DESIGN, PROTOTYPING AND TESTING OF MODIFIED ALTMANN LINKAGE NETWORKS TO BE USED AS DEPLOYABLE STRUCTURES

A Thesis Submitted to
The Graduate School of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in Mechanical Engineering

by
Murat DEMİREL

December 2024
İZMİR

We approve the thesis of **Murat DEMİREL**

Examining Committee Members:

Prof. Dr. Gökhan KİPER
Department of Mechanical Engineering, İzmir Institute of Technology

Prof. Dr. Mehmet İsmet Can DEDE
Department of Mechanical Engineering, İzmir Institute of Technology

Prof. Dr. Engin AKTAŞ
Department of Civil Engineering, İzmir Institute of Technology

Assoc. Prof. Dr. Erkin GEZGİN
Department of Mechatronics Engineering, İzmir Katip Celebi University

Assist. Prof. Dr. Özgür KİLİT
Department of Mechanical Engineering, Yaşar University

10 December 2024

Prof. Dr. Gökhan KİPER
Supervisor, Department of Mechanical Engineering, İzmir Institute of Technology

Prof. Dr. Mehmet İsmet Can DEDE
Head of the Department of Mechanical Engineering

Prof. Dr. Mehtap EANES
Dean of the Graduate School

ACKNOWLEDGEMENTS

I am very thankful to my supervisor Dr. Gökhan Kiper, and to the thesis monitoring committee members Dr. M. İ. Can Dede and Dr. Erkin Gezgin for their support, advice and patience throughout this research.

I would like to express my gratitude to the former and current members of the Rasim Alizade Mechatronics Laboratory at the İzmir Institute of Technology. Special thanks to M.Sc. Tarık Kadak for his invaluable support during the case study project. Furthermore, I am very thankful to M.Sc. Görkem Yıldız for his previous study on the topic. I sincerely thank Mr. Bora Tekin, Mr. Erol Yiğit Dündar, Ms. Gizem Merve Gündüz and M.Sc. Merve Özkahya for their help during the prototyping activities of the case study research project. Additionally, I am grateful to Mr. Gökhan Arslan, Ms. Yağmur Demir, Mr. Emirhan İnanç and M.Sc. Tuğrul Yılmaz for their support in the design and manufacturing of the deployable network prototypes.

Furthermore, a part of this research is carried out in cooperation with Bias Engineering Company. I wish to express my sincere thanks to the entire Bias team, and especially to M.Sc. Ender Koç, for his guidance during the finite element analysis studies and for providing the facilities of the Bias Technopark İzmir Office.

My deepest gratitude goes foremost to my love, Ms. Burcu Deveci Demirel, for encouraging me and being here. I would like to acknowledge Dr. Emre Uzunoğlu and Mr. Bülent Aslantürk for their significant impact on my academic and engineering careers that resulted in this thesis.

The case study research project of this thesis has been funded by the Scientific and Technological Research Council of Türkiye through the 1005 National New Ideas and New Products Research Funding Program, with the project identification number 123M090.

ABSTRACT

DESIGN, PROTOTYPING AND TESTING OF MODIFIED ALTMANN LINKAGE NETWORKS TO BE USED AS DEPLOYABLE STRUCTURES

This thesis focuses on the two modified Altmann linkages and deployable networks obtained from them. The modified linkages are special cases of general line-symmetric Bricard linkages. These linkages provide constructional advantages compared to the original Altmann linkage, and one of the modified linkages is novel for the literature. Kinematic analyses of both linkages are performed. The deployable networks composed of these linkages are classified. One of the two linkages is selected to describe the characteristics and motion behavior of each network. Built prototypes of some networks are presented.

A dome-like deployable network is selected as a case study. The case study covers the design, prototyping and testing of a deployable tent mechanism. Two physical prototype designs, structural simulations and some test results are presented. The parametric model for the geometry of the designs is developed and implemented in Microsoft Excel®. A custom API is designed to integrate the parametric model into the solid modeling environment (Solidworks®) as an Excel add-in. Simulations are run to explore the nonlinear elastic deformation behavior of the first prototype under static load conditions. MSC Apex® is used as the pre- and post-processor, whereas MSC Nastran® is the solver. The presented finite element models are generated utilizing the API library of MSC Apex®. Another custom API workflow is introduced for this task. The construction and test results of the first prototype are presented, whereas the second prototype is to be built as a future study.

ÖZET

KATLANABİLİR YAPI OLARAK KULLANILAN DEĞİŞTİRİLMİŞ ALTMANN MEKANİZMA AĞLARININ TASARIMI, PROTOTİP VE İMALAT TESTLERİ

Bu tez çalışması, iki değiştirilmiş Altmann mekanizması ve bunlardan elde edilen katlanabilir ağ yapılar üzerine odaklanmaktadır. Değiştirilmiş mekanizmalar, genel doğru-simetrik Bricard mekanizmasının özel durumlarıdır. Bu mekanizmalar, orijinal Altmann mekanizmasına göre yapısal avantajlara sahip olup değiştirilmiş mekanizmalardan biri özgündür. Her iki mekanizmanın kinematik analizleri yapılmıştır. Bu mekanizmalarla oluşturulan katlanabilir ağ yapılar sınıflandırılmıştır. Ağ yapıların her birinin özellikleri ve hareket davranışlarını açıklamak için iki mekanizmadan biri seçilmiştir. Bazı ağ yapıların üretilmiş prototipleri sunulmuştur.

Kubbe benzeri bir katlanabilir ağ yapısı vaka çalışması olarak seçilmiştir. Vaka çalışması, bir katlanabilir çadır mekanizmasının tasarımını, prototip üretimini ve imalat testlerini kapsamaktadır. İki fiziksel prototip tasarımı, yapısal benzetim çalışmaları ve bazı test sonuçları sunulmuştur. Tasarım geometrisinin parametrik modeli geliştirilerek Microsoft Excel®'de tanımlanmıştır. Katı model ortamına (Solidworks®) bu parametrik modelin aktarımını sağlamak için Excel eklentisi olarak özel bir API geliştirilmiştir. İlk prototipin statik yük koşulları altındaki doğrusal olmayan elastik deformasyon davranışını incelemek için benzetimler koşturulmuştur. Ön- ve son-işlemci olarak MSC Apex®, çözücü olarak ise MSC Nastran® kullanılmıştır. Sunulan sonlu eleman modelleri MSC Apex®'in API kütüphanesi kullanılarak oluşturulmuştur. Bu görev için bir başka özel API iş-akışı oluşturulmuştur. İlk prototipin üretimi ve test sonuçları sunulmuş olup ikinci prototipin üretim ve testleri ileriki çalışmalardır.

TABLE OF CONTENTS

LIST OF FIGURES ix

LIST OF TABLES xiv

CHAPTER 1. INTRODUCTION 1

1.1. Problem Definition 2
1.2. Aim of the Thesis 5
1.3. Outline of the Thesis 5

CHAPTER 2. LITERATURE SURVEY 6

2.1. Sarrus Linkage 7
2.2. Bennett Linkage 7
2.3. Myard Linkage 9
2.4. Bricard Linkage 10
2.5. Altmann Linkage 10

CHAPTER 3. DEPLOYABLE NETWORKS OF THE MODIFIED ALTMANN LINKAGES 13

3.1. Description of the Basic Modules 13

3.1.1. Special Case-I: Zero Link Length Hubs 16
3.1.2. Special Case-II: Non-Zero Link Length Hubs 19

3.2. Deployable Networks 22

3.2.1. Type-I: Two Common Bars — Version-I: Diagonally Multiplied Version 25
3.2.2. Type-II: One Common Bar and One Common Hub 29

3.2.2.1. Version-I: Linearly Multiplied Version 30
3.2.2.2. Version-II: Diagonally Multiplied Version 30
3.2.2.3. Version-III: 4-loop Version 32

3.2.3. Type-III: Two New Joints 33

3.2.3.1. Version-I: Extra Modified Altmann Loop Version 34
3.2.3.2. Version-II: Extra Parallelogram Loop Version 35
3.2.3.3. Version-III: 4-loop Version 37

3.2.4. Type-IV: Four New Joints 38

3.2.4.1. Version-I: Extra Modified Altmann Loop Version 38
3.2.4.2. Version-II: 5-loop version 40

3.3. Prototypes 40
3.4. Discussion of the Chapter 45

CHAPTER 4. CUSTOM API DESIGN FOR PARAMETRIC DESIGN AND SIMULATIONS OF DEPLOYABLE STRUCTURES 46
4.1. Custom API Design for Parametric Design and CAD Simulations 47
4.1.1. Description of the Parametric Design Integrator 48
4.1.2. Workflow of the Program 51
4.1.3. Kinematics Library 52
4.2. Custom API Design for FEA Simulations Automation 57
4.2.1. Description of the Program 58
4.2.2. Workflow of the Program 60

CHAPTER 5. CASE STUDY: DESIGN AND PARAMETRIC MODEL 64
5.1. Description of the Case Study 64
5.1.1. Spherical Network of the Case Study 65
5.1.2. Products in the Market and Their Specifications 65
5.1.3. Design Criteria of the Case Study 70
5.2. Conceptual Design Alternatives 71
5.3. Design of the Spherical Network 73
5.3.1. Design of the Module 75
5.3.2. Design of the Module Connections 78
5.3.3. Design of the Ground Fixing System 80
5.4. Parametric Model of the Parts 80
5.4.1. Bolt (Bo_{1-7}), Nut (N_{1-2}), Washer (W_{1-3}) and Spring Pin (SP_{1-2}) Parameters 82
5.4.2. Tube (T_{1-7}) Parameters 86
5.4.3. Sheet Metal End (S_{1-4}) Parameters 88
5.4.4. End Cap (E_{1-4}) Parameters 90
5.4.5. Binary Hub (B_{1-2}) and Quaternary Hub (Q_{1-2}) Parameters 91
5.4.6. Sheet Link (SL_{1-6}) Parameters 95
5.4.7. Hollow Connector (HC_{1-2}) and Connector (C_{1-7}) Parameters 97
5.5. Kinematic Analysis of a Module 102
5.5.1. Analysis of Loops L_{1-4} 103
5.5.2. Analysis of Loops L_{5-8} 106
5.6. Specifications of the Network 110

CHAPTER 6. CASE STUDY: FEA SIMULATIONS AND PROTOTYPES 116
6.1. Prototype-I: Design, FEA Simulations and Tests 117
6.1.1. Design of Prototype-I 117

6.1.2. FEA Simulations of Prototype-I 118
6.1.3. Manufacturing and Assembly of Prototype-I 122
6.1.4. Tests and Results of Prototype-I 123
6.1.5. Design of Revised Prototype-I 124
6.1.6. FEA Simulations of Revised Prototype-I 127
6.2. Prototype-II Design 131

CHAPTER 7. CONCLUSIONS 133

REFERENCES 135

LIST OF FIGURES

Figure **Page**

Figure 1.1. Linkages and their deployment: a) Altmann 6R linkage and b) its deployment, c) the modified Altmann linkage and d) its deployment. 3

Figure 1.2. Deployment of the spherical network in Kiper (2016). 4

Figure 2.1. Sarrus 6R linkage and a deployable network of its a) the linkage schematics and a prototype (Source: Chen and You, 2011), b) a tetrakis hexahedral polyhedral linkage (Source: Kiper and Söylemez, 2010). 8

Figure 2.2. Bennett (1903) 4R linkage and a deployable network of its a) the linkage schematics and the original prototype by Bennett (Source: Chen and You, 2011), b) a network in the folded and deployed forms (Source: Melin, 2004) 8

Figure 2.3. Myard 5R linkage and a deployable network of its a) the linkage schematics (Source: Chen and You, 2011), b) a network of three Myard units in the folded and deployed forms (Source: Huang et al., 2012). 9

Figure 2.4. Trihedral Bricard 6R linkage a) a prototype (Source: Baker, 1980), b) a network in the folded and deployed forms (Source: Cui et al., 2012). 10

Figure 2.5. a) Altmann linkage (Source: modified from (Altmann, 1954)), b) a deployable network schematic based on the modified Altmann linkage (Source: Atarer et al., 2017). 11

Figure 3.1. DH convention and general line-symmetric Bricard linkage a) convention redrawn from (McCarthy 2010, 256-57) b) linkage. 14

Figure 3.2. Altmann's 6R linkage a) joint numbering, b) deployment sequence. 15

Figure 3.3. Special case-I: zero link length hubs a) CAD model, b) schematic. 17

Figure 3.4. Details of a Special case-I: zero link length hubs loop a) deployment, b) assembly modes. 19

Figure 3.5. Special case-II: non-zero link length hubs a) CAD model, b) schematic. 20

Figure 3.6. Details of a Special case-II: zero link length hubs loop a) deployment, b) assembly modes 22

Figure **Page**

Figure 3.7. Details of Design alternative-I: short link to short link a) schematic, b) deployment. 27

Figure 3.8. Details of Design alternative-II: long link to short link a) schematic, b) deployment. 27

Figure 3.9. Details of Design alternative-III: short link to short link with a kink angle a) schematic, b) deployment. 28

Figure 3.10. Details of Design alternative-III: mirror-angulated variation a) schematic, b) deployment. 29

Figure 3.11. Details of Design alternative-IV: long link to short link with a kink angle a) schematic b) deployment. 29

Figure 3.12. Details of Version-I: linearly multiplied version a) schematic, b) deployment. 31

Figure 3.13. Details of Version-II: diagonally multiplied version a) schematic, b) deployment. 31

Figure 3.14. Details of Version-III: 4-loop version a) schematic, b) deployment. 33

Figure 3.15. Details of Version-I: extra modified Altmann loop version a) schematic, b) deployment. 35

Figure 3.16. Details of Version-II: extra parallelogram loop version a) schematic, b) deployment. 36

Figure 3.17. Details of Version-III: 4-loop version a) schematic, b) deployment. 38

Figure 3.18. Details of Version-I: extra Altmann loop version a) schematic, b) deployment. 39

Figure 3.19. Details of Type-IV Version-II: 5-loop version a) schematic, b) deployment. 40

Figure 3.20. A prototype of Type-I Version-I Design Alternative-I a) CAD model, b) deployment. 41

Figure 3.21. A prototype of Type-I Version-I Design Alternative-IV a) CAD model, b) deployment. 41

Figure 3.22. A prototype of Type-II Version-III a) CAD model, b) deployment. 42

Figure 3.23. Another prototype of Type-II Version-III a) CAD model, b) deployment. 43

Figure **Page**

Figure 3.24. The folded and deployed states of the prototype of a 72-loop Type-II Version-II assembly. 43

Figure 3.25. A prototype of Type-III Version-II a) CAD model, b) deployment. 44

Figure 3.26. A prototype of Type-III Version-III network) CAD model, b) deployment. 44

Figure 4.1. Main flow of the parametric design integrator program. 48

Figure 4.2. Details of PDI a) input worksheet of PDI, b) main program GUI. 49

Figure 4.3. Output worksheet of PDI. 51

Figure 4.4. Flowchart of PDI. 53

Figure 4.5. Line-symmetry example in y-direction 55

Figure 4.6. Plane-symmetry example about xy-plane. 56

Figure 4.7. Main program flow of custom API designs. 58

Figure 4.8. Object model diagram of the API library. 59

Figure 4.9. Flowchart of the custom API design for FEA simulation automation. 63

Figure 5.1. Spherical network of the case study from a loop to network. 66

Figure 5.2. DRASH Shelter a) XB series shelters b) deployment of a DRASH-6XB shelter (Source: ReeveEMS, 2024). 67

Figure 5.3. Snap-through effect (Source: Li and Xu, 2017). 68

Figure 5.4. Geometric incompatibilities of bistable SLE structures a) angular distortions b) peripheral incompatibilities (Source: Roovers, 2017). 69

Figure 5.5. ASAP shelters (a) ASAP-12 (Source: Deployed Logix, 2024) , (b) Deployment of a shelter (Source: snapshots taken from video by Deployed Logix (2021)). 69

Figure 5.6. Conceptual design alternatives a) conceptual design-I based on Kiper (2016) b) conceptual design-II. 72

Figure 5.7. Design of the spherical network. 74

Figure 5.8. Design of $Links_{1\text{-}3,7}$. 76

Figure 5.9. Design of the quaternary hub (Q). 76

Figure **Page**

Figure 5.10. Design of the binary hub (B). 77

Figure 5.11. Design of the three links connection. 77

Figure 5.12. Design of the half-modules connection. 78

Figure 5.13. Three modules connection design a) deployed form, b) folded form. 79

Figure 5.14. Module connection parts (SL_{1-6}, HC_{1-2}, C_{1-4}). 79

Figure 5.15. Network details a) nine-module network in the deployed form b) elongated square cupola. 80

Figure 5.16. Design of ground fixing system. 81

Figure 5.17. Parameters of machine elements: bolts ($Bo_{1,4}$), nut (N_1), washer (W_1), spring pins (SP_{1-2}). 82

Figure 5.18. Parameters of tube (T_1) in three views. 86

Figure 5.19. Parameters of sheet metal ends (S_{1-4}). 89

Figure 5.20. Parameters of end caps (E_{1-3}). 90

Figure 5.21. Binary hubs (B_{1-2}) parameters. 92

Figure 5.22. Parameters of quaternary hubs (Q_{1-2}). 94

Figure 5.23. Parameters of sheet links (SL_{1-2}). 96

Figure 5.24. Parameters of hollow connectors (HC_{1-2}). 97

Figure 5.25. Parameters of connectors (C_{1-2}). 98

Figure 5.26. Details of SL_{2-3} length computation. 100

Figure 5.27. Details of the lines L_{2-5} 101

Figure 5.28. Module with the attached points (P_{1-48}) of the loops (L_{1-8}). 102

Figure 5.29. Schematic of the loop (L_1). 103

Figure 5.30. Schematic of the loops (L_{2-4}) 105

Figure 5.31. Details of the symmetry between two equilateral half-modules. 107

Figure 5.32. Schematic of the loop (L_5). 108

Figure 5.33. Schematic of the loops (L_{5-8}) 109

Figure 5.34. Elongated square cupola parameters. 111

Figure **Page**

Figure 5.35. Points $P_{4C,4CS}$ defined to compute side length l_C. 111

Figure 5.36. Folded volume of the network and its parameters. 113

Figure 5.37. Floor area of the network a) rectangular octagon, b) CAD model. 114

Figure 5.38. Parametric model computations in Microsoft® Excel. 115

Figure 6.1. Prototyping activities. 116

Figure 6.2. Design of Prototype-I. 118

Figure 6.3. FEA simulation model of Prototype I. 120

Figure 6.4. FEA simulation results of Prototype-I a) translational displacement results, b) Von Mises stress results. 121

Figure 6.5. A module assembly of Prototype-I a) deployed form b) folded form. 122

Figure 6.6. Assembly of Prototype-I a) deployed form b) folded form. 123

Figure 6.7. Deployment test result of Prototype-I. 124

Figure 6.8. Scaled translational displacement results of Prototype-I FEA simulations (true scale = 7). 125

Figure 6.9. Design of revised Prototype-I. 125

Figure 6.10. Details of modules connection parts in revised Prototype-I a) joints R_{B1} and R_{SL1}, b) angular displacement of θ_1, θ_{B1} and θ_{SL1} during deployment. 127

Figure 6.11. FEA simulation model of revised Prototype-I. 128

Figure 6.12. FEA simulation results of revised Prototype-I a) translational displacement results, b) Von Mises stress results. 129

Figure 6.13. Module connection part displacements of revised Prototype-I. 130

Figure 6.14. Details of Prototype-II a) design b) angular displacement of θ_1, θ_{B1} and θ_{SL1} during deployment. 131

LIST OF TABLES

Table **Page**

Table 3.1. Network types (*: novel versions). 25

Table 5.1. Comparison of the available products in the market. 70

Table 5.2. Design criteria 71

Table 5.3. DH parameters of a module of the spherical network 81

Table 5.4. Parameters of Bo_{1-7}. 83

Table 5.5. Parameters of N_{1-2}. 84

Table 5.6. Parameters of W_{1-3}. 85

Table 5.7. Parameters of SP_{1-2}. 85

Table 5.8. Parameters of T_{1-7}. 87

Table 5.9. Parameters of S_{1-4} 89

Table 5.10. Parameters of E_{1-4}. 91

Table 5.11. Parameters of B_{1-2}. 92

Table 5.12. Parameters of Q_{1-2}. 94

Table 5.13. Parameters of SL_{1-6}. 96

Table 5.14. Parameters of HC_{1-2}. 98

Table 5.15. Parameters of C_{1-4}. 99

Table 5.16. Definition of the loop (L_1) points (P_{1-8}). 104

Table 5.17. Definition of the loops (L_{2-4}) points (P_{9-24}). 106

Table 5.18. Definition of the loop (L_5) points (P_{25-32}) 108

Table 5.19. Definition of the points P_{33-48} of loops L_{2-4}. 110

Table 5.20. Definition of points P_{25-32} of loop L_5. 112

Table 5.21. Folded volume parameters. 113

Table 5.22. Mass values. 114

Table 6.1. Design specifications of Prototype-II. 132

CHAPTER 1

INTRODUCTION

Deployable structures are prefabricated structures that can be transformed from a folded, compact configuration to a predetermined, deployed form. In their deployed form, they are stable and capable of carrying loads (Gantes, 2018, 1). Deployable structures is an interdisciplinary field for architects, civil engineers, mechanical engineers and aerospace engineers. These structures are used for different applications such as disaster relief shelters, construction and maintenance sites, military operations, recreational purposes, deployable bridges, towers, shelters, retractable roofs for sports stadiums, and adaptable building façades. They are also common in the space industry for deployable antennas, masts, and satellites, as well as in everyday items like umbrellas, foldable chairs, medical devices, and toys (Arnouts et al., 2019). Several classifications of deployable structures exist in the literature and have been reviewed by Fenci and Currie (2012). The most common characteristic in the classification of deployable structures is on the rigid and deformable components. Structures that use rigid components generally comprise pantograph, scissor and origami mechanisms. Structures that use rigid deformable components generally comprise compliant links and/or joints, tensegrity and inflatable structures. The main feature of all deployable structures is that they function as mechanisms, transforming between multiple predetermined configurations, and as structural systems, bearing loads in their stable configurations (Arnouts et al., 2019). Arnouts et al. (2019) classify scissor structures into three categories according to the structural behavior (internal stress status) of the links: In the first category, foldable, i.e., deployable mechanisms are geometrically compatible (stress-free situation or no internal stress) in the folded form, during deployment and in the deployed form. This type requires locking joints in the deployed form since it is still a mobile mechanism. As the second category, incompatible structures are geometrically incompatible (develop internal stress) in the deployed and folded form. This type allows designing self-deploying structures, but internal stress is not desired in the deployed form because the structure is more susceptible to buckling, so they are not much used in applications. As the last category, bistable structures develop internal stress during deployment (geometrically

incompatible) but have no internal stress in the deployed and folded form (geometrically compatible). This results in self-locking in the deployed form since the links will remain stable. This structural classification is valid for deployable structures with rigid (or more precisely stiff) components. Deployable mechanisms are more suitable for outer space applications since gravity effects are negligible. Usually, cables are used to provide the required stiffness in the deployed form. However, for earth applications due to sagging, extra attention is needed to lock the joints at the deployed form. Use of additional elements (bars) for locking can increase the deployment time. In contrast to deployable mechanisms, the self-locking feature of bistable mechanisms makes them suitable for earth applications. Such structures are frequently used for deployable tent and shelter applications. However, their design is complex and requires attention during deployment. They can easily break due to undesirable buckling effects on the links. This thesis focuses on deployable mechanisms.

Deployable networks are usually obtained by assembling over-constrained linkages. Chen (2003, 45-125) presents a methodology for the deployable networks of spatial over-constrained Bennett and Bricard linkages.

This thesis focuses on the deployable networks of two modified Altmann linkages A dome-like network to be used as a deployable shelter mechanism is worked out in detail as a case study. The following sections present the problem definition and state the aim and outline of the thesis.

1.1. Problem Definition

Altmann's (1954) 6R (R: revolute) linkage (Figure 1.1a) is one of the single-loop and single-dof spatial over-constrained linkages in the literature. The linkage is a special case of a general line-symmetric Bricard (1927, 316) 6R linkage, so three consecutive links remain line-symmetric with the other three links throughout the motion of the linkage. Altmann linkage has very suitable deployed and folded forms for deployable structures due to its symmetry (see Figure 1.1b). However, the construction of its networks is not easy due to zero joint offsets. To solve this problem, Kiper (2016) registered a patent for deployable networks of the modified Altmann linkages. This modified linkage is shown in Figure 1.1c. The modification is via the non-zero joint offsets. The pair of joint axes 1-5, 2-6, 3-5 and 4-2 do not meet due to joint offsets in blue links. The modified

Altmann linkage still has the line-symmetry property and is still a special case of a general line-symmetric Bricard linkage. The deployed and folded forms are suitable for deployable networks (see Figure 1.1d). An alternative modified linkage is obtained in this thesis, following a similar approach.

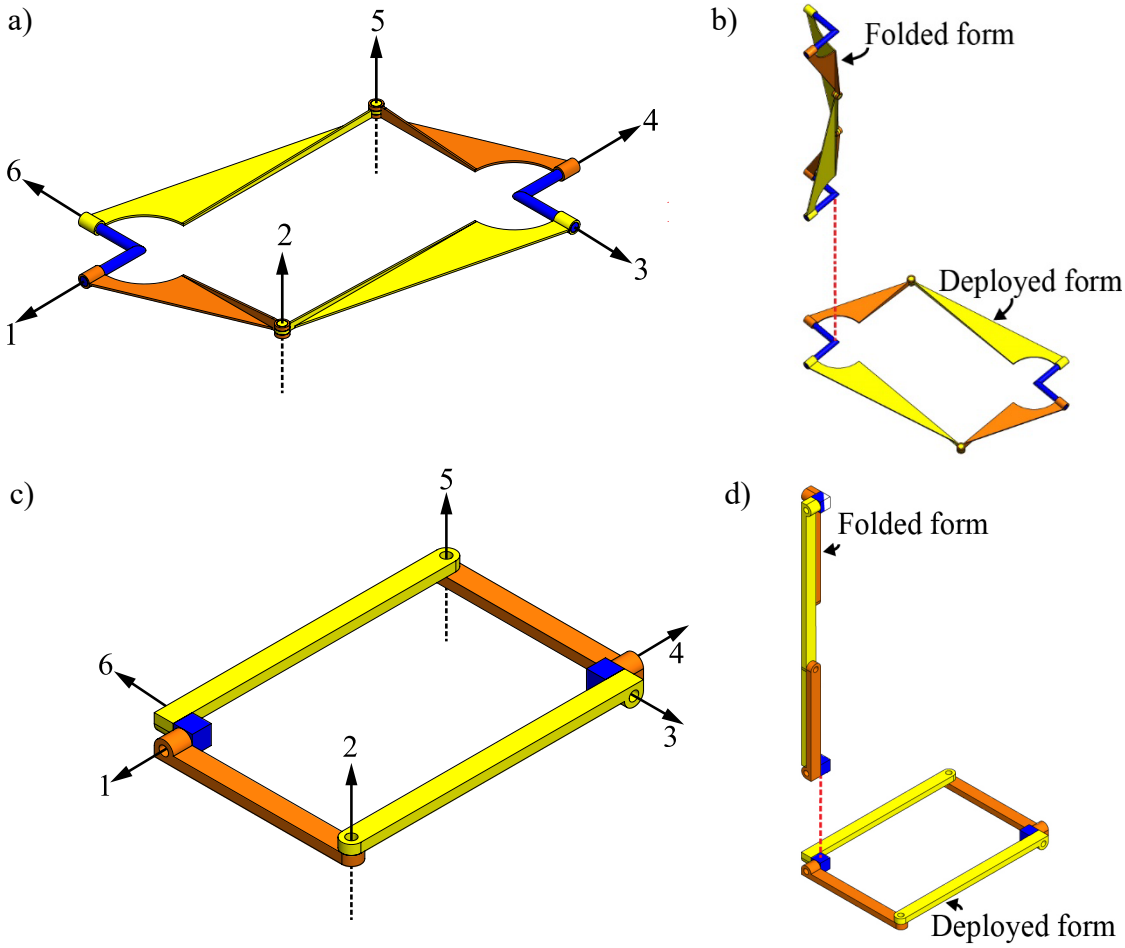


Figure 1.1. Linkages and their deployment: a) Altmann 6R linkage and b) its deployment, c) the modified Altmann linkage and d) its deployment.

The deployment of a dome-like deployable network of Kiper (2016) is shown in Figure 1.2. A double-layer structure is used as a module. Each module consists of eight modified Altmann linkage loops. The linkage in Figure 1.2 is composed of nine such modules. Moreover, different modified Altmann linkage networks are available in the literature (Song et al. 2016; Atarer et al. 2017). However, the mathematical models and proper classification of these models are not presented in these papers. In this thesis, new networks and parametric models of the networks are examined.

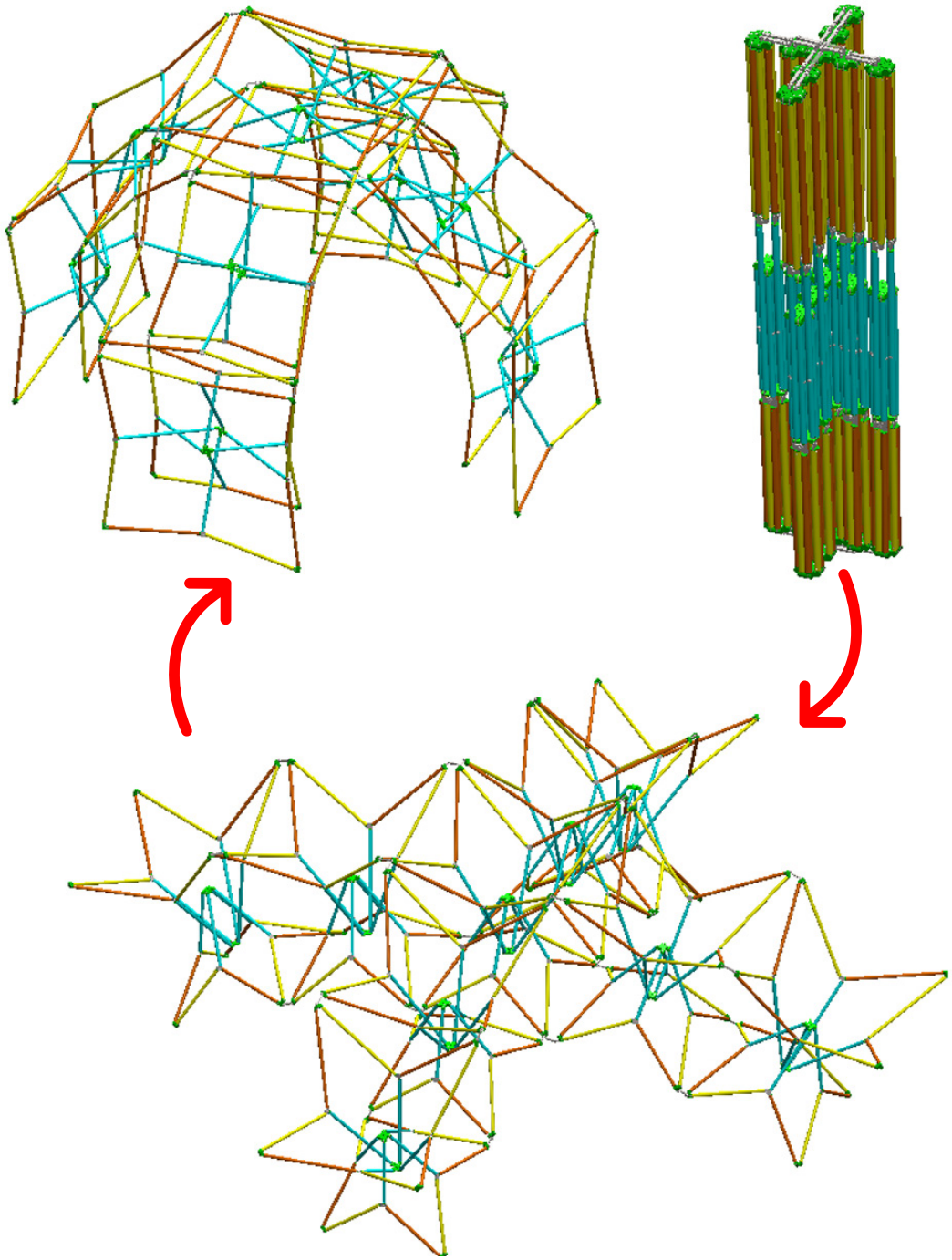


Figure 1.2. Deployment of the spherical network in Kiper (2016).

Common applications of these networks are deployable tents, shelters, billboard mechanisms, etc. The design of these deployable structures may be complicated and time-consuming. Mathematical models of the networks would be valuable for such designs. A detailed survey on single-loop and single-dof overconstrained linkages and their deployable networks is presented in Chapter 2.

1.2. Aim of the Thesis

This work aims to

1) Investigate existing networks of modified Altmann linkages and discover novel networks.

2) Develop mathematical models of some networks for parametric design.

3) Develop a parametric design environment by implementing a mathematical model into a software.

4) Work on constructional design and prototype for a specific application using one of the networks.

1.3. Outline of the Thesis

The thesis consists of 7 Chapters: Introduction, Literature Survey, Deployable Networks of the Modified Altmann Linkage, Custom API Design for Parametric Design and Simulation Automations of Deployable Structures, Case Study: Design and Parametric Model, Case Study: FEA Simulations and Prototypes, Conclusions.

Chapter 2 presents a literature review of existing single loop and single dof overconstrained linkages and their deployable networks.

In Chapter 3, two modified Altmann linkages are explained. Their position-level kinematic analysis is performed. Then, deployable networks of the linkages are classified. A suitable linkage among the two options is selected for each network and the motion of the network is described.

Chapter 4 introduces the custom API designs for integrating the parametric design environment (Microsoft Excel®) with the Computer-aided Drawing (CAD) environment (Solidworks®) and finite element analysis (FEA) simulation automation for deployable structures.

Chapters 5-6 focus on the dome-like deployable network case study as a tent application. Chapter 5 presents the design and parametric model of the case study, while Chapter 6 covers the FEA simulations, description of the prototype and test results.

The conclusions of the thesis are stated in Chapter 7. The contributions of this work are summarized and future research directions are discussed.

CHAPTER 2

LITERATURE SURVEY[1]

Deployable structures are mobile assemblies that, unlike most other applications, do not aim to move but rather to achieve specific configurations, usually due to compact storage/transfer constraints under various service conditions (Kiper and Söylemez 2009, 131). Among different application areas of deployable structures, architecture, furniture industry, aerospace industry and toy industry applications received attention where the aerospace industry takes the lead (Kiper and Söylemez 2009; Pellegrino 2021, v; Chen and You 2011, 1-3). New deployable structures and their networks are mostly obtained by combining over-constrained linkages. Over-constrained linkages do not obey the Chebychev-Grübler-Kutzbach (CGK) mobility formula (Philips 2007, 14, 28). Typically spatial over-constrained linkages with single-loop and single-degrees of freedom (dof) are used in deployable networks.

This Chapter presents a literature survey on the existing single-loop and single-dof spatial overconstrained linkages and their deployable networks. If only revolute (R) joints are used, at least four links are required to obtain spatial over-constrained linkages (Philips 2006, Chapter 20). Sarrus (1853) 6R linkage is the first published spatial over-constrained linkage in the 19th century. After that, Bricard (1897, 125-47; 2012, 12-35; 1927, 316-19) 6R linkages, Bennett (1903; 1914) 4R linkage, Myard (1931) 5R linkage, Goldberg (1943) 5R and 6R linkages, Bennett (1905) 6R linkages and Altmann (1954, 248-49) 6R linkage are presented as examples of single-loop and single dof spatial over-constrained linkages. Single-dof deployable networks of these linkages can be obtained typically by connecting identical or scaled copies of the loops. This operation can be achieved via two methods: 1) connecting adjacent loops with common links or joints and 2) connecting with new links or joints without any common elements.

In the following sections, Sarrus 6R linkage, Bennett 4R linkage, Myard 5R linkage, Bennett 6R linkages, Bricard 6R linkages, Altmann linkage and their deployable networks are reviewed.

[1] The contents of Chapters 2 and 3 will be submitted as to an SCIE journal. The journal draft is titled “Deployable Networks of Modified Altmann Linkages”.

2.1. Sarrus Linkage

Sarrus linkage (1853) comprises six binary links (see Figure 2.1a). The metric constraints of the linkage are to have two sets of three consequtive parallel R-joints, where the two sets are not parallel to each other. Nonparallel joint axes have a certain angle between them, which is $\pi/2$ in most applications. Sarrus linkage provides a linear motion between two opposite platforms of the linkage in the perpendicular direction to the two sets of R-joints. Hoberman (2001) sphere is a well-known example that includes Sarrus linkages and scissor-like elements (Figure 2.1b). Calavatra (1981), and Huang et al. (2012) presented different networks based on Sarrus linkage. Kiper and Söylemez (2010) proposed deployable polyhedral linkages comprising Sarrus loops. Bouten (2015) introduced flat and polar networks. Other successful examples of deployable structures based on Sarrus linkage are presented in (Yang and You, 2020) and (Yang et al., 2021).

2.2. Bennett Linkage

Bennett (1903) linkage shown in Figure 2.2a is the only possible 4R mechanism where the joint axes are neither concurrent nor parallel. Two opposite links have equal link lengths and twist angles. Additionally, offsets should be zero, and there is a relationship between the kinematic parameters according to the law of sine. Baker and Hu (1986) tried to connect two similar Bennett linkages with only R-joints, but they obtained immobile structures. Chen and You (2002) showed that it is possible to get mobile structures by connecting Bennet linkages with two extra R-joints where a new and scaled Bennett linkage is obtained between the connected linkages. Chen (2003) presented deployable Bennett networks with single- and multi-layers. The work is essential in guiding other studies on obtaining single and multi-layer networks of single-dof over-constrained linkages. Chen and Baker (2005), Chen and You (2005, 2008a), Melin (2004) (Figure 2.2b), Isaak (2006), Kiper and Söylemez (2009), Tian and Chen (2010), Guo and You (2012) and Yang and Chen (2015) presented successful networks of Bennett linkages.

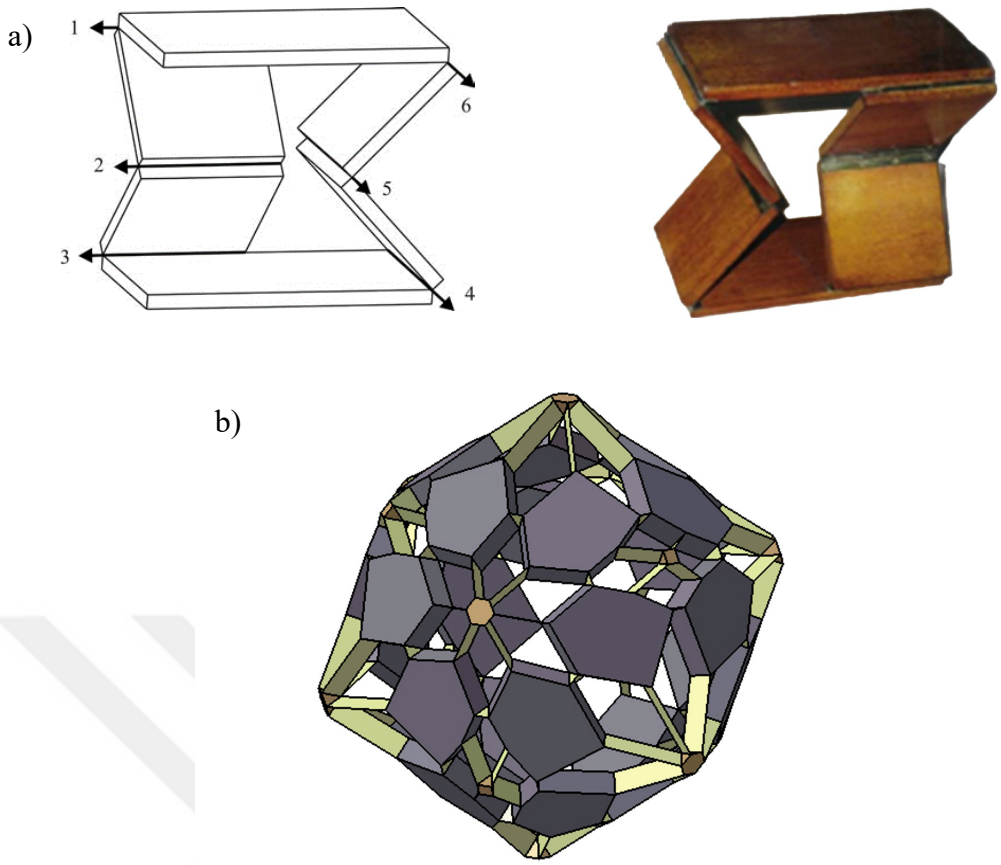
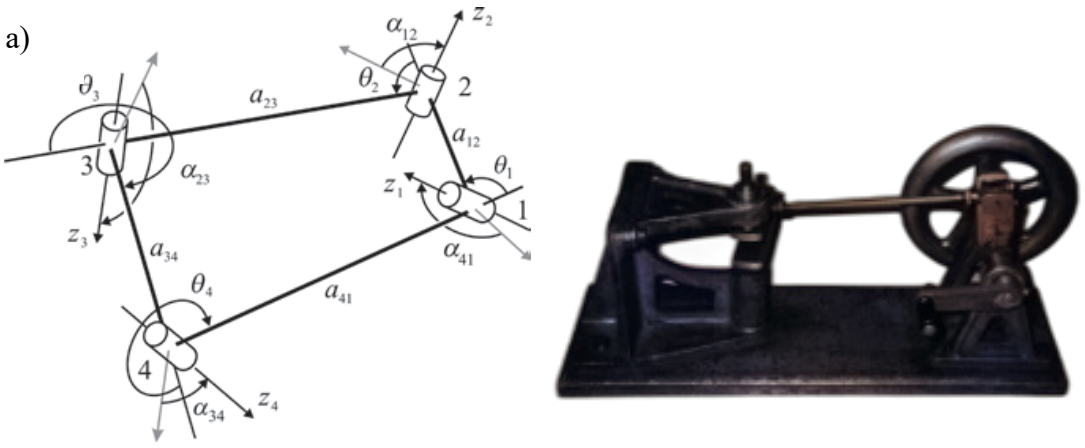


Figure 2.1. Sarrus 6R linkage and a deployable network of its a) the linkage schematics and a prototype (Source: Chen and You, 2011), b) a tetrakis hexahedral polyhedral linkage (Source: Kiper and Söylemez, 2010).



(cont. on next page)

Figure 2.2. Bennett (1903) 4R linkage and a deployable network of its a) the linkage schematics and the original prototype by Bennett (Source: Chen and You, 2011), b) a network in the folded and deployed forms (Source: Melin, 2004).

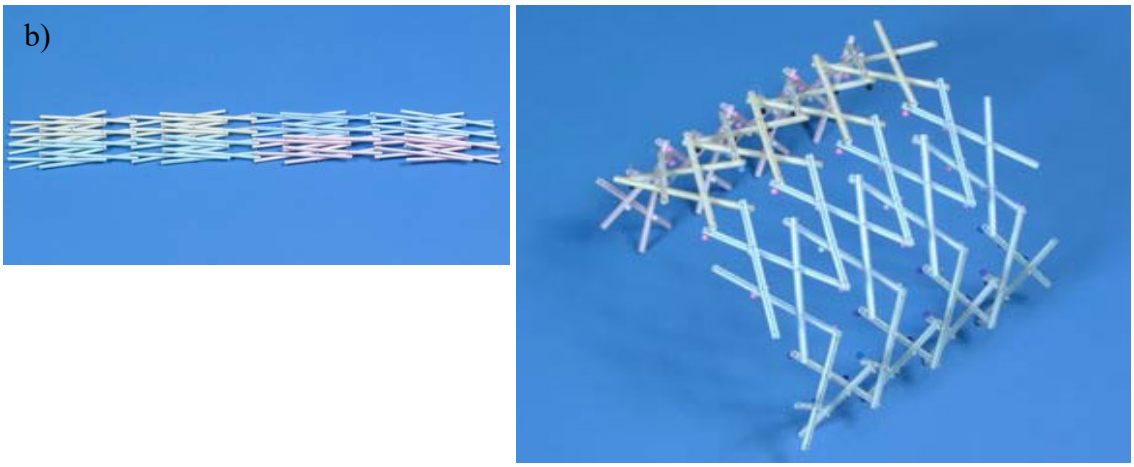


Figure 2.2 (cont.)

2.3. Myard Linkage

Myard (1931) 5R linkage shown in Figure 2.3a can be obtained by combining two mirror-symmetric Bennett loops and then removing two common links (Baker, 1979). Networks of Myard linkage are presented by Briand and You (2007), Chen and You (2008b), Liu and Chen (2009), Qi et al. (2011), and Huang et al. (2012) (Figure 2.3b).

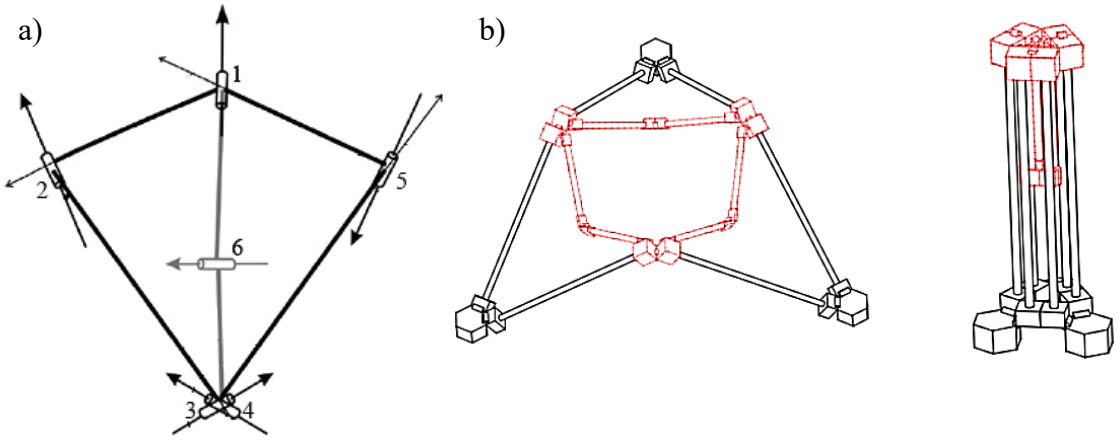


Figure 2.3. Myard 5R linkage and a deployable network of its a) the linkage schematics (Source: Chen and You, 2011), b) a network of three Myard units in the folded and deployed forms (Source: Huang et al., 2012).

2.4. Bricard Linkage

Six types of 6R linkage are discovered by Bricard (1897, 125-47; 2012, 12-35; 1927, 316-19): general line-symmetric, general plane-symmetric, trihedral (Figure 2.4a), line-symmetric octahedral, plane-symmetric octahedral, and doubly collapsible octahedral linkages. Bricard pointed out that line-symmetric octahedral Bricard linkage is a special case of general line-symmetric Bricard linkage (Bricard 1927, 316-19; Baker 1980). Chen (2003) made use of an equilateral Bricard linkage which is obtained by combining a general line-symmetric and a trihedral linkage. Cui et al. (2012) designed a deployable antenna using two equilateral general line-symmetric Bricard linkages to obtain tripod modules (Figure 2.4b). Three limbs are used to connect the modules. The deployed form of the mechanism is a triangular prism. Other deployable networks of Bricard linkages are presented by Huang et al. (2012; 2016), Zhang and Chen (2018), Lu et al. (2019) and Zhang et al. (2023).

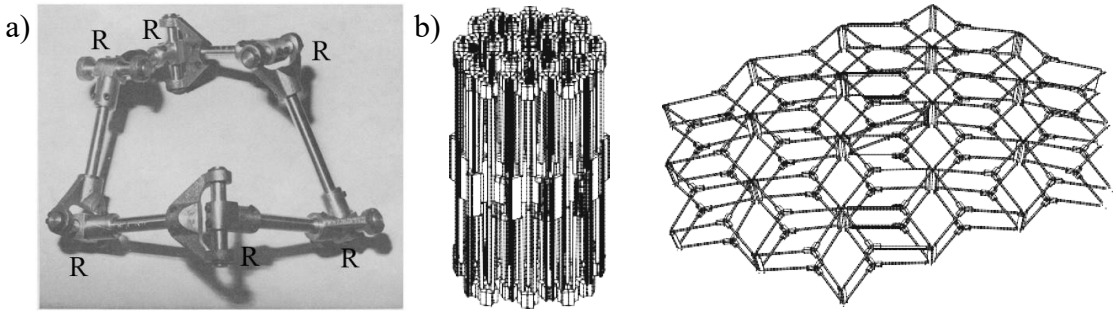


Figure 2.4. Trihedral Bricard 6R linkage a) a prototype (Source: Baker, 1980), b) a network in the folded and deployed forms (Source: Cui et al., 2012).

2.5. Altmann Linkage

Altmann (1954, 248-49) 6R linkage is a special case of both Bricard general line-symmetric linkage and Bricard trihedral linkage, where consecutive R-joint axes are perpendicular to each other, two of the link lengths and all joint offsets are zero. Since two of the link lengths are zero, the linkage can be named as a line-symmetric RURU (U:

universal joint) linkage as well. Altmann linkage resembles a rectangle in the deployed form and can fold into a bundle form. These forms make the linkage suitable as a deployable structure. However, constructional design of the linkage is complicated due to U-joints without offsets. Kiper (2016) proposed double-layer networks of Altmann linkage by modifying the original linkage by adding four non-zero joint offsets. Modified linkages can be connected with each other via two common links and a common joint. Four loops connected recursively construct a half module. Connection of two half modules with four new R-joints results in double-layer structures. Three different mobile assemblies are presented by changing link lengths: flat, cylindrical and spherical, according to the surface covered in the deployed form. Offsets allow designing U-joints with a finite thickness. Hence, more straightforward design solutions are possible with the modified version. Atarer et al. (2017) introduced five networks of modified Altmann linkages (Figure 2.5). Three networks are obtained by connecting adjacent loops with common links or joints. Two common links are used in these networks. The remaining two networks are obtained by connection with new links or joints method. Two extra R-joints connect loops, yielding a new loop between the connected linkages. The new loops are either an extra Altmann loop or an extra parallelogram loop. Song et al. (2016) connected five modified linkages to obtain a network. Altmann loops are connected by either two or four R-joints to have scaled extra Altmann loops. That results in four new small-sized modified Altmann loops.

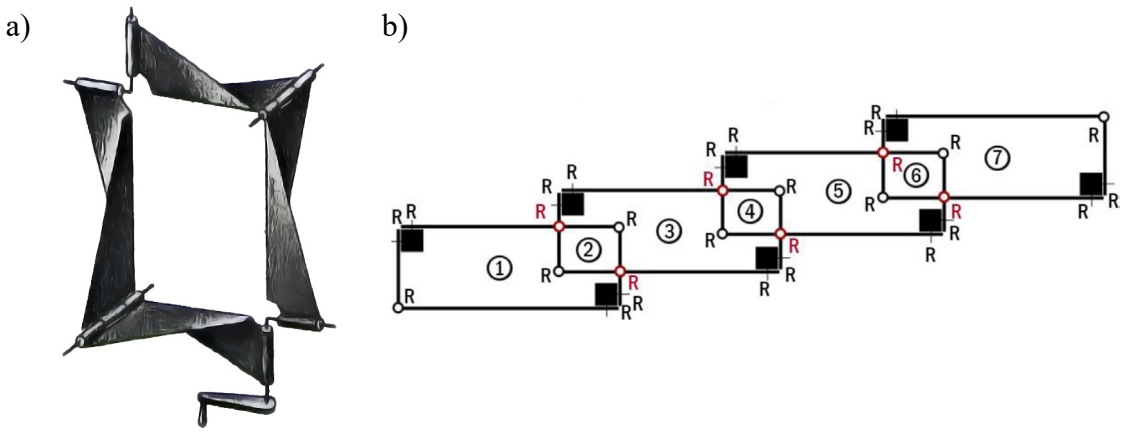


Figure 2.5. a) Altmann linkage (Source: modified from (Altmann, 1954)), b) a deployable network schematic based on the modified Altmann linkage (Source: Atarer et al., 2017).

