

VARIABLE GEOMETRY WING-BOX: TOWARD A ROBOTIC MORPHING WING

by

Amin Moosavian

B.Eng., Aerospace Engineering, Ryerson University, 2008

M.Eng., Aerospace Engineering, Ryerson University, 2010

A dissertation

presented to the School of Graduate Studies at

Ryerson University

in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
in the Program of
Aerospace Engineering

Toronto, Ontario, Canada, 2014

Copyright © Amin Moosavian 2014

Author's Declaration

I hereby declare that I am the sole author of this dissertation. This is a true copy of the dissertation, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

Abstract

Variable Geometry Wing-box: Toward a Robotic Morphing Wing

© Amin Moosavian, 2014

Doctor of Philosophy

Aerospace Engineering

Ryerson University

The ability to vary the geometry of a wing to adapt to different flight conditions can significantly improve the performance of an aircraft. However, the realization of any morphing concept will typically be accompanied by major challenges. Specifically, the geometrical constraints that are imposed by the shape of the wing and the magnitude of the air and inertia loads make the usage of conventional mechanisms inefficient for morphing applications. Such restrictions have served as inspirations for the design of a modular morphing concept, referred to as the Variable Geometry Wing-box (VGW).

The design for the VGW is based on a novel class of reconfigurable robots referred to as Parallel Robots with Enhanced Stiffness (PRES) which are presented in this dissertation. The underlying feature of these robots is the efficient exploitation of

redundancies in parallel manipulators. There have been three categories identified in the literature to classify redundancies in parallel manipulators: 1) actuation redundancy, 2) kinematic redundancy, and 3) sensor redundancy. A fourth category is introduced here, referred to as 4) static redundancy. The latter entails several advantages traditionally associated only with actuation redundancy, most significant of which is enhanced stiffness and static characteristics, without any form of actuation redundancy. Additionally, the PRES uses the available redundancies to 1) control more Degrees of Freedom (DOFs) than there are actuators in the system, that is, under-actuate, and 2) provide multiple degrees of fault tolerance. Although the majority of the presented work has been tailored to accommodate the VGW, it can be applied to any comparable system, where enhanced stiffness or static characteristics may be desired without actuation redundancy.

In addition to the kinematic and the kinetostatic analyses of the PRES, which are developed and presented in this dissertation along with several case-studies, an optimal motion control algorithm for minimum energy actuation is proposed. Furthermore, the optimal configuration design for the VGW is studied. The optimal configuration design problem is posed in two parts: 1) the optimal limb configuration, and 2) the optimal topological configuration. The former seeks the optimal design of the kinematic joints and links, while the latter seeks the minimal compliance solution to their placement within the design space. In addition to the static and kinematic criteria required for reconfigurability, practical design considerations such as fail-safe requirements and design for minimal aeroelastic impact have been included as constraints in the optimization process. The effectiveness of the proposed design, analysis, and optimization is demonstrated through simulation and a multi-module reconfigurable prototype.

Acknowledgements

Foremost, I want to thank my supervisor Prof. Jeff Xi. I am extremely grateful to him for giving me the opportunity to work on this project and for providing guidance throughout my studies. His encouraging attitude and openness to new ideas along with his comprehensive knowledge of robotics and mechanisms have been immensely inspiring and instrumental to what has been accomplished. Furthermore, I want to thank my co-supervisor, Prof. Seyed Hashemi for all the guidance, helpful feedback, and insightful discussions during the course of this work. I have the utmost respect and gratitude for both of them, personally and professionally. I have learnt a tremendous amount from them over the years, and consider myself lucky to have studied under their supervision.

I wish to express my gratitude to the defense committee, Prof. Brian Surgenor from Queens's University, Prof. Vincent Chan from the Dept. of Mechanical Engineering, Prof. Puren Ouyang, and Prof. Anton de Ruiter from the Dept. of Aerospace Engineering, for all their invaluable remarks and suggestions regarding this thesis.

This work was funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) and Bombardier Aerospace (BA). I would like to thank NSERC, and BA, especially Dr. Fassi Kafyeke and Dr. Patrick Germain, for all their support. I would also like to thank the government of Ontario and Ryerson University for the

various Ontario and Ryerson graduate scholarships and awards which helped to support this research.

Furthermore, I would like to thank my colleagues at the Intelligent Systems and Robotics Laboratory for their support. A special thanks goes out to my good friend and colleague, Mike Rizoiu, for all the stimulating conversations we had about this project and his assistance with the prototype.

Finally, a very special thanks goes out to my wonderful family, especially my amazing parents, and my lovely fiancée, Marlena, for their vast support and continuous encouragement.

To My Parents

*“It is possible to fly without motors,
but not without knowledge and skill.”*

Wilbur Wright

Contents

Author’s Declaration	iii
Abstract	iv
Acknowledgements	vi
Contents	x
List of Tables	xv
List of Figures	xvi
Acronyms	xxi
Nomenclature	xxii
Chapter 1 Introduction	1
1.1 Motivation.....	3
1.2 Objectives.....	3
1.3 Variable Geometry Wing-box	4
1.4 Thesis Organization	6
Chapter 2 Background and Literature Review	7
2.1 Morphing DOFs	10
2.1.1 Variable Sweep	11
2.1.2 Variable Span	12

2.1.3	Variable Twist	12
2.1.4	Variable Dihedral (Cant).....	12
2.2	Airfoil Morphing	13
2.3	Morphing Skins	14
2.4	Morphing for Control vs. Improved Performance.....	15
2.5	Actuation Concepts.....	16
2.5.1	VGTMs for Wing Morphing.....	18
2.5.2	Embedded Actuation.....	20
Chapter 3 Morphing Mechanism Design		22
3.1	Reconfigurability	23
3.2	Manipulator Redundancy.....	25
3.3	Robot Design.....	29
3.4	Class Synthesis.....	35
3.5	Architecture Design	37
3.6	Modularity in Design	42
3.7	Conclusion.....	50
Chapter 4 Kinematics		51
4.1	Local Kinematics.....	52
4.2	Global Kinematics.....	54
4.3	Under-actuation	56
4.3.1	Kinematic Constraints.....	57

4.3.2	Under-actuation with Virtual Alternating Constraints	60
4.3.2.1	UVAC for a $\mathcal{P}2\text{-}2$ PRES	62
4.3.3	Joint Placement for Under-actuation	70
4.4	Conclusion	72
Chapter 5 Kinetostatics		73
5.1	Local Stiffness (Module Level)	73
5.2	Local Statics	82
5.3	Global Stiffness (Wing Level)	84
5.4	Global Statics	87
5.4.1	Intrinsic Loads	88
5.4.2	Extrinsic Loads	90
5.5	Geometry Dependent Loading	91
5.6	Alternative Jacobian	93
5.7	Case Study: $\mathcal{S}\text{-}4\text{-}4$ PRES vs. Gough-Stewart vs. Pentapod	95
5.8	Conclusion	100
Chapter 6 Optimal Configuration Design		101
6.1	Optimal Limb Configuration	102
6.2	Optimal Topological Configuration	110
6.2.1	Symmetry	110
6.2.2	Optimization Formulation	112
6.2.2.1	Minimal Ground Topology (MGT)	117

6.2.2.2	Optimal Hyperstatic Topology (OHT)	121
6.2.2.3	Optimal Isostatic Topologies	124
6.3	Practical Realization	129
6.4	Rigidity Validation	132
6.5	Alternative Configuration	134
6.6	Conclusion.....	135
Chapter 7 Optimal Motion Planning		137
7.1	Control Implementation Scheme.....	137
7.2	Actuation Paths	139
7.3	Minimum Energy Actuation	142
7.4	Control Implementation.....	149
7.5	Variable Topology and Internal Loads	149
7.6	Case Study: Simulation, Validation, and Implementation.....	153
7.6.1	Simulation.....	155
7.6.2	Motion Control Validation	159
7.6.3	External Loads.....	160
7.6.4	MATLAB® Interface/Implementation	161
7.7	Conclusion.....	163
Chapter 8 Concluding Remarks		164
8.1	Summary of Contributions.....	165
8.2	Future Work	166

Appendix A VGW Under-actuation	169
Appendix B VGW Kinetostatics.....	176
Appendix C Alternative Stiffness Model Using FEM.....	179
C.1 Limb Element Formulation.....	180
C.2 Platform Element Formulation.....	183
C.3 Stiffness Matrix Assembly.....	187
Appendix D Designing for Uncertainties	191
D.1 Designing for Failure (Internal Uncertainties).....	191
D.1.1 Functional Failures.....	192
D.1.2 Safety Failures	193
D.2 Designing for Uncertainty in Loading (External Uncertainties).....	194
D.2.1 Passive Member Design	195
D.3 Sizing Guidelines	199
D.4 Conclusion.....	199
Appendix E SC(2)-0518 Airfoil Definition	201
Bibliography	203

List of Tables

Table 3.1: Classification of PRES's and their corresponding static and kinematic characteristics.	36
Table 3.2: Minimal architecture criteria for PRES's in \mathbb{R}^3 and \mathbb{R}^2	41
Table 3.3: Possible solutions for modularity with various degrees of static redundancy.	49
Table 6.1: Kinematic pairs: description and schematic.	104
Table 6.2: Connectivity permutations with three or less joints.	105
Table 6.3: Possible architectures with three joints.	106
Table 7.1: 16 Actuation sequences used for optimal motion control.	141
Table 7.2: Six scenarios capturing all possible combinations of loading conditions and motion requirements.	147
Table 7.3: Different views showing the wing shape by stage for the case-study. ...	156
Table 7.4: Pose variables for the case-study.	157
Table 7.5: Forces and stroke lengths by stage for the case-study.	157
Table 7.6: Actuation energy by stage for the case-study.	158
Table 7.7: Locking sequence for the passive members for the case-study.	158
Table 7.8: Schematics of limb types/configurations.	162
Table D.1: Four positions generated using two solenoid valves.	196
Table D.2: Physical interpretations for the four circuit configurations.	196
Table E.1: SC(2)-0518 airfoil definition.	201

List of Figures

Figure 1.1: Wright brothers' morphing wing concept (the Wright Flyer) [1].	2
Figure 1.2: Proposed modular morphing wing with six morphing DOFs per module:VGW.	5
Figure 2.1: Morphing classification based on the parameter of interest.	8
Figure 2.2: Four main DOFs associated with wing morphing.	8
Figure 2.3: Examples of high profile morphing aircraft [42-48].	14
Figure 2.4: Aircraft coordinates.	17
Figure 2.5: Conventional wing-box setup, capable of reacting shear, torsion, and bending using a combination of spars, skins, and stringers.	18
Figure 3.1: Geometric and topological approaches to change the static and stiffness characteristics of a PKM.	24
Figure 3.2: Four types of redundancies associated with parallel manipulators.	27
Figure 3.3: A Planar PRES with two actuators and two lockable passive limbs.	30
Figure 3.4: Three topologies effectively realized using the ground topology presented in Fig. 3.3.	31
Figure 3.5: Actuation procedure for a planar PRES with two actuators and two lockable passive limbs.	32
Figure 3.6: Examples of planar and spatial PRES's.	34
Figure 3.7: Examples of modular and non-modular planar PRES's.	42

Figure 3.8: Joint type breakdown of the active and passive limbs making up a PRES.	44
Figure 3.9: Examples of transformation of PRES's with hybrid limbs into functionally equivalent PRES's with only active and passive limbs.	45
Figure 4.1: Local kinematic model of the VGW.....	52
Figure 4.2: Global kinematic model of the VGW.....	55
Figure 4.3: Two isostatic and one hyperstatic topologies can be generated for the \mathcal{P} -2-2 PRES.....	62
Figure 4.4: Kinematic model of a \mathcal{P} -2-2 PRES.	63
Figure 4.5: Two distinct paths used to achieve under-actuation for the \mathcal{P} -2-2 PRES.	65
Figure 4.6: Virtual and actual kinematic constraints used in UVAC, pose = ${}^i\mathbf{X}_{i+1}^{Path 1}$	66
Figure 4.7: Under-actuation procedure for a planar mechanism with two actuators and two passive members using path 1.	68
Figure 4.8: Under-actuation procedure for a planar mechanism with two actuators and two passive members using path 2.	69
Figure 4.9: Topology effect on ability to under-actuate.....	71
Figure 5.1: Loop closure diagram for the VGW.....	74
Figure 5.2: View looking aft on the wing showing the intrinsic loads discretized by module.	89
Figure 5.3: Proposed variable topology robot with lower instantaneous mobility, full finite mobility, and enhanced stiffness and static characteristics, classified as an \mathcal{S} -4-4 PRES.	95
Figure 5.4: Schematic representation of the proposed robot.	96

Figure 5.5: Six different isostatic topologies used in the case-Study, four of which are generated from the hyperstatic ground topology of the proposed robot.	97
Figure 5.6: Variation of minimum stiffness as a function of topology and pose (rotation about the z axis).....	99
Figure 6.1: An example of a limb architecture that meets the necessary enumeration condition but has less than six non-redundant DOFs.	105
Figure 6.2: Highly geometrically constrained design with folding limbs	107
Figure 6.3: Configuration with the optimal limb architecture (from the \mathcal{SPS} family).	109
Figure 6.4: Representation of the cuboid design space.....	113
Figure 6.5: Three classes of structural optimization.....	114
Figure 6.6: Design variables for the shape optimization problem.	115
Figure 6.7: Summary of the three-stage optimization process.....	116
Figure 6.8: Examples of optimal solutions to the single-DOF sub-problems.	119
Figure 6.9: Equally optimal configurations with maximum bending stiffness (stiffness about the x axis).....	120
Figure 6.10: Union of the edge and the face diagonal members forming the minimal ground topology.....	120
Figure 6.11: Globally optimal hyperstatic topologies.	123
Figure 6.12: Numbering convention for the limbs of the VGW.....	124
Figure 6.13: Six-member configuration corresponding to set \mathbf{s}_1	125
Figure 6.14: Four optimal isostatic topologies used for under-actuation.	129
Figure 6.15: Schematic representation of the optimal configuration.	130
Figure 6.16: CAD representation of the optimal configuration for the VGW (prototype).	131
Figure 6.17: Two-module prototype built for validation.	132

Figure 6.18: Rigidity validation for the optimal configuration (view looking aft on the prototype).....	133
Figure 6.19: Schematic representation of the optimal configuration with two actuators and lower mobility.....	134
Figure 7.1: Typical mission profile for a commercial aircraft.....	138
Figure 7.2: Four isostatic topologies used for actuation.....	140
Figure 7.3: Representation of the stroke length and force associated with various stages of under-actuation.....	146
Figure 7.4: Internal loads under ${}^i\mathbf{F}_{i+1} = [F_x \ 0 \ 0 \ 0 \ 0 \ 0]^T$	150
Figure 7.5: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ F_y \ 0 \ 0 \ 0 \ 0]^T$	150
Figure 7.6: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ F_z \ 0 \ 0 \ 0]^T$	151
Figure 7.7: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ 0 \ M_x \ 0 \ 0]^T$	151
Figure 7.8: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ 0 \ 0 \ M_y \ 0]^T$	152
Figure 7.9: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ 0 \ 0 \ 0 \ M_z]^T$	152
Figure 7.10: Wing tip layout in initial pose (view looking down on the left wing).	153
Figure 7.11: Initial and final configurations.	155
Figure 7.12: Simulation for minimum energy actuation.....	156
Figure 7.13: Side and front views of the prototype showing poses 0, 1, 2, and 3..	159
Figure 7.14: Snapshot of the MATLAB® GUI used for pose input.	162
Figure 7.15: Schematic illustration of the VGW undergoing the under-actuated motion with the status of each member identified for each step using the scheme presented in Table 7.8.	163
Figure C.1: FE representation for one module of the VGW.	179
Figure C.2: Typical limb element with the associated DOFs.....	180

Figure C.3: Rigid element used to model the platform with the corresponding node numbering.....	185
Figure D.1: Eight fail-safe topologies representing all possible one-member configurations that could be tolerated in the event one of the limbs loses its load-bearing capability permanently.	193
Figure D.2: Proposed circuit for directional flow control of a self-circulating cylinder.	196
Figure D.3: Cutaway view of the proposed valve schematic in four positions to enable directional locking of a self-circulating cylinder.	197
Figure D.4: The implementation of the directional lockability for one lockable cylinder.	198
Figure D.5: Summary of approaches used in the VGW design to address internal and external uncertainties.	200
Figure E.1: Illustration of the SC(2)-0518 airfoil.	202

Acronyms

AAW	Active Aeroelastic Wing
DOF	Degree Of Freedom
FAR	Federal Aviation Regulation
FBW	Fly-By-Wire
FEM	Finite Element Method
MAW	Mission Adaptive Wing
MDO	Multi-disciplinary Design Optimization
MPC	Multi Point Constraint
PALM	Passive Alternating Lockable Member
PKM	Parallel Kinematic Manipulator
PRES	Parallel Robot with Enhanced Stiffness
SMA	Shape Memory Alloy
SPC	Single Point Constraint
UAV	Unmanned Aerial Vehicle
UVAC	Under-actuation with Virtual Alternating Constraints
VGTM	Variable Geometry Truss Manipulator
VGW	Variable Geometry Wing-box
VTM	Variable Topology Manipulator

Nomenclature

\mathbf{b}	module position vector
\mathbf{C}	configuration matrix
C	branch connectivity
c	chord length
\mathbf{D}	displacement vector
D	drag
d	translational displacement
\mathbf{F}	external load vector
F	force component of external load vector
\mathcal{F}	branch constraints
\mathbf{f}	internal load vector
f	limb force
i	module number
\mathbf{J}	Jacobian matrix
j	limb number
\mathbf{K}	stiffness matrix
k	limb stiffness
L	lift
\mathcal{L}	length parameter
l	length

M	moment component of external load vector
\mathcal{M}	mobility
m	DOF number
\mathbf{N}	connectivity matrix
N	summation scalar
n	number of load-bearing limbs
\mathcal{O}	coordinate system
P	joint identification
\mathbf{p}	platform vector
\mathbf{Q}	vector of limb lengths
\mathbf{q}	limb vector
q	limb length
\mathbf{R}	rotation matrix
S	wing area
s	actuation stage
\mathbf{T}	transformation matrix
t	time
v	actuator speed
\mathbf{W}	weighting matrix
W	weight
\mathcal{W}	work
\boldsymbol{w}	bias scalar
w	distribution function
\mathbf{X}	module pose vector
α	twist

Γ	dihedral/cant
ε	actuation energy
η	normalized wing span
θ	rotation angle
κ	modified limb stiffness
Λ	sweep
$\boldsymbol{\rho}$	simplified platform vector
ξ	degree of static redundancy
ϕ	rotational displacement
Υ	Cartesian motion parameter

Subscripts

A	related to an actuated joint/member/limb
C	related to the Cartesian space
D	dependent
e	related to an element or the e^{th} member
g	related to the g^{th} node
hyp	hyperstatic
I	isostatic or independent
i	related to the i^{th} module/platform
iso	isostatic
J/j	related to a joint/joint space or the j^{th} limb/member
l	related to a link/locked passive joint/member
P	related to a passive joint/member/limb
q	related to the joint space
R	redundant

s	actuation stage
u	related to an unlocked passive joint/member/limb
X	related to the Cartesian space
x	along/about x axis
y	along/about y axis
z	along/about z axis

Superscripts

A	related to an actuated joint
hyp	hyperstatic
i	corresponding coordinate frame/module number
iso	isostatic
PL	related to a lockable passive joint
PR	related to a regular passive joint

Chapter 1

Introduction

Traditionally, aircraft design represents a fine balance between performance, weight, and cost. The aerodynamicist strives for a design with minimal drag, the structural engineer strives for maximized stiffness and minimal weight, while the manufacturing partner desires to achieve the lowest cost of manufacturing and ease of assembly. The same principle of competing objectives applies to the design of a wing or any other lift producing surface on the aircraft. Conventionally, the engineer is faced with the following problem: generate a wing configuration that will maximize the performance index of the aircraft over one or a series of typical mission profiles. A typical index for this purpose could be the L/D , that is, the lift to drag ratio. The solution to this problem is usually a wing which is optimized for one configuration, typically the cruise configuration, in the case of most commercial and business aircraft. However, the configurations that give the best solutions, for example, for landing and take-off are drastically different than those for cruise.

One could imagine the possibilities created with the freedom to generate different wing configurations, each suited for a particular flight condition or a segment of the mission profile, or used to control the aircraft. This is a goal as old as manned aviation itself. In fact the Wright flyer, flown over a century ago utilized a warping

wing design (Fig. 1.1). This comes at no surprise as engineers have always been seeking inspirations from nature, and this is precisely the way nature behaves. For example, birds change the shape of their wings for control and to adapt to different flight conditions.

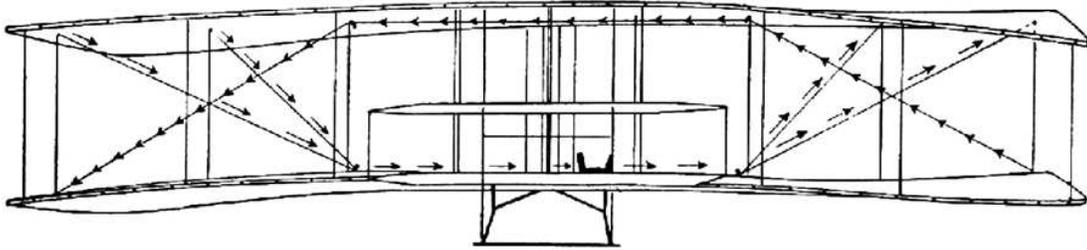


Figure 1.1: Wright brothers' morphing wing concept (the Wright Flyer) [1].

In practice, the further realization of a warping wing, or more generally the concept of bird-like morphing, proved to be a very challenging feat for aircraft engineers, and was eventually abandoned in favor of fixed-wing designs. However, the obsession with morphing wings has only grown with time. To realize the benefits of discrete shape changes engineers invented different means to partially mimic this natural phenomena of morphing. Some of these included the addition of hinged surfaces such as ailerons, used for control, and high-lift devices such as slats and flaps to the wings. In general, there is some ambiguity as to what constitutes morphing. From a categorization point of view, it is the common consensus that the previously noted hinged surfaces and high-lift devices are not considered morphing features due to their conventional nature. However, from a fundamental perspective, morphing is exactly what they do. The less conventional approaches to change the

shape of a wing during flight through planform and out-of-plane alterations as well as the shape of an airfoil are generally accepted as morphing.

1.1 Motivation

In wing morphing it is desirable to have a system that acts both as a load-bearing structure and a morphing mechanism, without any distinction between the two. When dealing with a rather large design space, designing for this requirement would be less challenging as the constituting structural elements could be topologically placed in a fashion to enhance the static characteristics of the system, without sacrificing its kinematic abilities. However, in the case of an aircraft wing, where the design space is highly restrictive, conventional approaches typically yield inefficient designs from a static perspective. Such restrictions have served as motivations for the proposed concept of a reconfigurable system, which is able to alter its kinematic and static characteristics to act both as a mechanism and a high-stiffness structure.

1.2 Objectives

In the current era the push for less conventional wings capable of morphing during flight beyond the conventional means of hinged surfaces and high-lift devices, seems to be exhibiting no signs of slowing down. In fact, there is no deficit of ideas and concepts for the design of morphing wings. However, from a practical perspective, there are still major areas that are currently attracting immense research focus in this field as identified by Reich and Sanders [2]. Amongst those are structural mechanization, and control law development, which are of significant interest to the current research efforts. The objectives of this thesis can be formally categorized into three groups:

- **Design:** design a new type of morphing system through the novel approach of separating static and kinematic requirements.
- **Analysis:** develop the kinematic and static relationships of the proposed type of morphing system.
- **Optimization:** develop the optimal configuration and motion control for the proposed morphing system.

1.3 Variable Geometry Wing-box

By replacing the primary structural elements of a conventional wing-box with a series of discrete load-bearing members, some of which are able to actively change their geometry, one can obtain a wing with variable geometry. This transformation will effectively turn the conventional semi-monocoque design of the wing-box into a variable truss-like structure, which herein will be referred to as the Variable Geometry Wing-box (VGW). The hybrid mechanism-structure design is possible due to the reconfigurable nature of the system, minimizing the drawbacks that have been raised in the literature about the weight penalties of actively actuated morphing wings.

The VGW consists of actively actuated modules that are laid out in a serial fashion. The number of modules and the discretization depend on the particular application. Each module consists of a series of kinematic chains which will enable the ribs or the so-called platforms to change their pose. These platforms are also used to connect the modules, as illustrated in Fig. 1.2. The VGW can provide morphing capabilities in all six Cartesian Degrees of Freedom (DOFs).

Unlike conventional Variable Geometry Truss Manipulators (VGTMs), where only the pose of the last platform (the end-effector) is of interest, in the VGW each

platform will have its own independent pose control. This allows for modular control, better suited for a morphing wing application. Each module in the VGW is in effect a parallel mechanism consisting of a series of kinematic chains or so-called limbs. From a kinematic perspective, a variety of different architectures could be used for these limbs. However, from a structural perspective, some of these limb architectures will be better suited for the applications outlined in this thesis. Additionally, due to the restrictive nature of the design space, compactness is highly desired. All of these parameters will be evaluated to determine the optimal configuration for the VGW.

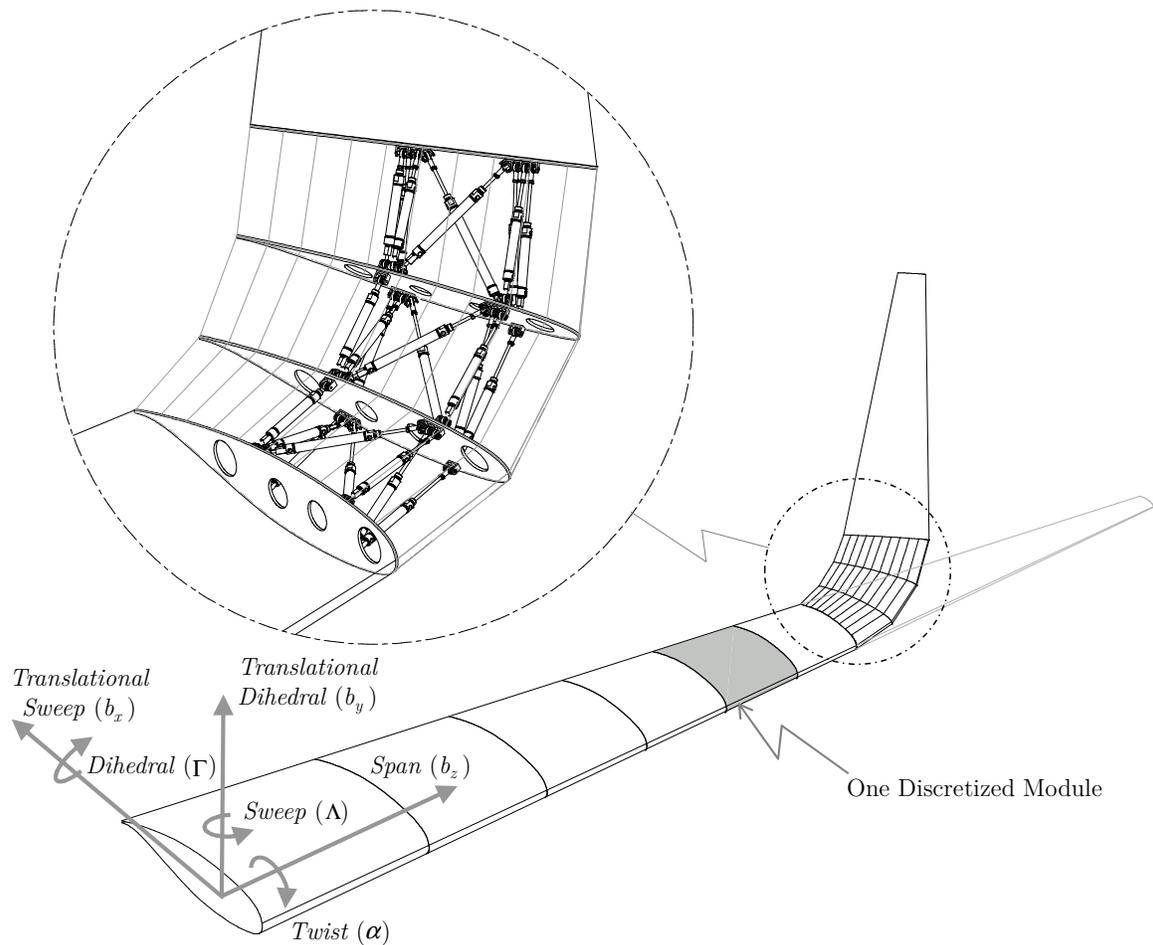


Figure 1.2: Proposed modular morphing wing with six morphing DOFs per module: VGW.

1.4 Thesis Organization

In **Chapter 2**, the background and literature review on morphing wings are presented. Thereafter, each chapter highlights the following:

Chapter 3 covers the design and development of a new family of under-actuated robots with specific applications in wing morphing.

Chapter 4 covers the development of the kinematics and the under-actuated motion control for the VGW.

Chapter 5 covers the development of the kinetostatic relationships for the VGW.

Chapter 6 covers the configuration optimization of the VGW based on the kinematic and static requirements of the system.

Chapter 7 covers the development of the optimal motion planning based on a novel minimal-energy actuation scheme.

The conclusion and the suggested future works are presented in **Chapter 8**.

Chapter 2

Background and Literature Review

Morphing wings at a conceptual level have proven to outperform the conventional fixed-wing design, and have been of special interest to aircraft designers since the first manned flight. One form of classification that is typically used to categorize the type of morphing deals with the variable of interest in the morphing process, such as the sweep angle, the planform area, or the thickness to chord ratio. Based on the type of motion, one can classify morphing parameters into two general categories: 1) wing morphing, and 2) airfoil morphing (Fig. 2.1). Wing morphing includes planform and out-of-plane parameters, i.e. variable sweep, dihedral, twist, and span, whereas airfoil morphing includes variable airfoil thickness, chord length, and camber. The proposed mechanism in this thesis is specifically designed for wing morphing. However, it could be used in combination with airfoil morphing.

By adopting the above classification, if the airfoil shape was to remain unchanged, then a maximum of four parameters could be manipulated by morphing; these are sweep, span, twist, and dihedral, as illustrated by Fig. 2.2.

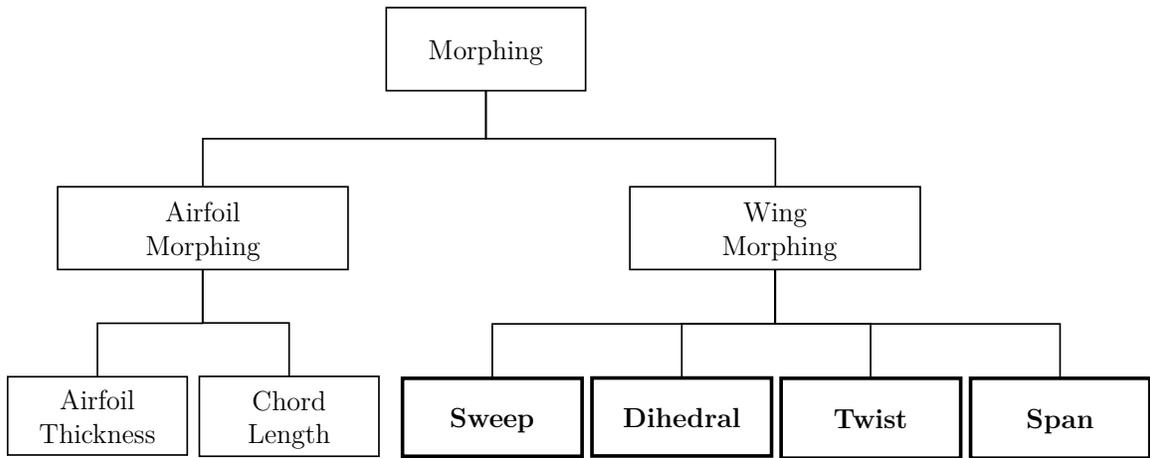


Figure 2.1: Morphing classification based on the parameter of interest.

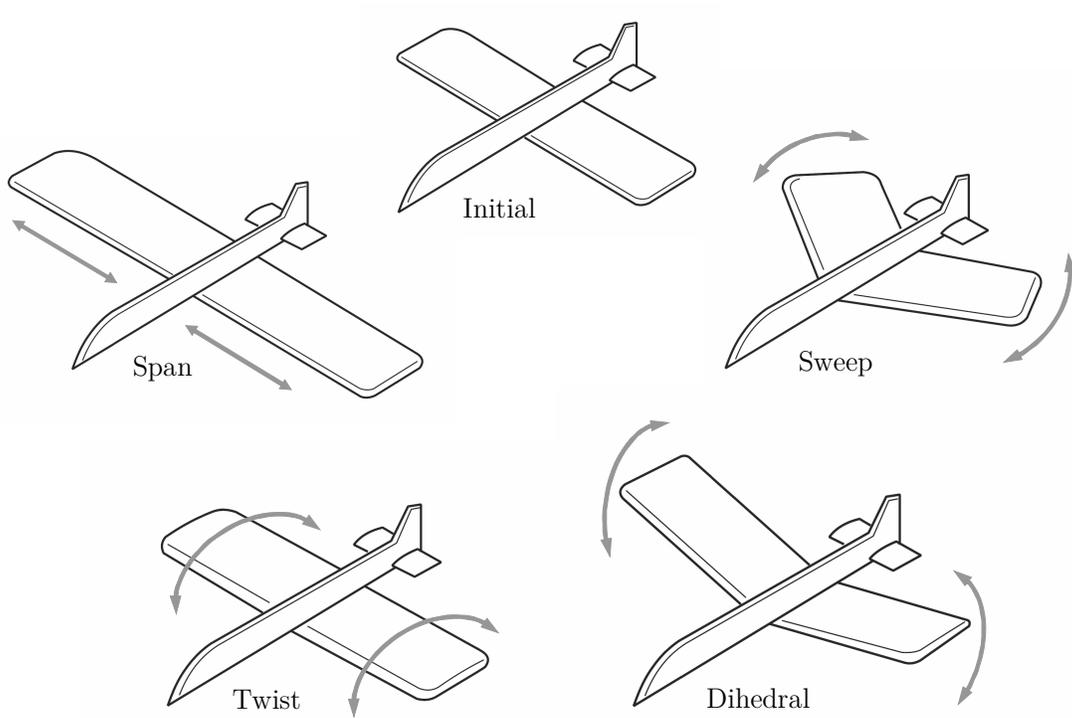


Figure 2.2: Four main DOFs associated with wing morphing.

Although the focus of this thesis is in the area of planform and out-of-plane morphing (see Fig. 2.1), the following review will also encompass a brief summary of various research efforts in the areas of airfoil morphing and morphing materials in the recent years.

The most notable government agencies that have been involved in developing and funding universities in the area of research for morphing wings include the National Aeronautics and Space Administration (NASA), the United States Air Force Research Laboratory (AFRL), and the Defense Advanced Research Project Agency (DARPA). Researchers at such agencies in collaboration with various universities have developed methodologies for the design and optimization of morphing wings that could serve as useful guidelines for future research efforts. Most of these efforts have been concerned with the high level layout design, Multi-disciplinary Design Optimization (MDO), and applicability of such vehicles. However, from a practical perspective, wing morphing research is still at its infancy in the areas of structural design, actuation, and control. This is where it is believed that most valuable contributions could stem from. Hence, of special interest to the current research are those works that have produced working prototypes, rather than the ones focusing only on the theoretical benefits of wing morphing.

Most contributions that have led to the construction and testing of prototypes have been focused on one of these four motions: sweep, span, twist, and dihedral. Very few have attempted to combine more than one of the above motions to achieve the morphing functionality, that is, polymorphing. One of those few is the wing morphing research group at Bristol University [3,4]. In these works, the possibility of having combined dihedral, twist, taper, and span with a variety of different actuation options is systematically analyzed for morphing winglets of a commercial

jet aircraft. Although their work can be regarded as a useful guideline for conceptual design and MDO of morphing wings/winglets, no details have been specified on the actuation process or construction of a prototype that is capable of simulating all four motions. Amongst other examples of polymorphing are [5,6]. A preliminary design and analysis framework for morphing wings that can undergo multi-DOF shape changes is presented in [5]. The design for a morphing Unmanned Aerial Vehicle (UAV) is presented in [6], in which three separate actuation systems are used to control the sweep, span, and twist of the UAV in a non-modular fashion. Some other examples of previous work in the areas of MDO and advanced design include [7-10].

2.1 Morphing DOFs

Although when considered from the aircraft perspective, there are four wing morphing parameters, once discretized into smaller actuating modules, the four parameters can be implemented via all six Cartesian variables. The twist and the span motions each have one distinct DOF associated with them. However, the cant and the sweep motions can comprise of both rotational and translational DOFs. A fully variable mechanism with the ability to independently control all six DOFs has a two-fold benefit. The first arises from having no unwanted coupling of the motions. Typically in kinematically constrained mechanisms, unless the architecture of the mechanism is specifically laid out in a fashion to minimize this coupling, any controllable DOF will have some considerable dependent motion. In the case of a morphing mechanism with only four controllable parameters, two out of the six DOFs will be dependent. However, if the remaining two DOFs can also be controlled then this coupling will vanish. In addition to the removal of dependent motions, being able to control the two additional DOFs can increase the workspace of the mechanism. Therefore, having control over these two additional translational DOFs

can improve the overall performance of the system. Traditionally, this would come at the cost of having more actuators than kinematically required. However, this issue has been addressed in the design of the proposed mechanism, as will be demonstrated in Chapter 3. Figure 1.2 illustrates the four main DOFs: sweep (Λ), span (b_z), twist (α), dihedral (Γ), and the two additional so-called auxiliary DOFs: translational sweep (b_x), and translational dihedral (b_y). It should be noted that the right-hand axis system applied to the left wing is consistently used throughout this thesis. Therefore, negative definitions for the rotational DOFs must not be mistaken with their conventional definitions.

With a few exceptions, most past research efforts in the field of wing morphing have concentrated on one of the four main wing morphing parameters. Furthermore, even fewer have attempted to combine more than two of the above motions to achieve the morphing functionality. The focus of this thesis is the development of a high-stiffness and fault-tolerant design for the internal actuating mechanism of a fully variable morphing wing. After the introduction of the mechanism design, the optimal actuation scheme and configuration design are presented. The optimization formulations are based on the kinematic and static relationships of the proposed system, which will be developed and presented in the upcoming chapters.

2.1.1 Variable Sweep

Among those who have successfully demonstrated morphing for sweep are [6,11-15]. Variable sweep can be highly effective for reducing compressibility induced drags at higher Mach numbers. Some examples of aircraft that have deployed this technique, some as early as 1952, include the Bell X-5, F-111, F-14, Tornado, Tu-160, and B1-B Lancer (Fig. 2.3). More recently, NextGen Aeronautics has successfully demonstrated a sweep morphing UAV [16]. This aircraft has been reported to be

capable of a 70% change in wing area, a 40% change in span, and a 200% change in aspect ratio [17].

2.1.2 Variable Span

The variable span has been used to obtain changes in the aspect ratio to optimize the wing for specific flight regimes. Some examples of variable span include [11, 13,18-22]. Due to practical aspects, sweep and span motions are typically combined. References [13,18] are great examples of this coupling. From a structural point of view, telescopic wings have been one of the most popular approaches to achieve wing span increase [6,19].

2.1.3 Variable Twist

The twist is perhaps one of the most common DOFs to morph due to the immense aeroelastic benefits. Being able to obtain variable twist for various flying conditions carries a large incentive with it, from both aerodynamic and structural aspects. One of the most popular examples is the NASA's modified F-18 with Active Aeroelastic Wing (AAW) [23,24], morphed through wing warping for aeroelastic tailoring and roll control. Other successful demonstrations of morphing for twist include [25-34].

2.1.4 Variable Dihedral (Cant)

The dihedral can help to improve lateral stability. In the case of wing tips, the dihedral or cant can help to reduce the strength of the wing tip vortices. The most common approaches to vary the cant angle are via mechanical rotary actuators, for example in [35], and Shape Memory Alloys (SMAs), for example in [36]. Dihedral is of special interest to this thesis, since this is one of the hardest motions to achieve with conventional actuation. The aerodynamic loads that are required to alter the dihedral angle are far larger than the ones required to change the sweep angle. Using

conventional revolute joints to manipulate the cant/dihedral angle can generally be very energy consuming. Other contributions in this area include [37-39].

A special reference should be made to the XB-70, Valkyre (Fig. 2.3), which achieved a dihedral of up to 65 deg [40], with reference to the convention defined in Fig. 1.2, that is, rotating towards the ground. It did so for improved stability, drag reduction, and increased compression lift at transonic and supersonic speeds. Compression lift is generated by shock waves and helps to support parts of the aircraft's weight. Other examples include the gull wing [38], which uses a jointed spar controlled by a linear actuator placed in the fuselage, and the Lockheed Martin folding wing [41]. Figure 2.3 illustrates some of the morphing aircraft discussed in the previous sections.

2.2 Airfoil Morphing

There is no lack of ideas in the field of airfoil morphing, some of which go as far back as 1916 [49]. Some of these concepts have evolved to become what are now referred to as conventional high lift devices, i.e. flaps and slats. The less conventional approaches such as hingeless trailing edge devices and chord-wise bending, also known as variable camber, are still attracting research attention. This area of morphing has attracted a significant amount of research both from actuation and material perspectives [50-54]. The Mission Adaptive Wing (MAW) is a working example of chord-wise bending/variable camber. It is used on the F-111 (Fig. 2.3) and has demonstrated significant drag reductions and the ability to change the lift distribution to decrease root bending moments [55]. Another demonstration is the Smart Wing wind tunnel model actuated using SMA torque tubes and wires to achieve a change in the twist and trailing edge shape, respectively [56]. This can be considered an example of combining wing and airfoil morphing.



Bell X-5 [42]



F-111A MAW [43]



F-18 AAW [44]



Tornado [45]



F-14 [46]



XB-70 [47]



Tu-160 [48]



B1-B Lancer [48]

Figure 2.3: Examples of high profile morphing aircraft [42-48].

2.3 Morphing Skins

One major challenge in the design of morphing wings lies in the creation of skins for the morphing modules. Although a morphing skin may not be contributing to the stiffness for the wing-box, it will still have to withstand the local pressure loads induced by the flow around the wing. Hence, any suitable material for a morphing

skin needs to be both flexible and be able to handle the loads at the same time. SMAs have proven to be a good candidate for the morphing skins. They have been proposed and demonstrated by [57]. However, they have their limitations and complications, some of which will be discussed in Sec. 2.5.1. Reich et al. [58] proposed a methodology for the design of a suitable composite skin using topology optimization, in order to address the conflicting requirements of low in-plane and high out-of-plane stiffness characteristics. Murray et al. [59] investigated the usage of pre-tension in Flexible Matrix Composites (FMCs) to obtain the desired response for morphing, that is, low in-plane and high out-of-plane stiffness.

Amongst new materials with potential for wing morphing applications is Metal Rubber [60]. Although appearing as a promising candidate for morphing skins, the research in this area is not extensive. Other materials that have attracted research interest in this field include the Veriflex[®]-based polymer [61], which is thermally activated, and the nastic shape changing materials [62]. The nastic materials refer to a class of substances that mimic biological systems to produce large deformations through the conversion of chemical energy available within the material. It should be noted that most of these materials are in their early stages of research and require more development to be utilized at an applied level. The development of a suitable material capable of supporting multiple DOFs is still a major ongoing area of research and is beyond the scope of this thesis. For a thorough literature review for morphing skins the reader is encouraged to see [63,64].

2.4 Morphing for Control vs. Improved Performance

Wing morphing is typically done to achieve one, or in some cases both, of the following: 1) morphing for control and 2) morphing for improved performance. The second category could be further broken down to other subclasses such as range

increase by means of drag reduction, flutter improvement, and vibration reduction, or expanding the flight envelope, for example, by tailoring for subsonic/supersonic flight regimes. There have been some major recent advancements and proposals in the morphing for control area. However, most of the efforts in the wing morphing field fall into the performance improvement category, perhaps due to the larger range of applicability. The presented research in this thesis also falls into the latter category. Nonetheless, some brief references are included for works in the field of morphing for control here.

With the advancement of Fly-By-Wire (FBW) technology in the recent years, the concept of implementing morphing technology in aircraft control has been of special interest. Some examples of morphing for control are the NASA's modified F-18 for improved roll control [65], NASA's Hyper Elliptic Cambered Span (HECS), which uses a biologically inspired approach for yaw, pitch, and roll control, and to reduce induced drag by means of wing tip morphing [66], as well as the variable cant angle winglets used for turning maneuver [35]. Researchers at University of Manchester [67] developed an adaptive vertical tail, which takes advantage of the aeroelastic effects in a desirable manner through alterations in the stiffness of the tail. This is achieved by rotating the spars. The spars could be varied between the two maximum positions: vertical (similar to a conventional spar) to obtain maximum bending stiffness, and horizontal to obtain minimum bending stiffness. Consequently, the torsional stiffness of the tail could then be varied by mechanical means.

2.5 Actuation Concepts

Depending on the DOFs required for morphing, several different actuation concepts have been proposed and tested. The common feature between most of these systems is the use of linear actuators, specially for the planform alteration purposes. However,

most of these systems are only capable of achieving one or two DOFs, for example in [11,19]. The authors in [19] used a combination of pneumatic actuators and linear bearings to achieve a change in span for a morphing UAV application, while the authors in [11] used a worm drive to extend the ribs. Linear actuators are attractive for morphing applications because of their ease of setup and application. Most linear actuators are designed to handle axial loads fairly well. However, they are generally not ideal for supporting out of plane loads. Since in most proposed designs, the linear actuators are packaged to actuate within the x - y plane (Fig. 2.4), additional structure has to be added in such cases to realize the necessary out of plane stiffness (along the z axis in Fig. 2.4). This can lead to significant weight penalties.

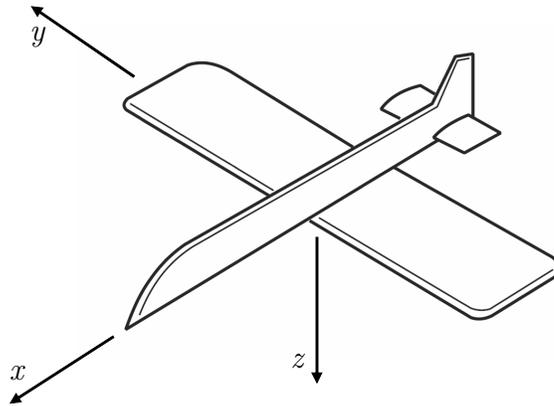


Figure 2.4: Aircraft coordinates.

The proposed design, which will be introduced and studied in the following chapters, employs linear actuators in the form of parallel actuation. This is a fairly appealing concept because it is not only capable of simulating the four DOFs (sweep, dihedral, twist, span), it also behaves as the primary structure. In a conventional aircraft, the combination of spars and skin-stringers typically provides the necessary

torsional stiffness (stiffness about the y axis in Fig. 2.4), the spars provide the shear stiffness (stiffness along the z axis in Fig. 2.4), and the combination of spars and the skin-stringers provide the necessary bending stiffness (stiffness about the x axis in Fig. 2.4). Figure 2.5. illustrates how the major three loads that are experienced by the wing are reacted in a conventional wing-box

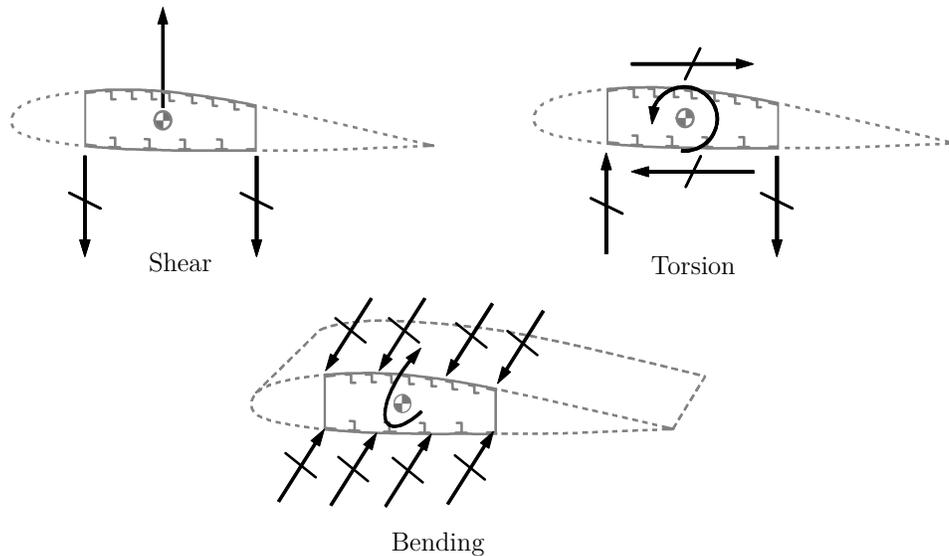


Figure 2.5: Conventional wing-box setup, capable of reacting shear, torsion, and bending using a combination of spars, skins, and stringers.

Replacing the spars, skins, and stringers with a series of linear members will essentially turn the wing-box into a parallel robot, which will be able to provide the appropriate stiffness in all the aforementioned DOFs. The realization and optimal design of such a concept will be discussed in detail in the upcoming chapters.

2.5.1 VGTMs for Wing Morphing

Parallel Kinematic Manipulators (PKMs), such as those belonging to the Gough-Stewart family of parallel manipulators [68,69] used in a modular fashion are referred to as Variable Geometry Truss Manipulators (VGTMs). VGTMs have been utilized

for various space robotics [70-72], and recently wing morphing applications [73-78]. In [73] the authors proposed a discretization method for determining the optimum number of VGTMs for a morphing wing based on maximizing L/D . In [74,75] a novel under-actuated design concept using VGTMs was proposed for a morphing wing along with an optimal configuration based on a cuboid design space. In [76] the authors used a cable-strut system to demonstrate actuation on the HECS wing, where the cables were used to actuate the wing into different positions. In [77] the authors developed a novel topology optimization formulation to obtain optimal structural arrangements for variable geometry truss structures. The optimization process determined the optimum placement for actuators, and attachments that connect the morphing structure to a non-morphing structure, for example a wing to a fuselage. In [78] the authors investigated the use of a cable-strut tensegrity system for a biomimetic application.

It is worth noting that the use of variable geometry trusses in adaptive aerospace structures has not been limited to lift producing surfaces. For example, the authors in [79] designed a morphing structure to change the inlet cross section of a supersonic wind tunnel. The proposed nozzle in [79] is capable of continuously changing the flow velocity for a range of Mach numbers.

Amongst all the proposed VGTMs used for wing morphing, with the exception of a few, for example [74], the modular motion control has not been extensively studied. It is believed that the majority of the types of VGTMs suggested for wing morphing will be generally difficult to control in a precise manner for high-load applications such as those experienced by a wing of a commercial or a business aircraft. Therefore, contributions in the areas of design for high stiffness and the modular motion control of morphing mechanisms are deemed valuable.

2.5.2 Embedded Actuation

Wing morphing via active actuation can be classified into two categories: 1) those using mechanisms in addition to the existing structural elements of the wing-box, and 2) those converting the structural elements into morphing mechanisms. In general the latter will result in a less conventional system; however, it will offer minimal weight impact, since no additional components would be required for morphing. This thesis will investigate the design, analysis, and optimization of a morphing concept that belongs to the second class.

Amongst those who have proposed morphing concepts embedded within the wing structural elements are [18,19,24,29,67,80,81]. The authors in [24] replaced the conventional wing spars with the variable stiffness spar to improve the aircraft roll performance. In [29,67], the authors applied the concept of actively moving and rotating spars to vary the aeroelastic shape of the wing by altering its stiffness. Kota et al. [80] explored various ideas to vary the torsional stiffness of the spars. Similarly, other concepts have been proposed that replace the existing structure within the wing-box using, for example, scissor mechanisms [18], telescopic spars [19], or multi-stable materials [81]. A thorough literature review on morphing wings, using active systems, is presented in [63]. Most of these embedded concepts, proposed for performance improvement or load alleviation, achieve the intended goal by varying only a single DOF, or a coupled set of DOFs.

VGTMs, due to their highly variable nature can easily accommodate morphing for more than just one DOF. This makes them very attractive for multi-DOF morphing applications. However, conventional VGTMs can suffer from poor static characteristics and low stiffness in highly constrained spaces, as in a wing. To improve this trait, a design approach for a wing morphing mechanism is presented

here, in which the static and kinematic characteristics are separated by reconfiguring the system to act both as a statically indeterminate structure and a kinematically determinate mechanism as required. All of this is implemented without redundant actuation, using a novel under-actuated motion control, offering more controllable DOFs than the number of actuators. In the proposed concept, the mechanism is foreseen to be statically redundant during most of its mission profile. This includes instances where the aircraft is performing a high- g maneuver or experiencing a high velocity gust. However, when it is required to morph, during the pre-specified morphing periods, for example, during cruise or before a maneuver, it acts as a conventional mechanism. In the morphing configuration, the system will be kinematically and statically determinate, that is, *isostatic*, which will allow for conventional actuation schemes. However, in the non-morphing configurations, the system will become statically and kinematically indeterminate, that is, *hyperstatic*. This reconfiguration is achieved using a series of lockable passive members that will be locked and unlocked at various times and sequences. By utilizing this implementation the members can be sized to provide the stiffness and the actuation forces required for the morphing configurations while offering the stiffness and the holding forces required for the non-morphing configurations.

Chapter 3

Morphing Mechanism Design

Static redundancy in a parallel manipulator can enhance the stiffness of the end-effector, improve its fault tolerance, minimize its singularity loci, and reduce the internal loads experienced by the joints. Traditionally, this form of redundancy would be accompanied by actuation redundancy. Introduced in this chapter is a new approach to statically enhance a manipulator without actuation redundancy. This is achieved through the use of lockable passive joints that are utilized in an alternating fashion to reconfigure the system into various isostatic and hyperstatic topologies without any external assistance. Although applicable to both kinematically non-redundant and constrained manipulators, this approach is specially effective for those with lower instantaneous mobility. The inherent redundancy in these reconfigurable robots is exploited to obtain full finite mobility with as few as one actuator through under-actuation with the use of virtual alternating constraints, which will be discussed in Chapter 4.

Although the application at hand is one that belongs to \mathbb{R}^3 , the general theory presented in this chapter has been developed to be applicable to cases in both \mathbb{R}^2 and \mathbb{R}^3 .

3.1 Reconfigurability

Reconfigurable parallel robots have attracted considerable interest both in the areas of machine design and motion control over the past two decades. Traditionally, the objective for reconfigurability has been the alteration of mobility [82,83], or motion characteristics [84,85]. However, with the exception of a few [85,86], reconfigurability for enhanced stiffness or static redundancy has not been extensively studied.

Reconfigurability in PKMs can be classified into geometric, topological, or a combination of the two [87]. Geometric morphing deals with the variation in size or orientation of the branches that make up the manipulator, without altering its kinematic architecture. For examples of geometric morphing see [85,86]. Topological morphing deals with variations in the kinematic architecture of the manipulator by changing the types or the sequence of the joints that make up the system. For examples of topological morphing see [82,83,88].

Analogous to the geometric and topological classifications of reconfigurability, the variation in the static and stiffness characteristics of a PKM can also be classified into similar groups (see Fig. 3.1). In this context, the geometric approach will take advantage of the change in the size and orientation of the branches to alter the stiffness of the end-effector or the internal loads experienced by the members, whereas the topological approach will utilize alterations in the connectivity of the branches and joint types to do the same. Generally, only the topological approach has the potential to generate a statically redundant system, which makes it more suitable for autonomous and aerospace applications.

