i,j) \in \mathcal{E}_l \backslash (1,2)$, is available to their adjacent agents.

Define an auxiliary variable $z = [z_1^T, \cdots, z_m^T]^T \in \mathbb{R}^{md}$ as follows

$$z = (H^T \otimes I_d)q,$$

where H is the incidence matrix, and $z_\iota = q_i - q_j, \iota = 1, \cdots, m$, with agents i and j being the head and tail of the ιth edge, respectively. To be consistent, we assume the specific edge connecting agents 1 and 2 is labeled as the 1st edge. Therefore, it follows $z_1 = q_1 - q_2$. Analogously, we have

$$z^* = (H^T \otimes I_d)q^*.$$

[1]Fig. 6.2 differs from Fig. 6.1 in the edge set $\mathcal{E}_l$ of the subgraph (b), where there is one more edge $(1,3)$ in Fig. 6.2(b) compared with Fig. 6.1(b).

The projection of z_ι along z_ι^* is given by

$$\kappa_\iota = \frac{1}{\|z_\iota^*\|^2}(z_\iota^*)^T z_\iota, \quad \iota = 2, \cdots, d+1. \tag{6.27}$$

The projection method is also employed in [27, 138], where the projection operator (6.27) is used to 'estimate' the scaling parameter [27] and to realize the bearing-based control [138].

The control inputs for agents 1 and 2 are designed as

$$u_1 = -\sum_{(1,j)\in\mathcal{E}} \omega_{1j}(q_1 - q_j) - h_{11}(z_1 - \kappa z_1^*) - \sum_{\iota=2}^{d+1} h_{1\iota}(z_\iota - \kappa_\iota z_\iota^*), \tag{6.28}$$

and

$$u_2 = -\sum_{(2,j)\in\mathcal{E}} \omega_{2j}(q_2 - q_j) - h_{21}(z_1 - \kappa z_1^*) - \sum_{\iota=2}^{d+1} h_{2\iota}(z_\iota - \kappa_\iota z_\iota^*). \tag{6.29}$$

where ω_{ij} is the stress associated with member (i,j). It is worth noting that even though κz_1^* is contained in the control laws (6.28) and (6.29), agents 1 and 2 have no knowledge of the value of κ, since κz_1^* is transmitted as a whole piece of information. The reason that the desired information of edge 1 is written as κz_1^* is to facilitate the stability analysis. For the rest of the agents, their control inputs $u_i, i = 3, \cdots, n$, are given by

$$u_i = -\sum_{(i,j)\in\mathcal{E}} \omega_{ij}(q_i - q_j) - \sum_{\iota=2}^{d+1} h_{i\iota}(z_\iota - \kappa_\iota z_\iota^*). \tag{6.30}$$

Similar to (6.3), the proposed control input for each agent consists of two parts: the internal force $-\sum_{(i,j)\in\mathcal{E}} \omega_{ij}(q_i - q_j)$ generated from the virtual tensegrity framework used to drive the whole group of agents to the affine space of the configuration q^*, and the rest used to fix the size of the formation. To implement the proposed control inputs (6.28)-(6.30) in practice, agents 1 and 2 can be arbitrarily chosen among the $d+1$ pairs of agents. The proposed control input has a similar part as the control laws proposed in [27, 138], while we introduce the negative weight that can model the antagonistic interactions between neighbor agents. Furthermore, using the stress matrix makes it possible that only a few number of agents are required to have the common knowledge of the global coordinate system, which will greatly broaden the applicability of the proposed control laws in practice. In addition, even though the conditions to achieve affine formations in the context of graph theory are presented in [78], no control law on formation

(scaling) control has been given.

Then we are ready to present another main result as follows.

Theorem 6.4. *Suppose the given generic framework $(\mathcal{G}, q^*)$ is universally rigid with an equilibrium stress ω. Then for a group of agents modeled by* (6.1)*, the formation scaling control task* (6.2) *can be achieved globally using the proposed distributed control laws* (6.28)- (6.30).

Proof. Since $\kappa_\iota z_\iota^* = z_\iota^* \kappa_\iota$, for κ_ι defined in (6.27) is a scalar, we have

$$z_\iota - \kappa_\iota z_\iota^* = \left(I_d - \frac{(z_\iota^*)(z_\iota^*)^T}{\|z_\iota^*\|^2} \right) z_\iota. \tag{6.31}$$

The vector corresponding to the right-hand side of (6.31) is in the direction of $(z_\iota^*)^\perp$. The (orthogonal) projection is to project vector z_ι to the orthogonal complement of z_ι^*. We denote the orthogonal projection operator as $Proj_\iota \triangleq I_d - \frac{(z_\iota^*)(z_\iota^*)^T}{\|z_\iota^*\|^2}$, $\iota = 2, \cdots, d+1$. Since κz_1^* is known to agents 1 and 2, to keep consistent with the notations, we denote $Proj_1 \triangleq I_d$.

Then the control laws (6.28)-(6.30) can be integrated as

$$\begin{aligned} u_i = &- \sum_{(i,j)\in\mathcal{E}} \omega_{ij}(q_i - q_j) - \sum_{\iota=1}^{d+1} h_{i\iota} Proj_\iota(z_\iota - \kappa z_\iota^*), \\ &i = 1, \cdots, n, \end{aligned} \tag{6.32}$$

where we have used the fact that

$$Proj_\iota(\kappa z_\iota^*) = \mathbf{0}_d, \quad \forall \kappa \in R, \quad \iota = 2, \cdots, d+1.$$

The compact form of (6.32) is in the form

$$u = -(\Omega \otimes I_d)q - (\bar{H}_{ll} \otimes I_d)\bar{P}_l(z - \kappa z^*), \tag{6.33}$$

where

$$\bar{H}_{ll} = \left(\begin{array}{c|c} H_{ll} & \mathbf{0}_{n_l \times (m-d-1)} \\ \hline \mathbf{0}_{(n-n_l)\times(d+1)} & \mathbf{0}_{(n-n_l)\times(m-d-1)} \end{array} \right),$$

and

$$\bar{P}_l = \mathrm{diag}(Proj_1, \cdots, Proj_{d+1}, \mathbf{0}_d, \cdots, \mathbf{0}_d).$$

Note that

$$(\Omega \otimes I_d)(\kappa q^*) = \kappa(\Omega \otimes I_d)q^* = \mathbf{0}. \tag{6.34}$$

Substituting (6.34) into (6.33), we have

$$u = -(\Omega \otimes I_d)(q - \kappa q^*) - (\bar{H}_{ll} \otimes I_d)\bar{P}_l(z - \kappa z^*). \tag{6.35}$$

In light of the fact that $z = (H^T \otimes I_d)q$, (6.35) can be rewritten as

$$u = -(\Omega \otimes I_d)(q - \kappa q^*) - (\bar{H}_{ll} \otimes I_d)\bar{P}_l(H^T \otimes I_d)(q - \kappa q^*). \tag{6.36}$$

Recalling that $H_{fl} = \mathbf{0}$ in (6.26), one has

$$\begin{aligned} &(\bar{H}_{ll} \otimes I_d)\bar{P}_l(H^T \otimes I_d) \\ &= \left(\begin{array}{c:c} H_{ll} \otimes I_d & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} \end{array}\right) \left(\begin{array}{c:c} P_l & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} \end{array}\right) \left(\begin{array}{c:c} H_{ll}^T \otimes I_d & \mathbf{0} \\ \hdashline H_{lf}^T \otimes I_d & H_{ff}^T \otimes I_d \end{array}\right) \\ &= \left(\begin{array}{c:c} (H_{ll} \otimes I_d)P_l(H_{ll}^T \otimes I_d) & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} \end{array}\right) \triangleq \Psi, \end{aligned} \tag{6.37}$$

where $P_l = \mathrm{diag}(Proj_1, \cdots, Proj_{d+1})$. Combining (6.36) and (6.37), we have

$$\dot{q} - \kappa\dot{q}^* = -\left((\Omega \otimes I_d) + \Psi\right)(q - \kappa q^*). \tag{6.38}$$

It can be checked that the eigenvalues of the matrix $(z_\iota^*)(z_\iota^*)^T/\|z_\iota^*\|^2$ are $\{0, \cdots, 0, 1\}$, where the algebraic multiplicity of eigenvalue 0 is $d-1$. Hence, the nonzero eigenvalue of the projection operator $Proj_\iota$ is 1 with the algebraic multiplicity $d-1$. This implies that the matrix $(H_{ll} \otimes I_d)P_l(H_{ll}^T \otimes I_d)$ is positive semi-definite, and so is the matrix Ψ. Note that for a universally rigid framework $(\mathcal{G}, q^*)$, its stress matrix Ω is positive semi-definite. Therefore, the equilibrium of the closed-loop system (6.38) is globally stable. In addition, the equilibrium points of system (6.38), denoted by q^e, satisfy

$$-\left((\Omega \otimes I_d) + \Psi\right)(q^e - \kappa q^*) = \mathbf{0}.$$

In view of Lemma 2.17, we have

$$\begin{cases} (\Omega \otimes I_d)(q^e - \kappa q^*) = \mathbf{0}, & (6.39) \\ \Psi(q^e - \kappa q^*) = \mathbf{0}. & (6.40) \end{cases}$$

Note that for a generic and universally rigid tensegrity framework $(\mathcal{G}, q^*)$, it follows from Lemma 2.9 that its corresponding stress matrix Ω is positive semi-definite with rank $n - d - 1$. Moreover, for the stress ω in equilibrium with q^*, in

view of the definition of Ω, we know

$$\text{null}(\Omega) = \text{span}\left(\begin{bmatrix} q_{11}^* \\ q_{21}^* \\ \vdots \\ q_{n1}^* \end{bmatrix} \begin{bmatrix} q_{12}^* \\ q_{22}^* \\ \vdots \\ q_{n2}^* \end{bmatrix} \cdots \begin{bmatrix} q_{1d}^* \\ q_{2d}^* \\ \vdots \\ q_{nd}^* \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}\right).$$

Then, it follows that q^e is an affine transformation of q^*, i.e.,

$$q_i^e = Mq_i^* + b, \tag{6.41}$$

where $M \in \mathbb{R}^{d\times d}$ and $b \in \mathbb{R}^d$. Substituting (6.41) into (6.40), we get

$$\Psi\left((I_n \otimes M)q^* + (\mathbf{1}_n \otimes b) - \kappa q^*\right) = \mathbf{0}. \tag{6.42}$$

In view of the structure of Ψ in (6.37), (6.42) can be reduced to

$$(D_{ll} \otimes I_d)P_l(D_{ll}^T \otimes I_d)\left[(I_{n_l} \otimes M)q_l^* + (\mathbf{1}_{n_l} \otimes b) - \kappa q_l^*\right] = \mathbf{0}. \tag{6.43}$$

Note that

$$(D_{ll}^T \otimes I_d)(\mathbf{1}_{n_l} \otimes b) = D_{ll}^T\mathbf{1}_{n_l} \otimes b = \mathbf{0}. \tag{6.44}$$

Then (6.43) can be equivalently written as

$$(D_{ll} \otimes I_d)P_l(D_{ll}^T \otimes I_d)\left[(I_{n_l} \otimes M)q_l^* - \kappa q_l^*\right] = \mathbf{0}. \tag{6.45}$$

To determine the value of matrix M, we write (6.45) in the componentwise form

$$\sum_{\iota=1}^{d+1} \xi_\iota = \mathbf{0},$$

where edge ι is assumed to be adjacent to vertices i and j, and ξ_ι is given by

$$\xi_\iota = \begin{pmatrix} \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & h_{i\iota}^2 Proj_\iota & \cdots & h_{i\iota}h_{j\iota}Proj_\iota & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \cdots & h_{j\iota}h_{i\iota}Proj_\iota & \cdots & h_{j\iota}^2 Proj_\iota & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix} \begin{pmatrix} \cdots \\ (M-\kappa I_d)q_i^* \\ \vdots \\ (M-\kappa I_d)q_j^* \\ \cdots \end{pmatrix}. \tag{6.46}$$

Noting that $h_{i\iota}h_{j\iota} = -1$, for each edge ι, we have

$$Proj_\iota(M-\kappa I_d)(q_i^* - q_j^*) = \mathbf{0},$$

i.e.,

$$Proj_\iota(M-\kappa I_d)z_\iota^*=\mathbf{0}.$$

Next, we will prove by contradiction that $M=\kappa I_d$. Assume $M\neq\kappa I_d$. Recalling that $Proj_1=I_d$, and that $Proj_\iota(\alpha z_\iota^*)=\mathbf{0}$, we get

$$\begin{cases}(M-\kappa I_d)z_1^*=0,\\(M-\kappa I_d)z_2^*=\alpha_2 z_2^*,\\\qquad\vdots\\(M-\kappa I_d)z_{d+1}^*=\alpha_{d+1}z_{d+1}^*,\end{cases}\tag{6.47}$$

where $\alpha_\iota\neq 0, \iota=2,\cdots,d+1$.

Since the dimension of the convex hull spanned by any d pairs of the agents in the set $\mathcal{V}_l$ is d, there exist $\beta_i, i=2,\cdots,d+1$, such that

$$z_1^*=\beta_2 z_2^*+\cdots+\beta_{d+1}z_{d+1}^*,\tag{6.48}$$

where at least one of the coefficients β_i is nonzero. Then multiplying $(M-\kappa I_d)$ on both sides of (6.48), we obtain

$$(M-\kappa I_d)z_1^*=(M-\kappa I_d)(\beta_2 z_2^*+\cdots+\beta_{d+1}z_{d+1}^*)=\mathbf{0}.\tag{6.49}$$

Combining (6.47) and (6.49), we have

$$\begin{aligned}&(M-\kappa I_d)(\beta_2 z_2^*+\cdots+\beta_{d+1}z_{d+1}^*)\\=&\beta_2(M-\kappa I_d)z_2^*+\cdots+\beta_{d+1}(M-\kappa I_d)z_{d+1}^*\\=&\alpha_2\beta_2 z_2^*+\cdots+\alpha_{d+1}\beta_{d+1}z_{d+1}^*\\=&\mathbf{0},\end{aligned}\tag{6.50}$$

where at least one of $\alpha_\iota\beta_\iota, \iota=2,\cdots,d+1$, is nonzero.

Considering again that any d pairs of agents in $\mathcal{V}_l$ linearly span $\mathbb{R}^d$, it is obvious that vectors $z_2^*,\cdots,z_d^*$, and z_{d+1}^* are linearly independent. This implies that $\mathbf{0}_d$ is the unique solution of $\gamma=[\gamma_1,\cdots,\gamma_d]^T$ to the following equation

$$\gamma_1 z_2^*+\cdots+\gamma_d z_{d+1}^*=\mathbf{0},$$

which contradicts (6.50). Therefore, the assumption $M\neq\kappa I_d$ does not hold. In other words, $M=\kappa I_d$. Then, from (6.41) we know

$$q_i^e=\kappa q_i^*+b,\quad i=1,\cdots,n_l.$$

Consequently, it follows

$$z_\iota = \kappa z_\iota^*, \quad \iota = 1, \cdots, d+1.$$

Then one can draw the conclusion from Lemma 2.16 that formation scaling for the whole group of agents is achieved. This completes the proof. □

6.5 Estimation-based formation scaling control

In this section, we further extend the results in Section 6.4 by assuming only one agent knows the desired formation size, i.e., the scaling parameter κ. With the intention to drive the agents to form the prescribed formation pattern with fixed scaling, we design a new type of distributed estimator-based control laws.

It has been shown in Section 6.3 that the formation can be scaled to the prescribed size if d pairs of agents with the associated connected subgraph $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$ knows κ. Following the same principle of constructing the subgraph $\mathcal{G}_l(\mathcal{V}_l, \mathcal{E}_l)$, we know there must exist a path $(1, 2, \cdots, n_l)$ through relabeling the agents due to the bidirectional property of an undirected graph. Without loss of generality, we assume only agent 1 knows the scaling parameter κ among the $|\mathcal{V}_l|$ agents.

Assumption 6.5. For any given q_i^*, $i = 2, \cdots, n$, there holds

$$|\mathcal{N}_{i-1}^l|(q_{i-1}^* - q_i^*) + \sum_{j \in \mathcal{N}_{i-1}^l \cap \mathcal{N}_i^l} (q_i^* - q_j^*) \neq \mathbf{0}_d. \tag{6.51}$$

With these background knowledge, the control input for agent 1 is the same as (6.3), given by

$$u_1 = -\sum_{j \in \mathcal{N}_1} \omega_{1j}(q_1 - q_j) - \sum_{j \in \mathcal{N}_1^l} a_{1j}\left((q_1 - q_j) - \kappa(q_1^* - q_j^*)\right), \tag{6.52}$$

where $\mathcal{N}_1^l$ denotes the set of neighbor agents of agent 1 in the subgraph $(\mathcal{E}_l, \mathcal{V}_l)$. For the rest, we introduce the following estimation-based control protocols

$$\begin{aligned} u_i = &-\sum_{j \in \mathcal{N}_i} \omega_{ij}(q_i - q_j) \\ &-\sum_{j \in \mathcal{N}_i^l} a_{ij}\left((q_i - q_j) - \hat{\kappa}_i(q_i^* - q_j^*)\right), \quad i = 2, \cdots, n, \end{aligned} \tag{6.53}$$

where $\hat{\kappa}_i$ is the estimation of κ by agent i. As illustrated in Section 6.3, the first part of the control input is used to achieve the affine formations associated with the stress ω, and the second part aims to fix the formation size from the

affine formations. It can be observed from (6.52)-(6.53) that only agent 1 knows the desired size of the formation, and the others employ the estimation variable $\hat{\kappa}_i, i = 2, \cdots, n_l$, in their control inputs. We propose the following estimators for agent 2

$$\begin{cases} \dot{\theta}_2 = -\Lambda_2 \xi_2^T \zeta_2 \\ \hat{\kappa}_2 = -\theta_2 - \Lambda_2 \xi_2^T (q_2 - q_1) \end{cases}, \tag{6.54}$$

and for agent $i, i = 3, \cdots, n_l$,

$$\begin{cases} \dot{\theta}_i = -\Lambda_i \xi_i^T \zeta_i \\ \hat{\kappa}_i = \hat{\kappa}_{i-1} - \theta_i - \Lambda_i \xi_i^T (q_i - q_{i-1}) \end{cases}, \tag{6.55}$$

where θ_i is an intermediate variable, and Λ_i is a positive scalar. The variables ξ_i and ζ_i are respectively given by

$$\xi_i = |\mathcal{N}_{i-1}^l|(q_{i-1}^* - q_i^*) + \sum_{j \in \mathcal{N}_{i-1}^l \cap \mathcal{N}_i^l} (q_i^* - q_j^*), \tag{6.56}$$

and

$$\begin{aligned} \zeta_i =& \hat{\kappa}_i \left(|\mathcal{N}_{i-1}^l| + 1 \right) (q_{i-1}^* - q_i^*) - \sum_{j \in \mathcal{N}_i} \omega_{ij}(q_i - q_j) \\ & - \sum_{j \in \mathcal{N}_i^l} a_{ij}(q_i - q_j) + \sum_{k \in \mathcal{N}_{i-1}} \omega_{(i-1)k}(q_{i-1} - q_k) \\ & + \sum_{k \in \mathcal{N}_{i-1}^l} a_{(i-1)k}(q_{i-1} - q_k), \quad i = 2, \cdots, n_l. \end{aligned} \tag{6.57}$$

Remark 6.6. As can be seen from (6.54) and (6.55), two-hop information is required to implement the relative-position-based estimator. Similar estimation problem was also addressed in [82] to estimate an unknown rotation parameter, in which the estimator is designed under the complete graph. In addition, it stated that constructing estimator using only relative position information under a general connected graph is an open problem.