The literature presents several networks of the single-dof and single-loop linkages (Baker and Hu, 1986; Calavatra, 1981; Hoberman, 2001; You and Chen, 2002; Chen, 2003; Melin, 2004; Chen and Baker, 2005; Chen and You, 2005, 2008a, 2008b; Isaak, 2006; Briand and You, 2007; Liu and Chen, 2009; Kiper and Söylemez, 2010; Tian and Chen, 2010; Qi et al., 2011; Cui et al., 2012; Huang et al., 2012; Guo and You, 2012; Bouten, 2015; Huang et al., 2016; Kiper, 2016; Song et al., 2016; Atarer et al., 2017; Yang et al., 2017, 2021; Zhang and Chen, 2018; Lu et al., 2019; Yang and You, 2020; Zhang et al., 2023). Symmetric Bricard linkages have great potential application for engineering due to their symmetry, as addressed by Zhang et al. (2023). Altmann linkage also has the potential to be a special case of the general line-symmetric Bricard linkage. However, the design of the original Altmann linkage is complicated. Kiper (2016), Song et al. (2016) and Atarer et al. (2017) solved these problems by adding new joint offsets to U-joints of the Altmann linkage.

Most common applications of such deployable networks are outer-space structures and architecture, such as deployable tents, shelters, billboard mechanisms. New networks of the modified Altmann linkage are still an open topic, and there needs to be a proper classification of the networks in the literature. Furthermore, new basic modules and their networks can be designed by adding new kinematic parameters to the Altmann linkage for simpler constructional design of some network types.

CHAPTER 3

DEPLOYABLE NETWORKS OF THE MODIFIED ALTMANN LINKAGES[2]

In this Chapter, networks of the two special cases of general line-symmetric Bricard linkage, i.e., two types of modified Altmann linkages are explored. One of the basic modules is used for deployable networks for the first time, whereas the other is the structure in Kiper (2016). Additional kinematic parameters to original Altmann linkage define the modules. The advantage of extra parameters is that they allow simpler constructional designs for deployable networks. Assembly modes and deployment of the modules are presented in the paper. The most significant contribution to the novel module is given by extra link lengths and joint offsets that allow a square cross-section for all links. Single-dof networks of the modules are investigated. Networks are classified with design alternatives of the possible versions to describe the effects of connection types and link dimensions on their deployed and folded forms. One of the module types is selected for each version, with a focus on favoring simpler constructional designs.

The Chapter is arranged as follows: in Section 3.1, general line-symmetric Bricard linkage is briefly explained, descriptions and assembly mode analyses of two special cases of general line-symmetric Bricard linkage (two modified Altmann linkages) are given. In Section 3.2, networks of the linkages are introduced and classified with feasible versions and design alternatives. Built prototypes of some networks are presented in Section 3.3.

3.1. Description of the Basic Modules

To present kinematic studies of the following linkages Denavit-Hartenberg (DH) convention shown in Figure 3.1a is followed. A_{ij} is the common normal from joint axis S_i to S_j for $j = i + 1$ (Denavit and Hartenberg 1954; McCarthy and Soh 2010, 256-57).

[2] The contents of this Chapter and Chapter 2 will be submitted to an SCIE journal. The journal draft is titled “Deployable Networks of Modified Altmann Linkages”.

The convention attaches the link frame B_i as the z-axis is directed along S_i and the x-axis is directed along A_{ij}. DH parameters d, θ, a and α represent joint offsets, joint angles, link lengths and twist angles, respectively. General line-symmetric Bricard 6R linkage is the most straightforward of Bricard linkages to analyze algebraically (Baker 1980). The colored link couples (orange-orange, yellow-yellow and blue-blue) are identical and symmetric about a central line L_1 (Figure 3.1b). Metric constraints of the linkage are as follows:

$$\begin{aligned} &d_1 = d_4,\ d_2 = d_5,\ d_3 = d_6 \\ &a_{12} = a_{45},\ a_{23} = a_{56},\ a_{34} = a_{61} \\ &\alpha_{12} = \alpha_{45},\ \alpha_{23} = \alpha_{56},\ \alpha_{34} = \alpha_{61} \end{aligned} \tag{3.1}$$

Baker (1980) derived the loop closure equations of the linkage. Due to line-symmetry constraints, opposite joint angles are equal and there are two more independent loop closure equations (Baker 1980). DH parameters of the linkage are related as follows:

$$d_i = d_{i+3},\ \theta_i = \theta_{i+3},\ a_{ij} = a_{(i+3)(j+3)},\ \ \alpha_{ij} = \alpha_{(i+3)(j+3)}\ \ (i = 1,2,3) \tag{3.2}$$

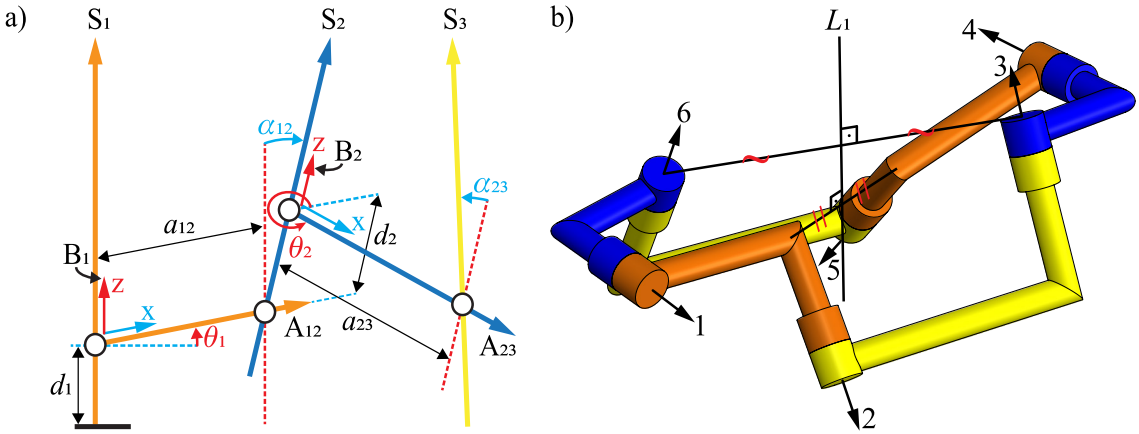


Figure 3.1. DH convention and general line-symmetric Bricard linkage a) convention redrawn from (McCarthy 2010, 256-57) b) linkage.

Mavroidis and Roth (1994) found a revised version of the general line-symmetric Bricard linkage with equal offsets and joint angles with opposite signs, i.e. $d_i = -d_{i+1}$ and $\theta_i = -\theta_{i+1}$. However, Song et al. (2014) showed that original and revised linkages are equivalent with different setups on S_i directions. Thus, the linkage has a single version with two assembly modes (Song et al. 2014). Both assembly modes are line-symmetric.

Altmann's 6R linkage (1954) is shown in Figure 3.2a. The linkage is a special case of the general line-symmetric Bricard linkage with zero offsets ($d_i = 0$), two zero link lengths ($a_{34} = a_{61} = 0$) and perpendicular twist angles ($\alpha_{34} = \alpha_{61} = \alpha_{23} = \alpha_{56} = \pi/2$, $\alpha_{12} = \alpha_{45} = -\pi/2$). Metric constraints of the linkage are given in Equation (3.3). Baker (1993) performed kinematic analysis of the linkage by using screw algebra. Motion of the linkage is illustrated in Figure 3.2b for $\theta_1 = -90°$, $-45°$ and $0°$. The linkage transforms from a planar rectangular form (deployed form) to a bundle form (folded form). In a typical construction, the links should be made of thin materials. When link thickness increases, links collusions might occur in the bundle form.

$$\begin{aligned} &d_i = 0 \ (i = 1, 2, \ \ldots, 6) \\ &a_{12} = a_{45} = a, \ a_{23} = a_{56} = b, \ a_{34} = a_{61} = 0 \\ &\alpha_{12} = \alpha_{45} = -\pi/2, \ \alpha_{23} = \alpha_{56} = \alpha_{34} = \alpha_{61} = \pi/2 \\ &\theta_i = \theta_{i+3} \ (i = 1, 2, 3) \end{aligned} \tag{3.3}$$

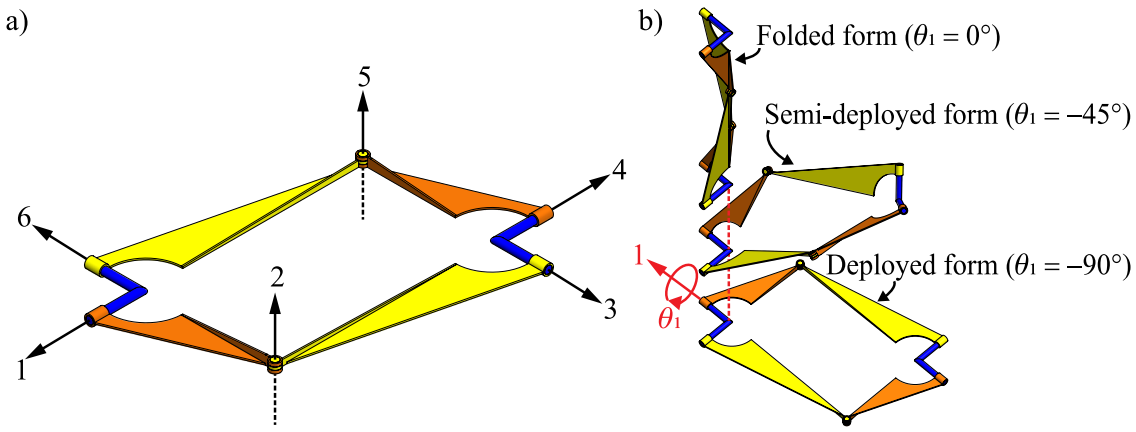


Figure 3.2. Altmann's 6R linkage a) joint numbering, b) deployment sequence.

Baker (2011) introduced a generalized version of the Altmann linkage by adding two extra link lengths ($a_{34} = a_{61} \neq 0$). The linkage can be named zero offsets case of general line-symmetric Bricard linkage. Baker (2012) showed that his generalized linkage is a special case of both general line-symmetric Bricard linkage and the second type of trihedral Bricard linkage and has two assembly modes (Wolhart 1993).

Altmann linkage offers great opportunities for deployable networks due to its symmetry in its rectangular and bundle forms. However, U-joints cause complicated constructional designs. Simplicity of the structure is critical for large deployable networks, because each component is used multiple times in the network and the design, manufacturing and test costs increase. The linkage meets the conditions of both general line-symmetric and trihedral linkages. The trihedral linkage requires no offsets ($d_i = 0$). This constraint limits the possibility of new solutions with only generalized Altmann linkage (Baker 2011) and variations with fewer parameters. Line-symmetry of the linkage is crucial in this manner. To overcome the U-joint design requirements of Altmann linkage, researchers (Kiper 2016; Atarer, Korkmaz and Kiper 2017; Song et al. 2016) added four non-zero offsets to the joints 1, 3, 4 and 6 such that $d_1 = d_4 = -d_3 = -d_6 = d$. Offsets maintain line symmetry, but lead to the loss of trihedral linkage properties, which is unnecessary for a deployable structure. This linkage is named Special case-I: zero link length hubs. We developed another linkage by adding the remaining two offsets (d_2, d_5) and link lengths (a_{34}, a_{61}) to have simpler constructional designs named Special case-II: non-zero link length hubs. Both linkages and their assembly modes are analyzed in the following subsections.

3.1.1. Special Case-I: Zero Link Length Hubs

Special case-I with zero link length hubs is shown in Figure 3.3a. This case is obtained by adding four non-zero equal offsets with two pairs of opposite signs ($d_1 = d_4 = -d_3 = -d_6 = d$). This allows the design of square cross-section hubs between bars (Atarer, Korkmaz and Kiper 2017). Orange and yellow links represent a bar, whereas the blue link is a hub. A kinematic diagram with attached coordinate frames is shown in Figure 3.3b. In a more general form, the linkage can be defined by taking $d_1 = d_4 \neq -d_3 = -d_6$. Different offset values represent hubs with rectangular cross-

sections, which is not preferrable in practice. Metric constraints of the linkage are given in Equation (3.4). The linkage is in accordance with the line-symmetry constraints defined in Equation (3.2).

$$
\begin{aligned}
&d_1 = d_4 = d,\ d_3 = d_6 = -d,\ d_2 = d_5 = 0 \\
&a_{12} = a_{45} = a,\ a_{23} = a_{56} = b,\ a_{34} = a_{61} = 0 \\
&\alpha_{12} = \alpha_{45} = -\pi/2,\ \alpha_{23} = \alpha_{56} = \alpha_{34} = \alpha_{61} = \pi/2 \\
&\theta_i = \theta_{i+3}\ \ (i = 1,2,3)
\end{aligned}
\tag{3.4}
$$

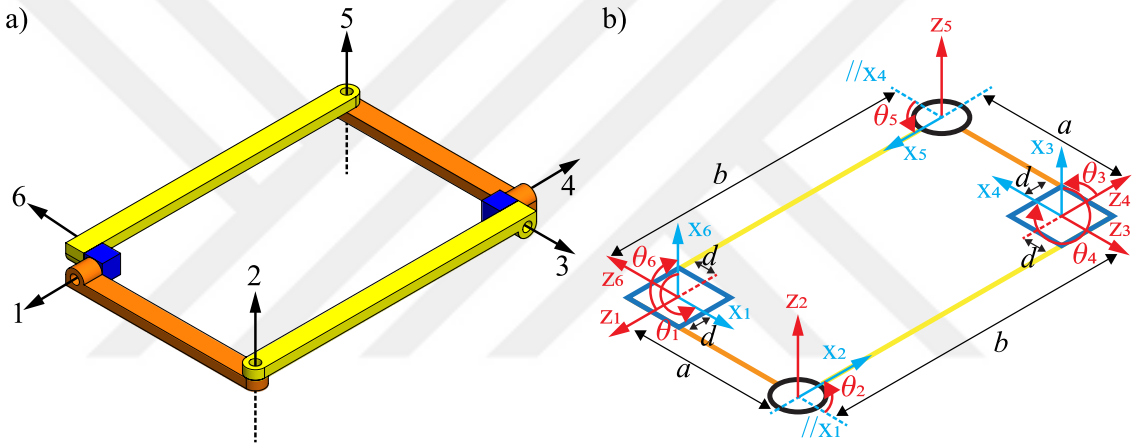


Figure 3.3. Special case-I: zero link length hubs a) CAD model, b) schematic.

If θ_1 is the input, two solutions for θ_2 and θ_3 can be found by implementing Equation (3.4) into the solution of Song et al. (2014):

$$
\begin{aligned}
\theta_2 &= 2\tan^{-1}\left(\frac{d\sin\theta_1 - a + \sigma\sqrt{\left(-a^2+b^2+d^2\right)\sin^2\theta_1 + 2bd\sin\theta_1 + a^2}}{(a-b)\sin\theta_1 - 2d}\right) \\
\theta_3 &= 2\tan^{-1}\left(\frac{-a\cos\theta_1 - \sigma\sqrt{\left(-a^2+b^2+d^2\right)\mathrm{s}^2\theta_1 + 2bd\sin\theta_1 + a^2}}{b\sin\theta_1 - d\cos\theta_1 + d}\right)
\end{aligned}
\tag{3.5}
$$

Sine and cosine functions are abbreviated as s and c, respectively, in Equation (3.5). $\sigma = \pm 1$ defines two assembly modes of the linkage. Note that there are typos in Eq. (24) of Song et al. (2014) (D_3, F_3 and G_3 terms), which are corrected for the solution. When link lengths are equal ($a = b$), Equation (3.5) becomes:

$$\begin{aligned} \theta_2 &= 2\tan^{-1}\left(\frac{d\sin\theta_1 - a + \sigma\left(a + d\sin\theta_1\right)}{-2d}\right) \\ \theta_3 &= 2\tan^{-1}\left(\frac{-a\cos\theta_1 - \sigma\left(a + d\sin\theta_1\right)}{a\sin\theta_1 - d\cos\theta_1 + d}\right) \end{aligned} \tag{3.6}$$

If Equation (3.6) specified for assembly modes and tangent half-angle substitution is applied for θ_3:

$$\begin{aligned} &\sigma = 1 \Rightarrow \theta_2^{\mathrm{I}} = 2\tan^{-1}\left(-\sin\theta_1\right),\ \theta_3^{\mathrm{I}} = 2\tan^{-1}\left(-\cot\frac{\theta_1}{2}\right) = \theta_1 + \pi \\ &\sigma = -1 \Rightarrow \theta_2^{\mathrm{II}} = 2\tan^{-1}\left(\frac{a}{d}\right),\ \theta_3^{\mathrm{II}} = 2\tan^{-1}\left(\frac{a\tan\frac{\theta_1}{2} + d}{d\tan\frac{\theta_1}{2} + a}\right) \end{aligned} \tag{3.7}$$

In an equilateral case, θ_2 does not depend on link lengths and $\theta_3 = \theta_1 + \pi$ for $\sigma = 1$. However, the mechanism becomes a spherical 4R linkage in the other mode ($\sigma = -1$) with joint axes 1, 3, 4 and 6. θ_2 and θ_5 become constant. Furthermore, if a and d are selected equal, then θ_3 and θ_6 also become constants. A reconfigurable linkage is obtained in these circumstances.

Deployed ($\theta_1 = -90°$), semi-deployed ($\theta_1 = -45°$) and folded forms ($\theta_1 = -0°$) of a zero link length hubs case are shown in Figure 3.4a for $\sigma = 1$. Links are not collinear anymore, but parallel in the folded form due to additional kinematic parameters. Open ($\sigma = 1$) and cross ($\sigma = -1$) assembly modes are presented in Figure 3.4b. While the two links opposite to the input joint moves downwards during deployment, as in Figure 3.4a, they move upwards in the cross mode. Cross mode can offer strengthened design

solutions in the deployed form due to the difference between parallelogram and antiparallelogram structures. The linkage allows the design of a deployable loop without collision to be simpler than the original linkage. Moreover, link shapes are not necessarily to be sheet material.

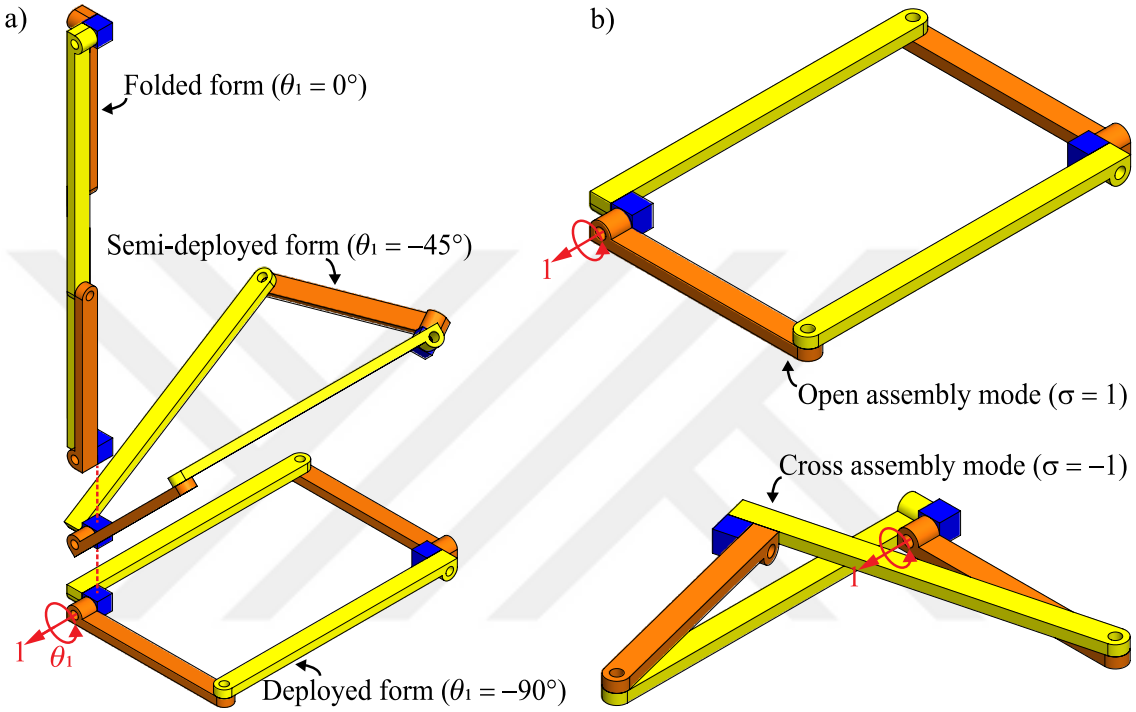


Figure 3.4. Details of a Special case-I: zero link length hubs loop a) deployment, b) assembly modes.

3.1.2. Special Case-II: Non-Zero Link Length Hubs

Special case-I allows the design hubs (blue links) with uniform square cross-sections; however, yellow and orange links cannot have that form. To obtain a linkage with all square cross-sectioned links, a link length is added to the hubs ($a_{34} = a_{61} = c$), and that causes an equal offset in the yellow links ($d_2 = d_5 = -c$). New link and joint parameters are consistent with the line-symmetry constraints in Equation (3.2). Special case-II: non-zero link length hubs and schematic with attached reference coordinate frames are shown in Figure 3.5. d_1 and d_6 are selected as negatively equaled offsets as

Special case-I. This linkage is the most generic modified Altmann linkage, with all link lengths and offsets defined in a non-zero domain by preserving line-symmetry property. Metric constraints of the linkage are given in Equation (3.8).

$$\begin{aligned} &d_1 = d_4 = d,\ d_3 = d_6 = -d,\ d_2 = d_5 = -c \\ &a_{12} = a_{45} = a,\ a_{23} = a_{56} = b,\ a_{34} = a_{61} = c \\ &\alpha_{12} = \alpha_{45} = -\pi/2,\ \alpha_{23} = \alpha_{56} = \alpha_{34} = \alpha_{61} = \pi/2 \\ &\theta_i = \theta_{i+3}\ \ (i = 1,2,3) \end{aligned} \tag{3.8}$$

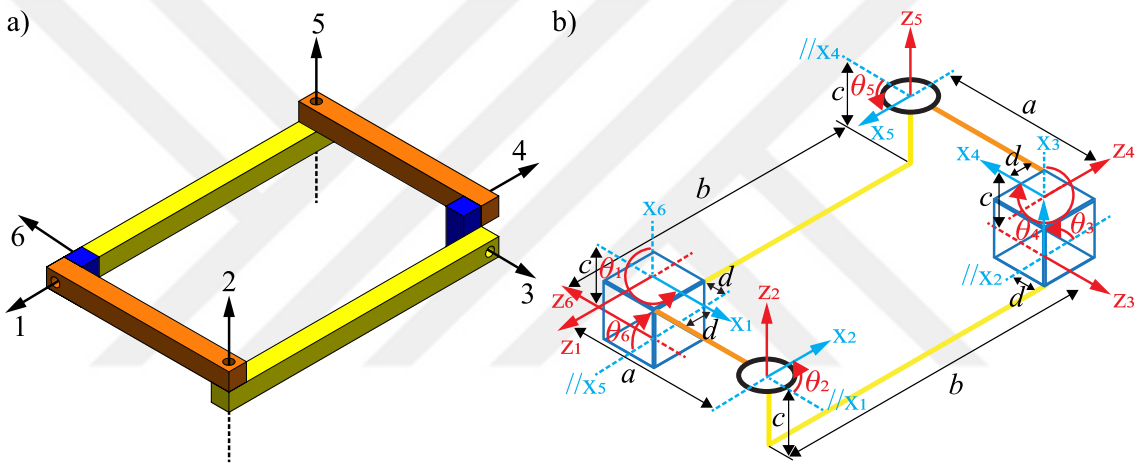


Figure 3.5. Special case-II: non-zero link length hubs a) CAD model, b) schematic.

Similarly, with Special case-I, θ_2 and θ_3 solved for input θ_1 by using solution of Song et al. (2014). Equation (21)-(22) constants are calculated as follows:

$$\begin{cases} A_2 = (a-b)\sin\theta_1 - c\cos\theta_1 - 2d,\ B_2 = 2(-d\sin\theta_1 + c\cos\theta_1 + a), \\ C_2 = (a+b)\sin\theta_1 - c\cos\theta_1, A_3 = (b-c)\sin\theta_1 - d\cos\theta_1 - c + d, \\ B_3 = 2(c\sin\theta_1 + a\cos\theta_1 + c),\ C_3 = (-b-c)\sin\theta_1 - d\cos\theta_1 - c - d \end{cases} \tag{3.9}$$

If we substitute Equation (3.9) into Equation (21)-(22), the solution can simplifed as follows :

$$\theta_2 = 2\tan^{-1}\left(\frac{d\,\mathrm{s}\theta_1 - c\,\mathrm{c}\theta_1 - a + \sigma\sqrt{\left(-a^2+b^2+d^2\right)\mathrm{s}^2\theta_1 + 2bd\,\mathrm{s}\theta_1 + 2c\,\mathrm{c}\theta_1(a-d)(\mathrm{s}\theta_1+1) + a^2}}{(a-b)\mathrm{s}\theta_1 - c\,\mathrm{c}\theta_1 - 2d}\right)$$

$$\theta_3 = 2\tan^{-1}\left(\frac{-c\,\mathrm{s}\theta_1 - a\,\mathrm{c}\theta_1 - c - \sigma\sqrt{\left(-a^2+b^2+d^2\right)\mathrm{s}^2\theta_1 + 2bd\,\mathrm{s}\theta_1 + 2c\,\mathrm{c}\theta_1(a-d)(\mathrm{s}\theta_1+1) + a^2}}{(b-c)\mathrm{s}\theta_1 - d\,\mathrm{c}\theta_1 - c + d}\right)$$

(3.10)

For equilateral case ($a = b$), Equation (3.10) becomes:

$$\theta_2 = 2\tan^{-1}\left(\frac{d\,\mathrm{s}\theta_1 - c\,\mathrm{c}\theta_1 - a + \sigma\sqrt{(a+d\,\mathrm{s}\theta_1)^2 + 2c\,\mathrm{c}\theta_1(a-d)(\mathrm{s}\theta_1+1)}}{-c\,\mathrm{c}\theta_1 - 2d}\right)$$

$$\theta_3 = 2\tan^{-1}\left(\frac{-c\,\mathrm{s}\theta_1 - a\,\mathrm{c}\theta_1 - c - \sigma\sqrt{(a+d\,\mathrm{s}\theta_1)^2 + 2c\,\mathrm{c}\theta_1(a-d)(\mathrm{s}\theta_1+1)}}{(a-c)\mathrm{s}\theta_1 - d\,\mathrm{c}\theta_1 - c + d}\right) \quad (3.11)$$

Since further simplification does not provide any specialty of the linkage, Equation (3.11) is left in the given form in Equation (3.11).

Deployment of a non-zero link length hubs case is shown in Figure 3.6a with deployed ($\theta_1 = -90°$), semi-deployed ($\theta_1 = -45°$) and folded forms ($\theta_1 = 0°$). All links can designed with square cross-sectioned bars. The linkage only provides line-symmetry property, i.e., links are not parallel in the folded form. The linkage has two assembly modes: open ($\sigma = 1$) and cross modes ($\sigma = -1$), as shown in Figure 3.6b. Cross mode requires a different geometry for yellow links to prevent collision between them.

To achieve parallel links in the folded form as Special case-I, θ_3 should be equal 180° when $\theta_1 = 0°$. We can show that the linkage exhibits parallel links in the folded form if and only if c = 0 by implementing these joint angles into the θ_3 solution in Equation (3.10) as given below. This condition corresponds to Special Case-I, so it is not possible. Parallel form is valuable for applications where 90° rotation between the deployed and folded is desired.

$$\tan\frac{\pi}{2} = \frac{-a-c-\sigma\sqrt{2c(a-d)+a^2}}{-c} \tag{3.12}$$

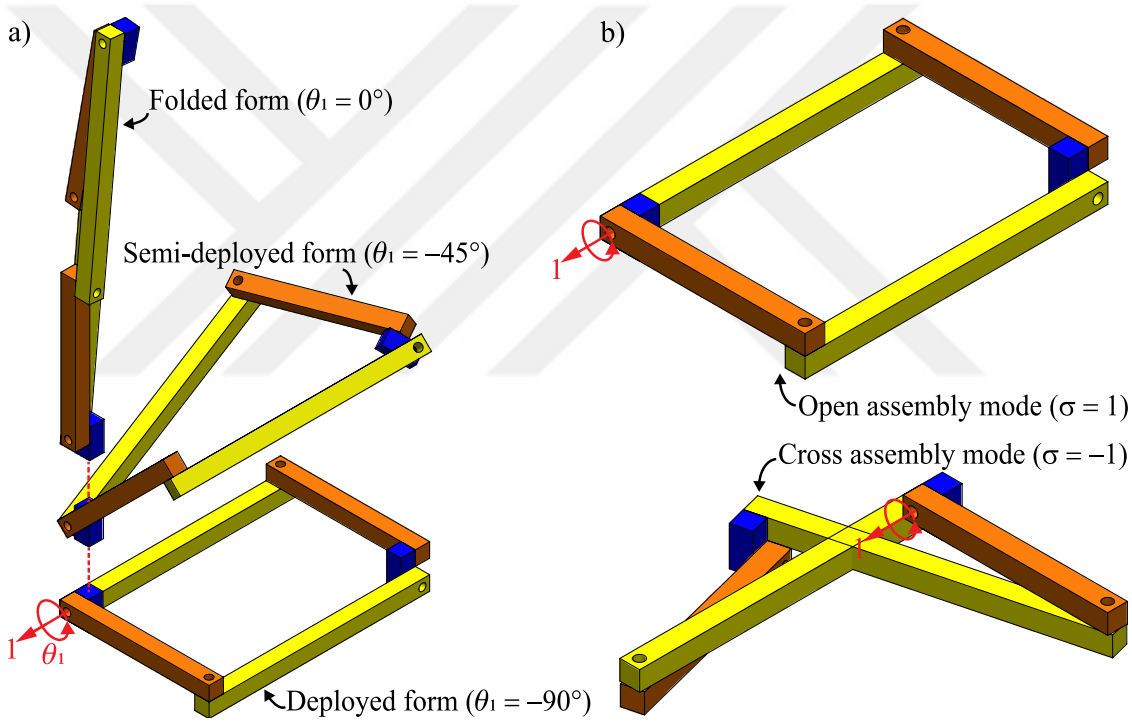


Figure 3.6. Details of a Special case-II: zero link length hubs loop a) deployment, b) assembly modes.

3.2. Deployable Networks

Identical or scaled copies of the selected single-loop modules are connected to have deployable networks as in (Kiper 2016, Atarer, Korkmaz and Kiper 2017, Song et

al. 2016). Two alternative methods of connections are possible. Method-I connects adjacent loops with common links or joints, and Method-II connects the modules with new links or joints without any common connections. The aim is to generate single dof and multi-loop networks. This section examines the networks of the zero and the non-zero link length hub modules (Special cases-I and II) and design alternatives of the possible versions with CAD models. Special case-I module is advantageous in terms of parallel link configuration in the folded form and it is more compact in the deployed form. On the other hand, Special case-II module benefits from link shapes because it allows a uniform square cross-section for all links. A proper type of module is selected for each version of the presented networks.

The following multi-loop mechanism mobility formula of Alizade (2010) is employed to ensure that the presented network versions are single-dof linkages:

$$M = \sum_{i=1}^{j} f_i - \sum_{k=1}^{L} \lambda_k + q \tag{3.13}$$

Here, j is number of joints in the mechanism, f_i is dof of the i^{th} joint, L is the number of independent loops, λ_k is the dof of space in which loop k operates ($\lambda = 2, \ldots, 6$) and q is the number of excessive constraints/elements (links, joints or loops). Since there are five independent loop closure equations for both modules presented in Section 3.1 (also valid for original Altmann and general line-symmetric Bricard linkage), loops live in a $\lambda = 5$ space.

Method-I reduces the number of joints (j) and links (l) in the network according to the number of common links (l_C), and joints (j_C) used to connect adjacent loops. If the common links are directly connected, there will be a common joint between these links. On the other hand, Method-II network modules are connected by a number of new joints (j_N) between links of adjacent modules, hence resulting in new loops between the connected modules.

The modules can be connected either iteratively, where the each module is connected to a new module or recursively to end up with a loop of finitely many modules. If loops are connected recursively, that will lead to loop of loops. Between the first and last modules, there can be an extra number of common or new termination links (l_{CT} or

l_{NT}) and joints (j_{CT} or j_{NT}). Termination parameters (l_{CT}, l_{NT}, j_{CT} and j_{NT}) should be considered as part of the terminal modules.

j and l can be generalized in the following equations for networks of a single type of basic modules with the same kind of connection in between. A basic module is defined with number of joints ($j_M = 6$) and links ($l_M = 6$). Number of links, l, in a network of n loops:

$$l = l_M n + (l_N - l_C)(n-1) + l_{NT} - l_{CT} \tag{3.14}$$

Number of joints, j, in a network of n loops:

$$j = j_M n + (j_N - j_C)(n-1) + j_{NT} - j_{CT} \tag{3.15}$$

L can obtained using Hochman's (1890) relationship for closed-loop mechanisms based on Euler's polyhedral formula (1756):

$$L = j - l + 1 \tag{3.16}$$

Table 3.1 lists four types of networks and design alternatives of nine versions. Networks obtained using Method-I are named per the number and names of common link(s). Since there are two types of binary links in an Altmann loop as bars and hubs, two different types of networks are possible by using Method-I depending on if only bars are common links (Type-I: two common bars) or not (Type-II: one common bar and one common hub). A two common hub version is not possible because the two hubs of a module are at opposite corners. The names of the networks using Method-II are defined by maximum number of new joints used to connect adjacent loops of the network. New loops are constructed with either two new joints (Type-III) or four new joints Type-IV.

Special case-II becomes prominent when two bars of the adjacent loops are connected with common or new R-joint(s). In addition, it is possible to design each

alternative with Special case-I modules. However, that is not valid for Special case-II. When there are two different modified Altmann loops of different link-length ratio, Special case-II modules cannot satisfy similarity conditions between the loops.

Table 3.1. Network types (*: novel versions).

Type	Method	Version	Module Type	Design Alternatives
I) Two common bars	I	I) Diagonally multiplied version	II	I) Short link to short link II) Long link to short link III) Short link to short link with a kink angle IV) Long link to short link with a kink angle
II) One common bar and one common hub	I	I) Linearly multiplied version	I	I) General case
		II) Diagonally multiplied version*	I	I) General case
		III) 4-loop version	I	I) General case
III) Two new joints	II	I) Extra modified Altmann loop version	I	I) Equilateral case
		II) Extra parallelogram loop version	II	I) General case
		III) 4-loop version*	I	I) Equilateral case
IV) Four new joints	II	I) Extra modified Altmann loop version	I	I) Equilateral case
		II) 5-loop version	I	I) Equilateral case

3.2.1. Type-I: Two Common Bars — Version-I: Diagonally Multiplied Version

Type-I Version-I is the only possible version of the two common bars case. The version is introduced in (Atarer, Korkmaz and Kiper 2017) (see Figure 3.7a). It is derived with the combination of two modules where two adjacent bars are common links ($l_C = 2$),

and there is a common joint ($j_C = 1$) between them. The networks are obtained by Method-I ($j_N = j_{NT} = l_{NT} = 0$) and iteratively multiplied ($j_{CT} = l_{CT} = 0$) in the diagonal direction. Special case-II module with non-zero link length hubs is selected for this network since two bars of the adjacent modules are directly connected and there is a common joint in between. Mobility of an n-loop Version-I network is calculated using Equations (3.13)-(3.15) in the following equations. There are no excessive elements in the network ($q = 0$). Number of links (l), joints (j) and independent loops (L) in a network of n modules:

$$l = 6n + (0-2)(n-1) + 0 - 0 = 4n + 2 \tag{3.17}$$

$$j = 6n + (0-1)(n-1) + 0 - 0 = 5n + 1 \tag{3.18}$$

$$L = 5n + 1 - 4n - 2 + 1 = n \tag{3.19}$$

$$M = \sum_{i=1}^{j} f_i - \sum_{k=1}^{L} \lambda_k + q = 5n + 1 - 5n + 0 = 1 \tag{3.20}$$

So, Type-I Version-I networks are single-dof linkages for any number of modules (n).

The network is classified into four design alternatives depending on the connection of I) short link to short link, II) long link to short link, and whether there is a kink angle between the bars (III-IV) for the general case ($a < b$) or a module. Design alternative-I, depicted in , can be named according to type of connected links as short link to short link or long link to short link. To distinguish special cases in the network schematics, dashed lines are used to represent Special case-II module links on top of each other situation. Green and brown links represent common bars where the common joint is given as light blue in the schematic in Figure 3.7a, and same color code is used for the CAD model in Figure 3.7b. Modules are multiplied with half turns about the axis of the common joint in the deployed form (Figure 3.7b). The network can form an arch-like shape in the semi-deployed form. In different assembly modes, the semi-deployed form looks like a zig-zag shape. Fixed hub of the network is aligned with a red dashed line for different poses in Figure 3.7b. Same color code and illustration style are valid for the rest of the networks presented in the paper.

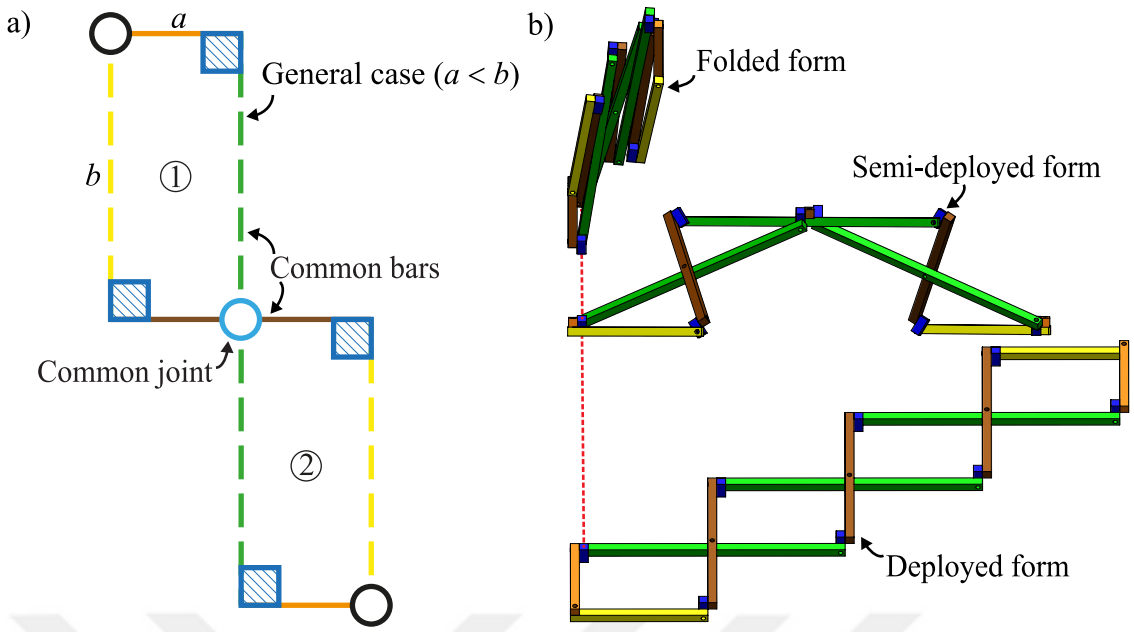


Figure 3.7. Details of Design alternative-I: short link to short link a) schematic, b) deployment.

Design alternative-II modules are multiplied by connection of long link to short link (see (see Figure 3.8a). The deployment of a 4-module network is illustrated in Figure 3.8b. This time, modules are folded along their diagonal which coincide with the adjacent loop's diagonal. Adjacent folds may be directed in the same or opposite directions according to the chosen assembly mode. The resulting transformation between the folded and deployed forms is unidirectional deployment along a straight line.

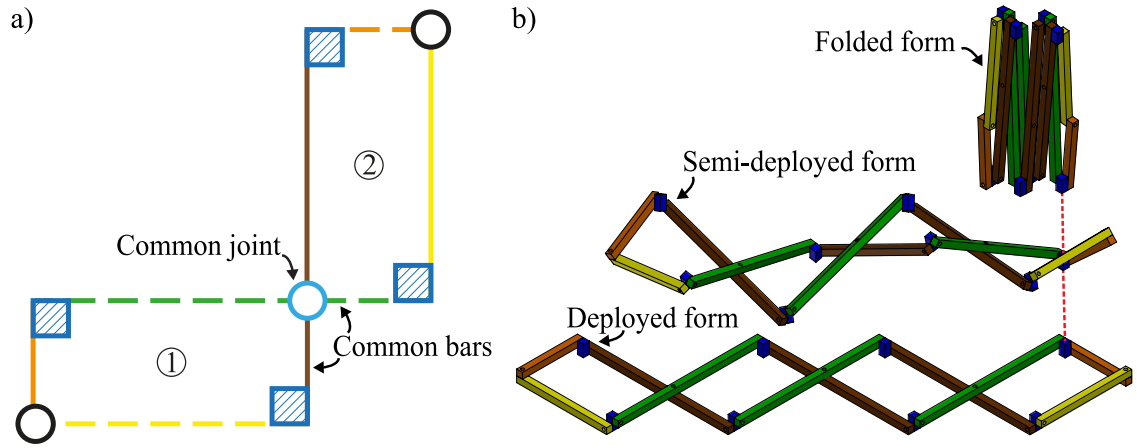


Figure 3.8. Details of Design alternative-II: long link to short link a) schematic, b) deployment.

Design alternative-III is a variation of Design alternative-I (short link to short link) with the addition of a kink angle between common bars (Figure 3.9a). Deployment of a 4-loop CAD model is shown in Figure 3.9b. In a semi-deployed state, the assembly forms a helical-like shape. The folded form may not be much compact due to link collisions. Design parameters such as link lengths, widths, heights, and kink angles can be optimized to increase motion capability.

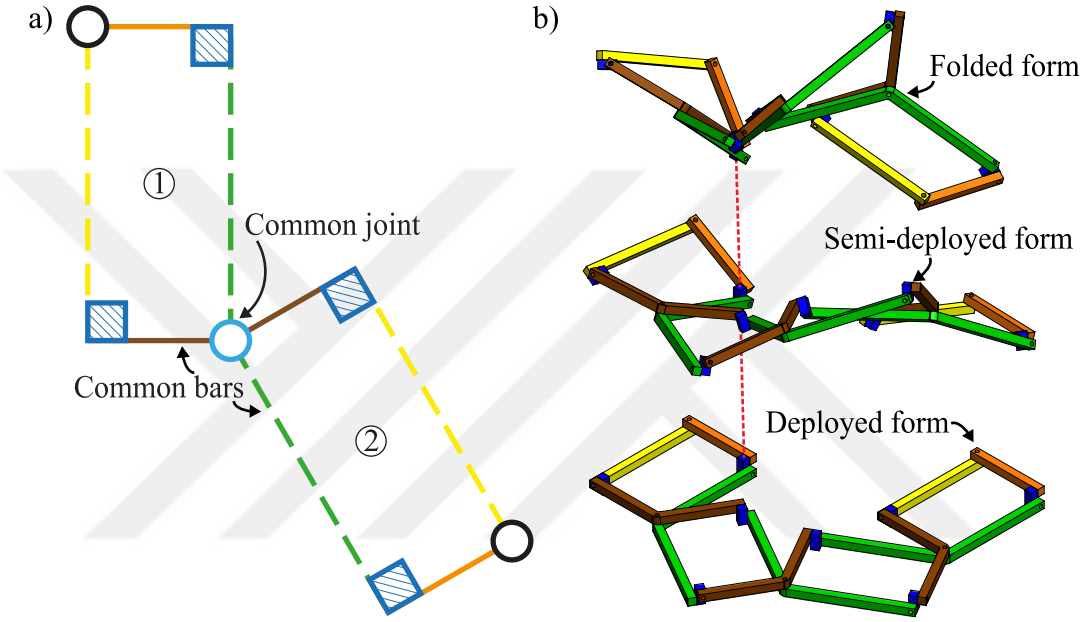


Figure 3.9. Details of Design alternative-III: short link to short link with a kink angle a) schematic, b) deployment.

If kink angles of the angulated elements are as in Figure 3.10a, a variation of Design alternative-III is obtained. Deployment of the network is shown in Figure 3.10b. Motion capability is increased with this variation. A similar helical-like shape features a more intricate structure with protrusions and indentations.

Design alternative-IV is a variation of Design alternative-II, obtained by connection of long link to short link with a kink angle (Figure 3.11a). Deployment is shown in Figure 3.11b. An arch-like shape is defined as in Design alternative-I, that is much different from Design alternative-II. Even though all networks are of Type-I, folded forms and motion during deployment can vary between design alternatives. This highlights the importance of investigating different networks of linkages

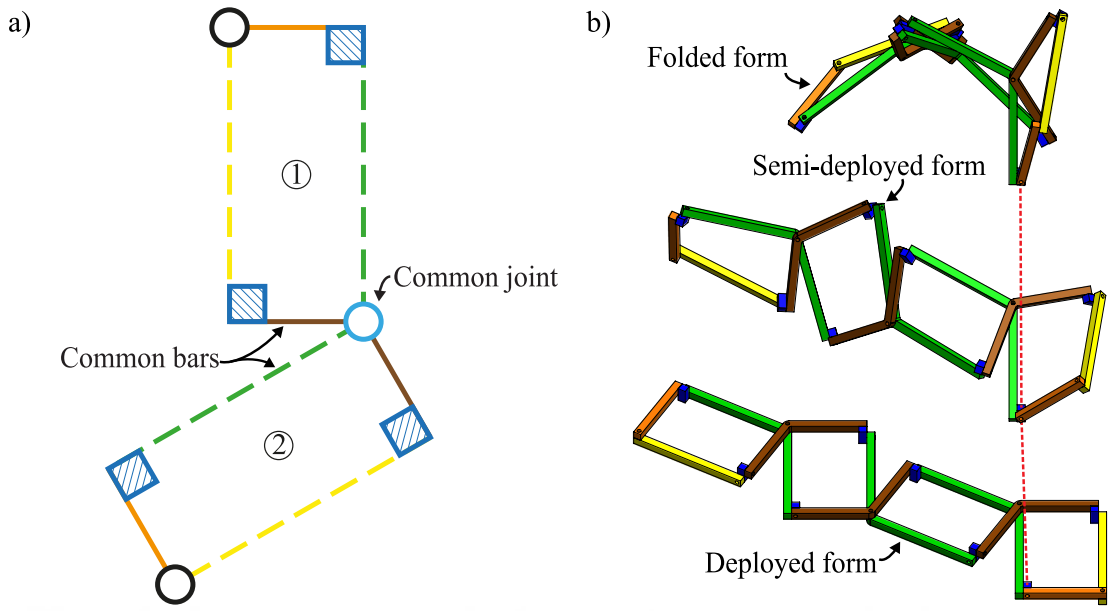


Figure 3.10. Details of Design alternative-III: mirror-angulated variation a) schematic, b) deployment.

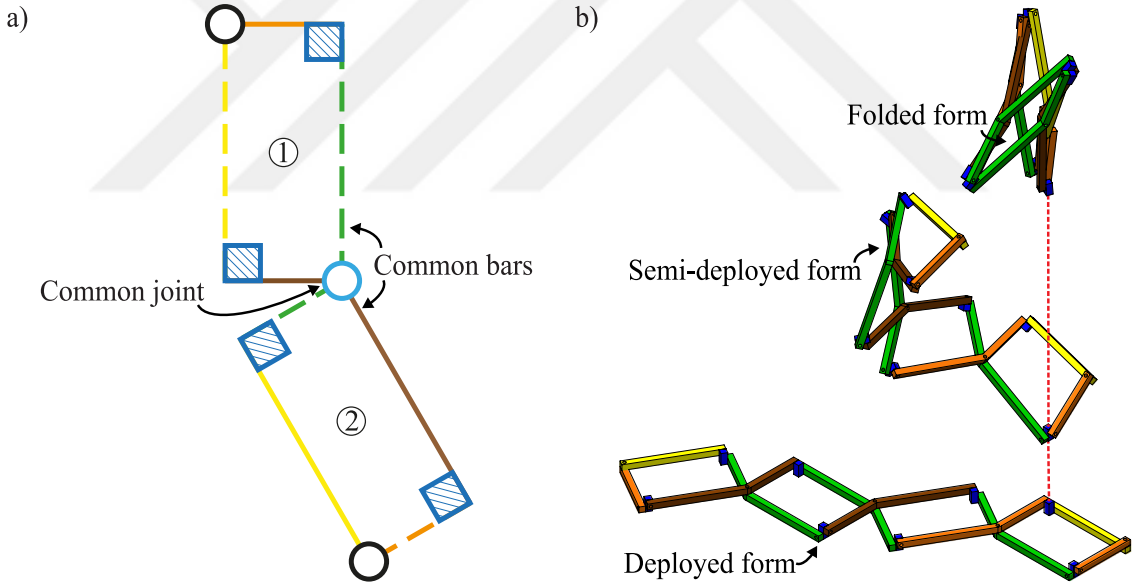


Figure 3.11. Details of Design alternative-IV: long link to short link with a kink angle a) schematic b) deployment.

3.2.2. Type-II: One Common Bar and One Common Hub

The type is obtained when a bar and a hub are two common links in two adjacent modules, e.g., Figure 3.12a. Three versions of Type-II networks are presented: I) linearly

multiplied version, II) diagonally multiplied version and III) 4-loop version. Version-II is a novel structure, whereas Versions-I and III are presented in (Atarer, Korkmaz and Kiper 2017). Double-layer networks of Version-III are presented in Kiper (2016). Special case-1 modules are used to construct Type-I networks since there is no direct connection between two bars in this type. There are two common links among adjacent modules, so $l_C = 2$, and there is a common joint, so $j_C = 1$.

3.2.2.1. Version-I: Linearly Multiplied Version

A general case ($a < b$) of Version-I is presented in Figure 3.12a. Loops 1 and 2 are connected through a common short (brown) link and hub (green) and a common joint between them. Modules are multiplied iteratively ($j_{CT} = l_{CT} = 0$) and linearly (vertical direction in Figure 3.12a). Since all parameters are the same with $l_C = 2$ and $j_C = 1$, Equations (3.13)-(3.15) parameters are the same as Type-I Version-I, Equations (3.17)-(3.20) are valid for this version and the assembly has single-dof for any number of modules. A CAD model of the network with four loops is built, and its deployment is shown in Figure 3.12b. The offset is increased to leave space to connect common hubs of the adjacent loops, allowing the hub design presented in the CAD model. This version can attain an arch-like shape during deployment in a properly chosen assembly mode. Note that two blue hubs can connect directly with a common shaft, and there is no need to increase d in that case (see Version-II). However, assembly of the hubs is complicated and may require high-precision jig/fixture designs. Depending on the application, a designer can choose that, but here we present the network with a simpler constructional design.

3.2.2.2. Version-II: Diagonally Multiplied Version

Version-II shown in Figure 3.13a, is a novel structure for the general case ($a < b$). This time loops are multiplied iteratively ($j_{CT} = l_{CT} = 0$) along the diagonal, passing through the hubs such that the adjacent loops share a hub (green), a bar (brown) and a joint. Equations (3.17)-(3.20) are valid for this version also; thus the assembly has 1-dof. Deployment of the version is illustrated in Figure 3.13b for $n = 4$. Designing a hub as the

offset *d* increases is difficult because link collisions may occur. Thus, two hubs are directly connected, as in Figure 3.13b. The modules twist about a diagonal during the deployment as in Type-I Version-I Design alternative-II. The consequent motion is like an arm rotating from a horizontal plane to a bundle in vertical direction in a proper assembly mode. Design alternatives of Type-II Versions-I and II are presented for connection of short link to short link and long link to long link. It is possible to derive short link to long link and kink angle versions like Type-I Version-I. Only one alternative of these versions is presented to keep the paper concise.

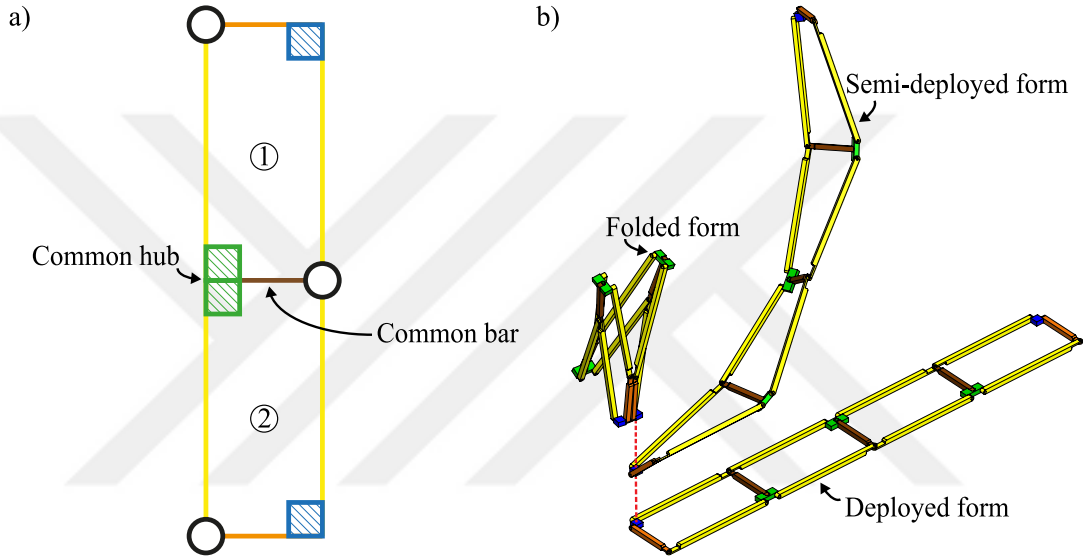


Figure 3.12. Details of Version-I: linearly multiplied version a) schematic, b) deployment.