Traditionally, in lower mobility manipulators, the topology is designed to accommodate the motion requirements. The static characteristics are then addressed by altering the geometric parameters such as joint locations or the sizing of the

actuators. By separating the kinematic and the static requirements, one can design manipulators with lower mobility and more appealing static traits. Such lower mobility systems can exhibit stiffness characteristics comparable to that of a PKM with full mobility and superior to conventional lower mobility manipulators with permanently constraining legs.

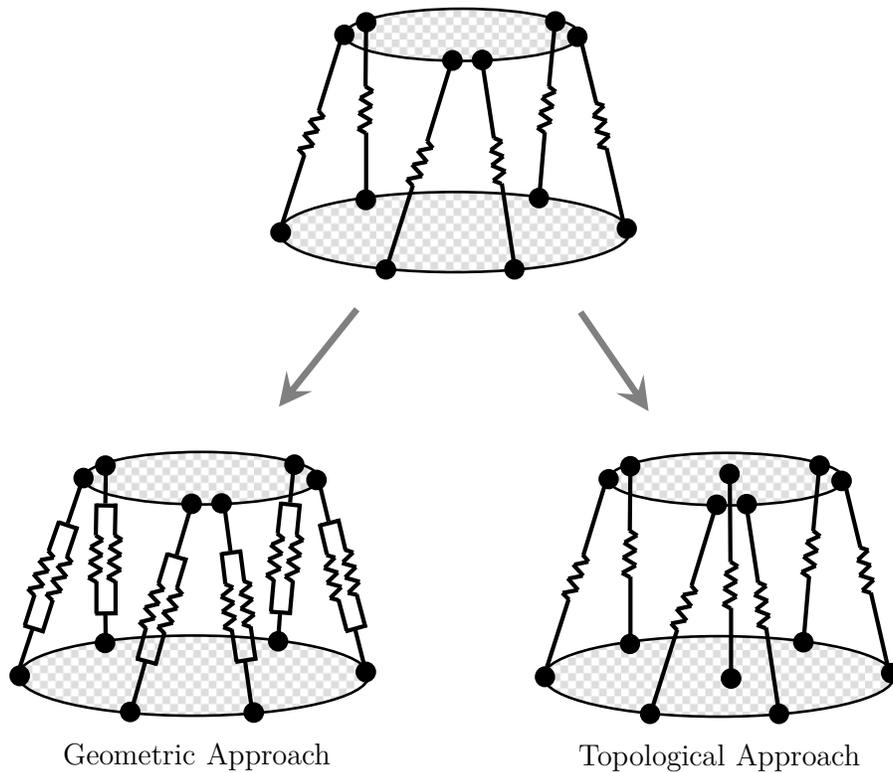


Figure 3.1: Geometric and topological approaches to change the static and stiffness characteristics of a PKM.

In addition to applications where varying the topology can enhance the static characteristics of a PKM in motion, there exist applications where a manipulator may be required to act as a structure for a significant period of time during which it will be experiencing external loads far greater than those required for actuation. One example would be a PKM belonging to an autonomously reconfigurable structure.

Although the manipulator may be in a stationary pose, any shift in the position of its payload, which may happen to be another PKM, could induce large internal loads on the manipulator. Another example is a PKM which would be experiencing large dynamic loads as it would be autonomously transported to reconfigure a larger structure. In any of these cases having a manipulator that can exhibit enhanced static characteristics is desirable. Traditionally, this enhancement comes at the cost of having more actuators than kinematically required.

In this chapter a family of reconfigurable parallel robots with enhanced static characteristics with only as many actuators as the required instantaneous mobility is introduced. These robots belong to the topologically reconfigurable category. Although the focus will be on robots with lower instantaneous mobility, the presented design and analysis methodologies are also applicable to enhance those with full mobility such as the Gough-Stewart parallel manipulators [68,69]. The proposed reconfigurable robots are fault tolerant and can provide enhanced static characteristics with minimal number of actuators. In addition, they offer a larger freedom for their structural design through the separation of static and kinematic requirements. Design of these reconfigurable robots will be presented and discussed in this chapter, while the kinematics/motion implementation, and the kinetostatic analysis will be covered in detail in the upcoming chapters. Before presenting the design, in order to classify these robots, one needs to first review various types of redundancies applicable to PKMs.

3.2 Manipulator Redundancy

A Variable Topology Manipulator (VTM) can effectively reconfigure its topology to accommodate particular situations that may arise from either a static requirement, i.e. increased external loads, or a motion requirement, i.e. enlarged workspace. In

either case, such a manipulator will have to be inherently redundant. Pierrot [89] classifies redundancies in parallel mechanisms into three types: *a)* actuation redundancy, *b)* kinematic redundancy, and *c)* sensor redundancy. A fourth type, *d)* static redundancy [90] is introduced here.

Static redundancy occurs when a mechanism or a structure becomes hyperstatic. A robot or a configuration with a hyperstatic topology is one that is statically and kinematically indeterminate, or in other words it is redundantly rigid, whereas an isostatic topology is one that is both statically and kinematically determinate, or in other words, it is minimally rigid. *Kinematic redundancy* occurs when the mobility of the manipulator is greater than the number of Cartesian motion parameters. This type of redundancy is typically used to enlarge the workspace of a manipulator; for examples see [91,92]. *Actuation redundancy* occurs when the number of actuators is greater than the mobility of the manipulator. This type of redundancy can increase the stiffness or minimize the singularity loci and internal loads of a parallel manipulator; for examples see [93-95]. *Sensor Redundancy* occurs when the number of sensors is greater than the number of actuated joints. This type of redundancy is typically used for solving forward kinematic problems. See [96] for an example of this type of redundancy.

Figure 3.2 illustrates four examples belonging to each type of redundancy in order to distinguish these types in terms of the number of actuators, N_j^A , and mobility, \mathcal{M} . Mobility of a system represents its number of independently controlled DOFs, and can be determined using Eq. (3.1) [97]:

$$\mathcal{M} = \Upsilon(N_l - N_j - 1) + \sum_{i=1}^{N_j} \mathcal{F}_i \quad (3.1)$$

where the motion parameter, Υ , takes a value of three in \mathbb{R}^2 , and six in \mathbb{R}^3 ; N_l , N_j , \mathcal{F}_i , and \mathcal{M} are the number of links, the number of joints, the degrees of relative motion permitted by joint i , and the DOF of the manipulator, respectively.

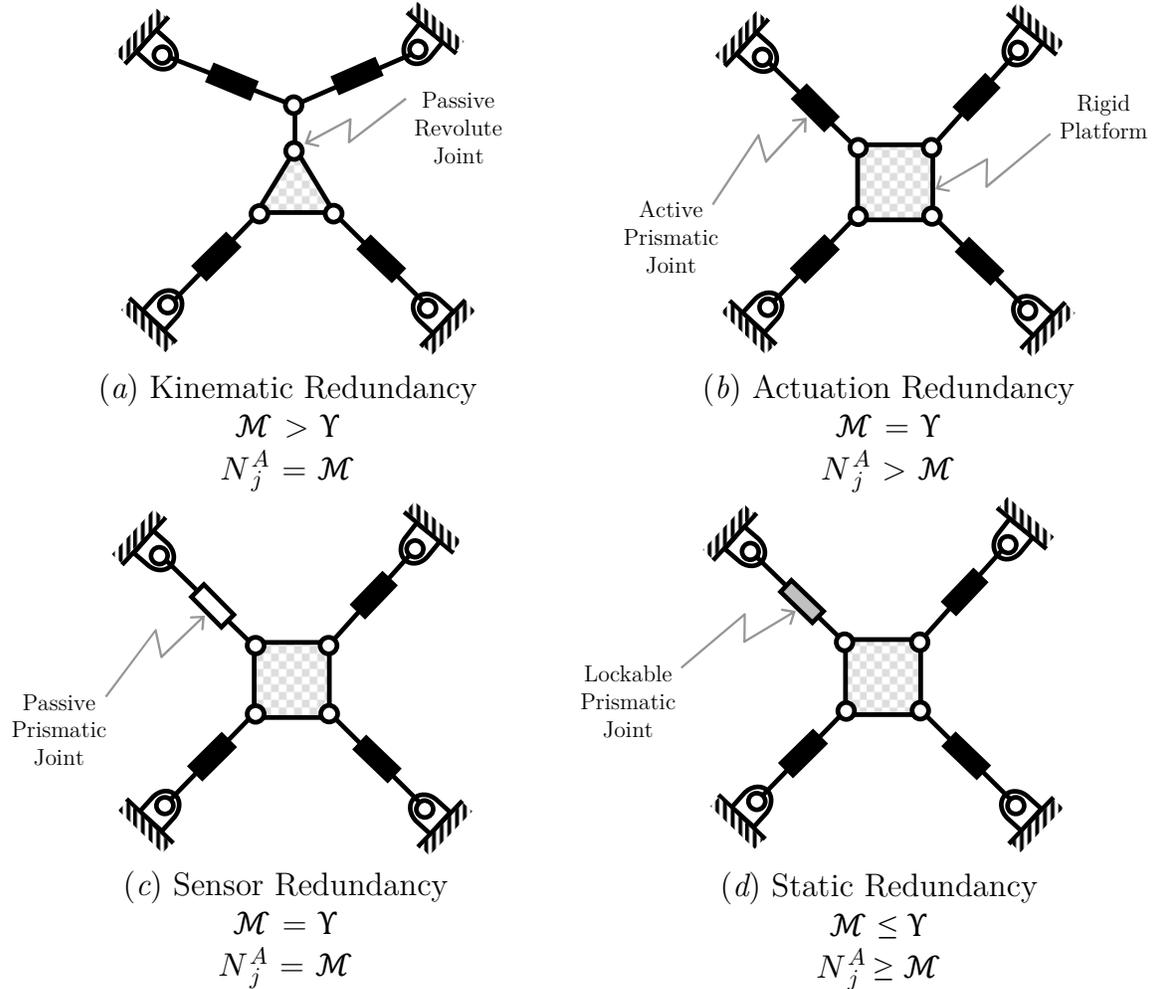


Figure 3.2: Four types of redundancies associated with parallel manipulators.

The kinematically redundant manipulator (Fig. 3.2 a) has 13 joints, and 11 links; it has a mobility of $\mathcal{M} = 4$. The redundantly actuated manipulator (Fig. 3.2 b) has 12 joints, and 10 links; it has a mobility of $\mathcal{M} = 3$. The sensor redundant manipulator

(Fig. 3.2 *c*) has 12 joints, and 10 links; it has a mobility of $\mathcal{M} = 3$. In the case of the statically redundant manipulator (Fig. 3.2 *d*) with the lockable passive limb in the unlocked position, there are 12 joints and 10 links; this gives a mobility of $\mathcal{M} = 3$. However, when the passive limb is locked, the number of joints and links become 11 and 9 respectively; this yields a mobility of $\mathcal{M} = 2$. The manipulator depicted by *d* is statically and kinematically determinate when the passive limb is unlocked. In this case it would be similar to *c* in terms of determinacy. However, when the passive limb becomes locked the system becomes both statically and kinematically indeterminate. In this case, it would be similar to *b* in terms of determinacy and could be considered statically redundant.

The kinematically redundant manipulator (*a*) does not offer static redundancy, that is, if one of the actuators were converted into a passive member the manipulator would become degenerate. The sensor redundant manipulator (*c*) is isostatic at all times; hence, it also offers no static redundancy. Therefore, the only two types that can deliver static redundancy are types *b* and *d*. To further distinguish between the applications of static redundancy, two terms are introduced referred to as the stagnant stiffness and the transitional stiffness. The *stagnant stiffness* refers to the stiffness of a kinematic module when it is not in motion, whereas the *transitional stiffness* simply refers to the stiffness of a kinematic module while it is in motion. Although, the stagnant and transitional stiffnesses are both static terms, the distinction between the two arises from a topological perspective. When transitional stiffness is achieved, the system is isostatic; however, when stagnant stiffness is achieved, it may remain isostatic or become hyperstatic. The distinction between the determinacy of the system has significant impacts on the actuation scheme of the manipulator.

A redundantly actuated manipulator can either have joints that are passive, but can turn into actuators upon command or have redundant actuators that may or may not be part of the kinematic system but can be included when required. Other than the obvious inefficiency of having more actuators than kinematically required, the control implementation of such manipulators entails various complexities. In the case where redundant actuators are active for internal load reduction, due to the indeterminate nature of the system, additional measures from the control point of view must be taken to ensure that actuators are not working against each other or that unnecessary loads are not imposed on the system due to over-constraints.

All published research in the area of parallel manipulator redundancy have been focused on kinematic, actuation, and sensor redundancies. However, static redundancy without redundant actuation has not been studied before. The manipulators introduced in this thesis belong to this type and can generally improve their stagnant stiffness over a conventional non-redundant manipulator in most of their workspace. Additionally, they can generally improve their transitional stiffness for certain poses and areas in their workspace and potentially reduce their singularity loci. Generating static redundancy without actuation redundancy can improve the static characteristics of the system while avoiding the inherent complexities and inefficiencies of actuation redundancy for particular applications such as the ones introduced earlier. In addition, through under-actuation a high degree of actuation efficiency can be maintained while attaining static redundancy.

3.3 Robot Design

The idea of using passive joints that could be locked and unlocked at particular times or sequences has been previously introduced and explored [82,84,88,98,99]. Up to this point the intention for using lockable passive joints have been for actuation efficiency

[88,99] or functional/workspace improvements [82,84,98]. In this thesis lockable passive joints are used to enhance the stiffness and static characteristics of the manipulator. Due to their sequential locking feature they are referred to as Passive Alternating Lockable Members (PALMs). Also, for brevity, any robot that falls in this variable topology family of manipulators will be referred to as a Parallel Robot with Enhanced Stiffness (PRES). Figure 3.3 illustrates a planar PRES with two actuated limbs and two passive lockable limbs.

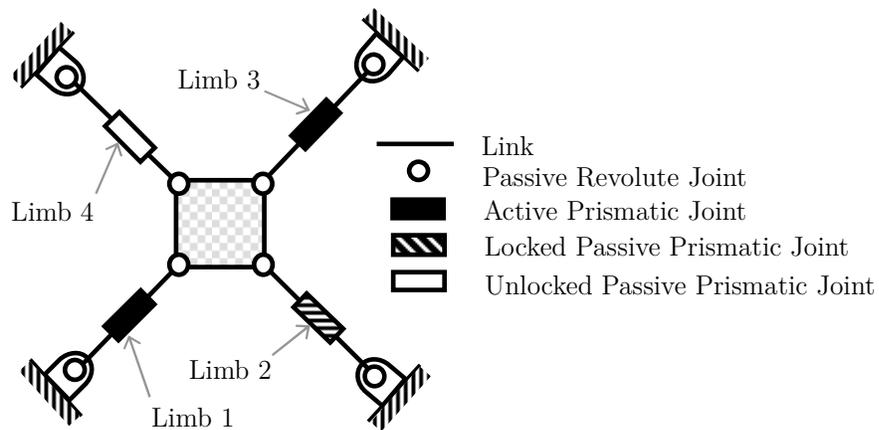


Figure 3.3: A Planar PRES with two actuators and two lockable passive limbs.

In Fig. 3.3 the passive limbs (limbs 2 and 4) can be locked and unlocked to vary the topology of the manipulator. In total, one can have three distinct topologies for this system, two of which are isostatic and one is hyperstatic. The isostatic configurations are used for actuation and enhanced transitional stiffness, whereas the hyperstatic topology is used for enhanced stagnant stiffness. In effect when a passive limb becomes locked it turns into a link connecting the moving platform to the ground/base, and when unlocked it has no effect on the system. For the mechanism illustrated by Fig. 3.3 this transformation results in the three topologies shown in Fig. 3.4.

The actuation process for this planar PRES will be demonstrated through an illustrative example (Fig. 3.5). The goal is to go from some initial pose, ${}_0\mathbf{X}$, to some final pose, ${}_2\mathbf{X}$, with only two actuators. The left subscript notation refers to the stage associated with a pose, i.e. initial (0), intermediate (1), and final (2) in this case.

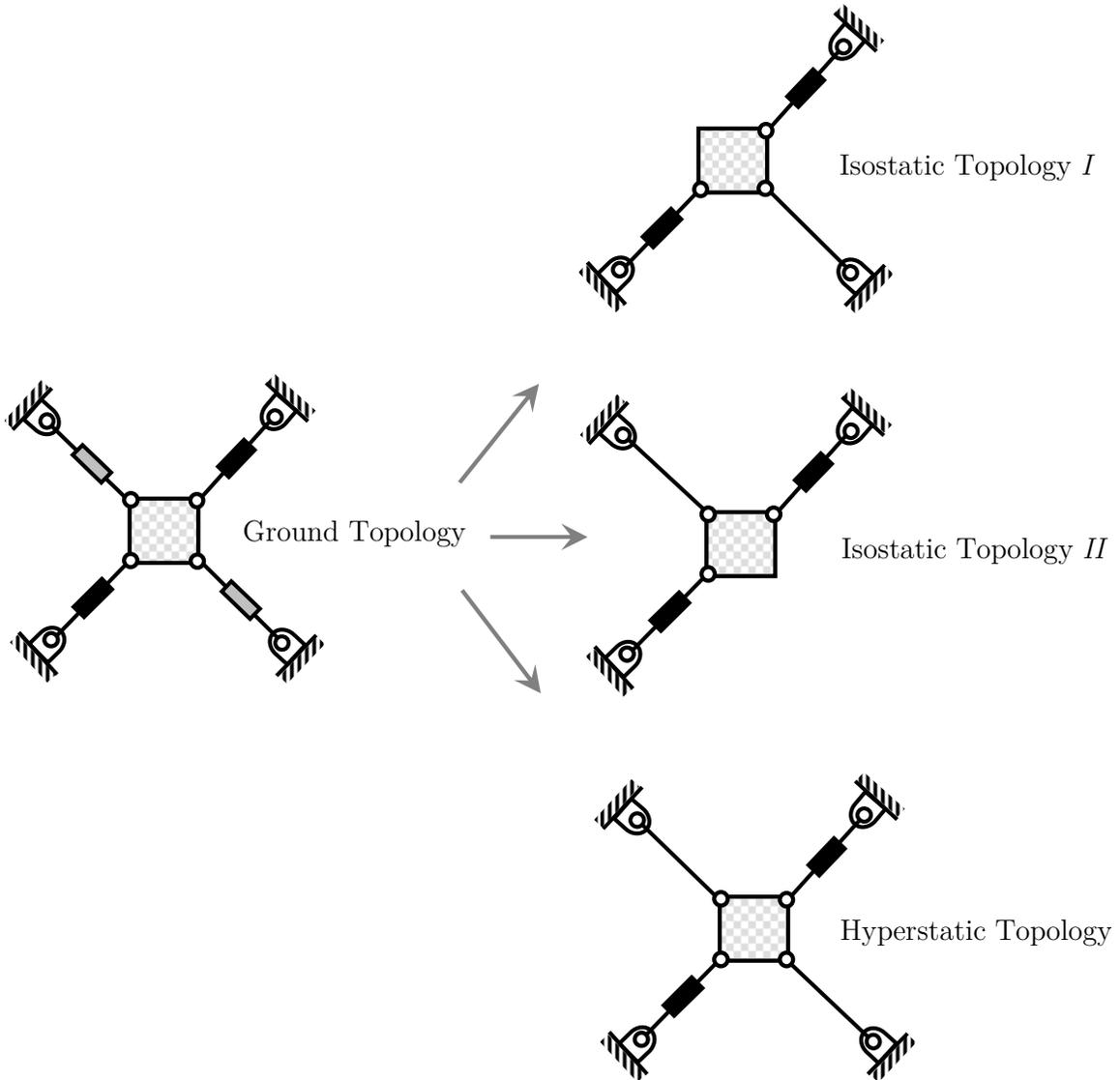


Figure 3.4: Three topologies effectively realized using the ground topology presented in Fig. 3.3.

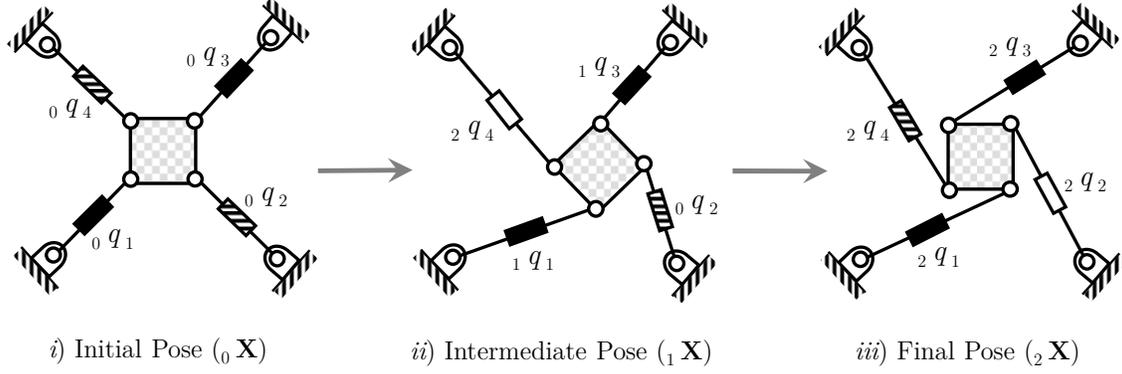


Figure 3.5: Actuation procedure for a planar PRES with two actuators and two lockable passive limbs.

With the module in the final pose, ${}_2 \mathbf{X}$, one will have a vector ${}_2 \mathbf{Q} = [{}_2 q_1 \quad {}_2 q_2 \quad {}_2 q_3 \quad {}_2 q_4]^T$ that describes the lengths of the actuated, unlocked, and locked passive limbs for the final pose. At this point the components associated with the actuated limbs can be ignored since they can be directly controlled. To attain ${}_2 \mathbf{X}$ one must initially realize the components associated with the passive limbs, i.e. ${}_2 q_2$ and ${}_2 q_4$. The notation ${}_2 q_4$ refers to the length of the fourth limb in the second (final) pose. Without loss of generality, first ${}_2 q_4$ will be obtained, followed by ${}_2 q_2$. In order to achieve this an intermediate pose, ${}_1 \mathbf{X}$, will be utilized. The intermediate pose, which is an indirect approach to obtain ${}_2 q_4$ will need to satisfy an actual constraint which is the length of the second limb that remains locked from the initial pose, i.e. ${}_0 q_2$, and a virtual constraint which is the desired final length of the fourth limb, i.e. ${}_2 q_4$. Once the desired final length for the fourth limb is obtained, it will become and remain locked, while the second limb will become unlocked and the same procedure will be repeated using the other isostatic topology to obtain the

desired final length for the second limb, i.e. ${}_2q_2$. Then, by obtaining the lengths associated with the actuated limbs, ${}_2\mathbf{Q}$ can be attained to give the final pose, ${}_2\mathbf{X}$.

The optimal sequence of actuation and the choice for the independent Cartesian variable for the intermediate pose can be determined through an optimization problem which may further enhance the transitional stiffness of the system or minimize the internal loads during actuation.

Similarly, a planar manipulator with the same topology as the one in Fig. 3.3 with only one actuated and three passive lockable limbs (see Fig. 3.6 *a*) can achieve the same in three stages. The additional stage is required to realize the final desired length for the additional passive member. In the case with only one actuator, one will have two actual constraints which correspond to two locked passive members to attain isostaticity and one virtual constraint corresponding to the final length of the unlocked limb. In this case, because the number of constraints is equal to the number of Cartesian variables, there will generally exist only one reachable pose within the workspace of the manipulator. Therefore, the only optimization problem that could be set up is for the choice of sequence in obtaining the desired final lengths for the passive limbs.

A similar actuation procedure can be applied to a spatial PRES. The generalization and the specific case applicable to the VGW will be presented in Chapters 4 and 7, respectively. Figure 3.6 illustrates a few examples of planar and spatial PRES's. For the spatial cases, the joints depicted using a full circle inside a half circle represent spherical joints. The solid lines with no passive/active members depicted in Fig. 3.6 *e* represent rigid links.

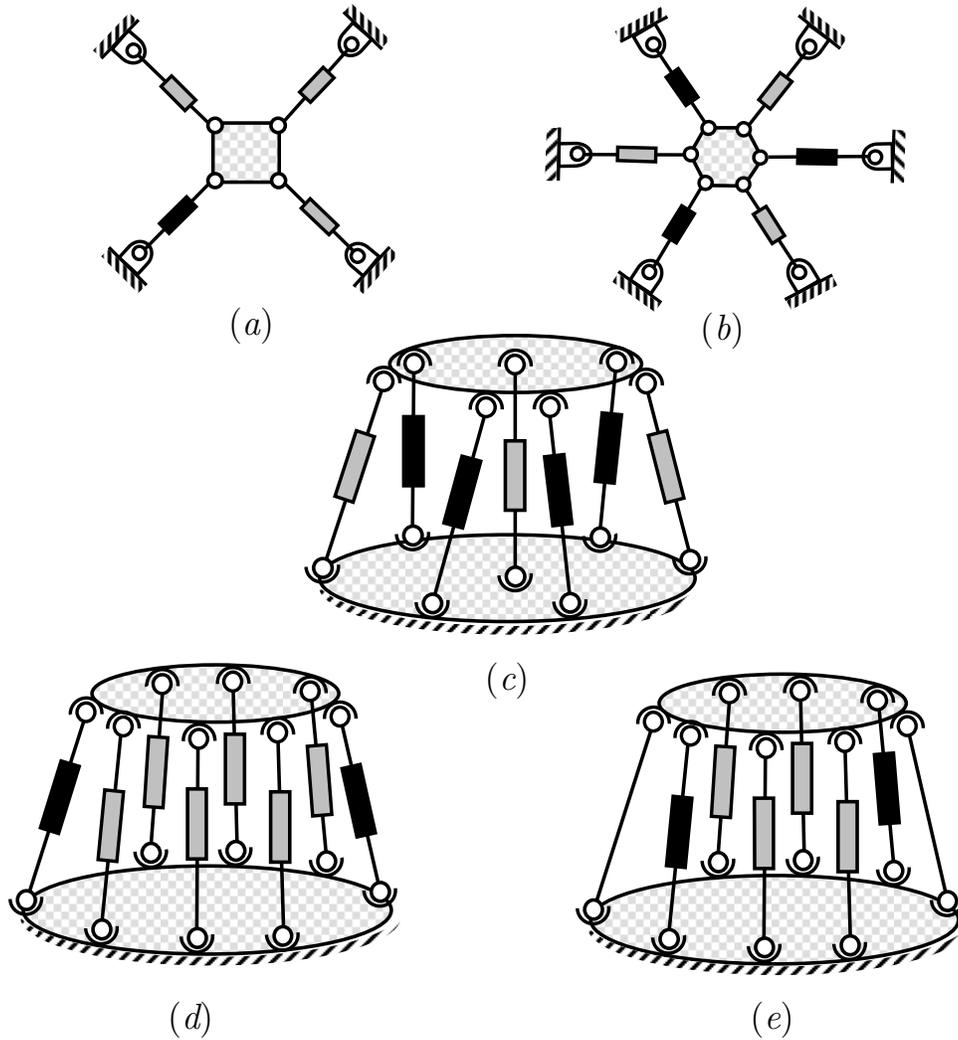


Figure 3.6: Examples of planar and spatial PRES's.

As it can be seen all these manipulators are either kinematically non-redundant or constrained. The degree of static redundancy can vary depending on the application. However, it is always equal to or greater than one. The degree of static redundancy in the hyperstatic state is equal to the number of locked DOFs associated with passive joints that would have to be unlocked to turn the system into an isostatic one. The degree of static redundancy in the hyperstatic state can be expressed as

$$\xi = (N_j^A + N_j^{PL}) - \Upsilon \quad (3.2)$$

where N_j^{PL} and N_j^A are the total number of lockable passive and actuated joints, respectively, and ξ represents the degree of static redundancy in the hyperstatic configuration, where $\xi \in \mathbb{N}_{>0}$. Equation (3.2) is expressed with the underlying assumption that the actuation or locking action is applied to one non-redundant DOF of the joint, regardless of the number of DOFs of the joint itself.

Although PRES's with $N_j^A < \Upsilon$ actuators are systems with lower mobility and are primarily intended for applications that require lower mobility but high stiffness, their inherent redundancy can be exploited to obtain full mobility through under-actuation, as will be discussed in Chapter 4.

3.4 Class Synthesis

Based on the mobility of the system, one can generalize PRES's into three classes as tabulated in Table 3.1. The first class is kinematically non-redundant, i.e. $\mathcal{M} = \Upsilon$, whereas the second and the third classes are kinematically constrained, i.e. $\mathcal{M} < \Upsilon$. All three classes could offer enhanced stiffness for particular applications over a conventional PKM, however their degrees of finite and instantaneous mobilities will vary.

There are two types of DOFs with reference to the end-effector's pose for any mechanical system: Instantaneous or velocity DOF and finite or configuration DOF [100]. *Finite* DOFs are the minimum number of geometric parameters required to uniquely identify the pose of a rigid body in space, i.e. three for planar and six for spatial cases. *Instantaneous* DOFs are those of a rigid body whose velocities and accelerations could be controlled. For fully-actuated PKMs the independent finite

and the instantaneous DOFs are the same. However, if the system is under-actuated, one could have more independent finite DOFs than instantaneous ones. The finite and instantaneous mobilities of the manipulator could then be defined by the achievable independent DOFs in either case.

From a motion control point of view, any PRES that falls under class **I** is fully-actuated with full finite and instantaneous mobilities; a PRES belonging to class **II** is under-actuated with full finite and lower instantaneous mobilities; and a PRES belonging to class **III** is partially-actuated with lower finite and instantaneous mobilities.

Table 3.1: Classification of PRES's and their corresponding static and kinematic characteristics.

Class	Kinematic Characteristic	Static Characteristic	Stagnant Stiffness	Transitional Stiffness	Finite Mobility	Instantaneous Mobility (\mathcal{M})
I	<i>Non-Redundant</i>	<i>Redundant</i>	<i>Enhanced</i>	<i>Not Enhanced</i>	$= \Upsilon$	$= \Upsilon^*$
II	<i>Constrained</i>	<i>Redundant</i>	<i>Enhanced</i>	<i>Enhanced</i>	$= \Upsilon$	$< \Upsilon$
III	<i>Constrained</i>	<i>Redundant</i>	<i>Enhanced</i>	<i>Enhanced</i>	$< \Upsilon$	$< \Upsilon$

* Corresponding to the isostatic state

In all three cases, the redundant rigidity in the system is obtained through the use of PALMs. The manipulator shown in Fig. 3.2 *d* is an example of class **I**; the manipulator depicted in Fig. 3.3 is an example of class **II**; and the manipulator illustrated by Fig. 3.6 *e* is an example of class **III**.

Manipulators that fall under class **I** could be used to only improve the stagnant stiffness. However, class **II** manipulators could enhance both the stagnant and the transitional stiffnesses while maintaining a higher level of actuation efficiency through under-actuation. Similarly Class **III** manipulators can enhance both the stagnant and the transitional stiffnesses; however, they will not be able to attain full finite mobility. Although the third class is generally classified as not under-actuated, in general the under-actuation scheme could be used to achieve a lower finite mobility, still higher than the instantaneous mobility.

Although all three classes vary from the mobility and actuation aspects, the same kinematic constraint and kinetostatic formulations can be applied to all three. The second class can be considered the most general of the three, due to the added complexity introduced by under-actuation. Therefore, the fully-actuated class (class **I**) and the partially-actuated class (class **III**) can be treated as special cases of the under-actuated one (class **II**). Consequently, the focus of the remainder of the chapter will be on the design and analysis of the second class.

In what follows, three areas are addressed: 1) generalizing the enumeration rule for determining the minimum number of required passive members (as presented in this chapter), 2) generating a systematic approach for the kinematic analysis and the implementation of the under-actuation through the systematic implementation of kinematic constraints (see Chapter 4), and 3) efficiently evaluating the stiffness and the internal loads for all configurations and topologies (see Chapter 5).

3.5 Architecture Design

The architecture criteria define a series of design equations that could be used to generate planar and spatial PRES's capable of under-actuation. By inspecting Eq.

(3.1), it becomes clear that the only viable limb architectures for the PRES are those with Υ non-redundant DOFs. Some examples are $\mathcal{R}\mathcal{P}\mathcal{R}^1$ and $\mathcal{R}\mathcal{R}\mathcal{R}$ for the planar case, and $\mathcal{S}\mathcal{P}\mathcal{S}$, $\mathcal{S}\mathcal{R}\mathcal{S}$, $\mathcal{U}\mathcal{P}\mathcal{S}$, $\mathcal{U}\mathcal{C}\mathcal{S}$, and $\mathcal{S}\mathcal{C}\mathcal{S}$ for the spatial case. When unlocked passive limbs with such architectures are added to an already isostatic manipulator they will have no effect on the mobility potential of the system, i.e. three for a planar manipulator and six for a spatial one. Without loss of generality, for the purposes of demonstration and presentation of the proposed design and analysis, mostly $\mathcal{R}\mathcal{P}\mathcal{R}$ and $\mathcal{S}\mathcal{P}\mathcal{S}$ limb architectures have been employed throughout this chapter, for planar and spatial cases, respectively. The optimality of various limb architectures for the VGW will be studied in Chapter 6.

In the planar case, three $\mathcal{R}\mathcal{P}\mathcal{R}$ limbs would be required to maintain isostaticity. Similarly, in the spatial case, six $\mathcal{S}\mathcal{P}\mathcal{S}$ limbs would be required to do the same. This is generally true for systems with injective joints, i.e. those with limbs that are not sharing any joints. Even then, isostaticity is not guaranteed. An example of a singular or a degenerate configuration for a parallel manipulator that possesses six non-injective $\mathcal{S}\mathcal{P}\mathcal{S}$ limbs is the *6-6* Gough-Stewart platform with similarly oriented base and platform [101]. Therefore, due to the lack of a combinatorial condition that would hold for all architectures and configurations, one must rely on computational methods for detecting such singularities. This is a much larger concern in the spatial case, as one may be able to generate hyperstatic topologies that offer a significant increase in the stagnant stiffness of the robot, but in actuation, i.e. for isostatic topologies, the system may suffer from an abundance of singularities. With that said, it is assumed that meeting the necessary conditions for isostaticity is sufficient for generating the architecture criteria for the PRES, as the singularity issue is indeed

¹ See Table 6.1 for the definitions of the symbols and the corresponding kinematic joints

a topology/geometry design problem and not one of mobility or enumeration. Therefore, it can be assumed that the addition of any load-bearing limbs to an isostatic topology will generally create a hyperstatic configuration.

For a typical spatial PKM with six actuators, the six Cartesian variables, that is, three translations and three rotations, that represent the pose of the platform, \mathbf{X} , are usually independent of each other. By locking j prismatic joints, j independent Cartesian variables are effectively converted into dependent ones. For an isostatic topology, if one stores all the independent and dependent Cartesian variables into two vectors \mathbf{X}_I and \mathbf{X}_D , their dimensions will be equal to the number of actuated joints, and the number of locked passive joints, respectively. Therefore, to maintain isostaticity the following will have to be satisfied for any actuating topology:

$$\dim(\mathbf{X}_I) + \dim(\mathbf{X}_D) = \Upsilon. \quad (3.3)$$

If one lets the total number of lockable passive joints be $N_j^{PL} = \dim(\mathbf{X}_D)$, then one will only have one possible isostatic topology. In this case, if $\dim(\mathbf{X}_D) = r \neq 0$, one will have r coupled Cartesian variables that cannot be independently controlled in a finite sense. To gain full finite mobility through under-actuation, one needs more than one topology, specifically, a minimum of N_{iso} distinct isostatic. To realize this, a minimum of $(\dim(\mathbf{X}_D) + 1)$ passive lockable joints would be required. Hence, one can express N_{iso} as the following binomial coefficient:

$$N_{iso} = \binom{\dim(\mathbf{X}_D) + 1}{\dim(\mathbf{X}_D)}. \quad (3.4)$$

If a manipulator can provide a minimum of N_{iso} distinct isostatic topologies, then it can be under-actuated. For example, in the case of the planar manipulator in Fig.

3.3, $\dim(\mathbf{X}_D) = 1$; this gives $N_{iso} = \binom{2}{1} = 2$. Since one can generate two distinct isostatic topologies with the given architecture, under-actuation is possible in this case.

Alternatively, one can express this requirement in terms of the number of lockable passive joints, N_j^{PL} , as a function of the number of actuated joints, N_j^A , for any PRES as follows:

$$N_j^{PL} \geq (\Upsilon - N_j^A) + 1 \quad \forall 1 \leq N_j^A \leq \Upsilon. \quad (3.5)$$

Equation (3.5) is expressed with two underlying assumptions. First, it is assumed that $N_j^A = \dim(\mathbf{X}_I)$, since PRES's are either kinematically non-redundant or constrained. Second, it is assumed that the actuation or locking action is applied to one non-redundant DOF of the joint, regardless of the number of DOFs of the joint itself. In cases where more than one DOF belonging to a joint with multiple DOFs can be actuated or locked, Eq. (3.5) needs to be modified to reflect this. Moreover, the choice for N_j^A is driven by the tolerance of the application to utilize finite DOFs vs. instantaneous ones.

To distinguish between the architectures, a three-digit reference is adopted. The first letter will reference the operational dimensions of the robot, i.e. \mathcal{P} for planar manipulators and \mathcal{S} for spatial ones. The second digit will specify the number of actuators and the third digit will specify the number of lockable passive joints. For example the PRES depicted in Fig. 3.3 is $\mathcal{P}2-2$. This implies that it is a planar manipulator with two actuators and two lockable passive joints. The spatial PRES in Fig. 3.6 *c* can be described as $\mathcal{S}4-3$, since it is a spatial manipulator with four actuators and three lockable passive joints. Table 3.2 summarizes the minimal

architecture requirements for the planar and the spatial cases, determined by Eq. (3.5). Terms \mathcal{M} and $N_j^{PL}_{min}$ are the instantaneous mobility of the system and the minimum number of lockable passive joints, respectively.

Table 3.2: Minimal architecture criteria for PRES's in \mathbb{R}^3 and \mathbb{R}^2 .

	<i>Type</i>	\mathcal{M}	N_j^A	$N_j^{PL}_{min}$
\mathbb{R}^3	$\mathcal{S}-6-1$	6	6	1
	$\mathcal{S}-5-2$	5	5	2
	$\mathcal{S}-4-3$	4	4	3
	$\mathcal{S}-3-4$	3	3	4
	$\mathcal{S}-2-5$	2	2	5
	$\mathcal{S}-1-6$	1	1	6
\mathbb{R}^2	$\mathcal{P}-3-1$	3	3	1
	$\mathcal{P}-2-2$	2	2	2
	$\mathcal{P}-1-3$	1	1	3

Naturally, the minimum number of actuated/lockable joints to become hyperstatic is four and seven for the planar and the spatial cases, respectively. However, this is only the minimum requirement. With the minimal number of actuated/lockable joints, there will potentially be one degree of static redundancy if the topology is designed properly. However, more lockable passive joints could be added to increase the degree of redundancy. For instance, the robot shown in Fig. 3.6 *d* has two degrees of static redundancy.

3.6 Modularity in Design

From a kinematic perspective, there is no need to have limbs that are architecturally identical to each other. However, from a practical perspective, this is a valuable trait to have incorporated into the design. As will be shown in this section, useful relationships can be developed if modularity restrictions are imposed on the system. These modularity restrictions are as follows: *a)* all active limbs have the same architecture, *b)* all passive limbs have the same architecture when unlocked, and *c)* all passive limbs have the same architecture when locked. The distinction between active and passive limbs will become clear shortly. Figure 3.7 illustrates some examples of planar PRES's that can be considered modular and non-modular based on the proposed definition of modularity, defined through the above restrictions.

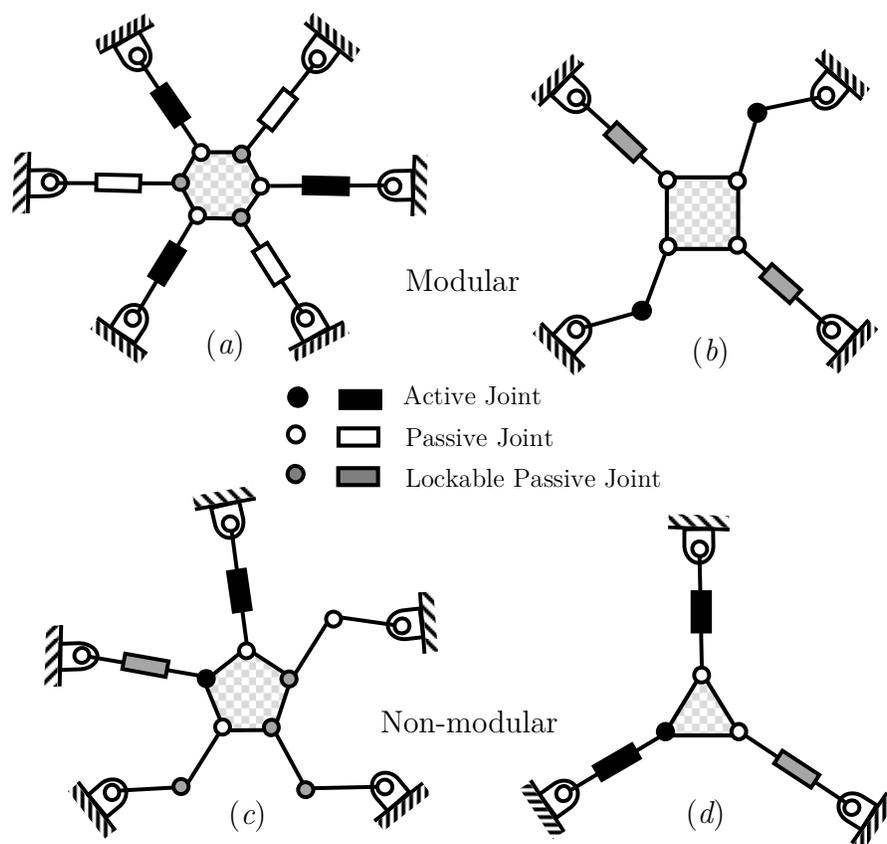


Figure 3.7: Examples of modular and non-modular planar PRES's.

A PRES could be made up of active, passive, or hybrid limbs. An *active* limb is a kinematic branch containing one or more actuated joints. A *passive* limb is a kinematic branch containing only passive joints, at least one of which is lockable. One can also have a hybrid limb, made up of both actuated and lockable passive joints. More on hybrid limbs will be discussed shortly. The total number of passive joints in the system, N_j^P , will be equal to the summation of all regular passive joints, N_j^{PR} , and all lockable passive joints, N_j^{PL} , that is,

$$N_j^P = N_j^{PR} + N_j^{PL}. \quad (3.6)$$

The total number of lockable passive limbs in the system, N_P , will be equal to the summation of all passive limbs in the locked state, N_{P_l} , and all passive limbs in the unlocked state, N_{P_u} , that is,

$$N_P = N_{P_l} + N_{P_u}. \quad (3.7)$$

Passive limbs accommodating Υ non-redundant DOFs without any lockable joints are not considered here as they would not affect the static characteristics of the system. Figure 3.8 illustrates the joint type breakdown of the active and passive limbs that make up a PRES.

As previously noted, in addition to the active and passive limbs, the PRES could also be made up of hybrid limbs. A *hybrid* limb contains both active and lockable passive joints. Any PRES with hybrid limbs, be it modular or non-modular, can be transformed into a PRES with only active and passive limbs. Figure 3.9 illustrates two examples of such transformations. In Fig. 3.9 the PRES's on the left and their

corresponding equivalent versions effectively have the same characteristics, that is, same instantaneous and finite mobilities, as well as the same degrees of static redundancy.

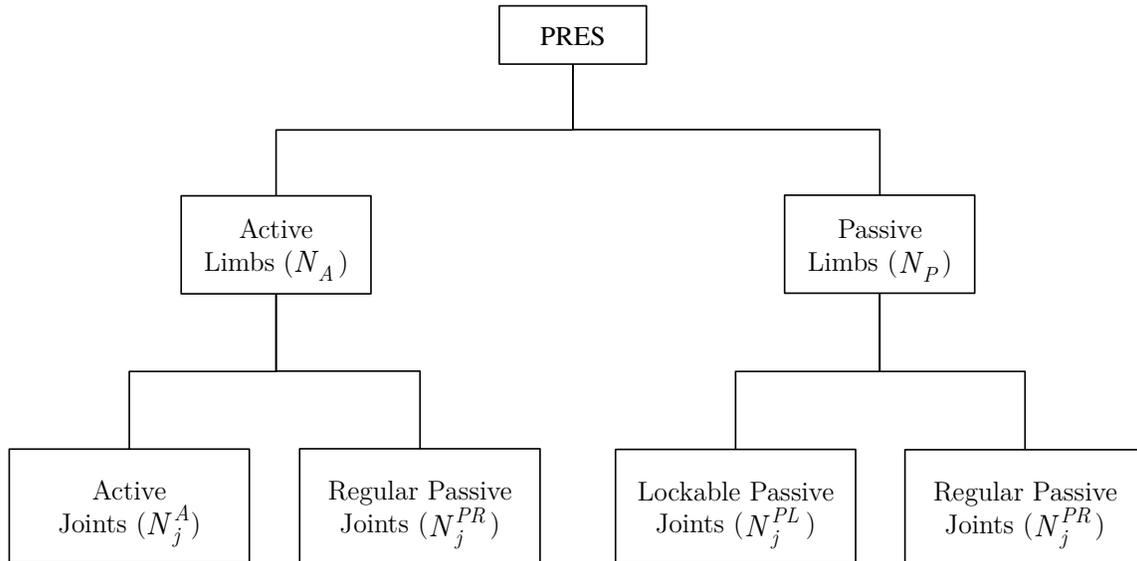


Figure 3.8: Joint type breakdown of the active and passive limbs making up a PRES.

It can be shown that the PRES's depicted in Fig 3.9 *a* and *b* have the same mobility, that is, $\mathcal{M} = 2$ when isostatic, and $\mathcal{M} = 1$ when hyperstatic. Similarly, the PRES's depicted in Fig. 3.9 *c* and *d* have the same mobility, that is $\mathcal{M} = 2$ when isostatic, and $\mathcal{M} = 1$ when hyperstatic. All four PRES's have a static redundancy of one. Due to the possibility of this transformation, only active and passive limbs are used in what follows. Hybrid limbs can be simply realized by superimposing active and passive limbs, in both modular and non-modular PRES's.

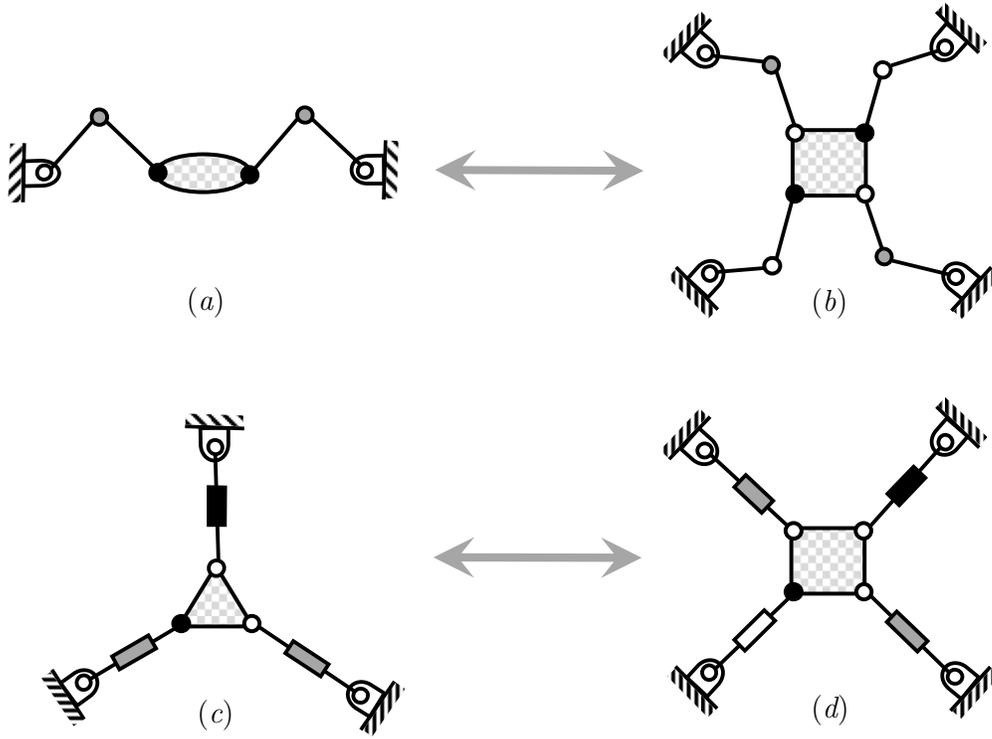


Figure 3.9: Examples of transformation of PRES's with hybrid limbs into functionally equivalent PRES's with only active and passive limbs.

To develop the required relationships for modularity one can initially start by expressing the mobility equation (Eq. (3.1)) in the following form:

$$\mathcal{M} = \Omega + \Phi \quad (3.8)$$

where

$$\Omega = \Upsilon(N_l - N_j - 1) \quad (3.9)$$

$$\Phi = \sum_{i=1}^{N_j} \mathcal{F}_i. \quad (3.10)$$

Based on the defined restrictions the components of Ω and Φ can be further broken down to separate the active, locked passive, and unlocked passive limbs:

$$N_l = \lambda_A N_A + \lambda_{P_l} N_{P_l} + \lambda_{P_u} N_{P_u} + 2 \quad (3.11)$$

$$N_j = (\lambda_A + 1)N_A + (\lambda_{P_l} + 1)N_{P_l} + (\lambda_{P_u} + 1)N_{P_u} \quad (3.12)$$

$$\sum_{i=1}^{N_j} \mathcal{F}_i = C_A N_A + C_{P_l} N_{P_l} + C_{P_u} N_{P_u} \quad (3.13)$$

where λ_A , λ_{P_l} , and λ_{P_u} represent the number of links per active, locked passive, and unlocked passive limbs, respectively; C_A , C_{P_l} , and C_{P_u} represent the branch connectivity of active, locked passive, and unlocked passive limbs, respectively; similarly, N_A , N_{P_l} , and N_{P_u} represent the number of active, locked passive, and unlocked passive limbs, respectively. The 2 in Eq. (3.11) represents to the ground and the moving platform.

For example the robot depicted in Fig. 3.6 *c* has $N_A = 4$, $\lambda_A = 2$, $\lambda_{P_l} = 1$, $\lambda_{P_u} = 2$, $C_A = 6$, $C_{P_l} = 5$, and $C_{P_u} = 6$ for all configurations. When the PRES is isostatic, that is, one of the passive members is unlocked, $N_{P_l} = 2$, and $N_{P_u} = 1$; and when hyperstatic, that is, all passive members are locked, $N_{P_l} = 3$, and $N_{P_u} = 0$. By inserting these values into Eqs. (3.11) through (3.13), subsequently into Eqs. (3.9) and (3.10), and finally into Eq. (3.8) it can be confirmed that $\mathcal{M} = 4$ for the isostatic case, while $\mathcal{M} = 3$ for the hyperstatic case, as expected.

Furthermore, by substituting Eqs. (3.11) and (3.12) into Eq. (3.9), Ω can be expressed as follows:

$$\Omega = \Upsilon(1 - N_A - N_{P_l} - N_{P_u}). \quad (3.14)$$

It can be observed from Eq. (3.14) that Ω is invariant with respect to changes in topology. For any given system, the summation of all the limbs, whether actuated, locked passive, or unlocked passive, are always the same, regardless of topology. Therefore, using Eq. (3.8) one can conclude the following relationship:

$$\mathcal{M}_{iso} - \Phi_{iso} = \mathcal{M}_{hyp} - \Phi_{hyp} \quad (3.15)$$

where \mathcal{M}_{iso} and \mathcal{M}_{hyp} represent the mobility of the system in the isostatic and hyperstatic configurations, respectively. Similarly, Φ_{iso} and Φ_{hyp} represent the same for the total relative DOFs of the system. Furthermore, the mobility of the system in the hyperstatic state can be represented by subtracting the degree of static redundancy, ξ , introduced through the locking action, from the isostatic mobility of the system:

$$\mathcal{M}_{hyp} = \mathcal{M}_{iso} - \xi. \quad (3.16)$$

For example the PRES in Fig. 3.6 *b* has $\xi = 3$ when all the passive members are locked, which yields $\mathcal{M}_{hyp} = 0$. The terms Φ_{iso} and Φ_{hyp} can also be represented as

$$\Phi_{iso} = C_A N_A + C_{P_l} N_{P_l}^{iso} + C_{P_u} N_{P_u}^{iso} \quad (3.17)$$

$$\Phi_{hyp} = C_A N_A + C_{P_l} N_{P_l}^{hyp} \quad (3.18)$$

where the terms with subscripts/superscripts *iso* and *hyp* correspond to the isostatic and hyperstatic states, respectively. It should be highlighted that the terms associated with the unlocked passive members do not appear in Eq. (3.18) since by definition unlocked passive members are non-existent in the hyperstatic configuration. By substituting Eqs. (3.16) through (3.18) into Eq. (3.15), one obtains

$$C_{P_u} N_{P_u}^{iso} - C_{P_l} (N_{P_l}^{hyp} - N_{P_l}^{iso}) = \xi. \quad (3.19)$$

Since by definition,

$$N_{P_u}^{iso} = N_{P_l}^{hyp} - N_{P_l}^{iso} \quad (3.20)$$

Eq. (3.19) can be further simplified to the following form:

$$C_{P_u} - C_{P_l} = \frac{\xi}{N_{P_u}^{iso}}. \quad (3.21)$$

The term $(C_{P_u} - C_{P_l})$ represents the number of non-redundant DOFs that must be removed from each unlocked passive limb for $N_{P_u}^{iso}$ limbs to attain ξ degrees of static redundancy in the system. After imposing the following physical constraints all possible solutions for a given degree of static redundancy can be obtained to accommodate modularity:

$$\left\{ \begin{array}{l} \mathbf{A}: (C_{P_u} - C_{P_l}) \in \mathbb{N}_{>0} \\ \mathbf{B}: N_{P_u}^{iso} \in \mathbb{N}_{>0} \\ \mathbf{C}: C_{P_u} > C_{P_l} \\ \mathbf{D}: \xi \in \mathbb{N}_{>0} \end{array} \right. . \quad (3.22)$$

For example for a system with $\xi = 1$, there is only one solution that satisfies both Eqs. (3.21) and (3.22), which is $(C_{P_u} - C_{P_l}) = 1$. Similarly, for a system with $\xi = 2$, two solutions exist: $(C_{P_u} - C_{P_l}) = 1, 2$. Table 3.3 summarizes all possible solutions for $\xi = 1, \dots, 6$ to accommodate modularity. These solutions and the above relationships are valid in \mathbb{R}^3 , as well as \mathbb{R}^2 , with the obvious limitation of $(C_{P_u} - C_{P_l}) \leq 3$ for cases in \mathbb{R}^2 .

Table 3.3: Possible solutions for modularity with various degrees of static redundancy.