Proposition 6.7. *Consider the estimator* (6.54) *and* (6.55) *for agent* $i, i = 2, \cdots, n_l$. *Then, we have* $lim_{t \to \infty} \hat{\kappa}_i = \kappa$.

Proof. First considering the control inputs for the first two agents, we obtain

their dynamics from (6.52) and (6.53) as

$$\begin{aligned}\dot{q}_2-\dot{q}_1=&-\sum_{j\in\mathcal{N}_2}\omega_{2j}(q_2-q_j)+\sum_{k\in\mathcal{N}_1}\omega_{1k}(q_1-q_k)\\&-\sum_{j\in\mathcal{N}_2^l}a_{2j}\left((q_2-q_j)-\hat{\kappa}_2(q_2^*-q_j^*)\right)\\&+\sum_{k\in\mathcal{N}_1^l}a_{1k}\big((q_1-q_k)-\kappa(q_1^*-q_k^*)\big).\end{aligned}\tag{6.58}$$

Define the estimation error for agent 2 by

$$\tilde{\kappa}_2=\hat{\kappa}_2-\kappa,\tag{6.59}$$

and denote the quantity associated with κ and $\hat{\kappa}$ in (6.58) by q_{2r}^*, i.e.,

$$q_{2r}^*\triangleq\sum_{j\in\mathcal{N}_2^l}a_{2j}\hat{\kappa}_2(q_2^*-q_j^*)-\sum_{k\in\mathcal{N}_1^l}a_{1k}\kappa(q_1^*-q_k^*).\tag{6.60}$$

By invoking the fact that $q_1^*-q_k^*=q_1^*-q_2^*+(q_2^*-q_k^*)$, we have

$$\begin{aligned}q_{2r}^*=&\sum_{j\in\mathcal{N}_2^l}a_{2j}\hat{\kappa}_2(q_2^*-q_j^*)-\sum_{k\in\mathcal{N}_1^l}a_{1k}\kappa(q_2^*-q_k^*)\\&-\sum_{k\in\mathcal{N}_1^l}a_{1k}\kappa(q_1^*-q_2^*)\\=&\sum_{j\in\mathcal{N}_1^l\cap\mathcal{N}_2^l}(\hat{\kappa}_2-\kappa)(q_2^*-q_j^*)+a_{21}\hat{\kappa}_2(q_2^*-q_1^*)\\&-\sum_{k\in\mathcal{N}_1^l}a_{1k}\kappa(q_1^*-q_2^*)+\sum_{k\in\mathcal{N}_1^l}a_{1k}\hat{\kappa}_2(q_1^*-q_2^*)\\&-\sum_{k\in\mathcal{N}_1^l}a_{1k}\hat{\kappa}_2(q_1^*-q_2^*)\\=&\tilde{\kappa}_2\sum_{j\in\mathcal{N}_1^l\cap\mathcal{N}_2^l}(q_2^*-q_j^*)+\tilde{\kappa}_2|\mathcal{N}_1^l|(q_1^*-q_2^*)\\&+\hat{\kappa}_2(|\mathcal{N}_1^l|+1)(q_2^*-q_1^*).\end{aligned}\tag{6.61}$$

Substituting (6.61) into (6.58), we get

$$\dot{q}_2-\dot{q}_1=\zeta_2+\mathcal{N}_1^l|(q_1^*-q_2^*)+\sum_{j\in\mathcal{N}_1^l\cap\mathcal{N}_2^l}(q_2^*-q_j^*),\tag{6.62}$$

where ξ_2 and ζ_2 are defined in (6.56) and (6.57). By differentiating $\hat{\kappa}_2$ in (6.54),

and replacing $\dot{q}_2 - \dot{q}_1$ with (6.62), it follows

$$\dot{\hat{\kappa}}_2 = -\tilde{\kappa}_2 \Lambda_2 \| |\mathcal{N}_1^l|(q_1^* - q_2^*) + \sum_{j \in \mathcal{N}_1^l \cap \mathcal{N}_2^l} (q_2^* - q_j^*) \|^2. \tag{6.63}$$

Recall that the scaling parameter is constant, there holds $\dot{\kappa} = 0$. Hence, it is straightforward to have

$$\dot{\tilde{\kappa}}_2 = \dot{\hat{\kappa}}_2 = -\tilde{\kappa}_2 \Lambda_2 \| |\mathcal{N}_1^l|(q_1^* - q_2^*) + \sum_{j \in \mathcal{N}_1^l \cap \mathcal{N}_2^l} (q_2^* - q_j^*) \|^2. \tag{6.64}$$

Therefore, it is easy to know $\tilde{\kappa}_2$ converges to zero exponentially under Assumption 6.5, namely, $\lim_{t\to\infty} \hat{\kappa}_2(t) = \kappa$.

Analogously, define the estimation error for agent $i, i = 3, \cdots, n_l$ by

$$\tilde{\kappa}_i = \hat{\kappa}_i - \hat{\kappa}_{i-1}. \tag{6.65}$$

Similar to the calculations for agent 2, we get

$$\dot{\tilde{\kappa}}_i = -\tilde{\kappa}_i \Lambda_i \| |\mathcal{N}_{i-1}^l|(q_{i-1}^* - q_i^*) + \sum_{j \in \mathcal{N}_{i-1}^l \cap \mathcal{N}_i^l} (q_i^* - q_j^*) \|^2 \tag{6.66}$$

In light of Assumption 6.5, we know $\lim_{t\to\infty} \tilde{\kappa}_i(t) = 0$, which implies $\lim_{t\to\infty} \hat{\kappa}_i(t) = \hat{\kappa}_{i-1}(t)$, $i = 3, \cdots, n_l$. Since $\lim_{t\to\infty} \hat{\kappa}_2 = \kappa$, we can conclude that $\lim_{t\to\infty} \hat{\kappa}_{n_l} = \cdots = \hat{\kappa}_2 = \kappa$. This completes the proof. □

Theorem 6.8. *Suppose the given generic framework $(\mathcal{G}, q^*)$ is universally rigid with an equilibrium stress ω. Under Assumption 6.5, for a group of agents modeled by* (6.1)*, the formation scaling control problem can be solved in the sense of global stability using the proposed estimation-based control laws* (6.52) *and* (6.53)*.*

Proof. Note that (6.53) can be written as

$$\begin{aligned} u_i = &- \sum_{j \in \mathcal{N}_i} \omega_{ij}(q_i - q_j) - \sum_{j \in \mathcal{N}_i^l} a_{ij} \left((q_i - q_j) - \kappa(q_i^* - q_j^*) \right), \\ &+ (\hat{\kappa}_i - \kappa) \sum_{j \in \mathcal{N}_i^l} a_{ij}(q_i^* - q_j^*), \quad i = 2, \cdots, n. \end{aligned} \tag{6.67}$$

Recalling (6.10), the compact form of (6.67) is given by

$$\dot{\tilde{q}} = \left((\Omega + \mathcal{L}_s) \otimes I_d \right) \tilde{q} + \tilde{K}(L_s \otimes I_d) q^*, \tag{6.68}$$

where $\tilde{K}$ is a diagonal matrix defined by $\tilde{K} \triangleq \mathrm{diag}(()\hat{\kappa}_1 - \kappa, \cdots, \hat{\kappa}_{n_l} - \kappa)$. If follows from Theorem 6.2 that the autonomous part of system (6.68) is globally stable. In

view of the fact that q^* is fixed and $\tilde{K}$ globally converge to zero from Proposition 6.7 , by invoking the input-to-state stability theorem [68], we can conclude that

$$\lim_{t\to\infty}(q_i(t)-q_j(t))=\kappa(q_i^*-q_j^*),\ \forall(i,j)\in\mathcal{E}. \tag{6.69}$$

This completes the proof. □

6.6 Simulation results

In this section, we present simulation results to validate the effectiveness of the theoretical results. Consider a generic configuration in $\mathbb{R}^2$, given by

$$q^*=\begin{bmatrix}0 & -0.8 & -2 & -2 & -1\\ 0 & 1.6 & 2 & -2 & -2\end{bmatrix}.$$

With q^*, the prescribed formation shape is depicted in Fig. 6.3. One universally rigid tensegrity framework associated with the configuration q^* is shown in Fig. 6.4, in which the dashed and solid lines represent the cables and struts, respectively.

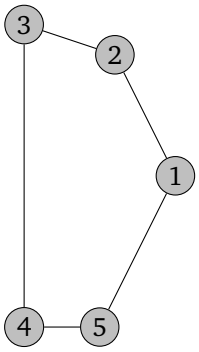


Figure 6.3: *Prescribed formation shape.*

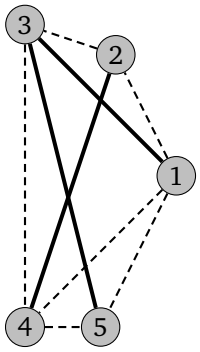


Figure 6.4: *Universally rigid tensegrity framework.*

Correspondingly, the stress matrix has the form

$$\Omega=\begin{bmatrix}27.5 & -45 & 26.75 & -8.25 & -1\\ -45 & 75 & -45 & 15 & 0\\ 26.75 & -45 & 27.1250 & -9.375 & 0.5\\ -8.25 & 15 & -9.3750 & 4.125 & -1.5\\ -1 & 0 & 0.5 & -1.5 & 2\end{bmatrix}.$$

The initial positions for the five agents are randomly chosen as

$$q(0) = \begin{bmatrix} -0.0573 & -1.4483 & -2.053 & -2.3178 & -1.6165 \\ -0.9285 & 2.0435 & 1.3054 & -1.7852 & -1.5231 \end{bmatrix}.$$

6.6.1 Formation scaling control using the proposed control law (6.3)

First, we consider the formation scaling control using only the stress. Let the formation scaling parameter κ be

$$\kappa = \begin{cases} 6, & 0 \leqslant t < 6, \\ 12, & 6 \leqslant t \leqslant 12. \end{cases}$$

To implement the control law(6.3), 2 pairs of nodes, $(1, 2)$ and $(2, 3)$, are chosen to constitute $\mathcal{V}_l$, and consequently $\mathcal{E}_l = \{(, 2), (2, 3)\}$, both of which are marked in blue in Fig. 6.5. To clearly show the variations of the formation shape at different

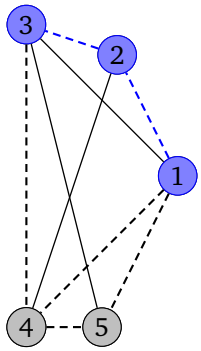


Figure 6.5: *The universally rigid framework with* $\mathcal{V}_l = \{1, 2, 3\}$ *and* $\mathcal{E}_l = \{(1, 2), (2, 3)\}$.

time instants, we design an extra input, $u_e = [18, 0]^T$ for each agent. Since the extra input is constant and the same for each agent, it will not affect the stability of the closed-loop system. Then under the control law (6.3), the formation shapes at $t \in \{0, 2, 4, 6, 8, 10, 12\}$s are sequentially shown in Fig. 6.6, where the initial formation shape is zoomed in on the top. It can be seen that the desired formations with prescribed sizes are achieved for a piecewise constant scaling parameter κ. Fig. 6.7 shows that the scaling length errors, i.e., $\kappa\|q_i^* - q_j^*\| - \|q_i - q_j\|$, where the errors of the cables are plotted in the upper part and struts in the lower part. We can observe from Fig. 6.7 that all the edge lengths converge to their desired ones.

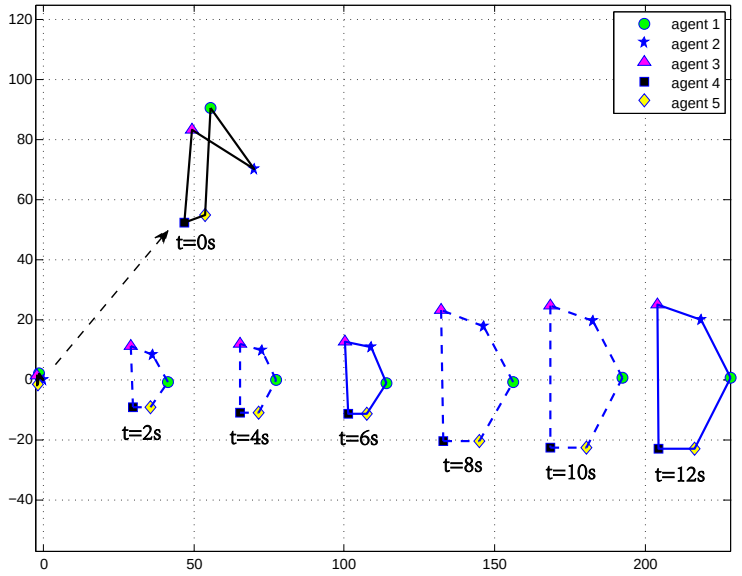


Figure 6.6: *Formation evolution using the control law* (6.3).

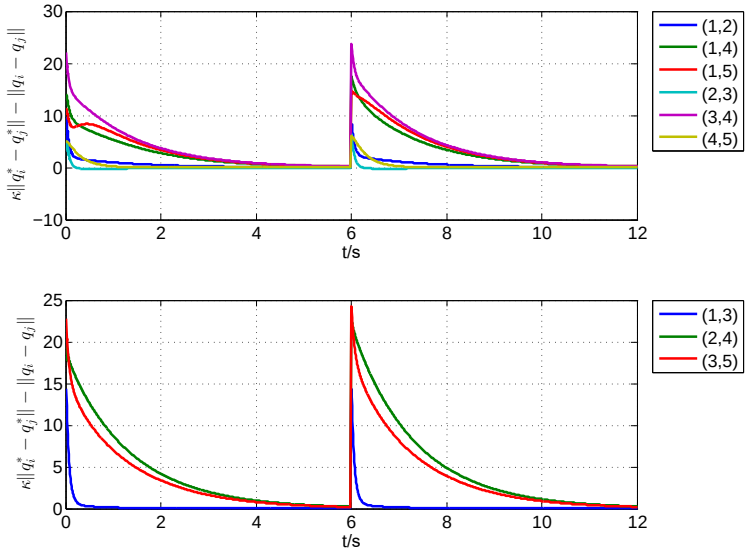


Figure 6.7: *Scaling length errors using the control law* (6.3).

6.6.2 Formation scaling control using the proposed control law (6.28)-(6.30)

We then consider the formation scaling control using the stress and the orthogonal projections. In this case, the formation scaling parameter is defined by

$$\kappa = \begin{cases} 6, & 0 \leqslant t < 6, \\ 12, & 6 \leqslant t < 12, \\ 6, & 12 \leqslant t < 18. \end{cases}$$

According to the principle of choosing $d+1$ pairs of nodes illustrated in Section 6.4, let $\mathcal{E}_l = \{(1,2),(2,3),(1,3)\}$ and $\mathcal{V}_l = \{1,2,3\}$, shown in Fig. 6.8. Following

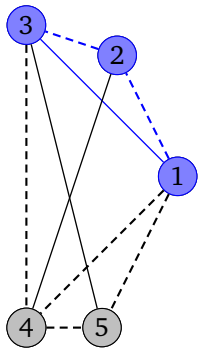


Figure 6.8: *The universally rigid framework with* $\mathcal{E}_l = \{(1,2),(2,3),(3,1)\}$.

the formation scaling control laws (6.28)-(6.30) and the extra input $[18,0]^T$, the formation changes are sequentially shown in Fig. 6.9, in which the initial formation shape is again zoomed in on the top. It can be seen from Fig. 6.9 that the formation expands from $t = 0$s to 12s, and then shrinks until $t = 18$s, which agrees with the setup of the formation scaling parameter κ. The scaling length errors of cables and struts are presented in the upper and lower part of Fig. 6.10, which clearly shows the convergence of the lengths of all edges to the desired ones.

6.6.3 Formation scaling control using the proposed control law (6.52)-(6.53)

In this subsection, we present the numerical simulation results of the proposed estimation-based controller (6.52)-(6.53). The scaling parameter κ is set to be a constant scalar 10 at all times. The subgraph $\mathcal{G}(\mathcal{V}_l, \mathcal{E}_l)$ is constructed the same as Fig. 6.5, in which only agent 1 knows the precise information of κ, while agents 2 and 3 approach the scaling information by estimation. Again, to separate the

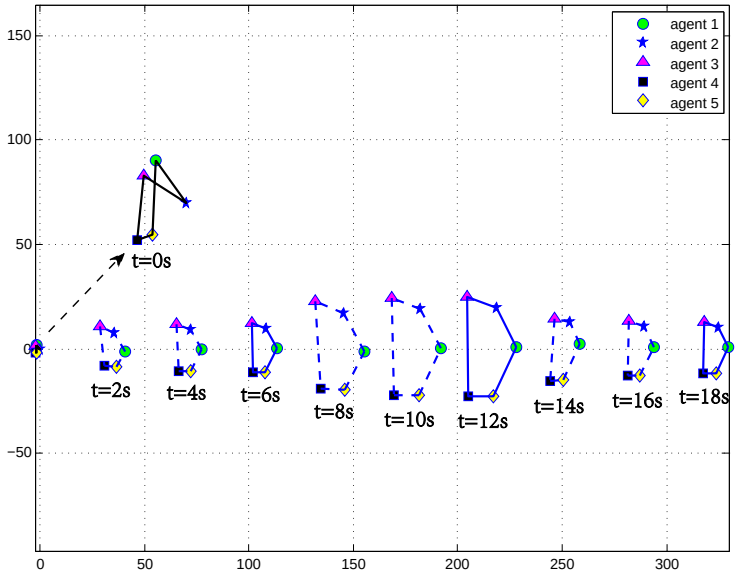


Figure 6.9: *Formation evolution using the control laws* (6.28)-(6.30).