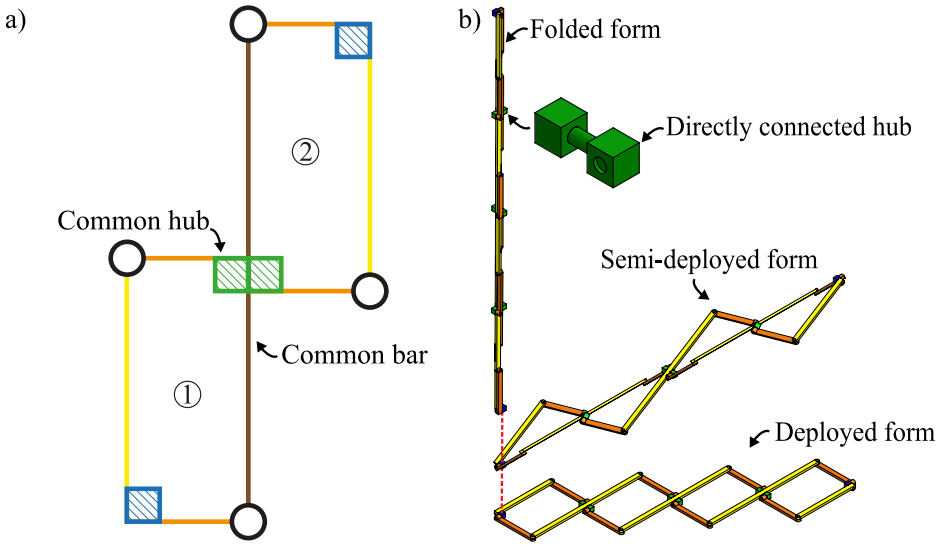


Figure 3.13. Details of Version-II: diagonally multiplied version a) schematic, b) deployment.

3.2.2.3. Version-III: 4-loop Version

Figure 3.14a illustrates an assembly where four loops are connected as in Version-I, but recursively. We obtain a loop of four loops for the general case ($a < b$). Adjacent loops share a bar, a hub and a joint between them. The last loop is connected by a common termination joint ($j_{CT} = 1$) and a link ($l_{CT} = 1$). All four modules share a common hub (green). We can evaluate the mobility of the 4-loop version using Equations (3.13)-(3.16) . Due to the loop of loops, there is one excessive constraint ($q = 1$). This excessive constraint can be demonstrated, for instance, by replacing one of the revolute joints with a cylindrical joint, where the translational dof of the cylindrical joint would be passive. The number of links (l), the number of joints (j) and the number of independent loops (L) for $n = 4$ modules are as follows:

$$l = 6\times 4 + (0-2)(4-1) + 0 - 1 = 17 \tag{3.21}$$

$$j = 6\times 4 + (0-1)(4-1) + 0 - 1 = 20 \tag{3.22}$$

$$L = 20 - 17 + 1 = 4 \tag{3.23}$$

The mobility of the network calculated as:

$$M = \sum_{i=1}^{j} f_i - \sum_{k=1}^{L} \lambda_k + q = 20 - 5\cdot 4 + 1 = 1 \tag{3.24}$$

With a common hub, the network can be multiplied with new 3-loops such as loops 5, 6, and 7 in Figure 3.14a. As many additional 3-loops as desired can be added, but the deployability becomes more limited as new loops are added. One can compute mobility with the addition of the new loops, resulting in a single-dof linkage. Deployment of the 4-loop assembly is shown in Figure 3.14b when the common hub is fixed. A

symmetry line passes through the center of the common hub (the red alignment line in Figure 3.14b).

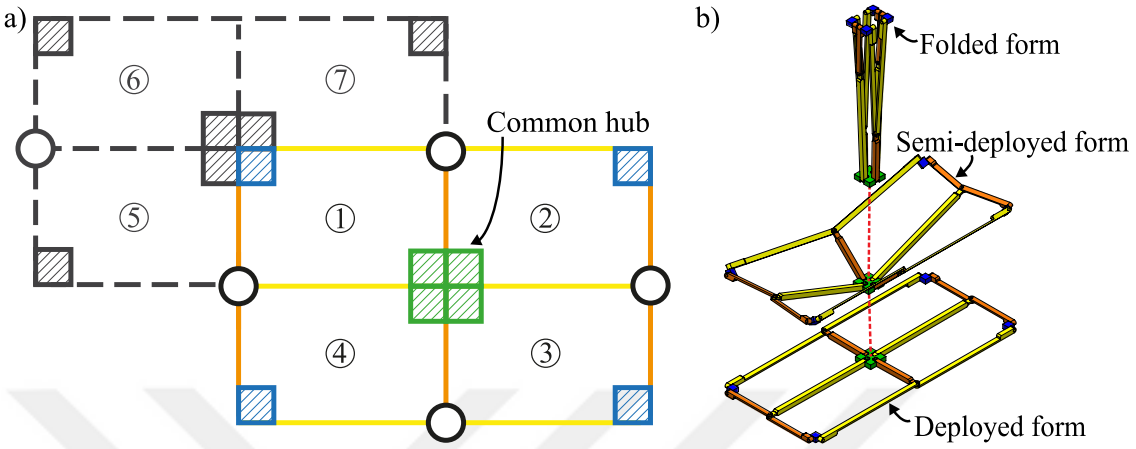


Figure 3.14. Details of Version-III: 4-loop version a) schematic, b) deployment.

Double-layer networks composed of two mirror-symmetric Version-III networks connected by new joints and loops method are presented in Kiper (2016). The invention proposes three alternatives where deployed assemblies can form flat, cylindrical or spherical shapes. Different hub and other component design alternatives are presented in Kiper (2016). The invention is essential to highlight a combination of Methods-I and II. We present single-layer networks obtained from Method-I or Method-II in this paper, but combining the methods for multi-layer applications will result in several new assemblies.

3.2.3. Type-III: Two New Joints

Type-III is a network obtained with Method-II where adjacent loops are connected with two new joints ($j_N = 2$) on their bars, and a new loop is obtained between the modules (see Figure 3.15a). Two possibilities of the new loop structure arise: I) Altmann loop version and II) parallelogram loop version (Atarer, Korkmaz and Kiper 2017). Version-III is novel and derived as a combination of Versions-I and II with four loops connected recursively. Special case-I modules are selected to be used for Type-III networks. Since there are new joints between links of adjacent loops, it is better to position

the links on top of each other, so non-zero link lengths case (II) is suitable. Suppose that there is a scaled version of the basic module as a new loop between connected modules. In that case, Special case-II cannot satisfy the similarity requirements of the loops due to the extra link length c. Thus, Versions-I and III are built with Special case-I, while Version-II is built with Special case-II. In networks obtained with Method-II, there are no common links ($l_C = l_{CT} = 0$) and common joints ($j_C = j_{CT} = 0$) between adjacent loops. Also, these networks are connected without new links ($l_N = l_{NT} = 0$). The number of links, l, in a network of n loops can be calculated using Equation (3.14) as

$$l = 6n + (0-0)(n-1) + 0 - 0 = 6n \tag{3.25}$$

3.2.3.1. Version-I: Extra Modified Altmann Loop Version

When two Special case-I loops are iteratively multiplied ($j_{NT} = 0$) with two new joints (outlined with red in Figure 3.15a) by connecting links of adjacent modules in the vicinity of hub corners, a new modified Altmann loop is obtained. The network is only mobile in the case where links are equilateral ($a = b$). Because, in case of different link lengths ($a \neq b$), the newly generated loop cannot follow the similarity conditions of the original two loops and leads to an immobile structure. The number of joints j, and the number of independent loops, L, in a network of n loops according to Equations (3.15)-(3.16) are

$$j = 6n + (2-0)(n-1) + 0 - 0 = 8n - 2 \tag{3.26}$$

$$L = 8n - 2 - 6n + 1 = 2n - 1 \tag{3.27}$$

There are two excessive constraints in each connection ($q = 2(n - 1)$. So, mobility of the network calculated using Equation (3.13) as follows:

$$M = \sum_{i=1}^{j} f_i - \sum_{k=1}^{L} \lambda_k + q = 8n - 2 - 5(2n-1) + 2(n-1) = 1 \qquad (3.28)$$

The deployment of a 4-loop network is depicted in Figure 3.15b. The bars on two sides close on top of each other about the common diagonals of the modules during the deployment. The resulting motion looks like an arm rotating from a horizontal plane to a vertical bundle about a pivot point, similar to Type-II Version-II.

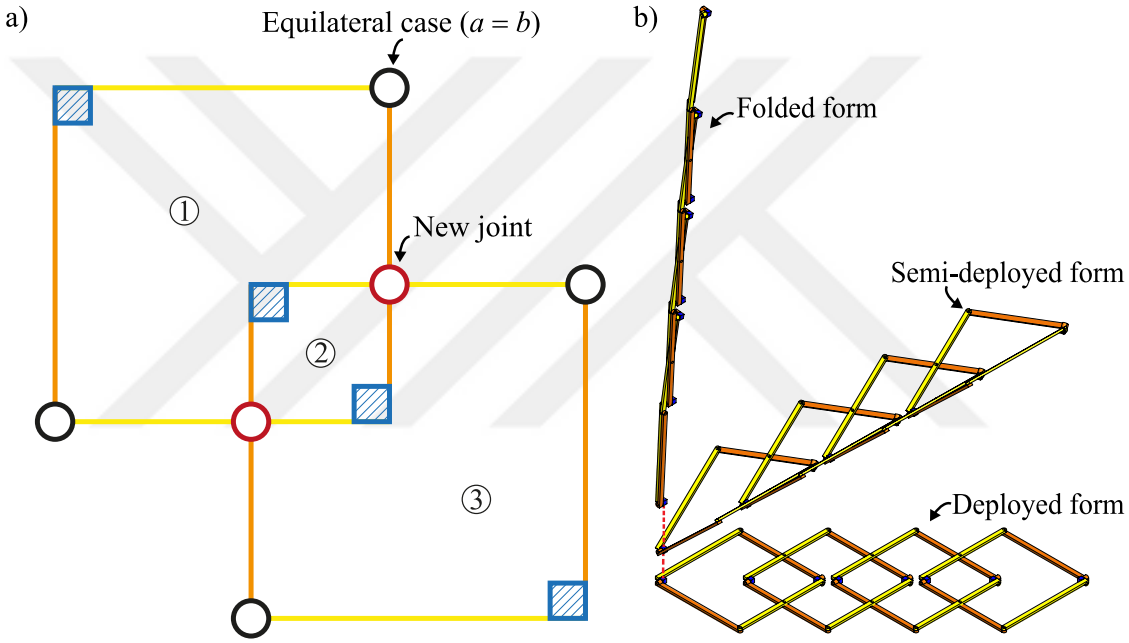


Figure 3.15. Details of Version-I: extra modified Altmann loop version a) schematic, b) deployment.

3.2.3.2. Version-II: Extra Parallelogram Loop Version

If the R-joint corners of the modules are used to connect adjacent loops iteratively ($j_{NT} = 0$), an extra parallelogram loop version is formed (Figure 3.16a). Since the new loop is a parallelogram, both modules are suitable for the proposed version and the network is not necessarily equilateral. Thus, Special case-II modules are used and illustrated for the general case ($a < b$). The new loop is a planar loop with $\lambda = 3$ and does

not add excessive constraints to the system ($q = 0$). Equations (3.26)-(3.27) are valid for the version. There are $n - 1$ parallelogram loops in a network of n loops according to Equation (3.27). If we calculate mobility of the network by using Equation (3.13):

$$M = \sum_{i=1}^{j} f_i - \sum_{k=1}^{L} \lambda_k + q = 8n - 2 - 5n - 3(n-1) + 0 = 1 \tag{3.29}$$

Motion of the version is similar to Type-I Version-I in a proper assembly mode: modules twist about their diagonals, and the resulting motion is a unidirectional deployment as depicted for the 4-loop model shown in Figure 3.16b. The geometry of the deployment motion depends on both selection of initial link lengths (a, b, c, d), and the election of the location of the new joints that determines the dimensions of the parallelogram loops. Also, note that the modules need not be identical and not even similar. Even if the version is obtained with connection of short link to long link, other alternatives are possible.

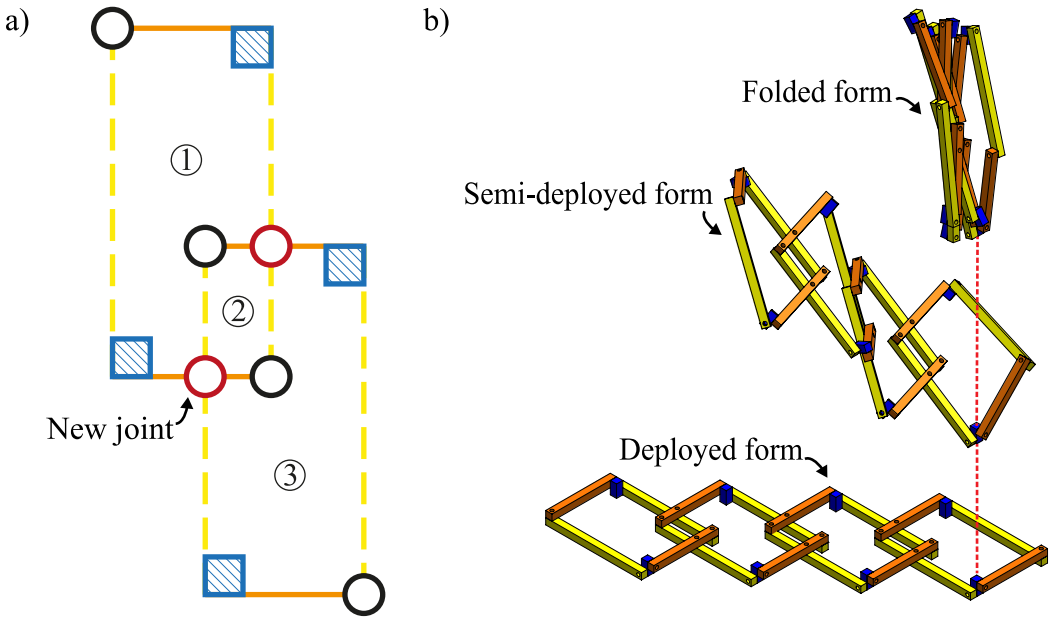


Figure 3.16. Details of Version-II: extra parallelogram loop version a) schematic, b) deployment.

3.2.3.3. Version-III: 4-loop Version

Version-III is a recursively multiplied ($j_{NT} = 2$) network as a combination of Version-I and II networks with four loops (see Figure 3.17a). Since only Special case-I with equilateral modules ($a = b$) is suitable for Version-I, it is used to form Version-III. In addition to the extra two parallelogram and two scaled Special case-II loops, there is an extra parallelogram loop in the middle (loop 9). We can calculate l, j and L for $n = 4$ using Equations (3.15)-(3.16) and (3.25) as

$$l = 6 \times 4 = 24 \tag{3.30}$$

$$j = 6 \times 4 + (2 - 0)(4 - 1) + 2 - 0 = 32 \tag{3.31}$$

$$L = 32 - 24 + 1 = 9 \tag{3.32}$$

Three of the loops are parallelogram loops, while the remaining six are modified Altmann loops. Due to loop of loops there are 8 excessive constraints ($q = 8$). Thus, mobility of the network, according to Equation (3.13) is

$$M = \sum_{i=1}^{j} f_i - \sum_{k=1}^{L} \lambda_k + q = 32 - 5 \cdot 6 - 3 \cdot 3 + 8 = 1 \tag{3.33}$$

Deployment of the network in a chosen assembly mode is shown in Figure 3.17b. This time, a yellow bar in the network is fixed instead of a hub. The network folds about the common diagonal of three parallelogram loops. New loops can be connected on each side of Version-III to multiply with either extra Altmann loops or parallelogram loops.

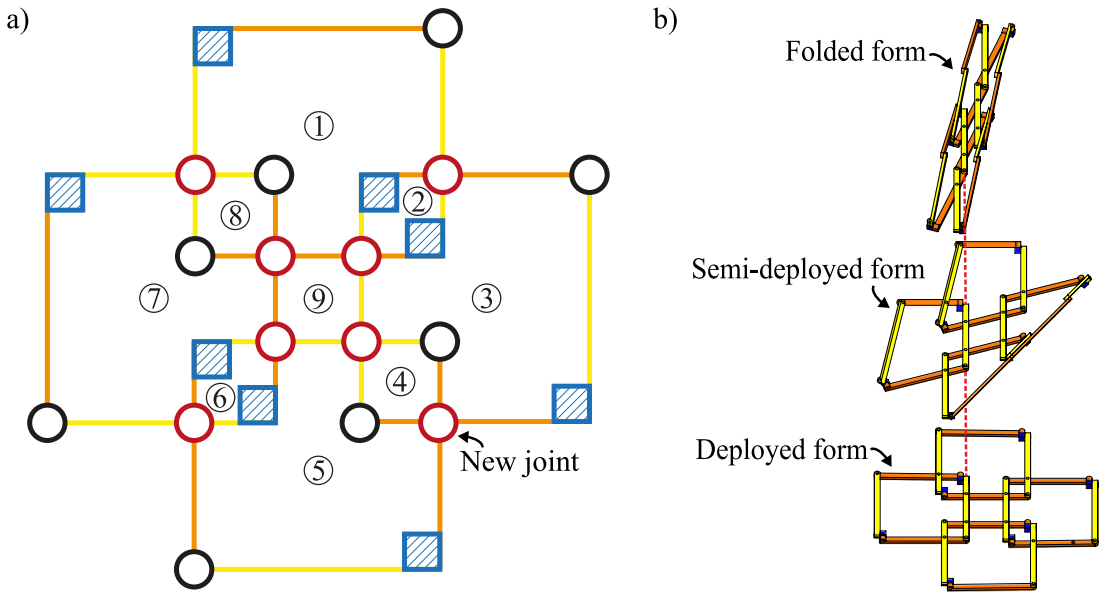


Figure 3.17. Details of Version-III: 4-loop version a) schematic, b) deployment.

3.2.4. Type-IV: Four New Joints

Type-IV is the second network type obtained with Method-II ($l_C = 0$, $j_C = 0$, $l_{CT} = 0$, $j_{CT} = 0$) that includes connection of the adjacent modules with new R-joints and new hubs yielding a new extra modified Altmann loop as shown in Figure 3.18a. There are two versions, Version-I: extra modified Altmann loop and Version-II: 5-loop version. Version-II is introduced in Song et al. (2016). Since there are scaled modified Altmann loops, Special case-I modules are utilized to build this type of networks.

3.2.4.1. Version-I: Extra Modified Altmann Loop Version

By connecting the adjacent loops in the vicinity of R-joints using two new hubs ($l_N = 2$) and four new R-joints ($j_N = 4$) the assembly shown in Figure 3.18a is obtained. The network can be extended by iteratively multiplying new modules ($l_{NT} = 0$, $j_{NT} = 0$,). There are no excessive constraints ($q = 0$). The number of links (l), the number of joints (j) and the number of independent loops (L) in a network of n-loops are as follows:

$$l = 6n + (2-0)(n-1) + 0 - 0 = 8n - 2 \tag{3.34}$$

$$j = 6n + (4-0)(n-1) + 0 - 0 = 10n - 4 \tag{3.35}$$

$$L = 10n - 4 - 8n + 2 + 1 = 2n - 1 \tag{3.36}$$

Then, mobility of the network is calculated as follows:

$$M = \sum_{i=1}^{j} f_i - \sum_{k=1}^{L} \lambda_k + q = 10n - 4 - 5(2n-1) + 0 = 1 \tag{3.37}$$

The CAD model of a 4-loop network is shown in Figure 3.18b. The assembly can attain an arch-like shape during the deployment with a limited motion capability due to link collisions. Common orange and yellow links of loops 1, 2 and 3 can easily collide due to rotation of the newly added hubs (red). Link collisions can be avoided, or at least the motion range can be extended via proper constructional designs of the links, but this would result in difficulty of production of the links.

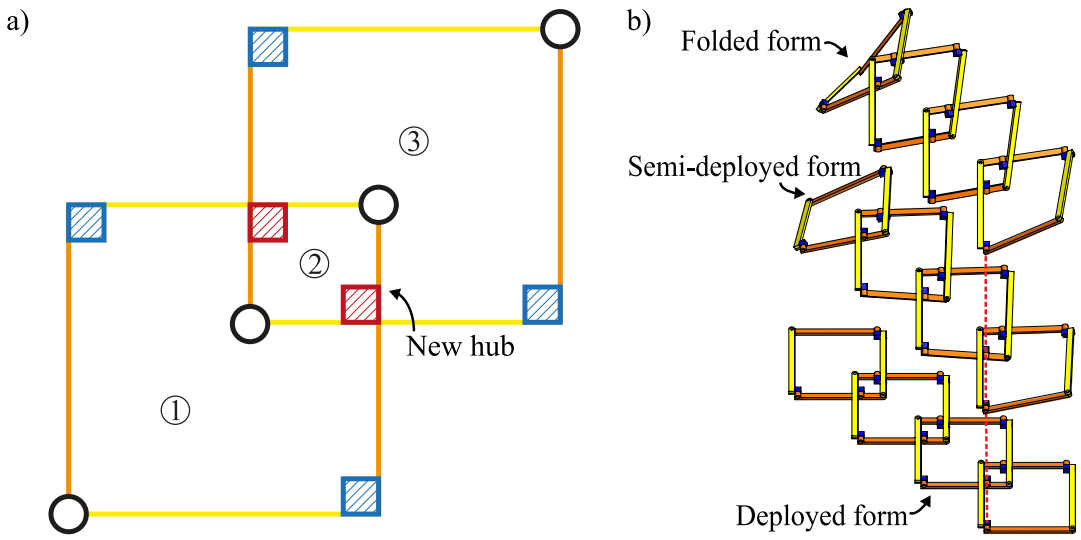


Figure 3.18. Details of Version-I: extra Altmann loop version a) schematic, b) deployment.

3.2.4.2. Version-II: 5-loop version

Version-II is obtained through recursive connection of five loops (loops 1, 3, 5, 6, 8 in Figure 3.19a), yielding four new modified Altmann loops (loops 2, 4, 7, 9 in Figure 3.19a). Schematic of the network is shown in Figure 3.19a for equilateral case ($a = b$). Loops 2 and 7 can be seen as the result of Type-III Version-I connection of loops 1, 3 and 6. Loops 4 and 9 can be seen as the result of Type-IV Version-I connection of loos 5, 3 and 8. The middle module (loop 3) is common for the Type-III and IV connections. According to Equations (3.28)-(3.37) the dof of the assembly is 1. Deployment of the 5-loop network is shown in Figure 3.19b for a selected assembly mode. An arch-like shape is obtained in the folded form. Still, the motion is limited due to Type-IV Version-II sub-assembly in the network.

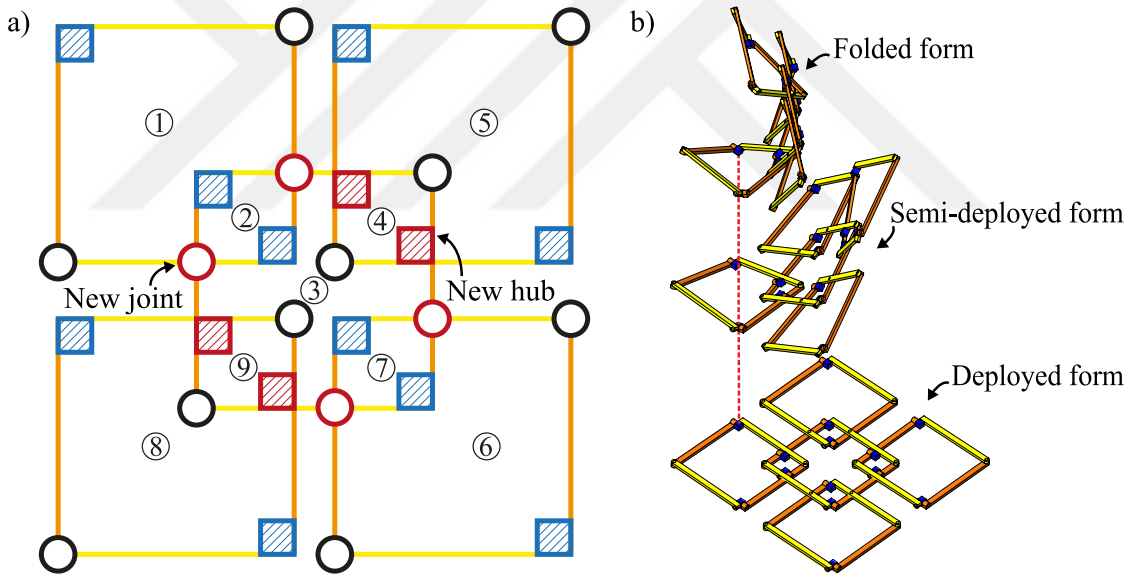


Figure 3.19. Details of Type-IV Version-II: 5-loop version a) schematic, b) deployment.

3.3. Prototypes

In this section, we present prototypes of some of the networks to verify the mobility and motion of them. The first prototype is a 6-loop Type-I Version-I Design alternative-I network. The CAD model of the prototype is shown in Figure 3.20a. The

design is based on Special case-I modules. Bolts, nuts and washers are used for the joints in all presented prototypes. Washers prevent contact between same type of materials and high friction. A brown common bar moves inside a green common bar slot for each pair of adjacent modules, hence a compact design is obtained. The prototype parts are 3D printed from polylactic acid (PLA) material. Deployment of the prototype is illustrated in Figure 3.20b.

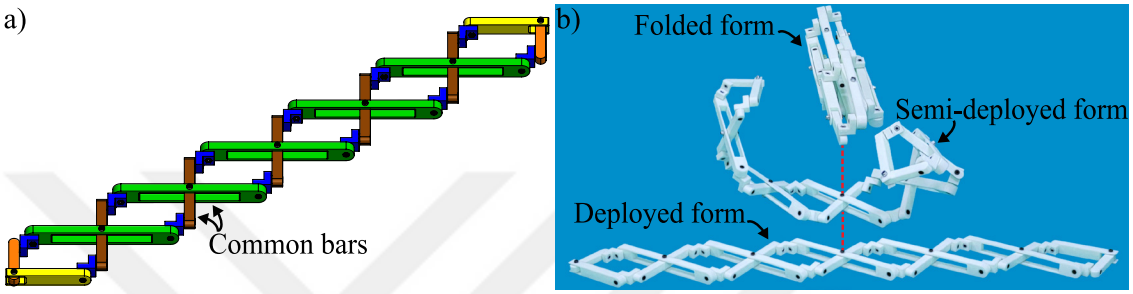


Figure 3.20. A prototype of Type-I Version-I Design Alternative-I a) CAD model, b) deployment.

A Type-I Version-I Design alternative-IV prototype shown in Figure 3.21a is designed for an assembly of four Special case-II modules. All links are aluminum bars with square cross-section. To provide kink angle, common bars of the adjacent basic modules are welded. Deployment of the prototype is illustrated in Figure 3.21b.

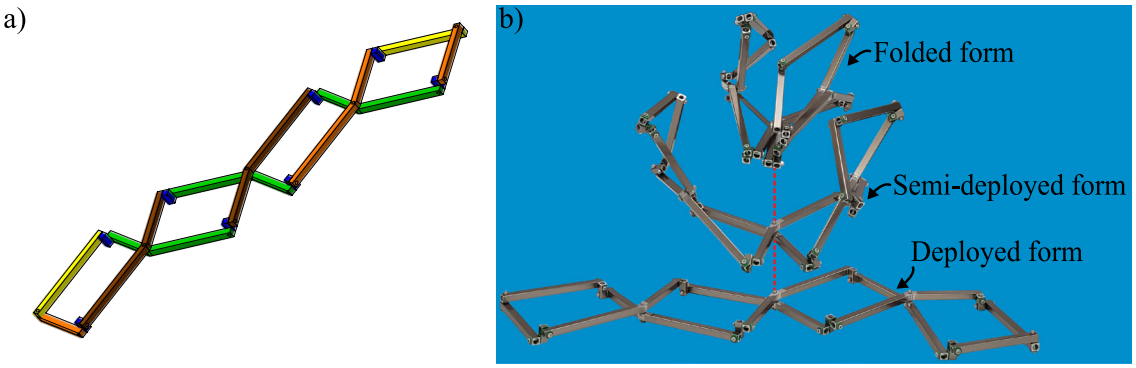


Figure 3.21. A prototype of Type-I Version-I Design Alternative-IV a) CAD model, b) deployment.

A Type-II Version-III prototype is designed for an assembly of equailateral Special Case-I modules (see Figure 3.22a). The prototype features a 4-loop Type-II Version-III network multiplied by four additional 3-loops. Links are 3D printed from PLA material. The deployment process of the prototype is illustrated in Figure 3.22b, where the planar network transforms into a bulging cylinder form.

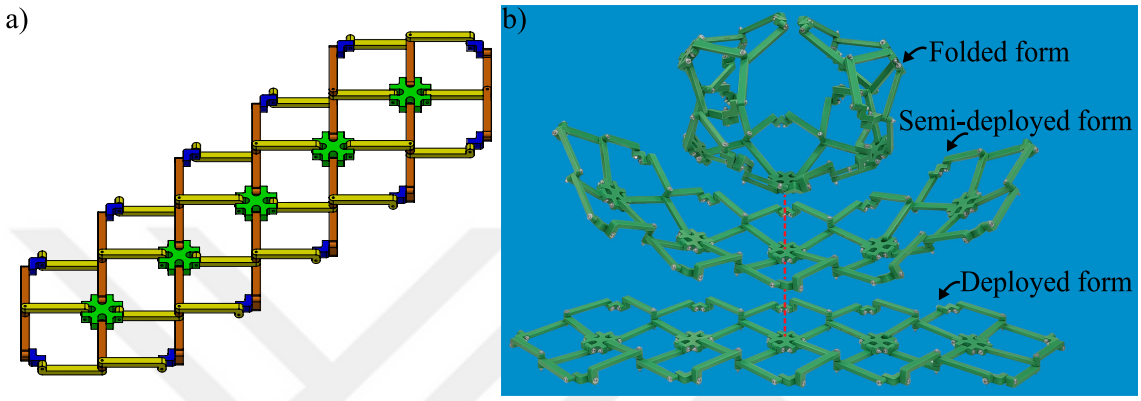


Figure 3.22. A prototype of Type-II Version-III a) CAD model, b) deployment.

A double-layer network example of a Type-II Version-III network prototype is presented in Figure 3.23. First, four Special case-I modules are connected to have a Version-III loop. Then two of them are connected with four new R-joints in a mirror-symmetric position. Two such sub-assemblies of 8-loops are connected with extra hubs and R-joints to obtain the network shown in Figure 3.23a. For the prototype, slender aluminum bars are used for the links. The hubs and kinematic elements of the links are 3D printed from PLA material. Deployment of the prototype is shown in Figure 3.23b. The network is in the form of a flat surface in the deployed state.

A prototype of the dome-like structure in Kiper (2016) built from 9 pairs of Type-II Version-III subassemblies is presented in Figure 3.24. A 4-loop Type-II Version-III assembly with equilateral modified Altmann loops is connected in a mirror-symmetric manner via 4 R-joints to another, but smaller 4-loop assembly to obtain a double layer unit. Such 9 units are assembled with each other via proper connections (see Kiper (2016)) to result in a single-dof dome-like structure. In this prototype, glass-fiber composite tubes are used for the main body of the links, whereas 3D printed plastic taps and laser cut sheet metal inserts are used for the kinematic elements of the links. Sheet metal inserts are

connected via bolts and nuts. The hubs are also 3D printed and metal pints are used to connect them to sheet metal inserts of the associated bars. The metal connection parts of the 9 double layer units are produced via turning, drilling and laser cutting. This is a first prototype of a tent frame and suffers bending down due to gravity without a central support column.

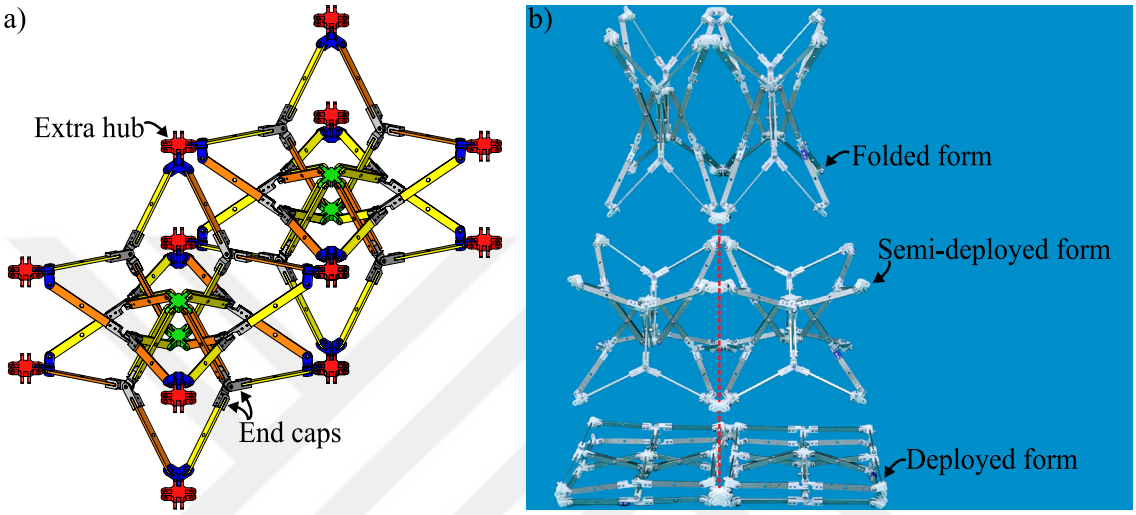


Figure 3.23. Another prototype of Type-II Version-III a) CAD model, b) deployment.

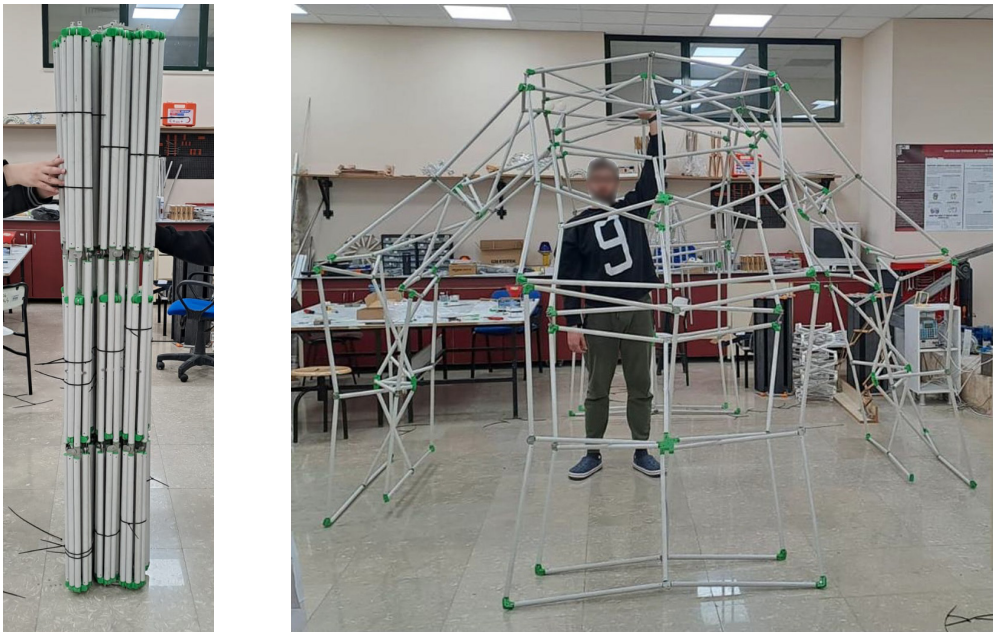


Figure 3.24. The folded and deployed states of the prototype of a 72-loop Type-II Version-II assembly.

A Type-III Version-II prototype is designed for an assembly of six Special case-II modules (Figure 3.25a). Links, including the hubs, are aluminum bars with square cross-section. Deployment of the prototype is illustrated in Figure 3.25b. The prototype shows how simple design solutions can be obtained with Special case-II modules.

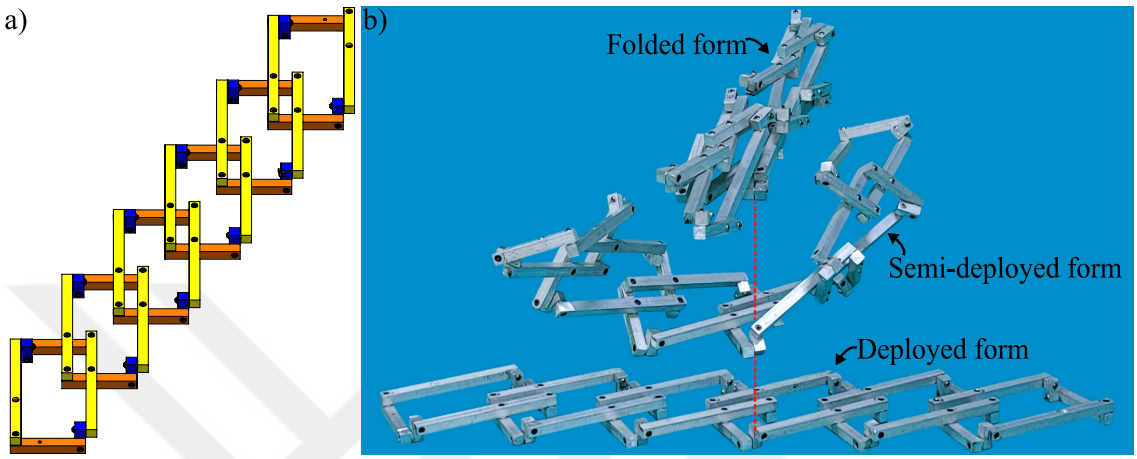


Figure 3.25. A prototype of Type-III Version-II a) CAD model, b) deployment.

The last prototype is shown in Figure 3.26a as an example of Type-III Version-III networks of equilateral Special case-I modules. Links are 3D printed with PLA material. The deployment of the prototype is shown in Figure 3.26b.

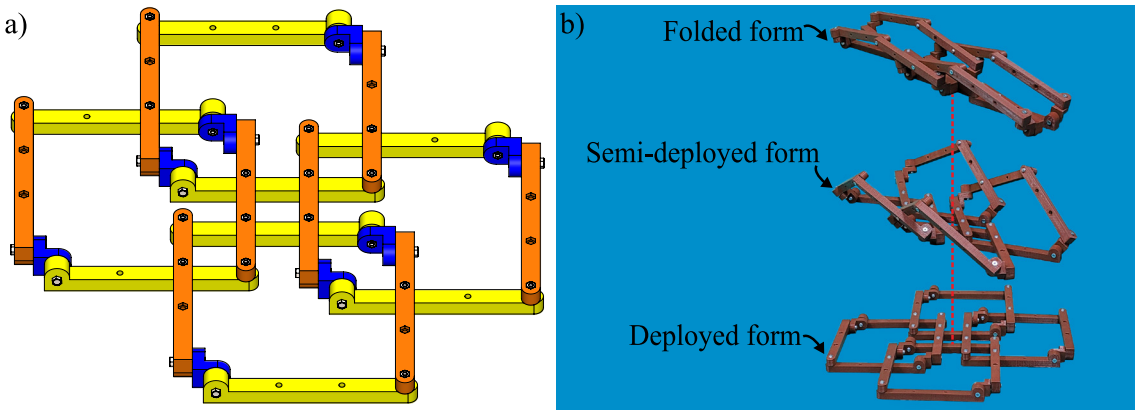


Figure 3.26. A prototype of Type-III Version-III network) CAD model, b) deployment.

3.4. Discussion of the Chapter

In this Chapter, two modifications of Altmann linkages are introduced, which are still special cases of a general line-symmetric Bricard linkage. Single-layer networks of these linkages are investigated. The modules provide an improvement to the complex constructional design requirements of an Altmann linkage. The special case involving zero link length hubs results in a configuration where all links are parallel in a bundle in the folded form. This case is beneficial when a network needs to be fully folded. The novel case with non-zero link length hubs only maintains the line-symmetry property of Bricard linkage. This second case is advantageous over the first case in terms of simpler link geometries. Existing and two novel networks are classified with possible versions. The networks are obtained either with common links/joints method, or new links/joints method for a selected module type. Mobility analyses of the networks are performed and their motion characteristics during deployment are described. The results show that even minor changes in connection parameters substantially affect the network deployment and resulting motion. The motion characteristics make them suitable for different application fields of engineering and architecture such as deployable antennas or shelter frames. To limit the length of the paper, versions are not further analyzed, and some versions are presented with only general cases; though design alternatives are possible. The generic forms ($a \neq b$) of the modules are used for illustration whenever possible, however, networks yield more compact folded forms and desirable motions for the equilateral case ($a = b$). The feasibility of certain networks is proven through built prototypes. Overall, the paper contributes to the literature with a classification of several single-layer networks. Further work can be conducted on mathematical modeling and design of the networks to optimize the deployment ratio for specific applications. In addition, the methodology presented in this Chapter to multiply the modules can be extended to obtain multi-layer networks of the modules.

CHAPTER 4

CUSTOM API DESIGN FOR PARAMETRIC DESIGN AND SIMULATIONS OF DEPLOYABLE STRUCTURES

This Chapter presents the developed parametric design environment and CAD software automation for deployable structures composed of rigid deployable components. The mathematical model of these structures is used to satisfy a range of constraints for the desired poses in the folded form, during deployment or in the deployed form. The model can include constraints at the following levels from specific to generic: 1) kinematic parameters, 2) cross-section constructional parameters and 3) 3D geometry constructional parameters. Even though parametric model development is simpler for proof-of-concept prototypes, functional or further developed prototypes of engineering application models are complex. Generic-level models can be developed to ensure scalability, allowing easy creation of design variations with different-sizes and their corresponding CAD models. Additionally, implementing constructional parameters in CAD models is time-consuming if done manually during design iterations.

Simulations are run to explore the structural behavior of digital prototypes under desired load conditions. Earth or outer space applications of these structures can require different types of simulations such as modal, linear statics, non-linear statics, buckling, dynamic frequency response, direct transient response, power spectral density (PSD) analysis etc. The physical prototype budget can be decreased by the simulations, and structural optimizations for topology, topometry, shape and size can be performed. When there are large numbers of joints and links in the design, preprocesses of FEA models are time-consuming. Several repetitive tasks are performed during the model generation phase.

This Chapter aims to

1) develop a parametric design environment that allows the integration of a deployable structure parametric model to a CAD environment for design and simulation purposes.
2) develop custom API for automation of FEA simulation processes.

The parametric design environment and its CAD environment integration are introduced in the following section. Then, automation of FEA simulation processes is given.

4.1. Custom API Design for Parametric Design and CAD Simulations

Microsoft Excel is used for parametric model calculations and to obtain design environment, while Solidworks is the CAD environment. The automatic recalculation feature of Excel allows dynamic recalculation of the whole model response to each cell (parameter) change. This makes it suitable for the mathematical modeling studies of mechanisms and visualizing desired parameters and shape of the mechanism. Several examples of applications for kinematic synthesis of mechanisms can be found in (Söylemez, 2023). Modifying the parameters and updating the Solidworks model during the parametric modeling process and design iterations is often necessary. Therefore, an effective way to set up a model between Solidworks and Excel should sought. While Solidworks can import data from Excel via the "Design Table" feature, this approach is found to be less practical as the number of parts increases, requiring significant computational resources. Furthermore, it is not a viable solution because it necessitates opening the relevant Excel file for each part when the CAD file is opened.

Recording macros and communicating with other programs is common in CAD programs. Many programs provide API libraries for this. Solidworks provides API library support and can communicate with other programs in VBA, VB.NET, Visual C#, Visual C++ 6.0 and Visual C++/CLI (Solidworks, 2024). Sending commands and receiving responses to Solidworks via Excel in VBA language is possible using the API library. As a solution, an efficient method for transferring parameters in terms of computational time is to use the “Equations” tool in Solidworks. This tool enables the user to define and modify the parameters of equations and link them to a .txt file. It is possible to command equations directly via the API library. However, in such a case, updating the common parameters on each part individually during the design changes is necessary. In contrast, Excel can update the linked .txt file and issue commands to Solidworks open parts in a shorter time or rebuild opened parts with new parameters. A custom API design main flow is shown in Figure 4.1 for this task.

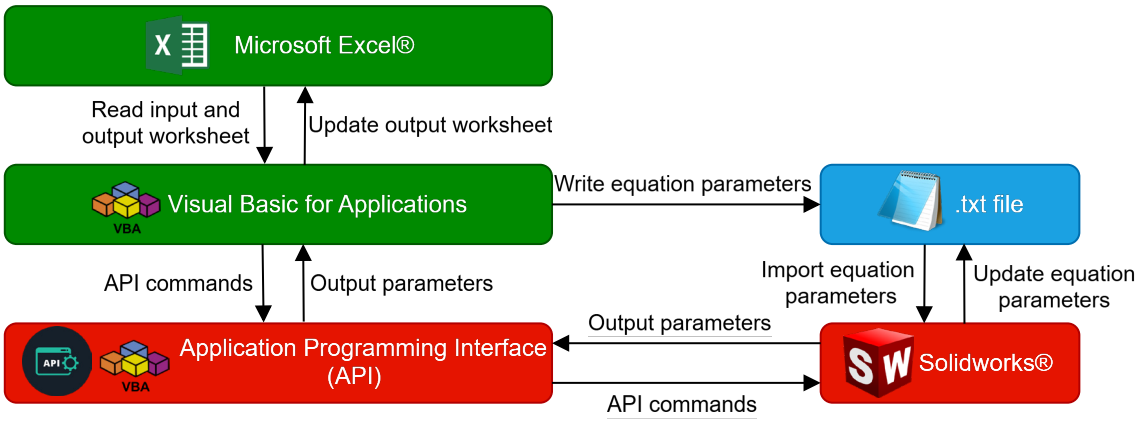


Figure 4.1. Main flow of the parametric design integrator program.

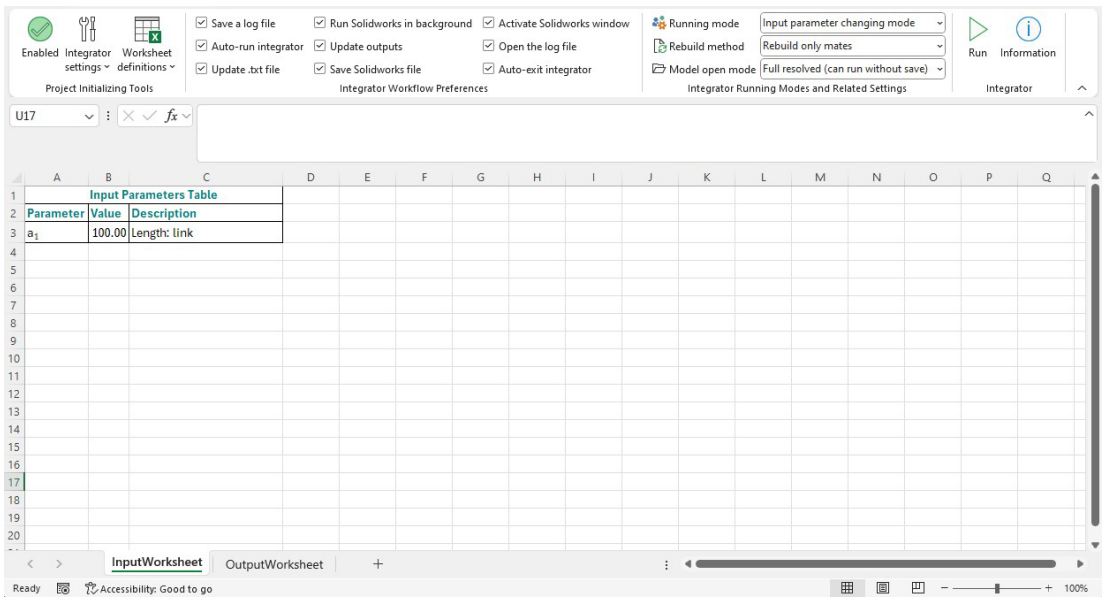
Numerous additional API functions and code libraries are available for almost every tool in Solidworks. These include opening a file, initiating a drawing, drawing, defining a point/axis/plane, reading/writing their values, controlling model properties such as material, mass and volume, drilling and extrusion. These features can be defined and turned on/off. Additionally, simulations can be run from add-ins such as Solidworks Simulation and Solidworks Motion for FEA or multi-body dynamics simulations. In addition, the angle and distance parameters can be controlled, which is useful for mechanism simulations. Points can be defined in Solidworks, and their coordinates can be sent to Excel as feedback. The model can be verified by comparing these coordinates with the parametric model coordinates in Excel. Furthermore, the orientation of the Solidworks model allows the creation of a synchronized view of the mechanism in both programs.

4.1.1. Description of the Parametric Design Integrator

A VBA code is written and related graphical user interfaces (GUI) are designed to implement required API features for parametric mechanism design. The program is converted to an Excel add-in called Parametric Design Integrator (PDI). “Office RibbonX Editor” (Github 2024) is used to develop the add-in. Figure 4.2a shows an input worksheet example. a_1 is a parameter defined to use in the CAD environment. A parameter can be defined by specifying parameter name, value and description.

Top side of the Figure 4.2a shows the PDI ribbon. The ribbon includes four groups: 1) project initializing tools, 2) integrator workflow preferences, 3) integrator running modes and related settings, and 4) integrator. The program is defined in two worksheets for parametric model inputs and outputs (see Figure 4.2a). The user defines input worksheet, output worksheet and other initializing settings from the project initializing tools group. Then, workflow steps are set in the second group as save a log file, auto-run integrator, update .txt file, run Solidworks in the background, update outputs, activate Solidworks window, and auto-exit integrator options. The third group specifies the rebuild method (force rebuild all, rebuild all, rebuild only mates and do not rebuild) and model open mode (fully resolved, lightweight). Running mode allows the selection of predefined modes (input parameter adding mode, input parameter changing mode, output parameter update mode). After selecting these options, the main program GUI is called from the run button in the integrator group (see Figure 4.2b). In this interface, the user is informed about the defined input and program settings. Errors are handled with user messages. When the program is run, it provides information about the update steps and task establishing percentage.

a)



(cont. on next page)

Figure 4.2. Details of PDI a) input worksheet of PDI, b) main program GUI.

b)

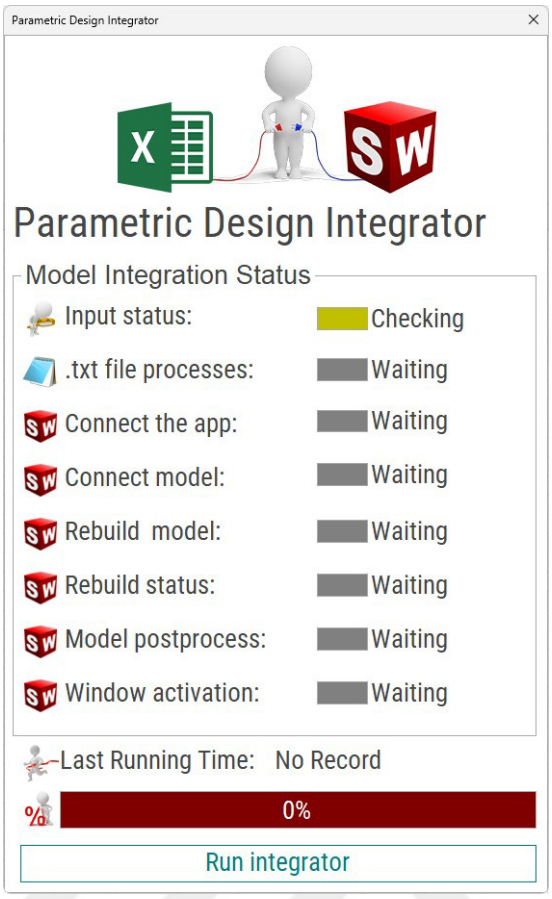


Figure 4.2 (cont.)

An output worksheet example is shown in Figure 4.3. The output worksheet includes a table of output parameters and a Solidworks rotation matrix table. Users define an output parameter by specifying the parameter name, type and three values. An angle feature can be used to control the pose of the model in the CAD environment. This corresponds to advanced angle mate in the CAD software. Values of an angle represent the current value, minimum limit and maximum limit, respectively. A point can be used to verify the coordinates of a parametric model. Values represent the 3D coordinates of a point. Similarly, other mate/feature types (axis, plane, etc.) can be added to the integrator. Currently only point feature and advanced angle mate are defined in the library. Additionally, the Solidworks rotation matrix table on the right is for animating the mechanism. 3D coordinates of a point can be projected on a 2D view in Solidworks model view orientation by this output. An animation example is presented in Figure 5.38 for the case-study project.