ξ	$N_{P_u}^{iso}$	$C_{P_u} - C_{P_l}$
1	1	1
2	1	2
2	2	1
3	1	3
3	3	1
4	1	4
4	2	2
4	4	1
5	1	5
5	5	1
6	1	6
6	2	3
6	3	2
6	6	1

As it can be observed from Table 3.3, the only solution that satisfies all degrees of static redundancy is $(C_{P_u} - C_{P_l}) = 1$. Furthermore, if more modularity restrictions are introduced into the problem by, for example, requiring the passive and the active limbs to be of similar architecture with one actuator per limb, then

the only solution will be $(C_{P_u} - C_{P_l}) = 1$, that is, if the same DOF being actuated on the active limb is to be locked on the passive limb. In such cases, the number of actuated and lockable passive joints will simply be equal to the number of actuated and lockable passive limbs, respectively, that is, $N_j^A = N_A$ and $N_j^{PL} = N_P$. The choice for the optimal architecture for the VGW will be discussed in more details in Chapter 6.

3.7 Conclusion

A new type of parallel robots that can enhance their static and stiffness characteristics by varying their topology while maintaining actuation efficiency have been presented in this chapter. These robots can attain static redundancy without any actuation redundancy. In addition, a class of these VTMs can deliver full finite mobility through under-actuation with as few as one actuator. This particular class is of special interest for the morphing wing mechanism.

The proposed robots can be reconfigured to change their motion characteristics to increase their workspace over conventional lower mobility systems with permanently constraining limbs. The static redundancy in the design of these reconfigurable robots improves their fault tolerance over conventional non-redundant ones, making them desirable for autonomous and aerospace applications. In what follows, first, the general kinematic and kinetostatic analyses of these robots will be presented with the underlying theme of wing morphing application. Afterwards, the optimal design and motion control specific to wing morphing will be studied.

Chapter 4

Kinematics

There are two components involved in the kinematics of the proposed modular morphing wing: a global component and a local one. The global kinematics is concerned with determining the pose of each platform in the open kinematic chain which is made up of multiple morphing modules, i.e. open-loop kinematics. The local kinematics is concerned with establishing the inverse kinematic relationships needed for the actuation of each module, i.e. closed-loop kinematics.

Although from a stiffness point of view, the hyperstatic structure is superior to a conventional parallel robot placed inside of an envelope similar to a wing, the hyperstatic configuration cannot be conventionally controlled since the system is kinematically indeterminate. To resolve this issue, an under-actuated control scheme has been proposed. This will not only help to achieve the desired final motion, it will also offer a very efficient actuation scheme. This motion control scheme, which is based on the use of alternating kinematic constraints, is the distinguishing feature of the proposed reconfigurable mechanism. In this chapter, initially, the general local and global kinematic models of the VGW are presented. Then the proposed under-actuated motion control scheme is presented and discussed.

4.1 Local Kinematics

Once the pose of the i^{th} platform defined by its corresponding frame is established with respect to the $(i-1)^{\text{th}}$ platform, one can apply the inverse kinematic relationship to the i^{th} module to determine the stroke lengths required for actuation. The conventions and coordinates illustrated in Fig. 4.1 are used to present the local kinematic model for the proposed mechanism.

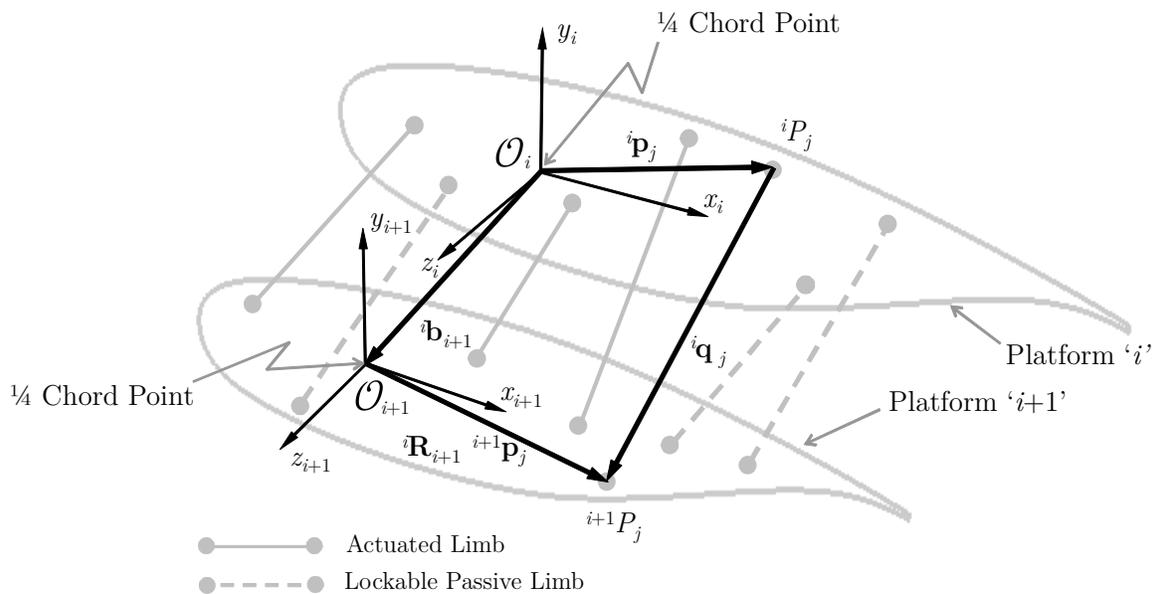


Figure 4.1: Local kinematic model of the VGW.

As illustrated, the coordinate frame i is attached to the i^{th} platform, and the coordinate frame $(i+1)$ is attached to the $(i+1)^{\text{th}}$ platform. Using the defined coordinate frames, the loop closure equations can be set up as follows for each of the limbs:

$${}^i\mathbf{p}_j + {}^i\mathbf{q}_j = {}^i\mathbf{b}_{i+1} + {}^i\mathbf{R}_{i+1} {}^{i+1}\mathbf{p}_j \quad (4.1)$$

where ${}^i\mathbf{b}_{i+1} = [{}^ib_{i+1_x} \quad {}^ib_{i+1_y} \quad {}^ib_{i+1_z}]^T$ is the vector representing the coordinates of the $\frac{1}{4}c$ (chord) point on the $(i+1)^{\text{th}}$ platform in the i^{th} coordinate frame. The components of ${}^i\mathbf{b}_{i+1}$ represent the main span DOF and the two auxiliary translational DOFs. Vector ${}^i\mathbf{p}_j$ represents the position of the joint belonging to limb j on the i^{th} platform in the i^{th} coordinate frame; ${}^i\mathbf{q}_j$ is the limb vector in the i^{th} coordinate frame; and ${}^{i+1}\mathbf{p}_j$ represents the position of the joint belonging to limb j on the $(i+1)^{\text{th}}$ platform in the $(i+1)^{\text{th}}$ coordinate frame. The term ${}^i\mathbf{R}_{i+1}$ represents the rotation matrix, where ${}^i\mathbf{R}_{i+1} \in SO(3)$. Expressed in terms of the variables of interest, that is, dihedral, Γ (rotation about the body-fixed x axis), sweep, Λ (rotation about the body-fixed y axis), and twist, α (rotation about the body-fixed z axis), ${}^i\mathbf{R}_{i+1}$ could be written as follows:

$${}^i\mathbf{R}_{i+1} = \mathbf{R}_x({}^i\Gamma_{i+1})\mathbf{R}_y({}^i\Lambda_{i+1})\mathbf{R}_z({}^i\alpha_{i+1}) \quad (4.2)$$

where

$$\mathbf{R}_x({}^i\Gamma_{i+1}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos({}^i\Gamma_{i+1}) & -\sin({}^i\Gamma_{i+1}) \\ 0 & \sin({}^i\Gamma_{i+1}) & \cos({}^i\Gamma_{i+1}) \end{bmatrix} \quad (4.3)$$

$$\mathbf{R}_y({}^i\Lambda_{i+1}) = \begin{bmatrix} \cos({}^i\Lambda_{i+1}) & 0 & \sin({}^i\Lambda_{i+1}) \\ 0 & 1 & 0 \\ -\sin({}^i\Lambda_{i+1}) & 0 & \cos({}^i\Lambda_{i+1}) \end{bmatrix} \quad (4.4)$$

$$\mathbf{R}_z({}^i\alpha_{i+1}) = \begin{bmatrix} \cos({}^i\alpha_{i+1}) & -\sin({}^i\alpha_{i+1}) & 0 \\ \sin({}^i\alpha_{i+1}) & \cos({}^i\alpha_{i+1}) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (4.5)$$

The term ${}^i\mathbf{R}_{i+1}$ in its current order of sequence (Eq. (4.2)) could alternatively be expressed in its compact form as

$${}^i\mathbf{R}_{i+1} = \begin{bmatrix} \text{CA C}\alpha & -\text{CA S}\alpha & \text{S}\Lambda \\ \text{S}\Gamma \text{S}\Lambda \text{C}\alpha + \text{C}\Gamma \text{S}\alpha & -\text{S}\Gamma \text{S}\Lambda \text{S}\alpha + \text{C}\Gamma \text{C}\alpha & -\text{S}\Gamma \text{C}\Lambda \\ -\text{C}\Gamma \text{S}\Lambda \text{C}\alpha + \text{S}\Gamma \text{S}\alpha & \text{C}\Gamma \text{S}\Lambda \text{S}\alpha + \text{S}\Gamma \text{C}\alpha & \text{C}\Gamma \text{C}\Lambda \end{bmatrix} \quad (4.6)$$

where $\text{C}\alpha$ and $\text{S}\alpha$ imply $\cos(\alpha)$ and $\sin(\alpha)$, respectively. The superscripts and the subscripts have been omitted for clarity.

Generally, the sequence at which the rotation matrices are applied changes the final pose of the platform, since rotation matrices are not commutative. The sequence represented by Eq. (4.2) has been chosen for the demonstration of the current work; however, any valid rotation sequence can be used instead.

The percentage of contribution towards the overall motion of the wing can be determined for each actuating module using any suitable discretization algorithm such as the one proposed by [73]. Subsequently, the local kinematics is used to establish the inverse kinematic relationships associated with each module. Each module is then treated as a kinematic building block, generating a modular serial robot, whose analysis is presented in the following section.

4.2 Global Kinematics

The position and the orientation of every actuating rib/platform are determined with respect to the previous platform using a series of coordinates attached to each platform which is considered to be a rigid body. One can determine the position and the orientation of each module by using these coordinate frames. The point at the $\frac{1}{4}c$ of each platform is used as the origin of the corresponding coordinate frame. As illustrated in Fig. 4.2, the pose of each platform can then be described and

manipulated using a vector, ${}^i\mathbf{b}_{i+1}$, which defines the position of the $(i+1)^{\text{th}}$ coordinate frame, \mathcal{O}_{i+1} , with respect to the i^{th} coordinate frame, \mathcal{O}_i , followed by a rotation matrix, ${}^i\mathbf{R}_{i+1}$, which captures the changes in the orientation of the $(i+1)^{\text{th}}$ coordinate frame with respect to the i^{th} coordinate frame.

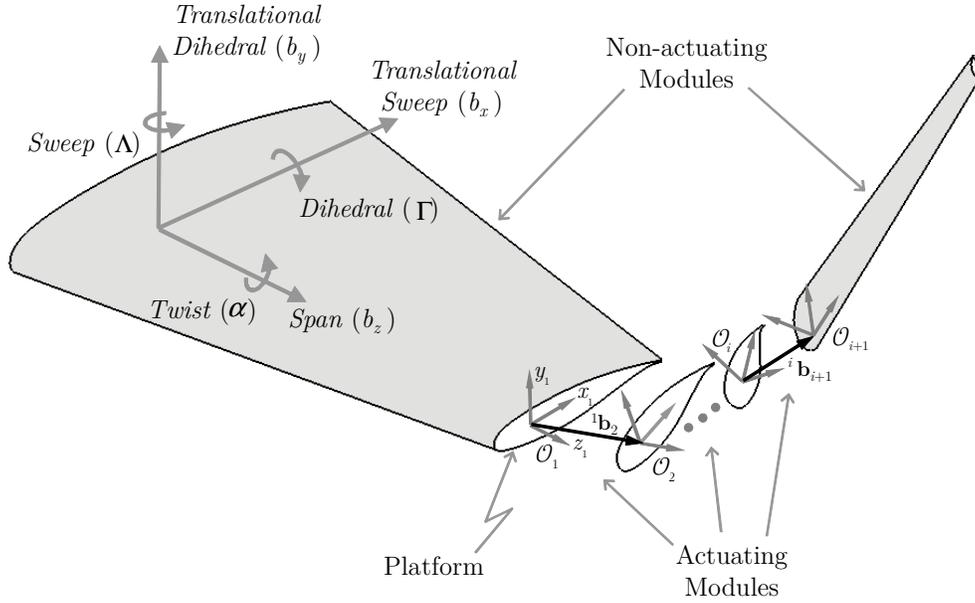


Figure 4.2: Global kinematic model of the VGW.

The position and the orientation of the a^{th} platform with respect to a reference coordinate system, here denoted by 0 (also referred to as global), can be found using a recursive method as follows:

$${}^0\mathbf{b}_a^* = {}^0\mathbf{b}_1 + \sum_{i=1}^{a-1} \left(\prod_{r=0}^{i-1} {}^r\mathbf{R}_{r+1} \right) {}^i\mathbf{b}_{i+1} \quad (4.7)$$

and

$${}^0\mathbf{R}_a = \prod_{i=0}^{a-1} {}^i\mathbf{R}_{i+1} \quad (4.8)$$

where ${}^0\mathbf{b}_a^*$ and ${}^0\mathbf{R}_a$ represent the position and the orientation of the a^{th} platform with respect to the global reference frame, respectively.

4.3 Under-actuation²

There are various approaches to implement the under-actuation, such as by locking the passive joints [88,99] or using guiding racks [102]. Traditionally, the intents for the design of such manipulators have been to implement a control scheme for forward kinematics of serial manipulators with passive joints [99], or simply to minimize the number of actuators in the case of parallel manipulators [88,102]. However, to the author's best knowledge, no under-actuated parallel manipulator has been designed for the purpose of redundant rigidity or enhanced stiffness.

The variable topology approach which enables the under-actuation in the proposed system is made possible through the use of lockable passive joints. These lockable passive joints assist in the under-actuation by imposing kinematic constraints in a systematic fashion, as illustrated in the example shown in Fig. 3.5. There are several possible ways to implement such constraints in order to obtain the desired end results. From a kinematic perspective, the conversion of passive joints into fully rigid or partially rigid joints can be implemented on any type of joint. A *fully rigid* joint refers to a joint that has lost all of its DOFs. A prismatic (\mathcal{P}) or a revolute (\mathcal{R}) joint would become fully rigid if it was to be locked. A *partially rigid* joint refers to a joint that still has some DOFs left after being locked. A spherical (\mathcal{S}), a universal (\mathcal{U}), or a cylindrical (\mathcal{C}) joint could potentially become partially rigid. The change in topology could also occur by locking multiple joints belonging to one limb. From a practical perspective, locking and unlocking a passive \mathcal{P} or \mathcal{R} joint, or

² under-actuation in the present context only deals with holonomically constrained systems

the axial component of a \mathcal{C} joint is simpler to implement. Additionally, having limbs that are similar in architecture helps with maintaining modularity in design. As it will become clear in Chapter 6, lockable prismatic joints with similar \mathcal{SPS} limbs will be employed for the VGW. Although having similar limbs with lockable \mathcal{P} or \mathcal{R} joints are preferred, they are not necessary features for the proposed design. Without loss of generality, lockable prismatic joints are used here with similar \mathcal{RPR} and \mathcal{SPS} limbs for the planar and the spatial cases, respectively.

For a fully-actuated PRES the inverse kinematic relationship could be directly used to control the manipulator. However, an under-actuated or a partially-actuated PRES is a lower mobility system from an instantaneous point of view, in which case one needs to be able to express the dependent Cartesian variables in terms of the independent ones. Essentially, solving the kinematic constraint equations is required in order to solve the inverse kinematic problem.

4.3.1 Kinematic Constraints

It was shown in Chapter 3 that in order to independently control all six Cartesian variables, one will need N_{iso} distinct topologies. In other words, to obtain full finite mobility, one will need $(\dim(\mathbf{X}_D) + 1)$ actuation stages, where each stage corresponds to a change in the pose. For instance, for a planar manipulator with two actuators, a minimum of two lockable passive limbs and two actuation stages will be required to get to the final pose. The transitions from ${}_0\mathbf{X}$ to ${}_1\mathbf{X}$ and ${}_1\mathbf{X}$ to ${}_2\mathbf{X}$ illustrated by Fig. 3.5 represent the two aforementioned stages. For a spatial manipulator with four actuated limbs (see Fig. 3.6 *c*), a minimum of three lockable passive limbs and three actuation stages will be required to independently control six finite DOFs, and so on.

By locking one passive $\mathfrak{S}\mathfrak{P}\mathfrak{S}$ limb the branch connectivity, C_j , has changed from six to five, effectively reducing the independent Cartesian variables of the module from six to five. This is true since for any holonomically constrained kinematic system,

$$\sum_{j=1}^{N_L} C_j = \sum_{i=1}^{N_j} \mathcal{F}_i \quad (4.9)$$

where N_L and N_j represent the total number of limbs and joints in the module, respectively, and \mathcal{F}_i represents the number of relative DOFs permitted by joint i .

Therefore, in the case with one locked passive member, one out of six parameters needs to be expressed in terms of the other five. The number of locked passive members in the system directly corresponds to the number of dependent Cartesian variables, which belong to vector \mathbf{X}_D . In general, for every distinct isostatic topology, one needs to express the dependent parameters in terms of the independent ones and the length(s) of the locked limbs. Symbolically this can be written as

$$\mathbf{X}_D = \mathbf{g}(\mathbf{X}_I, \mathbf{Q}_P) \quad (4.10)$$

where \mathbf{Q}_P is a vector of scalars containing the lengths of the locked passive members and \mathbf{g} is the constraint function. The branch kinematic constraint is essentially the parametric equation of a circle and a sphere for the planar and the spatial cases, respectively, imposed by the fixed length of the locked passive member. Therefore, to develop the parametric constraint equation of the j^{th} locked passive limb, one can rewrite Eq. (4.1) in terms of its Euclidian norms:

$$\| {}^i \mathbf{q}_j \|^2 = \| {}^i \mathbf{b}_{i+1} + {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_j - {}^i \mathbf{p}_j \|^2 \quad (4.11)$$

which could be decomposed into

$$\| {}^i \mathbf{q}_j \|^2 = \| {}^i \mathbf{b}_{i+1} - {}^i \mathbf{p}_j \|^2 + 2 \left[({}^i \mathbf{b}_{i+1} - {}^i \mathbf{p}_j)^T {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_j \right] + \| {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_j \|^2. \quad (4.12)$$

Since ${}^i \mathbf{R}_{i+1} \in SO(3)$, then $\| {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_j \| = \| {}^{i+1} \mathbf{p}_j \|$. Also, $\| {}^i \mathbf{q}_j \| \equiv {}^i q_j$. Therefore, Eq. (4.12) can be further simplified to

$${}^i q_j^2 - 2 \left[({}^i \mathbf{b}_{i+1} - {}^i \mathbf{p}_j)^T {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_j \right] - \| {}^i \mathbf{b}_{i+1} - {}^i \mathbf{p}_j \|^2 - \| {}^{i+1} \mathbf{p}_j \|^2 = 0. \quad (4.13)$$

Equation (4.13) represents the parametric constraint corresponding to the j^{th} passive limb. A kinematically constrained system with r locked passive limbs will have r constraint equations that need to be simultaneously solved in order to obtain the inverse kinematic relationships of the system.

The positions of all the joints that lie on the platform with respect to the moving frame, as well as the length(s) of the locked passive member(s), are known. Therefore, the only variables to solve would be $(b_x, b_y, b_z, \Gamma, \Lambda, \alpha)$. In total there will be $\dim(\mathbf{X}_D)$ constraint equations, which implies $\dim(\mathbf{X}_D)$ unknown variables. The choice of the elements of \mathbf{X}_D is somewhat arbitrary from the finite DOF point of view, since full mobility is attainable with the under-actuated system, given that joint limits are not reached. The optimal sequence could generally be determined using an optimization algorithm, for example, to minimize the actuation energy, i.e. minimize both the internal loads and the stroke lengths, or to maximize the transitional stiffness of the system. The optimal actuation sequences will be

investigated in Chapter 7. Any suitable numerical methods such as Newton-Raphson could be used to obtain the solutions.

While the kinematic constraint of Eq. (4.13) is presented for a lockable prismatic joint, the proposed scheme could be easily modified and applied to architectures with lockable \mathcal{R} joints; for example an $\mathcal{R}\mathcal{R}\mathcal{R}$ limb for a planar PRES, or an $\mathcal{S}\mathcal{R}\mathcal{S}$ limb for a spatial PRES where the middle \mathcal{R} joint is locked to generate the kinematic constraint. However, as it will become evident in Chapter 6, such limb architectures may not be desirable for the application at hand.

Solving the constraint equations is required to determine the dependent components of the pose; however, it is not sufficient to implement the under-actuation. To do so, a systematic approach is proposed, which will be referred to as Under-actuation with Virtual Alternating Constraints (UVAC).

4.3.2 Under-actuation with Virtual Alternating Constraints

For any given pose in the Cartesian space of the manipulator, there would be a unique vector representing the lengths for all the limbs in the reachable workspace of the manipulator. To obtain that particular pose one needs to realize those lengths. The lengths of the actuated limbs can be simply varied through actuation; however, the lengths of the passive limbs must be realized indirectly since they cannot be actuated. One can achieve those lengths by imposing a virtual constraint in addition to the physical constraints needed to generate an isostatic topology. The virtual constraint will treat an unlocked passive member as if it were locked when solving the constraint equations. Therefore, any solution that satisfies the actual constraints, i.e. constraints required to maintain isostaticity, must also satisfy the length constraint imposed by the virtual constraint. The virtual constraint would be the desired length of a particular passive member in the final pose. By achieving the

lengths in a sequential manner, one can obtain the final lengths of all the passive limbs without directly actuating any of them as demonstrated through the example in Chapter 3.

The lengths of the locked passive members in the final pose must satisfy the constraint function \mathbf{g} , which expresses the dependent Cartesian variables in terms of the independent ones for going from the last intermediate pose to the final one. A minimum of $(\dim(\mathbf{X}_D) + 1)$ constraint equations are needed for every intermediate stage. Out of these, $\dim(\mathbf{X}_D)$ constraint equations are real, as they are imposed by the locked passive member(s) to maintain isostaticity and one is virtual as it corresponds to the one passive member that is not locked.

The idea of using active joints to drive lockable passive joints into a particular position in a closed-loop system has been previously introduced and investigated [84]. Similarly, the use of inverse kinematics to determine locking values of lockable passive joints for a parallel manipulator has also been investigated [98]. The robot in [84] is kinematically non-redundant, whereas the robot in [98] operates in two modes, one of which is redundantly actuated, and the other one is kinematically redundant. The robots introduced in the current work that utilize the proposed virtual constraints are kinematically constrained.

What sets the proposed approach apart from the previous works is the utilization of under-actuation through an alternation of locking and unlocking of passive joints to give full finite mobility to an otherwise kinematically constrained robot. This alternating scheme, which could utilize the same passive member more than once in one actuation cycle, i.e. going from an initial pose to a final pose, enables the under-actuation and gives the UVAC approach its novelty compared to the previous works.

The same principle that applies to the under-actuated motion control of the proposed three-dimensional wing morphing mechanism also applies to the two-dimensional case. Therefore, for clarity in demonstration, a two-dimensional case is initially used to present the UVAC approach. Although the general methodology for any similar under-actuated system in \mathbb{R}^2 and \mathbb{R}^3 are alike, as it will shortly become clear the exact implementation of the under-actuation is topology dependent. Therefore, the implementation for the VGW will be presented in Chapter 7 after the presentation of the optimal configuration (Chapter 6).

4.3.2.1 UVAC for a $\mathcal{P}2-2$ PRES

The planar mechanism depicted in Fig 4.3 has two actuators and two passive members. The presented ground topology yields two isostatic and one hyperstatic topologies.

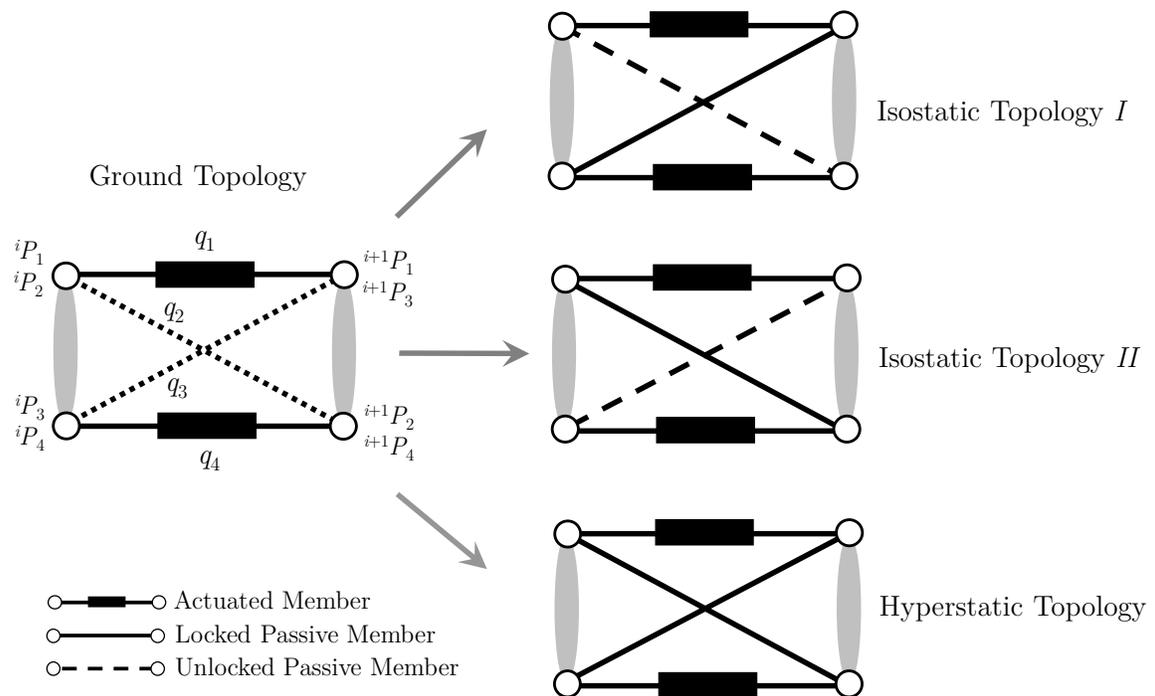


Figure 4.3: Two isostatic and one hyperstatic topologies can be generated for the $\mathcal{P}2-2$ PRES.

In Fig. 4.3 the solid lines with the thick bars represent linear actuators, the plain solid lines represent locked passive limbs, and the dashed lines represent unlocked passive limbs. Each limb consists of two revolute joints and one prismatic joint, i.e. $\mathcal{R}\mathcal{P}\mathcal{R}$, with \mathcal{P} being the actuated/lockable joint. Topologies *I* and *II* represent two distinct isostatic topologies that could be attained using the ground topology. The hyperstatic topology is obtained through the locking of all lockable passive joints. Fig. 4.4 represents the kinematic model of the $\mathcal{P}\mathcal{2}\mathcal{2}$ PRES.

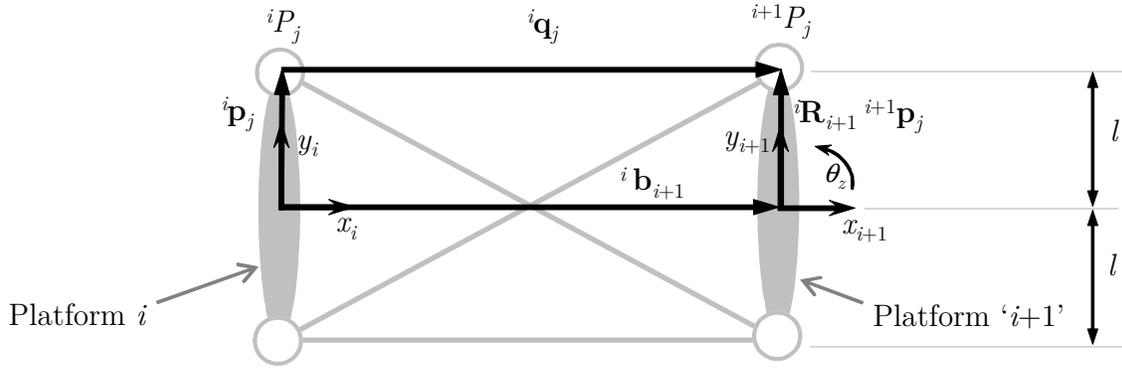


Figure 4.4: Kinematic model of a $\mathcal{P}\mathcal{2}\mathcal{2}$ PRES.

The loop closure equation (Eq. (4.1)) is also applicable to the $\mathcal{P}\mathcal{2}\mathcal{2}$ PRES. In this case, the rotation matrix, ${}^i\mathbf{R}_{i+1} \in SO(2)$, can be represented as

$${}^i\mathbf{R}_{i+1} = \begin{bmatrix} \cos {}^i\theta_{i+1_z} & -\sin {}^i\theta_{i+1_z} \\ \sin {}^i\theta_{i+1_z} & \cos {}^i\theta_{i+1_z} \end{bmatrix}. \quad (4.14)$$

The rest of the terms are similar to those of \mathbb{R}^3 with the z component of the vectors missing. Using the loop closure equations, the inverse kinematic relationships can be set up for each of the limbs as

$${}^i_s q_j = \mathbf{f}({}^i_s \mathbf{X}_{i+1}) \quad \forall j = 1, 2, 3, 4; \quad s = 0, 1, 2 \quad (4.15)$$

where

$${}^i_s \mathbf{X}_{i+1} = \begin{bmatrix} {}^i_s b_{i+1_x} & {}^i_s b_{i+1_y} & {}^i_s b_{i+1_z} \end{bmatrix}^T \quad (4.16)$$

and j and s are the number of limbs and stages, respectively, and \mathbf{f} is the function representing the loop closure equation. The term ${}^i_s q_j$ represents the length of the j^{th} limb of the i^{th} module in the s^{th} actuation stage.

The objective is to go from some initial pose to some final pose in the 2D Cartesian space. Since there are two actuators available, one intermediate pose will be needed to achieve this. Notation ${}^i_0 \mathbf{X}_{i+1}$ will be adopted to represent the initial pose of the $(i+1)^{\text{th}}$ platform in the i^{th} coordinate frame. Subsequently, ${}^i_1 \mathbf{X}_{i+1}$ and ${}^i_2 \mathbf{X}_{i+1}$ will be used to represent the intermediate and the final poses, respectively. With the given topology, there are two possible actuation paths that could be utilized. Each path presents a distinct sequence of how to get from some initial pose to some final pose. Figure 4.5 illustrates the two options for under-actuation. Generally, the higher the number of paths the higher the number of possible solutions, which could lead to more optimal actuation. The details of the optimization for motion control will be discussed in Chapter 7.

Similar to the example presented in Chapter 3, with the module in the final pose, ${}^i_2 \mathbf{X}_{i+1}$, one has a vector ${}^i_2 \mathbf{Q} = [{}^i_2 q_1 \quad {}^i_2 q_2 \quad {}^i_2 q_3 \quad {}^i_2 q_4]^T$ that describes the lengths of the actuated, unlocked, and locked passive limbs of the i^{th} module for the final pose. Since the lengths of the actuated limbs can be directly controlled, one needs not be

concerned with attaining those values through under-actuation; thus, i_2q_2 and i_2q_3 need to be achieved in order to obtain ${}^i_2\mathbf{X}_{i+1}$. With path 1, when going from ${}^i_0\mathbf{X}_{i+1}$ to ${}^i_1\mathbf{X}_{i+1}^{Path\ 1}$ using topology *I*, the attained pose will give i_2q_2 . This pose must satisfy a real constraint, that is i_0q_3 , and a virtual constraint, that is i_2q_2 . The two constraint arcs that must be satisfied can be visually represented by Fig. 4.6.

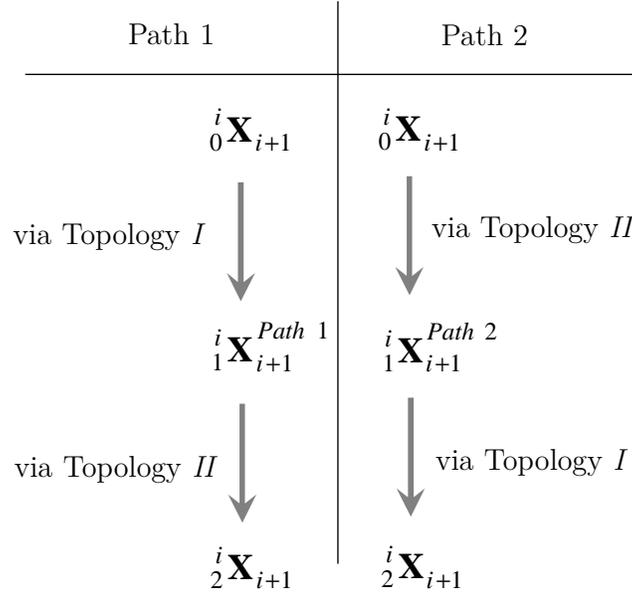


Figure 4.5: Two distinct paths used to achieve under-actuation for the \mathcal{P} -2-2 PRES.

The dashed line implies that the passive member is unlocked, whereas the solid line implies that it is locked. Figure 4.6 shows the state of ${}^i_1\mathbf{Q}^{Path\ 1} = [{}^i_1q_1 \quad {}^i_1q_2 \quad {}^i_1q_3 \quad {}^i_1q_4]^T$, where ${}^i_1q_3 = {}^i_0q_3$, i.e. the actual constraint, and ${}^i_1q_2 = {}^i_2q_2$, i.e. the virtual constraint. With i_2q_2 achieved, and the third limb unlocked, the inverse kinematics problem can be simply solved to attain ${}^i_2\mathbf{X}_{i+1}$. For systems

with more than one intermediate pose, the step described above will be repeated until all the passive members have achieved the desired stroke lengths.

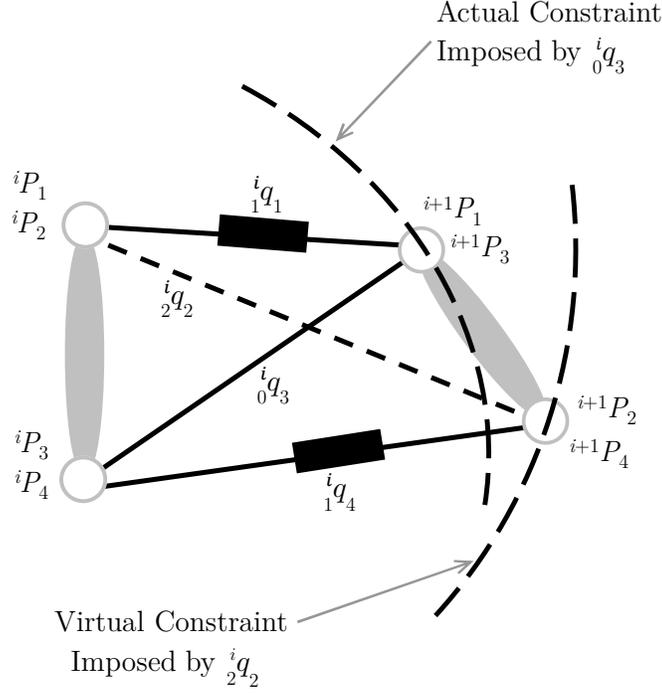


Figure 4.6: Virtual and actual kinematic constraints used in UVAC,

$$\text{pose} = {}_1^i \mathbf{X}_{i+1}^{\text{Path } 1}.$$

For the spatial case, i.e. the VGW, a suitable numerical method (see Appendix A) should be used to solve the kinematic constraints, presented by Eq. (4.13). However, in this planar case with a symmetric platform, the two constraint equations corresponding to ${}_1^i \mathbf{X}_{i+1}^{\text{Path } 1}$ can be analytically solved by simplifying them and expressing them as follows:

$${}^i q_3^2 - ({}^i b_{i+1_x} - l \sin {}^i \theta_{i+1_z})^2 + ({}^i b_{i+1_y} + l(\cos {}^i \theta_{i+1_z} + 1))^2 = 0 \quad (4.17)$$

$${}^2q_2^2 - ({}^1b_{i+1_x} + l \sin {}^1\theta_{i+1_z})^2 + \left({}^1b_{i+1_y} - l(\cos {}^1\theta_{i+1_z} + 1) \right)^2 = 0. \quad (4.18)$$

Although, the choice of the independent DOF is not important for the purpose of demonstrating the under-actuation, it would be of significance in the case of the VGW or any other practical application. Since $(\dim(\mathbf{X}_D) + 1) < \Upsilon$ there will be an infinite number of solutions to choose from. The optimality of the intermediate poses will be studied in Chapter 7. Without loss of generality the rotational DOF is allowed to be the independent one for this example, and given a value equal to that of the final pose, that is, ${}^1\theta_{i+1_z} = {}^2\theta_{i+1_z}$. With the final pose known, 0q_3 , 2q_2 , and ${}^1\theta_{i+1_z}$ can be substituted into Eqs. (4.17) and (4.18) to obtain ${}^1b_{i+1_x}$ and ${}^1b_{i+1_y}$, and solve the under-actuation problem. By letting $l = 1.0$, ${}^0\mathbf{X}_{i+1} = [1.000 \quad 0.000 \quad 0.0]^T$, and ${}^2\mathbf{X}_{i+1} = [1.500 \quad 0.500 \quad 30.0]^T$, the following results could be obtained, where all translational and rotational components are presented in m and deg, respectively:

$${}^1\mathbf{X}_{i+1}^{Path\ 1} = [1.301 \quad 0.013 \quad 30.0]^T.$$

The \mathbf{Q} vectors for the three stages will be

$${}^0\mathbf{Q} = [1.000 \quad 1.414 \quad 1.414 \quad 1.000]^T$$

$${}^1\mathbf{Q}^{Path\ 1} = [1.052 \quad 1.803 \quad 1.414 \quad 1.553]^T$$

$${}^2\mathbf{Q} = [1.323 \quad 1.803 \quad 1.902 \quad 1.840]^T.$$

All the steps required to achieve the under-actuated motion for the $\mathcal{P}2\text{-}2$ robot can be demonstrated as shown in Fig. 4.7. This figure illustrates how three Cartesian variables, i.e. x , y , and θ_z , can be controlled with only two actuators, with the use of two isostatic topologies.

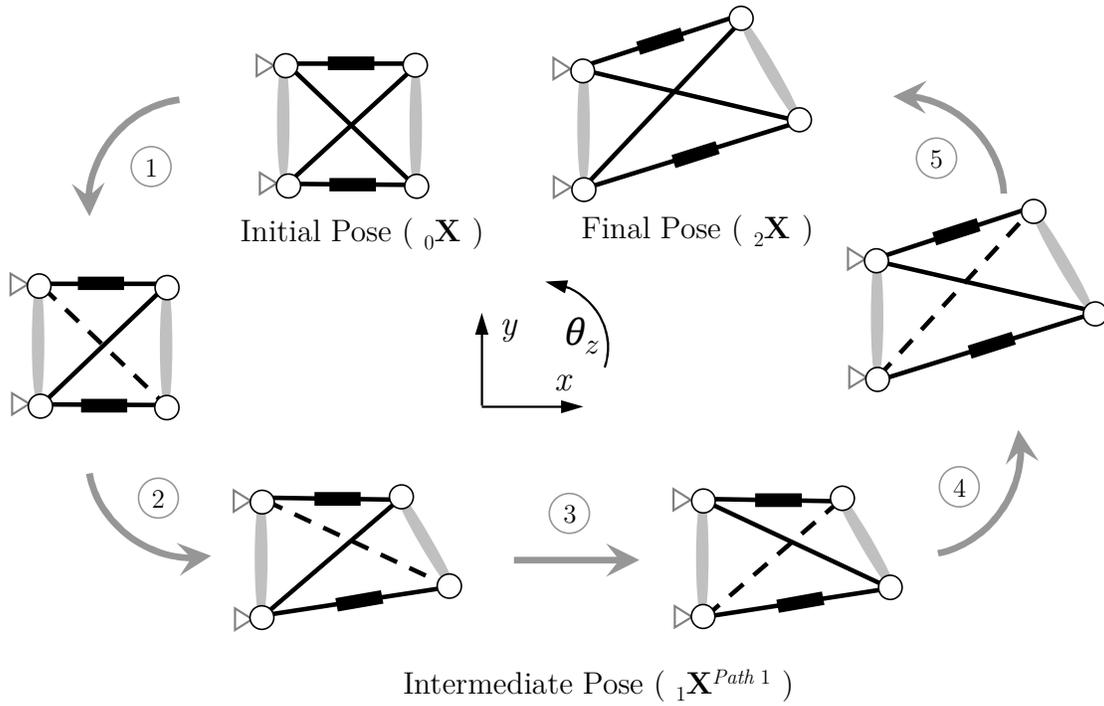


Figure 4.7: Under-actuation procedure for a planar mechanism with two actuators and two passive members using path 1.

As illustrated in Fig. 4.7 five steps are used to go from the initial pose to the final one, where each step represents a change in either the topology or the pose. A step is not to be confused with the previously defined stage, which only represents a change in the pose. In the first step, isostatic topology I is activated by unlocking one of the locked passive members. In the second step, some intermediate pose which satisfies the actual and the virtual constraints is realized. In the third step, isostatic topology II is activated by locking the previously unlocked passive member and unlocking the other passive member. In the fourth step, the final pose is realized. In the fifth step, the mechanism returns to the hyperstatic configuration, i.e. both passive members become locked. Alternatively, the same results could be achieved using path 2, as illustrated in Fig. 4.8.

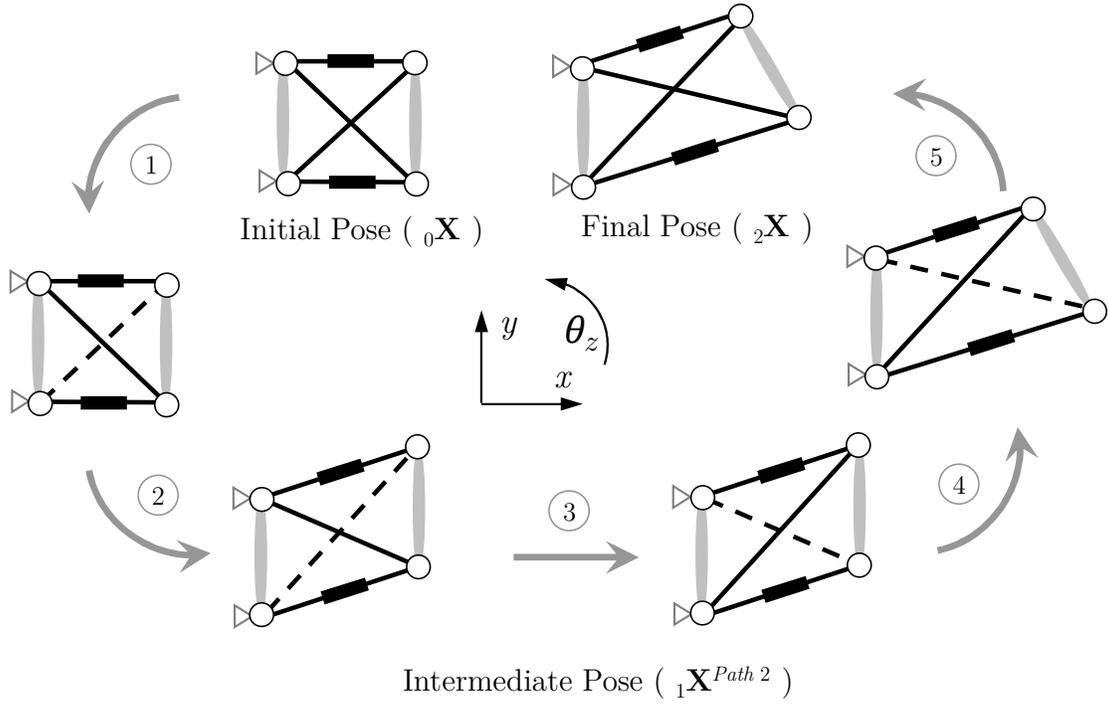


Figure 4.8: Under-actuation procedure for a planar mechanism with two actuators and two passive members using path 2.

The results corresponding to initial and final poses for path 2 are the same as those from path 1, while the results for the values for the intermediate stage are as follows:

$${}^i\mathbf{X}_{i+1}^{Path\ 2} = [1.283 \quad 0.405 \quad 0.0]^T$$

$${}^i\mathbf{Q}^{Path\ 2} = [1.345 \quad 1.414 \quad 1.902 \quad 1.345]^T.$$

In the case of path 2 the independent DOF is set to be ${}^i\theta_{i+1_z} = {}^i\theta_{i+1_z}$.

In general, the UVAC can be applied to generate more intermediate poses and sub-stages for higher fidelity. Ultimately, the choice of the independent parameter(s), their values, and the amount of discretization for the UVAC could be part of a universal optimization scheme that can capture responses related to modularization,

stiffness, kinematics, topology layout, and any other potential design criteria. In the case of the VGW, as will be discussed in detail in Chapter 7, far more paths exist than the simplified planar example presented in this section, leading to several possible solutions, and a more complex algorithm for the motion control.

4.3.3 Joint Placement for Under-actuation

It is worthwhile to note that in the case where the two passive limbs of the $\mathcal{P}2-2$ robot do not share a node on the $(i+1)^{\text{th}}$ platform, that is ${}^{i+1}P_2 \neq {}^{i+1}P_3$ (Fig. 4.9 *a*), multiple solutions for under-actuation exist. However, where the two passive limbs of the $\mathcal{P}2-2$ robot share a node on the $(i+1)^{\text{th}}$ platform, that is ${}^{i+1}P_2 = {}^{i+1}P_3$ (Fig. 4.9 *b*), there would generally be one solution in the reachable workspace of the manipulator. In the case of the $\mathcal{P}1-3$ robot, where ${}^{i+1}P_2 = {}^{i+1}P_3$ (Fig. 4.9 *c*), generally no solutions for under-actuation would exist within the reachable workspace. However, when the passive joints on the $(i+1)^{\text{th}}$ platform are separated in the case of the $\mathcal{P}1-3$ robot (Fig. 4.9 *d*), multiple solutions for under-actuation can be found.

As it can be observed, separating the joints associated with the passive limbs on the platform will generally lead to a wider range of solutions. From a practical perspective, when a passive member is locked, it is the pivoting motion about its end joint on the platform that allows the mechanism to achieve the desired length for the unlocked passive member. Therefore, if two passive members are sharing the same joint or are in very close proximity of each other on the platform, this will create an undesired redundancy in the under-actuation and one will not be able to control all the Cartesian variables with the minimum number of passive members. Therefore, in order to optimize the actuation procedure, care must be taken from a

layout design point of view to ensure that the passive limbs are not in very close proximities of each other.

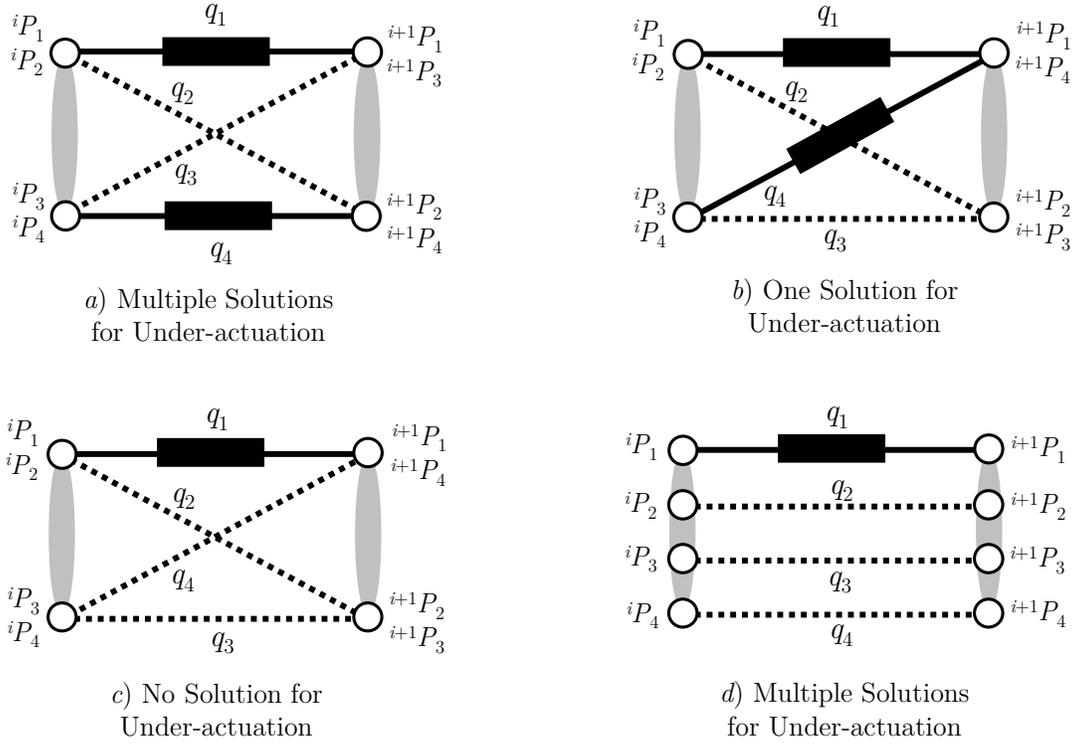


Figure 4.9: Topology effect on ability to under-actuate.

In order to avoid such configurations, one can impose the following condition on all PRES's with one lockable DOF per limb:

$$N_j^V = N_P \quad (4.19)$$

where, N_j^V and N_P are the number of joints connected to passive limbs on the platform, and the number of passive limbs, respectively. This relationship implies that passive limbs cannot share joints on the moving platform, if they are to be utilized in the under-actuation. However, if there are a sufficient number of passive

members to undertake the under-actuation, then any extra passive members that may be added for additional stiffness enhancement may be sharing joints on the moving platform.

4.4 Conclusion

The local and global kinematic formulations of the VGW were presented in this chapter, along with a proposed under-actuated motion methodology, UVAC. This novel method exploits the inherent static redundancy in the system to improve the actuation efficiency of the robot. A planar example was presented to demonstrate the application of the alternating constraint formulation, developed to accommodate the under-actuated motion of the PRES. The use of different paths for under-actuation was introduced in this chapter. The optimality of the paths will be investigated for the VGW in Chapter 7, where minimum-energy actuation is discussed.

Chapter 5

Kinetostatics

To design and optimize the topological and geometric features of VGW, one must be able to parametrically evaluate the stiffness and the internal loads (actuator and joint loads) for various loading conditions as a function of the pose and the configuration. Similar to the kinematic relationships, the static relationships can also be broken up into local and global components. The local statics is concerned with determining the internal loads in the i^{th} module for a given external load acting on the $(i+1)^{\text{th}}$ platform, while the global statics is concerned with determining the external forces and moments acting on each platform. Similarly, the local and the global stiffness are concerned with the stiffness of a single module and a series of modules, respectively.

5.1 Local Stiffness (Module Level)

Defining the stiffness terms and the coupling is highly dependent on the definition of the point of interest on the platform. From the motion control point of view, the choice is somewhat arbitrary, as identical motions can be obtained using different points. However, the most obvious and convenient choice would be the quarter-chord point ($1/4c$), since this will facilitate conventional wing/planform motion definitions, for example, a twist about the $1/4c$ point. From a stiffness point of view, the choice

expressed in the joint space. The term ${}^i\dot{\mathbf{X}}_{i+1}$ is the vector of Cartesian rates of the $(i+1)^{\text{th}}$ platform expressed in the i^{th} coordinate frame, that is, a twist vector containing the linear and angular velocities of a point on the platform, where ${}^i\dot{\mathbf{X}}_{i+1} \in \mathbb{R}^6$ for the VGW, that is, the spatial case. Although the analysis is applicable to both planar and spatial cases, for brevity, in the remainder of the derivations the presentation of the terms will be limited to the spatial case only. Matrices \mathbf{J}_{q_i} and \mathbf{J}_{X_i} can be expressed as

$$\mathbf{J}_{q_i} = \text{diag}[{}^i q_1, \dots, {}^i q_n] \quad (5.2)$$

$$\mathbf{J}_{X_i} = \begin{bmatrix} {}^i \mathbf{q}_1^T & {}^i \Psi_1^T \\ \vdots & \vdots \\ {}^i \mathbf{q}_n^T & {}^i \Psi_n^T \end{bmatrix} \quad (5.3)$$

where

$${}^i \mathbf{q}_j = {}^i \mathbf{b}_{i+1} + {}^i \rho_j - {}^i \mathbf{p}_j \quad (5.4)$$

$${}^i \Psi_j = {}^i \rho_j \times {}^i \mathbf{q}_j \quad (5.5)$$

and

$${}^i \rho_j = {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_j. \quad (5.6)$$

The term n represents the total number of limbs; ${}^i q_j$ is the length of the j^{th} limb in the i^{th} module; ${}^i \mathbf{q}_j$ is the j^{th} limb vector in the i^{th} coordinate frame; ${}^i \mathbf{p}_j$ represents

the position of the joint belonging to limb j on the i^{th} platform in the i^{th} coordinate frame; ${}^{i+1}\mathbf{p}_j$ represents the position of the joint belonging to limb j on the $(i+1)^{\text{th}}$ platform in the $(i+1)^{\text{th}}$ coordinate frame; ${}^i\mathbf{b}_{i+1}$ is the vector representing the coordinates of a point of interest on the $(i+1)^{\text{th}}$ platform in the i^{th} coordinate frame; and ${}^i\mathbf{R}_{i+1}$ represents the rotation matrix of the platform, as defined in Sec. 4.1.