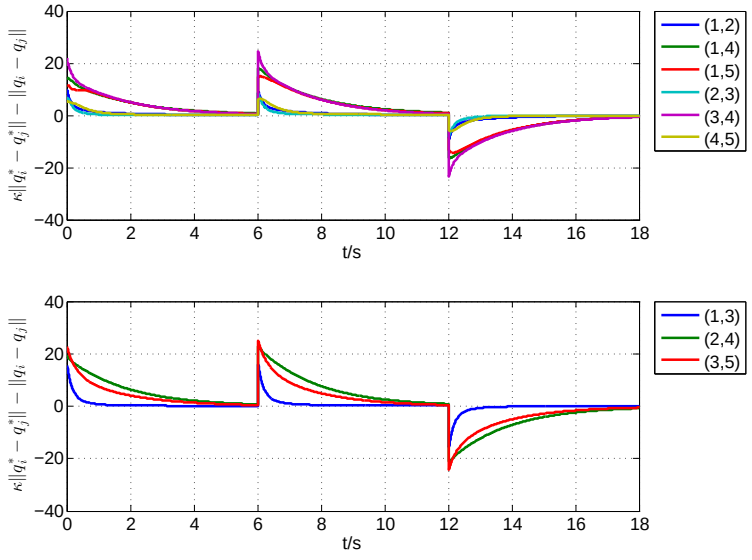


Figure 6.10: *Scaling length errors using the control laws* (6.28)-(6.30).

formation patterns at different time instants, we design an additional input $[2.5, 0]^T$ accompanying the control law (6.52)-(6.53). From Fig. 6.11, we can see that

the formation shape starts from an anomalous status and finally converge to the desired shape. The corresponding scaling length errors are shown in Fig. 6.12, where the errors of cables and struts are presented in the upper and lower part, respectively. It is clear that the errors of all the members converge to zero.

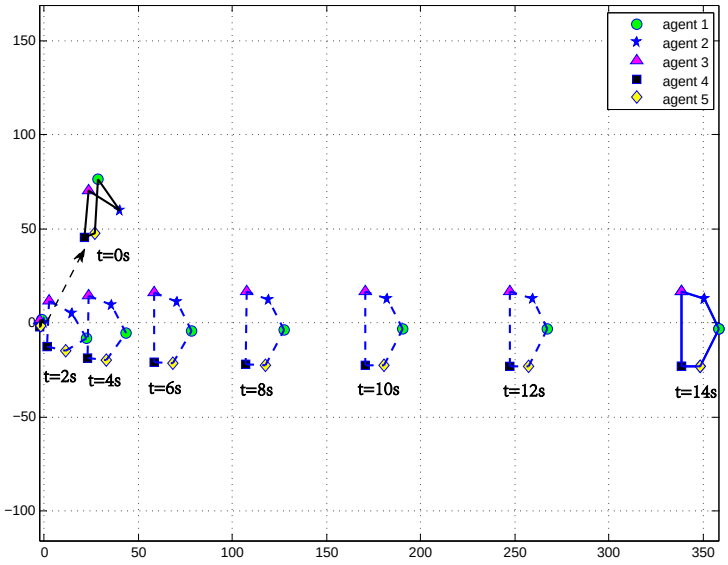


Figure 6.11: *Formation evolution using the control laws* (6.28)-(6.30).

6.7 Concluding remarks

In this chapter, we have addressed the formation scaling problem for multi-agent systems. First, by employing the stress of universally rigid tensegrity frameworks, we have designed distributed control laws to achieve the formation shape with the prescribed size. Then to relax the constraint that the formation scaling parameter has to be known to d pairs of agents in $\mathbb{R}^d$, we have proposed a class of new distributed control laws that utilize the (orthogonal) projections. It has been shown that the desired formation scaling can be achieved under the mild assumption that only one pair of agents knows their desired relative positions. Moreover, we have constructed a relative-position-based estimator to further reduce the number of agents knowing the scaling parameter, so that only one agent is informed of the scaling size of the formation. Relying on the estimator, all the agents can be driven to form the desired formation under the proposed control laws.

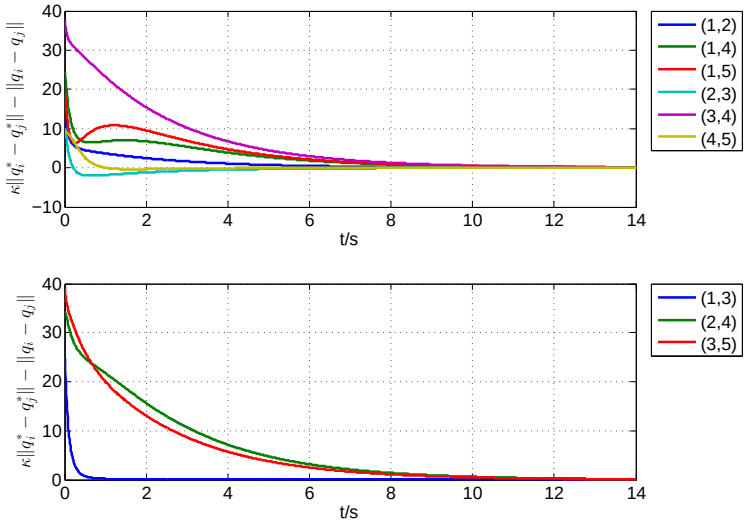


Figure 6.12: *Scaling length errors using the control laws* (6.28)-(6.30).

Chapter 7

Distributed formation tracking using local coordinate systems

This chapter studies the formation tracking problem for multi-agent systems, for which a distributed estimator-controller scheme is designed relying only on the agents' local coordinate systems such that the centroid of the controlled formation tracks a given trajectory. By introducing a gradient descent term into the estimator, the explicit knowledge of the bound of the agents' speed is not necessary in contrast to existing works, and each agent is able to compute the centroid of the whole formation in finite time. Then, based on the centroid estimation, a distributed control algorithm is proposed to render the formation tracking and stabilization errors to converge to zero, respectively. Finally, numerical simulations are carried to validate our proposed framework for solving the formation tracking problem.

7.1 Introduction

Formation control for multi-agent systems has attracted increasing attention from control scientists and engineers due to its broad applications [12, 65, 131]. A central problem is to drive the agents to realize some prescribed formation shape, and such a problem is usually referred to as the formation stabilization problem. In this line of research, formation stabilization for those with different shapes has been investigated, see, for example, circular formation [109, 137], acyclic formation [86], and formations associated with tree graphs [31], minimally rigid graphs [15, 117], and more general rigid graphs [89]. Time-varying formation control problems for linear multi-agent systems under switching directed topologies are also investigated in [32]. In addition, the effects of the measurement inconsistency between neighboring agents on the formation's stability are addressed in [87], where it is shown the resulted distorted formation will move following a closed circular orbit in the plane for any rigid, undirected formation consisting of more than two agents. In [47], the steady-state rigid formation is achieved using an estimator-based gradient control law; in addition, both the static and time-varying mismatched compasses are studied in [82].

Another key problem concerned with formation control for multi-agent systems is formation tracking, which requires to stabilize the prescribed formation, and, additionally, requires that the whole formation follows a given reference trajectory. One commonly reported approach to deal with the formation tracking problem is to use the virtual structure strategy. This technique is built upon assigning a virtual leader to the centroid of the formation to be tracked while achieving the prescribed formation shape [103]. Under this framework, it is shown that the formation tracking can be achieved in finite time by employing the signum function if the virtual leader has directed paths to all the followers [16]. The virtual structure approach is also reported in [95], in which the control and estimation of a common virtual leader is addressed using a consensus algorithm. Integrating the techniques from nonsmooth analysis, collective potential functions and navigation feedback, a distributed algorithm for second-order systems is designed such that the velocity consensus to the virtual leader is achieved [135]. The formation tracking problem can also be solved using the distributed receding horizon control (RHC), for a group of nonholonomic multi-vehicle systems [125]. By applying RHC, some additional tasks, e.g., collision avoidance and consistency, can be realized through adding constraints on the allowed uncertain deviation.

Akin to the virtual structure approach, the leader-follower strategy has also been widely employed to solve formation tracking problems (e.g., [33, 34, 100, 126, 132]). In [100], the formation tracking problem is solved based on formation stabilization with one designated leader among the group. To deal with the intrinsic unknown parameters for a class of nonlinear systems, an adaptive control law using the backstepping technique is proposed in [126], such that all the subsystems' outputs are regulated to achieve consensus tracking. In [132], to compensate the unknown slippage effect of mobile robots, a distributed recursive design strategy involving the adaptive function approximation technique is developed. More recently, the formation tracking problem for second-order multi-agent systems under switching topologies is studied in [34], where one of the agents is set to be the leader to perform tracking tasks. The results therein are also feasible to the target enclosing problem for multi-quadrotor unmanned aerial vehicle systems. In [33], different from the one-leader tracking case, the formation tracking problem with multiple leaders is addressed. To drive the followers to the convex hull spanned by the leaders, a protocol is designed via solving an algebraic Riccati equation.

It should be noted that in the results discussed above, almost all the desired formations are specified by offset vectors with respect to the virtual/real leader or virtual centroid of the group. Those offset vectors are required to be set a priori in a common global coordinate system. In addition, each agent needs to know its corresponding desired offsets as well as its neighbors'. In particular, the agreement reached on the estimations of the virtual centroid is normally different from the

real centroid of the group. However, it is sometimes meaningful to locate the real centroid when performing tasks like the transportation of objects. Furthermore, the approaches developed in these existing works are only applicable to the scenarios where the reference trajectory is an exogenous signal that is independent of the states of the system. To estimate the centroid of the formation, a consensus-based algorithm is proposed in [45], wherein the estimation of each agent is updated by averaging their projections and directions. However, the convergence can be ensured only when the underlying graph is complete. In [52], a tree-based algorithm is adopted to estimate the centroid, while, each agent is required to maintain a list of trees with constant size. Recently, the weighted-centroid tracking problem has been considered in [7, 8, 130]. Unlike the leader-follower structures in which the dynamics of the followers and leaders can be separated, the control objective therein is to track some globally assigned function which is implicitly related to all agents' dynamics. In [7], a controller-observer scheme is designed for the single integrator dynamics such that the weighted centroid of the whole formation follows some given trajectory. As an extension, one additional task function for the formation is introduced in [8]. In [130], a finite-time centroid observer is constructed, and the distance-based control laws are developed by employing rigidity graph theory.

In the present chapter, we consider the formation tracking problem, in which the centroid of the formation moves as the agents move and is unknown to all of the agents. In this case, the problem becomes more challenging due to the inner coupling and conflict between centroid estimation, formation stabilization, and reference tracking. By adopting the feedback term from the gradient descent control, we design a new class of finite-time centroid estimator that is continuously differentiable. Based on the output of the estimator, the proposed distance-based control laws render the convergence to the prescribed formation shape while keeping its centroid following the reference. Compared with the previous work of using virtual/real leader structure, the proposed estimator-controller framework can be implemented in agents' local coordinate systems, which not only increases the robustness to the noises in the sensing signals but also reduces the equipment cost of the overall system. Moreover, the control law in this chapter is more scalable and distributed in the sense that some constraints are removed, including the a priori knowledge of the position information of the reference trajectory [7, 8] and the agents' maximum speed [130]. In addition, the precise knowledge of the time-varying centroid can be obtained in finite time via the proposed smooth centroid estimator, which renders a faster convergence speed than that in [45, 52]. In addition, the centroid estimator in [45] is only valid under complete graphs whereas the one in this chapter can be directly applied to any general undirected graphs.

The rest of this chapter is organized as follows. Section 7.2 introduces the

formation tracking problem and basic concepts of graph rigidity. In Section 7.3, the main results are presented including the estimator-controller scheme and the theoretical analysis. Section 7.4 extends the results to a more general case. The numerical simulations are presented in Section 7.5. Finally, we give the conclusions in Section 7.6.

7.2 Problem formulation

A team of $n > 1$ agents is considered, each of which is characterized by the single integrator dynamics

$$\dot{q}_i^g = u_i^g, \quad i = 1, \cdots, n, \tag{7.1}$$

where $q_i^g \in \mathbb{R}^d$ and $u_i^g \in \mathbb{R}^d$ are, respectively, the position and the control input of mobile agent i with respect to the global coordinate system ${}^g\Sigma$. Each agent i is also assigned with the local coordinate system ${}^i\Sigma$, whose origin is exactly the point q_i^g. In this chapter, the local coordinate systems are assumed to share the same orientations. We use q_j^i to denote agent j's position with respect to ${}^i\Sigma$. This definition also applies to other variables. Note that the local variable q_j^i and the global one q_j^g have the following relationship

$$q_j^g = q_j^i + q_i^g.$$

Here, q_i^g is actually unknown to the agents, since the global coordinate system is introduced only for analysis purposes.

The neighboring relationships between the agents are defined by an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. The interaction relationships among the agents and the reference signal is denoted by matrix $B = \mathrm{diag}(b_1, \cdots, b_n)$, where $b_i = 1$ if agent i has access to the reference signal directly, and $b_i = 0$ otherwise.

Now, we formulate the problem to be investigated in this chapter. On one hand, to achieve a desired shape of the formation, each agent i is required to keep some prescribed distance $d_{ij}, j \in \mathcal{N}_i$, namely, the agents are driven to the following target set

$$\mathcal{T}_d = \{q^g \in \mathbb{R}^{nd} | \, \|q_i^g - q_j^g\| = d_{ij}, \forall (i,j) \in \mathcal{E}\}. \tag{7.2}$$

On the other hand, at the same time, the stabilized formation is guided through the control law such that its centroid q_c^g tracks some smooth reference signal $q_d^g(t) : t \to \mathbb{R}^d$, where the centroid of the formation is defined by

$$q_c^g = \frac{1}{n} \sum_{i=1}^{n} q_i^g. \tag{7.3}$$

Equivalently, the tracking task can be written as

$$\lim_{t\to\infty}(q_c^g - q_d^g(t)) = \mathbf{0}. \tag{7.4}$$

7.3 Formation tracking control

In this section, we first present the estimation algorithm for each agent to obtain the centroid information in finite time. Then, distributed control laws are proposed in local coordinate systems such that the formation tracking problem is solved.

Some useful lemmas are introduced as follows.

Lemma 7.1. *[92]. For an undirected connected graph, the following property holds,*

$$\min_{\substack{x\neq 0\\ \mathbf{1}_n^T x=0}} \frac{x^T L x}{\|x\|^2} = \lambda_2(L),$$

where λ_2 is the algebraic connectivity of the undirected graph, i.e., the smallest non-zero eigenvalue of the Laplacian matrix.

Lemma 7.2. *[124] Let $\xi_1, \cdots, \xi_n \geqslant 0$ and $0 < p \leqslant 1$, then*

$$\sum_{i=1}^{n} \xi_i^p \geqslant \left(\sum_{i=1}^{n} \xi_i\right)^p.$$

Lemma 7.3. *[124]. Suppose that the function $V(t) : [0,\infty) \to [0,\infty)$, is differentiable (the derivative of V(t) at 0 is in fact its right derivative) and*

$$\frac{dV(t)}{dt} \leqslant -KV(t)^\alpha,$$

where $K > 0$ and $0 < \alpha < 1$. Then $V(t)$ will reach zero at some finite time $T_0 \leqslant V(0)^{1-\alpha}/(K(1-\alpha))$ and $V(t) = 0$ for all $t \geqslant T_0$.

Assumption 7.4. The reference signal is bounded, as well as its first derivative, satisfying $\sup_{t>0} \|\dot{q}_d^g(t)\| \leqslant \sigma$. In addition, at least one of the n followers has access to the reference signal.

Remark 7.5. The reference signal is defined locally, namely, the information of the reference known by agent i is q_d^i if agent i has access to the reference signal. And, the local variable can be transformed to the global one through the following equation

$$q_d^g = q_d^i + q_i^g.$$

We first introduce the vector $z^g = [(z_1^g)^T, \cdots, (z_m^g)^T]^T \in \mathbb{R}^{md}$ [36], defined as

$$z^g = (H^\top \otimes I_d) q^g,$$

where $H \in \mathbb{R}^{n\times m}$ is the incidence matrix. Then, it is straightforward to check that z^g lies in the column space of $(H^\top \otimes I_d)$, i.e., $z^g \in \mathrm{col}(H^\top \otimes I_d)$. $z_k^g = q_j^g - q_i^g$ denotes the relative position of agents i and j connected by the kth edge. Note that $z_k^g = z_k^i, i = 1, \cdots, n$, owing to the fact that the local coordinate systems share the same orientation with the global one. Let $\hat{q}_{ci}^i$ be agent i's estimation of the centroid with respect to ${}^i\Sigma$, then

$$\hat{q}_{ci}^g = \hat{q}_{ci}^i + q_i^g, \tag{7.5}$$

where $\hat{q}_{ci}^g$ is agent i's estimation of the centroid with respect to ${}^g\Sigma$.