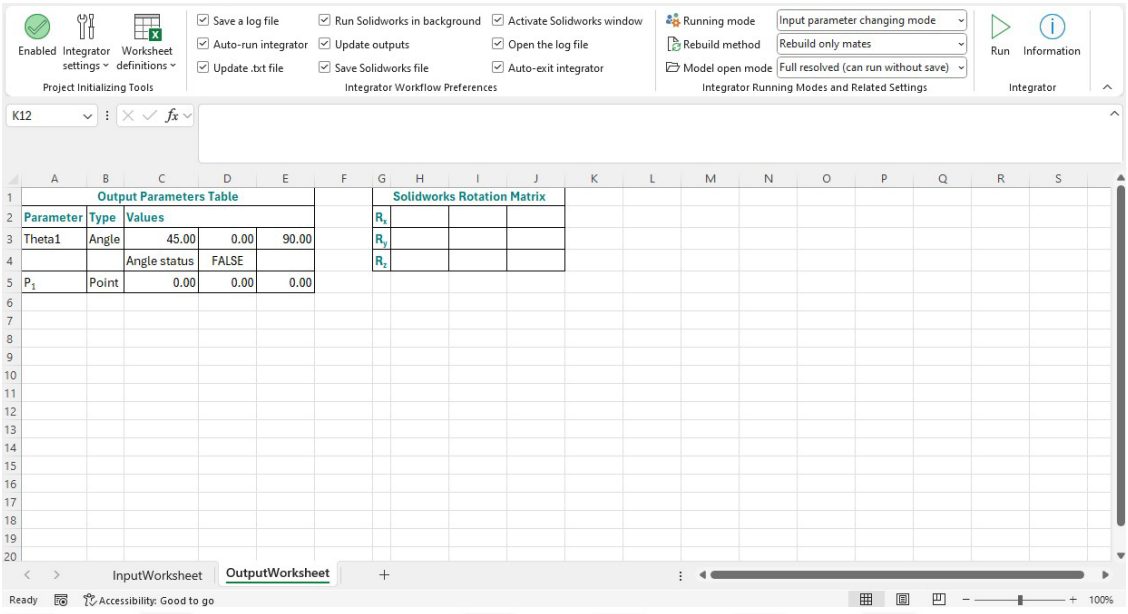


Figure 4.3. Output worksheet of PDI.

4.1.2. Workflow of the Program

The integrator's flowchart is shown in Figure 4.4. Similar task boxes have the same color in the figure. The program begins by reading the settings selected by the user, and updating GUI status labels according to that. Then, it shows the GUI. If auto-run is enabled, the program initiates automatically; otherwise, it waits for a user command to run or exit.

Once started, the program verifies the program settings, checking for the presence of the project folder, model file, and text file. Missing project folder and model file inputs trigger an error message. If that happens, PDI goes back to wait for a user command. Next, if logging is enabled, the program begins recording to a log file. The log file includes detailed information about each step and their running times. After that, .txt file operations are started. If updating the text file is requested, the integrator verifies the status of the file and creates it if necessary. Then it reads and writes input worksheet parameters to the text file. Otherwise, PDI proceeds with the Solidworks software connection step.

To connect Solidworks, the software should be set to run as administrator for integration. Since Windows does not allow changing the start permissions of software by

a code, the user should have done that before. If a Solidworks instance is already running, the program connects to it; otherwise, it opens a new session. A successful connection is required for the program to continue, and any connection failure will show an error message. The user can choose to run Solidworks in the background, and the program will be invisible if this option is activated. This setting results in faster computations since Solidworks does not need to update the graphics view.

The next stage involves opening the specified Solidworks model, with options to load it in either lightweight or fully resolved mode. Lightweight mode allows faster opening by loading only essential data, while fully resolved mode loads all part data. If the model opens without issues, the program moves forward; otherwise, it shows an error and goes back to wait for a user command.

If the user selects any other rebuild option than "don't rebuild", then the rebuild operation proceeds. Solidworks offers three rebuild options: force rebuild all, rebuild all, and rebuild only mates. The integrator applies the chosen rebuild mode before proceeding to output worksheet operations.

If output updates are specified, the integrator reads output worksheet parameters. If a mate (advanced angle mate) is defined in the output worksheet, then the mate is updated. Next, it reads Solidworks output data. This data includes feature parameters (point coordinates) and Solidworks view rotation matrix. The data are updated in the output worksheet.

The final sequence comprises the ending operations. If requested, the Solidworks window is activated, and logging stops if it is in the record. Finally, if auto-exit is enabled, the GUI closes.

4.1.3. Kinematics Library

An Excel VBA library is developed to perform kinematic computations. The library includes transformation (*Transformation*, *TransformationX*, *TransformationY*, *TransformationZ*), rotation (*Rotation*, *RotationX*, *RotationY* and *RotationZ*) and symmetry (*LineSymmetric*, *PlaneSymmetric*) functions. The *Transformation* and *Rotation* functions are designed to route inputs to their respective x-, y- and z-axis subfunctions.

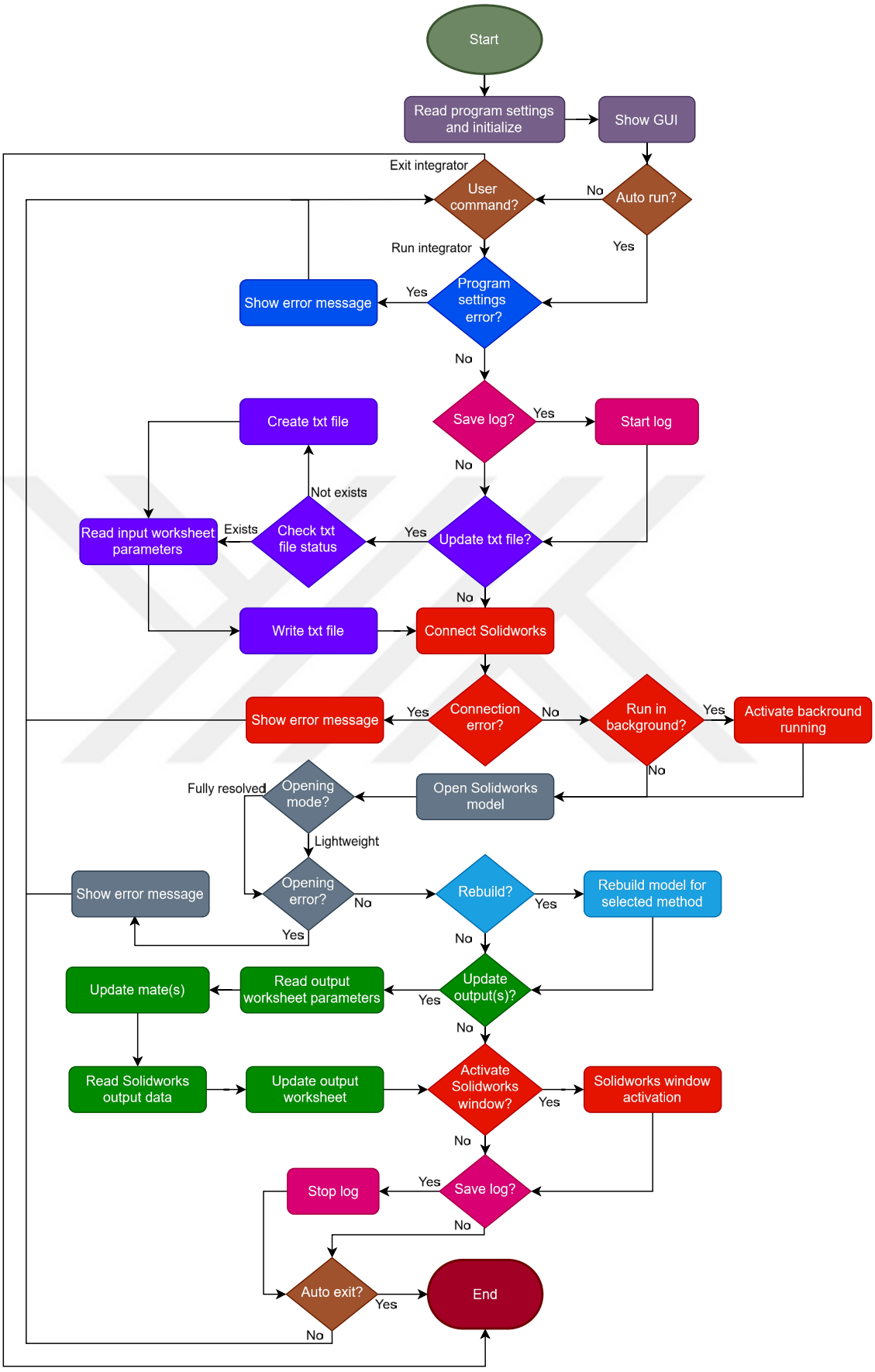


Figure 4.4. Flowchart of PDI.

The function syntax is *Transformation*(*Axis, Translation*, *Rotation*, *InitialPose*). There are four inputs where *InitialPose* is an optional input. Axis is a string type of input and the first letter should start with x, y or z. If the input is longer than one character, the function takes the first character as input, such that even inputs line "x1" or "y_rotation" are interpreted as "x" or "y". Axis input directs the function to *TransformationX*, *TransformationY* or *TransformationZ* for performing the specific transformation along the specified axis. The *Translation* input defines the displacement vector along the chosen axis, while the *Rotation* input specifies the angle of rotation (in radians) to be applied about the axis. These inputs work in combination to generate a transformation matrix, which is then used to compute the new position and orientation relative to the Cartesian frame. If provided, the *InitialPose* input represents the starting pose (as a matrix or coordinate set), allowing the function to apply the transformation relative to a predefined configuration.

The *TransformationX*, *TransformationY*, and *TransformationZ* functions use the syntax (*Translation*, *Rotation*, *InitialPose*) from the main function to perform transformations along the respective axes. The function returns the updated pose as a 4×4 homogenous transformation matrix output, integrating both translation and rotation effects for the specified axis. The transformation operations are defined as screw displacement matrix in McCarthy and Soh (2010, 255).

Equations (4.1) define the transformations along x-, y- and z-axes where $d = Translation$, $\theta = Rotation$, $T_i = InitialPose$ and $X(\theta, d)$, $Y(\theta, d)$ and $Z(\theta, d)$ are the transformation matrices. Here, notation of McCarthy and Soh (2010, 255) is adapted.

The rotation function computes a rotation matrix based on the specified axis and angle. The syntax is *Rotation*(*Axis*, *Angle*), where the *Axis* input (x, y or z) directs the function to *RotationX*, *RotationY* or *RotationZ* similar to the *Transformation* function. Extra equations are not needed for these rotations as they are directly defined within the *Transformation* function output Equation (4.1). The upper-left 3×3 submatrices of the resulting 4x4 transformation matrices correspond to outputs of *RotationX*, *RotationY* and *RotationZ*, respectively.

The *LineSymmetric* function computes the reflection of a point about a line of symmetry defined by a reference direction and the distance from the origin along the remaining two directions. The function syntax is *LineSymmetric*(*RefPoint*, *Direction*, *Axis1Distance*, *Axis2Distance*), where *RefPoint* is the point to be reflected, Direction specifies the reference direction of the symmetry line, and *Axis1Distance* and

Axis2Distance define the distances from the symmetry line to the origin along the two axes perpendicular to the reference direction. Figure 4.5 shows an example of the symmetry operation. The symmetry line L_1 is in the x-direction (v_{L1}), and the y- and z-axis distances to the origin B_1 are defined by $\Delta_1 = Axis1Distance$ and $\Delta_2 = Axis2Distance$. The function reflects the point ($RefPoint = P_R(x, y, z)$) based on these inputs and returns the new position of the point ($P'_R(x', y', z')$) after the reflection in Equation (4.2).

$$
\begin{aligned}
TransformationX(\theta,d,T_i) &= [X(\theta,d)]T_i = \begin{bmatrix} 1 & 0 & 0 & d \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} T_i, \\
TransformationY(\theta,d,T_i) &= [Y(\theta,d)]T_i = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & d \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} T_i, \\
TransformationZ(\theta,d,T_i) &= [Z(\theta,d)]T_i = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} T_i
\end{aligned}
\tag{4.1}
$$

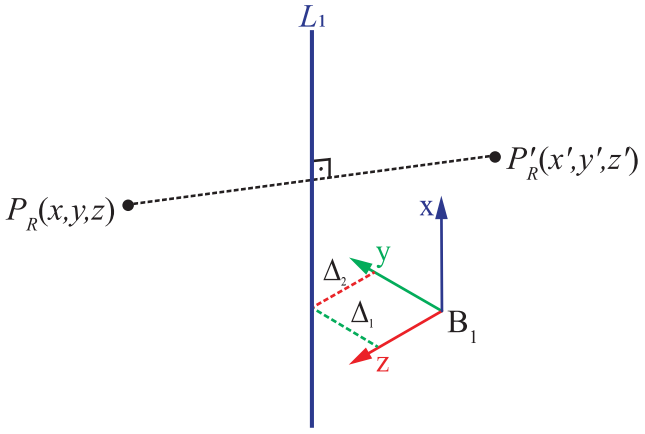


Figure 4.5. Line-symmetry example in y-direction.

$$LineSymmetric\left(P_R\left(x,y,z\right),v_{L1},\Delta_1,\Delta_2\right)=P'_R\left(x',y',z'\right)=\left(x+\Delta x,y+\Delta y,z+\Delta z\right)$$

where:

$$\Delta x=\begin{cases}0 & v_{L1}=x\\ -2x+2\Delta_1 & v_{L1}=y \text{ or } z\end{cases},\Delta y=\begin{cases}-2y+2\Delta_1 & v_{L1}=x\\ 0 & v_{L1}=y,\\ -2y+2\Delta_2 & v_{L1}=z\end{cases} \tag{4.2}$$

$$\Delta z=\begin{cases}-2z+2\Delta_2 & v_{L1}=x\\ -2z+2\Delta_1 & v_{L1}=y\\ 0 & v_{L1}=z\end{cases}$$

The *PlaneSymmetric* function computes the reflection of a point about a plane of symmetry defined by two reference axes and the perpendicular distance from the plane to the origin. The function syntax is *PlaneSymmetric*(*RefPoint*, *PlaneDistance*, *Axis1*, *Axis2*). Here, *RefPoint* is the point to be reflected, *PlaneDistance* specifies the distance of the symmetry plane to the origin along the axis perpendicular to the plane, and *Axis1* and *Axis2* define the two axes forming the symmetry plane. For instance, if the symmetry plane is defined by the x- and y-axes ($v_{A1}=Axis1=x, v_{A2}=Axis2=y$), then $\Delta_1=PlaneDistance$ is measured along the z-axis (see Figure 4.6). The function reflects the point $P_R(x,y,z)$ across the specified symmetry plane and returns the new position ($P'_R(x', y', z')$) in Equation (4.3).

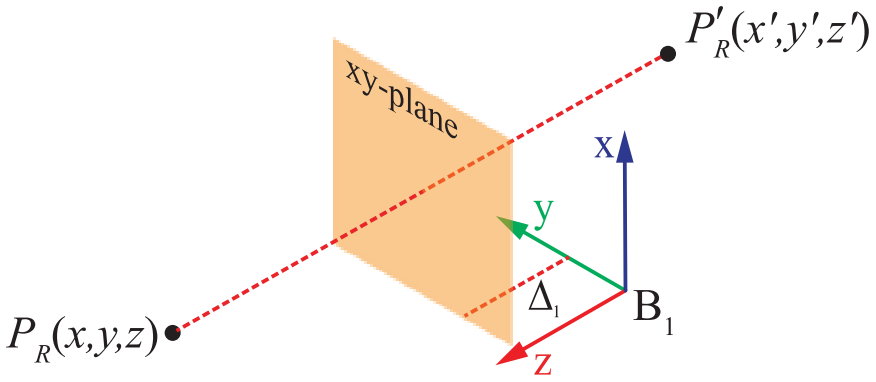


Figure 4.6. Plane-symmetry example about xy-plane.

$$PlaneSymmetric\left(P_R\left(x,y,z\right),\Delta_1,v_{A1},v_{A2}\right)=P_R'\left(x',y',z'\right)=\left(x+\Delta x,y+\Delta y,z+\Delta z\right)$$

where :

$$\Delta x=\begin{cases}0 & v_{A1}=x,v_{A2}=y\\ -2x+2\Delta_1 & v_{A1}=y,v_{A2}=z\\ 0 & v_{A1}=x,v_{A2}=z\end{cases},\Delta y=\begin{cases}0 & v_{A1}=x,v_{A2}=y\\ 0 & v_{A1}=y,v_{A2}=z\\ -2y+2\Delta_1 & v_{A1}=x,v_{A2}=z\end{cases}, \tag{4.3}$$

$$\Delta z=\begin{cases}-2z+2\Delta_1 & v_{A1}=x,v_{A2}=y\\ 0 & v_{A1}=y,v_{A2}=z\\ 0 & v_{A1}=x,v_{A2}=z\end{cases}$$

4.2. Custom API Design for FEA Simulations Automation

MSC Nastran® is one of the most powerful FEA solvers and is widely used in the aerospace industry. The software is selected to perform structural load simulations of the case-study prototypes. Nastran does not have a graphical user interface. It requires a preprocessor and postprocessor. This has two options: MSC Apex® and MSC Patran®. Apex is a newly developed software and has a more user-friendly interface. Hence, the MSC Apex-Nastran pair is selected for FEA simulations.

Model preprocesses take a long time and are prone to mistakes for a mechanism consisting of several parts. Deployable structures composed of rigid parts mostly have several parts, such as the spherical network shown in Figure 1.2. Even though Apex can transform copies of solids/surfaces/meshes, the software cannot copy some features, such as joints. If we continue with the joint example, the spherical network has 524 R-joints, and defining them manually takes a long time. Besides, even if copying of a property is allowed, it is not easy to transform copied components in the correct pose. It is needed to set the pose by position and orientation changes. FEA modelers do not present closed-loop mate solutions of assemblies such as in Solidworks.

Apex provides an API library and related stub files in Python language. The library has broad support on the preprocessor side but it is weak on the postprocess side. A custom API is designed to automate repeating tasks of preprocesses. The program's main flow is shown in Figure 4.7. The program imports the Parasolid model generated by the CAD environment and performs preprocesses. Then, it generates a model definition

file (.bdf) for running the simulation in Nastran. After sending the run command, Apex directly imports the results of a simulation by the .h5 file when the simulation is completed. API can access the postprocess environment for the results. Even though no sizing optimization is performed in this thesis; the program can be used to design a multi-model workflow for sizing optimization tasks. Because, if only dimensional changes are made to obtain different-size models in the CAD environment, a custom API design model developed for that CAD model is still valid for different-size models.

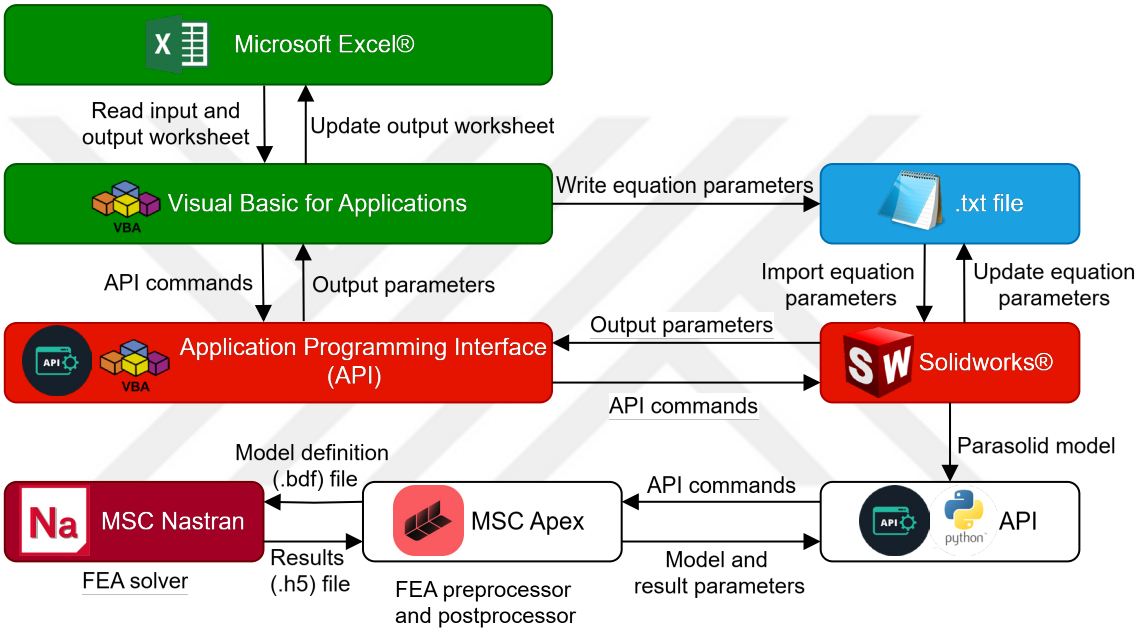


Figure 4.7. Main program flow of custom API designs.

4.2.1. Description of the Program

The object model diagram of MSC Apex is presented in Figure 4.8 (Hexagon 2024). A model represents the highest-level object in Apex, and each running session contains a single model object. This object includes assemblies, parts and their sub-objects. Assemblies can include sub-assemblies, parts, connectors, mesh dependent ties or mesh independent ties. A part includes two main sub-object types: 1) mesh bodies in 3D (solid and hex mesh), 2D (surface mesh), 1D (line mesh) and 0D (point mesh), and 2) geometry bodies including solid, surface, curve and point geometries. The geometry

bodies are composed of geometry topologies cells, faces, edges and vertices. For example, a solid may include all of these topologies, while a surface consists of faces, edges and vertices.

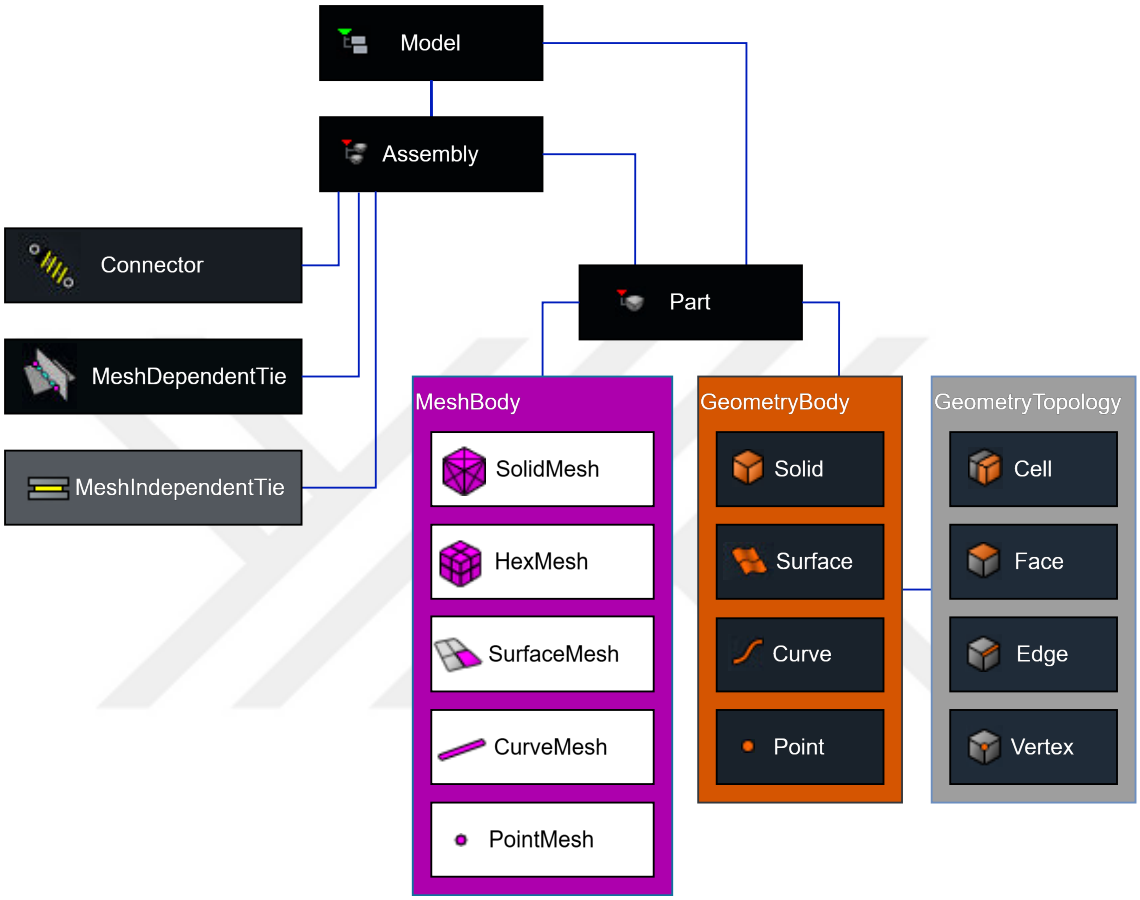


Figure 4.8. Object model diagram of the API library.

Each of these objects has a corresponding class in the API library, making them accessible, modifiable, and updatable through public member functions and attributes. For instance, a solid object provides public member functions such as getCells, getCentroid, getExteriorEdges, getExteriorFaces, getLocation, getOrientation, getVolume, getNodes, getParent, getVertex, getVertices, update, setParent, among others. These capabilities enable extensive manipulation of the FEA model.

Every entity, whether mesh bodies, geometry bodies, or geometry topologies, is assigned a unique entity ID within the model. For example, identical solid objects of two parts (same part type) will share the same cells but possess distinct entity IDs for those

cells. Entities can be stored in entity collections. The most generic collection is "entity collection" and there are sub-collection types such as node collection, surface collection, part collection, geometry collection, joint collection etc. Additionally, objects are associated with local coordinate frames (Cartesian, cylindrical, or spherical). However, the definition of these frames may vary if a part is in reverse orientation.

If a preprocessing operation is defined on a geometry body or topology, the same operation can repeated for parts of the same type by matching their entity IDs. Properties of these objects (length, area, # of faces/nodes, distance between faces, solids, etc.) can be used for comparing objects and matching their entity IDs. Also, the features defined between multiple geometry/mesh bodies and topologies can repeated by matching object and related sub-object IDs. For example, if an R-joint is defined between a hub and link by specific faces, a sample can define to repeat joint definition by matching each couple and related faces in the model.

One of Python's key strengths lies in its handy data types, particularly dictionaries. Dictionaries enable efficient data organization and manipulation. The API library provides functionality to convert entities and their collections into dictionary representations and retrieve them. This feature simplifies data handling in the API, facilitating the integration of geometric and mesh data with Python libraries.

The main structure of the program is defined by dictionaries and operations are structured in them. These dictionaries and their associated operations will be explained in the next section. Additionally, the custom API library is currently implemented in the backend. Although Apex provides frontend tools for designing user interfaces, no efforts have been made in this work to develop a frontend interface due to time limitations.

4.2.2. Workflow of the Program

The flow-chart of the custom API design is shown in Figure 4.9. Similar task groups have the same color in the figure. The program starts by model initializing and log record start. Log record is important in this design because the number of generated entities should be checked for each operation at the end of the program. Then, if the user selects a template, the program opens that template and saves it with the project name. The template is a trick for the program because the API library does not present all features of the program, and some settings can be done only manually, such as feature

mesh settings. A model file can set the specified settings and be used as a template to overcome that. Afterwards, the parasolid file is imported to be processed.

The first dictionary operations are related to the materials. The user defines materials, material models (MAT1, MAT2, etc.) and their properties (density, Poisson ratio, etc.). A material ID variable is defined as empty for each material in the dictionary. The program checks if material operations are defined or not. If they are defined, it generates a dictionary, defines dictionary items in the model, and updates their assigned IDs from the Apex model to the materials dictionary. The program checks if the dictionary operation is defined or not for each dictionary as in the materials dictionary (see Figure 4.9). Thus, this step is not explained for each dictionary for the sake of readers.

Then, element property dictionary operations are started. 2D and 3D element properties and their specifications (primary property 3D, primary property 2D, material name, thickness etc.) are defined in this dictionary. The element properties allow the use of the same material for different geometry/mesh body types. For example, a steel material can be assigned to two different element properties for 2D (PSHELL) and 3D (PSOLID) geometries. The dictionary has a material name variable for each property. This allows accessing material IDs of the materials dictionary by material name. The dictionary is generated, and element properties are defined in the Apex model.

Next, part dictionary operations begin. This step comprises most of the work. A part is defined in the dictionary by sample part path, mesh properties (mesh dimension, size), tool requirements (defeature tool, split tool, edge-offset tool, boolean tool), tool methods and their sample. The step starts with analyzing sample geometry bodies. Only solids are supported for now. There are two alternatives for analysis. If part/solid names are consistent with the part types (for example, the same link has the same name for all instances in the model), the program can call each instance of a part by its name. Or else, geometries (solids) are analyzed for comparison. Ten parameters of security are defined to match solids: solid volume, number of interior faces, number of exterior faces, number of interior vertices, number of exterior vertices, sum of interior face areas, sum of exterior face areas, mass center distance to centroid, sum of interior face distances to the geometry center and sum of exterior face distances to the geometry center. The number of comparison parameters can increase or decrease for different solid properties if desired. These parameters are compared within the defined error tolerances. Each part and its instances are matched and stored in an entity collection either by part/solid name or

parameter comparison. The first method is computationally more efficient than the second one. Comparison adds considerable time to the computation of the program.

After analyzing parts/solids, a similar procedure is followed for defeature, split tool, boolean and edge-offset operations. Each item of the parts dictionary has sample cell/face/edge/surface entity IDs for these operations. The program analyzes samples of each operation and stores comparison parameters in the dictionary. Then, sample solid collections generated in the analysis of parts step are called. Cells/faces/edges/surfaces of the same parts are compared and matched for the desired operation. Since both are 3D geometries, cells have the same comparison parameters of the solids. Faces/surfaces are compared for face area, face center distance to the parent geometry center, number of edges and number of vertices. Lastly, edge comparison parameters are edge length, number of vertices and edge center distance to the parent geometry center. The remaining operations of the dictonary are midsurface, mesh generation and element-property assigning. Midsurface operation converts a solid to a surface. Meshes can be generated in solid/cell/surface/face levels. If cell/face levels are preferred, sample entity IDs should be provided. Cell/face level definitions increase mesh generation time significantly. Element-property assigning operation defines each material property specified in the element properties dictionary of the previous step.

The next three steps include joint, bush and interaction dictonary operations. These features are defined by multiple parts and their sample entity IDs or geometry paths in their dictionaries. The custom design currently supports only an R-joint for the joints dictionary. Joints/bushes/interactions samples are defined by the user for two pairs. Each pair may include multiple geometry topologies (cell/face/edge). The pair comparison parameters are as sample geometry topology comparison parameters (explained in the previous paragraph), pair center distances and distance between pair center of one side to geometry topology of the other side. Then, part dictionary collections of the samples are called, compared, matched and joints/bushes/interactions are defined.

The program ends with single body operations and scenario operations. If an operation is not repeated, it is defined in single-body operations. The macro record tool of Apex can be used to easily generate a code for this section. In addition, scenarios (modal, linear static etc.) can be defined and run by the custom API design. However, simulation settings cannot be fully controlled by the provided API library. Template file can be used to define scenario settings before and FEA model can be implemented to the scenario in this step.

Figure 4.9. Flowchart of the custom API design for FEA simulation automation.

CHAPTER 5

CASE STUDY: DESIGN AND PARAMETRIC MODEL

Several deployable network prototypes are presented in Chapter 3 as a proof-of-concept. In this Chapter, the design and parametric model of a deployable tent mechanism are presented as a case study. The parametric model is based on one of the networks in Chapter 3 for a specific application.

The following sections begin with a definition of the case study. The spherical network to be designed as a dome-like deployable tent mechanism is explained. The design criteria are determined based on the data of equivalent products in the market. Afterwards, conceptual design alternatives are presented as a module of the network. One of the conceptual designs is selected for further development. Then, details of the designed network are explained. The parametric and CAD models are built simultaneously in Microsoft Excel® and Solidworks® via Parametric Design Integrator (PDI) presented in Chapter 4. A kinematic analysis of the designed module is performed and the developed model is verified. Finally, the specifications of the network are given.

5.1. Description of the Case Study

In the case study, the design and prototyping of a novel deployable tent mechanism is conducted. The system is called the "Yielding Unfolding Rigid-frame Tent" (YURT). The proposed system is planned to be the basis of a product family that can be customized for many applications, such as military command centers, post-disaster shelters, and other applications requiring rapidly deployable shelters based on the spherical network kinematic structure of Kiper (2016). This kind of deployable tent needs high deployment ratios, a short setup time, and a low number of people for setup and modular design. The designed mechanism is to be the frame of a tent. Parametric design of the mechanism are performed and implemented into software that allows the production of modular solutions according to the desired design criteria. Deployable tents can be transported as a single structure in folded form, which prevents the loss of parts

and assembly faults during the setup. Even if the tent frame is a mechanism with theoretically rigid links and joints, the links may be damaged due to mistakes during the setup. The study aims to design a system without these disadvantages and a competitive product for easy transportation and setup.

5.1.1. Spherical Network of the Case Study

The spherical network of Kiper (2016) is based on equilateral Special case-I modules. The network details are shown in Figure 5.1. In this design, each module compromises two Type-II Version-III networks introduced in Chapter 3 as half modules. A half module consists of four loops with a common central hub. Four new revolute joints connect two half-modules, which are not necessarily identical. In Figure 5.1 yellow and orange links represent upper and lower half-module links, respectively. For a spherical network, one of the modules, say the upper module, should be larger than the other module. Therefore, yellow links are longer than orange links in Figure 5.1. Four modules are connected to a central module other by 4-UU connections and four more modules are connected to the first four to form a spherical network. The resulting network comprising nine modules resembles a dome-like structure with four feet.

5.1.2. Products in the Market and Their Specifications

Available tent/shelter systems are investigated to decide the system's design criteria. Deployable Rapid Assembly Shelter (DRASH) (ReeveEMS, 2024) and As Soon As Possible (ASAP) (Deployed Logix, 2024) stand out among others. DRASH has composite titanite bars ($CaTiSiO_5$ - Calcium titanium nano silicate) and plastic hubs. Titanite is 2.7 times stronger than aluminum. The shelter has a modular design that allows for different sizes. An XB family varies from 1XB to 6XB according to the number of connected mid-modules (see Figure 5.2a). A DRASH 6XB shelter’s deployment is shown in Figure 5.2b. Starting from a folded form, the shelter first opens as a flat network, then takes a spherical shape by lifting with poles.

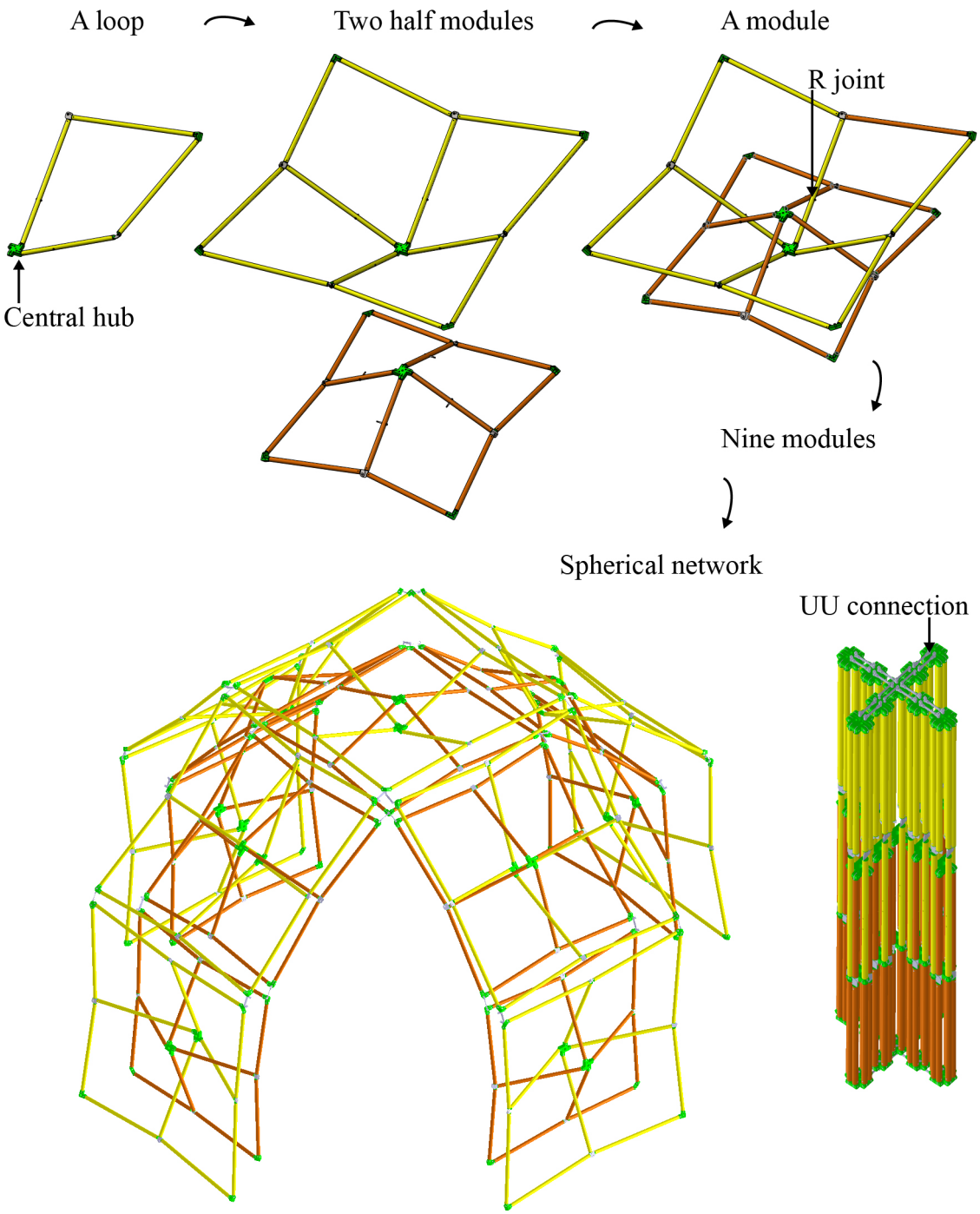


Figure 5.1. Spherical network of the case study from a loop to network.

The link lengths are intentionally produced longer than the required size in the system offered in DRASH shelters. This causes geometric incompatibilities. Thank to the bistable bars, the frame moves from a stable state to another state during the deployment. These types of deployable mechanisms are called bistable mechanisms, self-locking

mechanisms or clicking mechanisms. The phenomenon of changing between the bistable stats is called the snap-through effect. Tent frames that use this type of connection detail remain stable without additional locking mechanisms in the deployed form. They are in a stress-free state in the deployed and folded form. However, the links need to maintain linear elastic behavior during the deployment. The disadvantage of the snap-through effect is that links can be easily broken during installation. Figure 5.3 shows the load-displacement curve of a bistable structure. Li and Xu (2017) and Gantes (2018) explain the concept in detail. The sudden change in the shape of the structure is due to buckling. When a compressive vertical load is applied to one end of a beam that is fixed in the vertical axis, the deformation in the horizontal axis is called buckling (Hibbeler, 2010, 657).

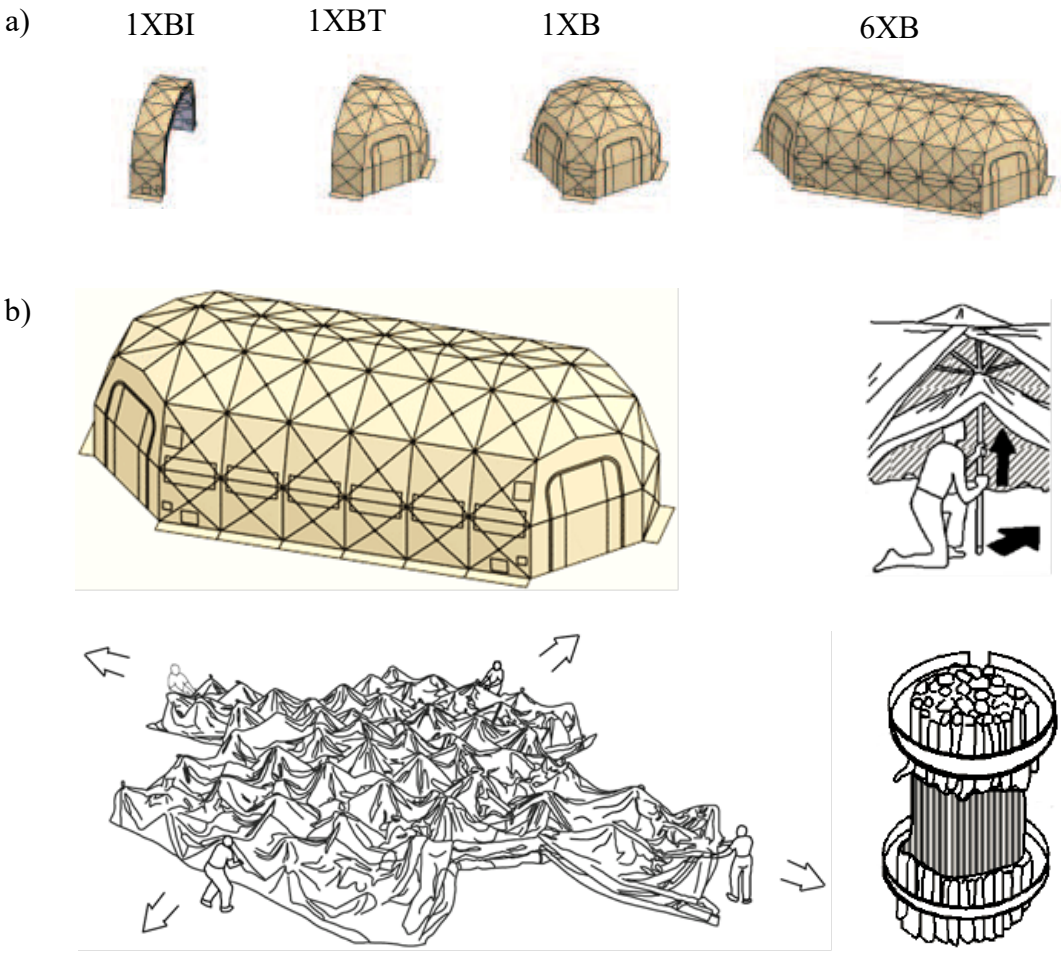


Figure 5.2. DRASH Shelter a) XB series shelters b) deployment of a DRASH-6XB shelter (Source: ReeveEMS, 2024).

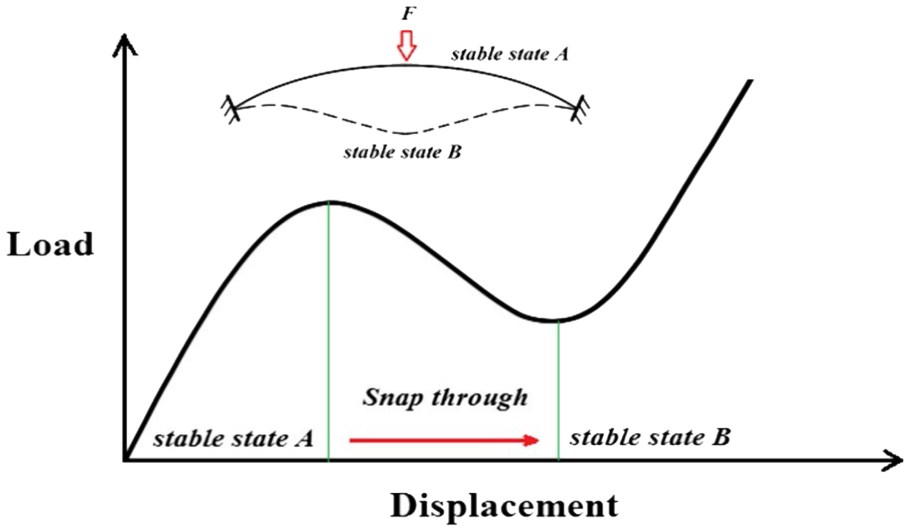


Figure 5.3. Snap-through effect (Source: Li and Xu, 2017).

Zeigler (1984) filed a patent on bistable deployable structures for the first time. However, Gantes et al. (1989) state that his invention is not stress-free in the deployed form and this results in lower load-bearing capacity. Later, successful examples of bistable mechanisms that are in a stress-free state in both deployed and folded form are developed by Krishnapillai and Zalewski (1985), Krishnapillai and Crawley (1986) and Gantes et al. (1989). In other words, these mechanisms are geometrically compatible in the folded and deployed forms but are geometrically incompatible during deployment. There are studies on design methodologies of scissor-like elements to obtain bistable mechanisms by Arnouts et al. (2019) and Zhao and Yu (2024). These structures are carefully designed as inner and outer loops that guarantee the snap-through effect by changing geometric properties (Arnouts et al., 2019). Roovers (2017) stated that there are two types of geometric incompatibilities in this type of deployable structure: 1) bending results in angular distortion or 2) perimeter deformation causes peripheral incompatibilities. Examples of these incompatibilities are shown in Figure 5.4.

ASAP shelter family also works with the snap-through effect. ASAP-12 is the smallest version of the family. Its main advantages are 1) easy setup, 2) short setup times and 3) less number of people are required for similar sizes. In addition to the setup time, the company defines the “time under canopy” specification as about 1 minute, which is remarkable compared to the DRASH setup. Deployment of an ASAP shelter is shown in

Figure 5.5. A comparison of the similar sizes DRASH-1XB and ASAP-12 is given in Table 5.1. Note that the snow-load capacity of DRASH is two times that of ASAP.

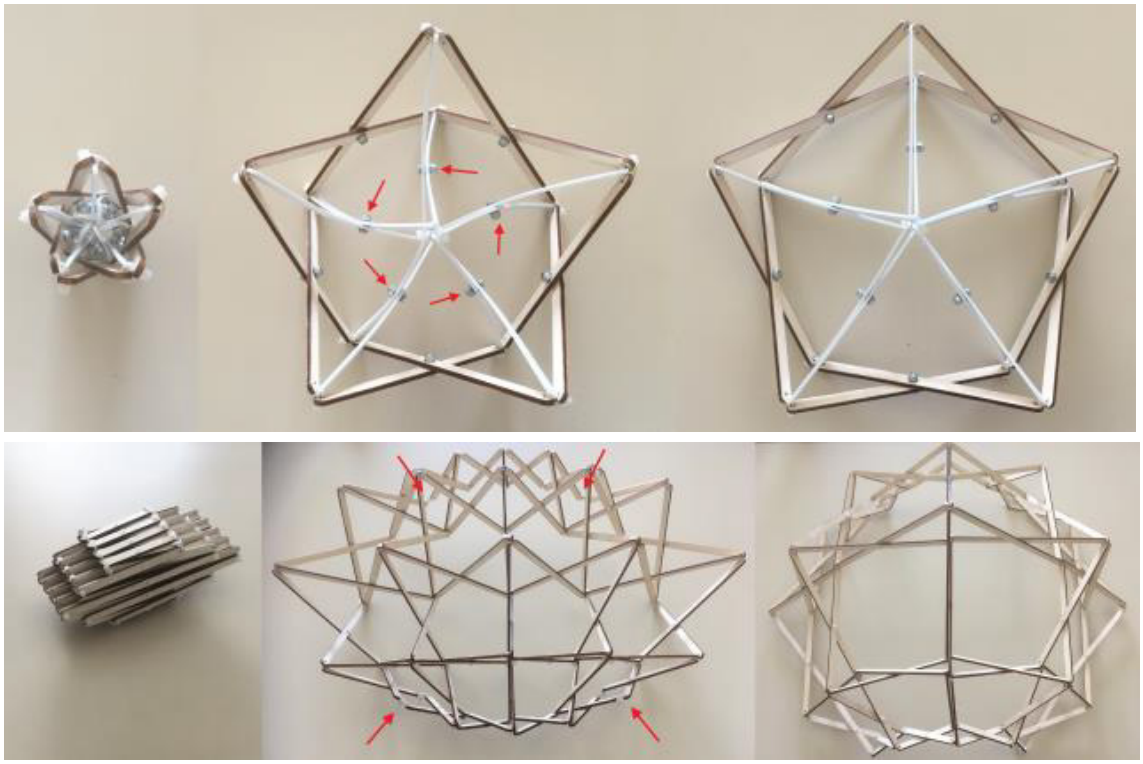


Figure 5.4. Geometric incompatibilities of bistable SLE structures a) angular distortions b) peripheral incompatibilities (Source: Roovers, 2017).

a)

b)

Figure 5.5. ASAP shelters (a) ASAP-12 (Source: Deployed Logix, 2024), (b) Deployment of a shelter (Source: snapshots taken from video by Deployed Logix (2021)).

Table 5.1. Comparison of the available products in the market.

	DRASH-1XB (ReeveEMS, 2024)	ASAP -12 (Deployed Logix, 2024)
Deployed Lengths Interior (L × W × H, m)	4.1 × 4.1 × 2.7	No information
Deployed Length Exterior (L × W × H, m)	4.7 × 4.7 × 3.2	3.66 × 4.88 × 2.44
Floor Area (m^2)	22.35	17.84
Usable Floor Area (m^2)	13	No information
Packed Volume (m^3)	In wrap: 0.8	In wrap: 0.363
Packed Dimensions (cm)	In wrap: 132 × 93 × 64	In wrap: 182 × 55 × 35
Deployment Ratio	≤ %2 (%1.1 calculated)	%1.7 (calculated)
Number of required personnel	4	2
Total Setup Time	15 minutes	5 minutes
Total Weight (kg)	110.2	72.6
Rain Load	5.08 cm/h free falling	5.08 cm/h free falling
Wind Load	Steady wind 89 km/h Gusts 105 km/h	Steady wind 89 km/h Gusts 105 km/h
Durability	50 erect/strike cycles	No information
Snow Load	48.82 kg/m^2 for 12 hr	24.41 kg/m^2

5.1.3. Design Criteria of the Case Study

Design criteria of the deployable tent mechanism are determined according to DRASH 1XB model (ReeveEMS, 2024). DRASH products are used by Turkish Armed Forces. DRASH shelters meet all U.S. Military requirements for soft-walled shelters and are tested according to test operation procedure TOP-10-2-175 (Everyspec, 2024) at U.S. Army's Aberdeen Test Center. Fabric, floor and climate control systems are kept out of the scope. Design criteria are given in Table 5.2.

Instead of a bistable mechanism, a geometrically compatible deployable mechanism is preferred. This type of deployable mechanism generates no internal stress (stress-free) during all stages of the deployment. It requires external bracing or locking mechanisms, as stated by Gantes et al. (1989) and Arnouts et al. (2019). Examples of

such works are given by Pinero (1965), Derus (1984), Escria (1985) and Nodskov and Thelander (1986).

Table 5.2. Design criteria.

Mechanism Type	Double-layer grid structure
Dof	1
Deployment Ratio	≤ 3 %
Floor Area in Deployed Form	≥ 13 m^2
Height in Deployed Form	≥ 2.7 m
Packed Volume	≤ 1 m^3
Mass	≤ 85 kg
Setup Time	≤ 15 mins
Number of People Required	4
Snow load	49 kg/m^2 for 12 hr

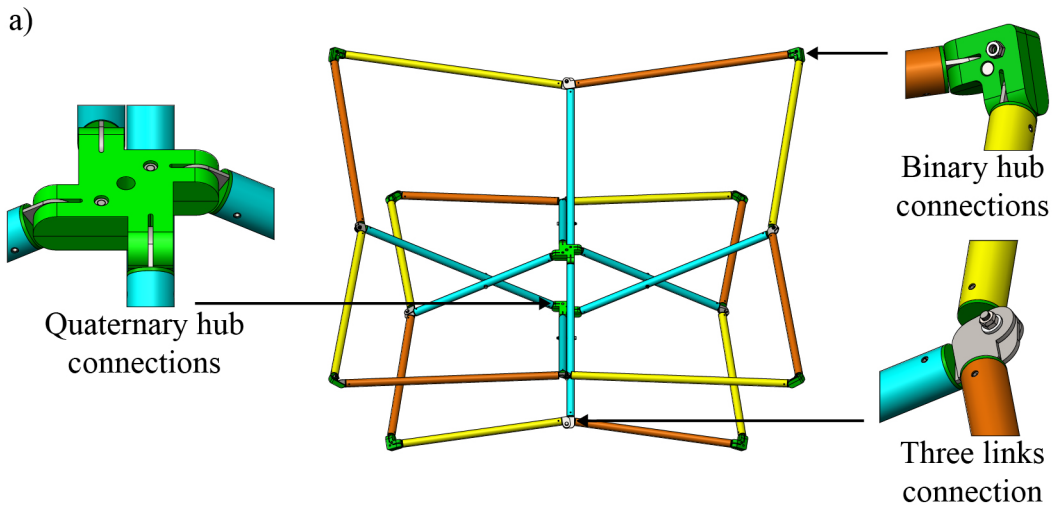
Prototyping is planned in two stages: a preliminary prototype and a final prototype. The preliminary prototype's design criterion is to have a statically stable tent frame in the deployed form, whereas the final prototype shoul withstand a snow load of 49 kg/m^2 for 12 hours in addition to being statically in the deployed form. The preliminary prototype has shorter link lengths than the final prototype, but with the same cross-sectional area of the links. The aim is to design the final prototype with the experience gained from the preliminary prototype joint design and other construction details. The first prototype's static stability is observed to understand necessary design revisions. The mechanism should be able to lock to achieve the desired stiffness in the deployed form in order to meet the snow-load criterion of the final prototype.