Equation (5.4) is defined based on the loop closure equation, corresponding to Fig. 5.1. The previously introduced vectors are defined as follows:

$${}^i\mathbf{q}_j = \begin{bmatrix} {}^i q_{j_x} & {}^i q_{j_y} & {}^i q_{j_z} \end{bmatrix}^T \quad (5.7a)$$

$${}^i\mathbf{p}_j = \begin{bmatrix} {}^i p_{j_x} & {}^i p_{j_y} & {}^i p_{j_z} \end{bmatrix}^T \quad (5.7b)$$

$${}^{i+1}\mathbf{p}_j = \begin{bmatrix} {}^{i+1} p_{j_x} & {}^{i+1} p_{j_y} & {}^{i+1} p_{j_z} \end{bmatrix}^T \quad (5.7c)$$

$${}^i\mathbf{b}_{i+1} = \begin{bmatrix} {}^i b_{i+1_x} & {}^i b_{i+1_y} & {}^i b_{i+1_z} \end{bmatrix}^T \quad (5.7d)$$

$${}^i\boldsymbol{\rho}_j = \begin{bmatrix} {}^i \rho_{j_x} & {}^i \rho_{j_y} & {}^i \rho_{j_z} \end{bmatrix}^T \quad (5.7e)$$

$${}^i\boldsymbol{\Psi}_j = \begin{bmatrix} {}^i \Psi_{j_x} & {}^i \Psi_{j_y} & {}^i \Psi_{j_z} \end{bmatrix}^T \quad (5.7f)$$

where

$${}^i \Psi_{j_x} = {}^i \rho_{j_y} {}^i q_{j_z} - {}^i \rho_{j_z} {}^i q_{j_y} \quad (5.8a)$$

$${}^i\Psi_{j_y} = {}^i\rho_{j_z} {}^i q_{j_x} - {}^i\rho_{j_x} {}^i q_{j_z} \quad (5.8b)$$

$${}^i\Psi_{j_z} = {}^i\rho_{j_x} {}^i q_{j_y} - {}^i\rho_{j_y} {}^i q_{j_x}. \quad (5.8c)$$

Equation (5.1) can also be expressed as

$$\dot{\mathbf{q}}_i = \mathbf{J}_i {}^i\dot{\mathbf{X}}_{i+1} \quad (5.9)$$

where

$$\mathbf{J}_i = \mathbf{J}_{q_i}^{-1} \mathbf{J}_{X_i}. \quad (5.10)$$

Matrix \mathbf{J}_i , referred to as the Jacobian, is the mapping from the Cartesian velocity vector to the joint velocity vector for module i , that is,

$$\mathbf{J}_i = \begin{bmatrix} \frac{\partial {}^i q_1}{\partial {}^i b_{i+1_x}} & \frac{\partial {}^i q_1}{\partial {}^i b_{i+1_y}} & \frac{\partial {}^i q_1}{\partial {}^i b_{i+1_z}} & \frac{\partial {}^i q_1}{\partial {}^i \Gamma_{i+1}} & \frac{\partial {}^i q_1}{\partial {}^i \Lambda_{i+1}} & \frac{\partial {}^i q_1}{\partial {}^i \alpha_{i+1}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial {}^i q_n}{\partial {}^i b_{i+1_x}} & \frac{\partial {}^i q_n}{\partial {}^i b_{i+1_y}} & \frac{\partial {}^i q_n}{\partial {}^i b_{i+1_z}} & \frac{\partial {}^i q_n}{\partial {}^i \Gamma_{i+1}} & \frac{\partial {}^i q_n}{\partial {}^i \Lambda_{i+1}} & \frac{\partial {}^i q_n}{\partial {}^i \alpha_{i+1}} \end{bmatrix}. \quad (5.11)$$

From Eq. (5.9) it can be concluded that

$$\delta \mathbf{q}_i = \mathbf{J}_i \delta {}^i \mathbf{X}_{i+1} \quad (5.12)$$

where $\delta \mathbf{q}_i$ and $\delta {}^i \mathbf{X}_{i+1}$ are the joint and Cartesian infinitesimal displacements, respectively. By applying the principle of kinematic/static duality, one can write

$${}^i \mathbf{F}_{i+1} = \mathbf{J}_i^T \mathbf{f}_i \quad (5.13)$$

where \mathbf{f}_i is the vector of actuator/locked passive member loads associated with module i defined in the joint space, and ${}^i \mathbf{F}_{i+1}$ is the vector of forces and torques acting on a point on the $(i+1)^{\text{th}}$ platform expressed in the i^{th} coordinate frame, that is, a wrench vector, where ${}^i \mathbf{F}_{i+1} \in \mathbb{R}^6$ for the VGW. In the presented forms of Eqs. (5.12) and (5.13), \mathbf{J}_i and \mathbf{J}_i^T are in fact the compatibility and the equilibrium matrices associated with module i , respectively. The size of the matrix and the values that make up \mathbf{J}_i will vary depending on configuration.

The actuator/locked passive member forces can be related to their displacements using Hooke's law:

$$\mathbf{f}_i = \mathbf{K}_{J_i} \delta \mathbf{q}_i \quad (5.14)$$

where

$$\mathbf{K}_{J_i} = \text{diag}[{}^i k_1, \dots, {}^i k_n] = {}_{c f g} \mathbf{C}_i \boldsymbol{\kappa}_i \quad (5.15)$$

and

$$\boldsymbol{\kappa}_i = \text{diag}[{}^i \kappa_1, \dots, {}^i \kappa_n]. \quad (5.16)$$

Matrix \mathbf{K}_{J_i} is the joint stiffness matrix of the i^{th} module. Terms ${}^i\kappa_1$ through ${}^i\kappa_n$ represent the equivalent stiffness of each limb when it is load-bearing. Actuators are always load-bearing, whereas passive members are load-bearing when they are locked. To minimize alterations in dimensions and forms of the Jacobian and the joint vectors, a configuration matrix, ${}_{cf} \mathbf{C}_i$, is used to change the stiffness of a passive limb when it is unlocked. Depending on the numbering convention used, ${}_{cf} \mathbf{C}_i$ can take on different forms, however it will be a $n \times n$ diagonal matrix regardless of the numbering and its components will take on values of zero and one, representing unlocked passive limbs, and locked passive/actuated limbs, respectively. For example, when all the passive members are locked, that is, a hyperstatic configuration, ${}_{cf} \mathbf{C}_i \equiv {}_{hyp} \mathbf{C}_i = \mathbf{I}$, where $\mathbf{I} \in \mathbb{R}^{8 \times 8}$ refers to the identity matrix³. The terms ${}^i k_1$ through ${}^i k_n$ represent the modified limb stiffness, where ${}^i k_j = {}^i \kappa_j$ if the corresponding limb is an actuator or a locked passive member, and ${}^i k_j = 0$ if the corresponding limb is an unlocked passive member. Alternatively, the Jacobians and the joint vectors could be modified to achieve the same results, instead of using a configuration matrix. Nevertheless, by substituting Eq. (5.12) into Eq. (5.14) one has

$$\mathbf{f}_i = \mathbf{K}_{J_i} \mathbf{J}_i \delta {}^i \mathbf{X}_{i+1}. \quad (5.17)$$

Then, by substituting Eq. (5.17) into Eq. (5.13) one can obtain

$${}^i \mathbf{F}_{i+1} = \mathbf{J}_i^T \mathbf{K}_{J_i} \mathbf{J}_i \delta {}^i \mathbf{X}_{i+1}. \quad (5.18)$$

³ the choice for eight load-bearing limbs will become clear in Sec. 6.2.1

By assuming that the platform portion of the VGW is rigid, the equivalent stiffness matrix of the $(i+1)^{\text{th}}$ platform, solely due to the compliance of the members, expressed in the i^{th} coordinate frame becomes

$${}^i\mathbf{K}_{C_{i+1}} = \mathbf{J}_i^T \mathbf{K}_{J_i} \mathbf{J}_i. \quad (5.19)$$

By substituting Eq. (5.10) into Eq. (5.19), ${}^i\mathbf{K}_{C_{i+1}}$ can be expressed as

$${}^i\mathbf{K}_{C_{i+1}} = \mathbf{J}_{X_i}^T \mathbf{B}_i \mathbf{J}_{X_i} \quad (5.20)$$

where

$$\mathbf{B}_i = \mathbf{J}_{q_i}^{-T} \mathbf{K}_{J_i} \mathbf{J}_{q_i}^{-1}. \quad (5.21)$$

Matrix \mathbf{B}_i can be expressed as

$$\mathbf{B}_i = \text{diag} [{}^i\beta_1, \dots, {}^i\beta_n] \quad (5.22)$$

where

$${}^i\beta_j = \frac{{}^i k_j}{{}^i q_j^2}. \quad (5.23)$$

By substituting Eqs. (5.22) and (5.3) into Eq. (5.20) one has

$${}^i\mathbf{K}_{C_{i+1}} = \begin{bmatrix} {}^i\mathfrak{K}_{i+111} & {}^i\mathfrak{K}_{i+112} \\ {}^i\mathfrak{K}_{i+121} & {}^i\mathfrak{K}_{i+122} \end{bmatrix} \quad (5.24)$$

where

$${}^i\mathcal{K}_{i+111} = \sum_{j=1}^n {}^i\beta_j ({}^i\mathbf{q}_j {}^i\mathbf{q}_j^T) \quad (5.25)$$

$${}^i\mathcal{K}_{i+112} = \sum_{j=1}^n {}^i\beta_j ({}^i\mathbf{q}_j {}^i\Psi_j^T) \quad (5.26)$$

$${}^i\mathcal{K}_{i+121} = \sum_{j=1}^n {}^i\beta_j ({}^i\Psi_j {}^i\mathbf{q}_j^T) \quad (5.27)$$

$${}^i\mathcal{K}_{i+122} = \sum_{j=1}^n {}^i\beta_j ({}^i\Psi_j {}^i\Psi_j^T). \quad (5.28)$$

The 6×6 Cartesian stiffness matrix can be expressed by expanding ${}^i\mathcal{K}_{i+111}$, ${}^i\mathcal{K}_{i+112}$, ${}^i\mathcal{K}_{i+121}$, and ${}^i\mathcal{K}_{i+122}$:

$$\begin{aligned} {}^i\mathcal{K}_{i+111} &= \begin{bmatrix} {}^iK_{i+111} & {}^iK_{i+112} & {}^iK_{i+113} \\ {}^iK_{i+121} & {}^iK_{i+122} & {}^iK_{i+123} \\ {}^iK_{i+131} & {}^iK_{i+132} & {}^iK_{i+133} \end{bmatrix} \\ &= \begin{bmatrix} \sum_{j=1}^n {}^i\beta_j {}^i q_{jx}^2 & \sum_{j=1}^n {}^i\beta_j {}^i q_{jx} {}^i q_{jy} & \sum_{j=1}^n {}^i\beta_j {}^i q_{jx} {}^i q_{jz} \\ & \sum_{j=1}^n {}^i\beta_j {}^i q_{jy}^2 & \sum_{j=1}^n {}^i\beta_j {}^i q_{jy} {}^i q_{jz} \\ \text{SYM} & & \sum_{j=1}^n {}^i\beta_j {}^i q_{jz}^2 \end{bmatrix} \quad (5.29) \end{aligned}$$

$${}^i\mathcal{K}_{i+122} = \begin{bmatrix} {}^iK_{i+144} & {}^iK_{i+145} & {}^iK_{i+146} \\ {}^iK_{i+154} & {}^iK_{i+155} & {}^iK_{i+156} \\ {}^iK_{i+164} & {}^iK_{i+165} & {}^iK_{i+166} \end{bmatrix} \quad (5.30)$$

$$\begin{aligned}
&= \begin{bmatrix} \sum_{j=1}^n i\beta_j i\Psi_{j_x}^2 & \sum_{j=1}^n i\beta_j i\Psi_{j_x} i\Psi_{j_y} & \sum_{j=1}^n i\beta_j i\Psi_{j_x} i\Psi_{j_z} \\ \text{SYM} & \sum_{j=1}^n i\beta_j i\Psi_{j_y}^2 & \sum_{j=1}^n i\beta_j i\Psi_{j_y} i\Psi_{j_z} \\ & & \sum_{j=1}^n i\beta_j i\Psi_{j_z}^2 \end{bmatrix} \\
{}^i\mathcal{K}_{i+1_{12}} &= \begin{bmatrix} {}^iK_{i+1_{14}} & {}^iK_{i+1_{15}} & {}^iK_{i+1_{16}} \\ {}^iK_{i+1_{24}} & {}^iK_{i+1_{25}} & {}^iK_{i+1_{26}} \\ {}^iK_{i+1_{34}} & {}^iK_{i+1_{35}} & {}^iK_{i+1_{36}} \end{bmatrix} \\
&= \begin{bmatrix} \sum_{j=1}^n i\beta_j i q_{j_x} i\Psi_{j_x} & \sum_{j=1}^n i\beta_j i q_{j_x} i\Psi_{j_y} & \sum_{j=1}^n i\beta_j i q_{j_x} i\Psi_{j_z} \\ \sum_{j=1}^n i\beta_j i q_{j_y} i\Psi_{j_x} & \sum_{j=1}^n i\beta_j i q_{j_y} i\Psi_{j_y} & \sum_{j=1}^n i\beta_j i q_{j_y} i\Psi_{j_z} \\ \sum_{j=1}^n i\beta_j i q_{j_z} i\Psi_{j_x} & \sum_{j=1}^n i\beta_j i q_{j_z} i\Psi_{j_y} & \sum_{j=1}^n i\beta_j i q_{j_z} i\Psi_{j_z} \end{bmatrix} \quad (5.31)
\end{aligned}$$

where

$${}^i\mathcal{K}_{i+1_{12}} = {}^i\mathcal{K}_{i+1_{21}}^T. \quad (5.32)$$

Although the formulation can be easily extended to account for the compliance of the platform, to simplify the formulation it is assumed that the platform portion of the mechanism is rigid. The derived components of ${}^i\mathbf{K}_{C_{i+1}}$ will be used in the upcoming optimization process.

5.2 Local Statics

To obtain a generalized form which could be applied to all isostatic and hyperstatic topologies for all PRES's, one can decompose Eq. (5.13) as follows:

$${}^i\mathbf{F}_{i+1} = [\mathbf{J}_{i_A}^T \quad \mathbf{J}_{i_P}^T]_{\Upsilon \times n} \begin{bmatrix} \mathbf{f}_{i_A} \\ \mathbf{f}_{i_P} \end{bmatrix}_{n \times 1} \quad (5.33)$$

where $n \geq \Upsilon$. Each configuration can be classified into one of the two following categories: *a*) isostatic, where $n = \Upsilon$; and *b*) hyperstatic, where $n > \Upsilon$. Terms *A* and *P* refer to actuated and passive limbs, respectively. Using the presented active/passive decomposition, for the isostatic cases where one or more of the passive limbs may become unlocked, the corresponding rows or columns can be simply deleted and the matrices can be reduced to a square form. Thus, in the statically determinate cases for the VGW, the internal loads, ${}_{iso}\mathbf{f}_i$, can be simply calculated by inverting \mathbf{J}_i^T , which is a 6×6 matrix:

$${}_{iso}\mathbf{f}_i = \mathbf{J}_i^{-T} {}^i\mathbf{F}_{i+1}. \quad (5.34)$$

To obtain the internal loads for the hyperstatic case, short of using a pseudo-inverse to invert a non-square Jacobian, the compliance of the limbs could be used to determine the internal loads. Similar to Eq. (5.34), Eqs. (5.12) and (5.19) can also be decomposed to active and passive components:

$$\begin{bmatrix} \delta \mathbf{q}_{i_A} \\ \delta \mathbf{q}_{i_P} \end{bmatrix}_{n \times 1} = \begin{bmatrix} \mathbf{J}_{i_A} \\ \mathbf{J}_{i_P} \end{bmatrix}_{n \times \Upsilon} \delta {}^i\mathbf{X}_{i+1} \quad (5.35)$$

$${}^i\mathbf{K}_{C_{i+1}} = [\mathbf{J}_{i_A}^T \quad \mathbf{J}_{i_P}^T] \begin{bmatrix} \mathbf{K}_{J_{i_A}} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{J_{i_P}} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{i_A} \\ \mathbf{J}_{i_P} \end{bmatrix}. \quad (5.36)$$

In order to obtain the general form for internal loads one could substitute

$$\delta {}^i \mathbf{X}_{i+1} = {}^i \mathbf{K}_{C_{i+1}}^{-1} {}^i \mathbf{F}_{i+1} \quad (5.37)$$

into Eq. (5.35) to get

$$\delta \mathbf{q}_i = \mathbf{J}_i {}^i \mathbf{K}_{C_{i+1}}^{-1} {}^i \mathbf{F}_{i+1}. \quad (5.38)$$

Finally, by substituting Eq. (5.38) into Eq. (5.14) one obtains

$${}_{hyp} \mathbf{f}_i = \mathbf{K}_{J_i} \mathbf{J}_i {}^i \mathbf{K}_{C_{i+1}}^{-1} {}^i \mathbf{F}_{i+1} \quad (5.39)$$

where ${}_{hyp} \mathbf{f}_i$ is a $n \times 1$ vector representing the internal loads for the hyperstatic configuration. The general form of the Jacobian for the i^{th} module can be expressed as

$$\mathbf{J}_i = \begin{bmatrix} {}^i \mathbf{q}_1^T / {}^i q_1 & {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_1 \times {}^i \mathbf{q}_1 / {}^i q_1 \\ \vdots & \vdots \\ {}^i \mathbf{q}_n^T / {}^i q_n & {}^i \mathbf{R}_{i+1} {}^{i+1} \mathbf{p}_n \times {}^i \mathbf{q}_n / {}^i q_n \end{bmatrix} \quad (5.40)$$

where ${}^i q_n$ is the length of the n^{th} load-bearing member of the i^{th} module. The size of the matrix and the values that make up \mathbf{J}_i will vary depending on configuration.

5.3 Global Stiffness (Wing Level)

After determining the local stiffness of each platform isolated from the rest of the structure, it would be useful to determine the actual stiffness at any particular platform with the contribution of the preceding kinematic modules. This will be referred to as the global or the upstream stiffness. In order to do this, one must first

transform the load and displacement vectors associated with each platform from some reference coordinate system, for example the global system denoted by 0, to the local coordinate system of that platform. These could be expressed as

$$\delta {}^0\mathbf{X}_{i+1} = {}^0\mathbf{T}_i \delta {}^i\mathbf{X}_{i+1} \quad (5.41)$$

$${}^0\mathbf{F}_{i+1} = {}^0\mathbf{T}_i {}^i\mathbf{F}_{i+1} \quad (5.42)$$

where $\delta {}^i\mathbf{X}_{i+1}$ denotes the displacement vector associated with the $(i+1)^{\text{th}}$ platform in the i^{th} coordinate system. Similarly, ${}^i\mathbf{F}_{i+1}$ denotes the load vector containing the external forces and moments acting on the $(i+1)^{\text{th}}$ platform in the i^{th} coordinate system; $\delta {}^0\mathbf{X}_{i+1}$ and ${}^0\mathbf{F}_{i+1}$ express similar vectors in the global coordinate frame; ${}^0\mathbf{T}_i$ is the transformation matrix from the i^{th} to the global coordinate system and can be expressed as

$${}^0\mathbf{T}_i = \begin{bmatrix} {}^0\mathbf{R}_i & \mathbf{0} \\ \mathbf{0} & {}^0\mathbf{R}_i \end{bmatrix} \quad (5.43)$$

where ${}^0\mathbf{R}_i$ represents the orientation of the i^{th} platform in the global reference frame. Here, the global coordinate system is used as the reference to express the overall stiffness of the structure. In general, any frame of interest could be used to find the equivalent stiffness.

The direct stiffness relationship of the $(i+1)^{\text{th}}$ platform expressed in the i^{th} coordinate system, using the adopted notation would be

$${}^i\mathbf{F}_{i+1} = {}^i\mathbf{K}_{C_{i+1}} \delta {}^i\mathbf{X}_{i+1}. \quad (5.44)$$

By rearranging Eqs. (5.41) and (5.42) and substituting them into Eq. (5.44), one has

$${}^0\mathbf{T}_i^T {}^0\mathbf{F}_{i+1} = {}^i\mathbf{K}_{C_{i+1}} {}^0\mathbf{T}_i^T \delta {}^0\mathbf{X}_{i+1} \quad (5.45)$$

where ${}^0\mathbf{T}_i^T \equiv {}^i\mathbf{T}_0$ and ${}^0\mathbf{T}_i^T \equiv {}^0\mathbf{T}_i^{-1}$. Thus, Eq. (5.45) can be simply expressed as

$${}^0\mathbf{F}_{i+1} = {}^0\mathbf{T}_i {}^i\mathbf{K}_{C_{i+1}} {}^i\mathbf{T}_0 \delta {}^0\mathbf{X}_{i+1}. \quad (5.46)$$

Since

$${}^0\mathbf{F}_{i+1} = {}^0\mathbf{K}_{C_{i+1}} \delta {}^0\mathbf{X}_{i+1} \quad (5.47)$$

where ${}^0\mathbf{K}_{C_{i+1}}$ is the stiffness of the $(i+1)^{\text{th}}$ module expressed in the global coordinate system, it can be concluded that

$${}^0\mathbf{K}_{C_{i+1}} = {}^0\mathbf{T}_i {}^i\mathbf{K}_{C_{i+1}} {}^i\mathbf{T}_0. \quad (5.48)$$

With every module expressed in the global coordinate system, the equivalent compliance of any platform of the VGW can be evaluated by adding the compliances of the preceding modules. Notation \mathbf{K}_G is used to distinguish between the stiffness of a module with and without accounting for the preceding modules which are laid out in series. The term ${}^0\mathbf{K}_{G_{i+1}}$ represents the equivalent stiffness of the $(i+1)^{\text{th}}$ module expressed in the global coordinate, while accounting for the compliance of

the preceding modules. Therefore, the global compliance of the a^{th} platform can be expressed in the global system, as

$${}^0\mathbf{K}_{G_a}^{-1} = \sum_{i=0}^a ({}^0\mathbf{K}_{C_{i+1}}^{-1}). \quad (5.49)$$

Finally, expressed in the global coordinate frame, the global stiffness of the a^{th} platform becomes

$${}^0\mathbf{K}_{G_a} = \left(\sum_{i=0}^a ({}^0\mathbf{T}_i {}^i\mathbf{K}_{C_{i+1}}^{-1} {}^i\mathbf{T}_0) \right)^{-1}. \quad (5.50)$$

Using the presented kinetostatic model, one can determine the global stiffness of any platform as a function of its pose.

5.4 Global Statics

The multi-module static relationships, i.e. global statics, of the modular morphing wing are similar to those of a serial robot with each module being represented by an equivalent link in the serial chain. The two major differences are the inclusion of the aerodynamic loads, and the dependency of these loads on the changes in geometry, as will be discussed in Sec. 5.5.

The loads acting on any platform can be classified into two general groups: intrinsic loads and extrinsic loads⁴. The *intrinsic* loads, \mathbf{F}^{IN} , acting on the $(i+1)^{\text{th}}$ platform include aerodynamic and inertia loads that are directly generated by the $(i+1)^{\text{th}}$ module. The *extrinsic* loads, \mathbf{F}^{EX} , acting on the same platform include the

⁴ The intrinsic and extrinsic classifications are not to be confused with internal and external ones, as both intrinsic and extrinsic loads are considered external loads. Furthermore, intrinsic loads are not to be mistaken with loads that are generated by the actuation system, for instance a spike in the hydraulic system pressure beyond what is required to counteract the external loads; such loads may occur in a jam case.

contribution of the forces and the resulting moments from the outboard modules. The total external loads can then be symbolically expressed as

$${}^i\mathbf{F}_{i+1} = {}^i\mathbf{F}_{i+1}^{IN} + {}^i\mathbf{F}_{i+1}^{EX} \quad (5.51)$$

where

$${}^i\mathbf{F}_{i+1} = \begin{pmatrix} \mathbf{F}_F \\ \mathbf{F}_M \end{pmatrix}_{i+1} = \begin{pmatrix} [F_x & F_y & F_z]^T \\ [M_x & M_y & M_z]^T \end{pmatrix}_{i+1} \quad (5.52)$$

consists of forces, \mathbf{F}_F , and moments, \mathbf{F}_M , defined in the corresponding Cartesian frame.

5.4.1 Intrinsic Loads

In the proposed formulation, the loads acting on the wing are discretized by modules, making it convenient to transform the external loads as required. Figure 5.2 illustrates the transformation of the distributed loads acting on the wing into discrete intrinsic loads acting on each platform.

The term ${}^{i+1}w_{i+1}^{aero}$ represents the distributed aerodynamic loads acting on the $(i+1)^{\text{th}}$ module defined in the $(i+1)^{\text{th}}$ coordinate frame. The term ${}^0w_{i+1}^{inertia}$ represents the distributed inertia loads acting on the $(i+1)^{\text{th}}$ module defined in the global coordinate frame. Similarly, ${}^{i+1}\mathbf{F}_{i+1}^{aero}$ and ${}^*\mathbf{F}_{i+1}^{inertia}$ represent the equivalent aerodynamic and the inertia loads acting on the $(i+1)^{\text{th}}$ platform, respectively. The force components in ${}^*\mathbf{F}_{i+1}^{inertia}$ are defined in the global coordinate frame, whereas the moment components are defined in the $(i+1)^{\text{th}}$ coordinate frame.

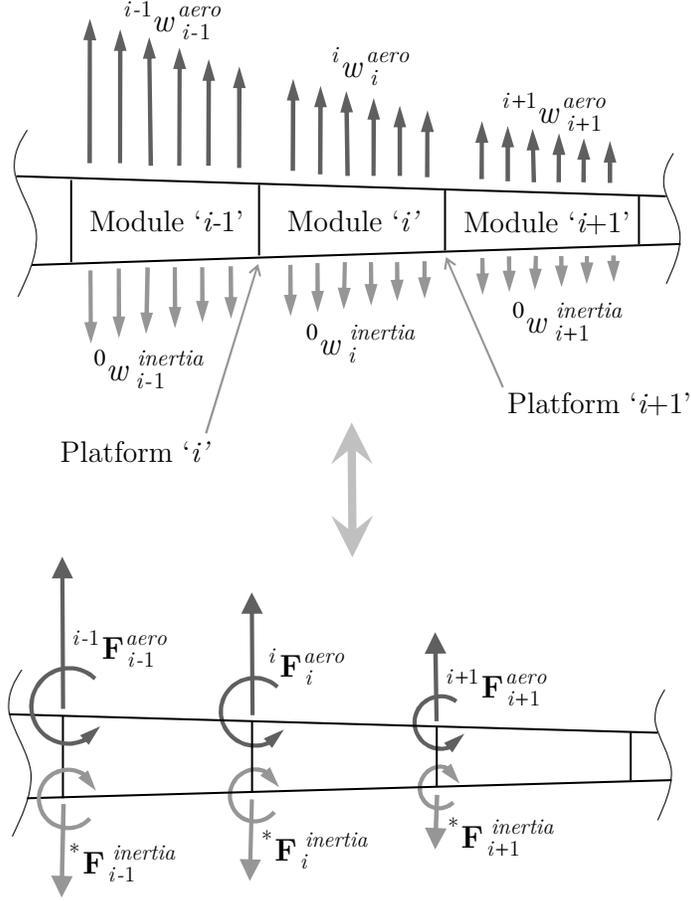


Figure 5.2: View looking aft on the wing showing the intrinsic loads discretized by module.

Any appropriate function could be used to define the load distributions. Without loss of generality, chordal distributions are used for the simulation, as presented in Sec. 7.6.3. The total intrinsic loads acting on the $(i+1)^{\text{th}}$ platform can be expressed as

$${}^i\mathbf{F}_{i+1}^{IN} = {}^i\mathbf{F}_{i+1}^{aero} + {}^i\mathbf{F}_{i+1}^{inertia} \quad (5.53)$$

where

$${}^i\mathbf{F}_{i+1}^{aero} = {}^i\mathbf{T}_{i+1} {}^{i+1}\mathbf{F}_{i+1}^{aero} \quad (5.54)$$

$${}^i\mathbf{F}_{i+1}^{inertia} = {}^i\mathbf{T}_* {}^*\mathbf{F}_{i+1}^{inertia}. \quad (5.55)$$

Transformation matrices ${}^i\mathbf{T}_{i+1}$ and ${}^i\mathbf{T}_*$ are defined as

$${}^i\mathbf{T}_{i+1} = \begin{bmatrix} {}^i\mathbf{R}_{i+1} & \mathbf{0} \\ \mathbf{0} & {}^i\mathbf{R}_{i+1} \end{bmatrix} \quad (5.56)$$

$${}^i\mathbf{T}_* = \begin{bmatrix} {}^0\mathbf{R}_i^T & \mathbf{0} \\ \mathbf{0} & {}^i\mathbf{R}_{i+1}^* {}^i\mathbf{R}_{i+1} \end{bmatrix} \quad (5.57)$$

where

$${}^i\mathbf{R}_{i+1}^* = \begin{bmatrix} \cos({}^0\Gamma_{i+1}) & 0 & 0 \\ 0 & \cos({}^0\Lambda_{i+1}) & 0 \\ 0 & 0 & \cos({}^0\alpha_{i+1}) \end{bmatrix}. \quad (5.58)$$

The term ${}^i\mathbf{R}_{i+1}^*$ captures the projection of the moment arm associated with the inertia loads acting on the $(i+1)^{\text{th}}$ module onto the i^{th} coordinate frame. The angles are defined with respect to the global frame. The significance of the frames used to define the intrinsic loads, the physical meanings of the transformations, and their range of validity will be discussed in Sec. 5.5.

5.4.2 Extrinsic Loads

In addition to the intrinsic loads acting on a platform, the contribution from the outboard sections also need to be considered when determining the total loads acting on the platform. The so-called extrinsic loads can be derived using the wrench

transformation, which transforms the total loads acting on the $(i+2)^{\text{th}}$ platform into equivalent forces and moments acting on the $(i+1)^{\text{th}}$ platform, while transforming the coordinate frame from $(i+1)$ to i .

The extrinsic loads acting on the $(i+1)^{\text{th}}$ platform in the i^{th} coordinate frame, resulting from the total loads acting on the $(i+2)^{\text{th}}$ platform, which are conventionally defined in the $(i+1)^{\text{th}}$ coordinate frame can be expressed as

$${}^i\mathbf{F}_{i+1}^{EX} = {}^i\tilde{\mathbf{T}}_{i+1} {}^{i+1}\mathbf{F}_{i+2} \quad (5.59)$$

where

$${}^i\tilde{\mathbf{T}}_{i+1} = \begin{bmatrix} {}^i\mathbf{R}_{i+1} & \mathbf{0} \\ {}^i\mathbf{R}_{i+1} {}^{i+1}\tilde{\mathbf{b}}_{i+2} & {}^i\mathbf{R}_{i+1} \end{bmatrix}. \quad (5.60)$$

The term ${}^{i+1}\tilde{\mathbf{b}}_{i+2}$ represents the skew symmetric cross product matrix of ${}^{i+1}\mathbf{b}_{i+2}$, expressed as

$${}^{i+1}\tilde{\mathbf{b}}_{i+2} = \begin{matrix} {}^{i+1} \\ \begin{bmatrix} 0 & -b_z & b_y \\ b_z & 0 & -b_x \\ -b_y & b_x & 0 \end{bmatrix} \\ {}^{i+2} \end{matrix}. \quad (5.61)$$

Once the resultant external load vector acting on each platform is defined using the global statics, the local statics is used to obtain the internal loads.

5.5 Geometry Dependent Loading

Any changes in the sweep, cant, twist, or span of the wing will alter the distribution of the loads acting on the wing. Although determining the exact distributions is beyond the scope of this work, a modified representation of the internal loads has

been introduced in Sec. 5.4, which will allow for the inclusion of geometry dependent loading.

It should be recalled that the i^{th} coordinate frame was used to define the aerodynamic loads acting on the i^{th} platform, which are the resultant of the distributed aerodynamic loads acting on the i^{th} module. This means that as module i changes its pose, the corresponding intrinsic aerodynamic loads will automatically change their orientation. Additionally, the intrinsic inertia loads acting on the i^{th} platform, which are the resultant of the distributed inertia loads acting on the i^{th} module were defined using two coordinate frames. The forces were defined in the global axis, which implies that they will not be affected by changes in the pose, whereas the moments induced by the inertia loads were defined in the i^{th} coordinate frame. This was done to account for the pose dependency of the inertia-induced moments with the help of a special matrix ${}^i\mathbf{R}_{i+1}^*$ (Eq. (5.57)). Generally, this transformation is valid for maximum accumulative rotations of up to 90 deg applicable to all three rotational DOFs, with an input for ${}^*\mathbf{F}_{i+1}^{inertia}$ (Eq. (5.55)), which corresponds to the wing configuration that gives the largest inertia-induced moments. This simplified projection approach for dealing with inertia-induced moments is a crude but reasonable approximation. A more general approach could be adopted using the wrench transformation, similar to the one presented in Sec. 5.4.2 for dealing with extrinsic loads. It should be noted that the previously mentioned inertia loads only include those induced by the gravity. Due to the relatively low speed of the mechanism and the proposed usage type, that is, performance enhancement, other inertia loads have been ignored. However, they can be easily included if required as an extension to the current loads.

Using only the transformation of the loads will ignore any variation in the magnitude of the loads as a function of the pose change, such as an increase in lift due to an increase in the angle of incidence, or an increase in the bending moment due to a span increase. These variations in magnitude can be considered using a weighting matrix, which can modify the load vector to correct for all the aforementioned parameters. Additionally, further enhancements could be made to the formulation to account for changes in the parasitic drag as a function of changes in the geometry. For simplicity, for the case-study presented in Chapter 7, these effects are ignored.

5.6 Alternative Jacobian

Alternatively, the Jacobian could be decomposed into an isostatic component which may include a combination of actuators and passive members, and a redundant component that includes the additional locked passive members required to transform the isostatic topology into a hyperstatic one. Hence the compatibility relationship can be written as follows:

$$\begin{bmatrix} \delta \mathbf{q}_{i_I} \\ \delta \mathbf{q}_{i_R} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{i_I} \\ \mathbf{J}_{i_R} \end{bmatrix} \delta {}^i \mathbf{X}_{i+1} \quad (5.62)$$

where I and R refer to the isostatic and redundant portions of the matrix, respectively. Depending on the class and the architecture of the manipulator, the isostatic portion of the Jacobian could contain only actuated limbs or a combination of actuated and locked passive limbs. Nonetheless, \mathbf{J}_{i_I} will be a square matrix. From Eq. (5.62) one can obtain

$$\delta \mathbf{q}_{i_R} = \mathbf{J}_{i_R} \mathbf{J}_{i_I}^{-1} \delta \mathbf{q}_{i_I}. \quad (5.63)$$

By decomposing the equilibrium relationship in a similar fashion, one has

$${}^i \mathbf{F}_{i+1} = \begin{bmatrix} \mathbf{J}_{i_I}^T & \mathbf{J}_{i_R}^T \end{bmatrix} \begin{bmatrix} \mathbf{K}_{J_{i_I}} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{J_{i_R}} \end{bmatrix} \begin{bmatrix} \delta \mathbf{q}_{i_I} \\ \delta \mathbf{q}_{i_R} \end{bmatrix}. \quad (5.64)$$

By substituting Eq. (5.63) into Eq. (5.64) and some further manipulation, it can be shown that

$${}^i \mathbf{F}_{i+1} = \mathbf{J}_{i_I}^T \mathbf{f}_{i_I} + \mathbf{J}_{i_R}^T \mathbf{K}_{J_{i_R}} \mathbf{J}_{i_R} \mathbf{J}_{i_I}^{-1} \mathbf{K}_{J_{i_I}}^{-1} \mathbf{f}_{i_I}. \quad (5.65)$$

Using a modified Jacobian, \mathbf{J}_i^+ , where $\mathbf{J}_i^+ \in \mathbb{R}^{\Upsilon \times \Upsilon}$, one could write

$${}^i \mathbf{F}_{i+1} = \mathbf{J}_i^{+T} \mathbf{f}_{i_I} \quad (5.66)$$

where

$$\mathbf{J}_i^+ = \left[\mathbf{J}_{i_I}^T + \mathbf{J}_{i_R}^T \mathbf{K}_{J_{i_R}} \mathbf{J}_{i_R} \mathbf{J}_{i_I}^{-1} \mathbf{K}_{J_{i_I}}^{-1} \right]^T. \quad (5.67)$$

The presented kinetostatic and stiffness formulations in this chapter can be applied to all classes of PRES's for both isostatic and hyperstatic topologies.

In addition to the presented methodology to model the local stiffness of the VGW, other means could be used to do the same. Appendix C presents the derivation of the stiffness model of the VGW using the Finite Element Method (FEM).

5.7 Case Study: \mathcal{S} -4-4 PRES vs. Gough-Stewart vs. Pentapod

An example of a statically redundant robot is proposed in Fig. 5.3.

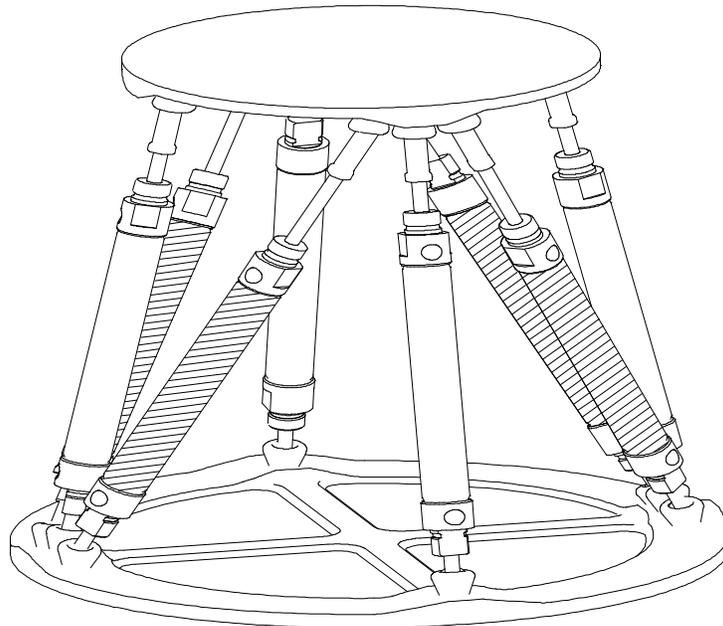


Figure 5.3: Proposed variable topology robot with lower instantaneous mobility, full finite mobility, and enhanced stiffness and static characteristics, classified as an \mathcal{S} -4-4 PRES.

The robot illustrated by Fig. 5.3 has an instantaneous mobility of four and a finite mobility of six, with four actuated limbs and four passive limbs which can be locked and unlocked to vary the topology of the manipulator. The four actuated limbs are distinguished using dashed lines in Fig 5.3. The limbs, which are hydraulic cylinders, are attached to the base and the platform using spherical joints. Schematically, the topology can be illustrated by Fig. 5.4. This reconfigurable PKM can achieve full finite mobility through under-actuation and has multiple degrees of

fault tolerance. This robot can be classified as an \mathcal{S} -4-4 PRES. It will be demonstrated shortly that such a robot has comparable stiffness to that of a hexapod when isostatic, while exhibiting superior stiffness when hyperstatic. Additionally, it will be shown that this robot has far superior stiffness characteristics compared to a conventional lower mobility manipulator with a constraining limb, of similar overall dimensions and joint spacing.

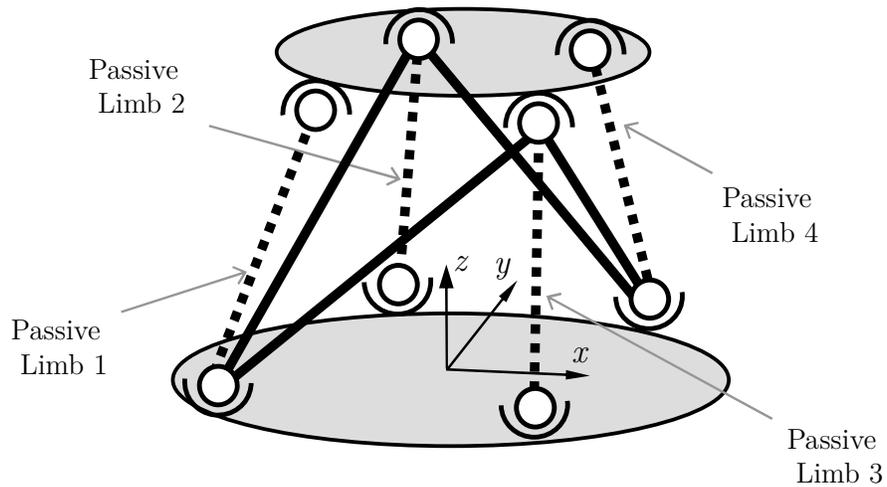


Figure 5.4: Schematic representation of the proposed robot.

By using the ground topology shown in Fig. 5.4 one can generate six potential isostatic topologies. However, two of these will be nearly degenerate and generally not usable. Those are the topologies with passive limbs 1, 4 or 2, 3 unlocked simultaneously. In addition, two other robots are included in this study. The first one is a Gough-Stewart manipulator with the same overall dimensions and joint spacing as the proposed \mathcal{S} -4-4 PRES. The second one is a pentapod with four identical \mathcal{UP} limbs, which are active, and a \mathcal{P} constraining limb, which is passive. Figure 5.5 summarizes the schematics of the six various robots/topologies that are included in the case-study.

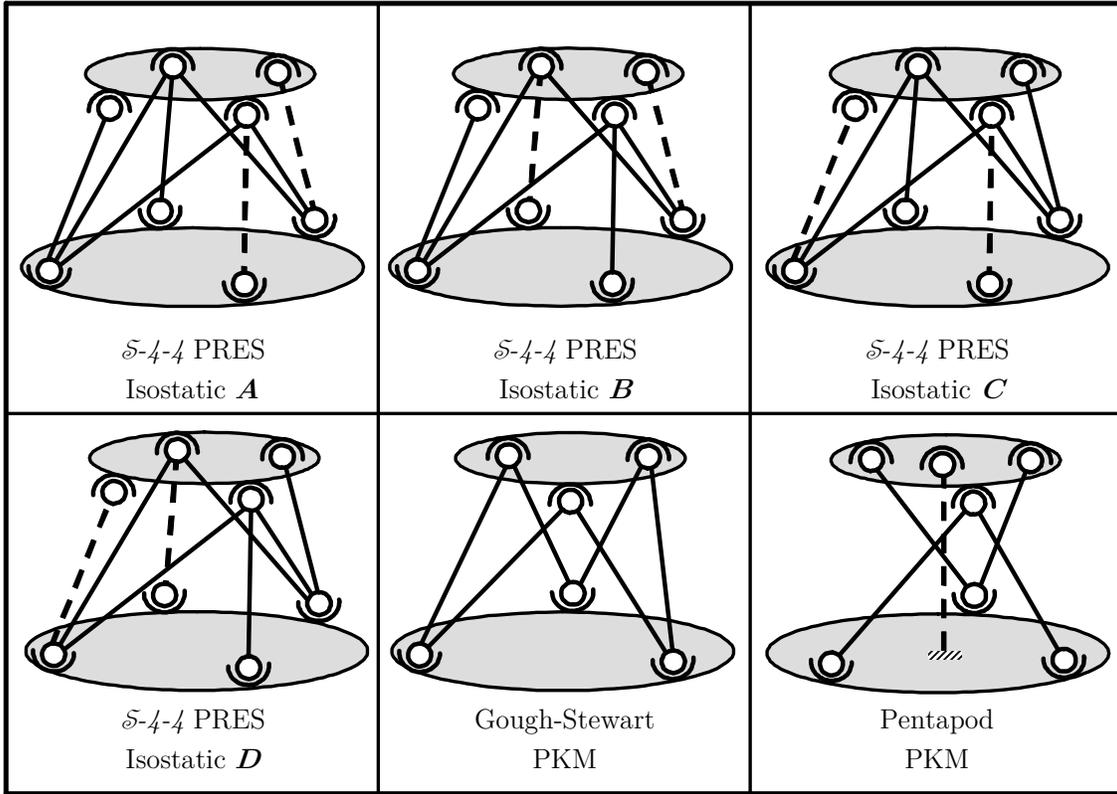


Figure 5.5: Six different isostatic topologies used in the case-Study, four of which are generated from the hyperstatic ground topology of the proposed robot.

The topology used for the pentapod yields constrained x and y motions. Therefore, two fictitious linear springs aligned with the x and the y Cartesian axes are used to calculate the Cartesian stiffness of the platform. The two constraints will appear as $[1 \ 0 \ 0 \ 0 \ 0 \ 0]$ and $[0 \ 1 \ 0 \ 0 \ 0 \ 0]$ in the two rows of the Jacobian relating δq_5 to δx and δq_6 to δy , respectively. δq_5 and δq_6 are infinitesimal displacements associated with the two fictitious springs. For simplicity it is assumed that the stiffnesses of the constraining leg are ten times that of a load-bearing $\mathcal{UP}\mathcal{S}/\mathcal{SP}\mathcal{S}$ limb. Also, it is assumed that all the other limbs have the same stiffness. This reduces Eq. (5.19) to

$${}^i\mathbf{K}_{C_{i+1}} = k_{eq}\mathbf{J}_i^T\mathbf{J}_i \quad (5.68)$$

where k_{eq} is the equivalent axial stiffness of a $\mathcal{UP}\mathcal{S}/\mathcal{SP}\mathcal{S}$ limb. The relative stiffness of these robots/topologies are evaluated by comparing the lowest eigenvalue of the Cartesian stiffness matrix. To obtain a dimensionally-uniform stiffness matrix for solving the eigenvalue problem, a weighting matrix, \mathbf{W}_{f_i} , is used [106]:

$$\mathbf{W}_{f_i} = \text{diag}[1, 1, 1, \mathcal{L}_i^{-1}, \mathcal{L}_i^{-1}, \mathcal{L}_i^{-1}] \quad (5.69)$$

where \mathcal{L}_i^{-1} is a parameter with the dimension of length; for the purpose of this study, one can let $\mathcal{L}_i = \|\mathbf{b}_{i+1}\|$. The dimensionally-uniform stiffness matrix, ${}^i\tilde{\mathbf{K}}_{C_{i+1}}$, may then be expressed as

$${}^i\tilde{\mathbf{K}}_{C_{i+1}} = \mathbf{W}_{f_i} {}^i\mathbf{K}_{C_{i+1}} \mathbf{W}_{f_i}. \quad (5.70)$$

The following measure of stiffness could then be used as a relative index to compare different robots/topologies in various poses:

$$k_{min} = \min\left(\text{eig}\left({}^i\tilde{\mathbf{K}}_{C_{i+1}}\right)\right). \quad (5.71)$$

The variation in k_{min} for $\mathbf{X}_r = [0 \ 0.3 \ 0 \ 0 \ 0 \ {}^i\alpha_{i+1}]$ is illustrated in Fig. 5.6. The term \mathbf{X}_r represents a subset of poses within the Cartesian space of the robot. All the translational components are in m, and all the rotational components are in deg; the directions are defined based on the coordinate frame illustrated in Fig. 5.4.

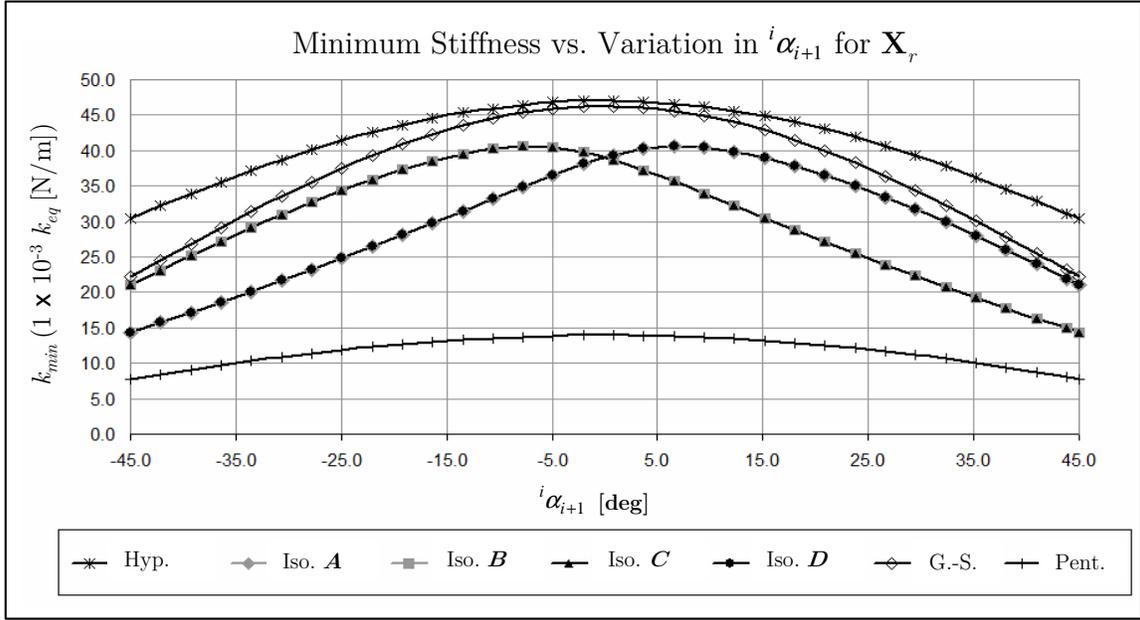


Figure 5.6: Variation of minimum stiffness as a function of topology and pose (rotation about the z axis).

The variation in minimum stiffness as a function of topology and pose can be clearly observed in Fig. 5.6. Compared to a $3-3$ Gough-Stewart platform, the $\mathcal{S}-4-4$ PRES offers the same finite mobility with less number of actuators, while offering comparable stiffness. As expected, isostatic topologies A and D have the same minimum stiffness due to the symmetry of the robot. Similarly, isostatic topologies B and C share the same characteristic.

Additionally, it can be seen that the $\mathcal{S}-4-4$ PRES possesses superior stiffness compared to the pentapod with four DOFs. Although the eigenvectors associated with the derived dimensionally-uniform principal stiffness terms are generally not aligned with the Cartesian coordinate system, it was observed that the dominant DOF associated with the lowest eigenvalue for the pentapod was consistently the rotational DOF about the z axis. This explains the poor performance of the pentapod compared to the $\mathcal{S}-4-4$ PRES when comparing the lowest principal stiffnesses. The

only way to improve the torsional stiffness of the pentapod, i.e. the component associated with the rotation about the z axis, is by replacing the \mathcal{PS} constraining limb with one that constrains the rotational motion about the z axis such as the $\mathcal{RP}\mathcal{U}$ limb or the $\mathcal{RR}\mathcal{U}$ limb similar to the one proposed in [107]. However, such a constraining limb will completely eliminate the rotational DOF of the platform about the z axis, and significantly reduce the workspace of the robot. Also, the torsional stiffness of the robot will then be mostly dependent on the torsional stiffness of the constraining limb and less so on the remainder of the topological and geometric features. Therefore, it would serve no practical purpose to include a rotationally constrained system in this comparison.

5.8 Conclusion

The local and global kinetostatic and stiffness models of the VGW were presented in this chapter. The models can be used to develop both an optimal design and an optimal motion control algorithm. Both of these will be explored in the following chapters.

Furthermore, to display the effectiveness of the proposed design, a case-study was conducted using a spatial PRES. In this particular example, the results indicated that the proposed manipulator can attain stiffness characteristics comparable to that of a Gough-Stewart manipulator of similar dimensions and joint spacing, and superior stiffness to a lower mobility manipulator with a constraining limb. Although when dealing with a regular design space, the stiffness benefits of the proposed robot compared to that of a conventional hexapod do not appear drastic, when the design space starts to become less conventional, as in a wing, the stiffness variations become more significant. The actual design space for the VGW will be explored in the following chapter to obtain the optimal configuration.

Chapter 6

Optimal Configuration Design

The optimal configuration design of the reconfigurable mechanism poses an interesting design problem which can be broken up into two distinct sets: 1) the optimal limb configuration, and 2) the optimal topological configuration, both of which are addressed in this chapter. The optimal limb configuration problem seeks the optimal design of the kinematic joints and links that make up the mechanism to satisfy a series of static and kinematic design requirements. On the other hand, the optimal topological configuration seeks the minimal compliance solution to the placement of the aforementioned limbs within the design space, while satisfying a series of design requirements such as fail-safe criteria, and minimal aeroelastic impact. In general, this type of configuration optimization is comparable to structural optimization [108-110], which has been extensively studied and applied in the field of structural design (see, for example, [111]) and compliant mechanism design (see, for example, [112]). However, with the exception of a few [113-115], the compliance-based configuration optimization has not been widely applied in the design of active mechanisms and robotics. In the field of topological optimization for robotics, it is more common to use the condition number of the robot as the objective (see, for example, [116,117]), as opposed to its compliance. The *condition number* is a measure of the isotropy of the robot, for instance from the stiffness or the velocity

point of view, over the anticipated workspace. Such measures will ensure that the static and kinematic abilities of the robot do not change drastically as the manipulator moves through the workspace. The rather structural approach of compliance-based optimization formulation for the wing morphing application will be discussed in detail in this chapter. Additionally, the design of the optimal limb architecture will be discussed and the optimal limb configuration will be presented. The optimal limb architecture is then used as a building block in the compliance-based topological optimization of the VGW, formulated as a multi-stage optimization process. Finally, the optimal configuration is presented along with a rigidity validation of the optimal design using a prototype.

6.1 Optimal Limb Configuration

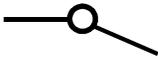
Unlike conventional spatial mechanisms, generally not designed to be reconfigurable, the VGW must be able to reconfigure itself without adversely affecting its kinematic capabilities. This requirement imposes a special constraint on the limb architecture. The architecture for the lockable passive limb must be such that when unlocked in an isostatic configuration, it must not reduce the maximum instantaneous mobility of the mechanism, and while locked, it must impose the constraint needed to generate static redundancy. The only limb architecture that can have no effect on the aforementioned mobility by imposing a potential constraint is one with a connectivity of six. The *connectivity* of a limb is defined as the total number of DOFs associated with all the joints in that limb. This could be appreciated intuitively, as a spatial rigid body, that is, the moving platform, can undergo rigid body motion in six DOFs; therefore, any limb attached to the platform that can support all six DOFs without imposing any kinematic constraints will not affect the mobility of the moving platform. However, when one or more non-redundant DOFs are removed from that

limb, as many constraints are imposed on the motion of the moving platform. The switch between isostatic and hyperstatic states affects the mobility of the system by adding and removing these constraints, which will in effect introduce one or more degrees of static redundancy to the system.

Therefore, to find the most optimal limb architecture(s) for the VGW, one shall only search for limbs that can provide six non-redundant DOFs. For practicality, the search will be limited to lower kinematic pairs, that is, line and point contact, for example cam pairs, will be excluded from the search. Additionally, the planar joint will also be excluded from the search due to practical considerations. This leaves the kinematic pairs shown in Table 6.1 to be used in the search.

A combinatorial search could be performed to find all suitable limb architectures that meet the minimum required condition of having six DOFs. Meeting the minimum requirement for the enumeration is a necessary condition for the limb architecture; however, this is not a sufficient condition due to the existence of possible kinematic redundancies in the architecture. For example, consider the limb architecture shown in Fig. 6.1. This architecture has six DOFs; however, despite meeting the minimum enumeration requirement, body **B** cannot rotate about the axis normal to the viewing plane with respect to body **A**. In order to enable this DOF, additional joint(s) must be added to the limb, resulting in a total of more than six DOFs. Although in general, having more than six DOFs suggests the potential for the existence of kinematic redundancies in the limb, in many cases, such redundancies do not contribute to the motions of the body/platform, for example, the free rotation of a link about its own axis. Nonetheless, to maximize the search space for the optimal architecture, the possibility for the existence of such redundancies will be considered.

Table 6.1: Kinematic pairs: description and schematic.

Joint	Symbol	DOFs	Motion	Schematic
Helical	\mathcal{H}	1	Coupled Translation and Rotation	
Prismatic	\mathcal{P}	1	Translation	
Revolute	\mathcal{R}	1	Rotation	
Cylindrical	\mathcal{C}	2	Uncoupled Translation and Rotation	
Universal	\mathcal{U}	2	Uncoupled Rotations	
Spherical	\mathcal{S}	3	Uncoupled Rotations	

Six DOFs will be used as the lower bound of the search as previously discussed, whereas nine will be used as the upper bound. The reason for choosing nine as the upper bound is due to the limit imposed on the number of joints per limb. To minimize the complexity of the system, the number of joints per limb is limited to three. Also, since the maximum number of DOFs per joint is three (see Table 6.1), one shall never have more than nine DOFs per limb. It should also be highlighted that each joint will be connected to the next one via a rigid link. Table 6.3 represents all the possible permutations having a maximum of three joints with six, seven, eight, and nine DOFs. The summation of a connectivity permutation equals the total number of limb DOFs, while the numbers that make up the permutation represent

the DOFs of the joints in the limb. For instance, 123 refers to a six-DOF architecture with a one-DOF joint connected to a two-DOF joint, which is connected to a three-DOF joint.

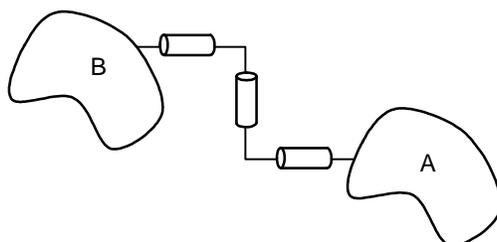


Figure 6.1: An example of a limb architecture that meets the necessary enumeration condition but has less than six non-redundant DOFs.