For controlling an infinitesimally rigid formation shape, we employ the standard quadratic potential function [87]

$$P(q^g) = \frac{1}{4}\sum_{k=1}^{m}(\|z_k^g\|^2 - d_k^2)^2. \tag{7.6}$$

Correspondingly, the gradient of $P(q)$ with respect to q_i^g, denoted by $\nabla_{q_i^g} P(q)$ is given by

$$\nabla_{q_i^g} P(q^g) = \sum_{j\in\mathcal{N}_i}(\|z_k^g\|^2 - d_k^2)(q_i^g - q_j^g) = -\sum_{j\in\mathcal{N}_i}(\|z_k^i\|^2 - d_k^2) z_k^i. \tag{7.7}$$

It can be aggregated as

$$\nabla P(q^g) = R(q^g)^T \phi(q^g), \tag{7.8}$$

where $R(q^g)$ is the rigidity matrix defined in (2.2) and $\phi(q^g)$ is as follows

$$\phi(q^g) = \left[\cdots, \|z_k^g\|^2 - d_k^2, \cdots\right]^T \in \mathbb{R}^m.$$

For achieving the tracking of the centroid to the reference with a prescribed formation shape, we propose the following control law for each agent i with respect to the reference q_d in ${}^i\Sigma$

$$u_i^d = \dot{q}_i^d = -k_p b_i \frac{\hat{q}_{ci}^g - q_d^g}{\delta + \|\hat{q}_{ci}^g - q_d^g\|} - k_s \nabla_{q_i^g} P(q^g), \tag{7.9}$$

where $\delta > 1$ is a constant scalar, and k_p and k_s are positive control gains. It also follows from (7.5) and (7.5) that

$$\hat{q}_{ci}^g - q_d^g = \hat{q}_{ci}^i + q_i^g - (q_d^i + q_i^g) = \hat{q}_{ci}^i - q_d^i.$$

Then, the control law u_i^d can be equivalently written as

$$u_i^d = -k_p b_i \frac{\hat{q}_{ci}^i - q_d^i}{\delta + \|\hat{q}_{ci}^i - q_d^i\|} + k_s \sum_{j\in\mathcal{N}_i} (\|z_k^i\|^2 - d_k^2) z_k^i. \tag{7.10}$$

The first term of the control law (7.10) is responsible for driving the centroid of the formation to track the reference signal, and the second one aims for stabilizing the desired formation. Note that not all the agents need to implement the first term but only those having access to the reference signal q_d^i, which is encoded in the binary variable $b_i \in \{0, 1\}$ as described in Section 7.2. However, all the agents are required to estimate the centroid of the formation through $\hat{q}_{ci}^i$ and to share this information with their neighbors. The dynamics of $\hat{q}_{ci}^i$ will be given later. It can be shown that the estimator can be implemented in a fully distributed manner. For the second term of (7.10), the relative position z_k^i and the distance $\|z_k^i\|$ between neighbors can be measured by sensors in the local coordinate system ${}^i\Sigma$.

The dynamics of $\hat{q}_{ci}^i$ is given by

$$\dot{\hat{q}}_{ci}^i = -k_1 \sum_{j\in\mathcal{N}_i} a_{ij} sig\left(\hat{q}_{ci}^i - \hat{q}_{cj}^i\right)^\rho - k_2 \sum_{j\in\mathcal{N}_i} a_{ij} \frac{\hat{q}_{ci}^i - \hat{q}_{cj}^i}{f_{ij}(\hat{q}_{ci}^i, \hat{q}_{cj}^i)} - k_s \sum_{j\in\mathcal{N}_i} (\|z_k^i\|^2 - d_k^2) z_k^i. \tag{7.11}$$

$f_{ij}(\hat{q}_{ci}^i, \hat{q}_{cj}^i) = \|\hat{q}_{ci}^i - \hat{q}_{cj}^i\| + \left(\sqrt{1 + \|\hat{q}_{ci}^i - \hat{q}_{cj}^i\|} - 1\right)$, and k_1 and k_2 are positive constants, and k_s is defined in (7.9). a_{ij} is the (i, j)th entry of the adjacency matrix A. $\hat{q}_{cj}^i$ is the centroid estimation of agent j with respect to ${}^i\Sigma$. For any $x \in \mathbb{R}$,

$$\mathrm{sig}(x)^\rho = [\mathrm{sign}(x_1)|x_1|^\rho, \cdots, \mathrm{sign}(x_n)|x_n|^\rho]^T, \tag{7.12}$$

where $\mathrm{sign}(\cdot)$ is the signum function and $\rho \in (0, 1)$. For a vector $x \in \mathbb{R}^d$, the function $\mathrm{sig}(x)$ is defined componentwise. It can be shown that the function $\mathrm{sig}(\cdot)^\rho$ is continuous. The initial values for $\hat{q}_{ci}^i$ are chosen such that $\sum_{i=1}^n \hat{q}_{ci}^i(0) = \mathbf{0}$. Note that under the assumption that the orientation of the local coordinate systems are the same, the variable $\hat{q}_{cj}^i$ in (7.11) can be calculated by

$$\hat{q}_{cj}^i = \hat{q}_{cj}^j + q_{ji}^i. \tag{7.13}$$

where the neighbor's estimation $\hat{q}_{cj}^j$ is transmitted to agent i through communication. The relationship between $\hat{q}_{cj}^j$ and $\hat{q}_{cj}^i$ is shown in Fig. 7.1. Therefore, the estimator (7.11) can be implemented locally, and thus the proposed distributed control actions (7.10) and (7.11) can be implemented by only employing local information.

To precisely estimate the centroid, it is required that all the local coordinate systems share the same orientation with the global one. However, it will be shown

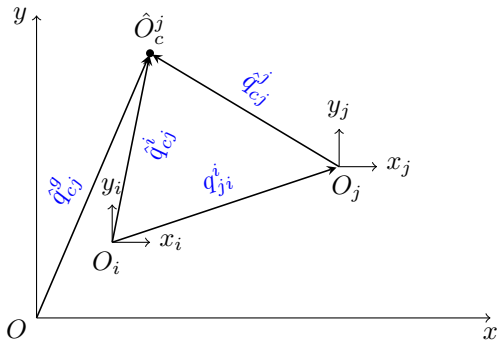


Figure 7.1: *Relationship between* $\hat{q}_{cj}^j$ *and* $\hat{q}_{cj}^i$.

in Section 7.4 that this constraint can be removed.

Now, we present the following main result.

Theorem 7.6. *Suppose the framework* $(\mathcal{G}, q)$ *is minimally and infinitesimally rigid. Under Assumption 7.4, the formation tracking task* (7.4) *is achieved using the control law* (7.20) *for each agent* i *together with the estimator* (7.11)*, if the parameters are chosen such that*

$$k_2 \geqslant \frac{(k_p + \sigma)\sqrt{n}}{\epsilon\sqrt{1 - cos\frac{\pi}{n}}}, \tag{7.14}$$

and

$$k_s > \frac{k_p n}{2\delta}, \tag{7.15}$$

where ϵ *is a positive scalar satisfying* $\epsilon \in (0, 2/3]$*. For an undirected connected graph, the estimation* $\hat{q}_{ci}^g, i = 1, \cdots, n$*, will converge to* q_c^g *in finite time.*

Proof. We carry out the proof in two steps. We first prove the estimation $\hat{q}_{ci}^g, i = 1, \cdots, n$, will converge to q_c^g in finite time. Consider the following equality

$$\hat{q}_{ci}^i - \hat{q}_{cj}^i = \hat{q}_{ci}^i - \hat{q}_{cj}^j - q_{ji}^i = -\left(\hat{q}_{cj}^j - \hat{q}_{ci}^i - q_{ij}^i\right).$$

In view of the definition (7.12), we have

$$sig\left(\hat{q}_{ci}^i - \hat{q}_{cj}^j - q_{ji}^i\right)^\rho = -sig\left(\hat{q}_{cj}^j - \hat{q}_{ci}^i - q_{ij}^i\right)^\rho.$$

Note that for an undirected graph, $a_{ij} = a_{ji}$, thus it follows

$$\sum_{i=1}^{n} \dot{\hat{q}}_{ci}^i = \mathbf{0}. \tag{7.16}$$

Define the estimation error with respect to the global coordinate system ${}^g\Sigma$ as

$$\tilde{q}_{ci}^g = \hat{q}_{ci}^g - q_c^g, \quad i = 1, \cdots, n.$$

Now, consider the following Lyapunov function candidate

$$V_1 = \frac{1}{2}\sum_{i=1}^{n} \|\hat{q}_{ci}^g - q_c^g\|^2 = \frac{1}{2}\sum_{i=1}^{n} (\tilde{q}_{ci}^g)^T (\tilde{q}_{ci}^g), \tag{7.17}$$

where the centroid q_c^g is defined in (7.3). The time derivate of V_1 is given by

$$\dot{V}_1 = \sum_{i=1}^{n} (\tilde{q}_{ci}^g)^T \left(\dot{\hat{q}}_{ci}^i + \dot{q}_i^g - \dot{q}_c^g \right). \tag{7.18}$$

By combining (7.16) and the initial conditions for the estimator, i.e., $\sum_{i=1}^n \hat{q}_{ci}^i(0) = \mathbf{0}$, it follows $\sum_{i=1}^n \hat{q}_{ci}^i(t) = \mathbf{0}, \forall t > 0$. Consequently, recalling (7.5), we have $\sum_{i=1}^n \hat{q}_{ci}^g = \sum_{i=1}^n q_i^g = n q_c^g$, and thus

$$\sum_{i=1}^{n} (\tilde{q}_{ci}^g)^T \dot{q}_c^g = \sum_{i=1}^{n} \left(\hat{q}_{ci}^g - q_c^g\right)^T \dot{q}_c^g = \left(\sum_{i=1}^{n} \hat{q}_{ci}^g - \sum_{i=1}^{n} q_c^g \right)^T \dot{q}_c^g = 0. \tag{7.19}$$

From the geometrical relationship, we know $q_i^d = -q_d^i$, and $q_d^i = q_d^g - q_i^g$. Then, in view of the system model (7.1), the control input with respect to the global coordinate system ${}^g\Sigma$, i.e., u_i^g is

$$\dot{q}_i^g = u_i^g = \dot{q}_d^g - k_p b_i \frac{\hat{q}_{ci}^i - q_d^i}{\delta + \|\hat{q}_{ci}^i - q_d^i\|} + k_s \sum_{j \in \mathcal{N}_i} (\|z_k^i\|^2 - d_k^2) z_k^i. \tag{7.20}$$

Then substituting (7.19) and (7.20) into (7.18), together with the facts that $q_{ji}^i = q_j^g - q_i^g$ and $\hat{q}_{ci}^g = \hat{q}_{ci}^i + q_i^g$, we have

$$\begin{aligned}
\dot{V}_1 = &- k_1 \sum_{i=1}^{n} (\tilde{q}_{ci}^g)^T \sum_{j \in \mathcal{N}_i} a_{ij} sig\left(\hat{q}_{ci}^g - \hat{q}_{cj}^g\right)^\rho - k_2 \sum_{i=1}^{n} (\tilde{q}_{ci}^g)^T \sum_{j \in \mathcal{N}_i} a_{ij} \frac{\hat{q}_{ci}^g - \hat{q}_{cj}^g}{f_{ij}(\hat{q}_{ci}^g, \hat{q}_{cj}^g)} \\
&- k_p \sum_{i=1}^{n} b_i (\tilde{q}_{ci}^g)^T \left(\frac{\hat{q}_{ci}^i - q_d^i}{\delta + \|\hat{q}_{ci}^i - q_d^i\|} \right) + \sum_{i=1}^{n} (\tilde{q}_{ci}^g)^T \dot{q}_d^g,
\end{aligned}$$

where $f_{ij}(\hat{q}_{ci}^g, \hat{q}_{cj}^g) = \|\hat{q}_{ci}^g - \hat{q}_{cj}^g\| + \left(\sqrt{1 + \|\hat{q}_{ci}^g - \hat{q}_{cj}^g\|} - 1\right)$.

Note that

$$\hat{q}_{ci}^g - \hat{q}_{cj}^g = \hat{q}_{ci}^g - q_c^g - (\hat{q}_{cj}^g - q_c^g) = \tilde{q}_{ci}^g - \tilde{q}_{cj}^g.$$

When $g(x_i - x_j)$ is an odd function, under an undirected graph, we have $\sum_{i,j} a_{ij} x_i g(x_i -$

$x_j) = \frac{1}{2}\sum_{i,j} a_{ij}(x_i - x_j)g(x_i - x_j)$. Therefore, $\dot{V}_1$ satisfies

$$\begin{aligned}\dot{V}_1 \leqslant & -\frac{k_1}{2}\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i} a_{ij}\left(\sum_{k=1}^{d}\left|\tilde{q}^g_{ci(k)} - \tilde{q}^g_{cj(k)}\right|^{\rho+1}\right) \\ & -\frac{k_2}{2}\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i} a_{ij}\frac{\left(\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\right)^T\left(\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\right)}{f_{ij}(\tilde{q}^g_{ci} - \tilde{q}^g_{cj})} \\ & + k_p\sum_{i=1}^{n} b_i\|\tilde{q}^g_{ci}\|\left(\frac{\|\hat{q}^i_{ci} - q^i_d\|}{\delta + \|\hat{q}^i_{ci} - q^i_d\|}\right) + \|(\tilde{q}^g_c)^T(\mathbf{1}_n \otimes \dot{q}^g_d)\|, \end{aligned} \tag{7.21}$$

where $f_{ij}(\tilde{q}^g_{ci}, \tilde{q}^g_{cj}) = \|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\| + \left(\sqrt{1 + \|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\|} - 1\right)$, and $\tilde{q}^g_{ci(k)}$ denotes the kth entry of the vector $\tilde{q}^g_{ci}$. In addition, we have

$$\frac{\left(\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\right)^T\left(\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\right)}{\|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\| + \left(\sqrt{1 + \|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\|} - 1\right)} \geqslant \epsilon\|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\|, \tag{7.22}$$

where $\epsilon \in (0, 2/3]$. The proof of (7.22) is given in Appendix. It is also straightforward to know

$$\frac{\|\hat{q}^i_{ci} - q^i_d\|}{\delta + \|\hat{q}^i_{ci} - q^i_d\|} < 1. \tag{7.23}$$

Substituting (7.22) and (7.23) into (7.21), we obtain

$$\begin{aligned}\dot{V}_1 \leqslant & -\frac{k_1}{2}\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i} a_{ij}\sum_{k=1}^{d}\left|\tilde{q}^g_{ci(k)} - \tilde{q}^g_{cj(k)}\right|^{\rho+1} - \frac{k_2}{2}\epsilon\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i} a_{ij}\|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\| \\ & + k_p\sum_{i=1}^{n} b_i\|\tilde{q}^g_{ci}\| + \sqrt{n}\sigma\|\tilde{q}^g_c\|. \end{aligned} \tag{7.24}$$

It is clear that

$$\sum_{i=1}^{n} b_i\|\tilde{q}^g_{ci}\| = \|(B\mathbf{1}_n)^T\tilde{q}^g_c\| \leqslant \|B\mathbf{1}_n\|\|\tilde{q}^g_c\| \leqslant \sqrt{n}\|\tilde{q}^g_c\|. \tag{7.25}$$

In light of Lemma 7.2 and Lemma 7.1, it yields

$$\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i} a_{ij}\|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\| \geqslant \left(\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i} a^2_{ij}\|\tilde{q}^g_{ci} - \tilde{q}^g_{cj}\|^2\right)^{\frac{1}{2}} \geqslant \sqrt{2\lambda_2(L_{A_s})}\|\tilde{q}^g_c\|, \tag{7.26}$$

where $A_s = [a^2_{ij}] \in \mathbb{R}^{n\times n}$ is an adjacency matrix and $\tilde{q}^g_c = [(\tilde{q}^g_{c1})^T, \cdots, (\tilde{q}^g_{cn})^T]^T$.

From [84], we know that $\lambda_2(L_{A_s}) \geqslant 2e(\mathcal{G})(1-\cos\frac{\pi}{n})$, where $e(\mathcal{G})$ is the *edge connectivity* of the underlying graph $\mathcal{G}$, i.e., the minimal number of those edges whose removal would result in losing connectivity of the graph $\mathcal{G}$. Obviously, for an undirected connected graph, $e(\mathcal{G}) > 1$. Under the condition (7.14), and combining (7.25) and (7.26), we have

$$\begin{aligned} &-\frac{k_2}{2}\epsilon\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i}a_{ij}\|\tilde{q}^g_{ci}-\tilde{q}_{cj}\|+k_p\sum_{i=1}^{n}b_i\|\tilde{q}^g_{ci}\|+\sqrt{n}\sigma\|\tilde{q}^g_c\| \\ \leqslant &-\frac{k_2}{\sqrt{2}}\epsilon\sqrt{\lambda_2(L_{A_s})}\|\tilde{q}^g_c\|+k_p\sqrt{n}\|\tilde{q}^g_c\|+\sqrt{n}\sigma\|\tilde{q}^g_c\|\leqslant 0. \end{aligned} \tag{7.27}$$

By Substituting (7.27) into (7.24), and applying Lemma 7.2, it can be obtained that

$$\begin{aligned} \dot{V}_1 &\leqslant -\frac{k_1}{2}\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i}a_{ij}\left(\sum_{k=1}^{d}\left|\tilde{q}^g_{ci(k)}-\tilde{q}^g_{cj(k)}\right|^{\rho+1}\right) \\ &\leqslant -\frac{k_1}{2}\sum_{i,j}a_{ij}\left[\sum_{k=1}^{d}(\tilde{q}^g_{ci(k)}-\tilde{q}^g_{cj(k)})^2\right]^{\frac{\rho+1}{2}} \\ &\leqslant -\frac{k_1}{2}(2\tilde{q}^g_c L_{A_\rho}\tilde{q}^g_c)^{\frac{1+\rho}{2}}, \end{aligned}$$

where $A_\rho = [a_{ij}^{\frac{2}{\rho+1}}] \in \mathbb{R}^{n\times n}$. From Lemma 7.1, we have

$$\dot{V}_1(t) \leqslant -\frac{k_1}{2}\left[2\lambda_2(L_{A_\rho})\right]^{\frac{1+\rho}{2}}\left(\|\tilde{q}_c\|^2\right)^{\frac{1+\rho}{2}} \leqslant -k_1 2^\rho[\lambda_2(L_{A_\rho})]^{\frac{1+\rho}{2}}V_1(t)^{\frac{1+\rho}{2}}.$$

Consequently, we conclude from Lemma 7.3 that

$$\lim_{t\geqslant T_0}(\hat{q}^g_{ci}(t)-q^g_c(t)) = \mathbf{0}, \tag{7.28}$$

where $T_0 \leqslant V_1(0)/k_1(1-\rho)2^{\rho-1}\left[\lambda_2(L_{A_\rho})\right]^{\frac{1+\rho}{2}}$. This completes the proof that $\hat{q}^g_{ci}$ converge to q^g_c in finite time.