5.2. Conceptual Design Alternatives

After preparing the parametric design environment, parametric model including constructional design parameters is constructed. In parametric modeling of deployable structures, there are three options: 1) kinematic parameters, 2) geometry of cross-section

constructional parameters and 3) 3D geometry constructional parameters. The study involves a deployable structure with a limited number of types of parts. However, the number of parts is large and some of them are purchased. The third method more easily allows the adaptation of the model for different sizes of parts. Additionally, models with different sizes can be generated easily for FEA simulations.

Two conceptual design alternatives of a module are shown in Figure 5.6. Conceptual design-I is based on the Kiper (2016) patent model, while conceptual design-II is an alternative. Both designs have links consisting of a composite tube, plastic end caps, and sheet metal inserts. One end of each link is connected to two links with a bolt / headed pin at three link connections. The other end of each link is connected to binary or quaternary plastic hubs. Conceptual designs differ in hub connections. In conceptual design-I, the links are connected to the hubs with pins in a slot. However, in conceptual design-II, they are connected by a bolt / headed pin. Conceptual Design-I is suitable for mass production, where the hubs may be manufactured by plastic injection. It is also expected to be more robust due to the hub-link contection with a sandwich structure. Conceptual design-II is more suitable for an initial prototype production since it requires simpler machining operations during production. The single face hub-link contact results in a less rigid structure compared to double face contact in the other design.



(cont. on next page)

Figure 5.6. Conceptual design alternatives a) conceptual design-I based on Kiper (2016) b) conceptual design-II.

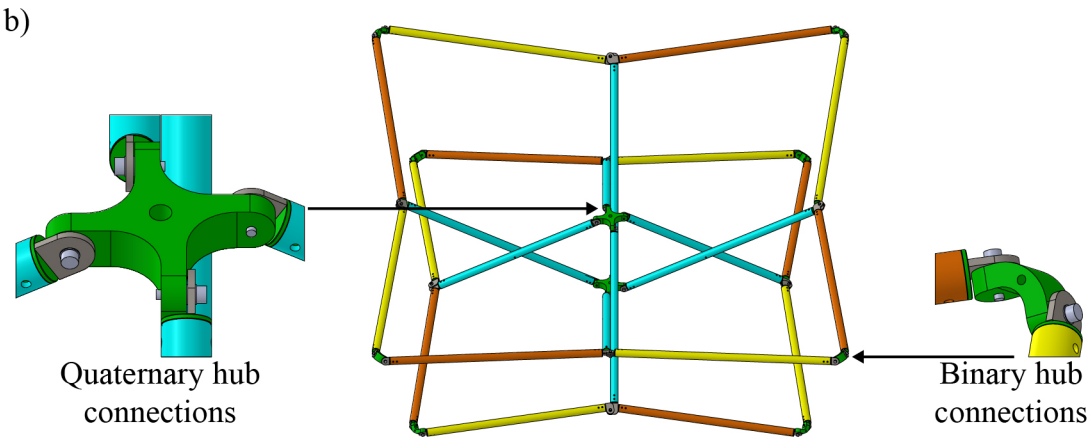


Figure 5.6 (cont.)

It is decided to proceed with conceptual design-I. However, due to budget limitations injection molding or milling of the plastic parts was not possible. So, it is planned to meet success load criterion in FEA simulations with mass produced material properties, but produce the parts with a 3D printer. After some trials it is observed that 3D printing is suitable for prototyping.

5.3. Design of the Spherical Network

The following sub-sections the design of both parts of both initial and final prototypes, but the network figures are based on the final prototype. The mechanical limits of module connection parts are not included in this Chapter. They are explained in Chapter 6.

Details of the spherical network are shown in Figure 5.7. A network is mainly composed of nine modules and four module connections between each connected module. A module is formed by seven types of bars ($Link_{1\text{-}7}$), a quaternary hub (Q) and a binary hub (B). Each half-module and the associated modified Altmann loops have three types of bars (orange, turquoise and yellow). The same colored links in upper and lower modules differ in link length, but they have the same type of connections. A dark blue link is designed as a mechanical stop for the modules at the deployed form. Module connections have two types of part combinations either C-SL or HC-SL (C: connector, HC: hollow connector and SL: sheet link). A ground-fixing system is designed to make a

connection between the corners of the half-modules at the bottom of the assembly and the ground.

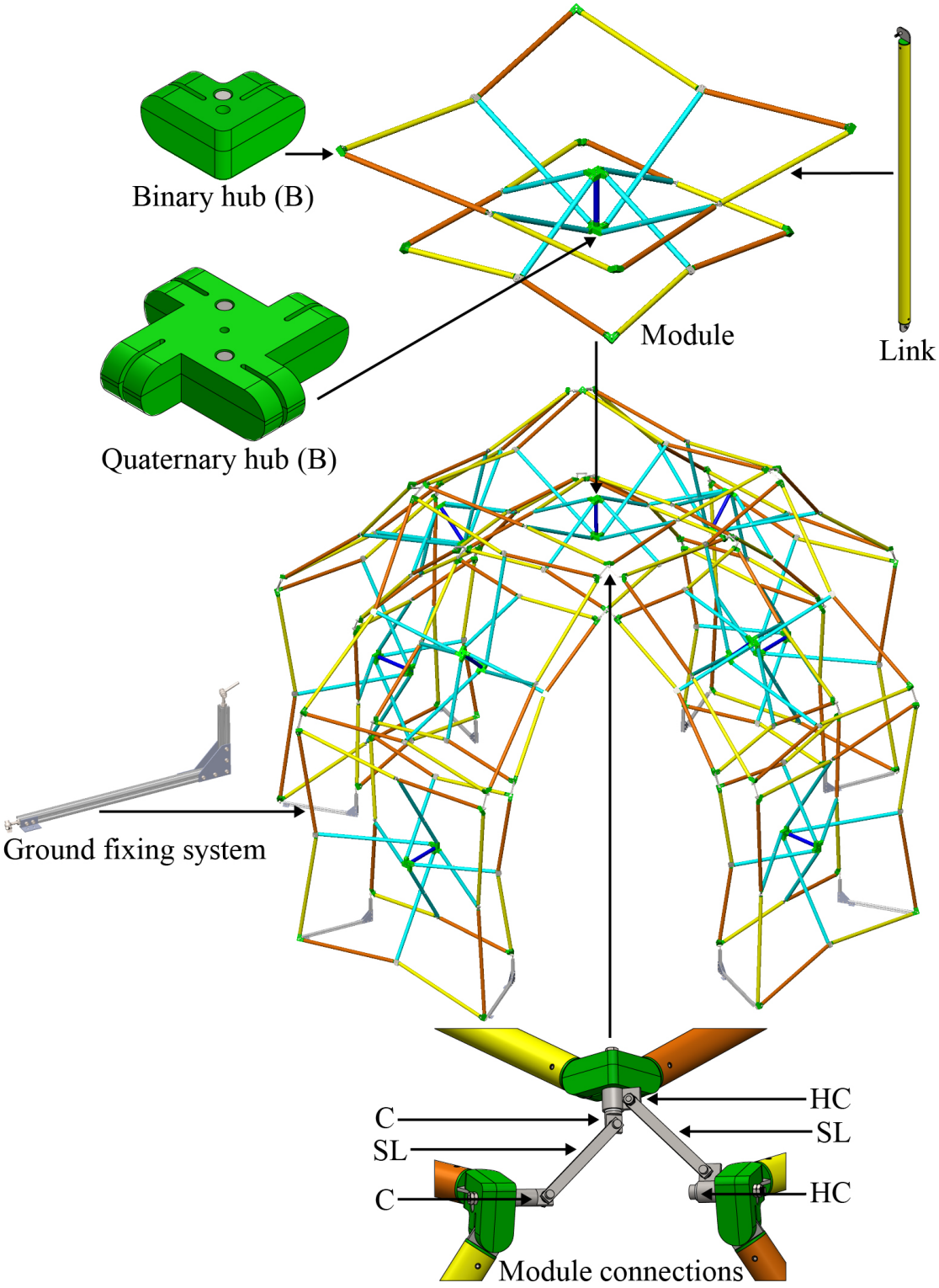


Figure 5.7. Design of the spherical network.

5.3.1. Design of the Module

The designs of $Link_{1-3,7}$ are shown in Figure 5.8. Links ($Link_{1-6}$) have the same type of assemblies but differ in part configurations. $Links_{1-6}$ have five types of components: tube (T_{1-6}), end cap (E_{1-2}), sheet metal end (S_{1-4}), coiled spring pin (SP_1) and slotted spring pin (SP_2). Sheet metal end (S_1) is used for either quaternary hub (Q) or binary hub (B) connections with coiled spring pin (SP_1). The other end of the links has sheet metal ends (S_{2-4}) meeting at the three links connection. $Link_7$ is the dark blue limiting link of a module. Instead of sheet metal ends (S_{1-4}) and end caps (E_{1-2}) it has end caps E_{3-4}, bolts Bo_{2-3} and nut N_1 in its design. It is fixedly assembled to quaternary hub of a smaller half-module in folded form and connected to the larger half-module's quaternary hub in the deployed form. Yellow, orange and blue tubes ($T_{2-3,5-7}$) are of the same type but have different lengths. Turquoise tubes ($T_{1,4}$) have an extra hole in the middle for half-modules connection, but in the same length as orange tubes ($T_{3,6}$), respectively in each half-module. End caps (E_{1-2}) have a rectangular slot for sheet metal ends (S_{1-4}) connections where S_{1-2} are used with E_1 and S_{3-4} with E_2. End caps (E_{3-4}) are designed to be used with bolts (Bo_{2-3}) and nut (N_1) and they have hexagon nut slots and circular holes for the nut and bolt connections. Each link has two slotted spring pins (SP_2) for the connection of end caps (E_{1-4}) and related parts.

Deciding the material for the tubes is important for the design in terms of weight and cost. Two alternatives are evaluated: carbon fiber and glass fiber-based tubes. Investigations showed that there is up to 100 times difference in fiber raw material cost between carbon fiber and glass fiber. As a result, glass fiber tubes produced by pultrusion method are prioritized. This production method is cost-effective with respect to filament winding and prepreg production methods. Glass-reinforced plastic (GRP) tubes are used in many areas such as water transportation lines, chemical plants, power plants, etc. The material is selected as plastic for the end caps that are to be connected to the end of the tubes. Sheet metal end (S) design is a low-cost steel solution and easy-to-manufacture material with laser cutting technology. Spring pins (SP_{1-2}), bolts (Bo_{2-3}) and nut (Nu_1) parts are available in the market.

Quaternary hub (Q) design is shown in Figure 5.9. The two main parts are female (Q_1) and male parts (Q_2). Two cylindrical extrusions on Q_2 fit into slots on Q_1 to prevent rotation between the parts. The bolt (Bo_1) and nut (N_2) fit into their slots on the

corresponding $Q_{1\text{-}2}$ parts. Four coiled spring pin (SP_1) connections are for the four R-joints between Q and $Links_{1,4}$. Coiled spring pin (SP_1) rotates in a slot on Q_2. A sandwich structure is obtained between $Q_{1\text{-}2}$ and sheet metal end (S_1) of the connected link ($Link_1$ or $Link_4$). Parts with different materials being in contact are advantageous in terms of friction. Thus, plastic hub-metal pin contact provides a suitable joint design between the hubs (B and Q) and $Links_{1\text{-}7}$. There is a central hole on quaternary hub ($Q_{1\text{-}2}$) for the connection with the mechanical stop link ($Link_7$).

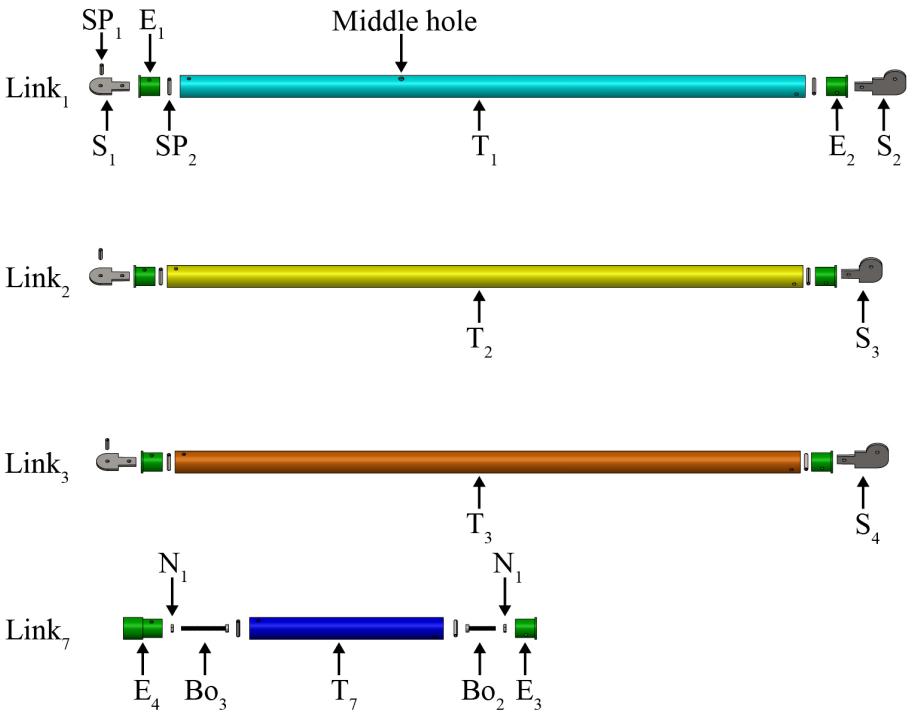


Figure 5.8. Design of $Links_{1\text{-}3,7}$.

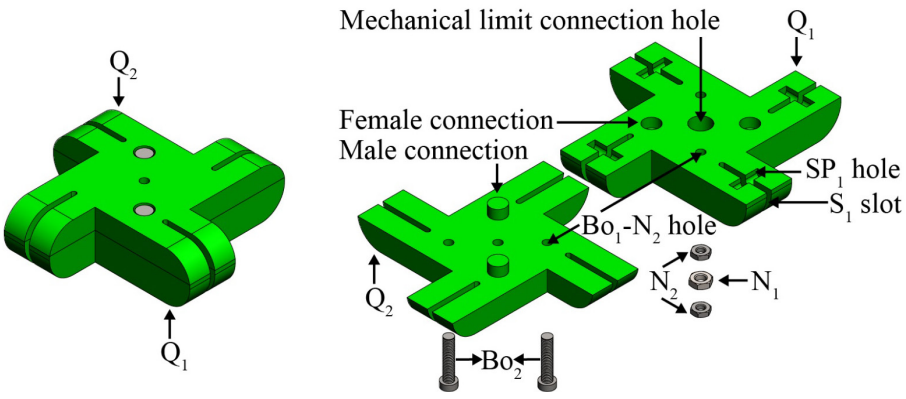


Figure 5.9. Design of the quaternary hub (Q).

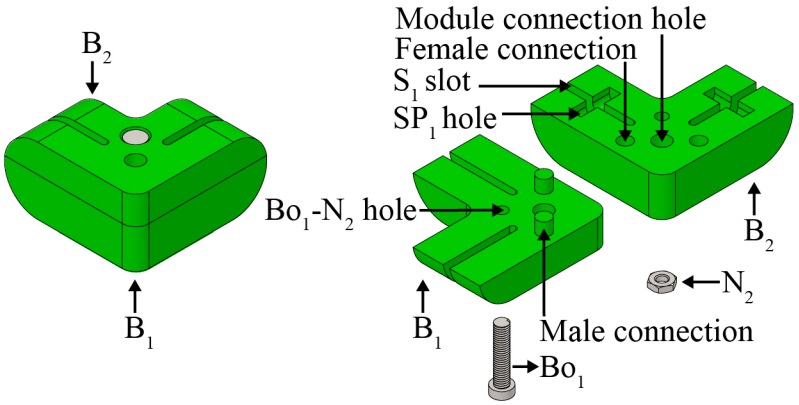


Figure 5.10. Design of the binary hub (B).

Figure 5.11 shows the details of the three links connection. Sheet metal end ($S_{2\text{-}4}$) on $Links_{1\text{-}3}$ or $Links_{4\text{-}5}$ are connected by bolt Bo_4, washer W_1 and nut N_2. Once the desired friction has been set with the first nut, it is planned to use a medium-strength thread locker product to prevent loosening in pins. Plastic washers (W_1) are placed between metal parts for low friction on the surface contacts with relative motion.

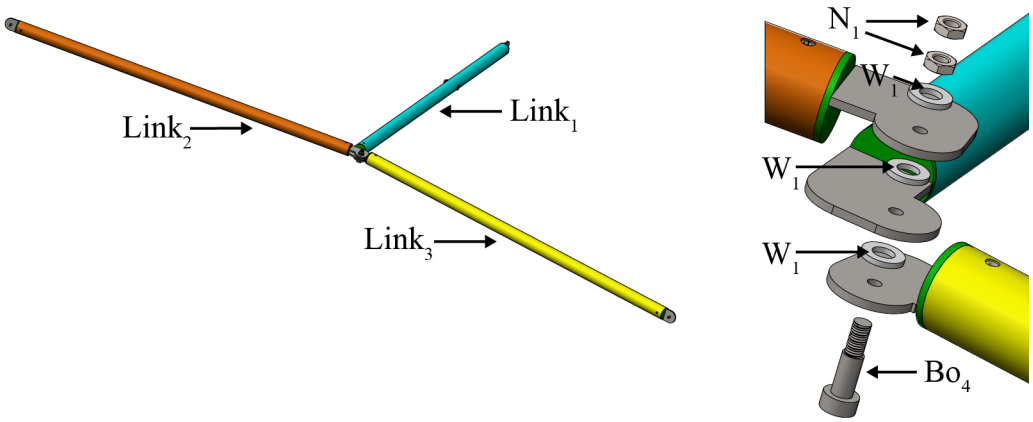


Figure 5.11. Design of the three links connection.

Connection of half-modules is shown in Figure 5.12. Larger and smaller half-modules are connected with four R-joints between $Links_{1,4}$. Bolts (Bo_6), washer (W_2) and nut (N_1) combination is used similar to three links connection.

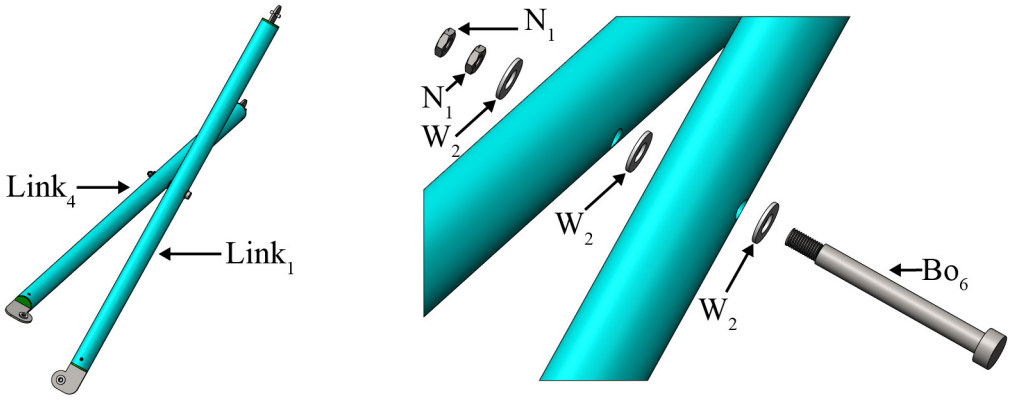


Figure 5.12. Design of the half-modules connection.

5.3.2. Design of the Module Connections

An assembly of three modules is shown in Figure 5.13a. Four UU chains connect two modules at each corner. Module connections have two types of assemblies. Type-I includes C and SL links (C: connector, SL: sheet link), whereas Type-II includes HC and SL (HC: hollow connector). Bolts (Bo_7) and washers ($W_{1,3}$) are used between C-SL or HC-SL couples. Type-I is identical in terms of sheet link (SL) between the pairs of lower and upper half-modules of adjacent modules. However, Type-II has different types of sheet links. Both types are required for four adjacent module connections of the central module in the network. Thus, the central module can be connect in two different directions. SL_2 link length of Type-I is defined according to the module parameters to have contact between binary hub (B) of the connected modules in folded form (Figure 5.13b). However, $SL_{4,6}$ lengths of Type-II are computed for the deployed form of the assembly. This results in slight height differences between connected modules in the folded form. The network has two Type-I connections and six Type-II connections between nine modules.

Three types of sheet links (SL), a single type of hollow connector (HC) and two types of connectors (C) are enough for the network. However, two different alternatives are developed during the work (see Figure 5.14). $SL_{1,3,5}$ are designed for HC_1 and $C_{1,3}$, and $SL_{2,4,5}$ for HC_2 and $C_{2,4}$. $C_{1\text{-}2}$ is designed to be used alone, but $C_{3\text{-}4}$ is on top of HC. A

bolt (Bo5) is used in Type-II connections instead of $C_{3,4}$ when the connected module is not the central module, i.e., the module is extended in a single direction by $HC_{1\text{-}2}$.

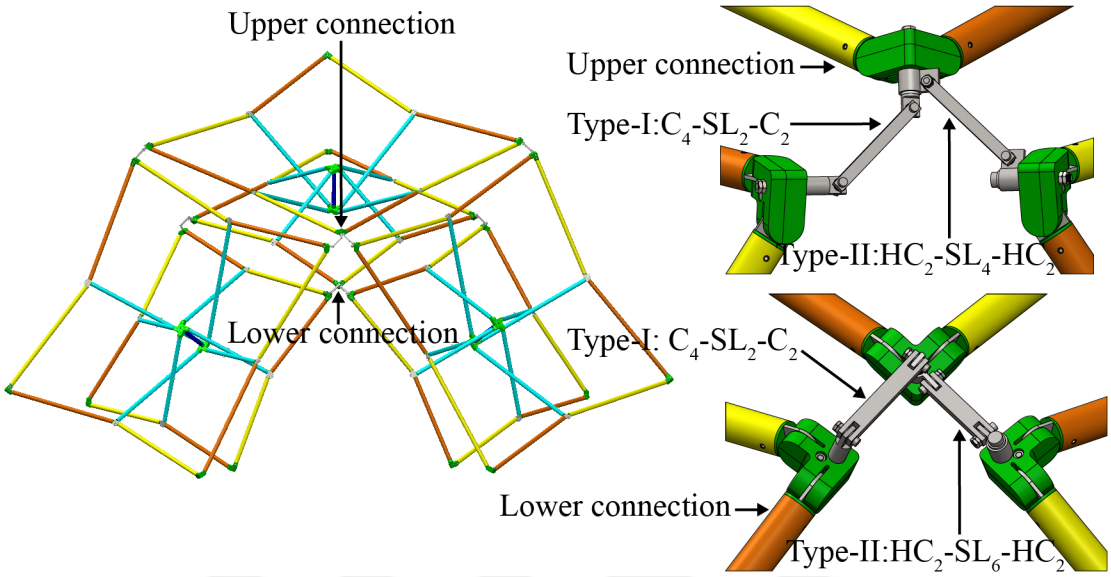


Figure 5.13. Three modules connection design a) deployed form, b) folded form.

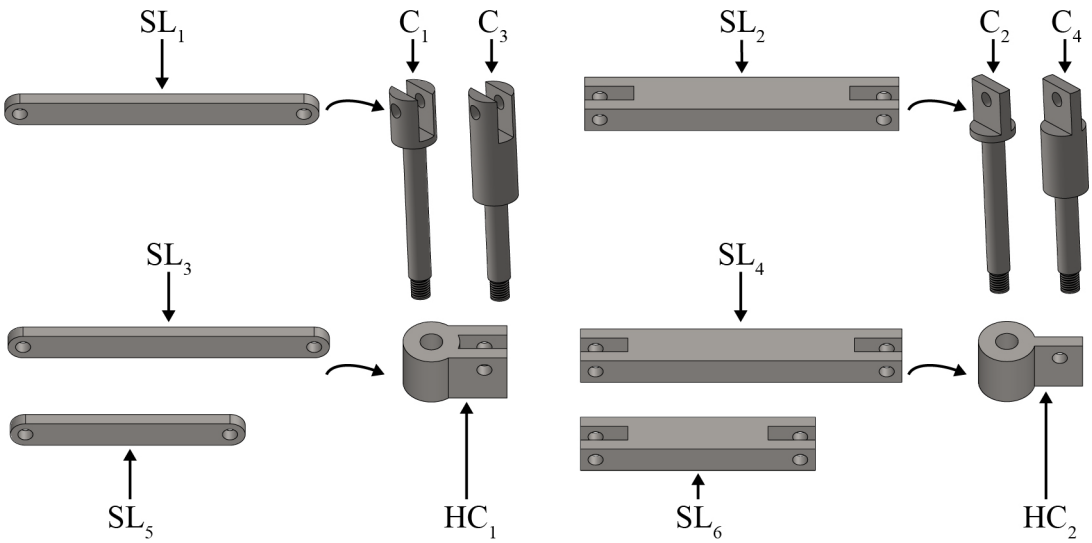


Figure 5.14. Module connection parts ($SL_{1\text{-}6}$, $HC_{1\text{-}2}$, $C_{1\text{-}4}$).

The deployed form of network is selected as the pose for the design of the lengths of $SL_{4,6}$. The pose is selected to have $\pi/4$ angle between planes defined by faces of quaternaru hub (Q) of the nine modules. Then, the modules resemble the nine faces of an elongated square cupola (Figure 5.15).

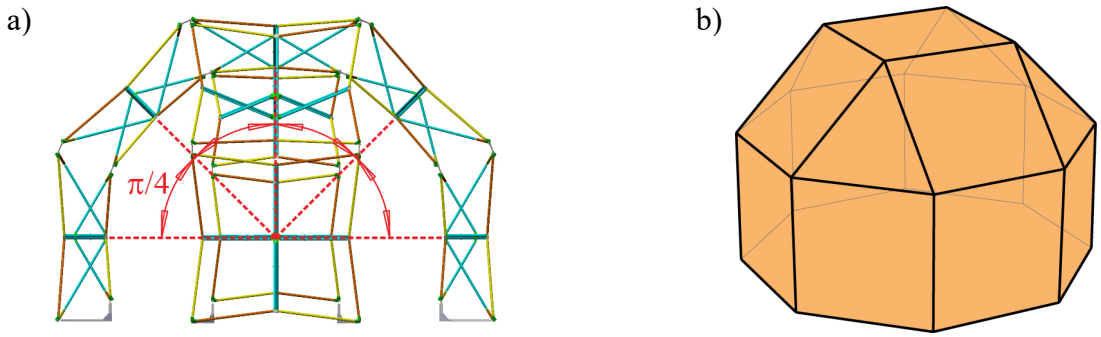


Figure 5.15. Network details a) nine-module network in the deployed form b) elongated square cupola.

5.3.3. Design of the Ground Fixing System

A ground fixing system is designed to fix the the deployed network on the ground (Figure 5.16). It connects to the B corners of the half-modules. Four pairs of fixing systems are used for the four modules on the ground. The system comprises swing bolts, aluminum profiles and connection components. The profile on the ground has two connection parts to fix the profile to the ground by four stakes. The design includes rotatable parts at two ends for a compliant connection with the modules. The parts of the fixing system are not included in the parametric model.

5.4. Parametric Model of the Parts

Parametric modeling starts with the definition of the DH parameters. Since equilateral special case-I loops are used and there are two types of them (half-modules), two parameters are required for each loop (a_1, d_1, a_2 and d_2) according to Equation (3.4).

Two half-modules need to be designed with different values for the bar lengths ($a_1 \neq a_1$), but the joint offsets are selected to be equal ($d_1 = d_2$). So, three different parameters are required (Table 5.3). Joint angles of upper and lower half-modules are the same due to the mirror symmetry between the half-modules. Tube ($T_{1\text{-}7}$) diameter ($\varnothing_{T1}$) is selected as an input parameter and joint offset (d_1) depends on it.

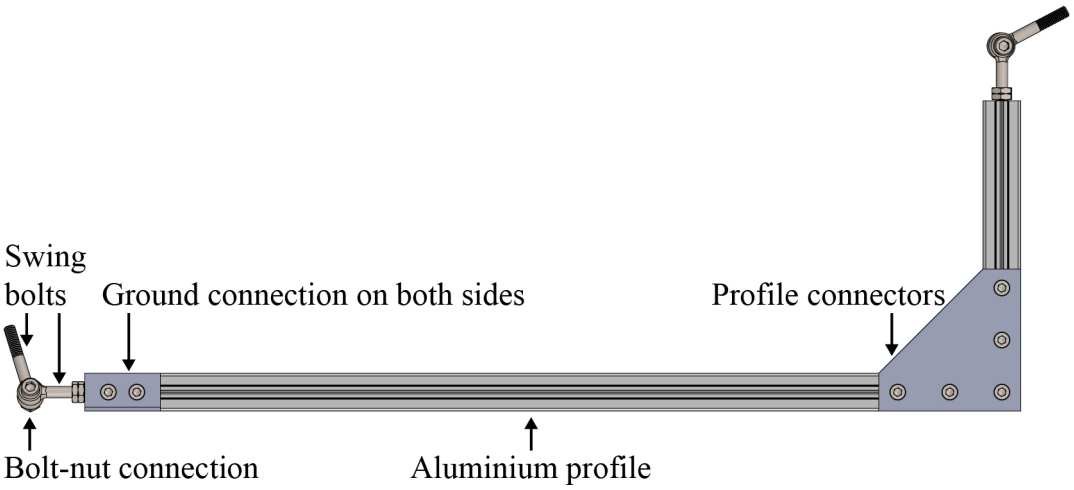


Figure 5.16. Design of ground fixing system.

Table 5.3. DH parameters of a module of the spherical network

Param.	Value	Description
d_1	$= 1.5\varnothing_{T1}$	Length: joint offset
a_1	Input	Length: link
a_2	Input	Length: link

If a part has more than one type, Solidworks' "configuration" tool is used to define parameters or properties according to the part type parameters. That allows keeping part types in a single file and easy change of parts without broken defined mates in assemblies if defined mate entities are not changed.

5.4.1. Bolt (Bo_{1-7}), Nut (N_{1-2}), Washer (W_{1-3}) and Spring Pin (SP_{1-2}) Parameters

Connection element parameters are used as input for the model. The idea here is to define the parameters of manufacturing parts according to the selected element. For example, if you decide to use a bolt, its parameters can be used to define the corresponding hole diameter, bolt head slot diameter, etc. Since the design only includes R joints and most of them have a bolt or coiled spring pin in their design, most of the kinematic element parameters can be defined depending on them. This approach allows us to vary solutions for different sizes and available products easily. Four main types of connection elements are used in the design: bolt (Bo_{1-7}), nut (Nu_{1-3}), washer (W_{1-3}) and spring pin (SP_{1-2}). $Bo_{1,4}$, N_1, W_1 and SP_{1-2} and their parameters are shown in Figure 5.17, where BV, FV, IV, RV and TV subscripts represent the back, front, isometric, right, and top views, respectively.

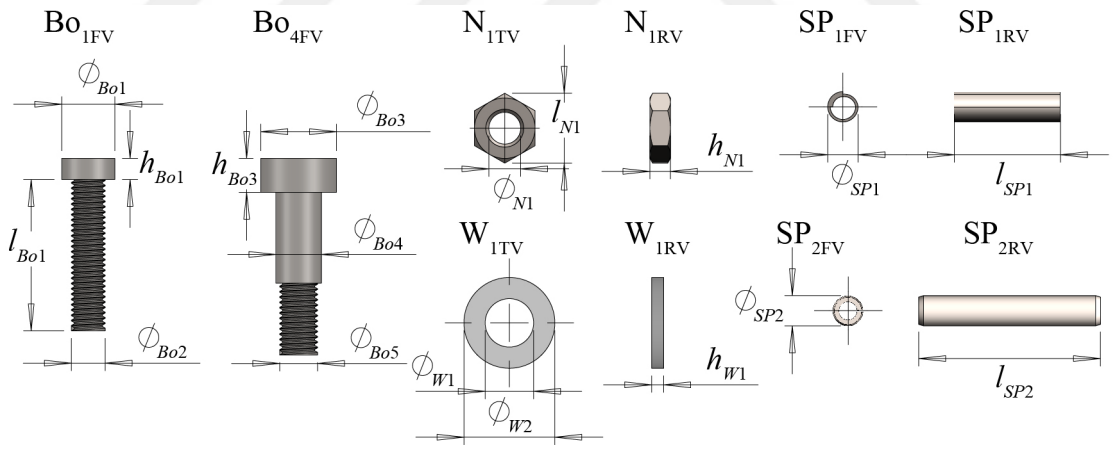


Figure 5.17. Parameters of machine elements: bolts ($Bo_{1,4}$), nut (N_1), washer (W_1), spring pins (SP_{1-2}).

Bolts $Bo_{1,4}$ are shown in Figure 5.17. Bo_1 is a standard bolt with a fully threaded shaft, whereas Bo_4 is a partially threaded shoulder bolt. Both types are configurations of a single part in the CAD environment. The unthreaded shaft part extrusion feature is suppressed for Bo_1. A new extrusion feature is defined for Bo_4. New configurations are defined based on these two configurations. Bo_{1-7} parameters are listed in Table 5.4. Bo_{2-}

$_3$ and Bo_{5-7} are variations of Bo_1 and Bo_4, respectively. A bolt is parametrized by its bolt head, shaft part and thread part heights and diameters. Two color codes are used in parameter tables throughout the Chapter for the sake of clarity: 1) if a parameter is altering the same dimension in other configurations, then the "description" column cell backgrounds have the same color (for example, the orange color in the description column represents bolt head height for different bolts by four different parameters ($h_{Bo1\text{-}4}$)); 2) if a configuration set of the "Used in" column is defined multiple times, then cell backgrounds have the same color. The usage of bolts is as follows: Bo_1 connects the two halves of B and Q hubs, Bo_2 and Bo_3 connect Q to $E_{3\text{-}4}$ at the mechanical stop, Bo_4 forms three links connection, Bo_5 is used at HC-B connection, Bo_6 connects half-modules and Bo_7 connects SL to C and HC.

Nuts ($N_{1\text{-}2}$) are defined according to their diameter ($\varnothing_{N1\text{-}2}$) and height ($h_{N1\text{-}2}$). Parameters are listed in Table 5.5. Two variations are used in the design. N_1 and N_2 are compatible with $Bo_{2\text{-}6}$ and $Bo_{1,7}$, respectively. Both are chosen from counter-nut standards. This thin type allows it to be used as a couple to prevent loosening when the nuts are not in a slot. A single nut is used when a nut is in a slot, such as N_2 parts in B and Q assembly. Nuts and spring pins ($SP_{1\text{-}2}$) are obtained from the software toolbox.

Table 5.4. Parameters of $Bo_{1\text{-}7}$.

Parameter	Value	Description	Used in
h_{Bo1}	Input	Height: bolt head	Bo_1
h_{Bo2}		Height: bolt head	$Bo_{2\text{-}3}$
h_{Bo3}		Height: bolt head	$Bo_{4\text{-}6}$
h_{Bo4}		Height: bolt head	Bo_7
$\varnothing_{Bo1}$		Diameter: bolt head	Bo_1
$\varnothing_{Bo2}$		Diameter: thread part	Bo_1
$\varnothing_{Bo3}$		Diameter: bolt head	$Bo_{2\text{-}3}$
$\varnothing_{Bo4}$		Diameter: thread part	$Bo_{2\text{-}3}$
$\varnothing_{Bo5}$		Diameter: bolt head	$Bo_{4\text{-}6}$
$\varnothing_{Bo6}$		Diameter: shaft part	$Bo_{4\text{-}6}$

(cont. on next page)

Table 5.4 (cont.)

$\varnothing_{Bo7}$		Diameter: thread part	$Bo_{4\text{-}6}$
$\varnothing_{Bo8}$		Diameter: bolt head	Bo_7
$\varnothing_{Bo9}$		Diameter: shaft part	Bo_7
$\varnothing_{Bo10}$		Diameter: thread part	Bo_7
l_{Bo1}		Length: thread part	Bo_1
l_{Bo2}		Length: thread part	$Bo_{2\text{-}3}$
l_{Bo3}		Length: thread part	$Bo_{2\text{-}3}$
l_{Bo4}		Length: shaft part	Bo_4
l_{Bo5}		Length: shaft part	Bo_5
l_{Bo6}		Length: shaft part	Bo_6
l_{Bo7}		Length: thread part	$Bo_{4\text{-}6}$
l_{Bo8}		Length: shaft part	Bo_7
l_{Bo9}		Length: thread part	Bo_7

Table 5.5. Parameters of $N_{1\text{-}2}$.

Parameter	Value	Description	Used in
h_{N1}	Input	Height: nut	N_1
h_{N2}		Height: nut	N_2
$\varnothing_{N1}$		Diameter: nut	N_1
$\varnothing_{N2}$		Diameter: nut	N_2
l_{N1}		Length: nut	N_1
l_{N2}		Length: nut	N_2

Washers ($W_{1\text{-}3}$) are defined by height ($h_{1\text{-}3}$), inner ($\varnothing_{W1,3,5}$) and outer ($\varnothing_{W2,4,6}$) diameters. Parameters are listed in Table 5.6. W_1 is used with $Bo_{4\text{-}5}$ and $C_{1\text{-}2}$. W_2 is placed between two tubes of a half-module connected by Bo_6. In addition, W_3 is compatible with Bo_7, which connects $SL_{1\text{-}6}$ to $HC_{1\text{-}2}$ or $C_{1\text{-}2}$.

Table 5.6. Parameters of $W_{1\text{-}3}$.

Parameter	Value	Description	Used in
h_{W1}	Input	Height: washer	W_1
h_{W2}	Input	Height: washer	W_2
h_{W3}	Input	Height: washer	W_3
$\varnothing_{W1}$	Input	Diameter: washer inner	W_1
$\varnothing_{W2}$	Input	Diameter: washer outer	W_1
$\varnothing_{W3}$	Input	Diameter: washer inner	W_2
$\varnothing_{W4}$	Input	Diameter: washer outer	W_2
$\varnothing_{W5}$	Input	Diameter: washer inner	W_3
$\varnothing_{W6}$	Input	Diameter: washer outer	W_3

As the last types of machine elements, two types of spring pins parameters defined in Table 5.7. SP_1 and SP_2 represent coiled and slotted spring pins, respectively. Both of them are represented by two parameters: diameter ($\varnothing_{SP1\text{-}2}$) and length ($l_{SP1\text{-}2}$). SP_1 forms a kinematic element on the links. SP_1 is attached to S_1 of the links. These links are connected to hubs B or Q by an R joint. SP_2 is used to fix link components P, E and S together.

Table 5.7. Parameters of $SP_{1\text{-}2}$.

Parameter	Value	Description	Used in
$\varnothing_{SP1}$	Input	Diameter: coiled spring pin	SP_1
$\varnothing_{SP2}$	Input	Diameter: slotted spring pin	SP_2
l_{SP1}	Input	Length: coiled spring pin	SP_1
l_{SP2}	Input	Length: slotted spring pin	SP_2

5.4.2. Tube ($T_{1\text{-}7}$) Parameters

24 tubes (T) are used in each module. Top, front and right views (TV, FV, RV) of T_1 are shown in Figure 5.18. The part is defined by seven parameters in the CAD environment. Tube diameter $\varnothing_{T1}$ and thickness t_{T1} are shown in the top view. Two perpendicular holes (SP_2 connection holes) defined by hole diameter $\varnothing_{T2}$ and distance to the near end l_{T8} are shown in the front and right views. The top part of these views represents the S_1–E_1–SP_2 side at the Q connection, whereas the bottom part is the S_2–E_1–SP_2 side at the connection to other bars. The two holes at the terminals of the tube provide connections to E_1. A third hole with diameter ($\varnothing_{T3}$) at a distance l_{T7} from the Q side is used for the connection of the half-modules (Bo_6 connection hole).

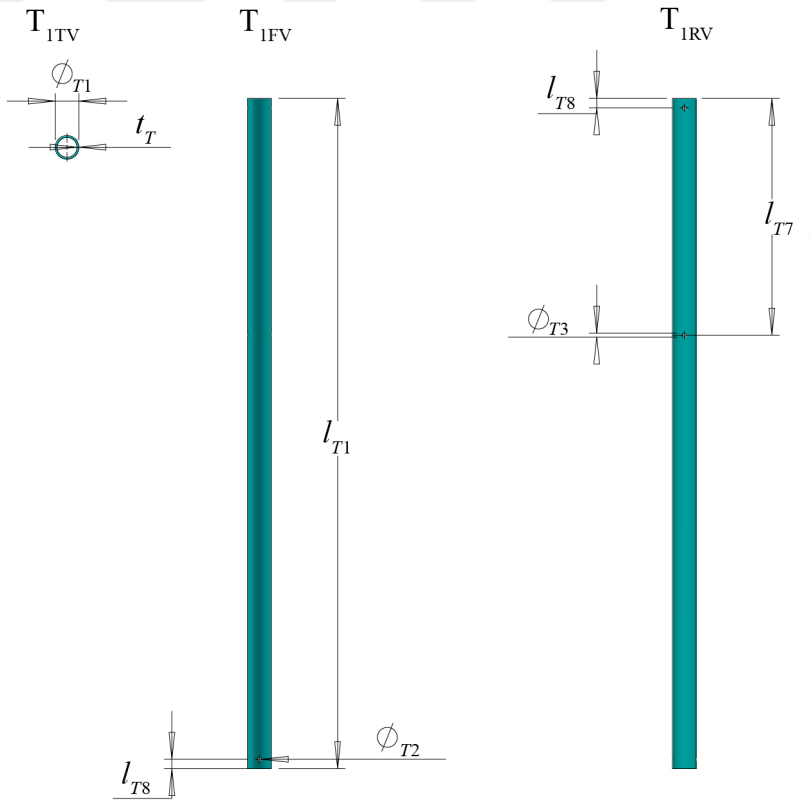


Figure 5.18. Parameters of tube (T_1) in three views.

Seven configurations of tube ($T_{1\text{-}7}$) parameters are listed in Table 5.8. $T_{1\text{-}3}$ and $T_{4\text{-}6}$ are used in the larger and smaller half-modules, respectively, where T_7 is the mechanical

limit tube. Tubes are parameterized using 13 parameters. There are five tube lengths and l_{P1} changes for different length tubes ($l_{T1\text{-}5}$). $l_{T1\text{-}5}$ is defined to satisfy DH parameters either a_1 or a_2. T_1–T_3 and T_4–T_6 pairs have the same length but differ by half-modules connection hole feature status. $T_{1,4}$ have half-modules connection holes (feature unsuppressed). Half-modules connection hole distance (l_{T7}) is defined according to the smaller or lower half-module tube's (T_4) half length and an extra distance l_{T6} to prevent collusion in the deployed form. Half-modules connection's effective link length is defined as l_{T9} for kinematic computations. The point definition will be given in the kinematic analysis section. The desired tolerance values ($\kappa_{1\text{-}2}$) are added to the SP_2 hole ($\varnothing_{T2}$) and Bo_6 hole ($\varnothing_{T3}$). Hole diameters are defined depending on SP_2 and Bo_6 diameters. SP_2 and Bo_6 are machine elements. Their tolerances are defined by the standard of the selected element. Corresponding hole diameters ($\varnothing_{T2\text{-}3}$) are defined as larger than them by $\kappa_{1\text{-}2}$. The same type of logic is followed by adding κ other terms in the parameters of other parts. Mechanical limit tube (T_7) length l_{T5} is computed by point 45's x position (P_{45x}) at deployed form (@DF). Since the deployed form input angle is measured in the CAD environment, the parameter is defined according to that pose after measuring the input angle.

Table 5.8. Parameters of $T_{1\text{-}7}$.

Parameter	Value	Description	Used in
t_{T1}	Input	Thickness: T	$T_{1\text{-}7}$
$\varnothing_{T1}$	Input	Diameter: T (outer)	$T_{1\text{-}7}$
$\varnothing_{T2}$	$= \varnothing_{SP2} + \kappa_1$	Diameter: SP_2 connection hole	$T_{1\text{-}7}$
$\varnothing_{T3}$	$= \varnothing_{Bo6} + \kappa_2$	Diameter: Bo_6 hole at half-modules connection	$T_{1,4}$
l_{T1}	$= a_1 - 2l_{E2} - l_{S2} - l_{S6}$	Length: T link	$T_{1,3}$
l_{T2}	$= a_1 - 2l_{E2} - l_{S8} - l_{S2}$	Length: T link	T_2
l_{T3}	$= a_2 - 2l_{E2} - l_{S2} - l_{S6}$	Length: T link	$T_{4,6}$
l_{T4}	$= a_2 - 2l_{E2} - l_{S8} - l_{S2}$	Length: T link	T_5

(cont. on next page)

Table 5.8 (cont.)

l_{T5}	$= P_{45x} - \varnothing_{T1} - l_{E2} - l_{E7}$ @DF	Length: T link	T_7
l_{T6}	Input	Length: extra distance for half-modules connection hole	$T_{1,4}$
l_{T7}	$= (l_{T3} + l_{S6} - l_{S2}) / 2 - l_{T6}$	Length: distance of half-modules connection hole from the parallel hole end	$T_{1,4}$
l_{T8}	Input	Length: SP_2 connection hole distance from the near end	$T_{1,4}$
l_{T9}	$= l_{T7} + l_{S2} + l_{E2}$	Length: effective link length of half module connection	$T_{1,4}$

5.4.3. Sheet Metal End ($S_{1\text{-}4}$) Parameters

Sheet metal end parts are defined in four configurations as $S_{1\text{-}4}$ (Figure 5.19). S_1 is common at each link end for hub connections. $S_{2\text{-}4}$ changes at the other end of the links for three link connections. 13 parameters of $S_{1\text{-}4}$ are listed in Table 5.9. The thickness of the sheet metal ends ($S_{1\text{-}4}$) is t_{S1}. l_{S6} and l_{S7} parameters differ between configurations where $l_{S6} = l_{S2}$ and $l_{S7} = 0$ for S_1. These two parameters are defined to satisfy the associated link length and joint offset, respectively. Two different extrusions are defined in the CAD environment to vary S_1 and S_2. Extrusion features are suppressed/unsuppressed for S_1 or $S_{2\text{-}4}$. S_3 and S_4 are derived from the S_2 configuration. l_{S6} and l_{S7} of S_2 are changed to obtain S_3 and S_4 for desired l_{S8}-l_{S2} and l_{S6}-l_{S2} pairs, respectively. S parts are designed depending on the tube diameter ($\varnothing_{T1}$). Each configuration's outer side width (l_{S3}) equals $\varnothing_{T1}$, where the inner or slot width (l_{S2}) starts by half of it. Slot part width decreases by an angle (β_{S1}) to provide easiness in assembly. $SP_{1\text{-}2}$ connection hole diameters ($\varnothing_{S1\text{-}3}$) are computed according to the related connector diameters ($\varnothing_{Bo4}$ and $\varnothing_{SP1\text{-}2}$) and desired tolerance values ($\kappa_{1\text{-}2}$). $\kappa_{1\text{-}2}$ are used from tube (T) parameters because they have the same parameters: κ_1 for SP_2 connection and κ_2 for $Bo_{4,6}$ shaft diameters. An extra distance (l_{S5}) added to S dimensions directly affects the related link lengths ($l_{S6\text{-}8}$) to prevent collusion due to manufacturing errors at three links connection ($S_{2\text{-}4}$). Since S_3 is the end of Q-connected links 1 and 4, l_{S5} is added along the tube axis direction. However, the extra distance is added in the horizontal direction of $S_{2,4}$. Connection details of the half-modules affect S_2

parameter l_{S7}. Due to deployment requirements, Three links connection distance l_{S7} need to be offset by half of the tube diameter ($\varnothing_{T1}/2$) and washer height ($h_{W2}/2$).

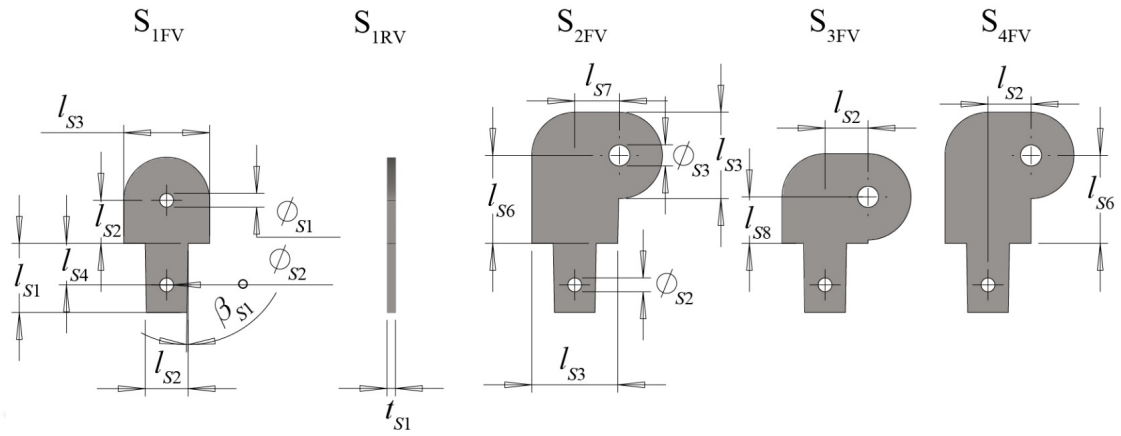


Figure 5.19. Parameters of sheet metal ends ($S_{1\text{-}4}$).

Table 5.9. Parameters of $S_{1\text{-}4}$.

Parameter	Value	Description	Used in
t_{S1}	Input	Thickness: S sheet metal	$S_{1\text{-}4}$
$\varnothing_{S1}$	$= \varnothing_{SP2}$	Diameter: SP_1 connection hole	S_1
$\varnothing_{S2}$	$= \varnothing_{SP2} + \kappa_1$	Diameter: SP_2 connection hole	$S_{1\text{-}4}$
$\varnothing_{S3}$	$= \varnothing_{Bo6} + \kappa_2$	Diameter: Bo_4 connection at three links connection	$S_{2\text{-}4}$
β_{S1}	Input	Angle: slot angle for fitting	$S_{1\text{-}4}$
l_{S1}	Input	Length: S inner length in the slot	$S_{1\text{-}4}$
l_{S2}	$= \varnothing_{T1} / 2$	Length: S link / Length: S link / Length: inner width	S_1 / $S_{3\text{-}4}$ / $S_{1\text{-}4}$
l_{S3}	$= \varnothing_{T1}$	Length: S outer width	$S_{1\text{-}4}$
l_{S4}	$= l_{T8} + l_{E2}$	Length: distance of SP_2 connection hole to outer side begin	$S_{1\text{-}4}$
l_{S5}	Input	Length: extra distance to prevent interference of different links S and E components at three links connection	$S_{2\text{-}4}$
l_{S6}	$= l_{S3} + l_{S5}$	Length: S link	$S_{2,4}$
l_{S7}	$= (\varnothing_{T1} + h_{W2})/2$	Length: S link	S_2
l_{S8}	$= l_{S2} + 2l_{S5}$	Length: S link	S_3

5.4.4. End Cap ($E_{1\text{-}4}$) Parameters

End caps are defined in four configurations ($E_{1\text{-}4}$). $E_{1\text{-}2}$ are used at link ends with $S_{1\text{-}4}$, while $E_{3\text{-}4}$ are used for the mechanical limit part. $E_{1,4}$ shown in Figure 5.20, and the associated parameters are listed in Table 5.10. Parameters of $E_{1\text{-}4}$..Each E has an SP_2 connection hole ($\varnothing_{E1}$), outer ($\varnothing_{E2}$) and inner ($\varnothing_{E3}$) diameters. $E_{1\text{-}2}$ have rectangular slots for S connection ($S_{1\text{-}2}$ are connected to E_1 and $S_{3\text{-}4}$ are connected to E_2). Slot height (h_{E3}), width (l_{E4}) and depth (l_{E3}) are defined. Slot length is increased by κ_3. $l_{E1\text{-}2}$ defines the inner and outer lengths of $E_{1\text{-}3}$. The location of the slot is defined by h_{E1} for E_1 and h_{E2} for E_2. E_2 is used at upper and lower sheets ($S_{3\text{-}4}$) in three links connection at two orientations. Washer height (h_{W1}) and sheet thickness (t_{S1}) are used to define h_{E2}. $E_{3\text{-}4}$ is defined for the mechanical limit part and has a central hole instead of the slots with rectangular cross-section in $E_{1\text{-}2}$. $\varnothing_{E4}$ is the continuous circular hole diameter for $Bo_{2\text{-}3}$ connections on $E_{3\text{-}4}$. $\varnothing_{E5}$ defines a circular hole diameter and hexagon socket length on $E_{3\text{-}4}$. A nut (N_1) and bolt head of $Bo_{2\text{-}3}$ are placed in this circular hole and a hexagon slot. The circular part depths are l_{E6} for Bo_2 heads, and the hexagon socket part depth is l_{E8} for the nut (N_1) slot in E_3. These parameters vary in order, as l_{E8} for Bo_3 and l_{E10}. Outer length l_{E1} is varied for E_4 as l_{E7} and defined by a new variable that has the ability to change the mechanical limit length from one side of $Link_7$ on physical prototypes.