Table 6.2: Connectivity permutations with three or less joints.

Σ DOF = 6	Σ DOF = 7	Σ DOF = 8	Σ DOF = 9
123	133	233	333
132	313	323	
213	331	332	
231	322		
312	232		
321	223		
222			
33			

In the case of limbs with three joints, the two numbers on the left and the right side represent the DOFs of the joints attaching the limbs to platforms i and $(i+1)$, respectively. The middle number represents the DOF of the intermediate joint. In the case of the limb with only two joints, that is 33 , there is no intermediate joint. The only possibility for this architecture is two \mathfrak{S} joints attached by a link. Although the total number of DOFs is six, this architecture can only provide five non-

redundant DOFs, that is, it can never allow any extension of the limb, which constitutes a kinematic constraint. Therefore, the 33 architecture will be excluded from the search. All the possible architectures (with the exception of 33) and their corresponding types and total DOFs are tabulated in Table 6.3.

Table 6.3: Possible architectures with three joints.

Σ DOF	Type	Architecture
6	123	$\mathcal{HCS}, \mathcal{PCS}, \mathcal{RCS}, \mathcal{HUS}, \mathcal{PUS}, \mathcal{RUS}$
6	132	$\mathcal{HCU}, \mathcal{PCU}, \mathcal{RCU}, \mathcal{HSC}, \mathcal{PSC}, \mathcal{RSC}$
6	213	$\mathcal{CUS}, \mathcal{CPS}, \mathcal{CRS}, \mathcal{UCS}, \mathcal{UPS}, \mathcal{URS}$
6	231	$\mathcal{CSK}, \mathcal{CSP}, \mathcal{CSR}, \mathcal{USK}, \mathcal{USP}, \mathcal{USR}$
6	312	$\mathcal{SKC}, \mathcal{SPC}, \mathcal{SRC}, \mathcal{SKU}, \mathcal{SPU}, \mathcal{SRU}$
6	321	$\mathcal{SKC}, \mathcal{SPC}, \mathcal{SRC}, \mathcal{SKU}, \mathcal{SPU}, \mathcal{SRU}$
6	222	$\mathcal{CUU}, \mathcal{UCU}, \mathcal{UUC}, \mathcal{UCC}, \mathcal{CUC}, \mathcal{CCU}, \mathcal{CCC}, \mathcal{UUU}$
7	133	$\mathcal{KSS}, \mathcal{PSS}, \mathcal{RSS}$
7	313	$\mathcal{SKS}, \mathcal{SPS}, \mathcal{RSK}$
7	331	$\mathcal{SSK}, \mathcal{SSP}, \mathcal{SSR}$
7	322	$\mathcal{SCK}, \mathcal{SCP}, \mathcal{SRC}, \mathcal{SKU}, \mathcal{SPU}, \mathcal{SRU}$
7	232	$\mathcal{CSK}, \mathcal{CSP}, \mathcal{CSR}, \mathcal{USK}, \mathcal{USP}, \mathcal{USR}$
7	223	$\mathcal{CCK}, \mathcal{CCK}, \mathcal{UCC}, \mathcal{UCC}, \mathcal{CUC}, \mathcal{CCU}, \mathcal{CCC}, \mathcal{UUU}$
8	233	$\mathcal{CSS}, \mathcal{USS}$
8	323	$\mathcal{SKS}, \mathcal{SPS}, \mathcal{RSK}$
8	332	$\mathcal{SSK}, \mathcal{SSP}, \mathcal{SSR}$
9	333	\mathcal{SSS}

All architectures with intermediate \mathcal{R} , \mathcal{U} , or \mathcal{S} joints, for example, the \mathcal{RSCU} or the \mathcal{PUS} architecture, are excluded from further consideration. This is done for two reasons: 1) geometrical constraints, and 2) structural design considerations. Primarily, any architecture that cannot maintain a compact form is not suitable for

the proposed application. Architectures with an \mathcal{R} , \mathcal{U} , or \mathcal{S} intermediate joint are susceptible to a folding motion when actuation is occurring, as illustrated by Fig. 6.2. Due to the highly restrictive design space of the VGW, any folding architecture can greatly reduce the chances of finding the most structurally optimal configuration, by imposing complex geometrical constraints on limb placement. Secondly, from a structural design perspective, it would be sub-optimal to introduce architectures into the system that can induce bending at the joints that are attached to the platforms. Any architecture that can take on a folding motion can induce a bending moment at the platform, when combined with a one-DOF joint at the platform. An example of such a configuration is the $\mathcal{R}\mathcal{S}\mathcal{S}$ architecture.

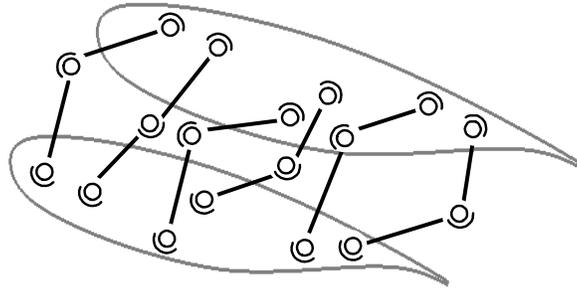


Figure 6.2: Highly geometrically constrained design with folding limbs.

Furthermore, any remaining architectures with less than six non-redundant DOFs are excluded from the list. These include architectures with a total of six DOFs possessing the following pairs of joints in their architecture: \mathcal{P} and \mathcal{C} , \mathcal{C} and \mathcal{K} , \mathcal{R} and \mathcal{C} , \mathcal{U} and \mathcal{C} , \mathcal{C} and \mathcal{C} , and architectures with a total of seven DOFs with at least two \mathcal{C} joints. Some examples of such architecture are the $\mathcal{P}\mathcal{C}\mathcal{S}$ and the $\mathcal{U}\mathcal{C}\mathcal{C}$ architectures. It should be noted that the compactness requirement is implicitly considered in all these exclusions. For instance, in the case of the excluded $\mathcal{U}\mathcal{C}\mathcal{C}$ architecture, it is assumed that the translational axes of the \mathcal{C} joints are in line, leading to the

generation of redundant DOFs. Placing these axes perpendicular to each other can generate a limb with six non-redundant DOFs; however, this would violate the compactness requirement.

Apart from those excluded above, the remaining architectures all satisfy the sufficient enumeration conditions of possessing six non-redundant DOFs, as well as the previously discussed geometrical and structural constraints. These architectures will be referred to as the optimal set and are listed as follows: $\mathcal{UP}\mathcal{S}$, $\mathcal{UK}\mathcal{S}$, $\mathcal{SP}\mathcal{U}$, $\mathcal{SK}\mathcal{U}$, $\mathcal{UC}\mathcal{U}$, $\mathcal{SP}\mathcal{S}$, $\mathcal{SK}\mathcal{S}$, $\mathcal{SC}\mathcal{U}$, $\mathcal{UC}\mathcal{S}$, and $\mathcal{SC}\mathcal{S}$. Depending on the actuation system and the required workspace of the mechanism, some of these configurations may be more suitable than others. For example, if mechanical actuation is used, the $\mathcal{SK}\mathcal{S}$ architecture, where \mathcal{K} represents a jackscrew actuator, may be the more suitable candidate for the actuated members, whereas the $\mathcal{UC}\mathcal{S}$ architecture may be more suitable where hydraulic actuation is used, with \mathcal{C} representing a hydraulic actuator. In general, the optimal architectures can be used in conjunction with any type of actuation system including, but not limited to, mechanical, hydraulic, or SMA systems.

To accommodate modularity in design, it would be desirable to use the same architecture for both actuated and passive members. The only difference between the two is that the former would be actively actuated, and the latter would be able to lock and unlock to generate static redundancy in the mechanism without requiring any energy. Thus, the same joint that would be actuated for an active limb would ideally be chosen as the lockable one for a passive limb. From a kinematic perspective, any of the passive joints could be locked or actuated. However, from a practical point of view, joints with one DOF are generally more suitable for actuation or locking. Furthermore, applying the actuation or locking to the \mathcal{U} or the \mathcal{S} joint

can unnecessarily complicate the mechanical design. From a structural perspective, having locked \mathcal{U} and \mathcal{S} joints could impose bending at the platforms which would not be desirable. While mindful of these implications, it is deemed most suitable to actuate/lock only the intermediate \mathcal{P} , \mathcal{K} , or the axial component of the \mathcal{C} joints. Without loss of generality, this will be adopted as the method for actuation and locking for the VGW in the remainder of this work.

With the intermediate axial DOF locked/actuated via the \mathcal{P} , \mathcal{K} , or \mathcal{C} joint, the actuated/locked limb is effectively a spatial truss element with only axial stiffness. When the intermediate joint is unlocked, the limb will have no stiffness. Therefore, without loss of generality, all these optimal architectures, which have similar static characteristics, will be grouped together and referred to as the $\mathcal{S}\mathcal{P}\mathcal{S}$ family, which can statically take on only two forms: 1) actuated/locked, and 2) unlocked. These two forms will be used as the building blocks for the design of the VGW in the upcoming sections. Figure 6.3 illustrates the schematics of a six-member configuration with the optimal limb architecture.

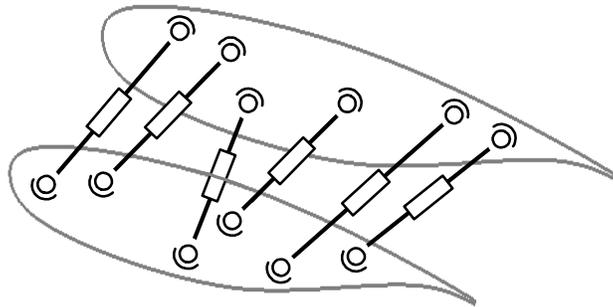


Figure 6.3: Configuration with the optimal limb architecture (from the $\mathcal{S}\mathcal{P}\mathcal{S}$ family).

The passive limbs can be directionally locked to provide additional measures against load uncertainties (for example a gust case encountered while in actuation), this feature will not affect the choice for the optimal configuration of the VGW; thus,

it is not included in the formulation. The implementation for the directional lockability is explained in detail in Appendix D.

6.2 Optimal Topological Configuration

To determine the most optimal configuration for the VGW, in addition to the optimal limb architecture, one must determine the optimal number of actuators and passive members, as well as their placements, which will constitute the optimal topological configuration. Before doing so, it is important to discuss the requirement for symmetry in the optimal configuration.

6.2.1 Symmetry

As previously noted, when seeking the optimal configuration for a robotic manipulator it is popular to use the velocity or stiffness isotropy of the end-effector (or platform) as the measure of optimality. However, such measures are not applicable to the present application due to the anisotropic nature of the design space and the loading, that is, large loads in shear, bending, torsion, and smaller magnitudes in other DOFs. Therefore, a more suitable measure of optimality could be the directional stiffness for the DOFs of interest. In the context of wing design, and particularly the VGW, shear, bending, and torsion DOFs, as identified in Fig. 5.1, correspond to a translation along the i^{th} y axis, a rotation about the i^{th} x axis, and a rotation about the i^{th} z axis, respectively.

The minimum requirements for the architecture of the PRES with common limbs, presented in Sec. 3.6, necessitates the limbs to have a connectivity of six, such as the $\mathfrak{S}\mathfrak{P}\mathfrak{S}$ limb. Such a mechanism must have a minimum of seven limbs connecting the two platforms to be able to both under-actuate and form a hyperstatic configuration. With seven limbs and only one lock per passive limb, the system will have one degree

of static redundancy. However, this could be increased by adding additional lockable passive limbs to the system. For the VGW, an additional passive lockable limb is included, so it will have a total of eight limbs. Eight is the lowest number that meets the minimum enumeration requirement and can generate symmetrical designs (as will be shown shortly, symmetry is a requirement for the optimal design). Although a seven-member configuration could also form a symmetric design by placing one of the members at the center of the module, such a configuration is inherently sub-optimal for the proposed application. The optimal design must be tailored to accommodate the typical loading encountered by a wing, that is, large shear, bending, and torsion loads. Placing a member at the center of the module will essentially align that member with the neutral axes of sections of interest, causing the member to have minimal contribution to the shear, bending, and torsional stiffness of the module. Although this may be intuitively appreciated, one can inspect the stiffness terms associated with the shear, bending, and torsion discussed in Sec. 6.2.2.1 to confirm this.

The aeroelastic properties of the wing, in particular in the case of flutter, are driven by several parameters, such as the wing geometry, airfoil shape, stiffness, and mass distribution. Consequently, the aeroelastic tailoring for any given wing will vary depending on these parameters, and is beyond the scope of this work. However, since the intent is to create a generic framework for the VGW, one can employ some design guidelines in the optimization process to improve the aeroelastic performance of the system. One of these measures is to uncouple the torsion and bending motion of the wing to improve its flutter characteristics [118]. In reality, eliminating this coupling can prove to be a very difficult task, specifically in the case of a wing with varying geometry. However, doing so for the cuboid design space in the home pose

is deemed to have some merit in the context of creating a generic framework for the VGW.

The bending-torsion coupling is typically viewed from the global point of view of the wing, where the bending is in fact a result of the shear force acting on the wing caused by the aerodynamic and inertia loads. Therefore, in order to have no bending-torsion coupling, one must attempt to eliminate the offset between the shear center and the mass center for the sections of interest, that is, the platforms in the case of the VGW. The only way this could be possible is if some form of symmetry or anti-symmetry exists in the topology. Therefore, symmetry is imposed as a constraint in the initial stage of the optimization process. In the later stages of the optimization, the symmetry constraint is replaced by actual coupling constraints implemented using a displacement method.

6.2.2 Optimization Formulation

With the minimum design requirements in mind, the design space will be explored in search of the optimal configuration for the VGW. The same design space taken up by a conventional wing-box, that is, the space between a rear and a front spar, will be utilized for the VGW, as illustrated by Fig 6.4. Each module will be separated by a platform/rib.

The highlighted volume in Fig. 6.4 will be used as the design space for the optimization. In order to simplify the design problem, taper and sweep effects will be neglected. Detailed sizing of actuators and passive members can offset the effect of such geometrical irregularities. Therefore, in order to develop the generic optimal configuration for the VGW, the design space will be idealized as a rectangular cuboid, with the two platforms acting as the boundaries of each module. The advantages of limiting the design space to that of a conventionally configured wing-box, made up

of spars, ribs, and stiffened panels, will become apparent when two or more conventionally configured sections of the wing are attached to each other via morphing modules (see, for example, Fig. 4.2). This will ensure that the structural elements that are responsible for the load transfer within the wing from a non-actuating section to an actuating section and back to a non-actuating section are all contained within the box and away from the leading edge and the trailing edge of the wing.

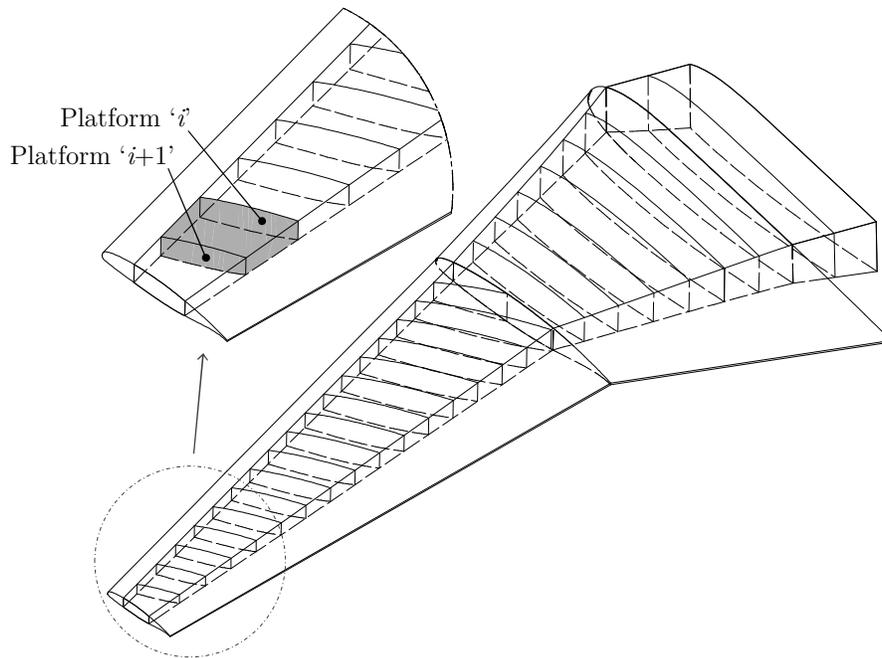


Figure 6.4: Representation of the cuboid design space.

Furthermore, since this work is concerned with generating a common generic framework for the VGW, which could be applicable to all modules, only the home pose represented by the cuboid will be used. In practice, any changes in the pose of the platform will result in changes in the stiffness of the module and thus the aeroelastic properties of the wing. However, if an appropriate number of modules are used with practical considerations in mind, drastic shape changes to a single module

can be minimized. Therefore, the general static behavior of the module in terms of internal load directions and approximate magnitudes will not change, as long as these shape changes are within some practical limits. See the example in Sec. 7.6 for typical limits for the proposed applications of the VGW.

Structural optimization problems are typically broken up into three different classes, based on their geometrical features. The first one is sizing optimization. This is when the geometrical feature to be optimized represents some form of thickness such as the cross section of a truss member. The second class is shape optimization. In this case the geometrical feature to be optimized represents the form or contour of some part of the boundary of the structural domain. The last and the most general class of optimization problems is topology optimization, which is concerned with the nodal connectivity. Figure 6.5 illustrates the three types of structural optimization.

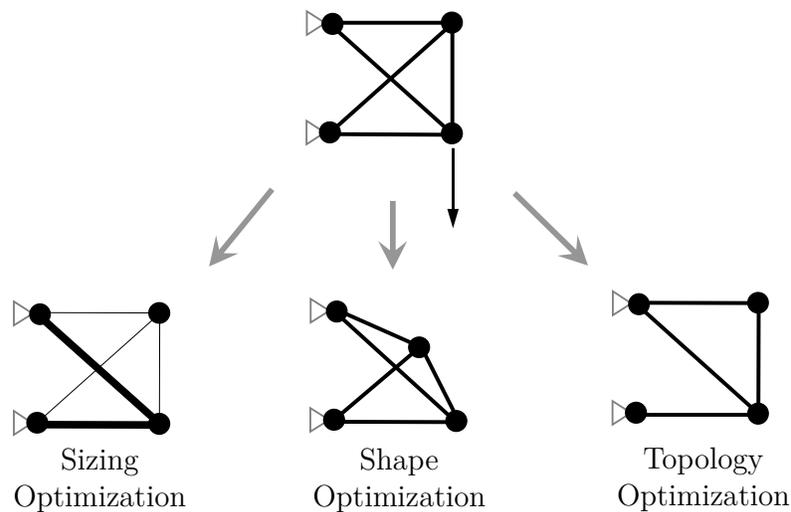


Figure 6.5: Three classes of structural optimization.

Since the number of limbs is predetermined, the limb placement problem can be classified as a shape optimization problem. Each limb will have two joints attached to the platforms, with each set of coordinates for these joints representing three

design variables. Since the boundaries of the design space are determined via appropriate discretization criteria, one of the three design variables per joint can be omitted. Consequently, the number of design variables can be reduced to two per joint, that is, the x and the y components of the joint vector. Therefore, with eight limbs, one will have 32 design variables for the limb placement problem (Fig. 6.6).

For clarity in demonstration, the more simplified structural schematic of the mechanism is adopted instead of the kinematic one. In the structural schematic representation, the load-bearing \mathcal{SPS} limb is represented via a solid line representing the two kinematic links plus the actuated/locked joint and two shaded spheres representing the spherical joints. This representation of the limbs is used throughout the upcoming sections.

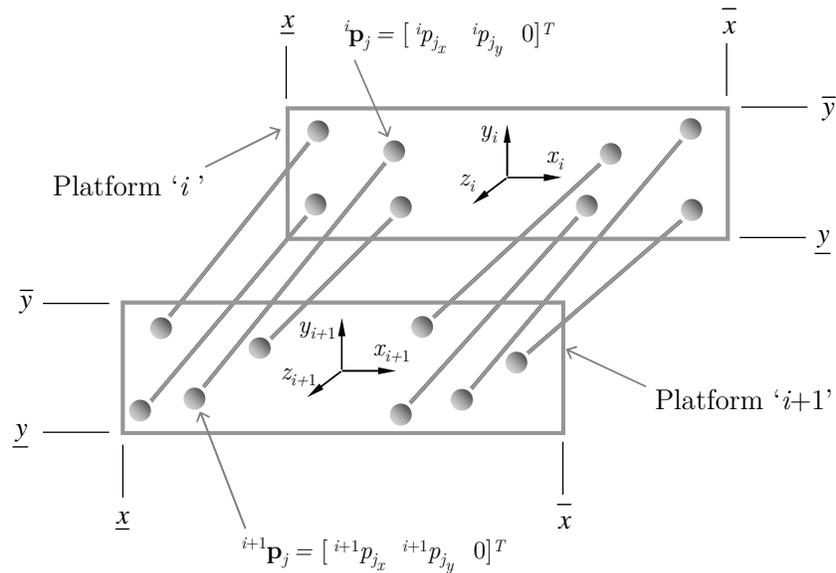


Figure 6.6: Design variables for the shape optimization problem.

Unlike conventional shape optimization problems used in static or geometrical design of conventional mechanisms, the problem at hand cannot be simply solved for a typical objective such as minimum compliance with boundary constraints. In

addition to the complex clashing constraints that need to be introduced into the optimization problem, the impact of the transformation between isostatic and hyperstatic configurations will need to be captured in the optimization problem. The variation in stiffness caused by the transformation between a structure and a mechanism needs to be implemented with the least amount of impact on the aeroelastic characteristics of the wing. Incorporating all of these conditions, objectives, and constraints into a single problem will create a complex problem with potential numerical difficulties. To address this, a multi-stage process has been developed to turn the general shape optimization problem into a series of smaller problems. This is achieved by using a reduced design space, utilized in a more structured problem belonging to the class of topology optimization problems. The proposed procedure starts off by generating a ground topology, which is then used to find the globally optimal hyperstatic configuration for the VGW. The feasible isostatic configurations will then be determined based on a series of design criteria. This leads to the solution for the limb type problem, that is, which limbs should be actuated and which should be passive. Figure 6.7 illustrates the three stages of the optimization process.

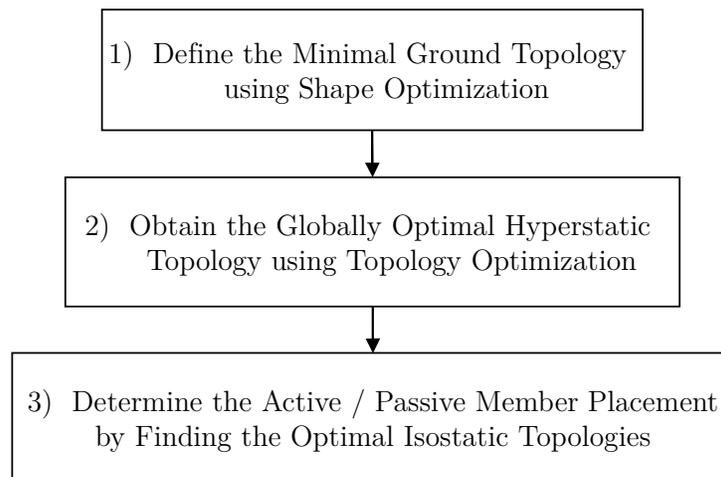


Figure 6.7: Summary of the three-stage optimization process.

6.2.2.1 Minimal Ground Topology (MGT)

When hyperstatic, the optimal design must have the maximum bending, torsional, and shear stiffness. Additionally, it must have adequate stiffness for the three remaining DOFs. Therefore, the ground topology, which would be used to generate such an optimal design, must contain elements that will contribute to the maximum directional stiffness for all six DOFs. It stands to reason that if one could produce the superposition of six configurations, where each configuration has the elements required to maximize a particular component of the Cartesian stiffness, then one would have a ground topology that contains all the necessary elements that may be required to obtain the optimal configuration for the VGW. To obtain such a ground topology, six sub-problems must be generated, where each one aims to maximize the stiffness of one Cartesian DOF. Each sub-problem can be set up as follows:

$$(\mathbb{P})_{MGT} = \left\{ \begin{array}{l} \mathbf{max} \quad {}^i K_{i+1_{mm}} \left({}^i p_{j_x}, {}^i p_{j_y}, {}^{i+1} p_{j_x}, {}^{i+1} p_{j_y} \right) \\ \mathbf{s. t.} \quad \left\{ \begin{array}{l} \underline{x} \leq ({}^i p_{j_x}, {}^{i+1} p_{j_x}) \leq \bar{x} \\ \underline{y} \leq ({}^i p_{j_y}, {}^{i+1} p_{j_y}) \leq \bar{y} \\ \text{sym about } x\text{-}z \text{ plane} \\ \text{sym about } y\text{-}z \text{ plane} \end{array} \right. \\ \forall j = 1, \dots, 8; m = 1, \dots, 6 \end{array} \right. \quad (6.1)$$

where \underline{x} and \bar{x} represent the lower and upper bounds of the design variables, respectively, that is, the boundaries of the platform, in the x direction. Similarly, \underline{y} and \bar{y} represent the same in the y direction. The scalar ${}^i K_{i+1_{mm}}$ is the component of the Cartesian stiffness matrix in the m^{th} row and the m^{th} column. As can be observed from Eq. (6.1), equilibrium constraints or requirements for positive-

definiteness are not included for any of the six sub-problems. Therefore, each of the six configurations may (and most likely will) be singular. The positive-definiteness will be imposed as a constraint only when the optimal configuration is to be determined. Additionally, to simplify the sub-problems, only the diagonal terms of the stiffness matrix are used. The contribution of the off-diagonal terms will be included in the optimization problem in the next stage, where the topology optimization problem is solved. Also, it is assumed that all the members, regardless of their lengths, have the same stiffness. Since the optimal configuration is to be determined in some nominal position, referred to as the home pose, the diagonal stiffness terms can be further simplified to let ${}^i\mathbf{R}_{i+1} = \mathbf{I}$, where $\mathbf{I} \in \mathbb{R}^{3 \times 3}$ and ${}^i b_{i+1_x} = {}^i b_{i+1_y} = 0$. Additionally, since it was established that the joints were placed on the i^{th} and the $(i+1)^{\text{th}}$ platform, one can set ${}^i p_{j_z} = {}^{i+1} p_{j_z} = 0$. The six derived diagonal stiffness terms for the VGW (from Eqs. (5.29) through (5.32)), used in the optimization sub-problems, can then be expressed as:

$${}^i K_{i+1_{11}} = \sum_{j=1}^8 {}^i \beta_j ({}^{i+1} p_{j_x} - {}^i p_{j_x})^2 \quad (6.2)$$

$${}^i K_{i+1_{22}} = \sum_{j=1}^8 {}^i \beta_j ({}^{i+1} p_{j_y} - {}^i p_{j_y})^2 \quad (6.3)$$

$${}^i K_{i+1_{33}} = \sum_{j=1}^8 {}^i \beta_j ({}^i b_{i+1_z})^2 \quad (6.4)$$

$${}^i K_{i+1_{44}} = \sum_{j=1}^8 {}^i \beta_j ({}^{i+1} p_{j_y} {}^i b_{i+1_z})^2 \quad (6.5)$$

$${}^i K_{i+1_{55}} = \sum_{j=1}^8 {}^i \beta_j ({}^{i+1} p_{j_x} {}^i b_{i+1_z})^2 \quad (6.6)$$

$${}^i K_{i+1_{66}} = \sum_{j=1}^8 {}^i \beta_j ({}^{i+1} p_{j_y} {}^i p_{j_x} - {}^{i+1} p_{j_x} {}^i p_{j_y})^2. \quad (6.7)$$

With the inclusion of symmetry, the six sub-problems produce the optimal eight-member configurations illustrated in Fig. 6.8.

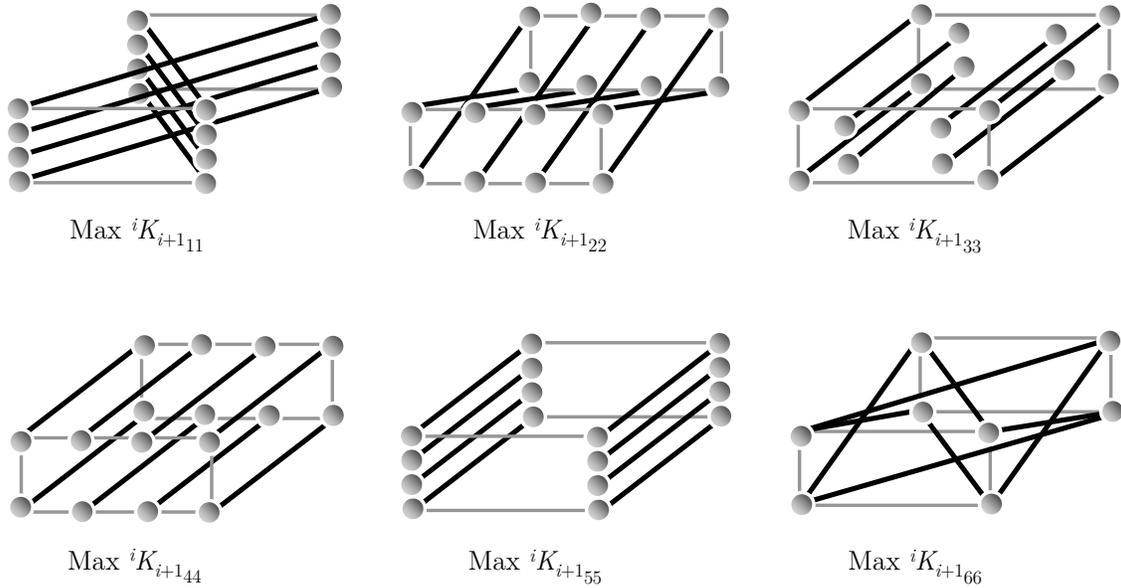


Figure 6.8: Examples of optimal solutions to the single-DOF sub-problems.

The solutions illustrated in Fig. 6.8 only capture some examples of the optimal solutions. There exist many more variations that are equally as optimal and symmetric with respect to the definition of optimality and symmetry expressed via Eq. (6.1). For example, Fig. 6.9 illustrates various forms that have the same ${}^i K_{i+1_{44}}$.

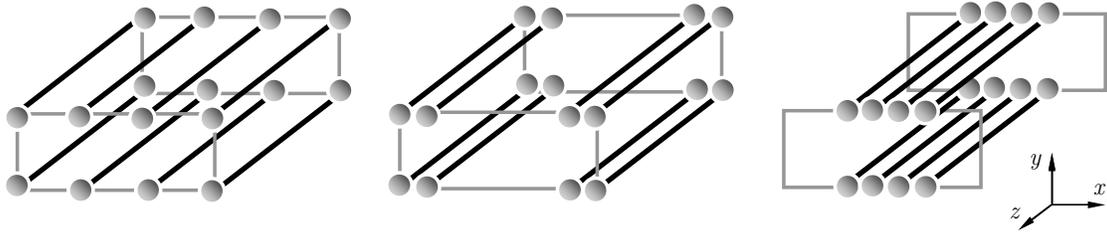


Figure 6.9: Equally optimal configurations with maximum bending stiffness (stiffness about the x axis).

The goal at this point is to develop a ground topology with the minimum number of nodes, and consequently minimal nodal connectivity that can still retain all the elements necessary to provide an optimal solution to the topology optimization problem. All the optimal sets of solutions can be grouped into two geometrically distinct sets: face diagonals, represented by \mathcal{D} , and edges, represented by \mathcal{E} . Set \mathcal{D} will capture the minimal nodal connectivity required for maximizing ${}^iK_{i+1_{11}}$, ${}^iK_{i+1_{22}}$, and ${}^iK_{i+1_{66}}$ while maintaining symmetry. Set \mathcal{E} will do the same for ${}^iK_{i+1_{33}}$, ${}^iK_{i+1_{44}}$, and ${}^iK_{i+1_{55}}$. The union of \mathcal{E} and \mathcal{D} provides the ground topology with the minimal required nodal connectivity. This set is referred to as \mathcal{G} :

$$\mathcal{G} = \mathcal{E} \cup \mathcal{D}. \quad (6.8)$$

Figure 6.10 illustrates \mathcal{G} .

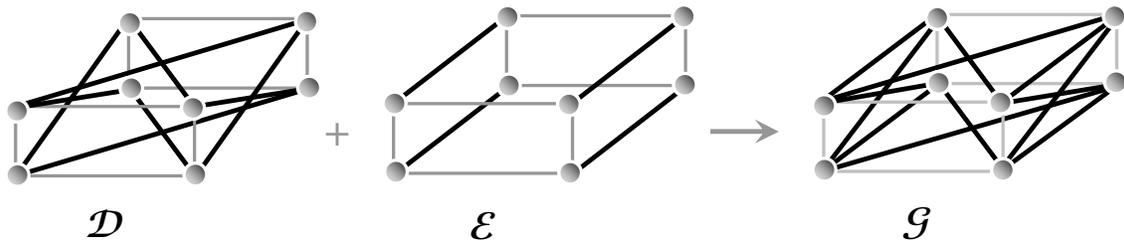


Figure 6.10: Union of the edge and the face diagonal members forming the minimal ground topology.

With this minimal nodal connectivity, any bias can be implemented using topology and sizing optimizations. This 16-member configuration represented by \mathcal{G} is utilized to find the globally optimal hyperstatic topology in the following section.

6.2.2.2 Optimal Hyperstatic Topology (OHT)

Using the proposed ground topology, the topology optimization problem to obtain the most optimal hyperstatic configuration can be posed as follows:

$$(\mathbb{P})_{OHT} = \left\{ \begin{array}{l} \min \sum_{m=1}^6 w_{i\ m} {}^i\mathbf{F}_{i+1\ m}^T {}^i\mathbf{D}_{i+1} \\ \text{s. t.} \left\{ \begin{array}{l} \mathbf{u}^T \mathbf{N}_i \mathbf{u} = 8 \\ \boldsymbol{\xi}_{hyp}^T {}^i\mathbf{K}_{C_{i+1}} \boldsymbol{\xi} > 0 \\ \boldsymbol{\xi}_{fs,e}^T {}^i\mathbf{K}_{C_{i+1}} \boldsymbol{\xi} > 0 \\ \left| {}^i\phi z_{i+1_{F_y}} \right| \approx 0 \\ \left| {}^i\phi z_{i+1_{M_x}} \right| \approx 0 \\ \text{clashing constraints} \end{array} \right. \\ \forall \boldsymbol{\xi} \in \mathbb{R}^6; \boldsymbol{\xi} \neq \mathbf{0}; e = 1, \dots, 8; \mathbf{u} = [1 \ 1 \ 1 \ 1]^T \end{array} \right. \quad (6.9)$$

where

$${}^i\mathbf{D}_{i+1} = \begin{pmatrix} [dx & dy & dz]^T \\ [\phi x & \phi y & \phi z]^T \end{pmatrix}_{i+1} \quad (6.10)$$

Vector ${}^i\mathbf{F}_{i+1}$ represents the unit load acting on the $(i+1)^{\text{th}}$ platform defined in the i^{th} coordinate frame, with m representing the DOF at which the unit load is applied; values of one through six represent translational DOFs along x , y , z , and rotational

DOFs about x , y , z , respectively. The term ${}^i_m\mathbf{D}_{i+1}$ represents the displacement vector of the $(i+1)^{\text{th}}$ platform defined in the i^{th} coordinate frame, corresponding to the load applied in the m^{th} DOF. The term ${}^{i}_{hyp}\phi z_{i+1_{F_y}}$ refers to the rotation of the $(i+1)^{\text{th}}$ platform, when hyperstatic, about the i^{th} z axis, that is, a twist deformation, caused by a shear force (F_y). Similarly, ${}^{i}_{hyp}\phi z_{i+1_{M_x}}$ refers to a twist deformation, caused by a bending moment (M_x). Constraining these deformations diminishes the coupling between bending and torsion for the optimal hyperstatic topology. The term ${}^m\mathbf{w}_i$ represents a weight scalar providing a bias in the order of magnitude for the shear, bending, and torsion DOFs versus the three remaining DOFs. Matrices ${}^{i}_{hyp}\mathbf{K}_{C_{i+1}}$ and ${}^{i}_{fs,e}\mathbf{K}_{C_{i+1}}$ correspond to the stiffness matrix of the hyperstatic and the fail-safe configurations, respectively. The term e represents the configuration of the fail-safe scenario; its implementation will be explained shortly. The term \mathbf{N}_i represents a 4×4 connectivity matrix, capturing the nodal connectivity and the volume fraction of the design. The four rows and columns of \mathbf{N}_i are associated with the four nodes on platform i and platform $(i+1)$, respectively. Its terms take on a value of one if there is connectivity between two nodes of the ground topology, and a value of zero otherwise. The term \mathbf{u} represents a dummy four-dimensional unit vector used to capture the volume fraction.

The proposed minimum compliance objective will yield the stiffest hyperstatic configuration. The positive-definiteness requirement needs to be satisfied for both the optimal eight-member hyperstatic configuration, as well as the seven-member fail-safe configurations. The fail-safe constraint will ensure that the VGW does not lose its structural integrity in the event where any one of the limbs (active or passive) loses its load-bearing capabilities permanently.

The hyperstatic stiffness matrix can be expressed as

$${}_{hyp}^i \mathbf{K}_{C_{i+1}} = \mathbf{J}_i^T {}_{hyp} \mathbf{C}_i \boldsymbol{\kappa}_i \mathbf{J}_i \quad (6.11)$$

where the configuration matrix, ${}_{hyp} \mathbf{C}_i$, as defined in Sec. 5.1, is used for efficiency, so the stiffness matrix of an isostatic configuration or a fail-safe one can be obtained without modifying the terms of the Jacobian matrix (Eq. (5.10)). Similarly, the fail-safe stiffness matrix can be expressed as

$${}_{fs,e}^i \mathbf{K}_{C_{i+1}} = \mathbf{J}_i^T {}_e \mathbf{C}_i \boldsymbol{\kappa}_i \mathbf{J}_i \quad \forall e = 1, \dots, 8 \quad (6.12)$$

where ${}_e \mathbf{C}_i$ is a 8×8 diagonal matrix with all the diagonal members equal to one, except the member in the e^{th} row and the e^{th} column, which is equal to zero. Each fail-safe configuration effectively represents a seven-member design, with the e^{th} member having no contribution to the stiffness matrix. The application of the fail-safe configurations are discussed in Appendix D as a part of the measures taken to design for uncertainties.

Due to the relatively small scale of the problem, the optimization is solved using a direct search and penalization method. The clashing, volume fraction, and non-singularity constraints are implemented using a single objective with penalization formulation. The two solutions that are equally globally optimal are illustrated in Fig. 6.11.

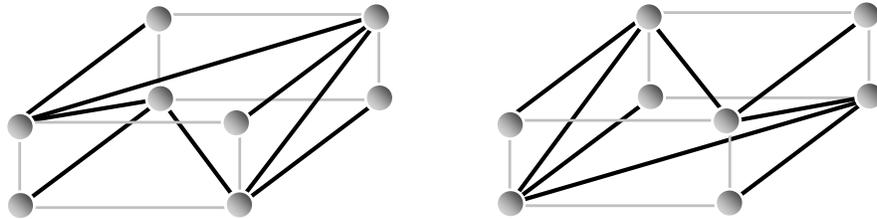


Figure 6.11: Globally optimal hyperstatic topologies.

Without loss of generality, the one on the left (Fig 6.11) is used in the remainder of this work. Fig. 6.12 represents the associated limb numbering.

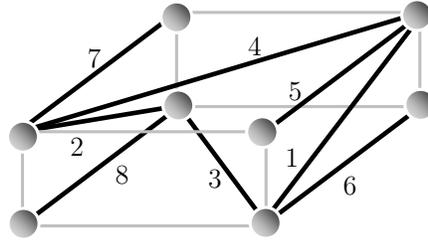


Figure 6.12: Numbering convention for the limbs of the VGW.

6.2.2.3 Optimal Isostatic Topologies

As the VGW switches from a hyperstatic to an isostatic configuration, or between various isostatic configurations, the main stiffness and the coupling terms can change drastically. This can significantly impact the aeroelastic properties of the wing. It would be desirable to tailor the configurations that could be potentially formed to have certain stiffness characteristics to minimize or eliminate this impact. To achieve this one can come up with a series of requirements that can help to minimize the impact of changes in configuration on the aeroelastic properties of the wing, for example, the minimization of bending-torsion coupling. In general, accounting for such requirements for all the possible poses and geometrical irregularities that the wing could have is a difficult task. As it was stated previously, it is assumed that an appropriate number of modules are used for the application to demand only relatively small changes in the geometry of each module. Therefore, some rather generic and regular home pose could be used to implement the general limb placement. This could be seen as a fundamental step to minimize the stiffness sensitivity to changes in configuration through the topological design of the mechanism. Additional

improvements can be implemented through motion control, path planning, and stiffness tailoring via sizing of the members. Prior to presenting these requirements, some set definitions will be introduced to assist with solving the active/passive limb placement problem.

From the given eight-member hyperstatic configuration (Fig. 6.12), one can generate 28 ($= {}_8\mathbf{C}_6$) six-member configurations, not all of which may be isostatic. The elements of each of these configurations or sets can take on non-repeating values between one and eight. The inclusion of a number in the set signifies the existence of a load-bearing limb, while the exclusion of a number identifies the corresponding limb as an unlocked passive member. For example, $\mathbf{s}_1 = \{1, 2, 3, 4, 5, 6\}$ represents the configurations shown in Fig. 6.13, where the solid lines represent load-bearing limbs, that is, actuated or locked passive, and the dashed lines represent unlocked passive limbs.

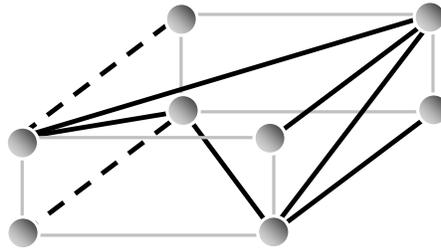


Figure 6.13: Six-member configuration corresponding to set \mathbf{s}_1 .

Each one of these 28 sets are members of a larger set, \mathcal{S} , containing all the possible six-element subsets of \mathcal{H} , denoted as

$$\mathcal{S} = \binom{\mathcal{H}}{6} = \{\mathbf{s}_1, \dots, \mathbf{s}_{28}\} \quad (6.13)$$

where $\mathcal{H} = \{1, \dots, 8\}$ represents the eight-member hyperstatic configuration with eight elements. Similarly, the inclusion of a number in this set signifies the existence of a corresponding load-bearing limb.

There exists a subset of \mathcal{S} , referred to as the feasible set, represented by \mathcal{R} , whose elements are made up of those sets of \mathcal{S} that satisfy the aforementioned requirements, which will be formally introduced shortly. In order to solve the limb placement problem, one must initially find the elements of \mathcal{R} , denoted by r_k . The common elements among the members that make up \mathcal{R} represent the elements that must be load-bearing in all configurations, or in general they must be actuated limbs. These members belong to a set represented by \mathcal{A} , expressed as follows:

$$\mathcal{A} = \bigcap_{r_k \in \mathcal{R}} r_k. \quad (6.14)$$

Similarly, one can store the members that are passive into a set represented by \mathcal{P} . Since by definition a limb can either be actuated or passive, sets \mathcal{A} and \mathcal{P} are mutually exclusive. Also,

$$\mathcal{H} = \mathcal{A} \cup \mathcal{P}. \quad (6.15)$$

Therefore, \mathcal{P} is the complement of \mathcal{A} in \mathcal{H} , expressed as

$$\mathcal{P} = \mathcal{H} \setminus \mathcal{A}. \quad (6.16)$$

Thus, the solution to the limb placement problem can be obtained by determining the members of \mathcal{R} , based on the desired requirements, which are as follows:

1) Any member of \mathcal{R} must have a positive-definite stiffness matrix. The subset of solutions that satisfy this requirement will belong to a set represented by \mathcal{R}_1 . This requirement can be expressed as

$$\boldsymbol{\xi}^T {}^i\mathbf{K}_{C_{i+1}} \boldsymbol{\xi} > 0 \quad \forall \boldsymbol{\xi} \in \mathbb{R}^6; \boldsymbol{\xi} \neq \mathbf{0}; t = 1, \dots, |\mathcal{S}| \quad (6.17)$$

where $|\mathcal{S}|$ represents the cardinality of \mathcal{S} . Matrix ${}^i\mathbf{K}_{C_{i+1}}$ represents the Cartesian stiffness of the t^{th} member of \mathcal{S} , and can be expressed as

$${}^i\mathbf{K}_{C_{i+1}} = \mathbf{J}_i^T {}_t\mathbf{C}_i \mathbf{k}_i \mathbf{J}_i \quad \forall t = 1, \dots, |\mathcal{S}| \quad (6.18)$$

where ${}_t\mathbf{C}_i$ is the configuration matrix corresponding to the t^{th} member of \mathcal{S} .

2) Any member of \mathcal{R} must maintain a relatively high degree of torsional stiffness. The subset of solutions that satisfy this requirement will belong to a set represented by \mathcal{R}_2 . This requirement can be expressed as

$$\mathbf{min} \left(\left| {}_{hyp}^i\phi z_{i+1M_z} \right| - \left| {}_t^i\phi z_{i+1M_z} \right| \right) \quad \forall t = 1, \dots, |\mathcal{S}| \quad (6.19)$$

where ${}_{hyp}^i\phi z_{i+1M_z}$ and ${}_t^i\phi z_{i+1M_z}$ represent a twist (rotation of the $(i+1)^{\text{th}}$ platform about the i^{th} z axis) due to a torsional moment (M_z) corresponding to the hyperstatic and the t^{th} configurations, respectively. To implement this criterion, a suitable relative threshold is used to identify those configurations with minimal change in the torsional stiffness relative to the hyperstatic configuration.

3) Any member of \mathcal{R} must possess minimal bending-torsion coupling. The subset of solutions that satisfy this requirement will belong to a set represented by \mathcal{R}_3 . This requirement can be expressed as

$$\mathbf{min} \left(\left| {}_{hyp}^i \phi z_{i+1_{M_x}} \right| - \left| {}_t^i \phi z_{i+1_{M_x}} \right| \right) \quad \forall t = 1, \dots, |\mathcal{S}| \quad (6.20)$$

where ${}_{hyp}^i \phi z_{i+1_{M_x}}$ and ${}_t^i \phi z_{i+1_{M_x}}$ represent a twist due to a bending moment (M_x) corresponding to the hyperstatic and the t^{th} configurations, respectively. Similarly, a suitable relative threshold is used to identify those configurations with minimal coupling relative to the hyperstatic configuration.

Set \mathcal{R} can then be simply defined as the set whose members satisfy all the above requirements:

$$\mathcal{R} = \mathcal{R}_1 \cap \mathcal{R}_2 \cap \mathcal{R}_3. \quad (6.21)$$

After applying the filters to all 28 sets, only four remain as feasible sets:

$$\mathcal{R} = \{\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \mathbf{r}_4\} \quad (6.22)$$

where

$$\mathbf{r}_1 = \{1, 2, 3, 4, 5, 6\}$$

$$\mathbf{r}_2 = \{1, 2, 3, 4, 5, 7\}$$

$$\mathbf{r}_3 = \{1, 2, 3, 4, 6, 8\}$$

$$\mathbf{r}_4 = \{1, 2, 3, 4, 7, 8\}.$$

These sets, which represent the feasible isostatic configurations, are illustrated in Fig. 6.14.

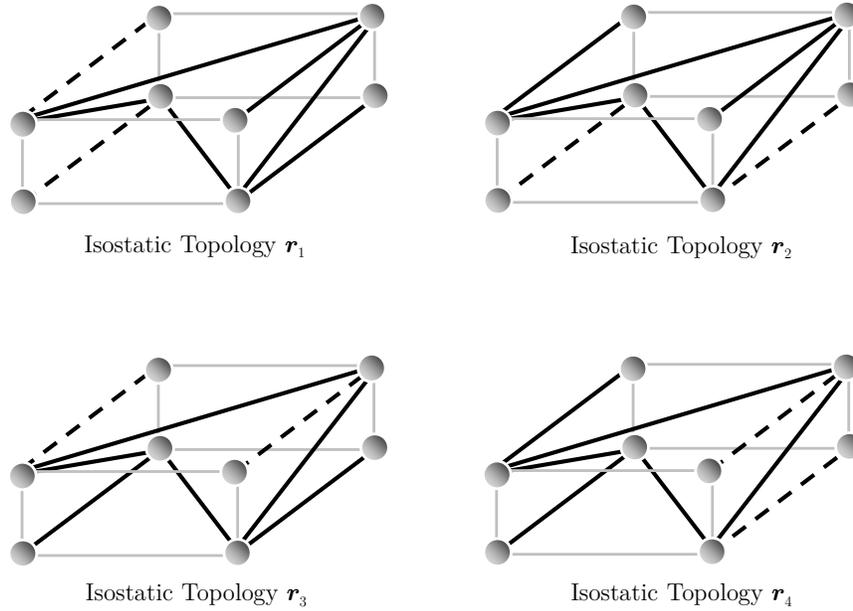


Figure 6.14: Four optimal isostatic topologies used for under-actuation.

The four elements of \mathcal{R} will be used to realize under-actuation, as will be demonstrated in Chapter 7. By applying Eqs. (6.14) and (6.16) the elements of \mathcal{A} and \mathcal{P} can be determined as follows:

$$\mathcal{A} = \{1, 2, 3, 4\}$$

$$\mathcal{P} = \{5, 6, 7, 8\}.$$

All three stages of the optimization are thereby complete.

6.3 Practical Realization

Using the proposed methodology, the optimal configuration design for the VGW has been determined, as illustrated by Fig. 6.15.

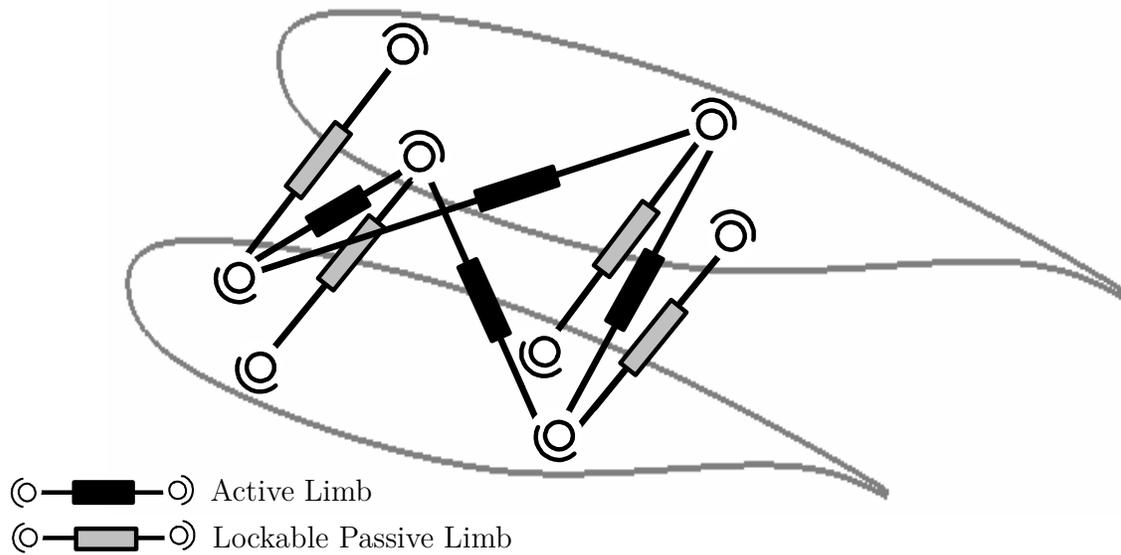


Figure 6.15: Schematic representation of the optimal configuration.

This configuration has the common \mathcal{SPS} architecture for all its limbs, four of which are actuated and the other four are passive and lockable. Depending on the actuation setup, mechanical, hydraulic, or SMA means can be used to drive the mechanism. Figure 6.16 illustrates the CAD realization of the optimal design using a hydraulic setup. In this setup, hydraulic cylinders are used for the active and passive members, and spherical plain bearings with high misalignment are used for the pin joints.

The active/passive limb placement determined by the requirements defined in Sec. 6.2.2.3 can be intuitively appreciated. As can be observed from Fig. 6.15, all four resulting isostatic topologies have the face diagonal members present. These members are the main contributors to the torsional stiffness of the VGW as demonstrated in Sec. 6.2.2.1. Also, it can be seen that in all four optimal isostatic configurations the edge members maintain symmetry about the x - z or the y - z plane. The presence of the diagonal members at all times and the bending symmetry result

in minimal change in the torsional stiffness and minimal bending-torsion coupling of the VGW as it switches between various topologies. This would result in minimal aeroelastic impact during the actuation process.

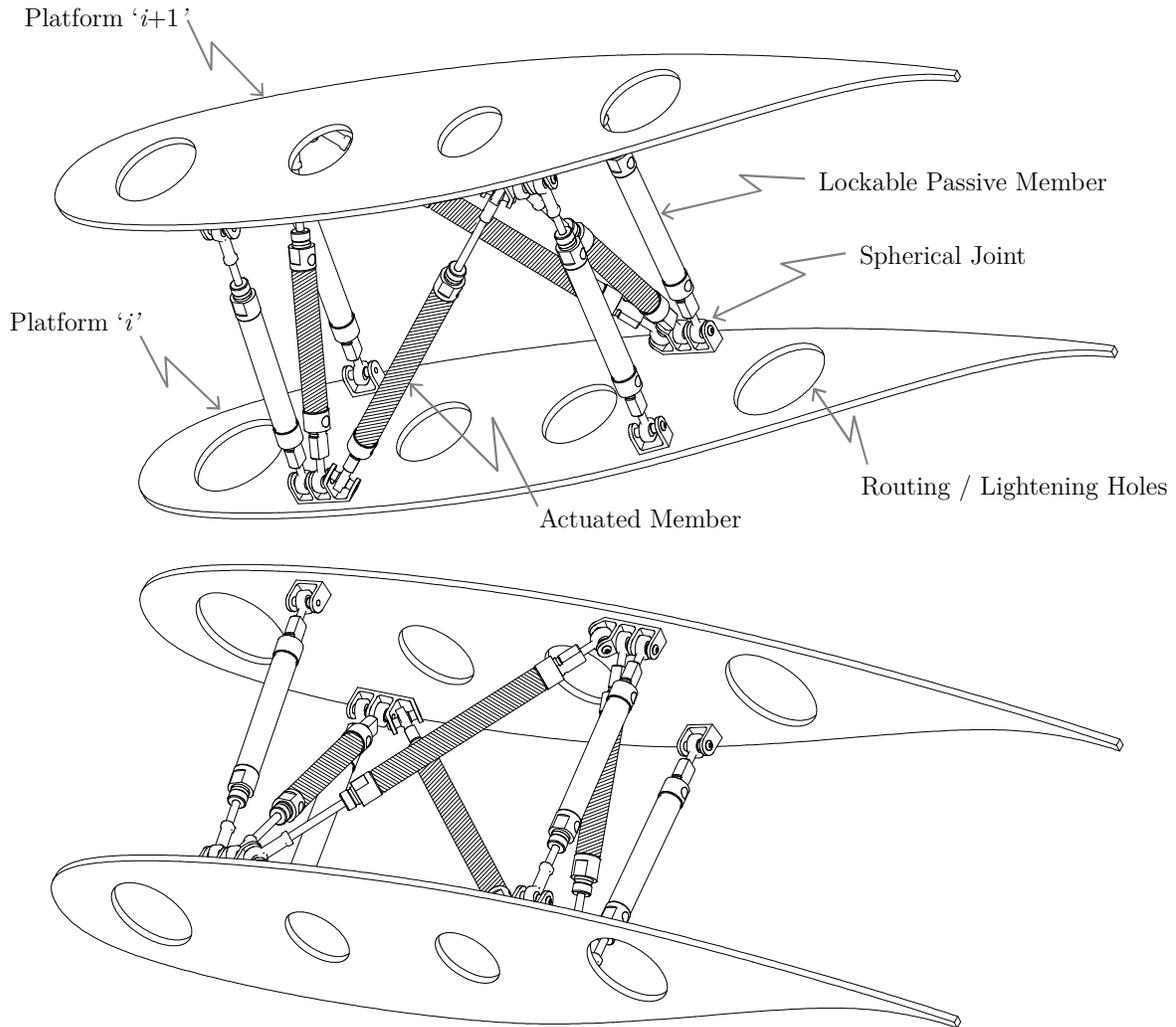


Figure 6.16: CAD representation of the optimal configuration for the VGW (prototype).