Now we prove that the tracking errors converge to zero.

We will prove in Appendix B.2 that, by applying the proposed estimator and control algorithms, the state of the closed-loop system, i.e., $\tilde{q}^g_d$, is bounded in $(0, T_0]$. In addition, the states q^g_i, the control signal u^g_i and the estimation variable $\hat{q}^i_{ci}$ are also bounded in finite time given bounded initial states $q^g_i(0)$ and $\hat{q}^g_{ci}(0)$.

Now we are in the position to show the effectiveness of our control laws in achieving estimation based average tracking. Note that control laws (7.9) can be

written in a stacked form as

$$u^g = \mathbf{1}_n \otimes \dot{q}_d^g - k_p \left(B\hat{Q}_\delta \otimes I_d \right) (\hat{q}_c^g - \mathbf{1}_n \otimes q_d^g) - k_s \nabla P(q^g), \tag{7.29}$$

where

$$\hat{Q}_\delta = \begin{bmatrix} \frac{1}{\delta + \|\hat{q}_{c1}^g - q_d^g\|} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \cdots & \cdots & \frac{1}{\delta + \|\hat{q}_{cn}^g - q_d^g\|} \end{bmatrix}.$$

It is easy to show the matrix Q_δ is positive definite. From Theorem 7.6, when $t \geqslant T_0$, $\hat{q}_{ci}^g$ can be replaced by q_c^g. Then, u^g becomes

$$u^g = \mathbf{1}_n \otimes \dot{q}_d^g - k_p \left(BQ_\delta \otimes I_d \right) \left[\mathbf{1}_n \otimes (q_c^g - q_d^g) \right] - k_s \nabla P(q^g), \tag{7.30}$$

where

$$Q_\delta = \frac{1}{\delta + \|q_c^g - q_d^g\|} I_n.$$

Multiplying both sides of (7.30) by $(\mathbf{1}_n^T \otimes I_d)$, we have

$$(\mathbf{1}_n^T \otimes I_d)(u^g - \mathbf{1}_n \otimes \dot{q}_d^g) = -k_p [\left(\mathbf{1}_n^T BQ_\delta \otimes I_d \right) \left[\mathbf{1}_n \otimes (q_c^g - q_d^g) \right] - k_s (\mathbf{1}_n^T \otimes I_d) \nabla P(q^g). \tag{7.31}$$

When $t \geqslant T_0$, the Lyapunov function candidate is chosen as

$$V = \frac{1}{2} (\tilde{q}_d^g)^T (\tilde{q}_d^g) + P(q^g), \tag{7.32}$$

where $\tilde{q}_d^g \triangleq q_c^g - q_d^g$ is the centroid tracking error. The derivative of V is given by

$$\dot{V} = (\tilde{q}_d^g)^T (\dot{q}_c^g - \dot{q}_d^g) + \nabla P(q^g)^T \dot{q}^g. \tag{7.33}$$

Note that

$$\dot{q}_c^g = \frac{1}{n} \sum_{i=1}^{n} \dot{q}_i^g = \frac{1}{n} (\mathbf{1}_n^T \otimes I_d) \dot{q}^g = \frac{1}{n} (\mathbf{1}_n^T \otimes I_d) u^g. \tag{7.34}$$

Then it follows

$$\dot{q}_c^g - \dot{q}_d^g = -k_p \left(BQ_\delta \otimes I_d \right) \left[\mathbf{1}_n \otimes (q_c^g - q_d^g) \right] - k_s \nabla P(q^g).$$

Substituting (7.29), (7.31), and (7.34) into (7.33), we get

$$\begin{aligned}\dot{V} =& -\frac{k_p}{n}(\tilde{q}_d^g)^T\left(\mathbf{1}_n^T B Q_\delta \mathbf{1}_n \otimes I_d\right)\tilde{q}_d^g - \frac{k_s}{n}(\tilde{q}_d^g)^T\left(\mathbf{1}_n^T \otimes I_d\right)\nabla P(q^g)\\ &- k_p \nabla P(q^g)^T\left(BQ_\delta \otimes I_d\right)\left(\mathbf{1}_n \otimes \tilde{q}_d^g\right) - k_s(\nabla P(q^g))^T\nabla P(q^g)\\ &+ (\nabla P(q^g))^T(\mathbf{1}_n \otimes \dot{q}_d^g).\end{aligned}$$

From (7.8), we have $(\tilde{q}_d^g)^T\left(\mathbf{1}_n^T \otimes I_d\right)\nabla P(q^g) = \mathbf{0}$ and $(\nabla P(q^g))^T(\mathbf{1}_n \otimes \dot{q}_d^g) = \mathbf{0}$ due to the fact that $R(q^g)\left(\mathbf{1}_n^T \otimes I_d\right) = \mathbf{0}$. In light of (7.34), we obtain that

$$\begin{aligned}\dot{V} \leqslant& -\frac{k_p}{n}\sum_i^n \frac{b_i}{\delta + \|\tilde{q}_d^g\|}\|\tilde{q}_d^g\|^2 - k_s\|\nabla P(q^g)\|^2 + \frac{k_p\sqrt{n}}{\delta + \|\tilde{q}_d^g\|}\|\tilde{q}_d^g\|\|\nabla P(q^g)\|\\ \leqslant& -\begin{bmatrix}\|\tilde{q}_d^g\|\\ \|\nabla P(q^g)\|\end{bmatrix}^T Q \begin{bmatrix}\|\tilde{q}_d^g\|\\ \|\nabla P(q^g)\|\end{bmatrix},\end{aligned} \tag{7.35}$$

where

$$Q = \begin{bmatrix}\frac{k_p}{\delta+\|\tilde{q}_d^g\|} & -\frac{k_p\sqrt{n}}{2\left(\delta+\|\tilde{q}_d^g\|\right)}\\ -\frac{k_p\sqrt{n}}{2\left(\delta+\|\tilde{q}_d^g\|\right)} & k_s\end{bmatrix}.$$

It can be checked that the matrix Q is positive definite when the control gains k_p and k_s are chosen such that

$$k_s > \frac{k_p n}{4(\delta + \|\tilde{q}_d^g\|)},$$

which naturally holds if the condition (7.15) is satisfied.

Then, we know $\tilde{q}_d^g$ is bounded, which implies q_c^g, and thus q_i^g are bounded under Assumption 7.4. It follows from (7.7) that $\nabla P(q^g)$ is bounded. Hence, the control input (7.20), i.e., the velocity $\dot{q}_i^g$ is bounded. Together with Assumption 7.4, we know $\dot{\tilde{q}}_d^g$ and $\nabla\dot{P}(q^g)$ are bounded. Therefore, taking the time derivative of (7.35), we know $\ddot{V}$ is bounded. It can be concluded from the Barbalat's Lemma [68] that $\dot{V} \to 0$, as $t \to \infty$, i.e., $\tilde{q}_d^g \to \mathbf{0}$ and $R(q^g)^T\phi(q^g) \to \mathbf{0}$, as $t \to \infty$, which implies the tracking objective is achieved. For a minimally and infinitesimally rigid framework, the rigidity matrix $R(q^g)$ is full row rank. Hence, we have $\phi(q^g) \to \mathbf{0}$, namely, all the agents converge to the target set $\mathcal{T}_d$ in (7.2)

The proof of Theorem 7.6 is completed. □

Remark 7.7. It is worth noting that u_i^g is employed in (7.20) for purposes of theoretical analysis. While the control input to be implemented in practice is (7.10) and (7.11).

Remark 7.8. The assumption that the framework is minimally and infinitesimally

rigid can be relaxed to that the framework is only infinitesimally rigid [87, 89]. In view of the developed techniques for analyzing non-minimally infinitesimally rigid frameworks in [87], the proof is omitted here for the sake of brevity.

Remark 7.9. In this chapter, to implement the centroid estimator (7.11), the underlying communication graph is only required to be a general undirected graph, which could be the same one as required for formation shape control. To explore whether the condition of an undirected graph is necessary for the convergence of the proposed estimator, we carried out a numerical example with three agents under directed graphs. The results show that all the estimation errors will reach a consensus, but not at zero, which implies the proposed estimator fails in directed graphs, even in the simplest case of three agents. Focusing on the second term of (7.10), i.e., the distance-based formation controller, there has been progress for achieving such formations by employing directed graphs using the notion of persistency [56].

7.4 Extension to more general scenarios

The results in Section 7.3 are obtained under the condition that the local coordinate systems ${}^i\Sigma, i = 1, \cdots, n$, have the same orientations with the global coordinate system ${}^g\Sigma$. However, this constraint may not be satisfied in some applications. In this section, we consider a more general case where the orientations of the local coordinate systems differ from the global one, which is depicted in Fig. 7.2.

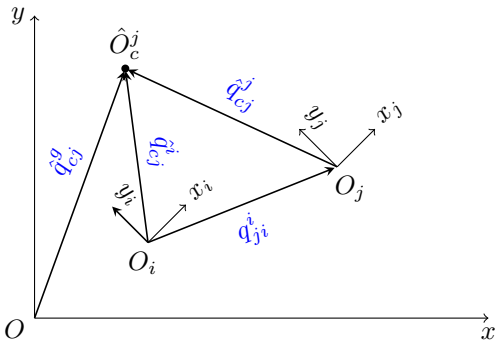


Figure 7.2: *Different orientations between local coordinate systems and the global one.*

From Fig. 7.2, we have

$$\hat{q}_{ci}^g = R_i^g \hat{q}_{ci}^i + q_i^g, \tag{7.36}$$

where $R_i^g \in SO(d)$ is a constant rotation matrix. The centroid estimator is now

given by

$$\dot{\hat{q}}_{ci}^i = -k_1 \sum_{j\in\mathcal{N}_i} a_{ij} sig\left(\hat{q}_{ci}^i - \hat{q}_{cj}^i\right)^\rho - k_2 \sum_{j\in\mathcal{N}_i} a_{ij} \frac{\hat{q}_{ci}^i - \hat{q}_{cj}^i}{f_{ij}(\hat{q}_{ci}^i, \hat{q}_{cj}^i)} + k_s \sum_{j\in\mathcal{N}_i} (\|z_k^i\|^2 - d_k^2) q_{ij}^i, \tag{7.37}$$

where k_1 and k_2 are chosen according to Theorem 7.6, and $f_{ij}(\hat{q}_{ci}^i, \hat{q}_{cj}^i) = \|\hat{q}_{ci}^i - \hat{q}_{cj}^i\| + \left(\sqrt{1 + \|\hat{q}_{ci}^i - \hat{q}_{cj}^i\|} - 1\right)$. Again, the variable $\hat{q}_{cj}^i$ is obtained through $\hat{q}_{cj}^i = \hat{q}_{cj}^j + q_{ji}^i$, where q_{ji}^i is the relative position between O_j and O_i with respect to ${}^i\Sigma$, which can be measured by agent i locally. It is worth noting that the variable q_{ji}^i employed in (7.37) is measured in the local coordinate system ${}^i\Sigma$, allowing the distinction of the orientations between the local coordinate systems and the global one, since the value of q_{ji}^i will not be altered in that case. Summing up both sides of (7.36), we have

$$\sum_{i=1}^n \hat{q}_{ci}^g = \sum_{i=1}^n R_i^g \hat{q}_{ci}^i + \sum_{i=1}^n q_i^g. \tag{7.38}$$

Since the local coordinate systems have the same orientation, we obtain that $R_i^g = R_j^g, i, j = 1, \cdots, n$. By denoting $R_l^g \triangleq R_i^g$, (7.38) can be written as

$$\sum_{i=1}^n \hat{q}_{ci}^g = R_l^g \sum_{i=1}^n \hat{q}_{ci}^i + \sum_{i=1}^n q_i^g.$$

Considering the estimator (7.37), we know $\sum_{i=1}^n \dot{\hat{q}}_{ci}^i = \mathbf{0}$. Then, in combination with the initial condition $\sum_{i=1}^n \hat{q}_{ci}^i(0) = \mathbf{0}$, it yields $\sum_{i=1}^n \hat{q}_{ci}^g = \sum_{i=1}^n q_i^g = n q_c^g$. Following the similar steps as in Section 7.3, it can be shown that $\hat{q}_{ci}^g$ converges to q_c^g in finite time.

In this scenario, the control law is designed as

$$u_i^d = \dot{q}_i^d = -k_p b_i \frac{\hat{q}_{ci}^i - q_d^i}{\delta' + \|\hat{q}_{ci}^i - q_d^i\|} - k_s \sum_{j\in\mathcal{N}_i} (\|z_k^i\|^2 - d_k^2) q_{ij}^i, \tag{7.39}$$

where $\delta' > 1$ is a constant scalar, and k_p and k_s are chosen such that (7.15) holds. It can be seen that (7.39) has the same form as that of (7.10), while the value of q_{ij}^i here differs from q_{ij}^g due to orientation difference between local and global coordinate systems.

Following the similar proof steps as in Section 7.3, the centroid of the formation can be proved to converge to the reference signal. The details of the proof are omitted in this section to avoid repetition.

Remark 7.10. For the scenario where the orientations of the local coordinate systems are different from each other, it can be shown that the estimator and the

control law remain to be the same as (7.37) and (7.39) without loss of stability. While the variable $\hat{q}^i_{cj}$ in (7.37) is now calculated by $\hat{q}^i_{cj} = R^i_j \hat{q}^j_{cj} + q^i_{ji}$, where R^i_j is the rotation matrix with respect to frames i and j. Note that the rotation matrix depends only on the relative rotation angle between local coordinate systems ${}^i\Sigma$ and ${}^j\Sigma$. Therefore, with the sensing capability of rotation angles with respect to neighbors, the proposed control framework is still applicable to the case when the orientations of local systems are not necessarily equal to each other. For those systems without such sensing capability, estimation techniques are reported in recent works, e.g., [82, 90].

7.5 Simulations

To validate the theoretical results, we consider the formation tracking problem for eight agents with dynamics (7.1), whose interaction relationship is given in Fig. 7.3.

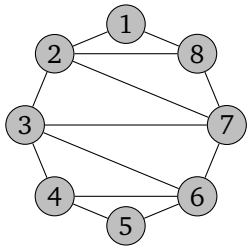


Figure 7.3: *The prescribed framework of the eight agents–regular octagon.*

Take the initial positions for the eight agents as, respectively, $[1,3]^T$, $[-1,1]^T$, $[-3,0.2]^T$, $[-2.7,-0.2]^T$, $[0.2,-4]^T$, $[2,-2]^T$, $[1,-0.5]^T$, $[1,2]^T$. The reference signal is given by $\sigma_d(t) = [6*t, 5*\cos(t)]^T$. Let the initial values of the centroid estimation be $\hat{q}^i_{ci}(0) = [4.5-i, i-4.5]^T$, $i = 1,\cdots,8$, which satisfies the condition that $\sum_{i=1}^{8} \hat{q}^i_{ci}(0) = \mathbf{0}$. The control parameters are chosen as $\rho = 1/4$, $k_1 = 3$, $k_2 = 12$, $k_p = 9$ and $k_s = 13$.

The simulation results are shown in Fig. 7.4 – 7.6, where we use $x^{(i)}, i = 1,2$, to denote the ith component of vector x. The formation geometries of the agents at $t \in \{0;1;2;3;4;5\}$s are shown in Fig. 7.4, where the red cross and the solid black line represent the centroid of the whole formation shape and the centroid's reference trajectory, respectively. From Fig. 7.4 we can see that the prescribed regular octagon is achieved with its centroid converging to the reference trajectory. The convergence of the centroid tracking error is further shown in Fig. 7.5. Fig. 7.6 depicts the centroid estimation errors associated with agents $1, 3, 5,$ and 7 as

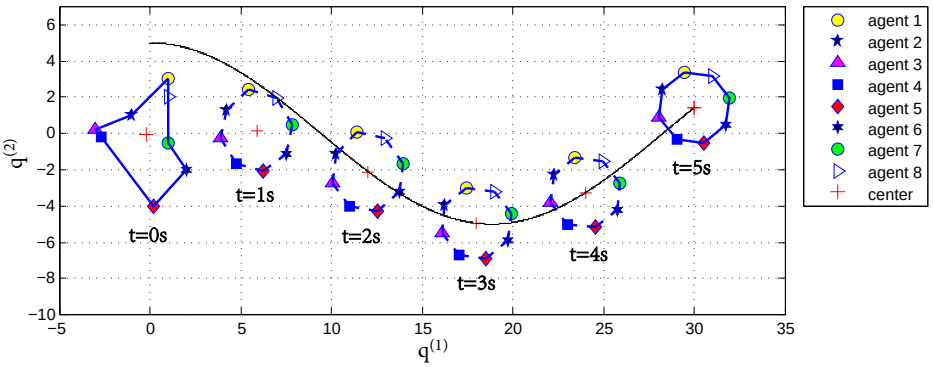


Figure 7.4: *Formation shape evolution.*

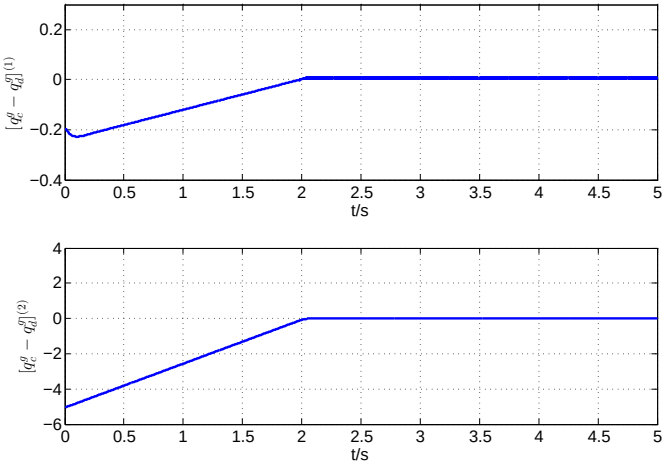


Figure 7.5: *Centroid tracking error* $q_c^g - q_d^g$.

representatives, which demonstrates the effectiveness of the proposed finite-time estimator.