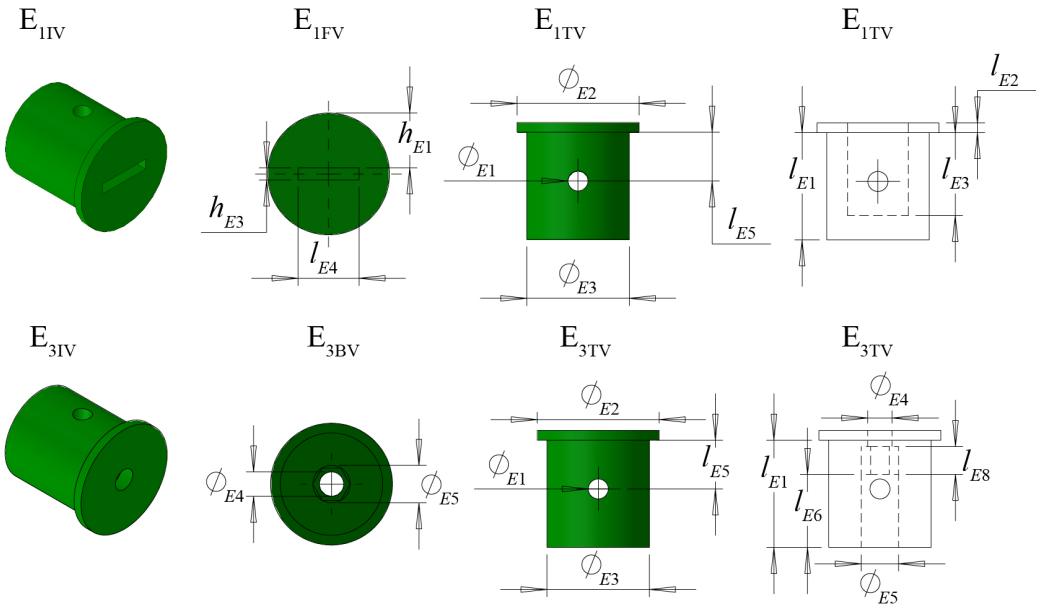


Figure 5.20. Parameters of end caps ($E_{1\text{-}3}$).

Table 5.10. Parameters of $E_{1\text{-}4}$.

h_{E1}	$= (\varnothing_{P1} - t_{S1}) / 2$	Height: S slot distance	E_1
h_{E2}	$= h_{E1} + t_{S1} + h_{W1}$	Height: S slot distance	E_2
h_{E3}	$= t_{S1}$	Height: S slot	$E_{1\text{-}2}$
$\varnothing_{E1}$	$= \varnothing_{P2}$	Diameter: SP_2 connetion hole	$E_{1\text{-}3}$
$\varnothing_{E2}$	$= \varnothing_{T1}$	Diameter: E outer	$E_{1\text{-}3}$
$\varnothing_{E3}$	$= \varnothing_{T1} - 2t_{P1}$	Diameter: E inner	$E_{1\text{-}3}$
$\varnothing_{E4}$	$= \varnothing_{Bo4}$	Diameter: $Bo_{2\text{-}3}$ connection hole	E_3
$\varnothing_{E5}$	$= l_{N1}$	Diameter: $Bo_{2\text{-}3}$ head hole / Length: N_1 hole	E_3
l_{E1}	Input	Length: E inner	$E_{1\text{-}3}$
l_{E2}	Input	Length: E outer	$E_{1\text{-}3}$
l_{E3}	$= l_{S1} - l_{E2} + \kappa_3$	Length: S slot length	$E_{1\text{-}3}$
l_{E4}	$= l_{S2}$	Length: S slot width	$E_{1\text{-}2}$
l_{E5}	$= l_{T8}$	Length: SP_2 connection hole distance	$E_{1\text{-}3}$
l_{E6}	$= l_{E8} + h_{N1} + h_{Bo2}$	Length: Bo_2 head hole depth	E_3
l_{E7}	Input	Length: E outer	E_4
l_{E8}	$= l_{E1} - l_{E5} + \varnothing_{SP2} / 2$	Length: N_1 hole depth	E_3
l_{E9}	$= l_{E10} + h_{N1} + h_{Bo2}$	Length: Bo_3 head hole depth	E_4
l_{E10}	$= l_{E1} - l_{E5} + \varnothing_{SP2}/2 + l_{E7} - l_{E2}$	Length: N_1 slot depth	E_4

5.4.5. Binary Hub ($B_{1\text{-}2}$) and Quaternary Hub ($Q_{1\text{-}2}$) Parameters

Binary hub ($B_{1\text{-}2}$) parameters are shown in Figure 5.21 and listed in Table 5.11. $B_{1\text{-}2}$ are connected to links by coiled spring pin (SP_1) and sheet metal end (S_1). The height of the configurations ($h_{B1\text{-}2}$) is defined to have SP_1 in B_1. A slot length (l_{B1}) is defined according to the thickness of S_1 with a tolerance κ_6. The wall thicknesses of $B_{1\text{-}2}$ is t_{B1}. $2t_{B1} + l_{B1}$ is defined according to the tube diameter ($\varnothing_{T1}$). SP_1 connection slot is defined by the diameter ($\varnothing_{B1}$) and κ_4 in it. The remaining wall length after SP_1 is l_{B7}. Joint axis distances are parametrized by $l_{B2,4,6}$. l_{B2} is defined to satisfy the joint offset d_1. An extra

distance l_{B3} is used in l_{B4} computations to prevent interference between B parts and E_1 of the connected link. $B_{1\text{-}2}$ are connected together by two female-male connections and a bolt nut connection (Bo_1-N_2). Male-female connections are parametrized by $\varnothing_{B5}$ and l_{B9}. N_2 is placed in a slot of B_1 ($\varnothing_{B3}$ diameter with κ_5 and depth l_{B8}), where Bo_1 head is in B_2 ($\varnothing_{B8}$ diameter and depth l_{B10}). The thread part hole is defined by $\varnothing_{B4}$ on both parts. Another hole is defined by $\varnothing_{B2}$ for one of the Bo_6 or $C_{1\text{-}6}$ in module connections. Two diameters are used for inner and outer fillets ($\varnothing_{B6\text{-}7}$).

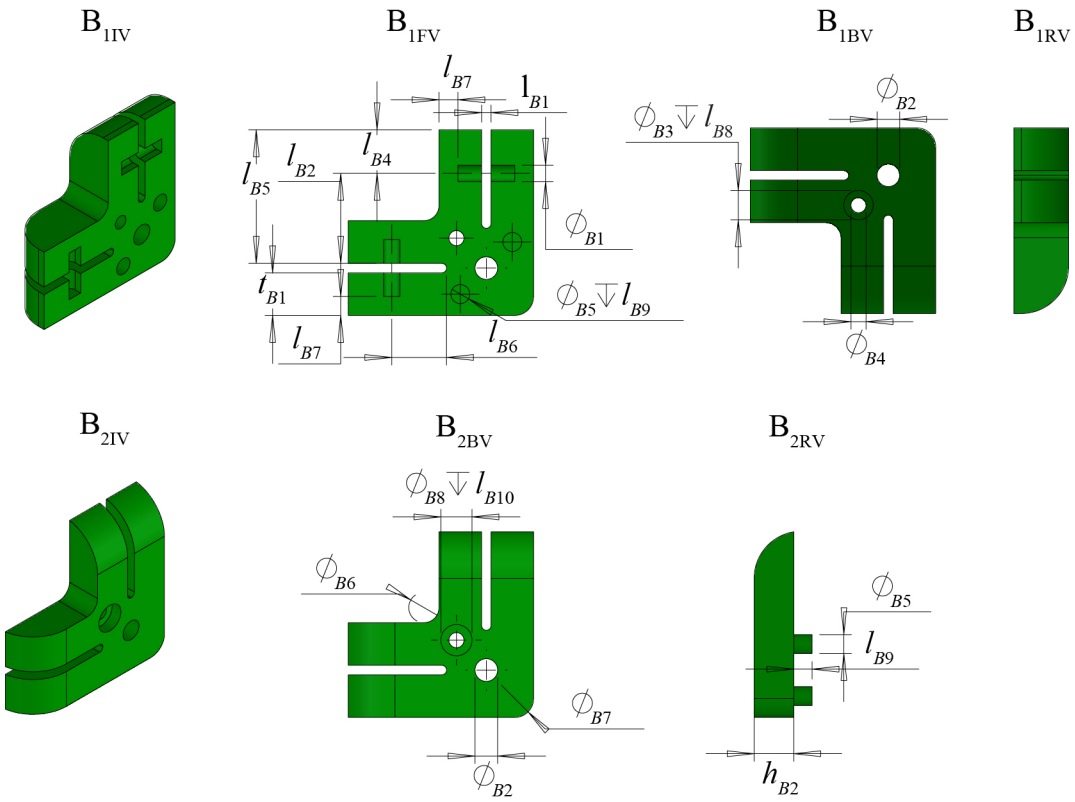


Figure 5.21. Binary hubs ($B_{1\text{-}2}$) parameters.

Table 5.11. Parameters of $B_{1\text{-}2}$.

Parameter	Value	Description	Used in
t_{B1}	$= (\varnothing_{T1} - t_{S1})/2$	Thickness: wall	$B_{1\text{-}2}$

(cont. on next page)

Table 5.11 (cont.)

h_{B1}	$= (\varnothing_{T1} + \varnothing_{SP1})/2$	Height: B	$B_{1\text{-}2}$
h_{B2}	$= \varnothing_{T1} - h_{B1}$	Height: B	B_2
$\varnothing_{B1}$	$= \varnothing_{SP1} + \kappa_4$	Diameter: SP_1 connection hole	$B_{1\text{-}2}$
$\varnothing_{B2}$	$= \varnothing_{C2}$	Diameter: Bo_6 and $C_{1\text{-}4}$ hole at module connection	$B_{1\text{-}2}$
$\varnothing_{B3}$	$= l_{N2} + k_5$	Diameter: N_2 hole	B_1
$\varnothing_{B4}$	$= \varnothing_{Bo2}$	Diameter: Bo_1 hole	$B_{1\text{-}2}$
$\varnothing_{B5}$	Input	Diameter: male/female connection	$B_{1\text{-}2}$
$\varnothing_{B6}$	Input	Diameter: inner fillet	$B_{1\text{-}2}$
$\varnothing_{B7}$	Input	Diameter: outer fillet	$B_{1\text{-}2}$
$\varnothing_{B8}$	$= \varnothing_{Bo1} + \kappa_5$	Diameter: Bo_1 head hole	B_2
l_{B1}	$= t_{S1} + \kappa_6$	Length: S_1 slot	$B_{1\text{-}2}$
l_{B2}	$= d_1 - (l_{S3} + l_{B1})/2$	Length: joint offset	$B_{1\text{-}2}$
l_{B3}	Input	Length: extra distance between to prevent interference	$B_{1\text{-}2}$
l_{B4}	$= l_{S2} - l_{B3}$	Length: distance between the end of the hub and the related joint axis	$B_{1\text{-}2}$
l_{B5}	$= l_{B2} + l_{B4}$	Length: largest vertical distance of the inner wall to the adjacent long side	$B_{1\text{-}2}$
l_{B6}	$= \varnothing_{T1} - l_{S2} + 2l_{B3}$	Length: distance between the beginning of the hub slot and the joint axis	$B_{1\text{-}2}$
l_{B7}	$= (\varnothing_{T1} - l_{S2})/2 - \kappa_6$	Length: length of the wall after SP_1	$B_{1\text{-}2}$
l_{B8}	$= \varnothing_{T1} - l_{B10} - l_{Bo1} - h_{N2}$	Length: N_2 hole depth	B_1
l_{B9}	$= h_{B1}/3$	Length: male/female connection	$B_{1\text{-}2}$
l_{B10}	$= h_{Bo1} + \kappa_7$	Length: Bo_1 head hole depth	B_2

Quaternary hub ($Q_{1\text{-}2}$) parameters are shown in Figure 5.22 and listed in Table 5.12. Some of the parameters ($h_{Q1\text{-}2}$, $\varnothing_{Q1,2\text{-}6}$, $l_{Q3\text{-}4,6\text{-}9}$) are analogous to $B_{1\text{-}2}$ parameters, and selected equivalent of the related parameters. This time hub parts are connected by two Bo_1-N_2 couples and two male-female connections. There is a new middle hole defined by different diameters ($\varnothing_{Q2,8}$). A different size nut (N_1) is placed in the $\varnothing_{Q8}$ hole for $Bo_{2\text{-}3}$

connections of the mechanical limit link, and $\varnothing_{Q2}$ is defined according to $Bo_{2\text{-}3}$ thread part diameters. l_{Q1} ensures joint offset (d_1). l_{Q2} is defined according to the center distance between the two connected half-module tubes. Due to the washer (W_2) at the half-modules connection, double wall thickness (t_{Q1}) plus S_1 slot length (l_{Q4}) should be smaller than the tube diameter ($\varnothing_{T1}$). Additionally, l_{Q10} ensures that a tube can fit outside of Q in the folded form.

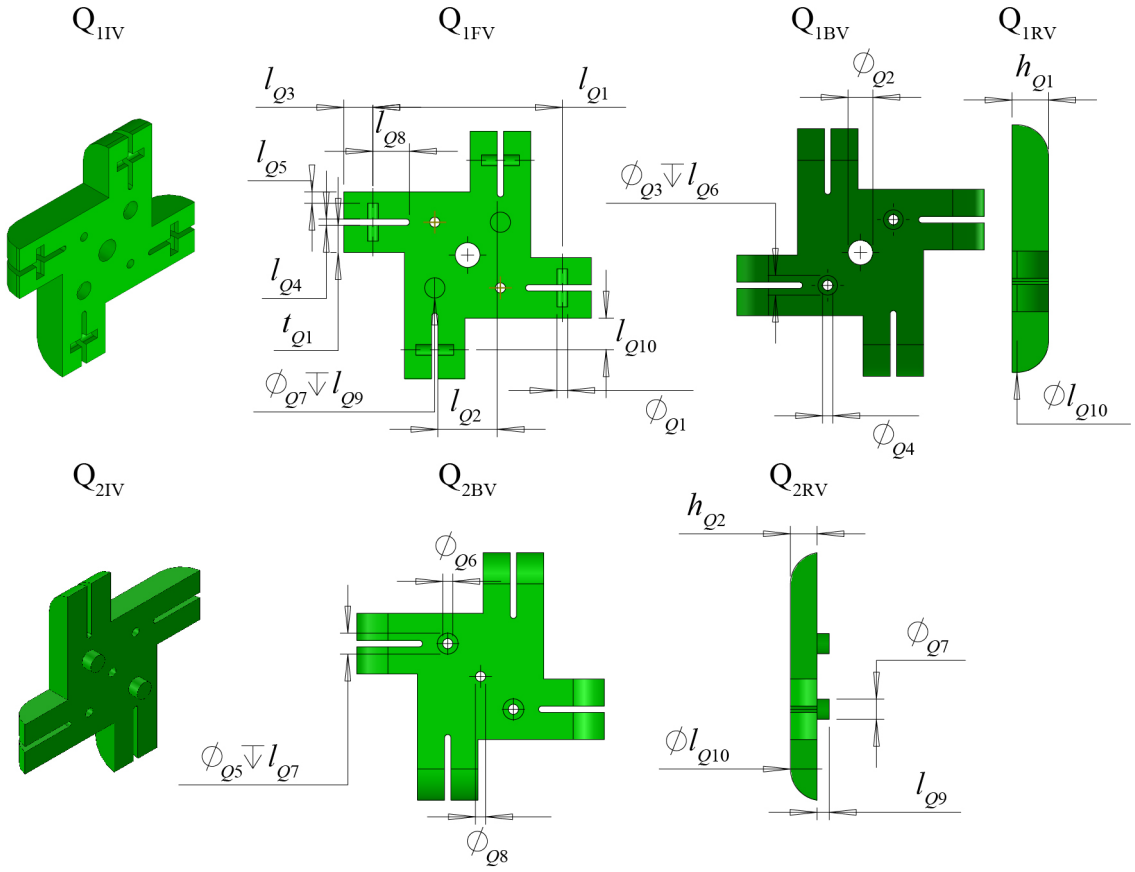


Figure 5.22. Parameters of quaternary hubs ($Q_{1\text{-}2}$).

Table 5.12. Parameters of $Q_{1\text{-}2}$.

Parameter	Value	Description	Used in
t_{Q1}	$= (l_{Q1} - 2l_{Q10} - l_{Q2} - 2t_{S2})/2$	Thickness: wall	$Q_{1\text{-}2}$

(cont. on next page)

Table 5.12 (cont.)

h_{Q1}	$= h_{B1}$	Height: Q	Q_1
h_{Q2}	$= h_{B2}$	Height: Q	Q_2
$\varnothing_{Q1}$	$= \varnothing_{B1}$	Diameter: SP_2 connection hole	$Q_{1\text{-}2}$
$\varnothing_{Q2}$	$= \varnothing_{Bo4}$	Diameter: $Bo_{2\text{-}3}$ hole at the middle of the part	Q_1
$\varnothing_{Q3}$	$= \varnothing_{B3}$	Diameter: N_2 hole	Q_1
$\varnothing_{Q4}$	$= \varnothing_{B4}$	Diameter: Bo_1 hole	Q_2
$\varnothing_{Q5}$	$= \varnothing_{B5}$	Diameter: Bo_1 head hole	Q_2
$\varnothing_{Q6}$	$= \varnothing_{B4}$	Diameter: Bo_1 hole	Q_2
$\varnothing_{Q7}$	Input	Diameter: male/female connection	$Q_{1\text{-}2}$
$\varnothing_{Q8}$	$= l_{N1}$	Diameter: N_1 hole	Q_1
l_{Q1}	$= 2d_1$	Length: distance between two joint axes	$Q_{1\text{-}2}$
l_{Q2}	$= \varnothing_{T1} + h_{W2} - t_{S1}$	Length: center distance between two connected half-module tubes	$Q_{1\text{-}2}$
l_{Q3}	$= l_{B4}$	Length: distance between the end of the hub and the related joint axis	$Q_{1\text{-}2}$
l_{Q4}	$= l_{B1}$	Length: S_1 slot	$Q_{1\text{-}2}$
l_{Q5}	$= (2t_{Q1} + l_{Q4} - l_{SP1})/2$	Length: length of wall after SP_2	Q_1
l_{Q6}	$= l_{B8}$	Length: N_2 hole depth	Q_1
l_{Q7}	$= l_{B10}$	Length: Bo_1 head hole depth	Q_2
l_{Q8}	$= l_{B6}$	Length: distance between the beginning of the hub slot and the joint axis	$Q_{1\text{-}2}$
l_{Q9}	$= l_{B9}$	Length: male/female connection	$Q_{1\text{-}2}$
l_{Q10}	$= \varnothing_{T1} / 2$	Length: joint axis distance to the near wall side	$Q_{1\text{-}2}$

5.4.6. Sheet Link ($SL_{1\text{-}6}$) Parameters

Sheet links ($SL_{1\text{-}2}$) are shown in Figure 5.23. These are two design alternatives for module connection. SL links are connected to hollow connectors ($HC_{1\text{-}2}$) and connectors

($C_{1\text{-}4}$). Here a sandwich structure can be obtained between the connected pairs. SL_1 voonects HC_1 and $C_{1,3}$. SL_2 connects HC_2 and $C_{2,4}$. A bolt (Bo_7) and two washers (W_3) are used between SL and the connected components. A sketch is drawn in the CAD environment that contains $SL_{1\text{-}2}$ parameters. Two different extrusions are defined for two configurations. Since three lengths of sheet links are required, $SL_{3\text{-}6}$ are defined as variations of these two. $SL_{1\text{-}6}$ parameters are listed in Table 5.13. Thicknesses of $SL_{1\text{-}2}$ are defined by three values ($t_{SL1\text{-}3}$). $t_{SL1,3}$ are sheet link thicknesses of $SL_{1\text{-}2}$, respectively. t_{SL2} is defined according to the connected washer (W_3) height. A Bo_7 connection hole is designed at each end where hole diameter $\varnothing_{SL1}$ and the associated distance l_{SL1} are defined. HC connection slot depth of SL_2 is designated as l_{SL2}. Link length is defined as l_{SL3} for $SL_{1\text{-}2}$ and changes to l_{SL4} and l_{SL5} for $SL_{3\text{-}4}$ and $SL_{5\text{-}6}$, respectively. $SL_{1\text{-}2}$ is used in both upper and lower $C_{1\text{-}4}$ connections of the modules. Its length is decided according to the $B_{1\text{-}2}$ parameters. The minimum link length is computed for a deployed form condition to have the B parts of adjacent modules in contact in folded form. SL_2 is used in upper and SL_3 is in lower $HC_{1\text{-}2}$ connections. Their lengths ($l_{SL4\text{-}5}$) are computed in deployed form according to the $HC_{1\text{-}2}$ and $C_{1\text{-}4}$ parameters. The computation is explained at the end of Section 5.4.7.

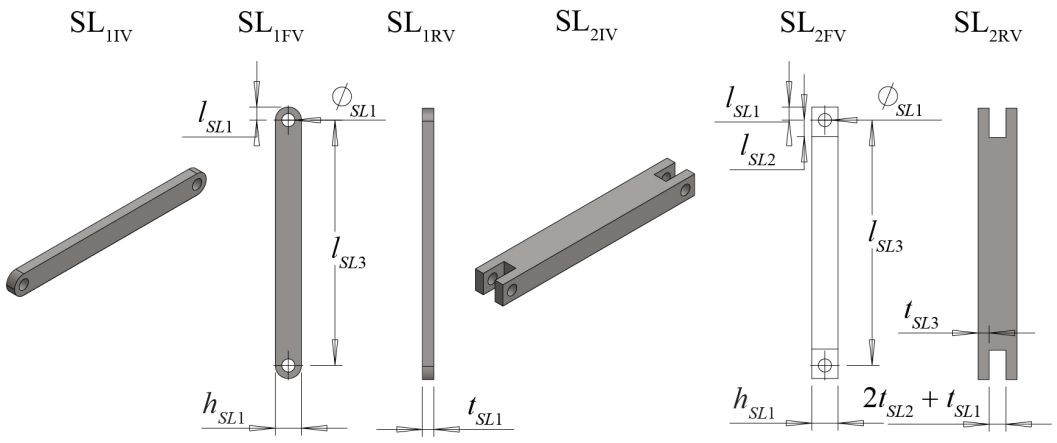


Figure 5.23. Parameters of sheet links ($SL_{1\text{-}2}$).

Table 5.13. Parameters of $SL_{1\text{-}6}$.

Parameter	Value	Description	Used in
h_{SL1}	Input	Height: SL	$SL_{1\text{-}6}$

(cont. on next page)

Table 5.13 (cont.)

t_{SL1}	Input	Thickness: HC connection wall	$SL_{1\text{-}6}$
t_{SL2}	$= h_{W3}$	Thickness: HC connection wall separation distance for W_3	$SL_{2,4,6}$
t_{SL3}	Input	Thickness: HC connection wall	$SL_{2,4,6}$
$\varnothing_{SL1}$	$= \varnothing_{Bo10}$	Diameter: Bo_7 hole at HC connection	$SL_{1\text{-}6}$
l_{SL1}	$= \varnothing_{SL1}$	Length: HC connection distance to the outer wall side	$SL_{1\text{-}6}$
l_{SL2}	Input	Length: HC connection slot depth	$SL_{2,4,6}$
l_{SL3}	$= (l_{B5} + l_{B1}/2 + 1) \times 2$	Length: SL link	$SL_{1\text{-}2}$
l_{SL4}	= Equation (5.1)	Length: SL link	$SL_{3\text{-}4}$
l_{SL5}	= Equation	Length: SL link	$SL_{5\text{-}6}$

5.4.7. Hollow Connector ($HC_{1\text{-}2}$) and Connector ($C_{1\text{-}7}$) Parameters

Hollow connectors ($HC_{1\text{-}2}$) are shown in Figure 5.24, and parameters are listed in Table 5.14. Some of the parameters ($t_{HC1\text{-}3}$ and $\varnothing_{HC3}$) are defined as analogous to $SL_{1\text{-}2}$ parameters. A hole diameter ($\varnothing_{HC1}$) is defined for Bo_6 and $C_{1\text{-}2}$ hole connection. A HC connects to a binary hub (B) by either Bo_6 or one of the $C_{1\text{-}2}$. Bo_6 is used when a module is connected from that side to a single module. However, $C_{1\text{-}2}$ is used when the two modules are at the corners. Another whole is defined for SL connection by Bo_7 as $\varnothing_{HC3}$. Corresponding washer (W_3) parameters are defined.

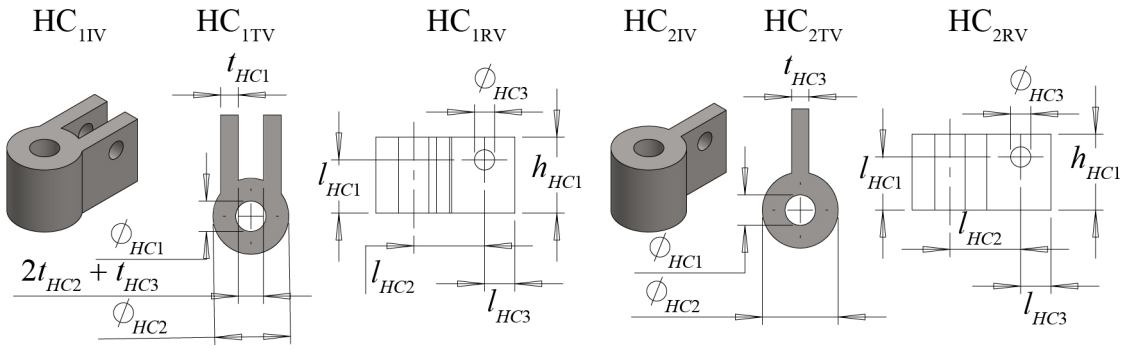


Figure 5.24. Parameters of hollow connectors ($HC_{1\text{-}2}$).

Table 5.14. Parameters of $HC_{1\text{-}2}$.

Parameter	Value	Description	Used in
h_{HC1}	Input	Height: HC	$HC_{1\text{-}2}$
t_{HC1}	$= t_{SL1}$	Thickness: SL connection wall	HC_1
t_{HC2}	$= t_{SL2}$	Thickness: SL connection wall separation distance for W_3	HC_1
t_{HC3}	$= t_{SL3}$	Thickness: SL connection wall	$HC_{1\text{-}2}$
$\varnothing_{HC1}$	$= \varnothing_{Bo6}$	Diameter: Bo_6 and $C_{1\text{-}4}$ hole at B connection	$HC_{1\text{-}2}$
$\varnothing_{HC2}$	Input	Diameter: B connection outer	$HC_{1\text{-}2}$
$\varnothing_{HC3}$	$= \varnothing_{SL1}$	Diameter: Bo_7 hole at SL connection	$HC_{1\text{-}2}$
l_{HC1}	Input	Length: HC joint offset	$HC_{1\text{-}2}$
l_{HC2}	Input	Length: HC link	$HC_{1\text{-}2}$
l_{HC3}	$= 1.5\varnothing_{HC3}$	Length: SL connection distance to the wall side	$HC_{1\text{-}2}$

Bolt-like connectors are defined in four configurations ($C_{1\text{-}4}$). $C_{1\text{-}2}$ is shown in Figure 5.25, and $C_{1\text{-}4}$ parameters are listed in Table 5.15. $C_{1\text{-}2}$ and $C_{3\text{-}4}$ are variations in terms of connector head height ($h_{C1\text{-}2}$) and shaft part length ($l_{C3\text{-}4}$), respectively. $C_{1\text{-}2}$ is designed to be used with $HC_{1\text{-}2}$ to connect binary hub (B) at the corner where two modules are connected. l_{C3} is computed according to B height ($h_{B1\text{-}2}$), washer height (h_{W1}) and hollow connector height (h_{HC1}). On the other hand, $C_{3\text{-}4}$ is designed for a single module connected side. l_{C4} increase by h_{HC1} and h_{W1}. A Bo_7 connection hole ($\varnothing_{C4}$) is defined at the SL connection. Thickness ($t_{C1\text{-}3}$) is defined similar to the analogous $SL_{1\text{-}6}$ parameters.

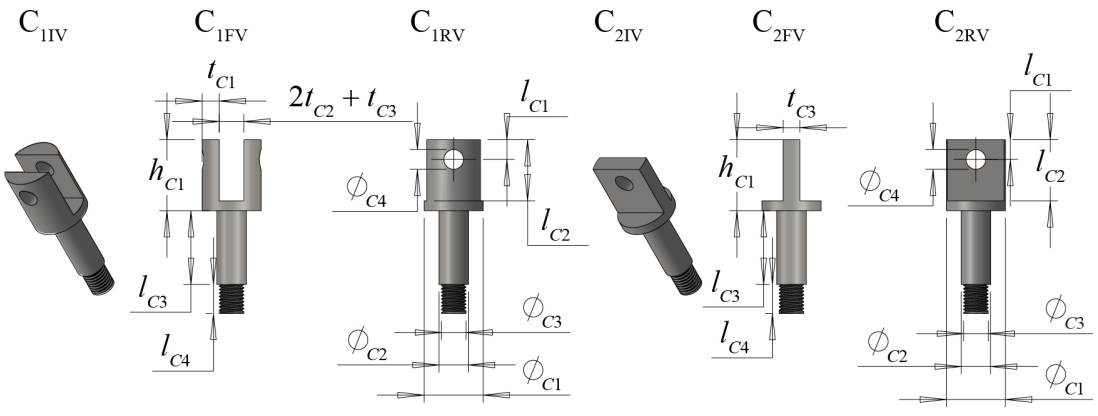


Figure 5.25. Parameters of connectors ($C_{1\text{-}2}$).

Table 5.15. Parameters of $C_{1\text{-}4}$.

Parameter	Value	Description	Used in
h_{C1}	$= l_{C2} + 2$	Height: connector head	$C_{1\text{-}2}$
h_{C2}	$= h_{HC1} + h_{W1} + h_{C1}$	Height: connector head	$C_{3\text{-}4}$
t_{C1}	$= t_{SL1}$	Thickness: SL connection wall	$C_{1,3}$
t_{C2}	$= t_{SL2}$	Thickness: SL connection wall separation distance for W_3	$C_{1\text{-}4}$
t_{C3}	$= t_{SL3}$	Thickness: SL connection wall	$C_{1\text{-}4}$
$\varnothing_{C1}$	Input	Diameter: head	$C_{1\text{-}4}$
$\varnothing_{C2}$	$= \varnothing_{Bo6}$	Diameter: shaft part	$C_{1\text{-}4}$
$\varnothing_{C3}$	$= \varnothing_{Bo7}$	Diameter: thread part	$C_{1\text{-}4}$
$\varnothing_{C4}$	$= \varnothing_{SL1}$	Diameter: Bo_7 hole at SL connection	$C_{1\text{-}4}$
l_{C1}	$= \varnothing_{C4}$	Length: SL connection distance to the head	$C_{1\text{-}4}$
l_{C2}	Input	Length: SL connection slot	$C_{1\text{-}4}$
l_{C3}	$= h_{B1} + h_{B2} + h_{W1} + h_{HC1} - 0.5$	Length: shaft part	$C_{1\text{-}2}$
l_{C4}	$= l_{C3} - h_{W1} - h_{HC1}$	Length: shaft part	$C_{3\text{-}4}$
l_{C5}	$= 2h_{N1} + 0.5$	Length: thread part	$C_{1\text{-}4}$

As mentioned before, $SL_{1\text{-}2}$ length l_{SL3} is defined according to the B dimensions for module connection Type-I. Type-II sheet link ($SL_{3\text{-}6}$) lengths are $l_{SL4\text{-}5}$. Both types can be defined as dependent to satisfy the same pose. The common pose of two connection types can selected as deployed form pose to ensure π/4 between planes defined by faces of Q hubs (see Figure 5.15). Input angle of a modified Altmann linkage (θ_1) for this orientation is measured in the CAD environment. Figure 5.26 demonstrates how l_{SL4} is computed at a common pose of $SL_{2,4}$ connections. l_{SL4} defines $SL_{3\text{-}4}$ links length where SL_3 or SL_4 are used to connect the larger half-module sides (l_{SL5} and SL_5 or SL_6 for the lower side). Two connected modules are always symmetric about the vertical line (L_1), which is a bisector of angle β_{SL1} between L_2 and L_3. L_1 is also a perpendicular bisector of l_{SL3} and l_{SL4}. l_{SL4} is computed in Eq. (5.1) according to known lengths l_{C1}, $l_{HC1\text{-}2}$, l_{SL4}, h_{C1}, h_{HC1}, h_{W1} and β_{SL1}:

$$l_{SL4} = l_{SL3} + 2\sin\left(\frac{\beta_{SL1}}{2}\right)\left(h_{HC1} - l_{HC1} + h_{W1} + h_{C1} - l_{C1}\right) - 2\cos\left(\frac{\beta_{SL1}}{2}\right) \tag{5.1}$$

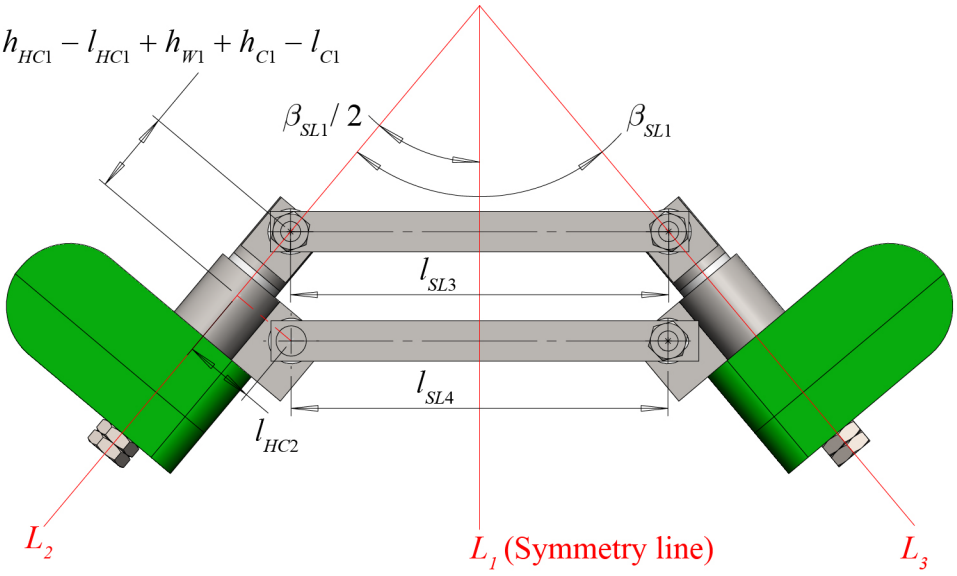


Figure 5.26. Details of $SL_{2\text{-}3}$ length computation.

L_2 and L_3 directions in Figure 5.27 are known by kinematics calculation. The connection directions ($v_{L2\text{-}3}$) are nothing other than x_3 of the related modified Altmann linkage loops which is the common normal of joint axes z_3 and z_4 (see Figure 3.3). The definitions of v_{L2} and v_{L5} are given in the computation of L_1 and L_4 in the kinematic analysis section. The desired pose is selected to have $\pi/4$ rotation between connected modules about the z-axis in Figure 5.27. Direction of a vector is independent of its position. Thus, the direction of L_2 and L_4 can be transformed into the direction of L_3 and L_5 by two successive rotations of a module. If v_{L2} rotates by $\pi/2$ about the x-axis and then rotates $-\pi/2$ about the z-axis to obtain v_{L4}. Then, v_{L3} rotates $-\pi/2$ about x- and z-axes in a sequence to obtain v_{L4} in Equation (5.2). *Rotation*(*Direction*, *Rotation*) function in the kinematics library is used to compute these vectors.

$$\begin{aligned} v_{L4} &= Rotation\left(z,-\pi/4\right)\cdot Rotation\left(x,\pi/2\right)\cdot v_{L2} \\ v_{L5} &= Rotation\left(z,-\pi/4\right)\cdot Rotation\left(x,-\pi/2\right)\cdot v_{L3} \end{aligned} \tag{5.2}$$

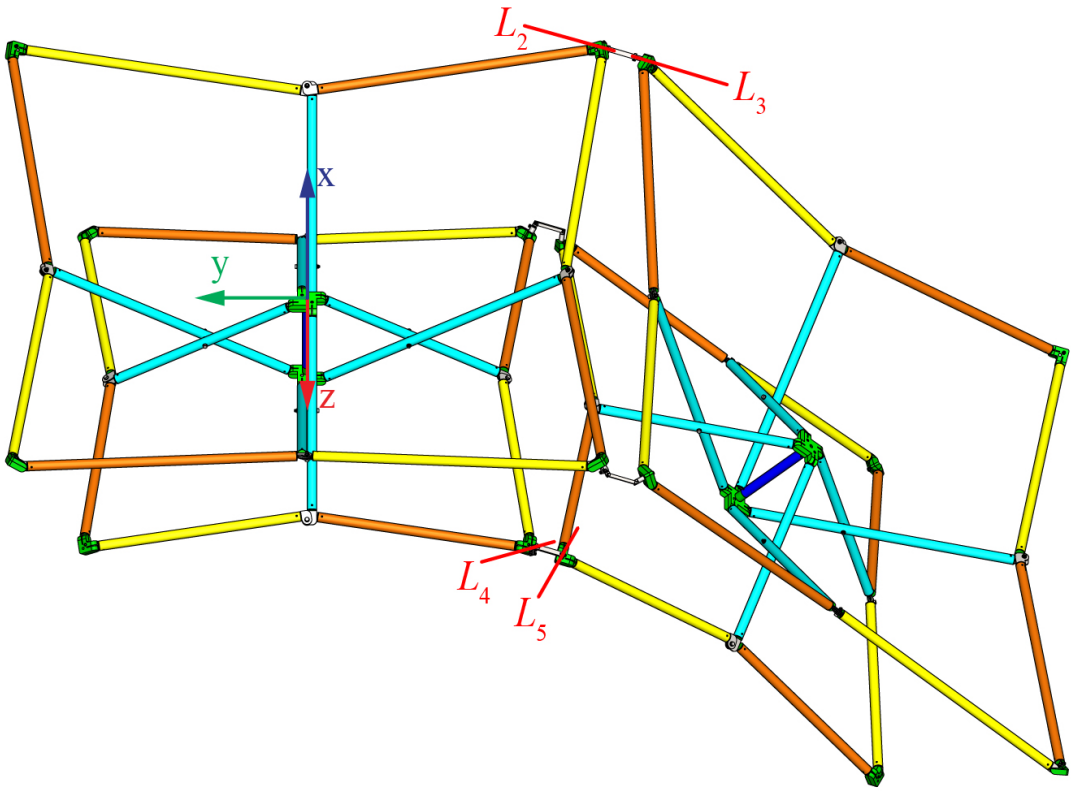


Figure 5.27. Details of the lines $L_{2\text{-}5}$.

The inverse cosine of dot products of $v_{L2,4}$ and $v_{L3,5}$ defines angle $\beta_{SL1\text{-}2}$ as:

$$\begin{aligned} \beta_{SL1} &= \cos^{-1}\left(v_{L2} \cdot v_{L4}\right) \\ \beta_{SL2} &= \cos^{-1}\left(v_{L3} \cdot v_{L5}\right) \end{aligned} \tag{5.3}$$

Similarly, l_{SL5} length of $SL_{5\text{-}6}$ can be found by substituting the angle between lower side lines ($L_{3,5}$) as β_{SL2} into Equation (5.1):

$$l_{SL5} = l_{SL3} + 2\sin\left(\frac{\beta_{SL2}}{2}\right)\left(h_{HC1} - l_{HC1} + h_{W1} + h_{C1} - l_{C1}\right) - 2\cos\left(\frac{\beta_{SL2}}{2}\right) \tag{5.4}$$

5.5. Kinematic Analysis of a Module

Once the part configurations and their parameters are defined in the input worksheet of Parametric Design Integrator (PDI), an assembly is created in the CAD environment. In this section, the loop closure equations are computed and verified with points ($P_{1\text{-}48}$) attached at the DH frame origins and desired locations on parts. Figure 5.28 shows the module with attached points and eight independent loops ($L_{1\text{-}8}$). The points are implemented in the output worksheet of the parametric design integrator in Excel and defined by the point feature in the CAD environment. Model parameters are debugged and verified easily by comparing computed values with the CAD data. Kinematics library functions (*Transformation*, *LineSymmetric* and *PlaneSymmetric*) are used to evaluate point coordinates and frames attached to them (see Section 4.1.3).

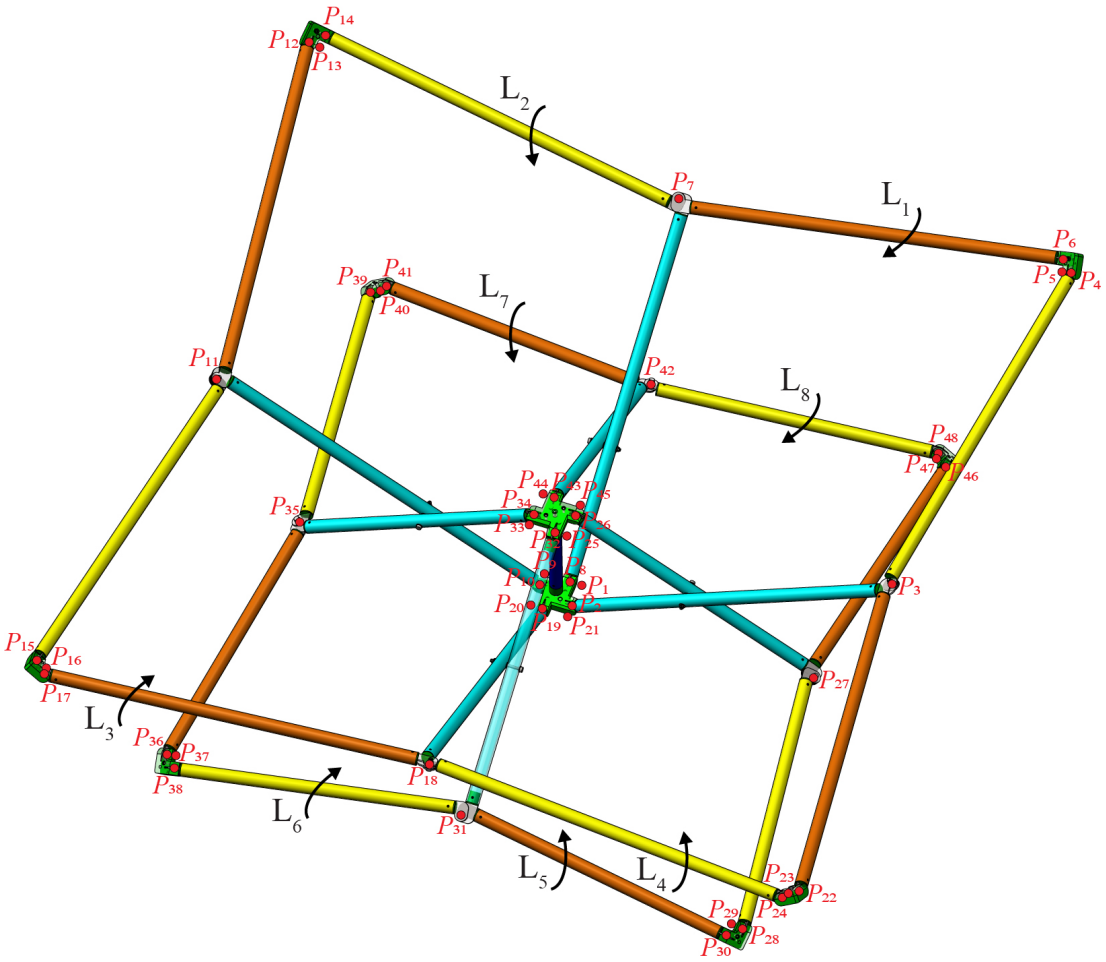


Figure 5.28. Module with the attached points ($P_{1\text{-}48}$) of the loops ($L_{1\text{-}8}$).

5.5.1. Analysis of Loops $L_{1\text{-}4}$

Kinematic analyses starts with loop L_1, and eight points $P_{1\text{-}8}$ defined in it (Figure 5.29). The quaternary hub (Q) of the module is fixed. $P_{1,3,5,7}$ are defined at DH frame origins of six joints (frames 1-6 and 2-3 intersecting in Figure 5.29b). A global coordinate frame is defined at P_1 by x_1- and z_1-axes. Points $P_{2,4,6,8}$ are defined at link-hub contacts, i.e., at the associated link and hub contact plane and joint axis intersections. Note that, no frame is attached to these points. θ_1 is the input of the linkage. θ_2 and θ_3 are independent of link lengths a_1 and d_1 for the open configuration ($\sigma = 1$) as defined in Eq. (3.7).

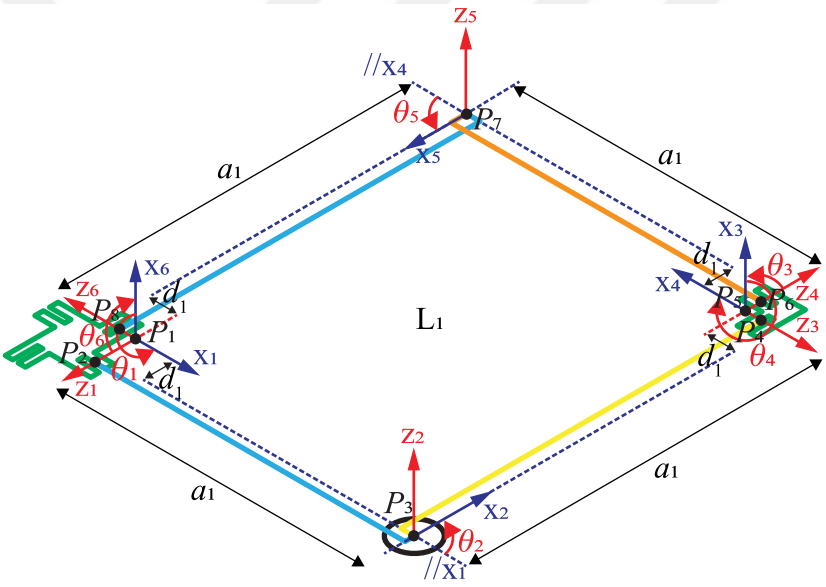


Figure 5.29. Schematic of the loop (L_1).

Point coordinates are computed by the *Transformation* function. The function syntax is *Transformation*(*Axis, Translation*, *Rotation*, *InitialPose*). It computes a 4 × 4 homogenous transformation matrix output according to desired translation and rotation about x-, y- and z-axis directions following (McCarthy and Soh 2010, 256-57). The initial pose is optional. In a conventional forward kinematic analysis, coordinate frames are attached on joint axes according to the DH parameters and the pose of the mechanism is obtained by two successive transformations about x- and z-directions in a sequence. L-

shaped links (links attached to $S_{2\text{-}4}$) in the design results in joint offset (d_1) being satisfied by these links and the hubs Q and B. Transformation about y-direction is defined in computations of coordinates of points $P_{2,4,6,8}$. Transformation inputs are defined according to the DH parameters and link lengths shown in Figure 5.29. Point definitions are listed in Table 5.16. Here, "*P"* represents the defined point, and "P_i" is the initial point location used in the transformation. The function does not require indices of the "axis" column. It only checks the first character and then makes computations. Some points are defined by more than one transformation and intermediate points are defined for that, such as P_{3x1} and its result of an operation in x-direction. At this point, we can define the direction vector of the line (L_2) defined in $SL_{2\text{-}3}$ computations. v_{L2} is computed in the first column of the transformation matrix output of P_5.

Table 5.16. Definition of the loop (L_1) points ($P_{1\text{-}8}$).

P	P_i	Axis	Translation	Rotation
P_1	Origin			
P_2	P_1	z_1	$= (l_{Q1} - l_{Q2})/2 + l_{Q2}$	$= \theta_1$
P_{3x1}	P_2	x_1	$= a_1$	$= \alpha_{12} = -\pi/2$
P_{3y1}	P_{3x1}	y_1	$= l_{S7} - t_{S1}/2$	$= 0$
P_3	P_{3y1}	z_2	$= d_2 = 0$	$= \theta_2$
P_{4y2}	P_3	y_2	$= (\varnothing_{T1} + t_{S1})/2$	$= 0$
P_4	P_{4y2}	x_2	$= a_1$	$= \alpha_{23} = \pi/2$
P_5	P_4	z_3	$= - l_{B2}$	$= \theta_3$
P_{6x3}	P_5	x_3	$= a_{34} = 0$	$= \alpha_{34} = \pi/2$
P_6	P_{6x3}	z_1	$= l_{B2}$	$= \theta_1$
P_{7x1}	P_6	x_1	$= a_1$	$= \alpha_{12} = -\pi/2$
P_{7y1}	P_{7x1}	y_1	$= -(l_{S3} + t_{S1})/2$	$= 0$
P_7	P_{7y1}	z_2	$= d_2 = 0$	$= \theta_2$
P_{8y2}	P_7	y_2	$= l_{S7} + t_{S1}/2$	$= 0$
P_{8x2}	P_{8y2}	x_2	$= a_1$	$= \alpha_{23} = \pi/2$
P_8	P_{8x2}	z_3	$= 0$	$= \theta_3$

Points $P_{9\text{-}24}$ are defined for $L_{2\text{-}4}$ as shown in Figure 5.30. The coordinates are computed by using the symmetry properties of the loops. The points can be defined by plane-symmetry operation(s) about xz- or xy-plane intersecting at the center of the quaternary hub Q. Symmetry planes have a normal distance of d_1 to the origin located at P_1. For instance, point P_9 is mirror-symmetryic point of P_1 with respect to xz-plane. *PlaneSymmetric*(*RefPoint*, *PlaneDistance*, *Axis1*, *Axis2*) function of the kinematics library is used for these computations. The function works for 3D coordinates of a reference point, normal distance of the plane to the origin and axes defining the symmetry plane. For example, if Axis-1 is x and Axis-2 is y, then PlaneDistance is in z-direction. Table 5.17 lists the plane-symmetry operations done for $L_{2\text{-}4}$ points ($P_{9\text{-}24}$). The plane distance is d_1 for all planes. Some of the points are obtained by two successive plane-symmetry operations with respect to both xz- and xy-plane, such as P_{10} is symmetric to P_2. When a point is reflected about two planes, then it is reflected about the line of intersection of the two planes. However, it is preferred to have a single type of operation. Therefore, an intermediate symmetry point is defined for the first plane-symmetry operation such as P_{10S} for xy-plane symmetry operation before obtaining P_{10}.