This particular placement also maximizes the number of isostatic configurations, which will be used for the motion control as will be discussed in Chapter 7. The

proposed mechanism is well-suited for modular applications, as illustrated in Fig. 1.2.

6.4 Rigidity Validation

To validate the rigidity of the proposed optimal configuration in various modes, that is, hyperstatic, isostatic, and fail-safe, loading tests were performed on a two-module prototype. The prototype, built for the validation of the rigidity and motion control (as will be discussed in Chapter 7) is illustrated by Fig. 6.17.

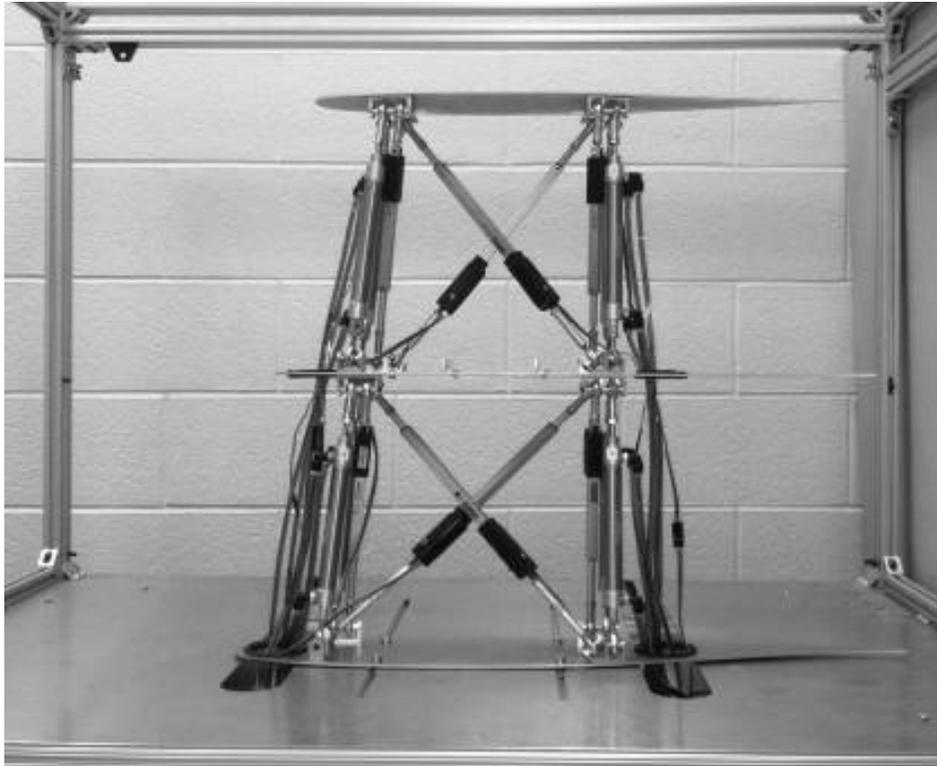


Figure 6.17: Two-module prototype built for validation.

The actuated limbs were set up using electromechanical linear actuators, while the passive limbs were set up using hydraulic cylinders and active/passive flow control valves, as discussed in Appendix D. A pulley system was used to apply the

loads to the platforms. Figure 6.18 illustrates the prototype undergoing simultaneous shear, bending, and torsion loads, while simulating a twist motion.

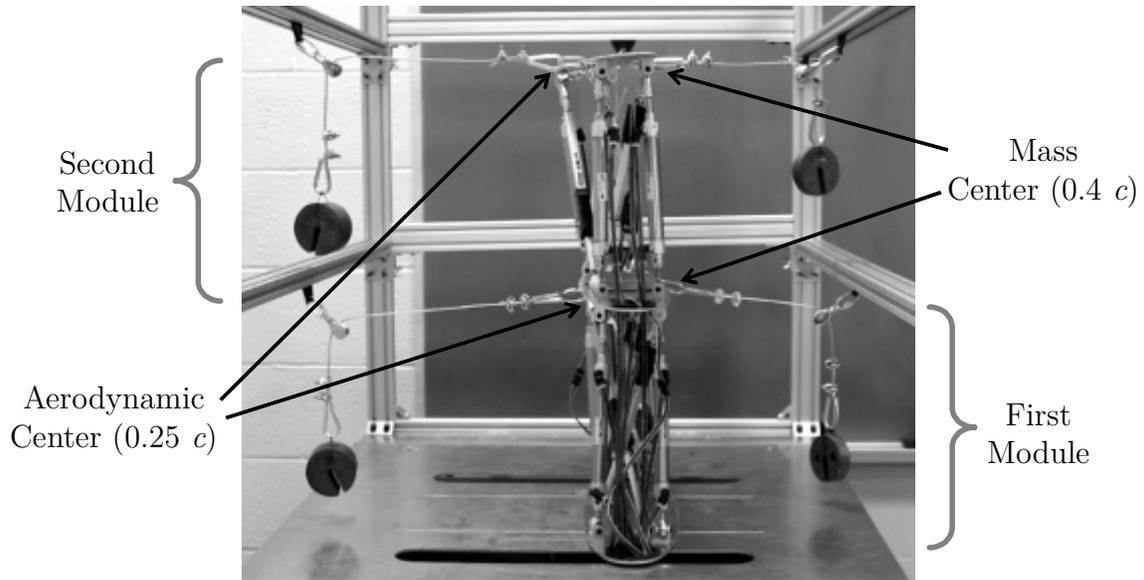


Figure 6.18: Rigidity validation for the optimal configuration (view looking aft on the prototype).

The shear forces acting at the platforms are results of the difference in magnitude between the loads applied at the aerodynamic center and the mass center of each platform, simulating lift and inertia, respectively. The torsional moments acting at the platforms are results of the offset between the load application points, that is, $0.25 c$ (chord length) and $0.4 c$ at each platform. The bending moment acting on the moving platform of the first module is due to the resultant shear force acting on the moving platform of the second module. The two-module setup allows for the generation of this bending moment. While isostatic and undergoing the simulated loading, the prototype was able to generate controlled motion in all six DOFs without encountering any singularities for all four isostatic topologies. Additionally, the

prototype was able to maintain rigidity under the simulated loading without encountering any singularities for the eight-member hyperstatic and the seven-member fail-safe topologies. These loading tests validated the rigidity of the proposed optimal configuration for the VGW.

6.5 Alternative Configuration

Depending on the application, some of the actuators can be replaced with non-extendable linear members. For example, in the case of a wing tip morphing, no considerable span increase may be required, and some degree of kinematic coupling between the translational and rotational DOFs may be acceptable. As a result one may replace two of the actuators with solid members to have the configuration shown in Fig. 6.19.

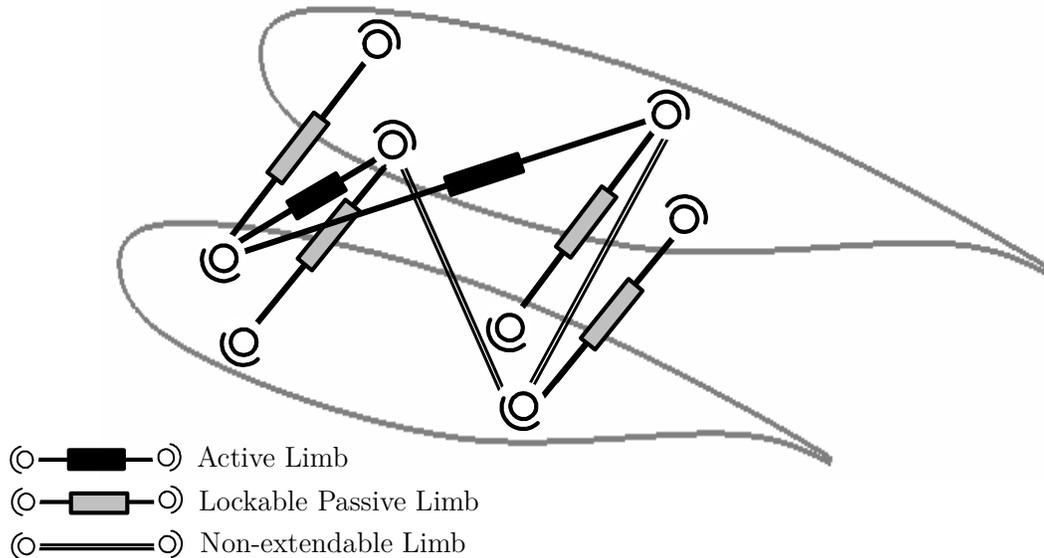


Figure 6.19: Schematic representation of the optimal configuration with two actuators and lower mobility.

The configuration shown in Fig. 6.19 can offer well coupled translational-rotational cant and sweep motions, as well as a minimally coupled twist motion, with

only two actuators, while maintaining all the optimal characteristics of the fully variable VGW (Fig. 6.15).

6.6 Conclusion

The optimal configuration design for the VGW was presented in this chapter. The design problem at hand was broken up into two components: 1) the optimal limb configuration, and 2) the optimal topological configuration. Initially, it was shown that the family of \mathcal{SPS} limbs, including \mathcal{UPS} , \mathcal{UKS} , \mathcal{SPU} , \mathcal{SKU} , \mathcal{UCU} , \mathcal{SPS} , \mathcal{SKS} , \mathcal{SCU} , \mathcal{UCS} , and \mathcal{SCS} , are all optimal solutions for the limb configuration of the VGW. Such architectures meet the kinematic and the compactness requirements while offering good static characteristics such as the elimination of bending at the joints. Additionally, it was shown through a multi-stage compliance-based optimization process that the optimal topological configuration is one with permanently load-bearing members placed on the face diagonals of the cuboid design space and lockable passive members placed along the edges. This configuration has a total of eight limbs and is fail-safe (in the event that any one of the eight limbs is permanently lost), if sized properly. Furthermore, the rigidity of the proposed configuration was validated using a multi-module prototype undergoing simultaneous shear, bending, and torsion loads. As a part of the validation, the prototype was reconfigured to test the rigidity of the eight-member hyperstatic topology, the six-member isostatic topologies, and the seven-member fail-safe topologies.

The geometrical and loading constraints associated with a wing have served as the inspirations for the development of the methodology presented in this chapter, specific to a variable geometry wing-box design application. However, the proposed compliance-based optimization formulation for the under-actuated system can be applied to any comparable mechanism.

It should be recalled that the proposed optimal configuration is to be treated as a starting framework. Further sizing optimization of the limbs and perhaps some additional local shape optimization for the joint locations may be required to tailor the design to a particular application and to account for geometrical irregularities in a highly swept wing. It should also be noted that additional measures need to be taken when dealing with less conventional shapes. For example, if cant angles of 90 degrees or larger are to be realized using multiple modules, other forms of couplings such as torsion-axial (axial being aligned with the base wing span) coupling may become significant when switching between isostatic topologies. These effects need to be accounted for in the detailed design and sizing of the VGW.

Chapter 7

Optimal Motion Planning

In this chapter, the criteria for optimality of the motion and the sequences that the VGW will go through to get from some initial pose to some final one will be presented. The proposed optimal actuation scheme is based on the kinematic and static relationships of the system which were presented in Chapters 4 and 5, respectively. A case-study for a wing tip morphing application will be presented at the end of this chapter to show the effectiveness of the proposed formulation.

7.1 Control Implementation Scheme

Given that the proposed application is morphing for performance improvement, as opposed to control, one can specify the instances when actuation is to take place. This implementation can yield significant weight reductions. For a typical mission profile, as shown in Fig. 7.1, one can pre-determine when to actuate the mechanism so that the actuators experience minimal external loads.

In Fig. 7.1, six predetermined zones for morphing are indicated on the profile using dashed circles. To minimize the required actuation loads, subsequently leading to the minimization of the weight of the actuating modules, morphing will only take place during steady level flight or just before take-off or after landing. A similar implementation could be carried out for turn maneuvers so that actuation takes place

just before the turn. This implementation means that the members and the joints do not need to be structurally over-sized to handle very large external loads in the isostatic/actuating configurations. The members can be sized to provide the stiffness and the actuation forces required for the actuating configurations while offering the stiffness and the holding forces required for the non-actuating configurations. Using this approach all eight members can be sized to provide adequate stiffness and load-bearing capabilities without over-designing the mechanism. When the wing is experiencing large aerodynamic and inertia loads due to a balanced maneuver or a gust, the eight-member hyperstatic configuration will provide the necessary stiffness and load-bearing capabilities, and when morphing is to take place during the pre-determined flight regimes, the isostatic configurations will be utilized.

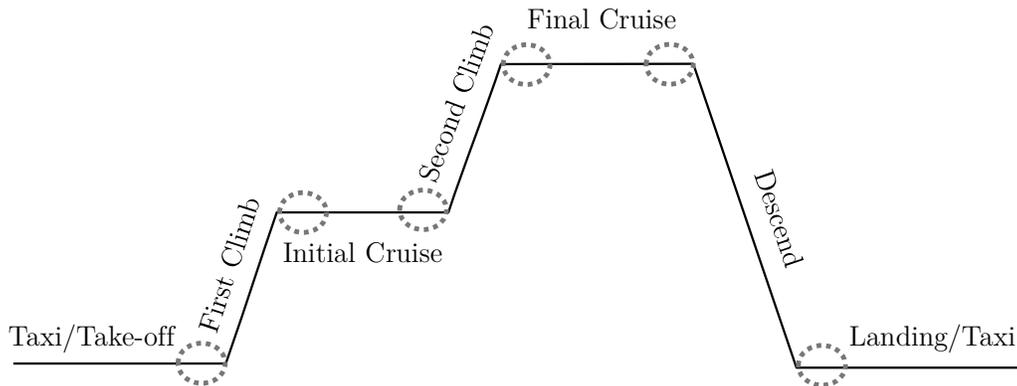


Figure 7.1: Typical mission profile for a commercial aircraft.

If a combination of actuators and passive members that could be locked and unlocked at appropriate times and sequences were utilized, one could still have a fully variable mechanism that is capable of serving a dual role as a structure and a mechanism. It was determined in Chapter 6 that for the optimal configuration, the four diagonal members should be actuators, whereas the four edge members should

be lockable passive members. This particular configuration offers four isostatic topologies that will be used in the under-actuated control scheme.

The passive members are axial members that are attached to the platforms using spherical joints, similar to the actuators, except they are not actuated. However, they can provide axial stiffness when required. This is accomplished by a lock/unlock feature. When locked, a passive member will be load-bearing; when unlocked, it will have no stiffness. In practice, this could be realized using a clutch in the case of a mechanical setup or a valve in the case of a hydraulic one. In addition to the lock/unlock feature, the passive members can be directionally locked. This is useful for improved stiffness when encountering uncertainties in loading, such as a high-velocity gust which may reverse the anticipated loading. See Appendix D for the implementation of the directional lockability.

7.2 Actuation Paths

For the proposed VGW mechanism, a minimum of three isostatic topologies would be required to achieve under-actuation, as discussed in Sec. 3.5. Each isostatic topology will be utilized to achieve the final length of at least one passive member. In total there will be three actuation stages to go from an initial pose to a final one while controlling all the DOFs, where each stage corresponds to a change in the pose. With the given ground topology (Fig. 6.15) one can form four isostatic topologies, as shown in Fig. 7.2.

Generally, one can start with any given isostatic topology in the first stage, which will have four actuators and two locked passive members. The two locked passive members are needed to provide the necessary constraints to attain isostaticity. In addition, a third constraint needs to be imposed which is the desired final length of

one of the two unlocked passive members. This yields a total of three constraints. Any intermediate pose that satisfies these constraints would be a possible solution. Once this final length is obtained, the corresponding member, i.e. the one imposing the third constraint, will be locked and remains locked for the remainder of the actuation. This locked member will be one of the two locked members used for the second stage. The same procedure will be repeated to obtain the final length of another passive member. At this point, there are two locked passive members that have their final lengths realized. In the third and the final stage of the actuation with these two locked passive members and the four actuators the final desired lengths of the two remaining passive members, and subsequently the final pose, can be realized.

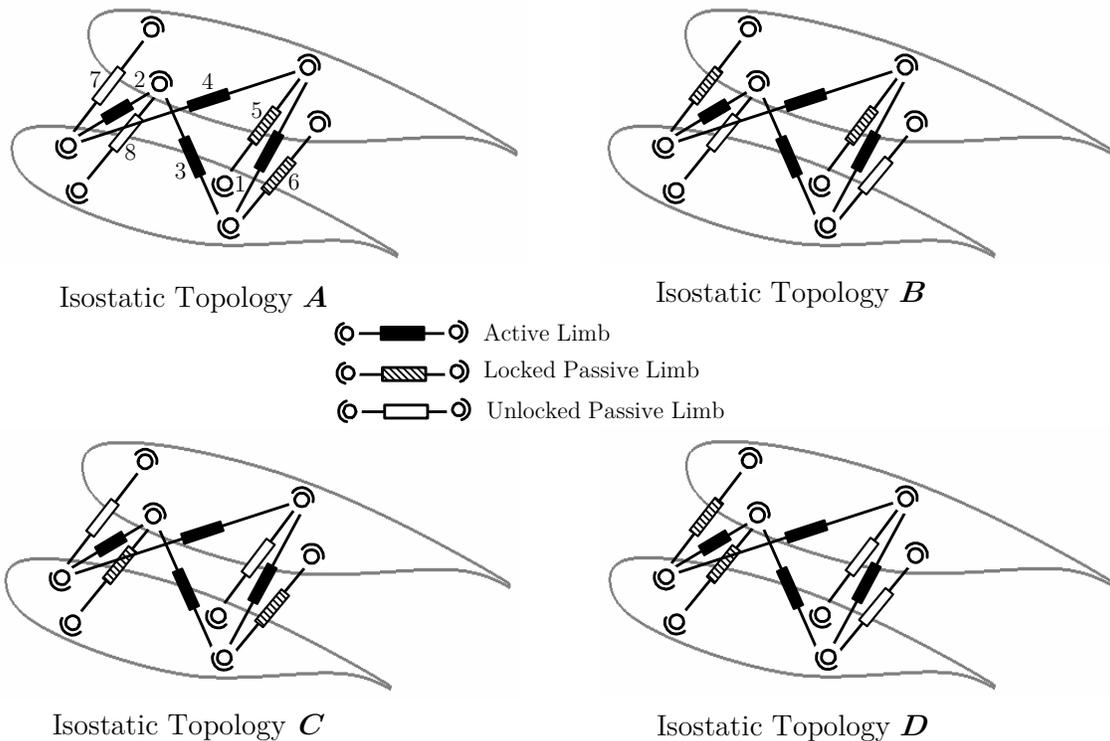


Figure 7.2: Four isostatic topologies used for actuation.

It should be noted that there are special cases where certain poses may be achieved with only two or even one stage(s). However, in general three stages will cover any reachable motion in the workspace of the mechanism. In addition, only three of the four isostatic topologies are required in the under-actuation. This provides the opportunity for optimization. By taking into account all the possible combinations and sequences, one can have 16 possible paths to get from an initial pose (${}^i_0\mathbf{X}_{i+1}$) to a final one (${}^i_3\mathbf{X}_{i+1}$), where the left subscript notation refers to the stage of the actuation. Subscript 0 refers to the initial pose prior to the start of the actuation; subscripts 1 and 2 refer to the two intermediate poses; subscript 3 refers to the final pose. All 16 paths are summarized in Table 7.1.

Table 7.1: 16 Actuation sequences used for optimal motion control.

Path	Stage 1		Stage 2		Stage 3	
1	Use A	to get ${}_3q_7$	Use B	to get ${}_3q_8$	Use D	to get ${}_3q_5$ and ${}_3q_6$
2	Use A	to get ${}_3q_7$	Use D	to get ${}_3q_5$	Use B	to get ${}_3q_6$ and ${}_3q_8$
3	Use A	to get ${}_3q_8$	Use D	to get ${}_3q_6$	Use C	to get ${}_3q_5$ and ${}_3q_7$
4	Use A	to get ${}_3q_8$	Use C	to get ${}_3q_7$	Use D	to get ${}_3q_5$ and ${}_3q_6$
5	Use B	to get ${}_3q_6$	Use A	to get ${}_3q_8$	Use C	to get ${}_3q_5$ and ${}_3q_7$
6	Use B	to get ${}_3q_6$	Use C	to get ${}_3q_5$	Use A	to get ${}_3q_7$ and ${}_3q_8$
7	Use B	to get ${}_3q_8$	Use C	to get ${}_3q_7$	Use D	to get ${}_3q_5$ and ${}_3q_6$
8	Use B	to get ${}_3q_8$	Use D	to get ${}_3q_6$	Use C	to get ${}_3q_5$ and ${}_3q_7$
9	Use C	to get ${}_3q_5$	Use A	to get ${}_3q_7$	Use B	to get ${}_3q_6$ and ${}_3q_8$
10	Use C	to get ${}_3q_5$	Use B	to get ${}_3q_6$	Use A	to get ${}_3q_7$ and ${}_3q_8$
11	Use C	to get ${}_3q_7$	Use B	to get ${}_3q_8$	Use D	to get ${}_3q_5$ and ${}_3q_6$
12	Use C	to get ${}_3q_7$	Use D	to get ${}_3q_5$	Use B	to get ${}_3q_6$ and ${}_3q_8$
13	Use D	to get ${}_3q_5$	Use B	to get ${}_3q_6$	Use A	to get ${}_3q_7$ and ${}_3q_8$
14	Use D	to get ${}_3q_5$	Use A	to get ${}_3q_7$	Use B	to get ${}_3q_6$ and ${}_3q_8$
15	Use D	to get ${}_3q_6$	Use A	to get ${}_3q_8$	Use C	to get ${}_3q_5$ and ${}_3q_7$
16	Use D	to get ${}_3q_6$	Use C	to get ${}_3q_5$	Use A	to get ${}_3q_7$ and ${}_3q_8$

Similarly, the left subscript notation for q refers to the stage of the actuation. For example ${}_3q_7$ refers to the length of the seventh member in the third and the final pose. Figure 7.2 illustrates the limb numbering convention. Each path presents a complete motion summary of how to get from some initial pose to some final pose. For example if one uses path 1, isostatic topology \mathbf{A} must be initially used to get the final desired length for passive member 7, that is, ${}_3q_7$. Then, while keeping ${}_3q_7$ locked for the remainder of the actuation, isostatic topology \mathbf{B} must be used to get ${}_3q_8$. With ${}_3q_7$ and ${}_3q_8$ locked one can only use isostatic topology \mathbf{D} to obtain ${}_3q_5$ and ${}_3q_6$. During the last stage, the actuators will also achieve their desired final lengths. With all the passive members and the actuators at their final desired lengths, the final pose is simply realized. At this point, it can be seen that the higher the number of isostatic topologies, the higher the number of combinations and potential solutions will be. Although not guaranteed, the higher the number of solutions, the higher the chances of obtaining the most optimal motion will be.

It should also be noted that 16 is the minimum number of potential paths. This number could be increased by introducing more mid-poses, thus increasing the number of stages. If implemented at a relatively high frequency, the under-actuation will appear seamless in practice.

7.3 Minimum Energy Actuation

Various objectives could be used to determine the most optimal sequence, including the minimization of the overall actuation energy, the minimization of the maximum actuator force, or the maximization of stiffness for improved stability. Here, the minimum actuation energy is used as the objective. The *actuation energy* of an actuator can be defined as its capacity to do work, which is equal to the product of the required actuation force and the stroke length.

Determining the intermediate poses that satisfy the three constraints is required before obtaining the stroke lengths. As previously explained, there will generally be three equality constraints corresponding to the stroke lengths of the two locked passive members required to maintain isostaticity and the final desired stroke length of the unlocked passive member. This general form will be maintained for all the actuation stages. However, it should be noted that when going from the second last to the last stage, the third equality constraint will disappear, since based on the inverse kinematics, the final pose is guaranteed to give the desired final lengths for the two remaining unlocked passive members.

With three constraint equations and six variables to solve, one has the freedom to add more constraints to the set of equations for finding the intermediate poses. These additional constraints can be imposed in the form of equality, inequality, or side constraints. In general, the constraints can be classified into three groups: actuation, pose, and stroke constraints. The actuation constraints, which are of the equality form, are required for realizing the under-actuation scheme. These constraints cannot be removed or changed as they are needed to obtain the final pose. The pose constraints are imposed to limit certain motions of the wing. They could be driven by a variety of criteria such as stability or stiffness, and are more flexible than the actuation constraints. Without loss of generality, for the proposed formulation this constraint is imposed on the twist motion of the wing to limit it to the range between the initial and the final pose of each module. The last group contains the stroke constraints. These side constraints are imposed to account for the physical limits on the stroke lengths of the actuators and the passive members. Equation (7.1) summarizes the optimization problem for minimum energy actuation:

$$\left\{ \begin{array}{l} \mathbf{min} \text{ total actuation energy: } \mathcal{E}_T = \sum_{i=1}^a \sum_{j=1}^4 \sum_{s=1}^3 ({}^i_s \mathcal{E}_j) \\ \mathbf{s. t.} \left\{ \begin{array}{l} \text{actuation constraints: 3 per stage per module} \\ \text{pose constraints: } {}^i_0 \alpha_{i+1} \leq {}^i_s \alpha_{i+1} \leq {}^i_3 \alpha_{i+1} \\ \text{stroke constraints: } {}^i q_{jL} \leq {}^i q_j \leq {}^i q_{jU} \end{array} \right. \end{array} \right. \quad (7.1)$$

where i , j , and s refer to the number of the module, the limb, and the actuation stage, respectively; a represents the total number of modules considered. The actuation energy refers to the work done by a force produced by the actuator to move it a certain distance. This can be defined through the movement of a point on the j^{th} member of the i^{th} module, along a trajectory ${}^i Q_j$, with a velocity of ${}^i \dot{\mathbf{q}}_j$ defined in the i^{th} coordinate system, not to be confused with $\dot{\mathbf{q}}_i$, that is, the vector of joint rates associated with module i , expressed in the joint space. The infinitesimal amount of work, $\delta {}^i \mathcal{W}_j$, that occurs over an instant of time, δt , can be expressed as

$$\delta {}^i \mathcal{W}_j = {}^i \mathbf{f}_j^{*T}(t) {}^i \dot{\mathbf{q}}_j \delta t \quad (7.2)$$

where ${}^i \mathbf{f}_j^*(t)$ is the force for the j^{th} member of the i^{th} module as a function of time defined in the i^{th} coordinate system, not to be confused with \mathbf{f}_i , that is, the vector of internal loads associated with module i , expressed in the joint space. The summation of these small amounts of work over trajectory ${}^i Q_j$, representing the movement of some point of interest on the actuator between stages $(s-1)$ and s , yields

$${}^i_s \mathcal{W}_j = \int_{t_{s-1}}^{t_s} {}^i \mathbf{f}_j^{*T}(t) \frac{d {}^i \mathbf{q}_j}{dt} dt = \int_{{}^i_s Q_j} {}^i \mathbf{f}_j^T ({}^i \mathbf{q}_j) d {}^i \mathbf{q}_j \quad (7.3)$$

where ${}^i_s\mathcal{W}_j$ represents the work done by the j^{th} member of the i^{th} module between stages $(s-1)$ and s , while t_s and t_{s-1} are the times corresponding to stages s and $(s-1)$, respectively. Furthermore, the trajectory iQ_j is always aligned with the actuator line of motion, that is, the joint coordinate. Therefore, one can simplify the integration over trajectory iQ_j by using the stroke lengths. Subsequently, ${}^i\mathbf{f}_j^*(t)$ can be transformed to ${}^i\mathbf{f}_j({}^iq_j)$:

$${}^i\mathbf{f}_j^*(t_s) = {}^i\mathbf{f}_j({}^iq_j) \quad (7.4a)$$

$${}^i\mathbf{f}_j^*(t_{s-1}) = {}^i\mathbf{f}_j({}_{s-1}{}^iq_j). \quad (7.4b)$$

Similarly, the force vector can be replaced by a scalar, if_j , representing that of the j^{th} member of the i^{th} module, leading to the following simplification for Eq. (7.3):

$${}^i_s\mathcal{W}_j = \int_{{}_{s-1}{}^iq_j}^{{}^iq_j} {}^if_j({}^iq_j) d {}^iq_j. \quad (7.5)$$

Using a trapezoidal approximation, the above integral can be expressed as

$${}^i_s\mathcal{W}_j = {}^if_{j,avg} \Delta {}^iq_j \quad (7.6)$$

where

$$\Delta {}^iq_j = {}^iq_j - {}_{s-1}{}^iq_j \quad (7.7)$$

$${}^i f_{s j_{avg}} = \frac{{}^{i-1} f_j + {}^i f_j}{2} \quad (7.8)$$

where ${}^i f_{s j_{avg}}$ is the average force of the j^{th} member of the i^{th} module while going from the $(s-1)^{\text{th}}$ to the s^{th} actuation stage; similarly, $\Delta_s^i q_j$ represents the change in the stroke length for that member between the two stages. Figure 7.3 schematically illustrates the forces and strokes associated with the two actuation stages.

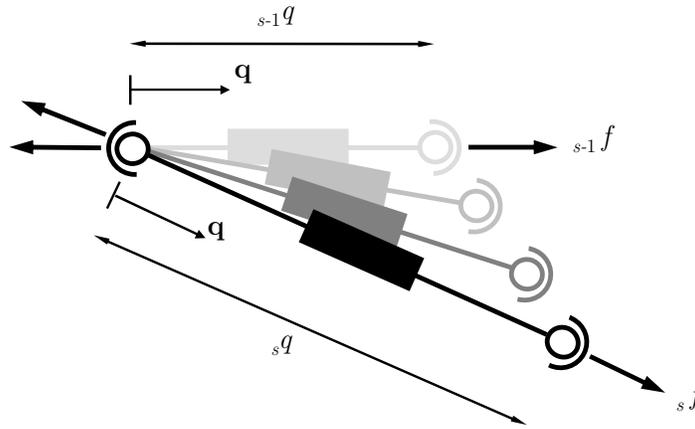


Figure 7.3: Representation of the stroke length and force associated with various stages of under-actuation.

As it can be observed from Eq. (7.6), one needs to know the force and the stroke length to evaluate the actuation energy. This requires the use of the kinematic and kinetostatic relationships of the mechanism, which were developed and presented in Chapters 4 and 5. The amount of actuation energy for the j^{th} member of the i^{th} module, required to do work when going from the $(s-1)^{\text{th}}$ to the s^{th} stage, can be classified by accounting for the external loads acting on the system and the desired motion. In general, there are a total of six possible scenarios that can take place when considering the force and stroke length associated with an actuator. These are summarized in Eq. (7.9):

$${}^i\mathcal{E}_j = \begin{cases} \mathbf{A}: & \left| {}^i f_{j_{avg}} \Delta_s^i q_j \right| \text{ if } {}^i f_{j_{avg}} < 0 \text{ and } \Delta_s^i q_j > 0 \\ \mathbf{B}: & \left| {}^i f_{j_{avg}} \Delta_s^i q_j \right| \text{ if } {}^i f_{j_{avg}} > 0 \text{ and } \Delta_s^i q_j < 0 \\ \mathbf{C}: & 0 \text{ if } {}^i f_{j_{avg}} < 0 \text{ and } \Delta_s^i q_j < 0 \\ \mathbf{D}: & 0 \text{ if } {}^i f_{j_{avg}} > 0 \text{ and } \Delta_s^i q_j > 0 \\ \mathbf{E}: & 0 \text{ if } {}^i f_{j_{avg}} = 0 \\ \mathbf{F}: & 0 \text{ if } \Delta_s^i q_j = 0 \end{cases} . \quad (7.9)$$

Positive values for ${}^i f_{j_{avg}}$ and $\Delta_s^i q_j$ imply a tension force and an increase in the stroke, whereas negative values imply a compression force and a reduction in the stroke. A value of zero for ${}^i f_{j_{avg}}$ or $\Delta_s^i q_j$ implies no load and no stroke change, respectively. This simplified evaluation of the actuation energy does not account for overcoming friction or any other resisting loads other than the internal load acting on the member, resulting from the aerodynamic and inertia loads acting on the wing. Table 7.2 illustrates the six scenarios.

Table 7.2: Six scenarios capturing all possible combinations of loading conditions and motion requirements.

Scenario	Stage ‘s-1’	Stage ‘s’
<i>A</i>		
<i>B</i>		
<i>C</i>		
<i>D</i>		
<i>E</i>		
<i>F</i>		

If the member is seeing a compression load and it is required to increase its length, i.e. scenario \mathcal{A} , or if the member is seeing a tension load and it is required to decrease its length, i.e. scenario \mathcal{B} , actuation energy is required. However, if the member is seeing a compression load and it is required to decrease its length, i.e. scenario \mathcal{C} , or if the member is seeing a tension load and it is required to increase its length, i.e. scenario \mathcal{D} , no actuation energy would be required. In scenarios \mathcal{C} and \mathcal{D} , the actuation force is not to be confused with the average force in the actuator, which would not be equal to zero. Finally, if either of the internal load or the required stroke is equal to zero, i.e. scenario \mathcal{E} or \mathcal{F} , no actuation energy would be required.

At this point, it is noteworthy to discuss how the previously mentioned directional lockability can be used as a measure against load uncertainties. This feature will allow an unlocked passive member to be load-bearing in the direction opposite to the pre-defined motion, in the case of a reversal while morphing. For example, consider a passive member that is unlocked and extending. Although, it is not load-bearing during this event, it can still be thought of as seeing an imaginary tension load. In other words, if a tension load was acting on it, the member would not be required to react to this load; instead it would just move wherever the platform would direct it as long as it would be extending. However, if the external loads changed such that the loads acting on the platform wanted to push this unlocked member back, the directional lockability would not allow this. At that point the passive member would react to the compression load. The member would remain locked in that position for as long as the reversal persists. The opposite would hold if the member was retracting, that is, it would react to any sudden tension loads that may arise while retracting. The novel design of the passive member, which enables the directional lockability, is presented in Appendix D.

7.4 Control Implementation

In addition to the actuation efficiency of the proposed robot, one major advantage of using the proposed under-actuated motion control is that conventional control schemes can be used while maintaining multiple degrees of static redundancy in the mechanism. The control can be set up using analogue position and speed inputs for the actuators, and digital inputs for the passive members. The digital inputs are determined using the proposed optimization algorithm presented in Sec. 7.3. The position inputs are determined using the kinematics of the VGW, presented in Chapter 4. The speed input for each actuator is determined via

$${}^i v_j = \frac{\Delta {}^i q_j}{\Delta t} \quad (7.10)$$

where ${}^i v_j$ and $\Delta {}^i q_j$ are the speed and the stroke change of the j^{th} member of the i^{th} module, respectively, and Δt is the specified time period, during which actuation is to occur. The time period is common to all actuators within the module for any given stage. This implementation will guarantee that all the actuators will start and stop simultaneously while going from one pose to another.

7.5 Variable Topology and Internal Loads

As demonstrated earlier, the proposed system can produce four distinct isostatic topologies. For any given external load, the internal loads can vary both in direction, i.e. tension or compression, and magnitude, as a function of the topology, as demonstrated in Figs. 7.4 through 7.9. The solid lines in Figs. 7.4 through 7.9 represent load-bearing limbs, whereas the dashed lines represent unlocked passive limbs.

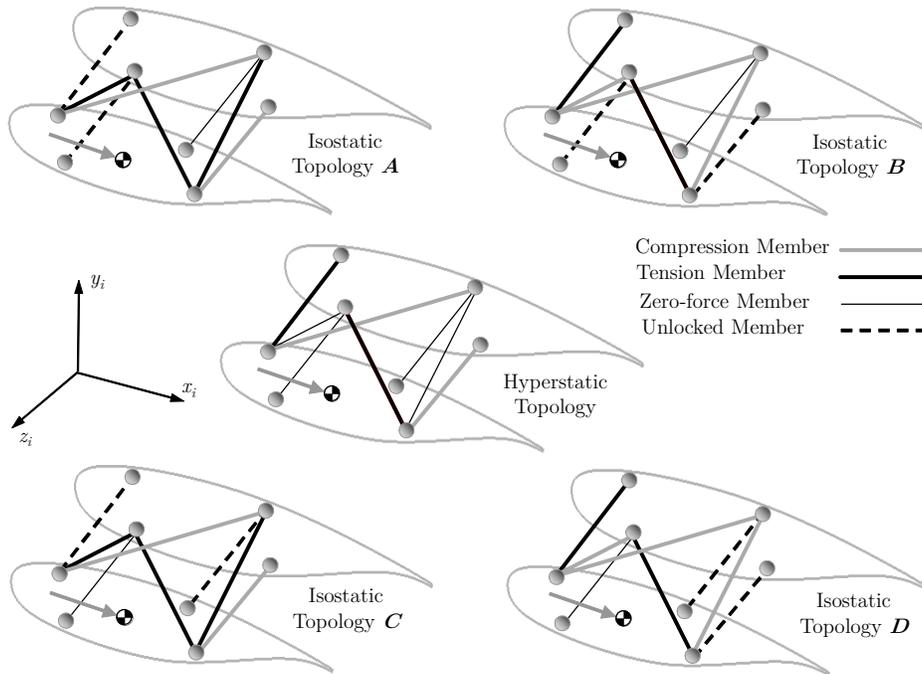


Figure 7.4: Internal loads under ${}^i\mathbf{F}_{i+1} = [F_x \ 0 \ 0 \ 0 \ 0 \ 0]^T$.

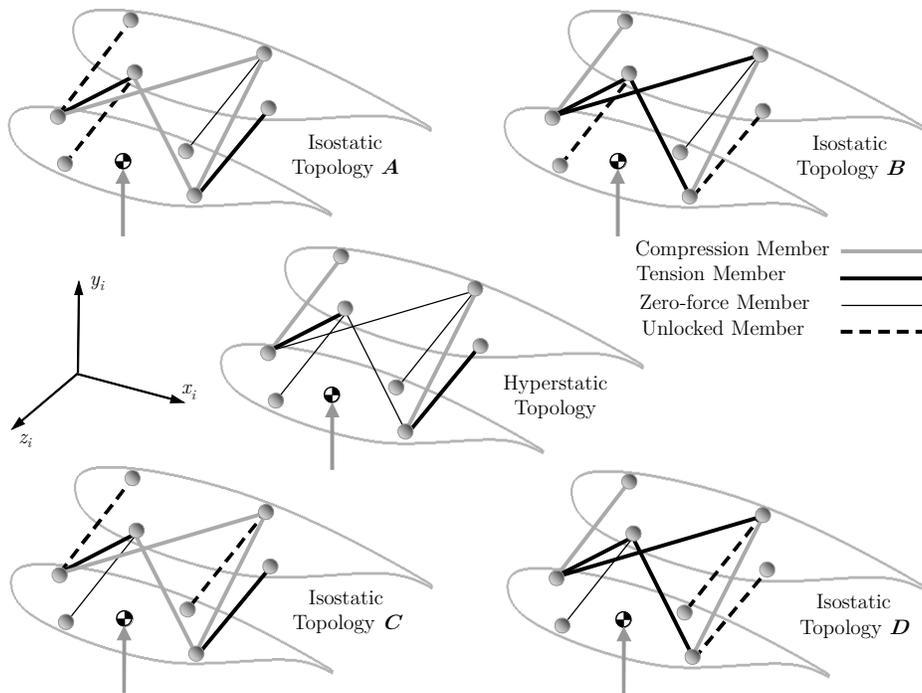


Figure 7.5: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ F_y \ 0 \ 0 \ 0 \ 0]^T$.

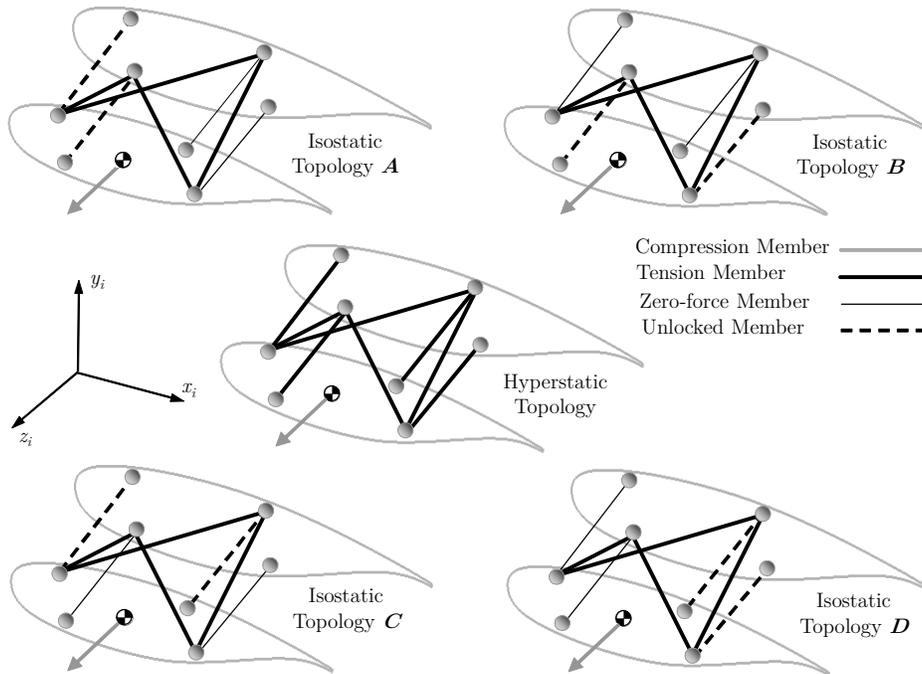


Figure 7.6: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ F_z \ 0 \ 0 \ 0]^T$.

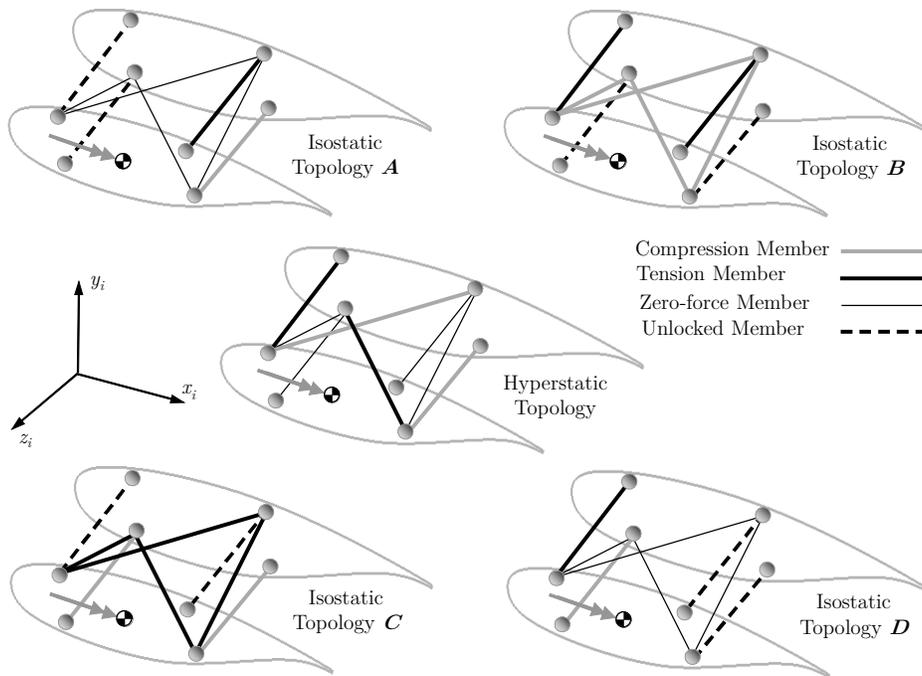


Figure 7.7: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ 0 \ M_x \ 0 \ 0]^T$.

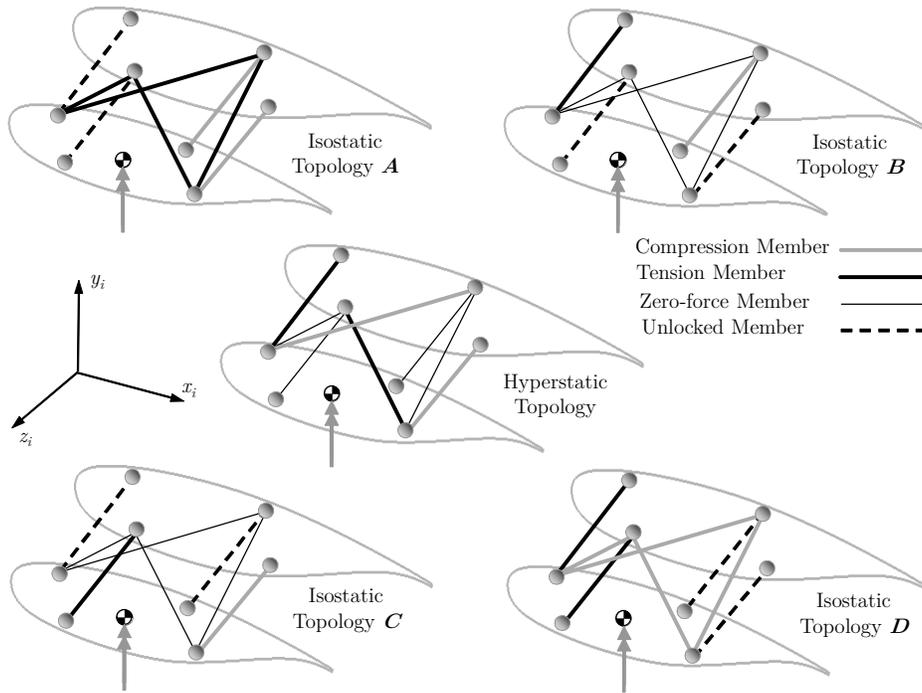


Figure 7.8: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ 0 \ 0 \ M_y \ 0]^T$.

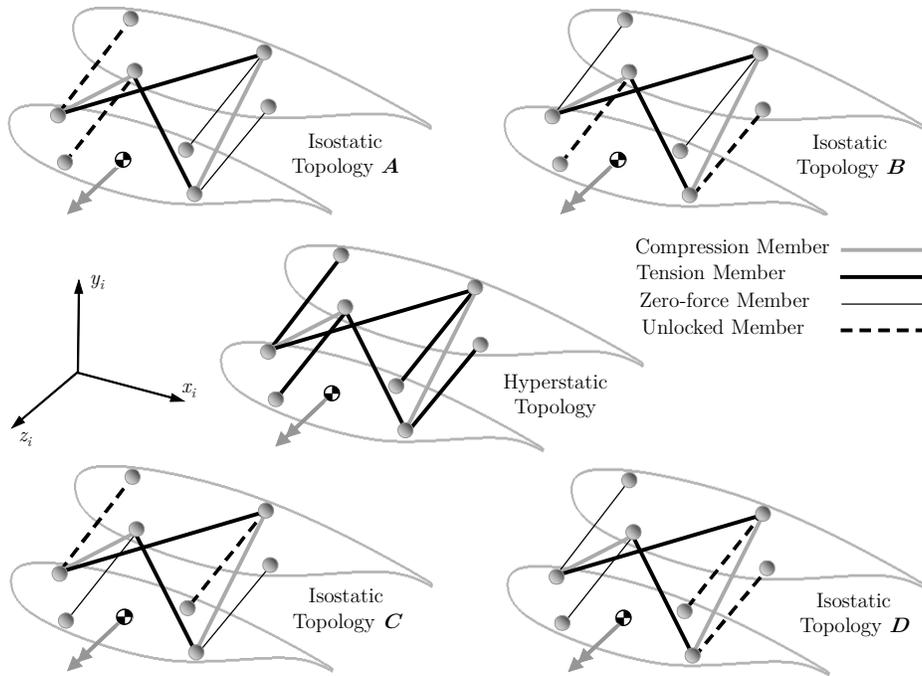


Figure 7.9: Internal loads under ${}^i\mathbf{F}_{i+1} = [0 \ 0 \ 0 \ 0 \ 0 \ M_z]^T$.

By inspecting the variation in internal loads as a function of topology, it can be appreciated that if the right sequence of topologies is used for a given set of external loads, the module can actuate with very little energy.

7.6 Case Study: Simulation, Validation, and Implementation

In this section, the implementation of the design will be demonstrated using a simulation for a wing tip morphing application, followed by the presentation of a multi-module prototype built as a proof of concept to validate the simulation for the under-actuated motion control. The design is applied to the outboard portion of a wing using two morphing modules as illustrated in Fig. 7.10. For clarity in demonstration, only two morphing modules are used. However, it should be highlighted that the proposed modular design is applicable to any number of modules within practical means. All wing sections are constructed using SC(2)-0518 airfoils [119].

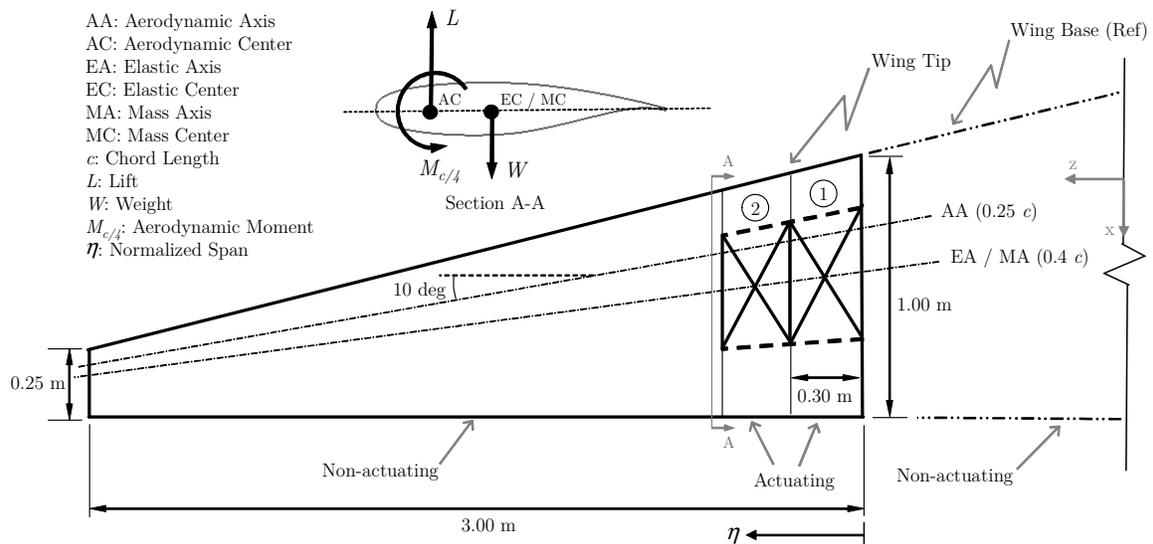


Figure 7.10: Wing tip layout in initial pose (view looking down on the left wing).

Figure 7.10 identifies the two morphing modules and the geometrical features of the wing tip in its home pose. The wing tip has a taper ratio of 0.25, with a sweep of 10 deg. The external loads acting on each section are identified in Fig. 7.10. The aerodynamic and inertia loads are applied at $0.25 c$ and $0.4 c$ of each section, measured from the leading edge, respectively.

To simplify the analysis, initial sweep-induced torsional moments and drag forces have been ignored when determining the intrinsic loads. However, induced drag/thrust forces and the torsional moments caused by changes in the pose have been considered in the formulation. Additionally, it is assumed that the elastic center, the mass center, and the center of rotation are all the same and lie at about $0.4 c$, and remain there throughout morphing.

To minimize the variation in axis definitions, the same axis convention introduced earlier for the wing has been adopted. Using this definition a positive value for $M_{c/4}$ will cause the leading edge to rotate down.

As illustrated in Fig. 7.11, the goal is to go from the initial pose to one with an overall additional 40 deg of cant (Γ), 20 deg of sweep (Λ), 10 deg of twist (α), i.e. toe-out, and 0.07 m of span (b_z). For the auxiliary motions, the modules are allowed to translate 0.05 m along b_x and 0.10 m along b_y . The ratio between the translational cant and sweep motions has been arbitrarily chosen to correspond to the ratio between the cant and the sweep angles. The simplified derivation of the loads for the 1- g loading condition is included in Sec. 7.6.3.

Although for a particular application there may not be a need to change the values of all six DOFs, the ability to control all DOFs gives the VGW the capability to achieve any desired pose without dependent motions, which would traditionally result where conventional lower mobility manipulators are used.

7.6.1 Simulation

Using the minimum actuation energy control scheme, the optimal sequence is determined to be path 16 for both modules. Using this sequence, initially isostatic topology D is used to obtain the final length for member 6. Subsequently, isostatic topology C is used to obtain the final length for member 5. Finally, isostatic topology A is used to simultaneously obtain the final lengths for members 7 and 8. For this particular motion, the external loads presented in Sec. 7.6.3 yield a total actuation energy of approximately 0.6 kJ.

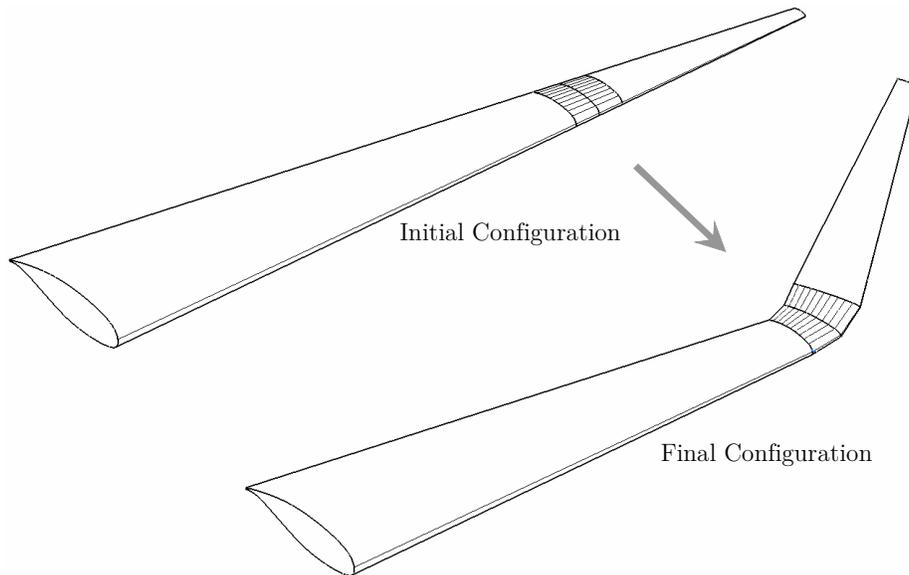


Figure 7.11: Initial and final configurations.

Generally, the sequence in which the modules will actuate can also be included in the optimization problem. However, for brevity it is assumed that both modules morph simultaneously. The proposed algorithm has been implemented using MATLAB[®] and CATIA[®]. Figure 7.12 and Table 7.3 illustrates the simulation for the optimal motion control under the assumed loading condition (see Sec. 7.6.3). Table 7.4 summarizes the pose variables for each stage for module 1.