7.6 Concluding remarks

In this chapter, we have investigated the formation tracking problem using local coordinate systems. By introducing a new gradient descent term, an alternative estimator is designed for each agent such that they can obtain the precise knowledge of the formation's centroid in finite time. Moreover, we propose a distributed estimator-controller strategy, which can be implemented using only agents' local

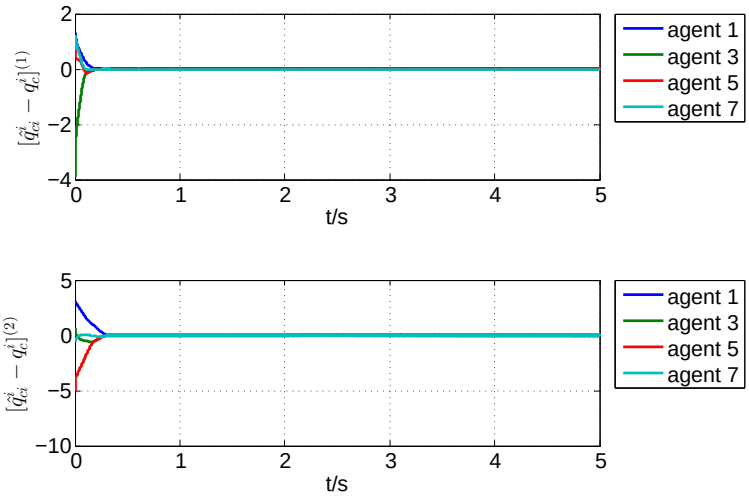


Figure 7.6: *Centroid estimation error* $\hat{q}_{ci}^i - q_c^i$.

coordinate systems.

Chapter 8

Conclusions and future work

This chapter summarizes the main results of this thesis and indicates the possible future research directions.

8.1 Conclusions

This thesis has addressed the problem of constructing tensegrity frameworks and has discussed the related applications in formation control for multi-agent systems. We have discussed how to design tensegrity frameworks in two situations: one is to assign different types of members to chosen pairs of vertices; the other is to grow existing tensegrity frameworks through merging. To fully utilize tensegrity frameworks, we have also explored stress-based formation controls in different scenarios. Now, we provide the specific conclusions for each technical chapter.

In Chapter 3 we have studied the merging of infinitesimally rigid and rigid tensegrity frameworks in the plane, respectively. In the case of merging infinitesimally rigid tensegrity frameworks, we have shown that infinitesimal rigidity can be preserved by guaranteeing the existence of the proper self-stress in combination with the infinitesimal rigidity of the corresponding bar framework. In addition, we further discussed the influence on the pre-existing members caused by merging. By looking at the sign of the new stress, it has been verified that the types of the members, viz. cable or strut, can also be preserved. With respect to rigid tensegrity frameworks, we have proved the existence of appropriate linking members assuring the rigidity of the combined tensegrity framework. We have also proposed a distance perturbation method to determine the type of the new members. We have presented the explicit expression on assigning the type of the fourth member based on rigidity matrix.

Chapter 4 has extended the results in Chapter 3 from local rigidity to super stability. In Chapter 4, we start with the problem of how to conduct vertex addition and edge splitting operations on super stable tensegrity frameworks in parallel to bar frameworks. By inserting a set of members, it has been proven that the obtained tensegrity framework is super stable as well. We have also clarified the type of the new members depending on the positions of the new vertex with respect to the existing vertices. In addition, a numerical algorithm has been proposed to specify

the values of the stresses of the new members. Analogously, we have illustrated that super stability is also preserved under edge splitting. We then proceed to another class of approaches used for growing tensegrity frameworks, i.e., merging. When the super stable tensegrity frameworks share at least $d+1$ vertices in d-dimensional Euclidean space, we provide a mild sufficient condition under which the merged tensegrity framework is still super stable. We finally studied optimal merging of super stable tensegrity frameworks when d reduces to 2 or 3. The detailed growing procedures have been provided for better illustration.

We have created some connections between tensegrity frameworks and formation control in Chapter 5, where the problem on how to construct universally rigid tensegrity frameworks under given configurations is first raised, followed by the formation stabilization problem for multi-agent systems modeled by single integrators. The construction of universally rigid tensegrity frameworks is equivalent to the design of a positive semi-definite stress matrix with rank $n-d-1$ since the members can be consequently assigned based on the associated stress. Because of the fact that the elements of the stress matrix lie in the null space of the generalized configuration matrix at equilibrium, we have designed a numerical algorithm to seek a stress matrix that agrees with a universally rigid tensegrity framework. In view of the desirable features that the cables and struts have strict upper and lower bounds on lengths, respectively, we proposed a class of nonlinear distance-based control laws by mapping the multi-agent system to a virtual tensegrity framework. It has been shown that the formation can be stabilized using the proposed control strategies and at the same time the inter-agent distances never exceed their limitations during the evolution.

In Chapter 6 we further promote the use of tensegrity frameworks in formation control. As a representative application, we investigate the formation scaling problem under the virtual tensegrity framework with vertices and their associated members denoting the agents and links between them, respectively. We first explore the conditions under which the formations can be scaled to the desired size using the proposed stress-based control laws. It has been shown that d pairs of agents are sufficient to change the size of the whole formation if their position vectors linearly span the whole space $\mathbb{R}^d$. By introducing the orthogonal projections to the controllers of a portion of agents, we have also proved that the size of the formation can be decided by only one pair of agents. In addition, the formation is shown to be uniquely determined up to the translation and scaling of the given configuration among all the possible affine transformations. These results have been further extended to satisfy the requirement that the scaling information is known to only one agent. In this case, we have proposed a new class of estimator-based control laws, such that the whole formation can be driven to the prescribed size.

Chapter 7 has solved the problem of formation tracking to a given reference signal. In order to precisely follow the given trajectory, we have designed a

distributed finite-time estimator with the advantage that the explicit knowledge of the bound of the agents' speed is not required in contrast to the existing results. Based on rigidity graph theory, we have proposed distance-based control laws by employing the output of the estimator. It has been shown that the shape of the formation converges to the pre-defined one, and its centroid follows the given reference.

8.2 Future work

Although in this thesis we have solved a sequence of problems related to the construction of tensegrity frameworks and the applications in formation control, several problems still need to be considered in future research. In this section, we identify some future topics listed following the order of the chapters.

- Chapter 3: It has been shown that rigidity and infinitesimal rigidity can be preserved in the process of merging given static separate tensegrity frameworks. However, these results might not hold when the tensegrity framework moves due to the change of geometric relationships with respect to each other. Hence it is desirable to analyze the influence imposed by motion and design rigidity (including infinitesimal and universal rigidity) maintenance control laws.

- Chapter 4: In addition to the research on local rigidity (Chapter 3) and super-stability of tensegrity frameworks, it is also important to investigate the strategies of augmenting globally rigid tensegrity frameworks systematically. The procedures therein can give more freedoms when setting stresses for newly added members. In addition, it is of great interest to study splitting tensegrity frameworks in contrast to merging as have been discussed in this chapter.

- Chapter 5: Regarding the construction of tensegrity frameworks, the cost and complexity of the whole framework are important criteria especially in practical applications. Therefore it is appealing to investigate optimization based construction algorithms to further reduce the number of members required for creating a tensegrity framework.

- Chapter 6: The scaling parameter in this chapter is assumed to be constant. However, the size of the formation might be time-varying. This motivate us to extend our current results to the case involving a dynamic signal indicating the size of the formation. In addition, it is of great interest to make use of tensegrity frameworks in more cooperative control tasks for robots, e.g., autonomous formation enclosing and transformation.

- Chapter 7: One future study is to generalize the results to fully local coordinate systems, where the orientations might be inconsistent. Another possible exploration is to develop more powerful control algorithms to cope with system constraints, such as input saturation and transmission delay.

Appendix A

Lemma on the rank of the matrix $\hat{\Omega}$ in (4.12)

Lemma A.1. *Consider the matrix $\hat{\Omega} \in \mathbb{R}^{(n+1)\times(n+1)}$ defined in* (4.12)*, where $\Omega \in \mathbb{R}^{n\times n}$ and $\Omega_u \in \mathbb{R}^{4\times 4}$ are the stress matrices associated with super stable tensegrity frameworks with three common vertices. Then*

$$rank(\hat{\Omega}) = n - 2. \tag{A.1}$$

Proof. We first consider the solution to the following equations

$$\Omega_a x = \mathbf{0}, \tag{A.2a}$$

$$\Omega_b y = \mathbf{0}, \tag{A.2b}$$

where $x, y \in \mathbb{R}^{n+1}$. In view of (4.12), (A.2a) can be equivalently written as

$$\left(\begin{array}{c:c} \Omega & \mathbf{0}_{n\times 1} \\ \hdashline \mathbf{0}_{1\times n} & 0 \end{array}\right) \left(\begin{array}{c} x_1 \\ x_2 \end{array}\right) = \left(\begin{array}{c} \mathbf{0}_{n\times 1} \\ 0 \end{array}\right), \tag{A.3}$$

where $x_1 \in \mathbb{R}^{n\times 1}$ and $x_2 \in \mathbb{R}$. After simple calculation, (A.3) can be reduced to

$$\begin{cases} \Omega x_1 = \mathbf{0}, \\ 0 x_2 = 0. \end{cases} \tag{A.4}$$

Since $null(\Omega) = span(q^T, \mathbf{1}_n)$, the solution space of (A.4) (equivalently, (A.2a)) is as follows

$$\begin{aligned} \mathbb{S}_a =& span\left(\left(\begin{array}{c} q_{\cdot 1} \\ p_1^a \end{array}\right), \left(\begin{array}{c} q_{\cdot 2} \\ p_2^a \end{array}\right), \left(\begin{array}{c} \mathbf{1}_n \\ c_a \end{array}\right)\right) \\ \triangleq& span\left(s_1^a, s_2^a, s_3^a\right), \end{aligned} \tag{A.5}$$

where $q_{\cdot 1} = [q_{11}, \cdots, q_{n1}]^T \in \mathbb{R}^n$ with q_{i1} being the first component of q_i, $i = 1, \cdots, n$, and $q_{\cdot 2}$ is defined analogously. p_1^a, p_2^a and c_a are any arbitrary scalars.

Similarly, the solution space of (A.2b) is given by

$$\begin{aligned}\mathbb{S}_b =& span\left(\begin{pmatrix} p^b_{11} \\ \vdots \\ p^b_{(n-3)1} \\ q_{(n-2)1} \\ \vdots \\ q_{(n+1)1} \end{pmatrix}, \begin{pmatrix} p^b_{12} \\ \vdots \\ p^b_{(n-3)2} \\ q_{(n-2)2} \\ \vdots \\ q_{(n+1)2} \end{pmatrix}, \begin{pmatrix} c_{b1} \\ \vdots \\ c_{b(n-3)} \\ 1 \\ \vdots \\ 1 \end{pmatrix}\right), \\ \triangleq& span\left(s^b_1, s^b_2, s^b_3\right)\end{aligned} \tag{A.6}$$

where $p^b_{ij}, i = 1, \cdots, n-3, j = 1, 2$, denote the jth component of an arbitrary real vector $p^b_i \in \mathbb{R}^2$, and c_{bi}, $i = 1, \cdots, n-3$, are arbitrary scalars. In view of Lemma 7.2, we know

$$null(\hat{\Omega}) = \mathbb{S}_a \cap \mathbb{S}_b. \tag{A.7}$$

To determine the non-trivial form of $\mathbb{S}_a \cap \mathbb{S}_b$, let

$$\alpha_1 s^a_1 + \alpha_2 s^a_2 + \alpha_3 s^a_3 = \beta_1 s^b_1 + \beta_2 s^b_2 + \beta_3 s^b_3, \tag{A.8}$$

where α_i and β_i, $i = 1, 2, 3$, are scalars, at least one of which is nonzero. Note that $\mathbb{S}_a$ and $\mathbb{S}_b$ share the same entries as follows

$$s_c = \left(\begin{pmatrix} q_{(n-2)1} \\ q_{(n-1)1} \\ q_{n1} \end{pmatrix}, \begin{pmatrix} q_{(n-2)2} \\ q_{(n-1)2} \\ q_{n2} \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}\right). \tag{A.9}$$

Combining (A.8) and (A.9), one has

$$(\alpha_1 - \beta_1)\begin{pmatrix} q_{(n-2)1} \\ q_{(n-1)1} \\ q_{n1} \end{pmatrix} + (\alpha_2 - \beta_2)\begin{pmatrix} q_{(n-2)2} \\ q_{(n-1)2} \\ q_{n2} \end{pmatrix} + (\alpha_3 - \beta_3)\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \mathbf{0}, \tag{A.10}$$

which can be equivalently written as

$$\begin{pmatrix} q_{(n-2)1} & q_{(n-2)2} & 1 \\ q_{(n-1)1} & q_{(n-2)1} & 1 \\ q_{n1} & q_{n2} & 1 \end{pmatrix}\begin{pmatrix} \alpha_1 - \beta_1 \\ \alpha_2 - \beta_2 \\ \alpha_3 - \beta_3 \end{pmatrix} = \mathbf{0}. \tag{A.11}$$

Recalling that vertices i, j and k are not collinear, it is equivalent to say that they

are in general positions in the plane, which implies

$$rank\begin{pmatrix} q_{(n-2)1} & q_{(n-2)2} & 1 \\ q_{(n-1)1} & q_{(n-2)1} & 1 \\ q_{n1} & q_{n2} & 1 \end{pmatrix} = 3. \tag{A.12}$$

Then in view of (A.11), the parameters α_i and β_i, $i = 1, 2, 3$, in (A.8) satisfy

$$\begin{cases} \alpha_1 = \beta_1, \\ \alpha_2 = \beta_2, \\ \alpha_3 = \beta_3. \end{cases} \tag{A.13}$$

From the fact that $\hat{\Omega}$ is a stress matrix associated with configuration $\bar{q}$, we know

$$(\bar{q}_{\cdot 1}, \bar{q}_{\cdot 2}, \mathbf{1}_{n+1}) \subseteq null(\hat{\Omega}), \tag{A.14}$$

where $\bar{q}_{\cdot 1} = [q_{\cdot 1}^T, q_{(n+1)1}]^T$, and $\bar{q}_{\cdot 2}$ is defined analogously. Since $rank\,(\bar{q}_{\cdot 1}, \bar{q}_{\cdot 2}, \mathbf{1}_{n+1}) = 3$, we have

$$rank(\hat{\Omega}) \leqslant n - 2. \tag{A.15}$$

Then, to prove $rank(\hat{\Omega}) = n-2$, we need to show that any other vector $v \in null(\hat{\Omega})$ can be represented as a linear combination of vectors $\bar{q}_{\cdot 1}, \bar{q}_{\cdot 2}$, and $\mathbf{1}_{n+1}$, namely, there exist scalars γ_1, γ_2, and γ_3, such that

$$v = \gamma_1 \bar{q}_{\cdot 1} + \gamma_2 \bar{q}_{\cdot 2} + \gamma_3 \mathbf{1}_{n+1}, \quad \forall v \in null(\hat{\Omega}), \tag{A.16}$$

where at least one of $\gamma_i, i = 1, 2, 3$, is nonzero. In light of Lemma 7.2, one has

$$v \in null(\hat{\Omega}) \Longleftrightarrow v \in \mathbb{S}_a \text{ and } v \in \mathbb{S}_b, \tag{A.17}$$

which implies

$$\begin{aligned} v =& \alpha_1 s_1^a + \alpha_2 s_2^a + \alpha_3 s_3^a \\ =& \beta_1 s_1^b + \beta_2 s_2^b + \beta_3 s_3^b. \end{aligned} \tag{A.18}$$

It follows from (A.13) that

$$\begin{pmatrix} v \\ v \end{pmatrix} = \alpha_1 \begin{pmatrix} s_1^a \\ s_1^b \end{pmatrix} + \alpha_2 \begin{pmatrix} s_2^a \\ s_2^b \end{pmatrix} + \alpha_3 \begin{pmatrix} s_3^a \\ s_3^b \end{pmatrix}. \tag{A.19}$$

Picking out respectively the first n entries of s_i^a and the last entry of s_i^b, $i = 1, 2, 3$, we get

$$v = \alpha_1 \begin{pmatrix} q_{\cdot 1} \\ q_{(n+1)1} \end{pmatrix} + \alpha_2 \begin{pmatrix} q_{\cdot 2} \\ q_{(n+1)2} \end{pmatrix} + \alpha_3 \begin{pmatrix} \mathbf{1}_n \\ 1 \end{pmatrix}, \tag{A.20}$$

equivalently,

$$v = \alpha_1 \bar{q}_{\cdot 1} + \alpha_2 \bar{q}_{\cdot 2} + \alpha_3 \mathbf{1}_{n+1}. \tag{A.21}$$

Therefore, there exist scalars $\gamma_i, i = 1, 2, 3$, such that any vector $v \in null(\hat{\Omega})$ can be written as a linear combination of $\bar{q}_{\cdot 1}, \bar{q}_{\cdot 2}$, and $\mathbf{1}_{n+1}$. This completes the proof. □

Appendix B

Technical Proofs for Chapter 7

B.1 Proof of (7.22)

Suppose $x \in \mathbb{R}^d$ and $\|x\| \neq 0$, when ϵ is chosen such that $0 < \epsilon \leqslant \frac{2}{3}$, we have

$$(1-\epsilon)^2\|x\|^2 + \epsilon(2-3\epsilon)\|x\| \geqslant 0.$$

Equivalently,

$$(1-\epsilon)^2\|x\|^2 + 2\epsilon\|x\| - 2\epsilon^2\|x\| \geqslant \epsilon^2\|x\|. \tag{B.1}$$

Then, adding ϵ^2 to both sides of (B.1), we obtain that

$$(1-\epsilon)^2\|x\|^2 + 2\epsilon(1-\epsilon)\|x\| + \epsilon^2 \geqslant \epsilon^2\|x\| + \epsilon^2,$$

which is can be written as

$$[(1-\epsilon)\|x\| + \epsilon]^2 \geqslant \epsilon^2(1+\|x\|). \tag{B.2}$$

By taking a square root of (B.2), it follows

$$(1-\epsilon)\|x\| + \epsilon \geqslant \epsilon\sqrt{1+\|x\|}.$$

After simple calculation, we get

$$\|x\| \geqslant \epsilon\|x\| + \epsilon\left(\sqrt{1+|x|} - 1\right). \tag{B.3}$$

Multiplying both sides of (B.3) by $|x|$, we have

$$|x|^2 \geqslant \epsilon|x|^2 + \epsilon\|x\|\left(\sqrt{1+\|x\|} - 1\right).$$