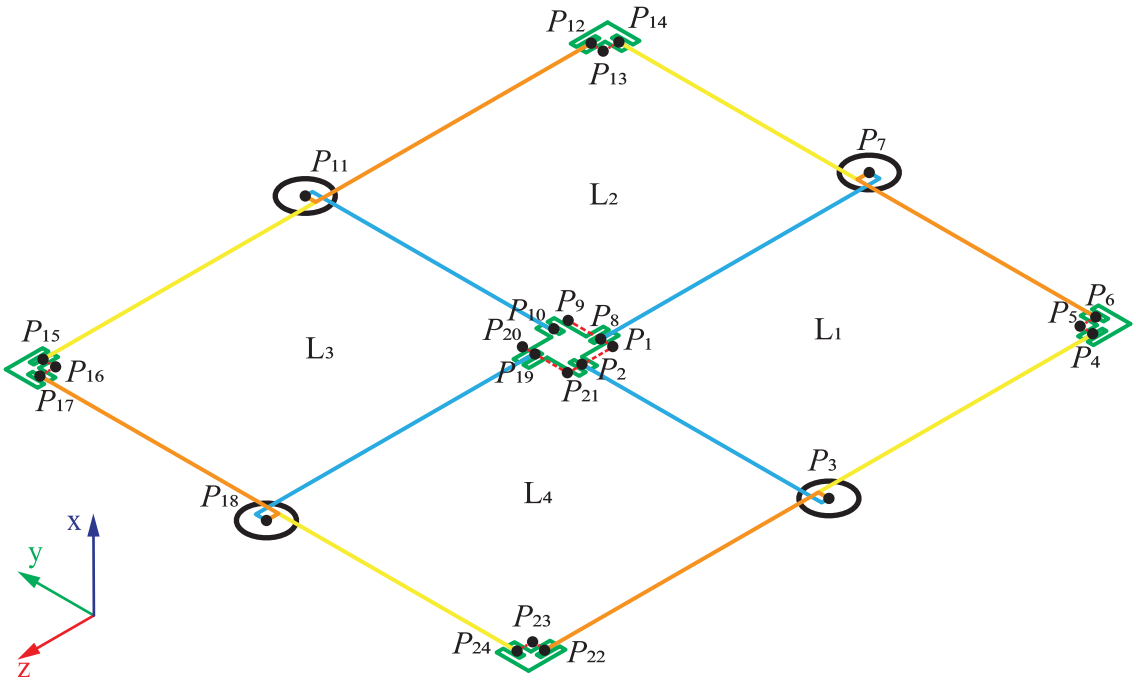


Figure 5.30. Schematic of the loops ($L_{2\text{-}4}$).

Table 5.17. Definition of the loops ($L_{2\text{-}4}$) points ($P_{9\text{-}24}$).

P	P_i	Axis-1	Axis-2		P	P_i	Axis-1	Axis-2
P_9	P_1	x_1	z_1		P_{17x3}	P_{14z3}	x_1	y_1
P_{10S}	P_2	x_1	y_1		P_{17}	P_{14}	x_1	y_1
P_{10}	P_{10S}	x_1	z_1		P_{18x1}	P_{7x1}	x_1	y_1
P_{11x1S}	P_{3x1}	x_1	z_1		P_{18y1}	P_{7y1}	x_1	y_1
P_{11y1S}	P_{3y1}	x_1	z_1		P_{18}	P_7	x_1	y_1
P_{11x1}	P_{11x1S}	x_1	y_1		P_{19y2S}	P_{8y2}	x_1	y_1
P_{11y1}	P_{11y1S}	x_1	y_1		P_{19x2S}	P_{8x2}	x_1	y_1
P_{11}	P_3	x_1	z_1		P_{19y2}	P_{19y2S}	x_1	z_1
P_{12y2}	P_{4y2}	x_1	z_1		P_{19}	P_{19x2S}	x_1	z_1
P_{12}	P_4	x_1	z_1		P_{20}	P_9	x_1	y_1
P_{13}	P_5	x_1	z_1		P_{21}	P_1	x_1	y_1
P_{14x3}	P_{6x3}	x_1	z_1		P_{22y2}	P_{4y2}	x_1	y_1
P_{14}	P_6	x_1	z_1		P_{22}	P_4	x_1	y_1
P_{15y2}	P_{12y2}	x_1	y_1		P_{23}	P_5	x_1	y_1
P_{15}	P_{12}	x_1	y_1		P_{24x3}	P_{6x3}	x_1	y_1
P_{16}	P_{13}	x_1	y_1		P_{24}	P_6	x_1	y_1

5.5.2. Analysis of Loops $L_{5\text{-}8}$

When two half-modules are identical ($a_1 = a_2$), they are symmetrical about a line in the y_1-direction. Figure 5.31 shows details of this symmetry. The symmetry line is the intersection of a yz-plane defined by the half-modules connections and an xy-plane defined at the center of the quaternary hubs (Q). Normal distances of the planes to the origin are $x = l_{T9}c\theta_1, z = d_1$ for yz- and xy-planes. Loop L_5 is reflection of L_{5S} with respect to the symmetry line for the equilateral half-module case. Thus, analysis of the smaller half-module is started with a virtual loop L_{5S}. The loop is solved by computations defined in Table 5.16 by changing a_1-related parameters to a_2-related parameters. Then, line-

symmetry operation is applied to obtain L_5. Lastly, $L_{6\text{-}8}$ is obtained by plane-symmetry operations as in Table 5.17.

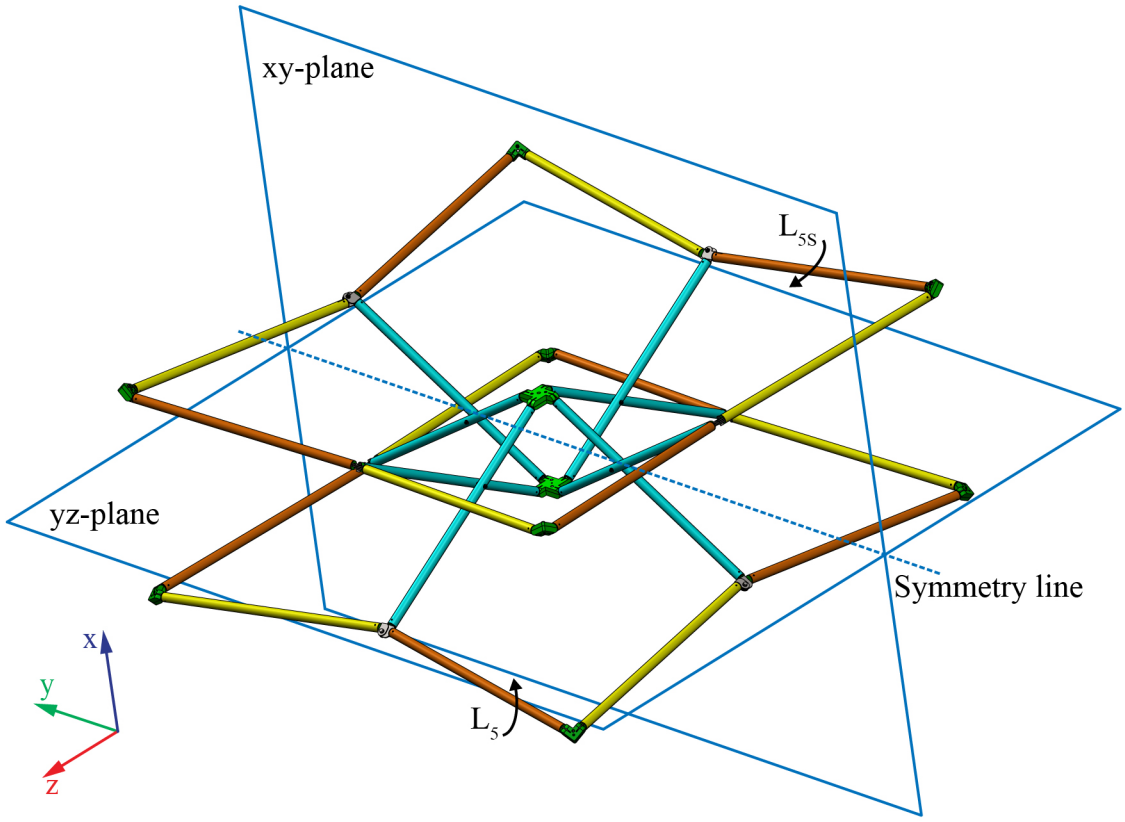


Figure 5.31. Details of the symmetry between two equilateral half-modules.

LineSymmetric(*RefPoint*, *Direction*, *Axis1Distance*, *Axis2Distance*) function defines a symmetry line by one reference direction and the distance of the symmetry line to the origin in the remaining two directions. For instance, if the symmetry line is in the y-direction, x-axis and z-axis distances are *Axis1Disance* and *Axis2Distance*.

$P_{25\text{-}32}$ are defined for L_5 (see Figure 5.32). *Transformation* and *LineSymmetric* function inputs are listed in Table 5.18. First, *Transformation* function is applied to solve L_{5S} for points $P_{25S\text{-}32S}$, then *LineSymmetric* function is used for L_5 to obtain points P_{25}-P_{32}. Since identical quaternary hubs (Q) are used in smaller and larger half-modules, there is no need to compute points $P_{25\text{-}26,32}$ by *Transformation* function on the quaternary hub. The points $P_{1\text{-}2,8}$ are already defined for L_1. Similar to v_{L2}, the direction vector v_{L3} of the line L_3 defined in $SL_{4\text{-}5}$ computations is found in the first column of the transformation matrix output of P_5.

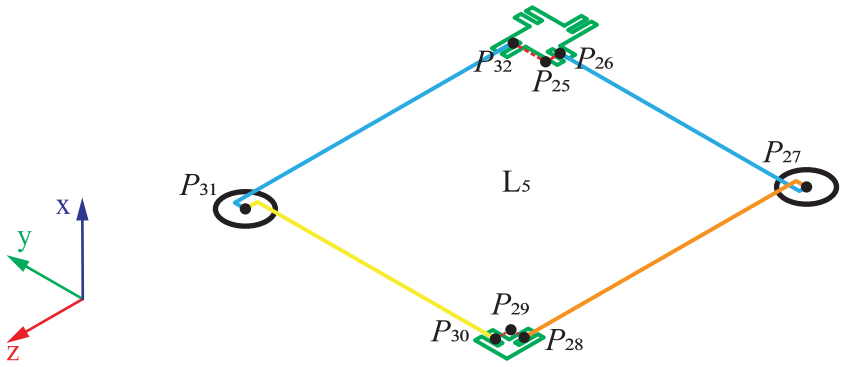


Figure 5.32. Schematic of the loop (L_5).

Table 5.18. Definition of the loop (L_5) points ($P_{25\text{-}32}$).

P	P_i	Axis	Translation	Rotation
P_{27x1S}	P_2	x_1	$= a_2$	$= \alpha_{12} = -\pi/2$
P_{27y1S}	P_{27x1S}	y_1	$= l_{S7} - t_{S1}/2$	$= 0$
P_{27S}	P_{27y1S}	z_2	$= d_2 = 0$	$= \theta_2$
P_{28y2S}	P_{27S}	y_2	$= (\varnothing_{T1} + t_{S1})/2$	$= 0$
P_{28S}	P_{28y2S}	x_2	$= a_2$	$= \alpha_{23} = \pi/2$
P_{29S}	P_{28S}	z_3	$= -l_{B2}$	$= \theta_3$
P_{30x3S}	P_{29S}	x_3	$= a_{34} = 0$	$= \alpha_{34} = \pi/2$
P_{30S}	P_{30x3S}	z_1	$= l_{B2}$	$= \theta_1$
P_{31x1S}	P_{30S}	x_1	$= a_2$	$= \alpha_{12} = -\pi/2$
P_{31y1S}	P_{31x1S}	y_1	$= -(l_{S3} + t_{S1})/2$	$= 0$
P_{31S}	P_{31x1S}	z_2	$= d_2 = 0$	$= \theta_2$
P	P_i	Direction	Axis-1 Distance	Axis-2 Distance
P_{25}	P_1	y	$= l_{T9}c\theta_1$	$= d_1$
P_{26}	P_2			
P_{27x1}	P_{27x1S}			
P_{27y1}	P_{27y1S}			
P_{27}	P_{27S}			

(cont. on next page)

Table 5.18 (cont.)

P_{28y2}	P_{28y2S}			
P_{28}	P_{28S}			
P_{29}	P_{29S}			
P_{30x3}	P_{30x3S}			
P_{30}	P_{30S}			
P_{31x1}	P_{31x1S}	y	$= l_{T9} \mathrm{c}\theta_1$	$= d_1$
P_{31y1}	P_{31y1S}			
P_{31}	P_{31S}			
P_{32y2}	P_{8y2S}			
P_{32x2}	P_{8x2S}			
P_{32}	P_{8S}			

Coordinates of points $P_{33\text{-}48}$ for the remaining loops $L_{6\text{-}8}$ are computed by *PlaneSymmetric* function using loop L_5 points $P_{25\text{-}32}$. The plane distance is d_1 for all planes. Points are shown in Figure 5.33 and are defined in Table 5.19.

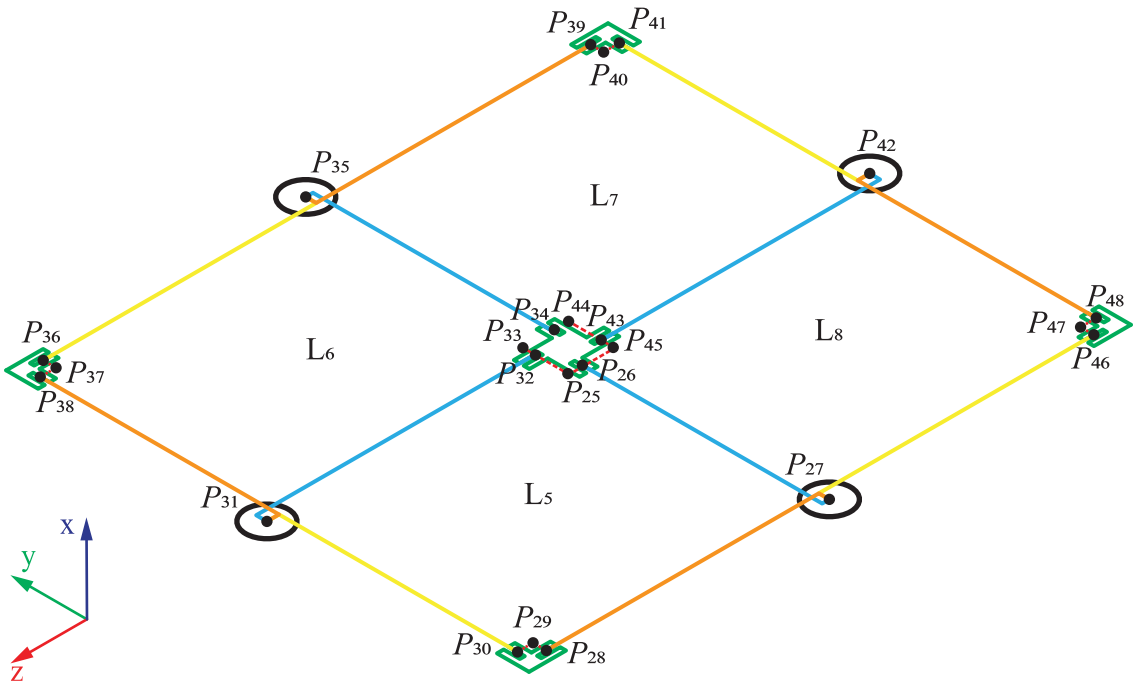


Figure 5.33. Schematic of the loops ($L_{5\text{-}8}$).

Table 5.19. Definition of the points $P_{33\text{-}48}$ of loops $L_{2\text{-}4}$.

P	P_i	Axis-1	Axis-2		P	P_i	Axis-1	Axis-2
P_{33}	P_{25}	x_1	z_1		P_{41x3}	P_{38x3}	x_1	y_1
P_{34S}	P_{26}	x_1	y_1		P_{41}	P_{38}	x_1	y_1
P_{34}	P_{34S}	x_1	z_1		P_{42x1}	P_{31x1}	x_1	y_1
P_{35x1S}	P_{27x1}	x_1	z_1		P_{42y1}	P_{31y1}	x_1	y_1
P_{35y1S}	P_{27y1}	x_1	z_1		P_{42}	P_{31}	x_1	y_1
P_{35x1}	P_{35x1S}	x_1	y_1		P_{43y2S}	P_{32y2S}	x_1	y_1
P_{35y1}	P_{35y1S}	x_1	y_1		P_{43x2S}	P_{32x2S}	x_1	y_1
P_{35}	P_{27}	x_1	z_1		P_{43y2}	P_{43y2S}	x_1	z_1
P_{36y2}	P_{28y2}	x_1	z_1		P_{43}	P_{43x2S}	x_1	z_1
P_{36}	P_{28}	x_1	z_1		P_{44}	P_{33}	x_1	y_1
P_{37}	P_{29}	x_1	z_1		P_{45}	P_{25}	x_1	y_1
P_{38x3}	P_{30x3}	x_1	z_1		P_{46y2}	P_{28y2}	x_1	y_1
P_{38}	P_{30}	x_1	z_1		P_{46}	P_{28}	x_1	y_1
P_{39y2}	P_{36y2}	x_1	y_1		P_{47}	P_{29}	x_1	y_1
P_{39}	P_{36}	x_1	y_1		P_{48x3}	P_{30x3}	x_1	y_1
P_{40}	P_{37}	x_1	y_1		P_{48}	P_{30}	x_1	y_1

5.6. Specifications of the Network

The specifications of the network presented in this section include deployment ratio, floor area and height in deployed form, packed volume (or folded volume) and mass. The network modules form an elongated square cupola in the deployed form. The deployed volume and height of the network can simply be computed by using this solid geometry. Real volume and height are larger than the solid's volume and height. Because, the module connection parts (HC, SL and C) enlarge the height and deployed volume of the network, and the deployed volume and height criteria are defined as a minimum value

in Table 5.2. Elongated square cupola parameters, side length (l_{C1}), and height (h_{C1}) are shown in Figure 5.34.

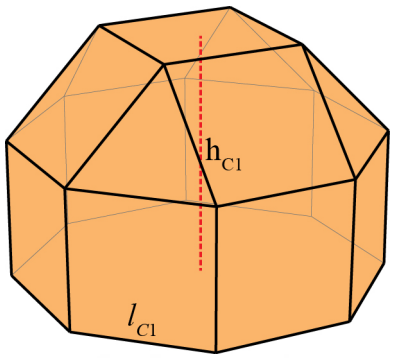


Figure 5.34. Elongated square cupola parameters.

Points $P_{4C,4CS}$ shown in Figure 5.35 are defined to compute the side length l_{C1} in the deployed form. The side length l_{C1} is computed by distances between the points (P_{4C}, P_{4CS}). Table 5.20 lists the *Transformation* and *PlaneSymmetric* function definitions of the points. First, P_{4C} is obtained from P_4 (see Figure 5.29) by transformations, then P_{4CS} is defined as a reflection of P_{4C} about the xz-plane.

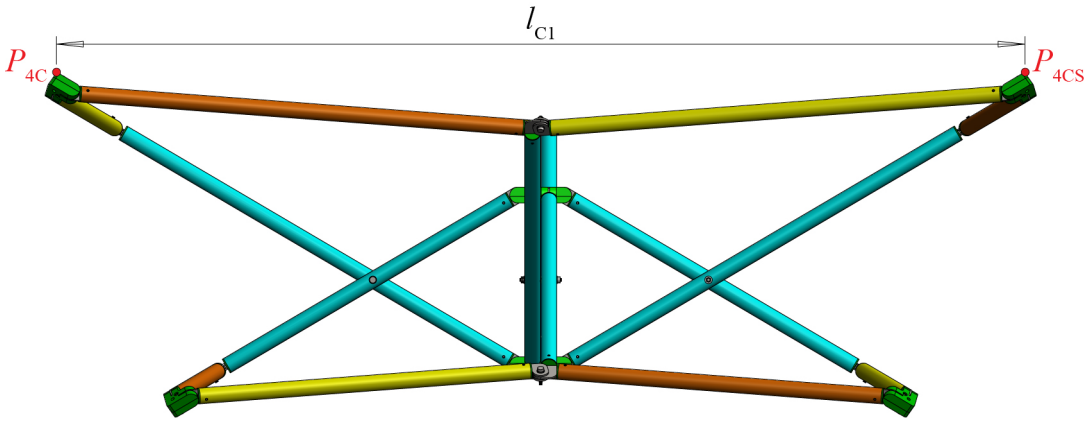


Figure 5.35. Points $P_{4C,4CS}$ defined to compute side length l_C.

Table 5.20. Definition of points $P_{25\text{-}32}$ of loop L_5.

P	P_i	Axis	Translation	Rotation
P_{4Cz3}	P_4	z_3	$= t_{B1} + t_{S1}$	$= \theta_3$
P_{4Cx3}	P_{4Cz3}	x_3	$= a_3 = 0$	$= \alpha_{34} = \pi/2$
P_{4Cx1}	P_{4Cx3}	z_1	$= \varnothing_{SP1} / 2 + h_{B2}$	$= 0$
P_{4C}	P_{4Cx1}	x_1	$= d_1$	$= \theta_1$
P	P_i	Plane dist.	Axis 1	Axis 2
P_{4CS}	P_{4C}	$= d_1$	x_1	z_1

Deployed volume and height of the network are computed in Eq. (5.5) for the side length l_{C1}:

$$V_D = \left(3 + \frac{8\sqrt{2}}{3}\right) l_{C1}^3,\ h_{C1} = l_{C1}\left(1 + \frac{\sqrt{2}}{2}\right) \tag{5.5}$$

Folded form volume V_F is computed using the parameters $l_{F1\text{-}4}$ shown in Figure 5.36. Two rectangular prism volumes composed of $l_{F1\text{-}3}$ and $l_{F2\text{-}4}$ are summed and then subtracted from the common volume ($l_{F2\text{-}3}$) in Eq. (5.6). The parameters $l_{F1\text{-}4}$ are defined in terms of the parameters in Table 5.21. V_F is greater than the actual volume due to $SL_{4,6}$ and HC_2 link angles in the folded form. Even SL_2–$C_{2,4}$ connected modules parallel to the middle module, the other type ($SL_{4,6}$–HC_2) is not. This makes the actual folded volume smaller than V_F.

$$V_F = l_{F2} l_{F3} \left(l_{F1} - l_{F2} + l_{F4}\right) \tag{5.6}$$

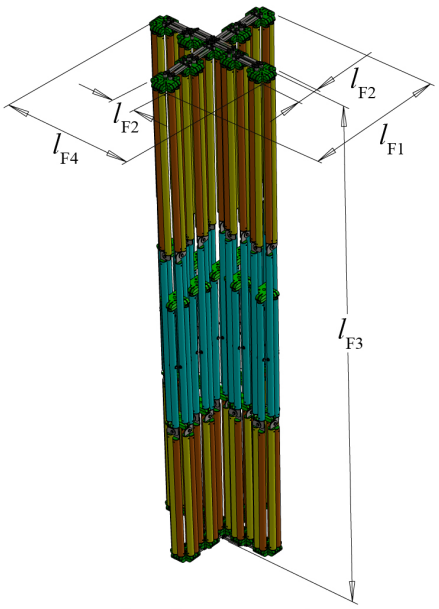


Figure 5.36. Folded volume of the network and its parameters.

Table 5.21. Folded volume parameters.

Parameter	Value
l_{F1}	$= l_{F2} + 5(l_{B1}/2 + t_{B1}) + 4(l_{HC2} + l_{SL5})$
l_{F2}	$= 2(l_{B1} + l_{B3} + l_{B5} + t_{B1})$
l_{F3}	$= 2(a_1 + a_2 - l_{T9}) + h_{Q1} + h_{Q2} + 2(h_{HC1} + h_{C1} + h_{W1})$
l_{F4}	$= l_{F2} + 5(l_{B1}/2 + t_{B1}) + 2(2l_{HC2} + l_{SL4} + l_{SL5})$

Deployment ratio of the network is computed as

$$r_D = \frac{V_F}{V_d} \tag{5.7}$$

The base of an elongated square cupola is a regular octagon. However, the actual area for the network is larger, as is in deployed volume and height computations. It is recognized that when a solution is obtained by using the parametric model, floor area is the most critical one out of the five criteria presented in this section. Thus, floor area is

computed for the real case which is a rectangular octagon (see Figure 5.37). The floor area (A_{Fl}) of the network is defined by two parameters ($l_{C1\text{-}2}$) where l_{C2} is obtained from the CAD environment:

$$A_{Fl} = l_{C1}^2 + 2\sqrt{2} l_{C1} l_{C2} + l_{C2}^2 \tag{5.8}$$

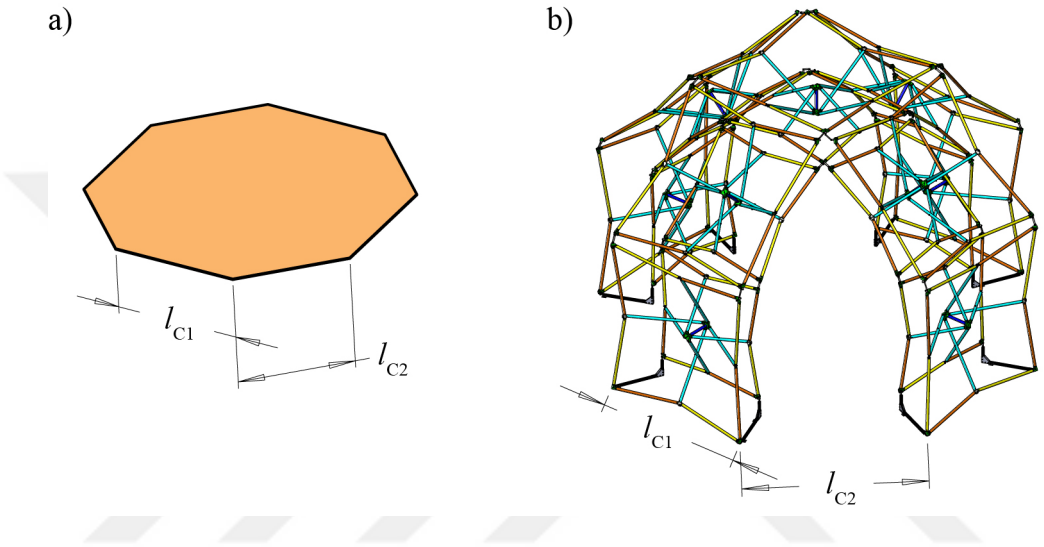


Figure 5.37. Floor area of the network a) rectangular octagon, b) CAD model.

Mass values of the parts are listed in Table 5.22. The CAD environment gives mass data of each part and the whole network according to the specified materials. The total mass of the network (m_{Total}) can also be obtained from the CAD environment.

Table 5.22. Mass values.

Name	# of parts	Mass		Name	# of parts	Mass		Name	# of parts	Mass
Bo_1	108	m_{Bo1}		SP_2	450	m_{SP1}		E_2	144	m_{E2}
Bo_2	9	m_{Bo2}		T_1	36	m_{T1}		E_3	9	m_{E3}
Bo_3	9	m_{Bo3}		T_2	36	m_{T2}		E_4	9	m_{E4}

(cont. on next page)

Table 5.22 (cont.)

Bo_4	72	m_{Bo4}		T_3	36	m_{T3}		B_1	72	m_{B1}
Bo_5	40	m_{Bo5}		T_4	36	m_{T4}		B_2	72	m_{B2}
Bo_6	36	m_{Bo6}		T_5	36	m_{T5}		Q_1	18	m_{Q1}
Bo_7	64	m_{Bo7}		T_6	36	m_{T6}		Q_2	18	m_{Q1}
N_1	296	m_{N1}		T_7	9	m_{T7}		SL_2	8	m_{SL2}
N_2	140	m_{N2}		S_1	216	m_{S1}		SL_4	12	m_{SL4}
W_1	336	m_{W1}		S_2	72	m_{S2}		SL_6	12	m_{SL6}
W_2	108	m_{W2}		S_3	72	m_{S3}		HC_2	48	m_{HC2}
W_3	128	m_{W3}		S_4	72	m_{S4}		C_2	8	m_{C2}
SP_1	216	m_{SP1}		E_1	288	m_{E1}		C_4	8	m_{C4}

Parametric model of the parts, kinematic analysis computations (including inverse kinematics) of a module and specification computations are defined in Microsoft Excel® and connected to the CAD environment (Solidworks®) via PDI (see Figure 5.38). The module is animated in the output worksheet. An angle mate feature is defined for θ_1. Point coordinates of the parametric model are verified using the CAD environment.

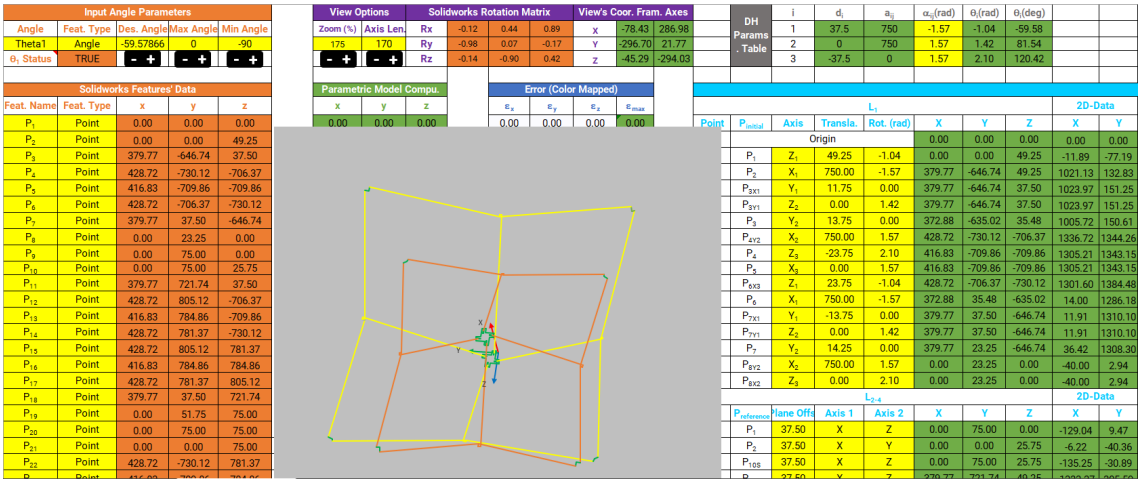


Figure 5.38. Parametric model computations in Microsoft® Excel.

CHAPTER 6

CASE STUDY: FEA SIMULATIONS AND PROTOTYPES

This Chapter presents the prototyping activities of the case study (Prototype-I and Prototype-II). Prototype-I is a scaled version of Prototype-II in terms of the link lengths (a_1 and a_2), and prioritizes exploring the static stability of the network without limiting the motion between module connection parts: binary hub, connector and hollow connector. Prototype-II aims to satisfy the design criteria presented in Table 5.2. A general view of the prototyping activities is given in Figure 6.1. Prototype-I is presented in two stages. A first prototype is designed, followed by structural simulations, manufacturing, and testing. Then, revisions are made, and simulations are repeated in the second stage. Only a digital prototype is built for the simulations of the revised version. Prototype-II is at the mechanical limit-drawing phase of the module-connected parts. The FEA studies carried out in MSC Apex® and MSC Nastran® comprise linear modal and linear-nonlinear static load simulations. The FEA models are generated by using the custom API design presented in Section 4.2. Details of the prototypes and achieved results are given in the next sections.

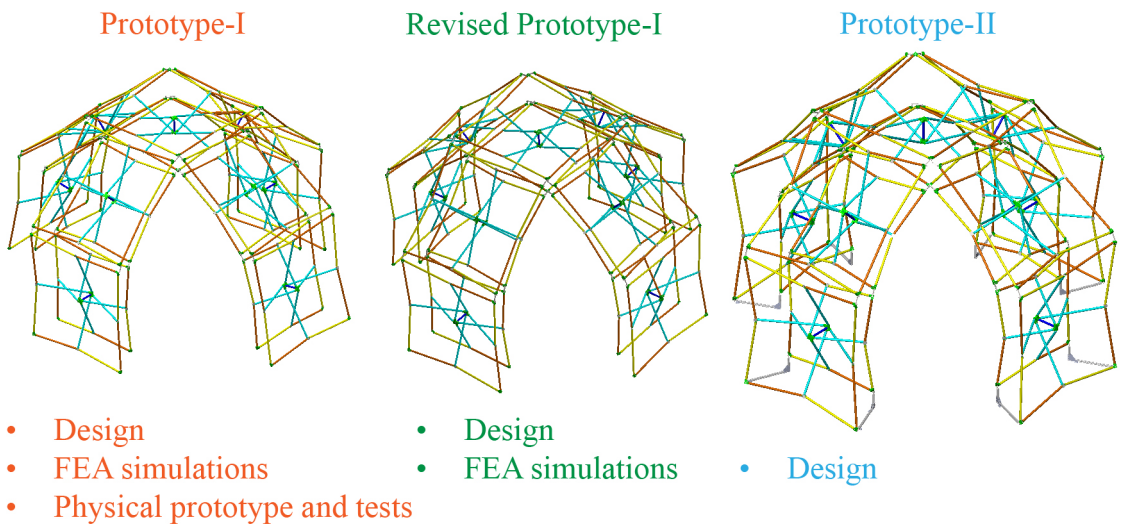


Figure 6.1. Prototyping activities.

6.1. Prototype-I: Design, FEA Simulations and Tests

In addition to observing the static stability of the proposed design in the deployed form, Prototype-I aims to verify the design and manufacturing methods of certain parts. Since the design criteria of the case study require a large network, Prototype-I is scaled by only decreasing the link lengths (a_1, a_2), i.e. lengths of the tubes. This allows experiencing part connections, joint frictions and joint clearances about providing desired motion capability within the tolerances that can be provided.

Geometrically compatible mechanisms require external bracing or lock to obtain desired stiffness and can move under small forces as stated before. Kiper (2016) offered adding springs between module connections instead of sheet links in the cylindrical network design. Making use of mechanical limits is an alternative to prevent the sagging of the network. In order to better understand the effects of these bracing systems, Prototype-I is planned in two stages. First, the prototype is designed without any bracing element, then the design is revised by adding mechanical limits on module connection parts (binary hub, connector and hollow connector) and sheet link part changes. Initially, it is planned to include contact effects of new mechanical limits in FEA simulations and build a physical prototype of the revised version.

6.1.1. Design of Prototype-I

The CAD model of Prototype-I is shown in Figure 6.2. Some module connection parts are different from the general design presented in the sheet links, connectors and hollow connector parts ($SL_{1,3,5}$, $C_{1,3}$ and HC_1). DH parameters are $a_1 = 600$, $a_2 = 500$ and $d_1 = 30$. a_1 and a_2 lengths are determined in order to have a design that can be worked on and be deployed in the laboratory. This prototype is designed after the parametric modeling of a module. Sheet link length computations in Equations (5.1)–(5.4), and a CAD model of the network are not done. Thus, a common pose is not selected for the module connections type-I and II (see Figure 5.13). Thus, related link lengths ($l_{SL4\text{-}5}$) are not computed and defined as input. Type-II connections have a 49.43° angle between the quaternary hub of two connected modules instead of 45° (see Figure 5.15). Note that

Prototype-I does not include the ground fixing system design. Tent stakes are used on the ground-contacting parts in the deployed form.

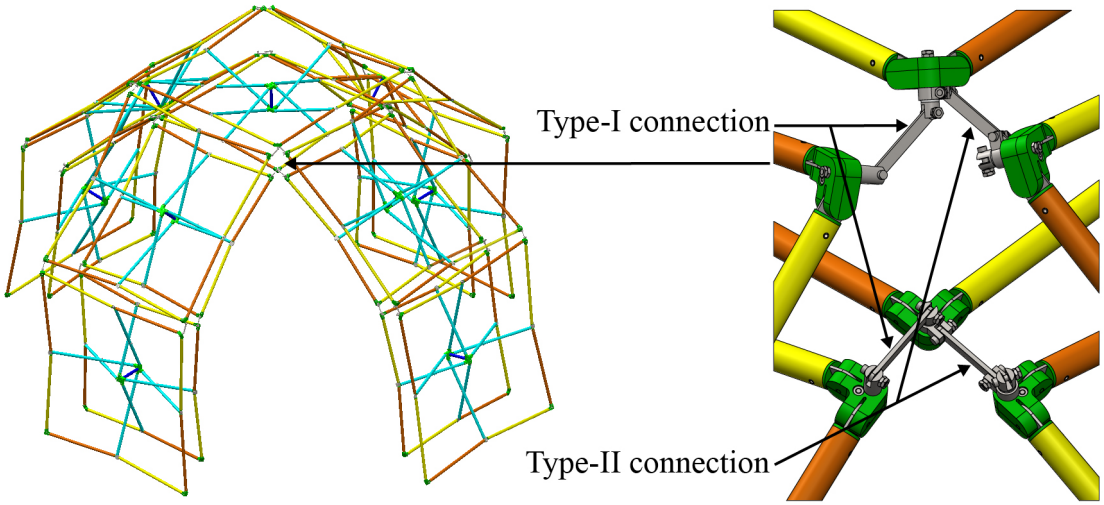


Figure 6.2. Design of Prototype-I.

Plastic parts (end cap, binary hub and quaternary hub) are produced with a 3D printer from PLA material, and the GRP tubes are purchased. 303 or 304 stainless steel parts are either manufactured (hollow connector, sheet link and connectors) or purchased. The hollow connector and sheet links are manufactured by laser-cutting and milling. Connectors are produced using a CNC lathe machine plus an additional milling operation.

6.1.2. FEA Simulations of Prototype-I

The FEA simulation model of Prototype-I is shown in Figure 6.3. The CAD model is simplified by omitting bolts, nuts, washers and coiled spring pins in FEA simulations. Sample operations are defined in the custom API for defeature, split tool, midsurface generation, joint definition, contact definition, constraint definition, material defining, material assigning and mesh generating operations. The tubes are modeled with 2D

elements (surface mesh) whereas the remaining parts (end caps, sheet metal ends, quaternary hubs, binary hubs, connectors, a hollow connector and sheet links) are modeled with 3D elements (solid mesh). The connectors are the only tetra mesh modeled parts. Other 3D elements are hex-mesh (2.5D mesh in MSC Nastran®). mm/t/s/N/K/° consistent unit system is used. The shell orthotropic material model (MAT8) is defined for the tubes ($T_{1\text{-}7}$), while the isotropic model (MAT1) is used for the remaining plastic and metal parts. Permanent glue contacts are defined at solid/surface level for assemblies of links, quaternary hubs and binary hubs, and mechanical limit connection of the modules for quaternary hub-end cap contact. Parts are initially in contact, and there are no relative normal and tangential displacements between contact nodes/segments in permanent glue contacts. Segment-to-segment contact method (distributed penalty) is used. The slotted spring pin is used to define the contacts of the link parts (tubes, end caps and sheet metal ends). Revolute joints are defined between Q_1-S_1, B_1-S_1, S_2-$S_{3,4}$, B_1-HC_1, P_1-P_4, $SL_{1,3,5}$-HC_1 and $C_{3\text{-}4}$-B_1. Boundary conditions are defined by 16 general constraints. Three translational dof of the ground-contacting binary hub surfaces are constrained. 1 g load is added in the negative x direction to the model. This simulates the initial release of the network under the effect of gravity. Meshes are generated at solid/surface levels uniform for each part (not cell/face levels). A mass model containing 1573279 2D and 1560127 3D elements is obtained. The cell/face level definitions and required preprocesses (split operations) should be added to the model to decrease the number of elements. In total, 1833 parts are processed, 1440 contact interactions and 524 revolute joints are defined out of 24 parts, 11 revolute joints and 17 contact samples defined in the custom API. The model is generated in 55 minutes and 11 seconds with a computer equipped with Ryzen 9 7950X 16-Core CPU, 64 GB 6400 MHz DDR5 memory, 2 TB storage offering 7000 MB/s read and write speeds, and running MSC Apex 2024.1.1 on Windows 11.

The model is run for the linear static force analysis simulation scenario, but large displacements with a maximum of 178 mm are observed. This causes geometric nonlinearity. Then, the model is run for the nonlinear scenario (classical SOL 400). The main solver settings are fixed steps of five increments, load case time as 1.0 (with respect to simulation time) and UPV convergence criteria (displacement, load, vector component error control). The simulation is completed in 5 hours 16 minutes and 9 seconds. Nonlinear stress and displacement results are given in Figure 6.4a. Maximum translational displacement is 78.2 mm, which occurs on the lower quaternary hub of the

central module due to the caving in of the network or simply sagging of the networkMaximum nonlinear Von Mises stresses shown in Figure 6.4b occur on sheet metal ends of the ground-contact modules at 526 MPa, which is more than two times of the yield strength of the 304 stainless steel parts. The stress results are found consistent, since the sagging amounts are too high. Simulation results show that the design suffers from sagging in the absence of bracing elements. The simulation model has ideal joints modeled as rigid joint (RJOINT) elements. Thus, they have no clearance and physical prototype sagging is expected to be much more.

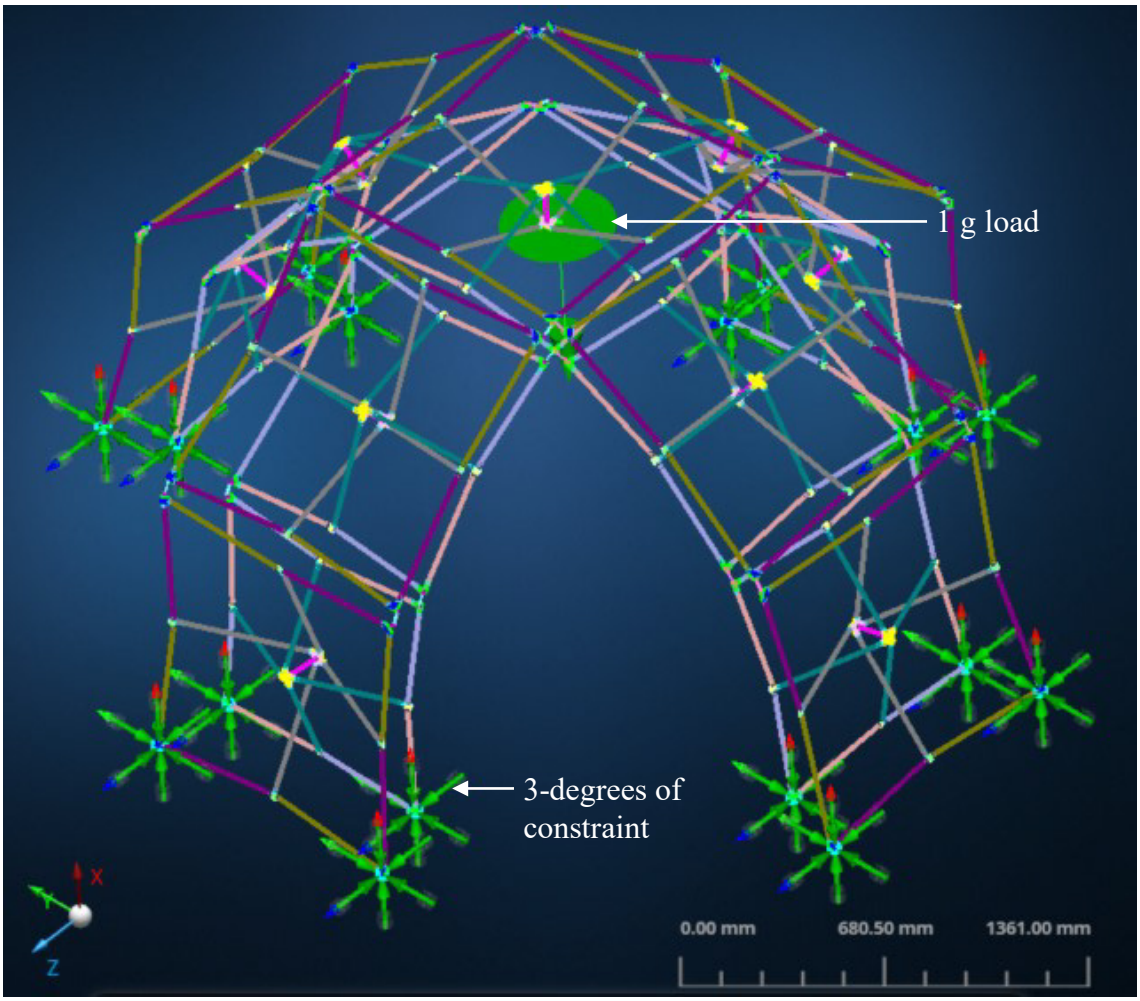


Figure 6.3. FEA simulation model of Prototype I.

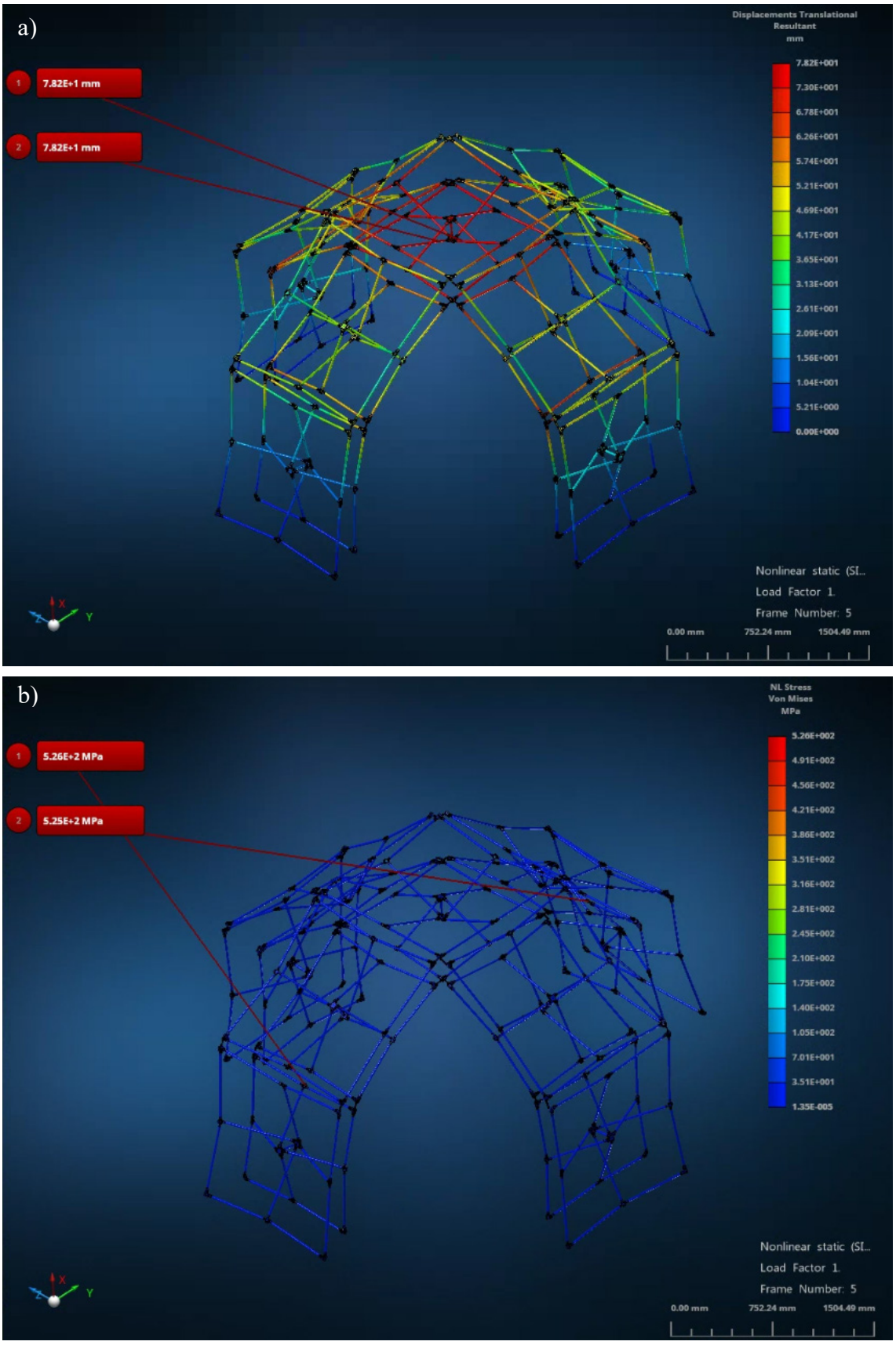


Figure 6.4. FEA simulation results of Prototype-I a) translational displacement results, b) Von Mises stress results.

6.1.3. Manufacturing and Assembly of Prototype-I

The network is an over-constraint linkage. Manufacturing over-constraint linkages requires precision because manufacturing errors can result in high stresses on the parts and even immobility of the assembly. Some clearance can be added to the connections for compensating these errors. The drawback of this is undesired motion freedom in joints, thus the design should be performed carefully. Since joint connection parts (binary hub, quaternary hub, hollow connector and sheet link) are either 3D-printed or laser-cut, trials are made to determine these tolerances. Deployed and folded forms of a module assembly are shown in Figure 6.5. Then, nine of the modules are connected to obtain the network (see Figure 6.6).

a) b)

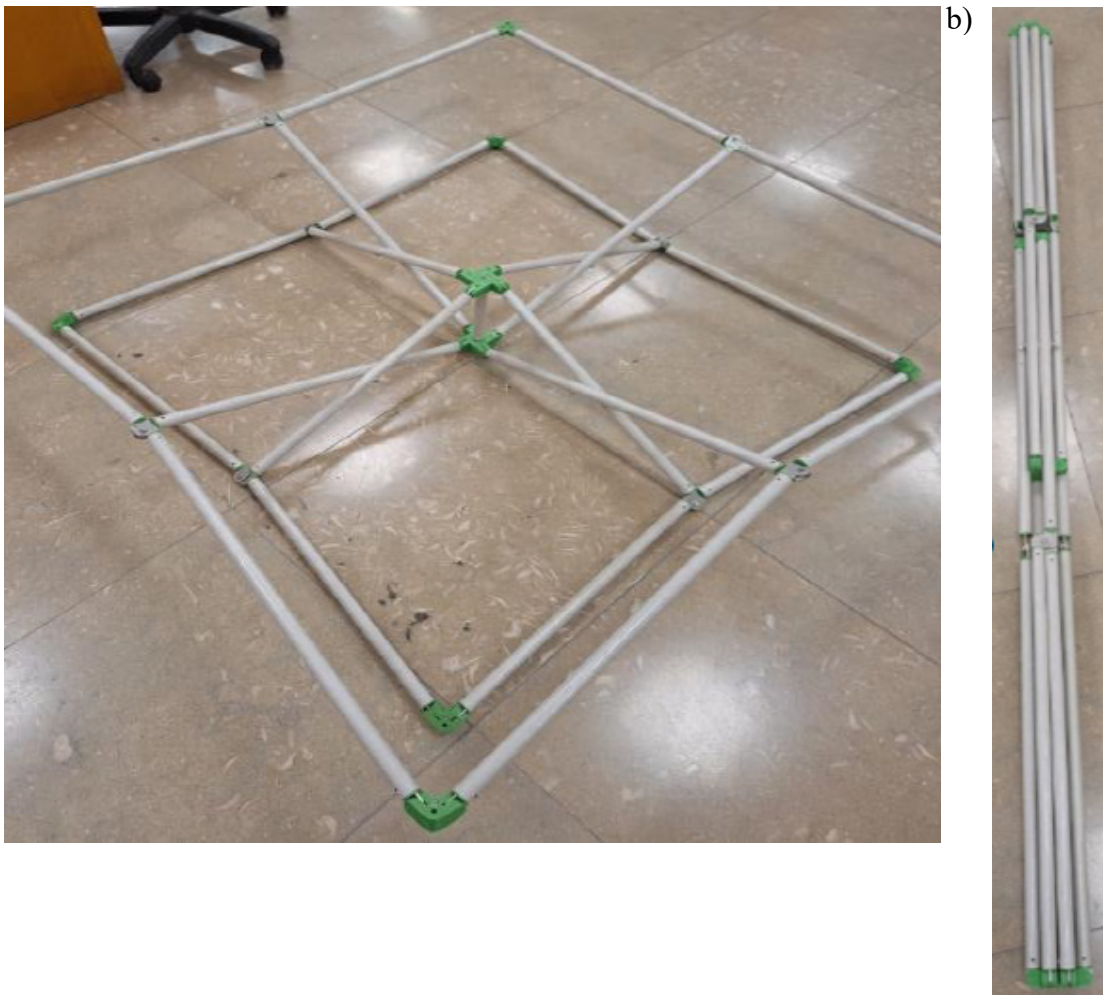


Figure 6.5. A module assembly of Prototype-I a) deployed form b) folded form.

a) b)

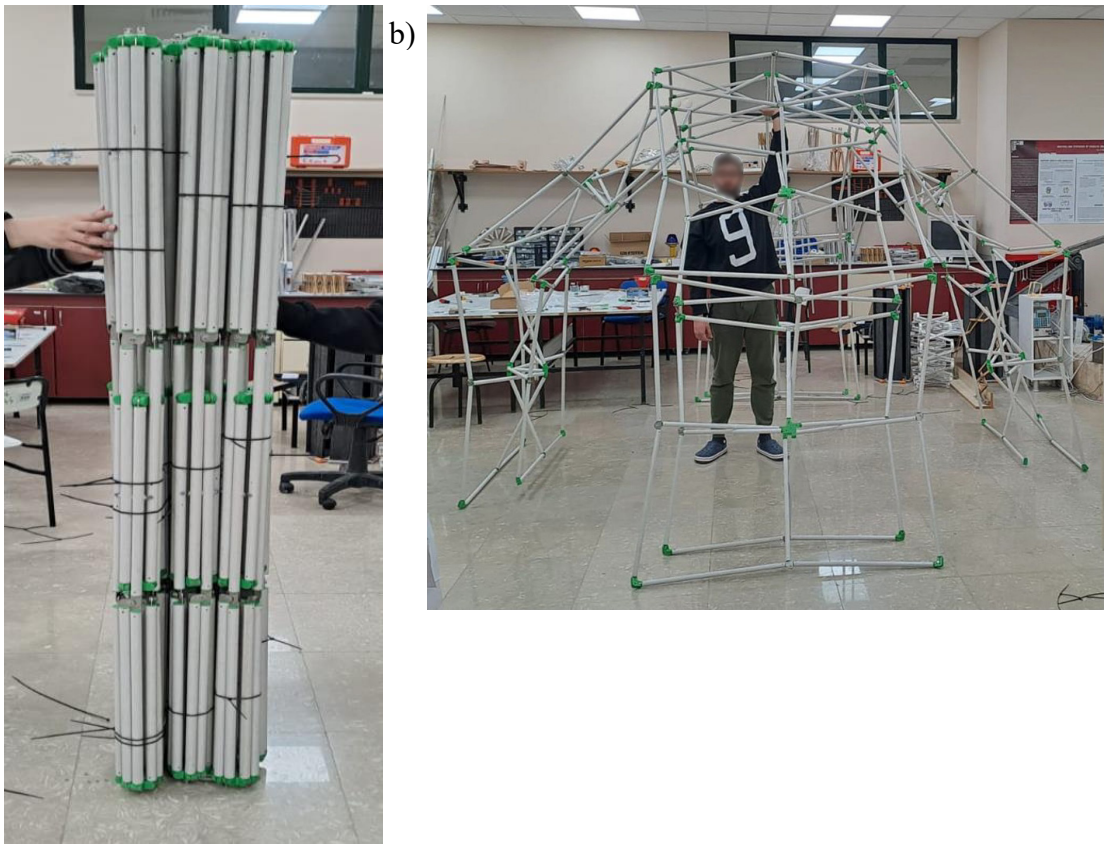


Figure 6.6. Assembly of Prototype-I a) deployed form b) folded form.