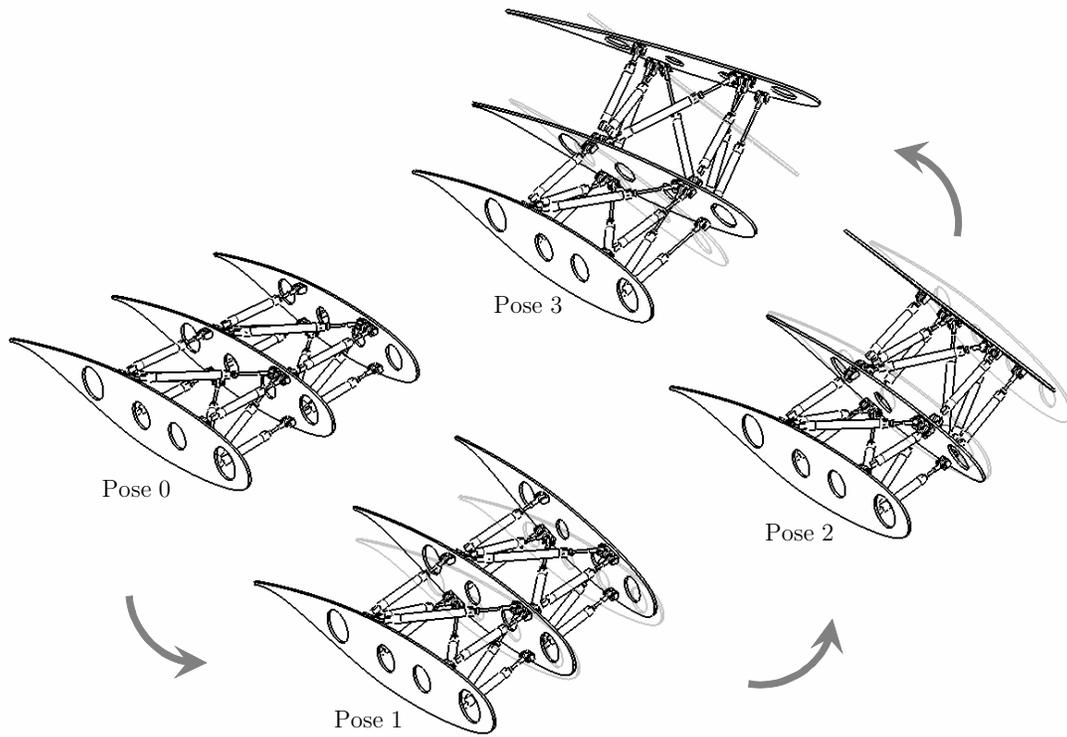


Figure 7.12: Simulation for minimum energy actuation.

Table 7.3: Different views showing the wing shape by stage for the case-study.

View	Pose 0	Pose 1	Pose 2	Pose 3
Isometric				
Looking Aft				
Looking Down				
Looking Outboard				

Table 7.4: Pose variables for the case-study.

	<i>Pose</i>	${}^i b_{i+1_x}$ [m]	${}^i b_{i+1_y}$ [m]	${}^i b_{i+1_z}$ [m]	${}^i \Gamma_{i+1}$ [deg]	${}^i \Lambda_{i+1}$ [deg]	${}^i \alpha_{i+1}$ [deg]
Module 1	0	0.056	0.000	0.300	0.0	0.0	0.0
	1	0.065	0.024	0.298	1.8	-0.5	5.0
	2	0.061	0.022	0.278	-19.1	-3.1	5.0
	3	0.081	0.050	0.333	-20.0	10.0	5.0

Module 2 (not shown in Table 7.4) has similar values for poses 0 and 3 and comparable values for poses 1 and 2 due to the similar motion and loading condition.

The average forces, stroke lengths, and the actuation energy for all three stages for each module are summarized in Tables 7.5 and 7.6.

Table 7.5: Forces and stroke lengths by stage for the case-study.

	Stage	${}^i f_{1_{avg}}$	${}^i f_{2_{avg}}$	${}^i f_{3_{avg}}$	${}^i f_{4_{avg}}$	$\Delta_s^i q_1$	$\Delta_s^i q_2$	$\Delta_s^i q_3$	$\Delta_s^i q_4$
		[N]	[N]	[N]	[N]	[m]	[m]	[m]	[m]
Module 1	0-1	-371	+6800	+1401	+3938	-0.015	+0.010	+0.013	-0.002
	1-2	-37733	-27548	-87202	-65447	+0.002	-0.041	-0.001	-0.029
	2-3	-8293	-1786	-2516	+367	+0.001	+0.071	+0.015	+0.035
Module 2	0-1	-486	+6623	-341	+1849	-0.010	+0.009	+0.014	-0.002
	1-2	-31800	-20609	-73702	-55167	+0.002	-0.038	0.000	-0.028
	2-3	-4850	+3637	-7275	-4154	+0.001	+0.069	+0.015	+0.037

Table 7.6: Actuation energy by stage for the case-study.

		${}^i_s\mathcal{E}_1$	${}^i_s\mathcal{E}_2$	${}^i_s\mathcal{E}_3$	${}^i_s\mathcal{E}_4$	$\sum_{j=1}^4 {}^i_s\mathcal{E}_j$
Stage		[J]	[J]	[J]	[J]	[J]
Module 1	0-1	0	0	0	7.9	7.9
	1-2	75.5	0	0	0	75.5
	2-3	8.3	126.8	37.7	0	172.9
Module 2	0-1	0	0	4.8	3.7	8.5
	1-2	63.6	0	0.0	0	63.6
	2-3	4.9	0	109.1	153.7	267.7

Table 7.7 summarizes the locking sequence for the passive members in both modules during the actuation using the three available modes. Mode L_{E+R} refers to complete locking of a passive member, i.e. both in extension and retraction. Mode L_R implies locked only in retraction, whereas mode L_E implies locked only in extension. For more details on how the passive members operate see Appendix D.

Table 7.7: Locking sequence for the passive members for the case-study.

		Passive Member			
Stage		5	6	7	8
Module 1 / 2	0-1	L_R	L_R	L_{E+R}	L_{E+R}
	1-2	L_E	L_{E+R}	L_E	L_{E+R}
	2-3	L_{E+R}	L_{E+R}	L_R	L_R

7.6.2 Motion Control Validation

To validate the proposed under-actuated motion control, as shown in the simulation (Fig. 7.12), the proposed motion was implemented on the multi-module prototype. Simulink[®] was used to implement the robot control. For any requested motion, the optimal outputs were determined by MATLAB[®] using the proposed formulations and fed to Simulink[®] as control inputs to drive the robot. The Simulink[®] outputs to the control board included the analogue position and speed signals for the four actuators, and the digital signals for the locking control of the four passive members for each module. The actuators were set up using electromechanical linear actuators with position and speed controls. The passive members were set up using hydraulic cylinders along with a series of solenoid and check valves. Figure 7.13 illustrates the prototype going through the proposed under-actuated motion.

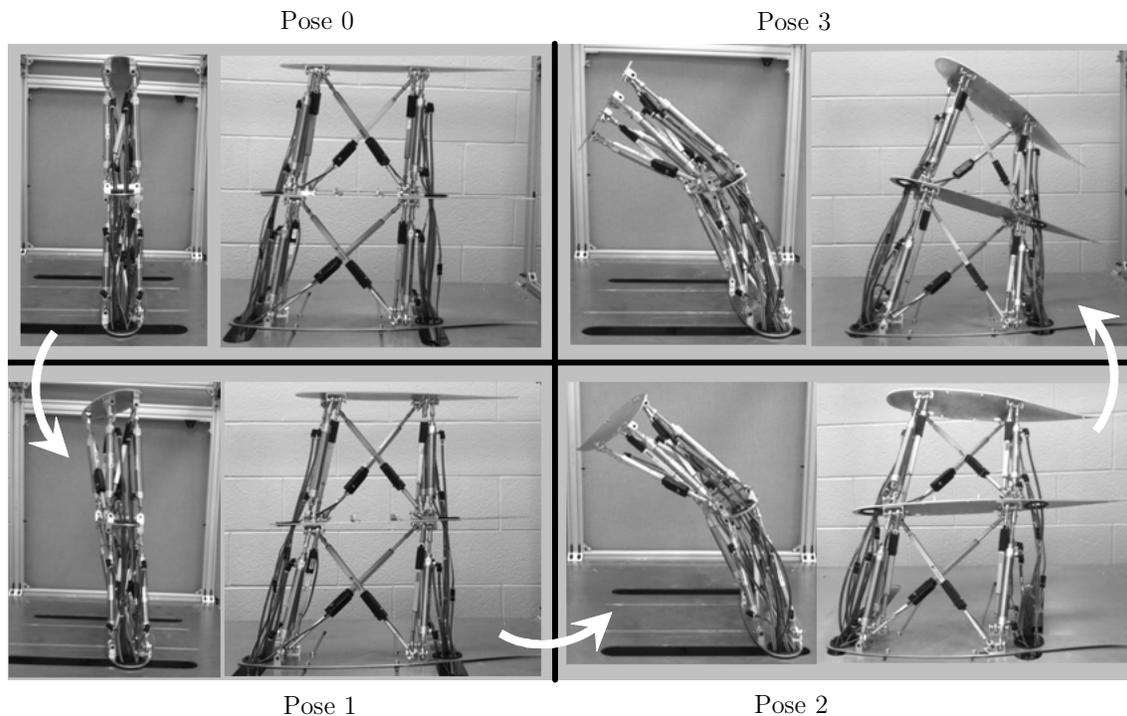


Figure 7.13: Side and front views of the prototype showing poses 0, 1, 2, and 3.

As illustrated in Fig. 7.13, the proposed motion control algorithm was successfully implemented on the prototype to carry out the under-actuated motion shown in the simulation (Fig. 7.12).

7.6.3 External Loads

To calculate the total external loads acting at each platform, a chordal distribution for the lift (L), the weight (W), and the aerodynamic moment ($M_{c/4}$) is assumed. For some arbitrary aircraft weight and configuration in steady level flight, it is assumed that the wing tip section produces 10 kN of lift and 2 kN.m of aerodynamic moment in the position shown in Fig. 7.11 (Initial). Additionally, it is assumed that the actuating modules have a total weight of 0.7 kN, and the non-actuating modules (tip section only) have a weight of 0.8 kN. Using the presented wing geometry (Fig. 7.10), the intrinsic shear force (F_y), bending moment (M_x), and torsional moment (M_z) acting on the i^{th} module are obtained via

$${}^i(F_y^{aero})_i^{IN} = \int_{\eta^{(i)}}^{\eta^{(i+1)}} w^L(\eta) d\eta \quad (7.11)$$

$${}^o(F_y^{inertia})_i^{IN} = \int_{\eta^{(i)}}^{\eta^{(i+1)}} w^W(\eta) d\eta \quad (7.12)$$

$${}^i(M_x^{aero})_i^{IN} = \int_{\eta^{(i)}}^{\eta^{(i+1)}} {}^i(F_y^{aero})_i^{IN} d\eta \quad (7.13)$$

$${}^i(M_x^{inertia})_i^{IN} = \int_{\eta^{(i)}}^{\eta^{(i+1)}} {}^i(F_y^{inertia})_i^{IN} d\eta \quad (7.14)$$

$${}^i(M_z^{aero})_i^{IN} = \int_{\eta^{(i)}}^{\eta^{(i+1)}} w^{M_{c/4}}(\eta) d\eta \quad (7.15)$$

$${}^i(M_z^{inertia})_i^{IN} = \int_{\eta(i)}^{\eta(i+1)} (0.15c(\eta))w^W(\eta)d\eta \quad (7.16)$$

where

$$w^L(\eta) = \frac{L}{S}c(\eta) \quad (7.17)$$

$$w^W(\eta) = \frac{W}{S}c(\eta) \quad (7.18)$$

$$w^{M_{c/4}}(\eta) = \frac{M_{c/4}}{S}c(\eta) \quad (7.19)$$

$$c(\eta) = (1 - 0.75\eta) \quad (7.20)$$

where w represents the load distribution as a function of the chord ($c(\eta)$); η and S are the normalized span and the wing area, respectively. The term $\eta(i)$ implies η at platform i .

7.6.4 MATLAB[®] Interface/Implementation

MATLAB[®] and Simulink[®] were used to implement the control scheme for the prototype. Figure 7.14 illustrates the Graphical User Interface (GUI).

The schematic representations of all four types/configurations of the limbs are summarized in Table 7.8. These schematics are used in the presented simulations (Fig. 7.15).

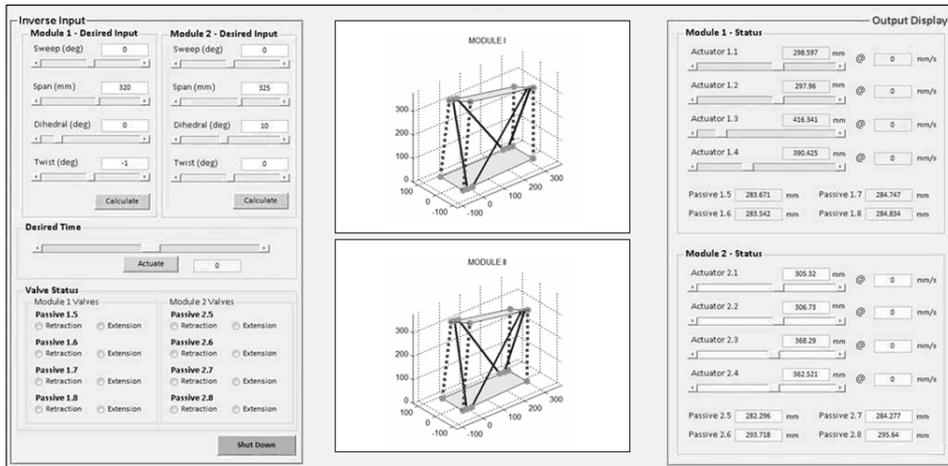


Figure 7.14: Snapshot of the MATLAB® GUI used for pose input.

Table 7.8: Schematics of limb types/configurations.

Schematic	Description
	Actuated Limb
	Locked Passive Limb
	Unlocked Passive Limb without Virtual Constraint
	Unlocked Passive Limb with Virtual Constraint

Figure 7.15 illustrates the results for the optimal motion control (only Module 1 is illustrated). There are a total of seven steps⁵ in each simulation, where each step represents a change in either the topology or the pose.

⁵ A step is not to be confused with the previously defined stage, which only represents a change in the pose.

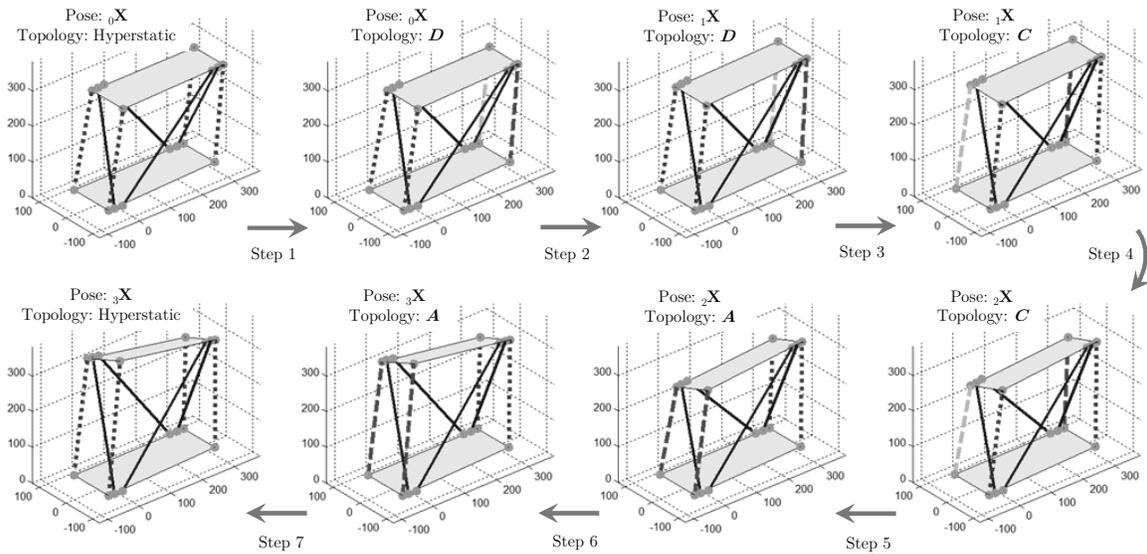


Figure 7.15: Schematic illustration of the VGW undergoing the under-actuated motion with the status of each member identified for each step using the scheme presented in Table 7.8.

7.7 Conclusion

The optimal motion control of modular under-actuated robots for wing morphing applications were presented in this chapter. The optimal motion control algorithm which is based on minimum energy actuation was implemented on a wing with two actuating modules. The actuator forces and stroke lengths were obtained through simulation to determine the optimal path for actuation with minimal energy. The under-actuated motion determined by the simulation was then validated using a prototype.

Chapter 8

Concluding Remarks

Design, analysis, and optimization of a new family of reconfigurable parallel robots with applications in wing morphing have been presented in this thesis. In the proposed design, a total of eight truss members form a rectangular box, four of which are actuated. The other four are passive members, which could be locked and unlocked to provide rigidity for the structure, while accommodating the kinematic requirements. The main feature of the design is the utilization of these active and passive linearly adjustable members to replace the structure of a conventional wing-box, subsequently providing the necessary stiffness and load-bearing capabilities for the wing. Hence, with the exception of the non-structural skin, no additional structure would be required, leading to a relatively light-weight design for the morphing wing.

Although the presented design is theoretically applicable to any part of the wing, in practice it is best suited for regions closer to the wing tips, as most conventional commercial/business aircraft use a large segment of the wing (usually up to the vicinity of the ailerons) as a fuel tank, leaving the most outboard portions dry. However, if a larger scope of implementation is envisioned then suitable compromises

must be made using less conventional configurations to make the application practical.

8.1 Summary of Contributions

A new family of under-actuated robots, referred to as Parallel Robots with Enhanced Stiffness (PRES) were introduced based on the concept of static redundancy. Additionally, the kinematic and kinetostatic analyses of these robots were presented. A novel motion control approach referred to as Under-actuation with Virtual Alternating Constraints (UVAC) was developed utilizing the redundancy present in the system to provide full finite mobility. This contribution led to the following peer-reviewed journal publication:

- **Moosavian, A.**, Xi, F., “Design and Analysis of Reconfigurable Parallel Robots with Enhanced Stiffness,” *Mechanism and Machine Theory*, Vol. 77, pp. 92-110, 2014.

The framework for designing the Variable Geometry Wing-box (VGW) was developed using PRES and UVAC. This included the development of the local and global static and kinematic formulations specific to a morphing wing. Based on this framework, the concept of minimal-energy actuation was introduced and used to generate a novel optimal motion control algorithm for the VGW. This contribution led to the following peer-reviewed journal publication:

- **Moosavian, A.**, Xi, F., and Hashemi, S. M., “Design and Motion Control of Fully Variable Morphing Wings,” *AIAA Journal of Aircraft*, Vol. 50, No. 4, pp. 1189-1201, 2013.

The optimal configuration design for the VGW was presented. The optimal configuration design consisted of 1) the optimal limb configuration, seeking the optimal design of the kinematic joints and links, and 2) the optimal topological configuration, seeking the minimal compliance solution to the placement of the limbs within the design space. This contribution led to the following peer-reviewed journal publication:

- **Moosavian, A.**, Xi, F., and Hashemi, S. M., “Optimal Configuration Design for the Variable Geometry Wing-Box,” *AIAA Journal of Aircraft*, Vol. 51, No. 3, pp. 811-823, 2014.

Modular design of the proposed robots with specific applications to wing morphing was presented. This included the study of the connectivity and mobility of the system to accommodate limb modularity in the design of the PRES. This approach provided the modular building blocks for designing the VGW. This contribution led to the following submission:

- **Moosavian, A.**, Xi, F., “Modular Design of Parallel Robots with Static Redundancy,” (submitted).

8.2 Future Work

As an extension of this work, it is deemed useful that the future efforts would include the evaluation of responses with respect to stability and the inclusion of dynamic parameters in the motion control algorithm. In addition to the previously discussed application types such as maximizing the lift to drag ratio for different flight regimes, the proposed mechanism can be used as a means for active aeroelastic tailoring or flutter suppression. Generally, this could be done in conjunction with range

improvement. In addition, from a structural design point of view, the development of a suitable methodology for detailed design and sizing of the active and passive members will also be valuable extensions to the current work. Furthermore, to build upon the work presented in this thesis with respect to the development and testing of the prototype, additional efforts in the area of calibration are deemed extremely valuable. The calibration efforts will help to increase the fidelity of the proposed under-actuated motion control.

Although further development of several areas such as investigating the stability of the VGW in motion, or accounting for the dynamic response of the system in the design, appear fitting to advance the theory and application presented in this thesis, they do not conceal the fact that a suitable skin concept must be developed in order to utilize the VGW design. Despite being beyond the scope of this thesis, some preliminary suggestions are included here to lay the groundwork for future research in the area of morphing skins.

From a structural perspective, the aircraft skins in contact with the air flow can be categorized into two classes: *a*) structural and *b*) non-structural. Structural skins assist in providing rigidity for the structure and transferring loads between primary structures. Non-structural skins, or fairings, do not generally participate in transferring loads between primary structures due to the fact that much stiffer load paths already exist between these structures being joined by the skins. The VGW has been designed with the aforementioned design philosophy in mind, in which the limbs provide the required stiffness and thus load-carrying capabilities for the wing, without having to rely on any structural skin elements such as those used on a conventional non-morphing wing-box. Therefore, primarily the morphing skin for the VGW will have to provide enough rigidity to handle the local aerodynamic loads.

Additionally, the non-structural skins may be subjected to some form of pre-load to minimize/avoid separation in areas where a high negative pressure (suction) exists (typically closer to the leading edge regions). Lastly, the skins must be able to withstand the loads brought about due to changes in the shape of the wing. Therefore, the loads required for structural analysis of the morphing skins can be categorized into three main types: 1) aerodynamic loads in the form of local suction and pressure, 2) pre-loads by design to minimize separation under suction, and 3) loads caused by the morphing motion of the wing.

Generating a systematic approach that can capture these various loadings and assess the structural responses can serve as a valuable tool to methodically evaluate different skin concepts. Ultimately, the proposed structural analysis approach would have to be combined with the required range of motion and the aerodynamic requirements to determine the suitable discretization of the skins.

Appendix A

VGW Under-actuation

The following MATLAB® code was created to numerically obtain solutions for the VGW under-actuation problem:

```
1    %%% A. MOOSAVIAN, Mar 2014
2    clear; clc;
3
4    % joint coordinates (module 1) [in]:
5    Ab{1} = [11.15; 2.708; 0.75];
6    Ab{2} = [-1.073; -2.603; 0.75];
7    Ab{3} = [-0.21; -2.679; 0.75];
8    Ab{4} = [10.287; 2.782; 0.75];
9
10   Pb{1} = [11.895; 2.645; 0.75];
11   Pb{2} = [11.895; -2.018; 0.75];
12   Pb{3} = [-1.819; 2.602; 0.75];
13   Pb{4} = [-1.819; -2.538; 0.75];
14
15   Ap{1} = [10.105; -1.981; -0.75];
16   Ap{2} = [-1.227; 2.4; -0.75];
17   Ap{3} = [9.254; -2.141; -0.75];
18   Ap{4} = [-0.366; 2.489; -0.75];
19
20   Pp{1} = [10.84; 2.406; -0.75];
21   Pp{2} = [10.84; -1.843; -0.75];
22   Pp{3} = [-1.971; 2.324; -0.75];
23   Pp{4} = [-1.971; -2.266; -0.75];
24
25   % upper pose limit [in,deg]:
26   Upper_Pose_Limit = [6; 6; 16; deg2rad(5); deg2rad(20);
27   deg2rad(15)];
27
28   % lower pose limit [in,deg]:
29   Lower_Pose_Limit = [-6; -6; 8; deg2rad(-30); deg2rad(-5);
30   deg2rad(-5)];
```

```

31 % upper bounds on actuators [in]:
32 Upper_Limit_qA{1} = 14.0;
33 Upper_Limit_qA{2} = 14.0;
34 Upper_Limit_qA{3} = 17.0;
35 Upper_Limit_qA{4} = 17.0;
36
37 % lower bounds on actuators [in]:
38 Lower_Limit_qA{1} = 10.0;
39 Lower_Limit_qA{2} = 10.0;
40 Lower_Limit_qA{3} = 13.0;
41 Lower_Limit_qA{4} = 13.0;
42
43 % upper bounds on passive members [in]:
44 Upper_Limit_qP{1} = 14.0;
45 Upper_Limit_qP{2} = 14.0;
46 Upper_Limit_qP{3} = 14.0;
47 Upper_Limit_qP{4} = 14.0;
48
49 % lower bounds on passive members [in]:
50 Lower_Limit_qP{1} = 8.0;
51 Lower_Limit_qP{2} = 8.0;
52 Lower_Limit_qP{3} = 8.0;
53 Lower_Limit_qP{4} = 8.0;
54
55 % initial pose [in,deg]:
56 Pose_0_0 = [0; 0; 11.811; 0; 0; 0];
57
58 % moving platform axis origin in the stationary platform system
for Pose_0_0:
59 h_0 = [Pose_0_0(1,1) ; Pose_0_0(2,1); Pose_0_0(3,1)];
60
61 % rotation matrix (Pose_0_0):
62 Rx_0 = Pose_0_0(4,1);
63 Ry_0 = Pose_0_0(5,1);
64 Rz_0 = Pose_0_0(6,1);
65
66 RX_0 = [1 0 0; 0 cos(Rx_0) -sin(Rx_0); 0 sin(Rx_0) cos(Rx_0)];
67 RY_0 = [cos(Ry_0) 0 sin(Ry_0); 0 1 0; -sin(Ry_0) 0 cos(Ry_0)];
68 RZ_0 = [cos(Rz_0) -sin(Rz_0) 0; sin(Rz_0) cos(Rz_0) 0; 0 0 1];
69
70 R_0 = RX_0 * RY_0 * RZ_0;
71
72 % loop closure equations (Pose_0_0):
73 for i = 1:4
74     QA_0{i} = h_0 + R_0*Ap{i} - Ab{i};
75     QP_0{i} = h_0 + R_0*Pp{i} - Pb{i};
76     qA_0{i} = norm(QA_0{i});
77     qP_0{i} = norm(QP_0{i});
78 end
79
80 % final pose [in,deg] (supplied by user):
81 Pose_3_0 = [3.2; 2.0; 13.0; deg2rad(-20); deg2rad(10);
deg2rad(5)];
82

```

```

83 % moving platform axis origin in the stationary platform system
    for Pose_3_0:
84 h_3 = [Pose_3_0(1,1) ; Pose_3_0(2,1); Pose_3_0(3,1)];
85
86 % rotation matrix (Pose_3_0):
87 Rx_3 = Pose_3_0(4,1);
88 Ry_3 = Pose_3_0(5,1);
89 Rz_3 = Pose_3_0(6,1);
90
91 RX_3 = [1 0 0; 0 cos(Rx_3) -sin(Rx_3); 0 sin(Rx_3) cos(Rx_3)];
92 RY_3 = [cos(Ry_3) 0 sin(Ry_3); 0 1 0; -sin(Ry_3) 0 cos(Ry_3)];
93 RZ_3 = [cos(Rz_3) -sin(Rz_3) 0; sin(Rz_3) cos(Rz_3) 0; 0 0 1];
94
95 R_3 = RX_3 * RY_3 * RZ_3;
96
97 % loop closure equations (Pose_3_0):
98 for i = 1:4
99     QA_3{i} = h_3 + R_3*Ap{i} - Ab{i};
100    QP_3{i} = h_3 + R_3*Pp{i} - Pb{i};
101    qa_3{i} = norm(QA_3{i});
102    qp_3{i} = norm(QP_3{i});
103 end
104
105 % checking the stroke lengths for Pose_3_0:
106 if qa_3{1}>Upper_Limit_qA{1} || qa_3{2}>Upper_Limit_qA{2} || ...
107    qa_3{3}>Upper_Limit_qA{3} || qa_3{4}>Upper_Limit_qA{4} || ...
108    qa_3{1}<Lower_Limit_qA{1} || qa_3{2}<Lower_Limit_qA{2} || ...
109    qa_3{3}<Lower_Limit_qA{3} || qa_3{4}<Lower_Limit_qA{4} || ...
110    qp_3{1}>Upper_Limit_qp{1} || qp_3{2}>Upper_Limit_qp{2} || ...
111    qp_3{3}>Upper_Limit_qp{3} || qp_3{4}>Upper_Limit_qp{4} || ...
112    qp_3{1}<Lower_Limit_qp{1} || qp_3{2}<Lower_Limit_qp{2} || ...
113    qp_3{3}<Lower_Limit_qp{3} || qp_3{4}<Lower_Limit_qp{4}
114    disp(' ');
115    disp('You have surpassed the stroke limits for Pose_3_0.');
```

116 disp('Let's try again!');

117 else

118

119 % 16 possible paths to choose from for under-actuation:

```

120 Order{1} = [1 2 3 3 1 4]; % Path 1
121 Order{2} = [1 2 3 3 4 1]; % Path 2
122 Order{3} = [1 2 4 4 3 2]; % Path 3
123 Order{4} = [1 2 4 4 2 3]; % Path 4
124 Order{5} = [1 3 2 2 1 4]; % Path 5
125 Order{6} = [1 3 2 2 4 1]; % Path 6
126 Order{7} = [1 3 4 4 2 3]; % Path 7
127 Order{8} = [1 3 4 4 3 2]; % Path 8
128 Order{9} = [2 4 1 1 2 3]; % Path 9
129 Order{10} = [2 4 1 1 3 2]; % Path 10
130 Order{11} = [2 4 3 3 1 4]; % Path 11
131 Order{12} = [2 4 3 3 4 1]; % Path 12
132 Order{13} = [3 4 1 1 3 2]; % Path 13
133 Order{14} = [3 4 1 1 2 3]; % Path 14
134 Order{15} = [3 4 2 2 1 4]; % Path 15
135 Order{16} = [3 4 2 2 4 1]; % Path 16

```

```

136
137 % initiating a starting point for the solver (the pose mid-way
    between the initial and final poses is used here):
138 Pose_MID = Pose_0_0 + (0.5*(Pose_3_0 - Pose_0_0));
139
140 for PATH = 1 %input the desired path (1-16)
141
142 % calculating the first intermediate pose (Pose_1_0):
143 S_1 =@(X_1)...
144 [(norm([X_1(1); X_1(2); X_1(3)]-Pb{Order{PATH}(1,1)}))^2 +
    2*([X_1(1); X_1(2); X_1(3)]-Pb{Order{PATH}(1,1)})' *
    [cos(X_1(5))*cos(X_1(6)), -cos(X_1(5))*sin(X_1(6)), sin(X_1(5));
    sin(X_1(4))*sin(X_1(5))*cos(X_1(6))+cos(X_1(4))*sin(X_1(6)), -
    sin(X_1(4))*sin(X_1(5))*sin(X_1(6))+cos(X_1(4))*cos(X_1(6)), -
    sin(X_1(4))*cos(X_1(5)); -
    cos(X_1(4))*sin(X_1(5))*cos(X_1(6))+sin(X_1(4))*sin(X_1(6)),
    cos(X_1(4))*sin(X_1(5))*sin(X_1(6))+sin(X_1(4))*cos(X_1(6)),
    cos(X_1(4))*cos(X_1(5))] * Pp{Order{PATH}(1,1)} +
    (norm(Pp{Order{PATH}(1,1)}))^2 - qP_0{Order{PATH}(1,1)}^2 ; ...
145 (norm([X_1(1); X_1(2); X_1(3)]-Pb{Order{PATH}(1,2)}))^2 +
    2*([X_1(1); X_1(2); X_1(3)]-Pb{Order{PATH}(1,2)})' *
    [cos(X_1(5))*cos(X_1(6)), -cos(X_1(5))*sin(X_1(6)), sin(X_1(5));
    sin(X_1(4))*sin(X_1(5))*cos(X_1(6))+cos(X_1(4))*sin(X_1(6)), -
    sin(X_1(4))*sin(X_1(5))*sin(X_1(6))+cos(X_1(4))*cos(X_1(6)), -
    sin(X_1(4))*cos(X_1(5)); -
    cos(X_1(4))*sin(X_1(5))*cos(X_1(6))+sin(X_1(4))*sin(X_1(6)),
    cos(X_1(4))*sin(X_1(5))*sin(X_1(6))+sin(X_1(4))*cos(X_1(6)),
    cos(X_1(4))*cos(X_1(5))] * Pp{Order{PATH}(1,2)} +
    (norm(Pp{Order{PATH}(1,2)}))^2 - qP_0{Order{PATH}(1,2)}^2 ; ...
146 (norm([X_1(1); X_1(2); X_1(3)]-Pb{Order{PATH}(1,3)}))^2 +
    2*([X_1(1); X_1(2); X_1(3)]-Pb{Order{PATH}(1,3)})' *
    [cos(X_1(5))*cos(X_1(6)), -cos(X_1(5))*sin(X_1(6)), sin(X_1(5));
    sin(X_1(4))*sin(X_1(5))*cos(X_1(6))+cos(X_1(4))*sin(X_1(6)), -
    sin(X_1(4))*sin(X_1(5))*sin(X_1(6))+cos(X_1(4))*cos(X_1(6)), -
    sin(X_1(4))*cos(X_1(5)); -
    cos(X_1(4))*sin(X_1(5))*cos(X_1(6))+sin(X_1(4))*sin(X_1(6)),
    cos(X_1(4))*sin(X_1(5))*sin(X_1(6))+sin(X_1(4))*cos(X_1(6)),
    cos(X_1(4))*cos(X_1(5))] * Pp{Order{PATH}(1,3)} +
    (norm(Pp{Order{PATH}(1,3)}))^2 - qP_3{Order{PATH}(1,3)}^2 ; ...
147 (X_1(6)-Pose_3_0(6,1));... % this minimizes unwanted twist
148 ]; % ...add more constraints as required
149
150 X_1_0 = Pose_MID; %starting guess
151 options = optimset('Algorithm', 'Levenberg-Marquardt', 'MaxIter',
    50000,...
152 'MaxFunEvals', 1000, 'Diagnostics', 'on', 'Display', 'on',
    'TolFun', 1e-2);
153 [X_1,fval] = fsolve(S_1, X_1_0, options); % calling the solver
154
155 Pose_1_0{PATH} = [X_1(1);X_1(2);X_1(3);X_1(4);X_1(5);X_1(6)];
156
157 % moving platform axis origin in the stationary platform system
    for Pose_1_0:

```

```

158 h_1 = [Pose_1_0{PATH}(1,1); Pose_1_0{PATH}(2,1);
159       Pose_1_0{PATH}(3,1)];
160 % rotation matrix (Pose_1_0):
161 Rx_1 = Pose_1_0{PATH}(4,1);
162 Ry_1 = Pose_1_0{PATH}(5,1);
163 Rz_1 = Pose_1_0{PATH}(6,1);
164
165 RX_1 = [1 0 0; 0 cos(Rx_1) -sin(Rx_1); 0 sin(Rx_1) cos(Rx_1)];
166 RY_1 = [cos(Ry_1) 0 sin(Ry_1); 0 1 0; -sin(Ry_1) 0 cos(Ry_1)];
167 RZ_1 = [cos(Rz_1) -sin(Rz_1) 0; sin(Rz_1) cos(Rz_1) 0; 0 0 1];
168
169 R_1 = RX_1 * RY_1 * RZ_1;
170
171 % using Pose_1_0 to obtain the stroke lengths:
172 for i = 1:4
173     QA_1{i} = h_1 + R_1*Ap{i} - Ab{i};
174     QP_1{i} = h_1 + R_1*Pp{i} - Pb{i};
175     qa_1{i} = norm(QA_1{i});
176     qp_1{i} = norm(QP_1{i});
177 end
178
179 % checking the stroke lengths for Pose_1_0:
180 if qa_1{1}>Upper_Limit_qA{1} || qa_1{2}>Upper_Limit_qA{2} || ...
181    qa_1{3}>Upper_Limit_qA{3} || qa_1{4}>Upper_Limit_qA{4} || ...
182    qa_1{1}<Lower_Limit_qA{1} || qa_1{2}<Lower_Limit_qA{2} || ...
183    qa_1{3}<Lower_Limit_qA{3} || qa_1{4}<Lower_Limit_qA{4} || ...
184    qp_1{1}>Upper_Limit_qp{1} || qp_1{2}>Upper_Limit_qp{2} || ...
185    qp_1{3}>Upper_Limit_qp{3} || qp_1{4}>Upper_Limit_qp{4} || ...
186    qp_1{1}<Lower_Limit_qp{1} || qp_1{2}<Lower_Limit_qp{2} || ...
187    qp_1{3}<Lower_Limit_qp{3} || qp_1{4}<Lower_Limit_qp{4}
188    disp(' ');
189    disp('You have surpassed the stroke limits for Pose_1_0.');
```

```

190    disp('Let's try again!');
```

```

191
192 elseif Pose_1_0{PATH}(1,1)>Upper_Pose_Limit(1,1) || ...
193         Pose_1_0{PATH}(1,1)<Lower_Pose_Limit(1,1) || ...
194         Pose_1_0{PATH}(2,1)>Upper_Pose_Limit(2,1) || ...
195         Pose_1_0{PATH}(2,1)<Lower_Pose_Limit(2,1) || ...
196         Pose_1_0{PATH}(3,1)>Upper_Pose_Limit(3,1) || ...
197         Pose_1_0{PATH}(3,1)<Lower_Pose_Limit(3,1) || ...
198         Pose_1_0{PATH}(4,1)>Upper_Pose_Limit(4,1) || ...
199         Pose_1_0{PATH}(4,1)<Lower_Pose_Limit(4,1) || ...
200         Pose_1_0{PATH}(5,1)>Upper_Pose_Limit(5,1) || ...
201         Pose_1_0{PATH}(5,1)<Lower_Pose_Limit(5,1) || ...
202         Pose_1_0{PATH}(6,1)>Upper_Pose_Limit(6,1) || ...
203         Pose_1_0{PATH}(6,1)<Lower_Pose_Limit(6,1)
204         disp(' ');
205         disp('You have surpassed the pose limits for Pose_1_0.');
```

```

206         disp('Let's try again!');
```

```

207 else
208
209 % calculating the second intermediate pose (Pose_2_0):
210 S_2 =@(X_2)...
```

```

211 [(norm([X_2(1); X_2(2); X_2(3)]-Pb{Order{PATH}(1,4)}))^2 +
2*(([X_2(1); X_2(2); X_2(3)]-Pb{Order{PATH}(1,4)})' *
[cos(X_2(5))*cos(X_2(6)), -cos(X_2(5))*sin(X_2(6)), sin(X_2(5));
sin(X_2(4))*sin(X_2(5))*cos(X_2(6))+cos(X_2(4))*sin(X_2(6)), -
sin(X_2(4))*sin(X_2(5))*sin(X_2(6))+cos(X_2(4))*cos(X_2(6)), -
sin(X_2(4))*cos(X_2(5)); -
cos(X_2(4))*sin(X_2(5))*cos(X_2(6))+sin(X_2(4))*sin(X_2(6)),
cos(X_2(4))*sin(X_2(5))*sin(X_2(6))+sin(X_2(4))*cos(X_2(6)),
cos(X_2(4))*cos(X_2(5))] * Pp{Order{PATH}(1,4)}) +
(norm(Pp{Order{PATH}(1,4)}))^2 - qp_1{Order{PATH}(1,4)}^2 ; ...
212 (norm([X_2(1); X_2(2); X_2(3)]-Pb{Order{PATH}(1,5)}))^2 +
2*(([X_2(1); X_2(2); X_2(3)]-Pb{Order{PATH}(1,5)})' *
[cos(X_2(5))*cos(X_2(6)), -cos(X_2(5))*sin(X_2(6)), sin(X_2(5));
sin(X_2(4))*sin(X_2(5))*cos(X_2(6))+cos(X_2(4))*sin(X_2(6)), -
sin(X_2(4))*sin(X_2(5))*sin(X_2(6))+cos(X_2(4))*cos(X_2(6)), -
sin(X_2(4))*cos(X_2(5)); -
cos(X_2(4))*sin(X_2(5))*cos(X_2(6))+sin(X_2(4))*sin(X_2(6)),
cos(X_2(4))*sin(X_2(5))*sin(X_2(6))+sin(X_2(4))*cos(X_2(6)),
cos(X_2(4))*cos(X_2(5))] * Pp{Order{PATH}(1,5)}) +
(norm(Pp{Order{PATH}(1,5)}))^2 - qp_1{Order{PATH}(1,5)}^2 ; ...
213 (norm([X_2(1); X_2(2); X_2(3)]-Pb{Order{PATH}(1,6)}))^2 +
2*(([X_2(1); X_2(2); X_2(3)]-Pb{Order{PATH}(1,6)})' *
[cos(X_2(5))*cos(X_2(6)), -cos(X_2(5))*sin(X_2(6)), sin(X_2(5));
sin(X_2(4))*sin(X_2(5))*cos(X_2(6))+cos(X_2(4))*sin(X_2(6)), -
sin(X_2(4))*sin(X_2(5))*sin(X_2(6))+cos(X_2(4))*cos(X_2(6)), -
sin(X_2(4))*cos(X_2(5)); -
cos(X_2(4))*sin(X_2(5))*cos(X_2(6))+sin(X_2(4))*sin(X_2(6)),
cos(X_2(4))*sin(X_2(5))*sin(X_2(6))+sin(X_2(4))*cos(X_2(6)),
cos(X_2(4))*cos(X_2(5))] * Pp{Order{PATH}(1,6)}) +
(norm(Pp{Order{PATH}(1,6)}))^2 - qp_3{Order{PATH}(1,6)}^2 ; ...
214 (X_2(6)-Pose_3_0(6,1));... % this minimizes unwanted twist
215 ]; % ... add more constraints as required
216
217 X_2_0 = Pose_MID; %starting guess
218 options = optimset('Algorithm', 'Levenberg-Marquardt', 'MaxIter',
50000,...
219 'MaxFunEvals', 1000, 'Diagnostics', 'on', 'Display', 'on',
'TolFun', 1e-2);
220 [X_2,fval] = fsolve(S_2, X_2_0, options); % calling the solver
221
222 Pose_2_0{PATH} = [X_2(1);X_2(2);X_2(3);X_2(4);X_2(5);X_2(6)];
223
224 % moving platform axis origin in the stationary platform system
for Pose_2_0:
225 h_2 = [Pose_2_0{PATH}(1,1); Pose_2_0{PATH}(2,1);
Pose_2_0{PATH}(3,1)];
226
227 % rotation matrix (Pose_2_0):
228 Rx_2 = Pose_2_0{PATH}(4,1);
229 Ry_2 = Pose_2_0{PATH}(5,1);
230 Rz_2 = Pose_2_0{PATH}(6,1);
231
232 RX_2 = [1 0 0; 0 cos(Rx_2) -sin(Rx_2); 0 sin(Rx_2) cos(Rx_2)];
233 RY_2 = [cos(Ry_2) 0 sin(Ry_2); 0 1 0; -sin(Ry_2) 0 cos(Ry_2)];

```

```

234 RZ_2 = [cos(Rz_2) -sin(Rz_2) 0; sin(Rz_2) cos(Rz_2) 0; 0 0 1];
235
236 R_2 = RX_2 * RY_2 * RZ_2;
237
238 % using Pose_2_0 to obtain the stroke lengths:
239 for i = 1:4
240     QA_2{i} = h_2 + R_2*Ap{i} - Ab{i};
241     QP_2{i} = h_2 + R_2*Pp{i} - Pb{i};
242     qA_2{i} = norm(QA_2{i});
243     qP_2{i} = norm(QP_2{i});
244 end
245
246 % checking the stroke lengths for Pose_2_0:
247 if qA_2{1}>Upper_Limit_qA{1} || qA_2{2}>Upper_Limit_qA{2} || ...
248    qA_2{3}>Upper_Limit_qA{3} || qA_2{4}>Upper_Limit_qA{4} || ...
249    qA_2{1}<Lower_Limit_qA{1} || qA_2{2}<Lower_Limit_qA{2} || ...
250    qA_2{3}<Lower_Limit_qA{3} || qA_2{4}<Lower_Limit_qA{4} || ...
251    qP_2{1}>Upper_Limit_qP{1} || qP_2{2}>Upper_Limit_qP{2} || ...
252    qP_2{3}>Upper_Limit_qP{3} || qP_2{4}>Upper_Limit_qP{4} || ...
253    qP_2{1}<Lower_Limit_qP{1} || qP_2{2}<Lower_Limit_qP{2} || ...
254    qP_2{3}<Lower_Limit_qP{3} || qP_2{4}<Lower_Limit_qP{4}
255     disp(' ');
256     disp('You have surpassed the stroke limits for Pose_2_0.');
```

```

257     disp('Let's try again!');
```

```

258
259 elseif Pose_2_0{PATH}(1,1)>Upper_Pose_Limit(1,1) || ...
260         Pose_2_0{PATH}(1,1)<Lower_Pose_Limit(1,1) || ...
261         Pose_2_0{PATH}(2,1)>Upper_Pose_Limit(2,1) || ...
262         Pose_2_0{PATH}(2,1)<Lower_Pose_Limit(2,1) || ...
263         Pose_2_0{PATH}(3,1)>Upper_Pose_Limit(3,1) || ...
264         Pose_2_0{PATH}(3,1)<Lower_Pose_Limit(3,1) || ...
265         Pose_2_0{PATH}(4,1)>Upper_Pose_Limit(4,1) || ...
266         Pose_2_0{PATH}(4,1)<Lower_Pose_Limit(4,1) || ...
267         Pose_2_0{PATH}(5,1)>Upper_Pose_Limit(5,1) || ...
268         Pose_2_0{PATH}(5,1)<Lower_Pose_Limit(5,1) || ...
269         Pose_2_0{PATH}(6,1)>Upper_Pose_Limit(6,1) || ...
270         Pose_2_0{PATH}(6,1)<Lower_Pose_Limit(6,1)
271     disp(' ');
272     disp('You have surpassed the pose limits for Pose_2_0.');
```

```

273     disp('Let's try again!');
```

```

274 else
275
276 end % corresponding to: if stroke lengths are OK for Pose_2_0
277 end % corresponding to: if stroke lengths are OK for Pose_1_0
278 end % corresponding to: for PATH Loop
279 end % corresponding to: if stroke lengths are OK for Pose_3_0

```

Appendix B

VGW Kinetostatics

The following MATLAB® code was created for obtaining the kinetostatic relationships for the VGW:

```
1    %%% A. MOOSAVIAN, Mar 2014
2    clear; clc;
3
4    % user input [in,deg]:
5    Pose = [2.215; 0; 11.811; deg2rad(0); deg2rad(0); deg2rad(0)];
6
7    x = Pose(1,1); y = Pose(2,1); z = Pose(3,1);
8    Rx = Pose(4,1); Ry = Pose(5,1); Rz = Pose(6,1);
9
10   k_const = 1.1e7; % stiffness constant
11
12   % joint coordinates (module 1) [in]:
13   Ab{1} = [11.15; 2.708; 0.75];
14   Ab{2} = [-1.073; -2.603; 0.75];
15   Ab{3} = [-0.21; -2.679; 0.75];
16   Ab{4} = [10.287; 2.782; 0.75];
17
18   Pb{1} = [11.895; 2.645; 0.75];
19   Pb{2} = [11.895; -2.018; 0.75];
20   Pb{3} = [-1.819; 2.602; 0.75];
21   Pb{4} = [-1.819; -2.538; 0.75];
22
23   Ap{1} = [10.105; -1.981; -0.75];
24   Ap{2} = [-1.227; 2.4; -0.75];
25   Ap{3} = [9.254; -2.141; -0.75];
26   Ap{4} = [-0.366; 2.489; -0.75];
27
28   Pp{1} = [10.84; 2.406; -0.75];
29   Pp{2} = [10.84; -1.843; -0.75];
30   Pp{3} = [-1.971; 2.324; -0.75];
31   Pp{4} = [-1.971; -2.266; -0.75];
```

```

32
33 % moving platform axis origin in the stationary platform system:
34 h = [x; y; z];
35
36 % rotation matrix:
37 RX = [1 0 0; 0 cos(Rx) -sin(Rx); 0 sin(Rx) cos(Rx)];
38 RY = [cos(Ry) 0 sin(Ry); 0 1 0; -sin(Ry) 0 cos(Ry)];
39 RZ = [cos(Rz) -sin(Rz) 0; sin(Rz) cos(Rz) 0; 0 0 1];
40 R = RX*RY*RZ;
41
42 % loop closure equations:
43 for i = 1:4
44     QA{i} = h + R*Ap{i} - Ab{i};
45     QP{i} = h + R*Pp{i} - Pb{i};
46     qA{i} = norm(QA{i});
47     qP{i} = norm(QP{i});
48 end
49
50 % Jacobians:
51 for i = 1:4
52     JA{i} = [QA{i}'/qA{i}, (cross(R*Ap{i},QA{i}))'/qA{i}];
53     JP{i} = [QP{i}'/qP{i}, (cross(R*Pp{i},QP{i}))'/qP{i}];
54 end
55
56 J_ISO_A = [JA{1};JA{2};JA{3};JA{4};JP{1};JP{2}];
57 J_ISO_B = [JA{1};JA{2};JA{3};JA{4};JP{1};JP{3}];
58 J_ISO_C = [JA{1};JA{2};JA{3};JA{4};JP{2};JP{4}];
59 J_ISO_D = [JA{1};JA{2};JA{3};JA{4};JP{3};JP{4}];
60 J_HYPER = [JA{1};JA{2};JA{3};JA{4};JP{1};JP{2};JP{3};JP{4}];
61
62 % limb stiffness:
63 for i = 1:4
64     KJA{i} = k_const/qA{i};
65     KJP{i} = k_const/qP{i};
66 end
67
68 K_J_ISO_A = blkdiag(KJA{1},KJA{2},KJA{3},KJA{4},KJP{1},KJP{2});
69 K_J_ISO_B = blkdiag(KJA{1},KJA{2},KJA{3},KJA{4},KJP{1},KJP{3});
70 K_J_ISO_C = blkdiag(KJA{1},KJA{2},KJA{3},KJA{4},KJP{2},KJP{4});
71 K_J_ISO_D = blkdiag(KJA{1},KJA{2},KJA{3},KJA{4},KJP{3},KJP{4});
72 K_J_HYPER = blkdiag(KJA{1},KJA{2},KJA{3},KJA{4},KJP{1},KJP{2},
73     KJP{3},KJP{4});
74
75 % Cartesian stiffness:
76 K_G_ISO_A = J_ISO_A' * K_J_ISO_A * J_ISO_A;
77 K_G_ISO_B = J_ISO_B' * K_J_ISO_B * J_ISO_B;
78 K_G_ISO_C = J_ISO_C' * K_J_ISO_C * J_ISO_C;
79 K_G_ISO_D = J_ISO_D' * K_J_ISO_D * J_ISO_D;
80 K_G_HYPER = J_HYPER'*K_J_HYPER*J_HYPER;
81
82 % external loads:
83 F=[0 ; 1000 ; 0 ; 0 ; 0; 0]; % modify as required
84
85 % platform displacement:

```

```
85 DX_ISO_A = K_G_ISO_A \ F;
86 DX_ISO_B = K_G_ISO_B \ F;
87 DX_ISO_C = K_G_ISO_C \ F;
88 DX_ISO_D = K_G_ISO_D \ F;
89 DX_HYPER = K_G_HYPER \ F;
90
91 % internal loads (isostatic cases):
92 f_ISO_A = J_ISO_A' \ F;
93 f_ISO_B = J_ISO_B' \ F;
94 f_ISO_C = J_ISO_C' \ F;
95 f_ISO_D = J_ISO_D' \ F;
96
97 % internal loads (hyperstatic case):
98 DQ_HYPER = J_HYPER * DX_HYPER;
99 f_HYPER = K_J_HYPER * DQ_HYPER;
```

Appendix C

Alternative Stiffness Model Using FEM

The FE method can be alternatively used to obtain the structural response of the VGW. This model is composed of a series of truss elements simulating the actuated/locked passive members, a series of Single Point Constraints (SPCs) simulating the constrained i^{th} platform, and a Multi Point Constraint (MPC) simulating the rigid $(i+1)^{\text{th}}$ platform. The i^{th} Cartesian coordinate system is used to define all the nodes and vectors presented in this appendix. This is denoted by the use of superscript i . Figure C.1 illustrates the general setup for the FE model.

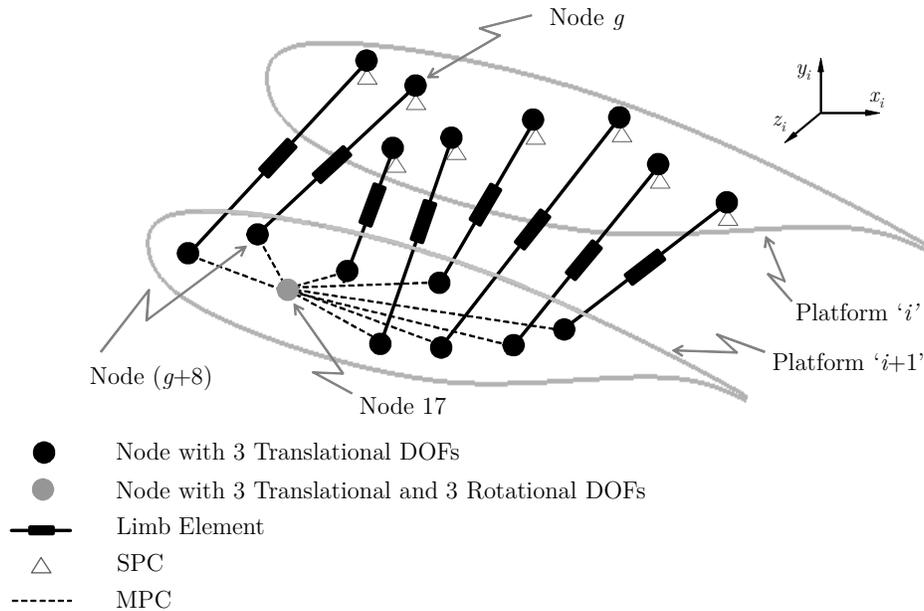


Figure C.1: FE representation for one module of the VGW.

C.1 Limb Element Formulation

Two axial DOFs are assumed for the limb element as identified in Fig. C.2.

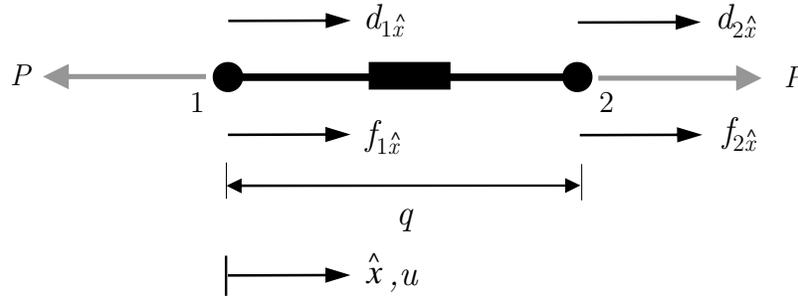


Figure C.2: Typical limb element with the associated DOFs.