Since $\|x\| + \|x\|\left(\sqrt{1+\|x\|} - 1\right) > 0$, it is straightforward to know

$$\frac{\|x\|^2}{\|x\| + \left(\sqrt{1+\|x\|} - 1\right)} \geqslant \epsilon\|x\|. \tag{B.4}$$

When $\|x\| \to 0$, we have

$$\lim_{\|x\|\to 0} \frac{\|x\|^2}{\epsilon\|x\|\left(\|x\| + \left(\sqrt{1+\|x\|}-1\right)\right)} = \lim_{\|x\|\to 0} \frac{\|x\|^2}{\epsilon\|x\|^2 + \frac{1}{2}\epsilon\|x\|^2} = \frac{2}{3\epsilon},$$

where we have used the equivalent infinitesimal $\left(\sqrt{1+\|x\|}-1\right) \sim \frac{1}{2}\|x\|$. In view of the condition that $0 < \epsilon \leqslant \frac{2}{3}$, we further obtain that

$$\lim_{\|x\|\to 0} \frac{\|x\|^2}{\epsilon\|x\|\left(\|x\| + \left(\sqrt{1+\|x\|}-1\right)\right)} \geqslant 1.$$

In addition, it holds that

$$\lim_{\|x\|\to 0} \frac{\|x\|^2}{\|x\| + \left(\sqrt{1+\|x\|}-1\right)} = \lim_{\|x\|\to 0} \frac{\|x\|^2}{\|x\| + \frac{1}{2}\|x\|} = 0.$$

Hence, $\forall x \in \mathbb{R}^n, 0 < \epsilon \leqslant \frac{2}{3}$, we have

$$\frac{\|x\|^2}{\|x\| + \left(\sqrt{1+\|x\|}-1\right)} \geqslant \epsilon\|x\|.$$

B.2 Proof of the boundedness of $\tilde{q}_d^g$ in $(0, T_0]$

Now we consider the system dynamics during $t \in (0, T_0]$. Then (7.29) can be equivalently written as

$$\begin{aligned} u^g =& \mathbf{1}_n \otimes \dot{q}_d^g - k_p \left(B\hat{Q}_\delta \otimes I_d\right)(\hat{q}_c^g - \mathbf{1}_n \otimes q_c^g + \mathbf{1}_n \otimes q_c^g - \mathbf{1}_n \otimes q_d^g) - k_s \nabla P(q^g) \\ =& \mathbf{1}_n \otimes \dot{q}_d^g - k_p \left(B\hat{Q}_\delta \otimes I_d\right)(\mathbf{1}_n \otimes q_c^g - \mathbf{1}_n \otimes q_d^g) - k_s \nabla P(q^g) \\ & - k_p \left(B\hat{Q}_\delta \otimes I_d\right)(\hat{q}_c^g - \mathbf{1}_n \otimes q_c^g) \end{aligned} \tag{B.5}$$

Note that the first line after the second equality sign in (B.5) is exactly (7.30). In addition, we know

$$\begin{aligned} -\frac{k_p}{n}(\tilde{q}_d^g)^T(\mathbf{1}_n^T B\hat{Q}_\delta \otimes I_d)\tilde{q}_c^g \leqslant& \frac{k_p}{2n}\sum_{i=1}^n \frac{b_i}{\delta + \|\hat{q}_{ci}^g - q_d^g\|}(\|\tilde{q}_d^g\|^2 + \|\tilde{q}_{ci}^g\|^2) \\ \leqslant& \frac{k_p}{2n}\sum_{i=1}^n \frac{b_i}{\delta + \|\hat{q}_{ci}^g - q_d^g\|}\|\tilde{q}_d^g\|^2 + \frac{k_p}{2n\delta}\|\tilde{q}_c^g\|^2 \end{aligned}$$

Then in view of (7.35), we have

$$\begin{aligned}\dot{V} \leqslant & -\frac{k_p}{2n}\sum_{i}^{n}\frac{b_i}{\delta+\|\hat{q}_{ci}^g-q_d^g\|}\|\tilde{q}_d^g\|^2 - k_s\|\nabla P(q^g)\|^2 + \frac{k_p\sqrt{n}}{\delta+\|\hat{q}_{ci}^g-q_d^g\|}\|\tilde{q}_d^g\|\|\nabla P(q^g)\| \\ & + \frac{k_p}{2n}\sum_{i=1}^{n}\frac{b_i}{\delta+\|\hat{q}_{ci}^g-q_d^g\|}\|\tilde{q}_{ci}^g\|^2 \\ \leqslant & -\begin{bmatrix} \|\tilde{q}_d^g\| \\ \|\nabla P(q^g)\| \end{bmatrix}^T Q' \begin{bmatrix} \|\tilde{q}_d^g\| \\ \|\nabla P(q^g)\| \end{bmatrix} + \frac{k_p}{2n\delta}\|\tilde{q}_c^g\|^2,\end{aligned} \tag{B.6}$$

where

$$Q' = \begin{bmatrix} \frac{k_p}{2\left(\delta+\sup_{t\in(0,T])}\|\hat{q}_{ci}^g-q_d^g\|\right)} & -\frac{k_p\sqrt{n}}{2\left(\delta+\sup_{t\in(0,T])}\|\hat{q}_{ci}^g-q_d^g\|\right)} \\ -\frac{k_p\sqrt{n}}{2\left(\delta+\sup_{t\in(0,T])}\|\hat{q}_{ci}^g-q_d^g\|\right)} & k_s \end{bmatrix}.$$

Then, Q' is positive definite if k_s is chosen such that

$$k_s > \frac{k_p n}{2\left(\delta+\sup_{t\in(0,T])}\|\hat{q}_{ci}^g-q_d^g\|\right)},$$

which automatically holds under the condition (7.15). It follows from (B.6) that

$$V(T_0) = V(0) - \int_0^{T_0}\begin{bmatrix} \|\tilde{q}_d^g\| \\ \|\nabla P(q^g)\| \end{bmatrix}^T Q' \begin{bmatrix} \|\tilde{q}_d^g\| \\ \|\nabla P(q^g)\| \end{bmatrix} dt + \int_0^{T_0}\frac{k_p}{2n\delta}\|\tilde{q}_c^g\|^2 dt$$

Recalling the convergence of $\|\tilde{q}_c^g\|$ from (7.28), we know $\int_0^{T_0}\frac{k_p}{2n\delta}\|\tilde{q}_c^g\|^2 dt$ is bounded for finite number T_0. It thus follows from the formula of V in (7.32) that $V(T_0)$ is bounded. Hence, during $t \in (0, T_0]$, $\tilde{q}_d^g$ and $P(q^g)$ are both bounded.

In addition, we can also infer $\nabla P(q^g)$ is bounded from the boundedness of $P(q^g)$, and thus the control input u_i^g in (7.20) is bounded. Hence, the position variable q_i^g becomes bounded in finite time. It can also be obtained from (4.4) that $\hat{q}_{ci}^i$ is bounded in finite time.

Bibliography

[1] A. Y. Alfakih. On the universal rigidity of generic bar frameworks. *Contributions to Discrete Mathematics*, 5(1), 2010.

[2] A. Y. Alfakih and V. H. Nguyen. On affine motions and universal rigidity of tensegrity frameworks. *Linear Algebra and its Applications*, 439(10): 3134–3147, 2013.

[3] A. Y. Alfakih and Y. Ye. On affine motions and bar frameworks in general position. *Linear Algebra and its Applications*, 438(1):31–36, 2013.

[4] A. Y. Alfakih, N. Taheri, and Y. Ye. On stress matrices of (d+1)-lateration frameworks in general position. *Mathematical Programming*, 137(1-2):1–17, 2013.

[5] B. D. O. Anderson, C. Yu, B. Fidan, and J. M. Hendrickx. Rigid graph control architectures for autonomous formations. *IEEE Control Systems*, 28 (6):48–63, 2008.

[6] B. D. O. Anderson, C. Yu, and J. M. Hendrickx. Rigid graph control architectures for autonomous formations. *IEEE Control Systems*, 28(6), 2008.

[7] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. A decentralized controller-observer scheme for multi-agent weighted centroid tracking. *IEEE Transactions on Automatic Control*, 58(5):1310–1316, 2013.

[8] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. Decentralized time-varying formation control for multi-robot systems. *The International Journal of Robotics Research*, pages 1029–1043, 2014.

[9] L. Asimow and B. Roth. The rigidity of graphs, II. *Journal of Mathematical Analysis and Applications*, 68(1):171–190, 1979.

[10] T. Balch and R. C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, 1998.

[11] F. Bullo, J. Cortés, and S. Martinez. *Distributed control of robotic networks: a mathematical approach to motion coordination algorithms*. Princeton University Press, 2009.

[12] M. Cao, A. S. Morse, and B. D. O. Anderson. Reaching a consensus in a dynamically changing environment: a graphical approach. *SIAM Journal on Control and Optimization*, 47(2):575–600, 2008.

[13] M. Cao, A. S. Morse, C. Yu, B. D. O. Anderson, and S. Dasgupta. Maintaining a directed, triangular formation of mobile autonomous agents. *Communications in Information and Systems*, 11(1):1, 2011.

[14] M. Cao, A. S. Morse, C. Yu, B. D. O. Anderson, and S. Dasgupta. Maintaining a directed, triangular formation of mobile autonomous agents. *Communications in Information and Systems*, 11(1):1, 2011.

[15] M. Cao, C. Yu, and B. Anderson. Formation control using range-only measurements. *Automatica*, 47(4):776–781, 2011.

[16] Y. Cao, W. Ren, and Z. Meng. Decentralized finite-time sliding mode estimators and their applications in decentralized finite-time formation tracking. *Systems and Control Letters*, 59(9):522–529, 2010.

[17] Y. Cao, W. Yu, W. Ren, and G. Chen. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial Informatics*, 9(1):427–438, 2013.

[18] D. Carboni, R. K. Williams, A. Gasparri, G. Ulivi, and G. S. Sukhatme. Rigidity-preserving team partitions in multiagent networks. *IEEE transactions on cybernetics*, 45(12):2640–2653, 2015.

[19] R. Connelly. Rigidity and energy. *Inventiones Mathematicae*, 66(1):11–33, 1982.

[20] R. Connelly. Tensegrity structures: why are they stable? In *Rigidity theory and applications*, pages 47–54. Plenum Press, New York, 1999.

[21] R. Connelly. Tensegrity structures: why are they stable? In *Rigidity theory and applications*, pages 47–54. Springer, 2002.

[22] R. Connelly. Generic global rigidity. *Discrete and Computatonal Geometry*, 33:549–563, 2005.

[23] R. Connelly. Tensegrities and global rigidity. In *Shaping Space*, pages 267–278. Springer, 2013.

[24] R. Connelly and S. J. Gortler. Iterative universal rigidity. *Discrete & Computational Geometry*, 53(4):847–877, 2015.

[25] R. Connelly and S. D. Guest. *Frameworks, Tensegrities and Symmetry: Understanding Stable Structures*. available online at http://www.math.cornell.edu/ web7510/framework.pdf, 2015.

[26] R. Connelly and W. Whiteley. Second-order rigidity and prestress stability for tensegrity frameworks. *SIAM Journal on Discrete Mathematics*, 9(3): 453–491, 1996.

[27] S. Coogan and M. Arcak. Scaling the size of a formation using relative position feedback. *Automatica*, 48(10):2677–2685, 2012.

[28] J. Cortés. Global and robust formation-shape stabilization of relative sensing networks. *Automatica*, 45(12):2754–2762, 2009.

[29] D. V. Dimarogonas and K. H. Johansson. On the stability of distance-based formation control. In *the 47th IEEE Conference on Decision and Control*, pages 1200–1205. IEEE, 2008.

[30] D. V. Dimarogonas and K. H. Johansson. Further results on the stability of distance-based multi-robot formations. In *American Control Conference*, pages 2972–2977. IEEE, 2009.

[31] D. V. Dimarogonas and K. H. Johansson. Stability analysis for multi-agent systems using the incidence matrix: quantized communication and formation control. *Automatica*, 46(4):695–700, 2010.

[32] X. Dong and G. Hu. Time-varying formation control for general linear multi-agent systems with switching directed topologies. *Automatica*, 73:47–55, 2016.

[33] X. Dong and G. Hu. Time-varying formation tracking for linear multi-agent systems with multiple leaders. *IEEE Transactions on Automatic Control*, 62 (7):3658–3664, 2017.

[34] X. Dong, Y. Zhou, Z. Ren, and Y. Zhong. Time-varying formation tracking for second-order multi-agent systems subjected to switching topologies with application to quadrotor formation flying. *IEEE Transactions on Industrial Electronics*, 64(6):5014–5024, 2017.

[35] F. Dorfler and B. Francis. Geometric analysis of the formation problem for autonomous robots. *IEEE Transactions on Automatic Control*, 55(10): 2379–2384, 2010.

[36] F. Dorfler and B. A. Francis. Formation control of autonomous robots based on cooperative behavior. In *Control Conference (ECC), 2009 European*, pages 2432–2437, 2009.

[37] O. Ekeberg and K. Pearson. Computer simulation of stepping in the hind legs of the cat: an examination of mechanisms regulating the stance-to-swing transition. *Journal of Neurophysiology*, 94(6):4256–4268, 2005.

[38] D. G. Emmerich. Constructions de reseaux autotendantes, 1963. patent no. 1.377. 290.

[39] T. Eren, P. N. Belhumeur, B. D. O. Anderson, and A . S. Morse. A framework for maintaining formations based on rigidity. *IFAC Proceedings Volumes*, 35 (1):499–504, 2002.

[40] T. Eren, B. D. O. Anderson, A. S. Morse, W. Whiteley, and P. N. Belhumeur. Operations on rigid formations of autonomous agents. *Communications in Information & Systems*, 3(4):223–258, 2003.

[41] T. Eren, O. K. Goldenberg, W. Whiteley, Y. R. Yang, A. S. Morse, B. D. O. Anderson, and P. N. Belhumeur. Rigidity, computation, and randomization in network localization. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2673–2684. IEEE, 2004.

[42] J. Fink, N. Michael, S. Kim, and V. Kumar. Planning and control for cooperative manipulation and transportation with aerial robots. *The International Journal of Robotics Research*, 30(3):324–334, 2011.

[43] W. Fleming. *Functions of several variables*. Springer Science & Business Media, 2012.

[44] T. I. Fossen. *Guidance and control of ocean vehicles*. John Wiley & Sons Inc, 1994.

[45] M. Franceschelli and A. Gasparri. Decentralized centroid estimation for multi-agent systems in absence of any common reference frame. In *American Control Conference, 2009. ACC'09.*, pages 512–517. IEEE.

[46] R. B. Fuller. Tensile integrity structures, 1959. US Patent 3,063,521.

[47] H. Garcia de Marina, M. Cao, and B. Jayawardhana. Controlling rigid formations of mobile agents under inconsistent measurements. *IEEE Transactions on Robotics*, 31(1):31–39, 2015.

[48] John R. Gilbert and Michael T. Heath. Computing a sparse basis for the null space. *SIAM Journal on Algebraic Discrete Methods*, 8(3):446–459, 1987.

[49] S. J. Gortler and D. P. Thurston. Characterizing the universal rigidity of generic frameworks. *Discrete & Computational Geometry*, 51(4):1017–1036, 2014.

[50] S. J. Gortler, A. D. Healy, and D. P. Thurston. Characterizing generic global rigidity. *American Journal of Mathematics*, 132(4):897–939, 2010.

[51] L. Gottlieb and T. Neylon. Matrix sparsification and the sparse null space problem. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 205–218. Springer, 2010.

[52] G. Habibi, Z. Kingston, W. Xie, M. Jellins, and J. McLurkin. Distributed centroid estimation and motion controllers for collective transport by multi-robot systems. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1282–1288. IEEE.

[53] T. Han, Z. Lin, and M. Fu. Three-dimensional formation merging control under directed and switching topologies. *Automatica*, 58:99–105, 2015.

[54] Z. Han, L. Wang, Z. Lin, and R. Zheng. Formation control with size scaling via a complex laplacian-based approach. *IEEE transactions on cybernetics*, 46 (10):2348–2359, 2016.

[55] B. Hendrickson. Conditions for unique graph realizations. *SIAM journal on computing*, 21(1):65–84, 1992.

[56] J. M. Hendrickx, B. Anderson, J. C. Delvenne, and C. D. Blondel. Directed graphs for the analysis of rigidity and persistence in autonomous agent systems. *International journal of robust and nonlinear control*, 17(10-11): 960–981, 2007.

[57] J. M. Hendrickx, C. Yu, B. Fidan, and B. D. O.Anderson. Rigidity and persistence for ensuring shape maintenance of multi-agent meta-formations. *Asian Journal of Control*, 10(2):131–143, 2008.

[58] L. Henneberg. *Die graphische Statik der starren Systeme*. Leipzig: Teubner, 1911.

[59] G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin. Quadrotor helicopter trajectory tracking control. In *AIAA guidance, navigation and control conference and exhibit*, pages 1–14, 2008.

[60] D. E. Ingber. The architecture of life. *Scientific American*, 278(1):48–57, 1998.

[61] D. J. Jacobs, A. J. Rader, L. A. Kuhn, and M. F. Thorpe. Protein flexibility predictions using graph theory. *Proteins: Structure, Function, and Bioinformatics*, 44(2):150–165, 2001.

[62] M. Ji and M. Egerstedt. Distributed coordination control of multi-agent systems while preserving connectedness. *IEEE Transactions on Robotics*, 23 (4):693–703, 2007.

[63] T. Jordán and V. H. Nguyen. On universally rigid frameworks on the line. *Contributions to Discrete Mathematics*, 10(2), 2016.

[64] S. H. Juan and J. M. M. Tur. Tensegrity frameworks: static analysis review. *Mechanism and Machine Theory*, 43(7):859–881, 2008.

[65] Z. Kan, L. Navaravong, J. M. Shea, E. L. Pasiliao, and W. E. Dixon. Graph matching-based formation reconfiguration of networked agents with connectivity maintenance. *IEEE Transactions on Control of Network Systems*, 2 (1):24–35, 2015.

[66] S. D. Kelly and A. Micheletti. A class of minimal generically universally rigid frameworks. *arXiv preprint arXiv:1412.3436*, 2014.