6.1.4. Tests and Results of Prototype-I

The network is initially deployed indoors on a flat marble floor, where it is observed that the ground-contacting modules slip on the surface. Afterwards, an outdoor deployment test is conducted on a ground with hard and stony soil. The deployment is performed with six people. After deployment, tent stakes are used to fix the prototype on the ground. Then, middle support is released and the balance is observed. 550 mm sagging is measured at the central module (Figure 6.7). This is seven times larger than the simulation results. Joint clearances are mainly responsible for the difference. The test shows that sagging mainly occurs due to the unwanted motion of the module connections. Assembly mode changes of the module connections are a problem during the deployment. Design revisions are planned according to the observations with this initial prototype.

Figure 6.7. Deployment test result of Prototype-I.

When seven times true-scale is applied to the the simulation model, Figure 6.8 is obtained. The simulation results in this scale shows similar behavior with the prototype except for the orientation of the first modules that make ground contact. The inner binary hubs of the ground contact modules are not well fixed during the test. A ground-fixing system can solve this problem.

6.1.5. Design of Revised Prototype-I

The design of the revised version is depicted in Figure 6.9. Three main revisions are made: 1) Sheet link computations are made for the deployed pose; 2) module connection part (sheet link, hollow connector and connector) type changes from $SL_{1,3,5}$ to $SL_{2,4,6}$, HC_1 to HC_2 and $C_{1,3}$ to $C_{2,4}$; 3) mechanical limits are added to the module connection parts B_1, HC_2 and $C_{2,4}$. Module connection computations in Equations (5.1)-(5.4) are performed to obtain the $\pi/4$ angle between type-II connected modules for the deployed pose ($\theta_1 = -66.7°$). Hollow connector-binary hub connection (HC_2-B_1) is limited on the binary hub part B_1, which has seven variations for different limits. In addition to the sagging problem, binary hub limits with circular cross-section are also designed to prevent assembly mode change problems.

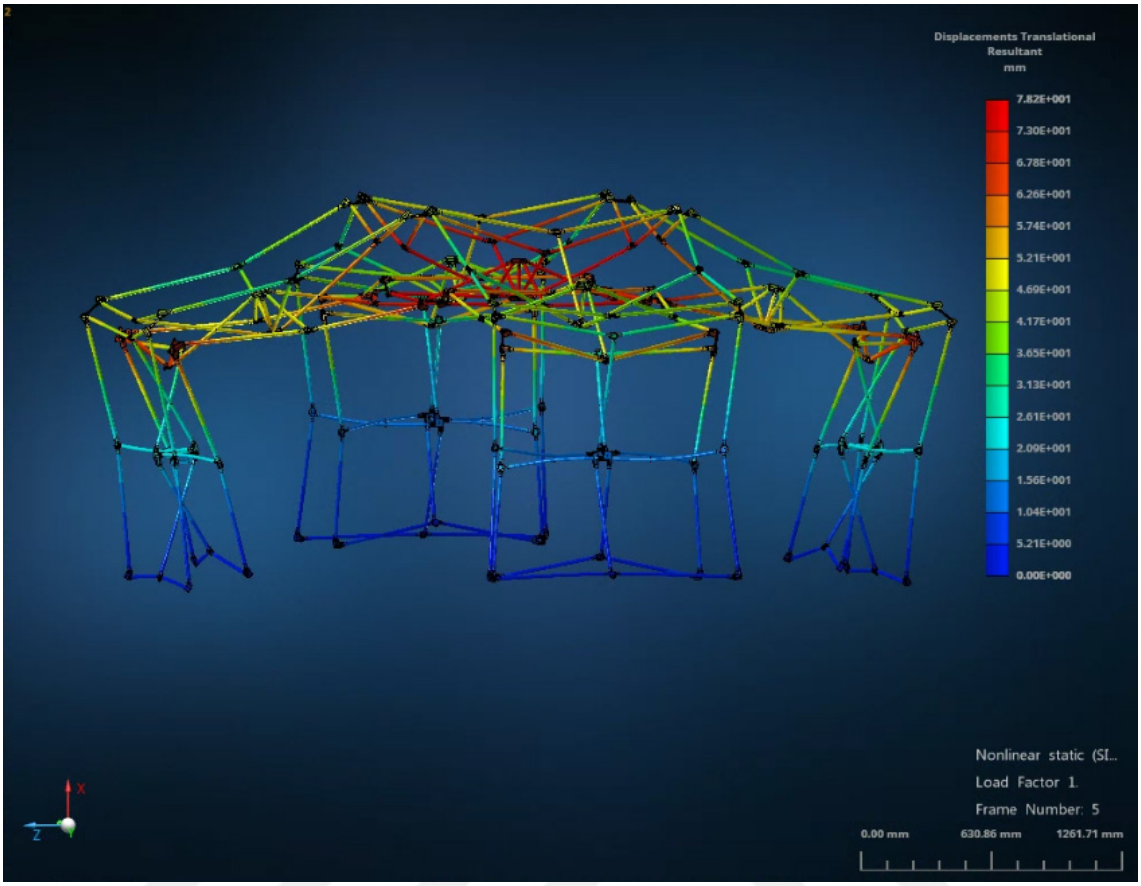


Figure 6.8. Scaled translational displacement results of Prototype-I FEA simulations (true scale = 7).

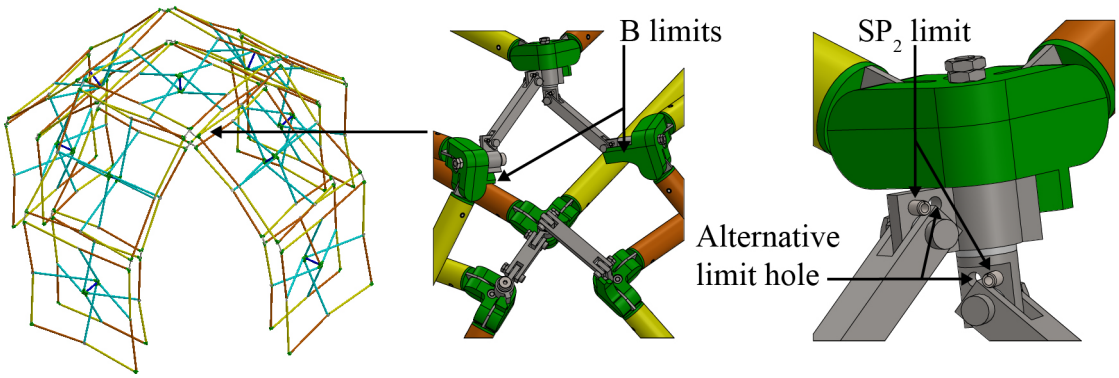


Figure 6.9. Design of revised Prototype-I.

Two alternative slotted spring pin connection holes are designed on hollow connector and connector parts to limit the motion of sheet links with respect to them. These alternatives work for different sense of rotations of the sheet link parts with respect to the hollow connector or connector parts. There are two alternatives for spring pin placement because sagging may occur for the rotation of sheet links towards one of the limits. Alternative limit holes block the motion of the network during deployment. Thus, no spring pin is connected initially, but it can be added to the assembly if desired. FEA simulations give information about which of these alternatives work in the sagging direction. Stiffness elements (springs) or mechanical limits are considered as alternatives to prevent sagging. Preliminary works show that torsion spring forces are too low for the module connection part sizes and have no effect on sagging with the generated forces. Thus, the focus is concentrated on mechanical limits.

When designing mechanical limits to restrict the motion of a revolute joint, it is crucial to consider whether the final position is a global extremum or not. If the end position of the joint is at an intermediate position, then this mechanical limit should be activated/mounted after the mechanism is deployed so that it does not interfere during the deployment. Even in the case of a global extremum, the deployment and sagging must be in the same direction to ensure the limit works against collapsing. Otherwise, the limit pin should be inserted after th assembly.

There are two types of joints in type-II connection: binary hub-hollow connector (R_B) and hollow connector-sheet link (R_{SL}). Figure 6.10a shows type-II connections of the central module (R_{B1} and R_{SL1}). These two joint angles (θ_{B1} and θ_{SL1}) represent all of the upper side module connections. Connected modules or adjacent corner joint angles are ideally equal. Deployment simulation is run using the "Motion Study" toolbox of the CAD environment for 10 s (input angle θ_1 from $\theta_{1i} = 0°$ to $\theta_{1f} = -66.7°$). Figure 6.10b shows the joint angles (θ_{B1} and θ_{SL1}) from folded to deployed form. The results show that the R_{B1} joint is suitable to limit motion in the deployment direction by initially designed mechanical limits (the same motion behaivour is valid for the lower side connections). Type-I connections have the same behavior. But since connector parts ($C_{2,4}$) have circular cross-section, no limit is designed for the binary hub connection. So, R_B joint limits can be designed on the binary hub, but R_{SL} needs to be activated on a hollow connector or a connector in the deployed form.

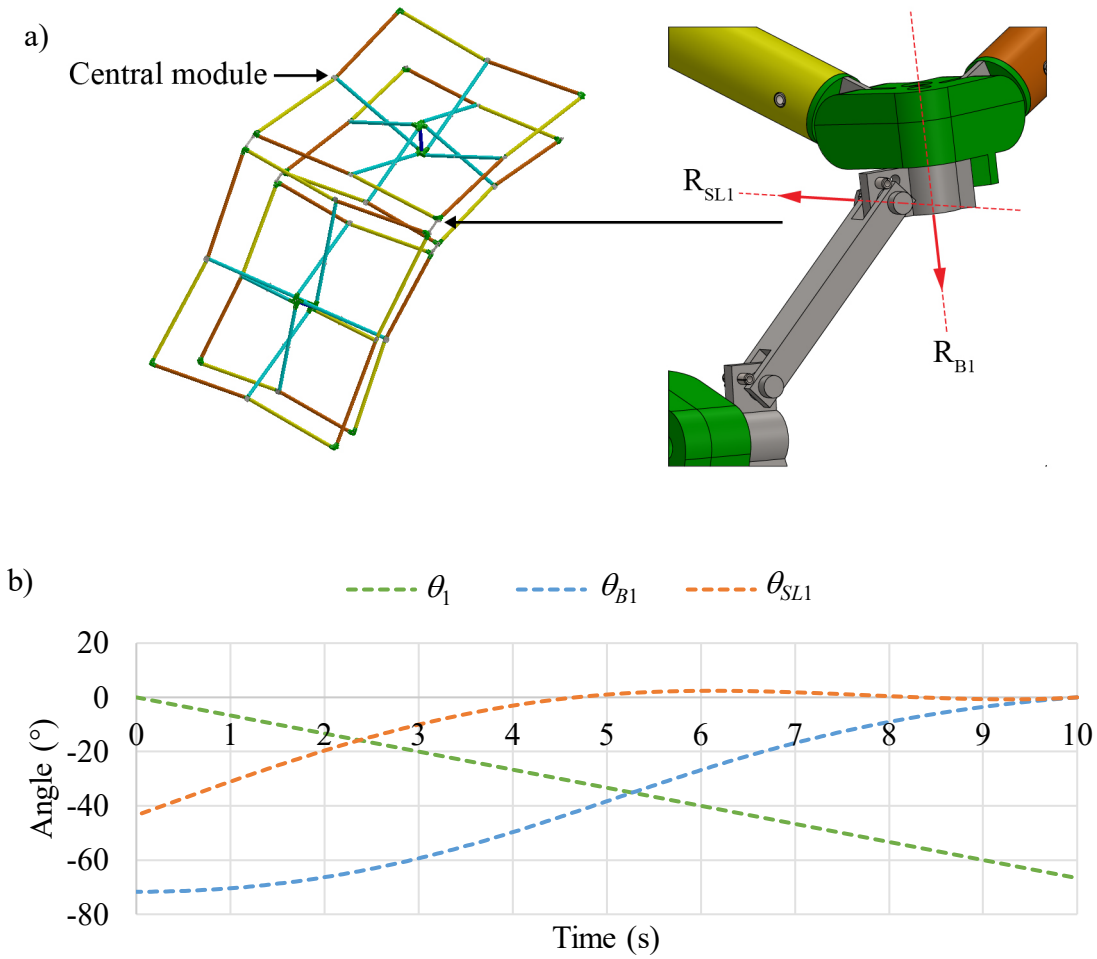


Figure 6.10. Details of modules connection parts in revised Prototype-I a) joints R_{B1} and R_{SL1}, b) angular displacement of θ_1, θ_{B1} and θ_{SL1} during deployment.

6.1.6. FEA Simulations of Revised Prototype-I

Figure 6.11 illustrates the FEA model of the revised design. To decrease the number of elements in the model, meshes are defined at cell/face levels. Boolean and edge offset operations are added to part preprocesses to obtain desired cells/faces. Thus, connection cells/faces of solids/surfaces are fine-meshed while the remaining are coarse-meshed. The number of elements in the model is decreased to 248522 and 636417 for 2D and 3D meshes, respectively. The mechanical limit contacts are not defined at this level to observe the motion of the module connection parts during sagging. Model preprocess

time increased to 3 hours because cell/face-based operations increase the Custom API run-time. Also, the complexity of new binary hub parts has an effect on that.

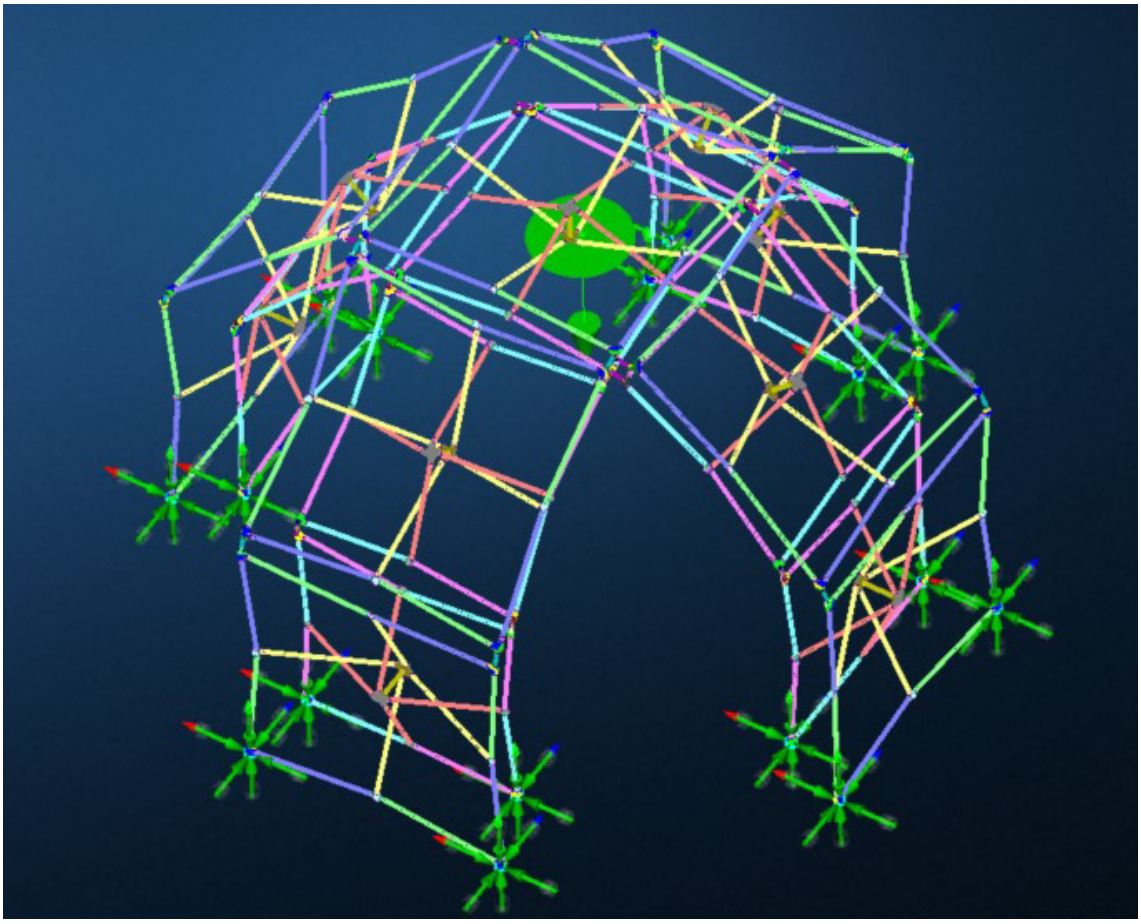


Figure 6.11. FEA simulation model of revised Prototype-I.

The model is run for nonlinear static simulation for the same amount of load (1 g) and solver settings. Simulation results are shown in Figure 6.12 (simulation time: 57 minutes). Sagging occurs without mechanical limit contact definitions as expected. This time, a maximum displacement of 104 mm occurs on the tubes of the central module and a maximum Von misses stress of 820 MPa occurs on the binary hubs. Sagging increased by 25.8 mm in this design. This difference is considered to be due to the altered module connection angles.

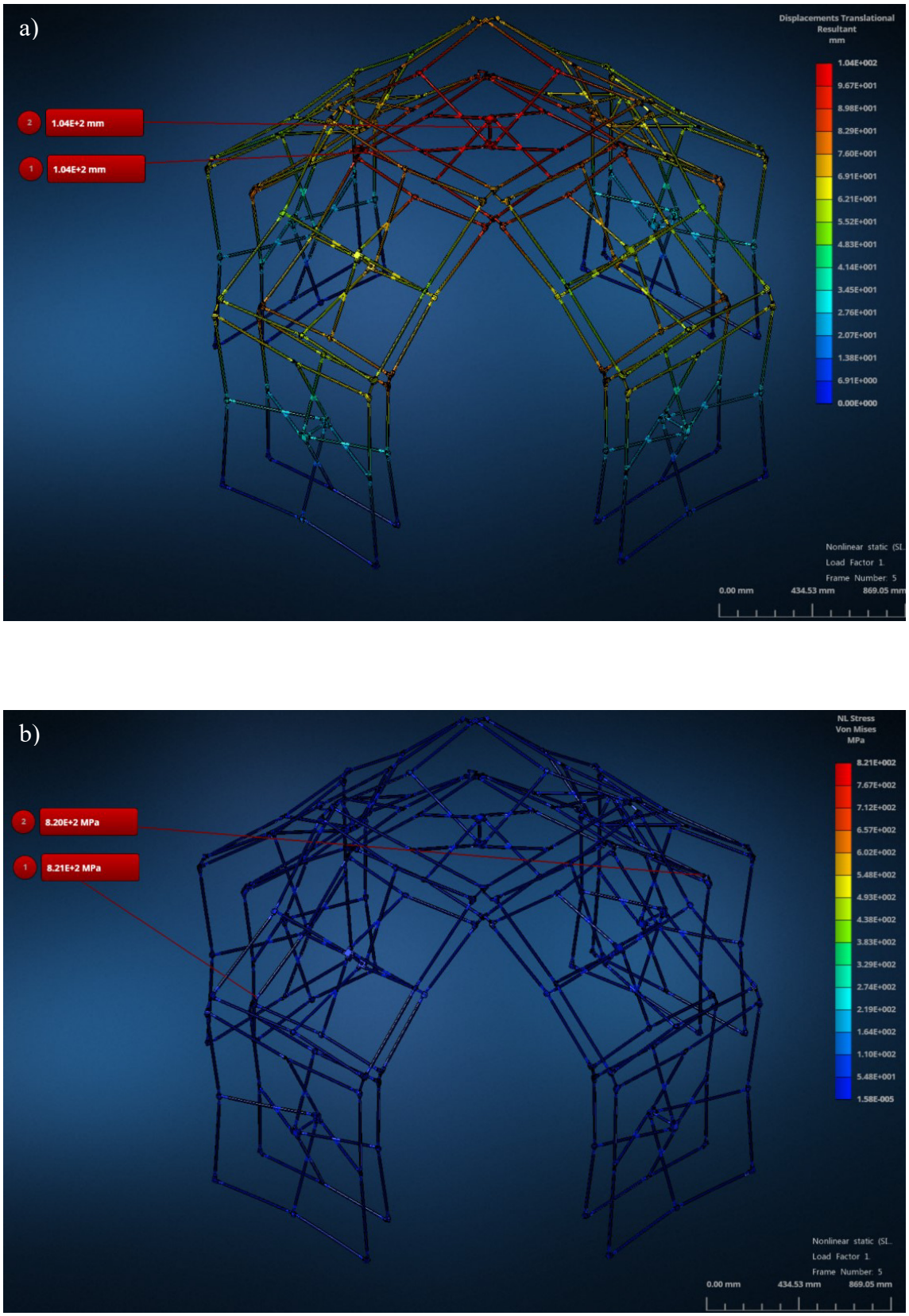


Figure 6.12. FEA simulation results of revised Prototype-I a) translational displacement results, b) Von Mises stress results.

The motions of the mechanical limit parts are investigated in Figure 6.13. Zoomed views of type-I and II module connections are shown for the initial and final poses of the simulation. If a limit is activated, a red circle is drawn to indicate it. Similarly, all of the module connections are examined. The results show that there is no general behavior of the connections, but both limit alternatives are required for different module connections. Limit contacts should be defined for further analysis. The general contact type is suitable for these contacts. Cell-based definitions are required. The type is separable, and initial contact is not required. Friction for sliding and force/stress-based separation criteria can be set.

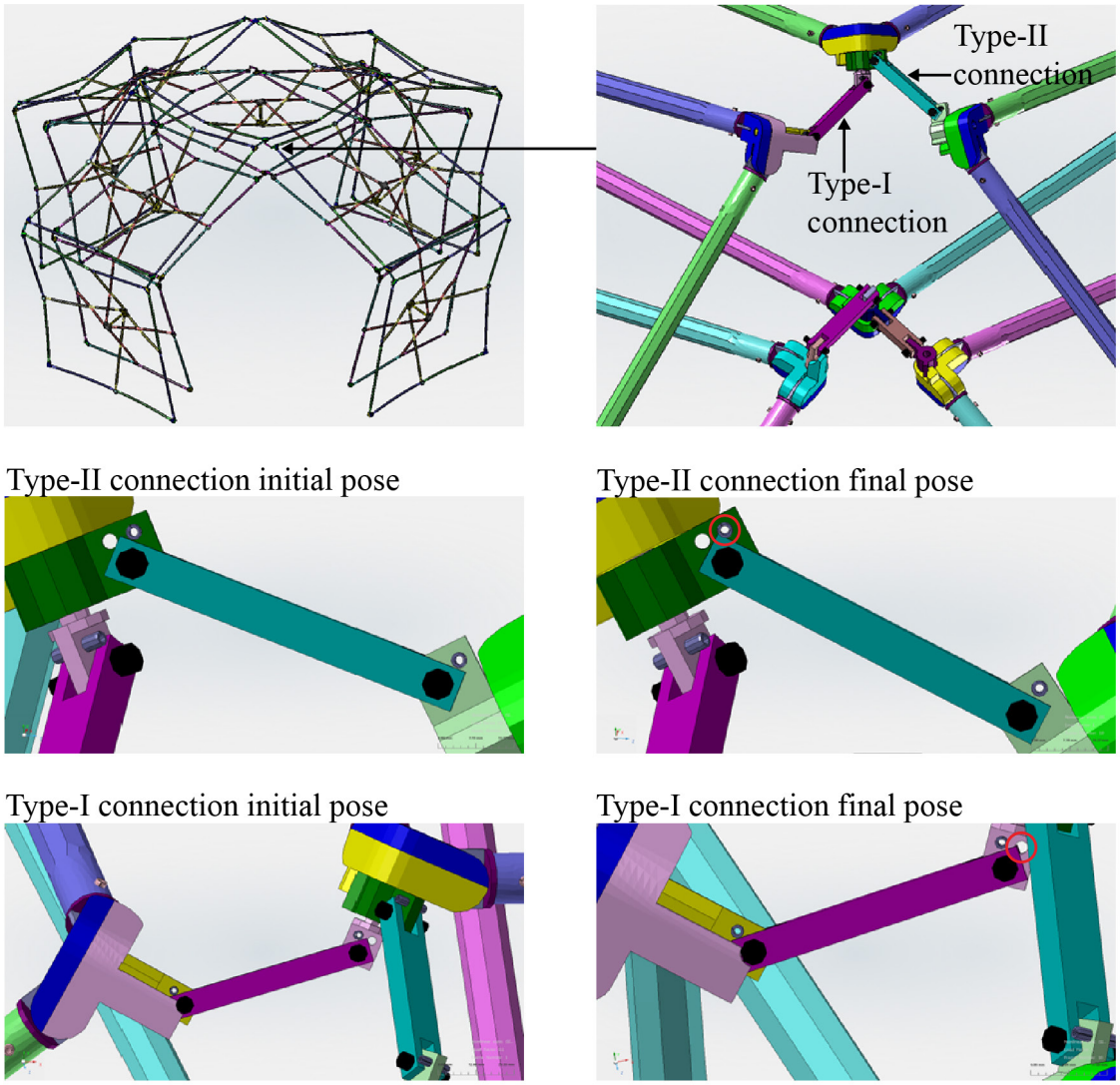


Figure 6.13. Module connection part displacements of revised Prototype-I.

6.2. Prototype-II Design

The CAD model of the Prototype-II is shown in Figure 6.14a. L shaped ground fixing systems are added. Here different trials are made to select link lengths (a_1 and a_2). It is observed that upper side R_{B1} and R_{SL1} joint motion characteristics change to have a global extremum in deployed form by increasing the ratio between a_1/a_2. But the thickness of a module in deployed form also increases due to this. Thus the link lengths are selected as a_1 = 800 mm and a_2 = 650 mm. Deployment simulation results are shown in Figure 6.14b for 10 s (input angle θ_1 from $\theta_{1i} = 0°$ to $\theta_{1f} = -59.7°$). Tube diameter ($\varnothing_{T1}$), tube thickness (t_{T1}) and sheet metal end thickness (t_{S1}) are different than Prototype-I. The tubes are 9.80% lighter due to the change in tube cross-sections, and the sheet metal ends are strengthened. The design provides larger contact surface for the R-joints.

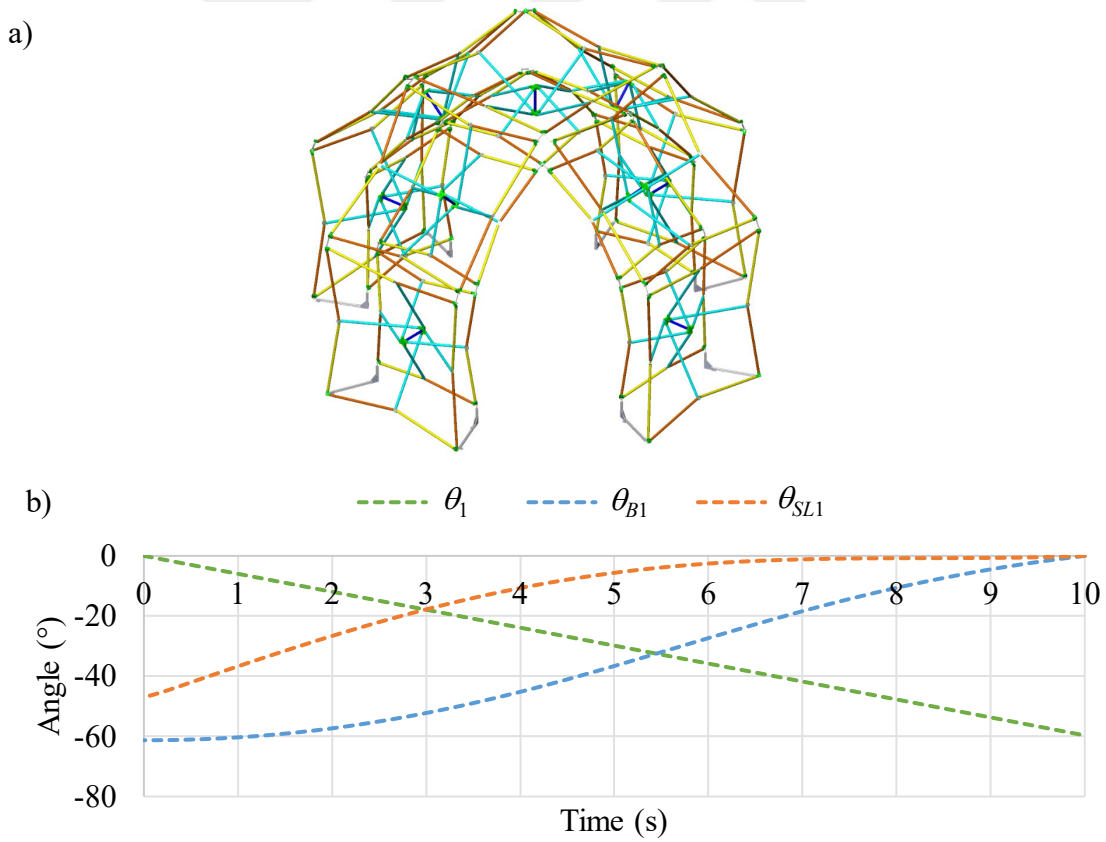


Figure 6.14. Details of Prototype-II a) design b) angular displacement of θ_1, θ_{B1} and θ_{SL1} during deployment.

Design specifications of the prototype and their comparisons with the design criteria are given in Table 6.1. The specifications are computed according to Section 5.6. With the chesen link lengths the floor area and height are close to the criteria limits. Floor area is 2.32%, the height is 4.94%, the deployment ratio is 75.95%, packed volume is 74.10% and the mass is 43.53% safer than the criteria. The mechanical limits are yet to be designed and FEA simulations are to be run as future studies.

Table 6.1. Design specifications of Prototype-II.

Specification	Criteria	Design
Deployment Ratio	≤ 3 %	0.78 %
Floor Area in Deployed Form	≥ 13 m^2	13.3 m^2
Height in Deployed Form	≥ 2.7 m	2.83
Packed Volume	≤ 1 m^3	0.24 m^3
Mass	≤ 85 kg	48 kg

CHAPTER 7

CONCLUSIONS

The first part of the work investigates two modified Altmann linkages and their networks. Networks are classified, and their motions are explored for specific assembly modes. Loop closure equations of two modified Altmann linkages are formulated. Input/output relationships are identified. Special case-II linkage and two of the networks are novel for the literature. Network classification is important to show the potential of modified Altmann linkages for different motions. The thesis focuses on single-layer networks. The work can be extended to multi-layer networks in future studies. Additionally, new modified loops can be investigated to determine whether they are valuable for deployable networks by altering the parameters of Special case-II.

Design of a deployable tent mechanism is presented as a case study. A parametric design environment is developed and FEA processes are automated by custom API designs. The custom API designs can be altered to meet the specifications of any mechanism design. The outcomes present a valuable method for parametric designing and simulating deployable networks with state-of-the-art CAD and simulation softwares. Detailed parametric modeling of deployable networks is not easy but valuable to generate simulation models or a product family. Even a washer parameter can affect a mechanism's deployment as in the case-study model. Likewise, debugging kinematic formulations developed for a deployable network is difficult. PDI add-in helps to overcome these design challenges of parametric modeling. Besides, preprocesses of large-scale deployable network FEA simulations require a lot of modeling time, and a user can easily make mistakes during modeling due to the high number of repetitive tasks. Automation of these processes avoids these mistakes and saves time in modeling. Custom API designs can be further developed to work on multi-models for optimization tasks such as sizing. Multi-model generation feature can added to PDI, and FEA API can adapt to work on these multi-models in a sequence for future studies.

A comprehensive mathematical model and two prototype designs of the case study are presented. Design specifications are formulated according to the needs of a deployable tent. The case-study task is challenging since a geometrically compatible

deployable mechanism is designed for an on-earth application. These type of networks are mostly used in outer-space applications due to negligible gravity. Bistable mechanisms may be preferable for earth applications. But, still a geometrically compatible system offers an opportunity for large-scale applications.

FEA simulations, manufacturing and tests of Protoype-I are presented. The test results show that joints should lock properly to prevent sagging due to module connection parts. Use of mechanical limits are practical if the motion of a joint is monotonic during the deployment. Otherwise, the locking pin should be installed in the deployed form, or a spring-pin lock system can be designed, or an electro-mechanical solution such as a normally closed solenoid valve can be used. Prototype-II simulations and tests may show the effect of mechanical limits. If the preinstalled mechanical limits are not sufficient to prevent sagging, extre locking pins can be installed in the deployed form. Additionally, a telescopic connection with a tension spring can connect two half-modules of each module to increase the stiffness of the modules and synchronise the motion of them.

The kinematic analysis of the 4-UU connections between the modules is not presented in theis thesis. In addition to the mathematical models presented in the thesis, the CAD environment is used to determine the input angle of the deployed pose and the trajectory of the module connection joints. This results in a loss of time during the determination/optimization of the input parameters and checking global extremums of joints for the mechanical limits. Finally, the effect of module connection angles (taken as $\pi/4$ in the final design) and the number of modules (9 in this thesis) is not worked out. Future research may involve the effects of these two parameters and an inverse kinematics solution of the module connection loops.

REFERENCES

Alizade, Rasim. 2010. "Structural Synthesis of Robot Manipulators." In *Proceedings of the AzC IFToMM 2010 International Symposium of Mechanism and Machine Science (IFToMM), 5-8 October, 2010*: 11-32.

Altmann, Fritz G. 1954. "Über raümliche sechsgliedrige koppelgetriebe." *Sonderdruck aus der Zeifschrift des Vereines Deutscher Ingenieure* 96 (8): 245–49.

Arnouts, Liesbeth I. W., Thierry J. Massart, Niels De Temmerman, and Péter Z. Berke. 2019. "Computational Design of Bistable Deployable Scissor Structures: Trends and Challenges." *Journal of the International Association for Shell and Spatial Structures*, 60, no. 1 (March): 19–34. https://doi.org/10.20898/j.iass.2019.199.031.

Atarer, Fulya, Koray Korkmaz, and Gökhan Kiper. 2017. "Design Alternatives of Network of Altmann Linkages." *International Journal of Computational Methods and Experimental Measurements*, 5, no. 4 (June): 495–503. https://doi.org/10.2495/cmem-v5-n4-495-503.

Baker, John E. 1979. "The Bennett, Goldberg and Myard Linkages—in Perspective." *Mechanism and Machine Theory*, 14, no. 4 (January): 239–53. https://doi.org /10.1016/0094-114x(79)90011-9.

Baker, John E. 1980. "An Analysis of the Bricard Linkages." *Mechanism and Machine Theory*, 15, no. 4 (January): 267–86. https://doi.org /10.1016/0094-114x(80)90021-x.

Baker, John E. and Hu Min. 1986. "On spatial networks of overconstrained linkages." *Mechanism and Machine Theory*, 21, no. 5 (January): 427–37. https://doi.org/10.1016/0094-114X(86)90091-1.

Baker, John E. 1993. "A Geometrico-Algebraic Exploration of Altmann's Linkage." *Mechanism and Machine Theory*, 28, no. 2 (March): 249–60. https://doi.org/10.1016/0094-114x(93)90091-9.

Baker, John E. 2011. "A Generalisation of Altmann's Linkage." *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 225, no. 11 (September): 2649–60. https://doi.org/10.1177/0954406211404854.

Baker, John E. 2012. "On the Closure Modes of a Generalised Altmann Linkage." *Mechanism and Machine Theory*, 52 (June): 243–47. https://doi.org/10.1016/j.mechmachtheory.2012.02.001.

Bennett, Geoffrey T. 1903. "A new mechanism." *Engineering* 76: 777.

Bennett, Geoffrey. T. 1914. "The Skew Isogram Mechanism*." *Proceedings of the London Mathematical Society* s2-13 (1): 151–73. https://doi.org/10.1112/plms/s2-13.1.151.

Briand, S. and Zhong You. 2007. New Deployable Mechanisms, Report 2293/07. University of Oxford.

Bricard, Raoul. 1897. “Mémoire Sur La Théorie De L’octaèdre Articulé.” *Journal de Mathématiques Pures et Appliquées*, 5(3): 113–48.

Bricard, Raoul. 1927. *Leçons De Cinématique*. Vol. 2. Gauthier-Villars.

Bricard, Raoul. 2012. “Memoir on the Theory of the Articulated Octahedron.” *arXiv preprint*, 1-40. https://doi.org/10.48550/arXiv.1203.1286.

Bouten, Sam. 2015. " Transformable Structures and their Architectural Application.", Universiteit Gent (Belgium). https://doi.org/10.3929/ethz-a-000240711.

Calatrava, Santiago. 1981. "Zur Faltbarkeit von Fachwerken.", ETH Zurich (Switzerland). https://doi.org/10.3929/ethz-a-000240711.

Chen, Yan. 2003. "Design of Structural Mechanisms." Order No. 10045263, University of Oxford (United Kingdom). https://www.proquest.com/dissertations-theses/design-structural-mechanisms/docview/1775215772/se-2.

Chen, Yan, and Zhong You. 2005. "Mobile assemblies based on the Bennett linkage." *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 461, no. 2056 (April): 1229–45. https://doi.org/10.1098/rspa.2004.1383.

Chen, Yann, and John E. Baker. 2005. "Using a Bennett linkage as a connector between other Bennett loops." *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics*, 219, no. 2 (June): 177–85. https://doi.org/10.1016/0094-114X(86)90091-1.

Chen, Yan, and Zhong You. 2008a. "On mobile assemblies of Bennett linkages." *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 464, no. 2093 (February): 1275–93. https://doi.org/10.1098/rspa.2007.0188.

Chen, Yan, and Zhong You. 2008b. "An Extended Myard Linkage and Its Derived 6R Linkage." *Journal of Mechanical Design*, 130, no. 5 (March). https://doi.org/10.1115/1.2885506.

Chen, Yan, and Zhong You. 2011. *Motion Structures: Deployable Structural Assemblies of Mechanisms*. Taylor & Francis Group. https://doi.org/10.1201/9781482266610/

Cui, Ji, Hailin Huang, and Zongquan Deng. 2012. " A novel surface deployable antenna structure based on special form of Bricard linkages." In *Advances in Reconfigurable Mechanisms and Robots*, edited by Jian S. Dai, Matteo Zoppi and Xianwen Kong. 783-92. London: Springer. https://doi.org/10.1007/978-1-4471-4141-9_70.

Denavit, Jacques, and Richard S. Hartenberg. 1955. "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices." *Journal of Applied Mechanics*, 22, no. 2 (June 1): 215–21. https://doi.org/10.1115/1.4011045.

Deployed Logix. 2021. "DLX // ASAP® Rapid Shelter." YouTube video, 0:35. August 25. https://www.youtube.com/watch?v=cY9mZNgUwtc.

Deployed Logix. 2024. "The ASAP-12 Rapid Shelter." Accessed September 1, 2024. https://deployedlogix.com/asap-rapid-shelter-systems/asap-12/.

Derus, David. L. Collapsable articulated wall structure. US Patent USRE31641E, filled 22 December, 1982 and issued 7 August, 1984.

Escrig, Felix. 1985. "Expandable Space Structures." *International Journal of Space Structures*, 1, no. 2 (June): 79–91. https://doi.org/10.1177/026635118500100203.

Euler, Leonhard. 1758. "Elementa doctrinae solidorum." *Novi commentarii academiae scientiarum Petropolitanae*, 4: 109-140.

Everyspec. 2024. "Test Operations Procedure (TOP) 10-2-175: Tents and Shelters." Accessed September 1. http://everyspec.com/ARMY/Test-Operations-Procedure/download.php?spec=TOP-10-2-175_15JUL2010.054512.pdf.

Excel. 2024. "VBA Reference." Accessed September 1, 2024. https://learn.microsoft.com/en-us/office/vba/api/overview/ excel.

Fenci, Giulia E., and Neil G. Currie. 2017. "Deployable Structures Classification: A Review." *International Journal of Space Structures*, 32, no. 2 (June): 112–30. https://doi.org/10.1177/0266351117711290.

Gantes, Charis J., Jerome J. Connor, Robert D. Logcher, and Yechiel Rosenfeld. 1989. "Structural Analysis and Design of Deployable Structures." *Computers & Structures*, 32, no. 3-4 (January): 661–69. https://doi.org/10.1016/0045-7949(89)90354-4.

Gantes, Charis J. 2018. "Overview of Scissor-type, Bi-stable Deployable Structures." In *Proceedings of International Association for Shell and Spatial Structures (IASS) Annual Symposium: IASS 2018 Boston Symposium: Transformables*, *16-20 July, 2018*, edited by Caitlin Mueller and Sigrid Adriaenssens. 1-8: IASS.

Github. 2024. "Office Ribbonx Editor." Accessed September 1. https://github.com/fernandreu/office-ribbonx-editor.

Guo, Hongwei, and Zhong You. 2012. "Deployable Masts Based on the Bennett Linkage." In *Advances in Reconfigurable Mechanisms and Robots*, edited by Jian S. Dai, Matteo Zoppi and Xianwen Kong. 739-47. London: Springer. https://doi.org/10.1007/978-1-4471-4141-9_66.

Hexagon. 2024. "MSC Apex Object Model Diagrams." Accessed September 1, 2024. https://nexus.hexagon.com/documentationcenter/en-US/bundle/apex_2024.1/page/node/2261.html.

Hibbeler, Richard C. 2010. Mechanics of Material (8th Edition). New York: Pearson Education.

Hoberman, Charles. 2001. Connections to Make Foldable Structures. US Patent US2002083675A1, filled 19 January, 2001 and issued 4 July, 2002.

Hochman, K. I. Kinematics of Machinery (in Russian), 1890.

Huang, Hailin, Zongquan Deng, and Bing Li. 2012. "Mobile Assemblies of Large Deployable Mechanisms." *Journal of Space Engineering* 5 (1): 1–14. https://doi.org/10.1299/spacee.5.1.

Huang, Hailin, Bing Li, Jianyang Zhu, and Xiaozhi Qi. 2016. "A New Family of Bricard-Derived Deployable Mechanisms." *Journal of Mechanisms and Robotics*, 8, no. 3 (March). https://doi.org/10.1115/1.4032119.

Isaak, RaNae R. 2006. " A study of overconstrained networks for morphing applications." Idaho State University (USA).

Kiper, Gökhan, and Eres Söylemez. 2009. "Deployable Space Structures." In *4th International Conference on Recent Advances in Space Technologies (RAST) 11-3 June, 2009*: IEEE. https://doi.org/10.1109/rast.2009.5158183.

Kiper, Gökhan, and Eres Söylemez. 2010. "Obtaining New Linkages from Jitterbug-like Polyhedral linkages." In *Proceedings of the AzC IFToMM 2010 International Symposium of Mechanism and Machine Science (IFToMM), 5-8 October, 2010*: 137-143.

Kiper, Gökhan. 2016. Katlanabilir Kafes Yapı. TR Patent TR201619595B, filled 26 December, 2016 and issued 23 September, 2019.

Krishnapillai, Sivam and Wacław P. Zalewski. 1985. The Design of Deployable Structures. Massachusetts Institute of Technology.

Krishnapillai, Sivam and Edward P. Crawley. 1986. Deployable Structures-Kinematic Design. Massachusetts Institute of Technology.

Li, Yi, and Yanlong L. Xu. 2017. "Tuning and Switching of Band Gap of the Periodically Undulated Beam by the Snap Through Buckling." *AIP Advances*, 7, no. 5 (May): 055210. https://doi.org/10.1063/1.4984607.

Liu, Shiyu Y., and Yann Chen. 2009. "Myard Linkage and Its Mobile Assemblies." *Mechanism and Machine Theory*, 44, no. 10 (October): 1950–63. https://doi.org/10.1016/j.mechmachtheory.2009.05.001.

Mavroidis, Constantinos, and Bernard Roth. 1994. "Analysis and Synthesis of Overconstrained Mechanism." In *ASME 1994 Design Technical Conferences Collocated with the ASME 1994 International Computers in Engineering Conference and Exhibition and the ASME 1994 8th Annual Database Symposium*: American Society of Mechanical Engineers. https://doi.org/10.1115/detc1994-0186.

McCarthy, John Michael, and Gim Song Soh. 2011. *Geometric Design of Linkages*. New York: Springer. https://doi.org/10.1007/978-1-4419-7892-9.

Melin, Nicholas O'Brien. 2004. "Application of Bennett Mechanisms to Long-Span Shelters." Order No. 27795766, University of Oxford (United Kingdom). https://www.proquest.com/dissertations-theses/application-bennett-mechanisms-long-span-shelters/docview/2341981307/se-2.

Myard, Francis Emile. 1931. "Contribution à la géométrie des systèmes articulés." *Bulletin de la Société mathématique de France* 2:183–210. https://doi.org/10.24033/bsmf.1179.

Nodskov, Preben, and Thelander, Finn. 1986. Collapsible Exhibit Panel. US Patent US4580373, filled 18 May, 1984 and issued 8 April, 1986.

Pellegrino, Sergio, ed. 2001. *Deployable Structures*. Vienna: Springer. https://doi.org/10.1007/978-3-7091-2584-7.

Philips, Jack. 2007. Freedom in Machinery: Cambridge University Press. https://doi.org/10.1017/CBO9780511751745.

Pinero, Emilio P. 1965. Three Dimensional Reticular Structure. US Patent US3185164, filled 15 November, 1961 and issued 25 May, 1965.

Qi, Xiao Zhi, Zong Quan Deng, Bing Ying Ma, Bing Li, and Rong Qiang Liu. 2011. "Design of Large Deployable Networks Constructed by Myard Linkages." *Key Engineering Materials*, 486 (July): 291–96. https://doi.org/10.4028/www.scientific.net/kem.486.291.

ReeveEMS. 2024. "DRASH XB Series Shelter." Accessed September 1, 2024. http://www.reevesems.com/Products/ Shelters/XBSeries.aspx.

Roovers, Kelvin. 2017. " Deployable Scissor Grids-Geometry and Kinematics." Vrije Universiteit (Belgium).

Sarrus, Pierre F. 1853. "Note sur la transformation des mouvements rectilignes alternatifs, en mouvements circulaires et réciproquement." *Comptes Rendus des Séances de l'Académie des Sciences de Paris*, 36: 1036–38.

Solidworks. 2024. "SolidWorks API Help." Accessed September 1, 2024. https://help.solidworks.com/2024/English/api/SWHelp_List.html?id=e8d62e64cbfd4b37ba6391dbd11bdf6d&_gl=1*e9dosh*_up*MQ..*_ga*MTA3MzczMjg0Ny4xNzMwNzQ5NDMy*_ga_XQJPQWHZHH*MTczMDc0OTQzMS4xLjEuMTczMDc0OTQ0MS4wLjAuMA..#Pg0.

Song, Chao-Yang, Yan Chen, and I.-Ming Chen. 2014. "Kinematic Study of the Original and Revised General Line-Symmetric Bricard 6R Linkages." *Journal of Mechanisms and Robotics*, 6, no. 3 (April). https://doi.org/10.1115/1.4026339.

Song, Xiaoke, Hongwei Guo, Bing Li, Rongqiang Liu, and Zongquan Deng. 2016. "Large Deployable Network Constructed by Altmann Linkages." *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 231, no. 2 (January): 341–55. https://doi.org/10.1177/0954406215626955.

Tian, Peilin, and Yann Chen. 2010. "Design of a Foldable Shelter." In *Proceedings of the AzC IFToMM 2010 International Symposium of Mechanism and Machine Science (IFToMM), 5-8 October, 2010*: 102-106.

Wohlhart, Karl. 1993. "The Two Types of the Orthogonal Bricard Linkage." *Mechanism and Machine Theory*, 28, no. 6 (November): 809–17. https://doi.org/10.1016/0094-114x(93)90023-o.

Yang, Fu-Fu, Jian-Min Li, Yan Chen, and Zhong You. 2015. "A Deployable Bennett Network in Saddle Surface." Proceedings of the 14th IFToMM World Congress: 428-434: Airiti Library. https://doi.org/10.6567/IFToMM.14TH.WC.OS8.015.

Yang, Yunfang, and Zhong You. 2020. "A Modular Origami-inspired Mechanical Metamaterial." *arXiv preprint*, 1-12. https://doi.org/10.48550/arXiv.2012.09567.

Yang, Yunfang, Perla Maiolino, Yan Chen and Zhong You. 2021. "Three-dimensional Kinematic Metamaterial with tuneable directional permeability." *arXiv preprint*, 1-35. https://doi.org/10.48550/arXiv.2102.08821.

You, Zhong, and Yan Chen. 2002. "Connectivity of Bennett Linkages." In *43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. *22-25 April, 2002*. Virigina: American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2002-1500.

Zeigler, Theodore R. 1984. Collapsable Self-Supporting Structures. US Patent US4437275A, filled 7 August, 1981 and issued 20 March, 1984.

Zhao, Zhanwei, and Lei Yu. 2024. "Geometric Design Methodology for Deployable Self-Locking Semicylindrical Structures." *Buildings*, 14, no. 6 (June): 1690. https://doi.org/10.3390/buildings14061690.

Zhang, Xiao, and Yan Chen. 2018. "The Diamond Thick-Panel Origami and the Corresponding Mobile Assemblies of Plane-Symmetric Bricard Linkages." *Mechanism and Machine Theory*, 130 (December): 585–604. https://doi.org/10.1016/j.mechmachtheory.2018.09.005.

VITA

Murat Demirel graduated from the Bachelor of Science program in Mechanical Engineering at İzmir Institute of Technology in September 2015. As part of his undergraduate studies, he participated in the Erasmus+ Exchange Program at Fachhochschule Südwestfalen, Germany, in the Spring Semester of 2013. In the fall semester of 2013, he joined the Rasim Alizade Mechatronics Laboratory (RAML) as an undergraduate student member, where he began developing his research skills under the supervision of Dr. Gökhan Kiper.

He pursued his Master of Science in Mechanical Engineering at the same institution starting in September 2015, focusing on biped walking machines and cable-constrained parallel manipulators. For his master's works, he joined the Erasmus+ Exchange Program at Università degli Studi di Cassino e del Lazio Meridionale, Italy, in the Spring Semester of 2017, where he conducted research at Laboratorio di Robotica e Meccatronica (LARM) under the supervision of Dr. Marco Ceccarelli and Dr. Giuseppe Carbone. He completed his thesis titled "Design of Hybrid Cable-Constrained Parallel Mechanisms for Walking Machines" under the supervision of Dr. Gökhan Kiper and Dr. Giuseppe Carbone in July 2018. The outcomes were published with the title "Design of a Novel Hybrid Cable-Constrained Parallel Leg Mechanism for Biped Walking Machines" in Robotica journal.

He began his Ph.D. studies in Mechanical Engineering at İzmir Institute of Technology in September 2018, focusing on deployable structures and over-constrained mechanisms. Murat completed his doctoral dissertation, "Design, Prototyping and Testing of Modified Altmann Linkage Networks to Be Used as Deployable Structures," under the supervision of Dr. Gökhan Kiper in December 2024.

Since January 2016, he has been working as a research assistant in the Mechanical Engineering Department at İzmir Institute of Technology. Murat is a member of the International Federation for the Promotion of Mechanism and Machine Science (IFToMM) and currently serves as the Secretary General of the Turkish Machine Theory Association (MakTeD).