[67] H. Kenner. *Geodesic math and how to use it*. Univ of California Press, 1976.

[68] H. K. Khalil. *Nonlinear Systems (3rd editon)*. Upper Saddle River, NJ: Prentice Hall, 2002.

[69] K. Kim, A. K. Agogino, A. Toghyan, D. Moon, L. Taneja, and A. M. Agogino. Robust learning of tensegrity robot control for locomotion through form-finding. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5824–5831. IEEE, 2015.

[70] L. Krick, M. E. Broucke, and B. A. Francis. Stabilisation of infinitesimally rigid formations of multi-robot networks. *International Journal of Control*, 82(3):423–439, 2009.

[71] S. Lau and W. Naeem. Tensegrity-based formation control of unmanned vehicles. In *Control (CONTROL), 2012 UKACC International Conference on*, pages 1–6. IEEE, 2012.

[72] S. Lau and W. Naeem. Co-operative tensegrity-based formation control algorithm for a multi-aircraft system. In *2015 American Control Conference (ACC)*, pages 750–756. IEEE, 2015.

[73] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis. Collective motion, sensor networks, and ocean sampling. *Proceedings of the IEEE*, 95(1):48–74, 2007.

[74] S. M. Levin. The tensegrity-truss as a model for spine mechanics: biotensegrity. *Journal of mechanics in medicine and biology*, 2(03n04):375–388, 2002.

[75] W. Li. Notion of control-law module and modular framework of cooperative transportation using multiple nonholonomic robotic agents with physical rigid-formation-motion constraints. *IEEE transactions on cybernetics*, 46(5): 1242–1248, 2016.

[76] T. Liedl, B. Högberg, J. Tytell, D. E. Ingber, and W. M. Shih. Self-assembly of three-dimensional prestressed tensegrity structures from dna. *Nature nanotechnology*, 5(7):520–524, 2010.

[77] Z. Lin, L. Wang, Z. Han, and M. Fu. Distributed formation control of multi-agent systems using complex laplacian. *IEEE Transactions on Automatic Control*, 59(7):1765–1777, 2014.

[78] Z. Lin, L. Wang, Z. Chen, M. Fu, and Z. Han. Necessary and sufficient graphical conditions for affine formation control. *IEEE Transactions on Automatic Control*, 61(10):2877–2891, 2016.

[79] T. Liu and Z. Jiang. Distributed formation control of nonholonomic mobile robots without global position measurements. *Automatica*, 49(2):592–600, 2013.

[80] R. W. Marks and R. B. Fuller. *The Dymaxion World of Buckminster Fuller*. 1960.

[81] D. Mellinger, N. Michael, and V. Kumar. Trajectory generation and control for precise aggressive maneuvers with quadrotors. *The International Journal of Robotics Research*, page 0278364911434236, 2012.

[82] Z. Meng, B. D. O. Anderson, and S. Hirche. Formation control with mismatched compasses. *Automatica*, 69:232–241, 2016.

[83] M. Mesbahi and M.Egerstedt. *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[84] B. Mohar and Y. Alavi. The laplacian spectrum of graphs. *Graph Theory, Combinatorics and Applications*, 2:871–898, 1991.

[85] R. Motro. *Tensegrity: structural systems for the future*. Elsevier, 2003.

[86] S. Mou, M. Cao, and A. S. Morse. Target-point formation control. *Automatica*, 61:113–118, 2015.

[87] S. Mou, M. A. Belabbas, A. S. Morse, Z. Sun, and B. D. O. Anderson. Undirected rigid formations are problematic. *IEEE Transactions on Automatic Control*, 61(10):2821–2836, 2016.

[88] B. Nabet and N. E. Leonard. Tensegrity models and shape control of vehicle formations. *arXiv: 0902.3710*, 2009.

[89] K. K. Oh and H. S. Ahn. Distance-based undirected formations of single-integrator and double-integrator modeled agents in n-dimensional space. *International Journal of Robust and Nonlinear Control*, 24(12):1809–1820, 2014.

[90] K. K. Oh and H. S. Ahn. Formation control and network localization via orientation alignment. *IEEE Transactions on Automatic Control*, 59(2):540–545, 2014.

[91] K. K. Oh, M. C. Park, and H. S. Ahn. A survey of multi-agent formation control. *Automatica*, 53:424–440, 2015.

[92] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and timedelays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[93] D. Pais, M. Cao, and N. E. Leonard. Formation shape and orientation control using projected collinear tensegrity structures. In *Proceedings of the 2009 American Control Conference*, pages 610–615, St. Louis, MO, June 2009.

[94] C. Paul, F. J. Valero-Cuevas, and H. Lipson. Design and control of tensegrity robots for locomotion. *IEEE Transactions on Robotics*, 22(5):944–957, 2006.

[95] M. Porfiri, D. G. Roberson, and D. J. Stilwell. Tracking and formation control of multiple autonomous agents: A two-level consensus approach. *Automatica*, 43(8):1318–1328, 2007.

[96] A. Pothen. *Sparse null bases and marriage theorems*. PhD thesis, Cornell University, 1984.

[97] A. Pugh. *An introduction to tensegrity*. Univ of California Press, 1976.

[98] S. Ramazani, R. Selmic, and M. de Queiroz. Rigidity-based multiagent layered formation control. *IEEE transactions on cybernetics*, 47(8):1902–1913, 2017.

[99] K. Ratmanski. Universally rigid framework attachments. *arXiv preprint arXiv:1011.4094*, 2010.

[100] W. Ren. Multi-vehicle consensus with a time-varying reference state. *Systems & Control Letters*, 56(7):474–483, 2007.

[101] W. Ren and R. Beard. *Distributed consensus in multi-vehicle cooperative control: theory and applications*. Springer-Verlag, 2008.

[102] W. Ren and Y. Cao. *Distributed coordination of multi-agent networks: emergent problems, models, and issues*. Springer Science & Business Media, 2010.

[103] W. Ren and N. Sorensen. Distributed coordination architecture for multi-robot formation control. *Robotics and Autonomous Systems*, 56(4):324–333, 2008.

[104] B. Roth and W. Whiteley. Tensegrity frameworks. *Transactions of the American Mathematical Society*, 265(2):419–446, 1981.

[105] A. P. Sabelhaus, J. Bruce, K. Caluwaerts, P. Manovi, R. F. Firoozi, S. Dobi, A. M. Agogino, and V. SunSpiral. System design and locomotion of superball, an untethered tensegrity robot. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 2867–2873. IEEE, 2015.

[106] S. Sadao. Fuller on tensegrity. *International Journal of Space Structures*, 11 (1-2):37–42, 1996.

[107] L. Sarkisov, R. L. Martin, M. Haranczyk, and B. Smit. On the flexibility of metal–organic frameworks. *Journal of the American Chemical Society*, 136 (6):2228–2231, 2014.

[108] J. B. Saxe. *Embeddability of weighted graphs in k-space is strongly NP-hard*. Carnegie-Mellon University, Department of Computer Science, 1980.

[109] R. Sepulchre, D. Paley, and N. E. Leonard. Stabilization of planar collective motion with limited communication. *IEEE Transactions on Automatic Control*, 53(3):706–719, 2008.

[110] I. Shames, B. Fidan, B. D. O. Anderson, and H. Hmam. Cooperative self-localization of mobile agents. *IEEE Transactions on Aerospace and Electronic Systems*, 47(3):1926–1947, 2011.

[111] A. H. Simmons, C. A. Michal, and L. W. Jelinski. Molecular orientation and two-component nature of the crystalline fraction of spider dragline silk. *Science*, 271(5245):84–87, 1996.

[112] R. E. Skelton and M. C. de Oliveira. *Tensegrity systems*, volume 1. Springer, 2009.

[113] R. E. Skelton, R. Adhikari, J. P. Pinaud, W. Chan, and J. W. Helton. An introduction to the mechanics of tensegrity structures. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 5, pages 4254–4259. IEEE, 2001.

[114] K. D. Snelson. Continuous tension, discontinuous compression structures, 1965. US Patent 3,169,611.

[115] T. H. Summers, C. Yu, S. Dasgupta, and B. D. O. Anderson. Control of minimally persistent leader-remote-follower and coleader formations in the plane. *IEEE Transactions on Automatic Control*, 56(12):2778–2792, 2011.

[116] Z. Sun, S. Mou, B. D. O. Anderson, and A. S. Morse. Formation movements in minimally rigid formation control with mismatched mutual distances. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 6161–6166. IEEE, 2014.

[117] Z. Sun, S. Mou, M. Deghat, B. D. O. Anderson, and A. S. Morse. Finite time distance-based rigid formation stabilization and flocking. In *Proc. of the 19th IFAC Congress*, pages 9183–9189, 2014.

[118] Z. Sun, M. C. Park, B. D. O. Anderson, and H. S. Ahn. Distributed stabilization control of rigid formations with prescribed orientation. *Automatica*, 78(4): 250–257, 2017.

[119] Z. Szabadka. *Globally rigid frameworks and rigid tensegrity graphs in the plane*. PhD thesis, Department of Operations Research, Eötvös Loránd University, 2010.

[120] T. Tay and W. Whiteley. Generating isostatic frameworks. *Structural Topology*, 11:21–69, 1985.

[121] Y. Termonia. Molecular modeling of spider silk elasticity. *Macromolecules*, 27(25):7378–7381, 1994.

[122] A. G. Tibert and S. Pellegrino. Review of form-finding methods for tensegrity structures. *International Journal of Space Structures*, 26(3):241–255, 2011.

[123] J. M. M. Tur and S. H. Juan. Tensegrity frameworks: dynamic analysis review and open problems. *Mechanism and Machine Theory*, 44(1):1–18, 2009.

[124] L. Wang and F. Xiao. Finite-time consensus problems for networks of dynamic agents. *IEEE Transactions on Automatic Control*, 55(4):950–955, 2010.

[125] P. Wang and B. Ding. Distributed RHC for tracking and formation of nonholonomic multi-vehicle systems. *IEEE Transactions on Automatic Control*, 59(6):1439–1453, 2014.

[126] W. Wang, J. Huang, C. Wen, and H. Fan. Distributed adaptive control for consensus tracking with application to formation control of nonholonomic mobile robots. *Automatica*, 50(4):1254–1263, 2014.

[127] G. Wen, Z. Duan, W. Ren, and G. Chen. Distributed consensus of multi-agent systems with general linear node dynamics and intermittent communications. *International Journal of Robust and Nonlinear Control*, 24(16):2438–2457, 2014.

[128] W. Whiteley. Rigidity of molecular structures: generic and geometric analysis. In *Rigidity Theory and Applications*, pages 21–46. Springer, 2002.

[129] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

[130] Q. Yang, M. Cao, H. Fang, and J. Chen. Weighted centroid tracking control for multi-agent systems. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 939–944. IEEE, 2016.

[131] Q. Yang, Z. Sun, M. Cao, H. Fang, and J. Chen. Construction of universally rigid frameworks and their applications in formation scaling control. In *the 36th Chinese Control Conference*, pages 163–168, 2017.

[132] S. J. Yoo and T. H. Kim. Distributed formation tracking of networked mobile robots under unknown slippage effects. *Automatica*, 54:100–106, 2015.

[133] C. Yu, B. Fidan, and B. D. O. Anderson. Principles to control autonomous formation merging. In *American Control Conference, 2006*, pages 7–pp. IEEE, 2006.

[134] C. Yu, B. Fidan, J. M. Hendrickx, and B. D. O. Anderson. Merging multiple formations: A meta-formation prospective. In *Decision and Control, 2006 45th IEEE Conference on*, pages 4657–4663. IEEE, 2006.

[135] W. Yu, G. Chen, and M. Cao. Distributed leader-follower flocking control for multi-agent dynamical systems with time-varying velocities. *Systems & Control Letters*, 59(9):543–552, 2010.

[136] D. Zelazo and F. Allgöwer. Growing optimally rigid formations. In *American Control Conference (ACC), 2012*, pages 3901–3906. IEEE, 2012.

[137] F. Zhang and N. E. Leonard. Coordinated patterns of unit speed particles on a closed curve. *Systems & Control Letters*, 56(6):397–407, 2007.

[138] S. Zhao and D. Zelazo. Translational and scaling formation maneuver control via a bearing-based approach. *IEEE Transactions on Control of Network Systems*, 4(3):429–438, 2017.

[139] Z. Zhu, A. M. C. So, and Y. Ye. Universal rigidity: Towards accurate and efficient localization of wireless networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.

Summary

Tensegrity frameworks have drawn substantial attention from a range of disciplines, including civil engineering, biology, and mechanics, due to their identified features, such as strong stability, flexible scalability, and robustness. Motivated by these advantages, we study how to grow tensegrity framework from the graphic and algebraic point of view such that the superior features can be inherited. Another central topic of this thesis is the distributed controller design for coordination of multi-agent systems using virtual tensegrity frameworks.

We first investigate the problem of merging two separate rigid and infinitesimally rigid tensegrity frameworks in the plane, respectively. For infinitesimally rigid tenserity frameworks, the existence of the proper self-stress of the linked framework has been proven, which implies that the infinitesimal rigidity can be preserved. In addition, the type of the linked members can also be indicated by checking the sign of the corresponding stress. When merging rigid frameworks, we have proposed a disturbance perturbation-based method to justify the rigidity of the combined framework by properly inserting new members, whose type can be determined using rigidity matrix. Moreover, the Henneberg construction has been extended to grow super stable tensegrity frameworks. It has been proven that the super stability can also been preserved under the operation of vertex addition, edge splitting, and merging.

Inspired by the " turning back" method to generate sparse matrix, we propose a numerical algorithm to construct universally rigid tensegrity frameworks given a generic configuration. Then by projecting multi-agent system into the virtual tensegrity framework, we study how to reach desired formations with the constraint that inter-agent distances are upper or lower bounded. We design a control strategy based on the idea that each edge is assigned to be a virtual cable or strut, with which the physical distance constraints can be obtained.

We also propose a control strategy using the stress matrix associated with a universally rigid tensegrity framework to scale the formation. We show that the size

of the formation can be controlled by d pairs of agents whose configuration spans $\mathbb{R}^d$. By employing the orthogonal projection operator, the number of the agents controlling formation size can be reduced to two. We further design estimator-based control laws in combination with the stresses, in which the prescribed formation can be realized even only one agent knows the desired size of the entire formation.

Finally, we address the issue of formation tracking for multi-agent systems using only local measurements in local coordinate systems. Finite-time continuous estimators are designed to dynamically estimate the centroid of the whole group. We then make use of the estimations to propose a class of control algorithms such that the desired formation shape can be achieved and at the same time the external reference signal is tracked by the real centroid.

Samenvatting

Tensegrities hebben vanwege hun kenmerkende eigenschappen zoals een hoge stabiliteit, flexibele opschaling en robuustheid aanzienlijke aandacht gekregen van verschillende wetenschappelijke disciplines waaronder civiele techniek, biologie en werktuigbouwkunde. Gemotiveerd door deze gunstige eigenschappen bestuderen we vanuit een grafische en algebraïsche invalshoek, hoe een tensegrity structuur dusdanig kan worden opgebouwd zodat de kenmerkende en superieure eigenschappen van deze structuren kunnen worden overgenomen. Een ander centraal onderwerp van dit proefschrift is het ontwerpen van gedistribueerde regelaars voor het coördineren van multi-agent systemen door middel van virtuele tensegrities.

Als eerste bestuderen we hoe twee afzonderlijke rigide en respectivelijk infinitesimaal rigide tensegrity structuren kunnen worden samengebracht in een plat vlak. Voor infinitesimaal rigide tensigrity structuren is het bestaan van een evenwicht tussen zelf genererende mechanische spanningen op de verbonden structuren aangetoond. Dit geeft aan dat de infinitesimale rigiditeit voor deze structuren kan worden behouden. Ook kan het soort van de verbonden elementen worden bepaald door het teken van de bijbehorende spanning te controleren. Om de rigiditeit van de gecombineerde structuren aan te tonen hebben we gebruikt gemaakt van een disturbance-perturbation gebaseerde methode die op basis van de rigiditeit matrix op een gepaste manier nieuwe elementen toevoegt aan het structuur. Bovendien is de Henneberg constructie uitgebreid om super stabiele tensigrity structuren te laten ‘groeien’. We tonen aan dat de eigenschap van super stabiliteit kan worden behouden door toevoeging van een knooppunt, splitsing van een verbinding, en door samenvoeging van structuren.

Geïnspireed op de “turning back” methode voor het genereren van ijle matrices dragen we een numeriek algoritme voor waarmee, gegeven een bepaalde generieke configuratie, een universeel rigide tensegrity kan worden geconstrueerd.

Door een multi-agent systeem te projecteren op een virtuele tensegrity bestuderen we hoe een gewenste formatie kan worden gerealiseerd waarbij de afstand

tussen de agenten een boven- en ondergrens heeft. We ontwerpen een regel strategie die gebaseerd is op het idee dat elke verbinding in de structuur kan worden gezien als een virtuele kabel of staaf en waarmee aan de fysieke afstandsbeperkingen kan worden voldaan.

Door gebruik te maken van de spannings matrix die is geassocieerd aan een universeel rigide tensegrity, dragen we een regel strategie voor die het mogelijk maakt de schaal van de formatie aan te passen.

We tonen aan dat de grootte van de formatie kan worden bestuurd door d paren van agenten wiens configuratie de vectorruimte $\mathbb{R}^d$ voortbrengt. Door gebruik te maken van loodrechte vectorprojecties kan het aantal agenten dat de formatie bestuurt gereduceerd worden tot twee. Vervolgens ontwerpen we een op schattingen gebaseerde regel algoritme die, in combinatie met de spanningen in de tensegrity, de gewenste formatie kan realiseren. Dit algoritme werkt zelfs wanneer alleen één agent de gewenste grootte van de gehele formatie kent.

Tot slot richten we ons op de kwestie van formatie volging voor multi-agent systemen waarbij alleen lokale metingen in lokale coordinatenstelsels worden gebruikt. Eindige-tijd continue schattingsmethoden zijn ontworpen om het zwaartepunt van de groep agenten op een dynamische manier te schatten. Een klasse van regel algoritmes wordt voorgesteld die op basis van de geschatte waardes de gewenste formatie kunnen realiseren en tegelijkertijd, met het echte zwaartepunt, een extern referentie signaal kunnen volgen.