In Fig. C.2, f and d are the nodal displacements and forces, respectively; q is the length of the element, u is the displacement along \hat{x} , that is, representing the local element axis. The following elementary relationships from solid mechanics are used to derive the limb element formulations:

Hooke's law:

$$\sigma_{\hat{x}} = E\varepsilon_{\hat{x}} \quad (\text{C.1})$$

strain/displacement relationship:

$$\varepsilon_{\hat{x}} = \frac{du}{d\hat{x}} \quad (\text{C.2})$$

normal stress/force relationship:

$$P = \sigma_{\hat{x}}A \quad (\text{C.3})$$

where $\sigma_{\hat{x}}$, $\varepsilon_{\hat{x}}$, and E represent the normal stress, strain, and modulus of elasticity of the limb element, respectively; P and A are the normal force and the cross sectional area of the element, respectively. By substituting Eq. (C.2) into Eq. (C.1), and then inserting that into Eq. (C.3), one can obtain

$$P = AE \frac{du}{d\hat{x}}. \quad (\text{C.4})$$

Equation (C.4) assumes constant A and E over the whole element. By assuming a linear shape function, the displacement of the element can be represented by

$$u(\hat{x}) = v_1 + v_2\hat{x} \quad (\text{C.5})$$

where the number of coefficients is equal to the number of DOFs, represented by $d_{1\hat{x}}$ and $d_{2\hat{x}}$. To obtain the values of v_1 and v_2 the boundary conditions are applied to Eq. (C.5) yielding the following solutions:

$$v_1 = d_{1\hat{x}} \quad (\text{C.6a})$$

$$v_2 = \frac{d_{2\hat{x}} - d_{1\hat{x}}}{q}. \quad (\text{C.6b})$$

By substituting v_1 and v_2 into Eq. (C.5) and differentiating with respect to \hat{x} , one can obtain

$$\frac{du}{d\hat{x}} = \begin{bmatrix} -\frac{1}{q} & \frac{1}{q} \end{bmatrix} \begin{bmatrix} d_{1\hat{x}} \\ d_{2\hat{x}} \end{bmatrix}. \quad (\text{C.7})$$

Also, corresponding to Fig. C.2,

$$f_{1\hat{x}} = -P \quad (\text{C.8a})$$

$$f_{2\hat{x}} = P. \quad (\text{C.8b})$$

Therefore, by substituting Eqs. (C.7) and (C.8) into Eq. (C.4) one has

$$\begin{bmatrix} f_{1\hat{x}} \\ f_{2\hat{x}} \end{bmatrix} = {}_e\mathbf{k} \begin{bmatrix} d_{1\hat{x}} \\ d_{2\hat{x}} \end{bmatrix} \quad (\text{C.9})$$

where

$${}_e\mathbf{k} = \frac{AE}{q} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (\text{C.10})$$

is the stiffness of a limb element in its local axis. For the VGW application, the term AE/q is simply replaced by an equivalent stiffness constant denoted by κ_j , that is, the stiffness constant of the j^{th} element, which is the same as the one used in Sec. 5.1. The element stiffness matrix of the j^{th} element, defined in the element local coordinate, is then transformed into the Cartesian coordinate, that is, the axis corresponding to platform i :

$${}^i\mathbf{k}_j = {}^i\mathcal{T}_j^T {}_e\mathbf{k}_j {}^i\mathcal{T}_j. \quad (\text{C.11})$$

The derivation of this transformation is similar to the one for obtaining the global stiffness of the VGW as presented in Sec. 5.3. Matrix ${}^i\mathcal{T}_j$ is the transformation matrix for the j^{th} element represented by

$${}^i\mathcal{T}_j = \begin{bmatrix} xC_j & yC_j & zC_j & 0 & 0 & 0 \\ 0 & 0 & 0 & xC_j & yC_j & zC_j \end{bmatrix} \quad (\text{C.12})$$

where xC_j , yC_j , and zC_j represent the direction cosines of the local x axis of the j^{th} element with respect to the i^{th} Cartesian coordinate system. After the transformations are complete, the global FEM stiffness matrix, ${}_{FEM}{}^i\mathbf{K}$, of the module is assembled by the superposition method using MATLAB[®]. The MATLAB[®] code is presented in this appendix following the presentation of the formulation for the platform element.

C.2 Platform Element Formulation

An MPC is used to generate a rigid element to represent the rigid platform. To incorporate the rigid element formulation, one can start with the general formulation for expressing force and displacement relationships for the 54-DOF system, that is, six per limb element plus six for the independent node of the platform element:

$${}_{FEM}{}^i\mathbf{F} = {}_{FEM}{}^i\mathbf{K} \quad {}_{FEM}{}^i\mathbf{D} \quad (\text{C.13})$$

54×54 54×1

where ${}_{FEM}{}^i\mathbf{D}$ and ${}_{FEM}{}^i\mathbf{F}$ represent the displacement and load vectors associated with the 54-DOF system defined in the i^{th} coordinate frame. The load vector can be decomposed into three components:

$${}_{FEM}{}^i\mathbf{F} = {}_{FEM}{}^i\mathbf{F}_P + {}_{FEM}{}^i\mathbf{F}_R + {}_{FEM}{}^i\mathbf{F}_C \quad (\text{C.14})$$

where ${}_{FEM}{}^i\mathbf{F}_R$, and ${}_{FEM}{}^i\mathbf{F}_C$ represent the loads associated with the rigid element, and the SPCs, respectively; ${}_{FEM}{}^i\mathbf{F}_P$ represents the loads associated with any other nodes.

Vector ${}_{FEM}{}^i\mathbf{F}_P$ is effectively null since all DOFs not associated with the SPCs are included in ${}_{FEM}{}^i\mathbf{F}_R$; thus, it can be removed. Furthermore, by eliminating the single-point constrained DOFs from the relationship, Eq. (C.13) can be reduced to

$${}_{FEM}{}^i\mathbf{F}_R = {}_{FEM}{}^i\mathbf{K} \quad {}_{FEM}{}^i\mathbf{D} \quad (C.15)$$

$30 \times 30 \quad 30 \times 1$

and further decomposed in terms of dependent and independent components:

$$\begin{bmatrix} {}_{FEM}{}^i\mathbf{K}_{dep} & \mathbf{0} \\ \mathbf{0} & {}_{FEM}{}^i\mathbf{K}_{ind} \end{bmatrix} \begin{bmatrix} {}_{FEM}{}^i\mathbf{D}_{dep} \\ {}_{FEM}{}^i\mathbf{D}_{ind} \end{bmatrix} = \begin{bmatrix} {}_{FEM}{}^i\mathbf{F}_{R_{dep}} \\ {}_{FEM}{}^i\mathbf{F}_{R_{ind}} \end{bmatrix} \quad (C.16)$$

where *dep* and *ind* correspond to the dependent and the independent DOFs, respectively. Using a dependency matrix, denoted by ${}^i\mathbf{G}$, the dependent DOFs can be expressed in terms of the independent ones:

$${}_{FEM}{}^i\mathbf{D}_{dep} = \begin{matrix} {}^i\mathbf{G} \\ 24 \times 6 \end{matrix} \begin{matrix} {}_{FEM}{}^i\mathbf{D}_{ind}. \\ 6 \times 1 \end{matrix} \quad (C.17)$$

The derivation of ${}^i\mathbf{G}$ will be presented shortly. Since one must have equilibrium across the rigid element, the total work done must be zero. Therefore,

$${}_{FEM}{}^i\mathbf{D}_{dep}^T {}_{FEM}{}^i\mathbf{F}_{R_{dep}} - {}_{FEM}{}^i\mathbf{D}_{ind}^T {}_{FEM}{}^i\mathbf{F}_{R_{ind}} = 0. \quad (C.18)$$

By substituting Eq. (C.17) into Eq. (C.18) one can obtain

$${}_{FEM}{}^i\mathbf{F}_{R_{ind}} = {}^i\mathbf{G}^T {}_{FEM}{}^i\mathbf{F}_{R_{dep}}. \quad (C.19)$$

Furthermore, by substituting Eqs. (C.17) and (C.19) into Eq. (C.16), the following relationship could be developed to express the 6×6 stiffness matrix associated with the independent node of the platform element:

$${}_{FEM}^i \mathbf{K}_{ind} = {}^i \mathbf{G}^T {}_{FEM}^i \mathbf{K}_{dep} {}^i \mathbf{G}. \quad (\text{C.20})$$

Matrix ${}_{FEM}^i \mathbf{K}_{dep}$ is assembled using the MATLAB[®] code presented at the end of this appendix. Matrix ${}^i \mathbf{G}$ is derived by relating the translational and rotational DOFs associated with the independent node to the translational DOFs associated with the nodes on the moving platform, as illustrated by Fig. C.3. The term ${}_{FEM}^i \mathbf{P}_g$ represents the coordinates of the g^{th} node belonging to platform $(i+1)$ defined in the i^{th} coordinate frame.

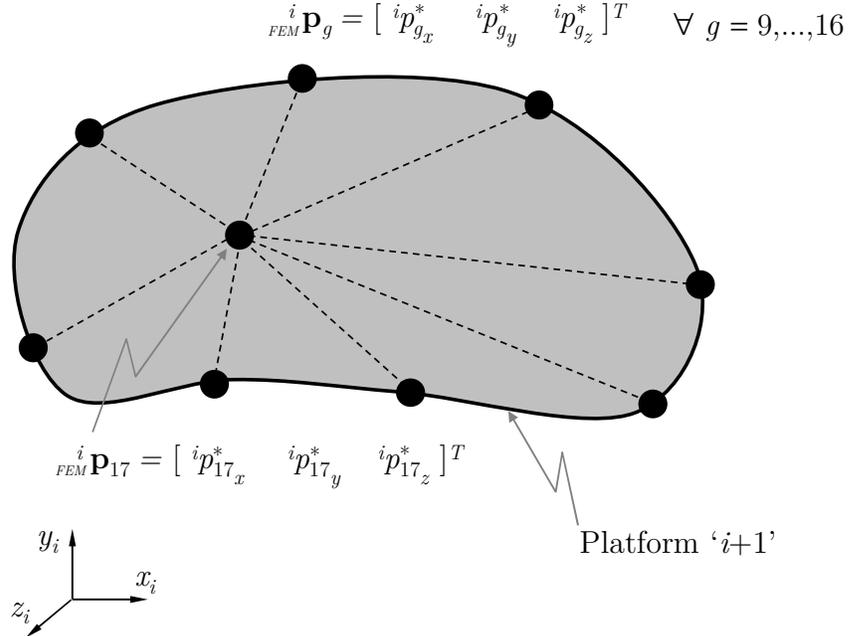


Figure C.3: Rigid element used to model the platform with the corresponding node numbering.

Vectors ${}_{FEM}{}^i\mathbf{D}_{dep}$ and ${}_{FEM}{}^i\mathbf{D}_{ind}$ from Eq. (C.17) can be expressed as

$${}_{FEM}{}^i\mathbf{D}_{dep} = \begin{bmatrix} {}_{FEM}{}^i\mathbf{d}_9 \\ \vdots \\ {}_{FEM}{}^i\mathbf{d}_{16} \end{bmatrix} \quad (\text{C.21})$$

where

$${}_{FEM}{}^i\mathbf{d}_g = \begin{bmatrix} {}_{FEM}{}^i d_{g_x} \\ {}_{FEM}{}^i d_{g_y} \\ {}_{FEM}{}^i d_{g_z} \end{bmatrix} = {}_{FEM}{}^i\mathbf{d}_{g_{trans}} + {}_{FEM}{}^i\mathbf{d}_{g_{rot}} \quad \forall g = 9, \dots, 16 \quad (\text{C.22})$$

and

$${}_{FEM}{}^i\mathbf{D}_{ind} = \begin{bmatrix} {}_{FEM}{}^i\mathbf{d}_{17} \\ {}_{FEM}{}^i\boldsymbol{\phi}_{17} \end{bmatrix} = \begin{bmatrix} [{}_{FEM}{}^i d_{17_x} & {}_{FEM}{}^i d_{17_y} & {}_{FEM}{}^i d_{17_z}]^T \\ [{}_{FEM}{}^i \phi_{17_x} & {}_{FEM}{}^i \phi_{17_y} & {}_{FEM}{}^i \phi_{17_z}]^T \end{bmatrix} \quad (\text{C.23})$$

where ${}_{FEM}{}^i d_{g_x}$, ${}_{FEM}{}^i d_{g_y}$, and ${}_{FEM}{}^i d_{g_z}$ represent the translational displacement of the g^{th} node, and ${}_{FEM}{}^i \phi_{g_x}$, ${}_{FEM}{}^i \phi_{g_y}$, and ${}_{FEM}{}^i \phi_{g_z}$ represent its rotational displacement. The translational displacements of nodes 9 to 17 have been broken up into a component due to the translational displacement of the independent node (node 17), denoted by ${}_{FEM}{}^i\mathbf{d}_{g_{trans}}$, and a component due to the rotational displacement of the independent node, denoted by ${}_{FEM}{}^i\mathbf{d}_{g_{rot}}$. Vector ${}_{FEM}{}^i\mathbf{d}_{g_{trans}}$ is simply equal to that of node 17:

$${}_{FEM}{}^i\mathbf{d}_{g_{trans}} = {}_{FEM}{}^i\mathbf{d}_{17} \quad (\text{C.24})$$

whereas vector ${}_{FEM}{}^i\mathbf{d}_{g_{rot}}$ can be related to ${}_{FEM}{}^i\boldsymbol{\phi}_{17}$ by assuming infinitesimal rotations:

$${}_{FEM}^i \mathbf{d}_{g_{rot}} = {}_{FEM}^i \tilde{\mathbf{p}}_g {}_{FEM}^i \Phi_{17} \quad (C.25)$$

where

$${}_{FEM}^i \tilde{\mathbf{p}}_g = \begin{bmatrix} 0 & ({}^i p_{g_z}^* - {}^i p_{17_z}^*) & -({}^i p_{g_y}^* - {}^i p_{17_y}^*) \\ -({}^i p_{g_z}^* - {}^i p_{17_z}^*) & 0 & ({}^i p_{g_x}^* - {}^i p_{17_x}^*) \\ ({}^i p_{g_y}^* - {}^i p_{17_y}^*) & -({}^i p_{g_x}^* - {}^i p_{17_x}^*) & 0 \end{bmatrix}. \quad (C.26)$$

By repeating the same procedure for every dependent node, the dependency matrix can be expressed as follows:

$${}^i \mathbf{G} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & {}_{FEM}^i \tilde{\mathbf{p}}_9 \\ \vdots & \vdots \\ \mathbf{I}_{3 \times 3} & {}_{FEM}^i \tilde{\mathbf{p}}_{16} \end{bmatrix}. \quad (C.27)$$

Using the above formulations for the limb and the platform elements, the global stiffness matrix is assembled using MATLAB[®], where the 6×6 Cartesian stiffness matrix is simply equal to ${}_{FEM}^i \mathbf{K}_{ind}$. The multi-module stiffness can be derived using the method presented in Sec. 5.3.

C.3 Stiffness Matrix Assembly

A short MATLAB[®] program was written to assemble the stiffness matrix of the VGW. The matrix assembly methodology has been adopted from [120].

```

1    %%% A. MOOSAVIAN, Mar 2014
2    clear; clc;
3
4    NE = 8;           % number of elements
5    NN = 2;          % number of nodes per element
6    ND = 3;          % number of DOFs per node
7    NNS = 17;        % total number of nodes
8    TD = NNS*ND;     % total DOFs
9
```

```

10 % node coordinates (module 1) [in]:
11 node(1,1)=11.15;   node(1,2)=2.708;   node(1,3)=0.75;
12 node(2,1)=-1.073; node(2,2)=-2.603;   node(2,3)=0.75;
13 node(3,1)=-0.21;  node(3,2)=-2.679;   node(3,3)=0.75;
14 node(4,1)=10.287; node(4,2)=2.782;   node(4,3)=0.75;
15 node(5,1)=11.895; node(5,2)=2.645;   node(5,3)=0.75;
16 node(6,1)=11.895; node(6,2)=-2.018;  node(6,3)=0.75;
17 node(7,1)=-1.819; node(7,2)=2.602;   node(7,3)=0.75;
18 node(8,1)=-1.819; node(8,2)=-2.538;  node(8,3)=0.75;
19
20 node(9,1)=12.32;  node(9,2)=-1.981;  node(9,3)=11.061;
21 node(10,1)=0.988; node(10,2)=2.4;      node(10,3)=11.061;
22 node(11,1)=11.469; node(11,2)=-2.141;  node(11,3)=11.061;
23 node(12,1)=1.849; node(12,2)=2.489;   node(12,3)=11.061;
24 node(13,1)=13.055; node(13,2)=2.406;   node(13,3)=11.061;
25 node(14,1)=13.055; node(14,2)=-1.843;  node(14,3)=11.061;
26 node(15,1)=0.244; node(15,2)=2.324;   node(15,3)=11.061;
27 node(16,1)=0.244; node(16,2)=-2.266;  node(16,3)=11.061;
28
29 node(17,1)=2.215;  node(17,2)=0;      node(17,3)=11.811;
30
31 % properties:
32 for i = 1:NE
33     Mod(i,1) = 1.1e07; % modulus of elasticity [psi]
34     Area(i,1) = 1;    % area [in^2]
35 end
36
37 % nodal connectivities:
38 El(1,1)=1; El(1,2)=9; % element 1 connects nodes 1 and 9
39 El(2,1)=2; El(2,2)=10; % element 2 connects nodes 2 and 10
40 El(3,1)=3; El(3,2)=11; % element 3 connects nodes 3 and 11
41 El(4,1)=4; El(4,2)=12; % element 4 connects nodes 4 and 12
42 El(5,1)=5; El(5,2)=13; % element 5 connects nodes 5 and 13
43 El(6,1)=6; El(6,2)=14; % element 6 connects nodes 6 and 14
44 El(7,1)=7; El(7,2)=15; % element 7 connects nodes 7 and 15
45 El(8,1)=8; El(8,2)=16; % element 8 connects nodes 8 and 16
46
47 % boundary conditions (the nodes on the stationary platform):
48 for i = 1:24
49     bcdof(i) = i;
50     bcval(i) = 0; %ith DOF constrained (with a value of 0)
51 end
52
53 % initiating the arrays:
54 ff = zeros(TD,1);
55 kk = zeros(TD);
56 index = zeros(NN*ND,1);
57
58 % element transformation and matrix construction:
59 for i = 1:NE
60     nd(1) = El(i,1);
61     nd(2) = El(i,2);
62
63     x1=node(nd(1),1);   y1=node(nd(1),2);   z1=node(nd(1),3);

```

```

64     x2=node(nd(2),1);   y2=node(nd(2),2);   z2=node(nd(2),3);
65
66     Leng = sqrt((x2-x1)^2+(y2-y1)^2+(z2-z1)^2);
67     Cx = (x2-x1)/Leng;   Cy=(y2-y1)/Leng;   Cz=(z2-z1)/Leng;
68
69     ED = NN*ND;
70     k = 0;
71
72     for i = 1:NN
73         start = (nd(i)-1)*ND;
74         for j = 1:ND
75             k = k+1;
76             index(k) = start+j;
77         end
78     end
79
80     k = (Area(i,1)*Mod(i,1)/Leng)*...
81     [Cx*Cx Cx*Cy Cx*Cz -Cx*Cx -Cx*Cy -Cx*Cz;...
82     Cx*Cy Cy*Cy Cy*Cz -Cx*Cy -Cy*Cy -Cy*Cz;...
83     Cx*Cz Cy*Cz Cz*Cz -Cx*Cz -Cy*Cz -Cz*Cz;...
84     -Cx*Cx -Cx*Cy -Cx*Cz Cx*Cx Cx*Cy Cx*Cz;...
85     -Cx*Cy -Cy*Cy -Cy*Cz Cx*Cy Cy*Cy Cy*Cz;...
86     -Cx*Cz -Cy*Cz -Cz*Cz Cx*Cz Cy*Cz Cz*Cz];
87     ED = length(index);
88
89     for i = 1:ED
90         ii = index(i);
91         for j = 1:ED
92             jj = index(j);
93             kk(ii,jj) = kk(ii,jj)+k(i,j);
94         end
95     end
96 end
97
98 % row and column elimination:
99 n = length(bcdof);
100 TD = size(kk);
101
102 for i = 1:n
103     c = bcdof(i);
104     for j = 1:TD
105         kk(c,j) = 0;
106     end
107     kk(c,c) = 1; % to avoid numerical difficulties
108     ff(c) = bcval(i);
109 end
110
111 % matrix reduction:
112 for i = 1:24
113     for j = 1:24
114         kk_red(i,j) = kk(i+24,j+24);
115     end
116 end
117

```

```

118 % generating the dependency matrix:
119 for i = 1:8
120     b_skew{i}=...
121         [0,(node(i+8,3)-node(17,3)),(node(17,2)-node(i+8,2));...
122          (node(17,3)-node(i+8,3)),0,(node(i+8,1)-node(17,1));...
123          (node(i+8,2)-node(17,2)),(node(17,1)-node(i+8,1)),0];
124     end
125
126 G = [eye(3) b_skew{1}; eye(3) b_skew{2}; eye(3) b_skew{3};...
127      eye(3) b_skew{4}; eye(3) b_skew{5}; eye(3) b_skew{6};...
128      eye(3) b_skew{7}; eye(3) b_skew{8}];
129
130 % generating the equivalent 6 x 6 matrix based on the FEM:
131 K = G'*kk_red*G;

```

Appendix D

Designing for Uncertainties

The presented wing morphing mechanism in this thesis has been designed to offer multiple degrees of fault tolerance in response to uncertainties. These design provisions can be categorized in response to two groups of uncertainties: 1) failure related, and 2) load related. The provisions for the first category respond to internal uncertainties, whereas those of the latter respond to external ones.

The points discussed and presented here are meant to serve as high-level provisions to accommodate the preliminary design of the VGW. Furthermore, the uncertainties are approached from a structural perspective in order to accommodate the conceptual design. Specific implementation of the detection provisions and the systems details will vary depending on the type of actuation and are beyond the scope of what is presented here.

D.1 Designing for Failure (Internal Uncertainties)

Failures can be grouped into two categories: 1) those that will affect the functionality of the system, and 2) those that will affect the safety of the system. Herein, the functional failures will be referred to as minor failures, whereas the safety failures will be referred to as major failures. Both minor and major failures are not considered catastrophic, as they would not prevent continued safe flight and landing.

D.1.1 Functional Failures

Functional failures can be contributed to possible systems failures, for example due to the failure of a valve or an electrical unit, or structural failures, such as the jamming of the movable skin. Any such event may lead to the jamming of a limb, causing loss of its functionality. However, the rigidity of the limb will not be affected due to such events. Furthermore, depending on the application, the system can be designed to be fail-jam or fail-operate. In the *fail-jam* case the system will no longer operate, but will be locked in its current position. In the *fail-operate* case the system will continue to operate after failure, but in a degraded mode, that is, with less controllable DOFs. The degraded operation mode is possible due to the redundant nature of the motion control, as discussed in Chapter 7.

The designation of fail-jam or fail-operate is entirely up to the particular application, as in general the VGW is meant for performance improvement, and not control. If the VGW is designated to be fail-jam, that is, no change of motion would be expected after a single failure, the response of the system would be the same regardless of the number of limbs jammed. In this case the VGW will maintain the position in which the first failure is detected until the problem is resolved. On the other hand, if the VGW is designated to be fail-operate, then the permanent locking of any actuated limb or a passive limb will reduce the controllable DOFs by the number of jammed limbs. As established in Chapter 4 each permanently locked member will impose a kinematic constraint, permanently reducing the number of controllable DOFs. However, the system can still operate in this mode, if required. If the system is to be designated as fail-operate, all possible scenarios must be anticipated during design and sizing so the safe operation of the system is not impacted.

D.1.2 Safety Failures

Safety failures occur when an actuated or a passive limb permanently loses its load-bearing capability. The optimal topology (Fig. 6.15) offers a single degree of tolerance in such an event. In Chapter 6, the optimization formulation was set up to capture the possible failure of any of the eight members. The configurations that were produced were referred to as fail-safe topologies. With the proposed optimal configuration, one can obtain eight of such topologies, all of which are illustrated in Fig. D.1.

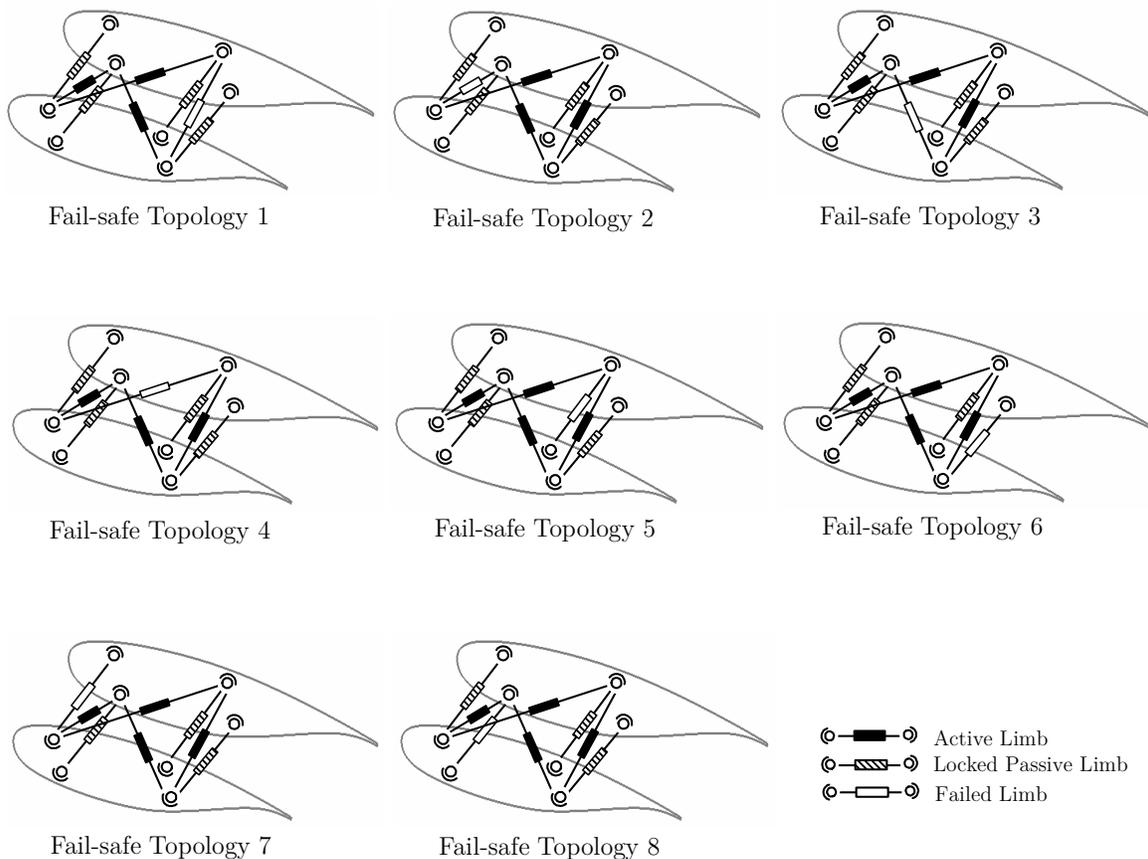


Figure D.1: Eight fail-safe topologies representing all possible one-member configurations that could be tolerated in the event one of the limbs loses its load-bearing capability permanently.

Although the system may still be able to operate in a degraded mode after the loss of the load-bearing capability of a limb, this is generally not recommended. Additionally, both in the case of functional (minor) and safety (major) failures, the detection provisions can be set up to annunciate the failure to the cockpit so the pilot can take the necessary measures.

D.2 Designing for Uncertainty in Loading (External Uncertainties)

As it was shown in the case-study in Chapter 7, the optimal motion can be determined for a given loading condition. However, any shift in the magnitude of the external loads caused by, for example, a sudden gust, can change the anticipated loading. Generally, during actuation there are two scenarios that could be taking place: 1) the actual internal loads will match those predicted, or 2) the actual internal loads deviate from the predicted loads. Depending on the fidelity and complexity of the on-the-fly internal load calculation required for the optimal motion control, some deviation from the predicted values may result. These deviations can be further amplified if the aircraft encounters sudden gust fields during actuation. Therefore, it is valuable to have some measures in place to deal with such uncertainties; just as valuable would be the autonomous application of such measures. In other words, it would be desirable to have a system in place that passively reacts to any uncertainties in loading without any additional sensors or active provisions.

This passive provision is addressed through the design of the passive members, which are directionally locked at all times. This is useful for restoring the required stiffness and providing the additional structural support when encountering uncertainties in external loading, which may alter the anticipated loading for the optimal motion control.

D.2.1 Passive Member Design

While actuation is occurring, a change in the external load may drive the actuators to reach their operational limits, after which they can only hold the applied load. Generally, a typical hydraulic or electro-mechanical actuator can support the additional load up to a structural limit, after which structural failure should be expected. In such a case, it would be desirable to reduce the load on the actuators by switching to the hyperstatic configuration. It would be highly beneficial to have a passively implemented system in place that could accomplish a similar effect without the complexity and potential reaction delay of an actively controlled one.

Although there exist flow control valves designed for conventional hydraulic systems, that is, those with a supply and a return, to the best of the author's knowledge no such system/package exists that can offer directional locking for a self-circulating hydraulic cylinder. Thus, a new type of valve is proposed here. The implementation of this type of valve has been carried out using a circuit with two two-way two-position solenoid valves and two check valves. The two positions of the solenoid valves include on and off, where in this setup the on position, that is, energized, corresponds to an open valve, and the off position, that is, de-energized, corresponds to a closed valve. Figure D.2 schematically illustrates the proposed circuit in the de-energized state.

There are four possible positions that could be generated with the proposed circuit. Table D.1 presents these positions. Terms *A* and *B* correspond to the two solenoid valves, while terms 1 and 0 represent the on and off states, respectively. The state in Table D.1 describes the functionality of the system for the corresponding position. The physical interpretation of each state in terms of its effect on the cylinder is described in Table D.2.

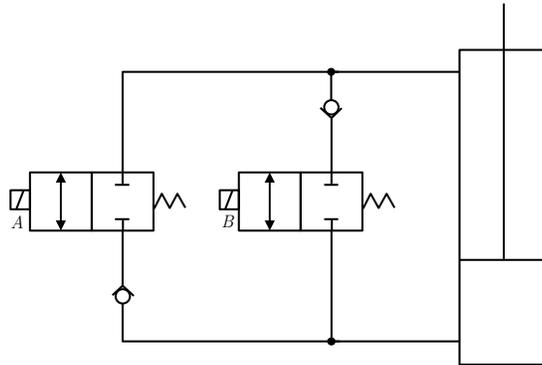


Figure D.2: Proposed circuit for directional flow control of a self-circulating cylinder.

Table D.1: Four positions generated using two solenoid valves.

Position	Solenoid Valve <i>A</i>	Solenoid Valve <i>B</i>	State
1	0	0	L_{E+R}
2	0	1	L_R
3	1	0	L_E
4	1	1	U_{E+R}

The first three positions are used in operation, while the fourth position is not. The fourth position provides the complete unlocking of the cylinder, which is useful for initial setup and calibration.

Table D.2: Physical interpretations for the four circuit configurations.

L_{E+R}	Locked in Extension; Locked in Retraction
L_R	Locked in Retraction; Unlocked in Extension
L_E	Locked in Extension; Unlocked in Retraction
U_{E+R}	Unlocked in Extension; Unlocked in Retraction

When a cylinder is locked in extension and retraction, i.e. position 1, it can react to loads both in tension and compression; when it is locked in retraction, i.e. position 2, it can only react to loads in compression; when it is locked in extension, i.e. position 3, it can only react to loads in tension; and when it is unlocked in extension and retraction, i.e. position 4, it cannot react to any loads.

The proposed circuit can effectively be represented as a two-way four-position valve, pictorially illustrated in Fig. D.3. The four positions are obtained through the actuation of the two solenoid valves.

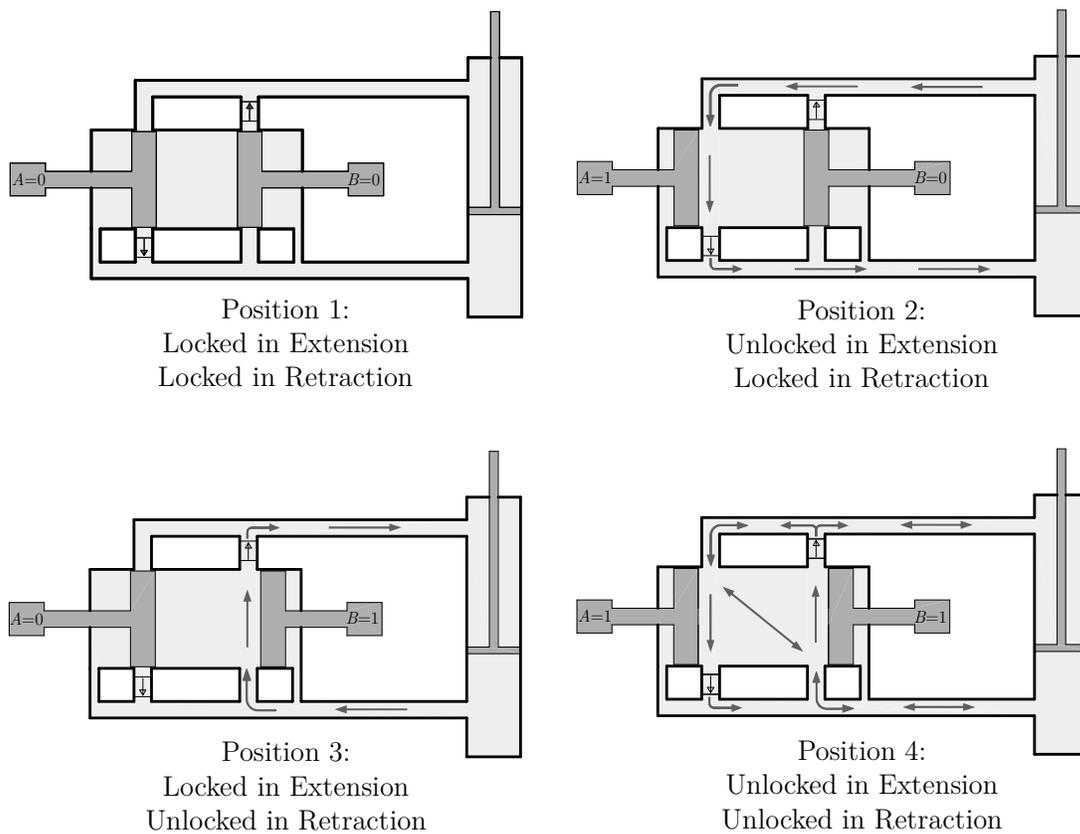


Figure D.3: Cutaway view of the proposed valve schematic in four positions to enable directional locking of a self-circulating cylinder.

With the current setup, if no power is supplied to the system, the passive members will remain locked in both directions. Figure D.4 illustrates the realization of the aforementioned circuit used on the prototype.

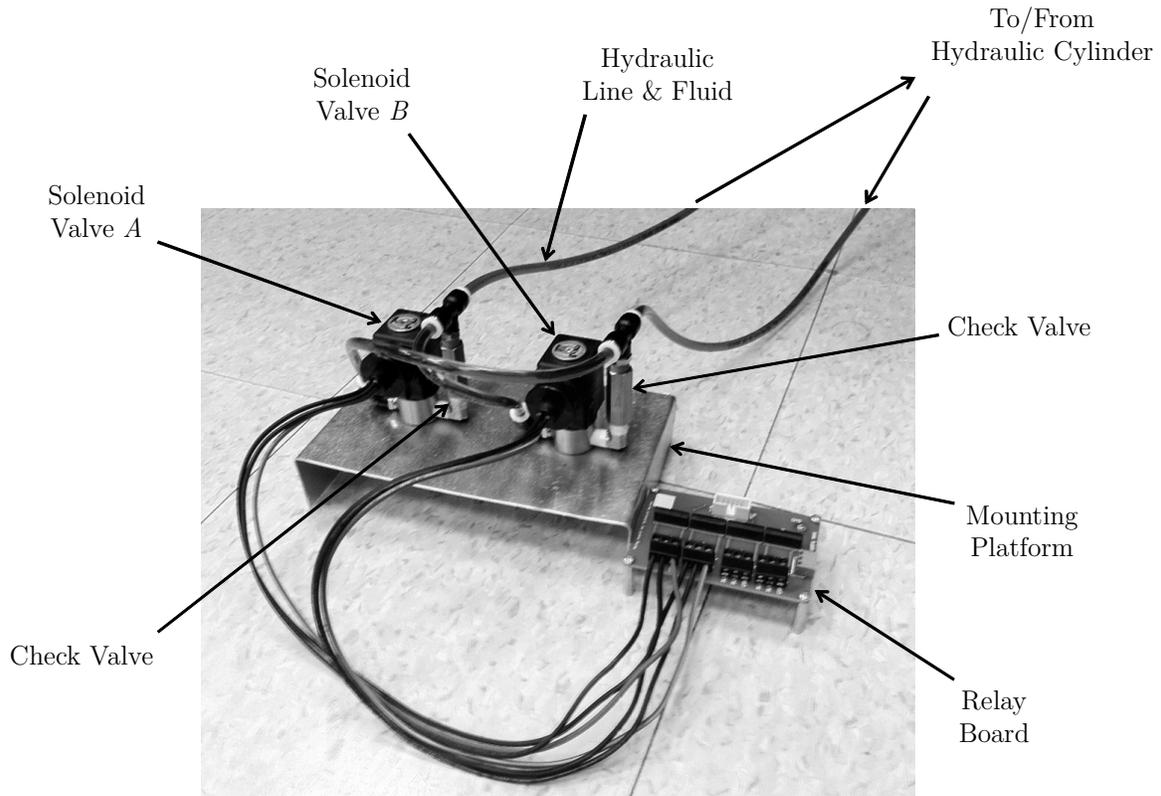


Figure D.4: The implementation of the directional lockability for one lockable cylinder.

Although this particular setup is only applicable to hydraulic systems, the same philosophy could be adopted to develop a mechanical setup using clutches and gears. Furthermore, it should be noted that the design of the VGW is not dependent on the ability of the passive members to offer directional lockability.

D.3 Sizing Guidelines

Both static and dynamic loads must be considered for the purpose of sizing the actuators, the lockable passive members, and the joints. Although this thesis is intended to provide a conceptual framework for the design of the VGW without much focus on the detailed design and sizing of these morphing modules, some general remarks about sizing are included here.

There are three scenarios that should be considered for sizing the actuators, the lockable passive members, and the joints: 1) ultimate loads applied to the hyperstatic (8-member) topology for all given poses for all flight and ground load-cases, 2) ultimate loads applied to the isostatic (6-member) topologies for all given poses for the pre-determined actuating flight and ground load-cases, and 3) limit loads applied to the fail-safe (7-member) topologies. *Ultimate* loads are equal to limit loads multiplied by an appropriate factor of safety, typically 1.5, while *limit* loads are the maximum loads to be expected in service (Federal Aviation Regulation (FAR) 25.301 [121]). It is recommended for the failures to be annunciated, so the pilot can take corrective action following the incident. The combination of all the above scenarios will provide minimum and maximum design loads for the actuators, the lockable passive members, and the joints, to be used for the purpose of structural sizing.

Although the proposed VGW is not intended for control, it is foreseen that it could be adopted for such an application. In such an event, the system should be sized/designed to adhere to the regulations of FAR 25.671 [121].

D.4 Conclusion

The incorporated features in the design can increase the robustness of the system by improving its ability to deal with uncertainties. The uncertainties discussed here

were broken up into those brought about by internal factors as well as those due to external factors. Figure D.5 summarizes these features.

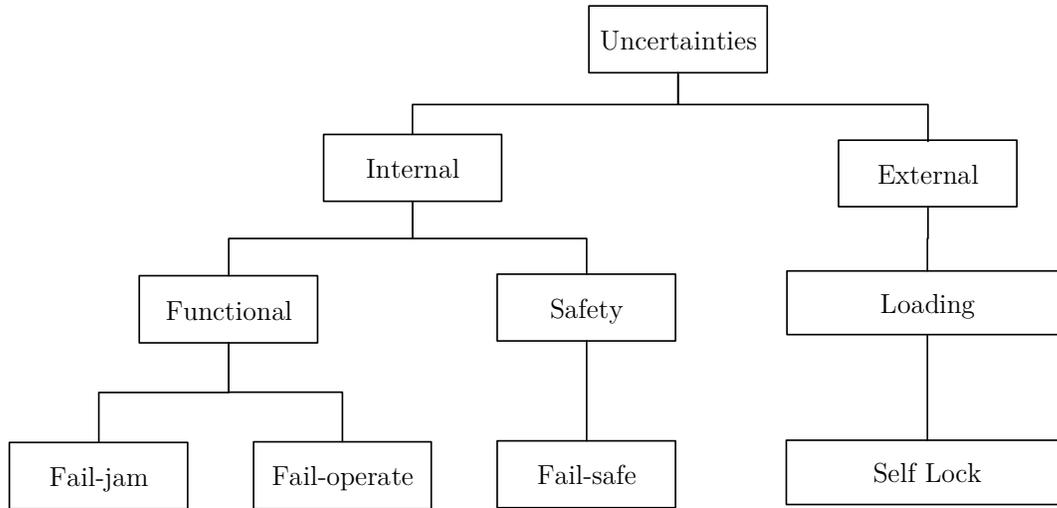


Figure D.5: Summary of approaches used in the VGW design to address internal and external uncertainties.

In the internal cases the system is set up to deal with safety failures through the fail-safe topological design of the VGW. Additionally, it can be set up to operate in a degraded mode in the case of a functional failure. In the external cases, the design of the passive members allows for a passive provision to be in place to automatically react to loads not anticipated during actuation.

Appendix E

SC(2)-0518 Airfoil Definition

The following table provides the definition for airfoil SC(2)-0518 (Fig. E.1) [119]:

Table E.1: SC(2)-0518 airfoil definition.

x/c	$(y/c)_u$	$(y/c)_l$	x/c	$(y/c)_u$	$(y/c)_l$	x/c	$(y/c)_u$	$(y/c)_l$
0.000	0.000	0.000	0.160	0.078	-0.079	0.340	0.090	-0.090
0.002	0.014	-0.014	0.170	0.080	-0.080	0.350	0.090	-0.090
0.005	0.021	-0.021	0.180	0.081	-0.081	0.360	0.090	-0.090
0.010	0.029	-0.029	0.190	0.082	-0.082	0.370	0.090	-0.090
0.020	0.039	-0.039	0.200	0.083	-0.083	0.380	0.090	-0.089
0.030	0.046	-0.046	0.210	0.084	-0.084	0.390	0.090	-0.089
0.040	0.051	-0.051	0.220	0.085	-0.085	0.400	0.090	-0.089
0.050	0.055	-0.055	0.230	0.085	-0.086	0.410	0.090	-0.088
0.060	0.059	-0.059	0.240	0.086	-0.086	0.420	0.090	-0.088
0.070	0.062	-0.062	0.250	0.087	-0.087	0.430	0.090	-0.087
0.080	0.064	-0.064	0.260	0.087	-0.088	0.440	0.089	-0.087
0.090	0.067	-0.067	0.270	0.088	-0.088	0.450	0.089	-0.086
0.100	0.069	-0.069	0.280	0.088	-0.088	0.460	0.089	-0.085
0.110	0.071	-0.071	0.290	0.089	-0.089	0.470	0.088	-0.084
0.120	0.073	-0.073	0.300	0.089	-0.089	0.480	0.088	-0.083
0.130	0.074	-0.074	0.310	0.089	-0.089	0.490	0.087	-0.082
0.140	0.076	-0.076	0.320	0.090	-0.090	0.500	0.087	-0.081
0.150	0.077	-0.077	0.330	0.090	-0.090	0.510	0.086	-0.079

x/c	$(y/c)_u$	$(y/c)_l$	x/c	$(y/c)_u$	$(y/c)_l$	x/c	$(y/c)_u$	$(y/c)_l$
0.520	0.085	-0.078	0.690	0.065	-0.042	0.860	0.033	-0.004
0.530	0.085	-0.076	0.700	0.063	-0.040	0.870	0.031	-0.003
0.540	0.084	-0.074	0.710	0.062	-0.037	0.880	0.028	-0.002
0.550	0.083	-0.073	0.720	0.060	-0.035	0.890	0.026	-0.001
0.560	0.082	-0.071	0.730	0.058	-0.032	0.900	0.024	0.000
0.570	0.081	-0.069	0.740	0.057	-0.030	0.910	0.022	0.001
0.580	0.080	-0.067	0.750	0.055	-0.028	0.920	0.019	0.001
0.590	0.079	-0.065	0.760	0.053	-0.025	0.930	0.017	0.001
0.600	0.078	-0.063	0.770	0.051	-0.023	0.940	0.015	0.001
0.610	0.076	-0.060	0.780	0.049	-0.021	0.950	0.012	0.000
0.620	0.075	-0.058	0.790	0.047	-0.018	0.960	0.009	-0.001
0.630	0.074	-0.056	0.800	0.045	-0.016	0.970	0.007	-0.002
0.640	0.072	-0.054	0.810	0.043	-0.014	0.980	0.004	-0.004
0.650	0.071	-0.051	0.820	0.041	-0.012	0.990	0.001	-0.006
0.660	0.070	-0.049	0.830	0.039	-0.010	1.000	-0.001	-0.008
0.670	0.068	-0.047	0.840	0.037	-0.008			
0.680	0.067	-0.044	0.850	0.035	-0.006			

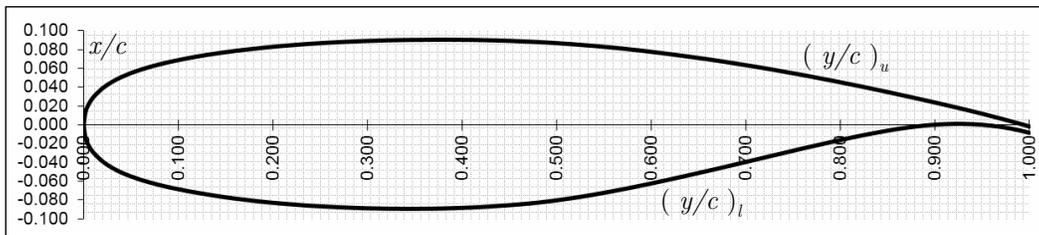


Figure E.1: Illustration of the SC(2)-0518 airfoil.

Bibliography

- [1] Wright, O., *How We Invented the Airplane: An Illustrated History*, Dover Publications, New York, 1988.
- [2] Reich, G., and Sanders, B., “Introduction to Morphing Aircraft Research,” *AIAA Journal of Aircraft*, Vol. 44, No. 4, pp. 1059-1059, 2007.
- [3] Ursache, N. M., Melin, T., Isikveren, A. T., and Friswell, M. I., “Morphing Winglets for Aircraft Multi-phase Improvement,” *Proceedings of the 7th AIAA Aviation Technology, Integration and Operations Conference*, Belfast, Northern Ireland, 2007.
- [4] Smith, D. D., Ajaj, R. M., Isikveren, A. T., and Friswell, M. I., “Multiobjective Optimization for the Multi-Phase Design of Active Polymorphing Wings,” *AIAA Journal of Aircraft*, Vol. 49, No. 4, pp. 1153-1160, 2012.
- [5] De Breuker, R., Abdalla, M. M., and Gürdal, Z., “A Generic Morphing Wing Analysis and Design Framework,” *Journal of Intelligent Material Systems and Structures*, Vol. 22, No. 10, pp. 1025-1039, 2011.
- [6] Neal, D. A., Good, M. G., Johnston, C. O., Robertshaw, H. H., Mason, W. H., and Inman, D. J., “Design and Wind-Tunnel Analysis of a Fully Adaptive Aircraft Configuration,” *Proceeding of 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Palm Springs, California, 2004.

- [7] Gamboa, P., Vale, J., Lau, F. J. P., and Suleman, A., "Optimization of a Morphing Wing Based on Coupled Aerodynamic and Structural Constraints," *AIAA Journal*, Vol. 47, No. 9, pp. 2087-2104, 2009.
- [8] Smith, D. D., Isikveren, A. T., Ajaj, R. M., and Firswell, M. I., "Multidisciplinary Design Optimization of an Active Nonplanar Polymorphing Wing," *Proceedings of 27th International Congress of the Aeronautical Sciences*, Nice, France, 2010.
- [9] Roth, B. D., and Crossley, W. A., "Application of Optimization Techniques in the Conceptual Design of Morphing Aircraft," *Proceedings of AIAA's 3rd Annual Aviation Technology, Integration, and Operations Conference*, Denver, Colorado, 2003.
- [10] Joshi, S. P., Tidwell, Z., Crossley, W. A., and Ramakrishnan, S., "Comparison of Morphing Wing Strategies Based Upon Aircraft Performance Impacts," *Proceedings of 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Palm Springs, California, 2004.
- [11] Neal, D. A., Farmer, J., and Inman, D., "Development of a Morphing Aircraft Model for Wind Tunnel Experimentation," *Proceedings of the 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Newport, Rhode Island, 2006.
- [12] Grant, D. T., Abdulrahim, M., and Lind, R., "Flight Dynamics of a Morphing Aircraft Utilizing Independent Multiple-joint Wing Sweep," *International Journal of Micro Air Vehicles*, Vol. 2, No.2, pp. 91-106, 2010.

- [13] Bharti, S., Frecker, M. I., Lesieutre, G., and Browne, J., "Tendon Actuated Cellular Mechanisms for Morphing Aircraft Wing," *Proceedings of SPIE*, Vol. 6523, pp. 1-13, 2007.
- [14] Mattioni, F., Weaver, P. M., Potter, K. D., and Friswell, M. I., "The Application of Thermally Induced Multistable Composites to Morphing Aircraft Structures," *Proceedings of SPIE*, Vol. 6930, pp. 12-1 - 12-11, 2008.
- [15] Andersen, G. R., Cowan, D. L., and Piatak, D. J., "Aeroelastic Modeling, Analysis and Testing of a Morphing Wing Structure," *Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu, Hawaii, 2007.
- [16] Flanagan, J. S., Strutzenberg, R. C., Myers, R. B., and Rodrian, J. E., "Development and Flight Testing of a Morphing Aircraft, the NextGen MFX-1," *Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu, Hawaii, 2007.
- [17] Bowman, J., Sanders, B., Cannon, B., Kudva, J., Joshi, S., and Weisshaar, T., "Development of Next Generation Morphing Aircraft Structures," *Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu, Hawaii, 2007.
- [18] Joo, J. J., Sanders, B., Johnson, T., and Frecker, M. I., "Optimal Actuator Location within a Morphing Wing Scissor Mechanism Configuration," *Proceedings of SPIE*, Vol. 6166, pp. 24-35, 2006.
- [19] Blondeau Samuel, J., and Pines, D., "Design and Testing of a Pneumatic Telescoping Wing for Unmanned Aerial Vehicles," *AIAA Journal of Aircraft*, Vol. 44, No.4, pp. 1088-1099, 2007.

- [20] Bae, J-S. , Seigler, T. M., and Inman, D. J., “Aerodynamic and Static Aeroelastic Characteristics of a Variable-Span Morphing Wing,” *AIAA Journal of Aircraft*, Vol. 42, No. 2, pp. 528-534, 2005.
- [21] Felício, J., Santos, P., Gamboa, P., and Silverstre, M., “Evaluation of a Variable-Span Morphing Wing for a Small UAV,” *Proceedings of the 52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Denver, Colorado, 2011.
- [22] Mestrinho, J., Gamboa, P., and Santos, P., “Design Optimization of a Variable-Span Morphing Wing for a Small UAV,” *Proceedings of the 52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Denver, Colorado, 2011.
- [23] Perry, B., Cole, S. R., and Miller, G. D., “Summary of an Active Flexible Wing Program,” *AIAA Journal of Aircraft*, Vol. 32, No. 1, pp. 10-15, 1995.
- [24] Chen, P. C., Sarhaddi, D., Jha, R., Liu, D. D., Griffin, K., and Yurkovich, R., “Variable Stiffness Spar Approach for Aircraft Maneuver Enhancement Using ASTROS,” *AIAA Journal of Aircraft*, Vol. 37, No. 5, pp. 865-871, 2000.
- [25] Garcia, H. M., Abdulrahim, M., and Lind, R., “Roll Control for a Micro Air Vehicle Using Active Wing Morphing,” *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Austin, Texas, 2003.
- [26] Stanford, B., Abdulrahim, M., Lind, R., and Ifju, P., “Investigation of Membrane Actuation for Roll Control of a Micro Air Vehicle,” *AIAA Journal of Aircraft*, Vol. 44, No. 3, pp. 741-749, 2007.

- [27] Abdulrahim, M., Garcia, H., and Lind, R., "Flight Characteristics of Shaping the Membrane Wing of a Micro Air Vehicle," *AIAA Journal of Aircraft*, Vol. 42, No. 1, pp. 131-137, 2005.
- [28] Guiler, R., and Huebsch, W., "Wind Tunnel Analysis of a Morphing Swept Wing Tailless Aircraft," *Proceedings of the 23rd AIAA Applied Aerodynamics Conference*, Toronto, Ontario, 2005.
- [29] Cooper, J. E., "Adaptive Stiffness Structures for Air Vehicle Drag Reduction," *Proceeding of the Multifunctional Structures/Integration of Sensors and Antennas Meeting*, Neuilly-sur-Seine, France, 2006.
- [30] Nam, C., Chattopadhyay, A., and Kim, Y., "Application of Shape Memory Alloy (SMA) Spars for Aircraft Maneuver Enhancement," *Proceedings of SPIE*, Vol. 4701, pp. 226-237, 2002.
- [31] Barrett, R., "Active Aeroelastic Tailoring of an Adaptive Flexspar Stabilator," *Smart Materials and Structures*, Vol. 5, No. 6, pp. 723-730, 1996.
- [32] Sofla, A. Y. N., Elzey, D. M., and Wadley, H. N. G., "Two-Way Antagonistic Shape Actuation Based on the One-Way Shape Memory Effect," *Journal of Intelligent Material Systems and Structures*, Vol. 19, No. 9, pp. 1017-1027, 2008.
- [33] Jacob, J. D., Simpson, A. D., and Smith, S. W., "Design and Flight Testing of Inflatable Wings with Wing Warping," SAE Technical Paper 2005-01-3392, 2005.
- [34] Lukens, J. M., Reich, G. W., and Sanders, B., "Wing Mechanization Design and Analysis for a Perching Micro Air Vehicle," *Proceeding of the 49th*

AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Schaumburg, Illinois, 2008.

- [35] Bourdin, P., Gatto, A., and Friswell, M. I., “Aircraft Control via Variable Cant-Angle Winglets,” *AIAA Journal of Aircraft*, Vol 45, No 2, pp. 414-423, 2008.
- [36] Manzo, J., and Ephraim, G., “Demonstration of an in situ Morphing Hyperelliptical Cambered Span Wing Mechanism,” *Smart Materials and Structures*, Vol. 19, No. 2, pp. 1-11, 2010.
- [37] Skillen, M. D., and Crossley, W. A., “Modeling and Optimization for Morphing Wing Concept Generation II, Part I: Morphing Wing Modeling and Structural Sizing Techniques,” NASA CR-2008-214902, 2008.
- [38] Abdulrahim, M., and Lind, R., “Flight Testing and Response Characteristics of a Variable Gull-Wing Morphing Aircraft,” *Proceeding of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, Rhode Island, 2004.
- [39] Ameri, N., Lowenberg, M. H., and Friswell, M. I., “Modelling the Dynamic Response of a Morphing Wing with Active Winglets,” *Proceedings of AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Hilton Head, South Carolina, 2007.
- [40] Wykes, J. H., and Mori, A. S., “XB-70 Aerodynamic, Geometric, Mass, and Symmetric Structural Model Data,” NASA CR-116773, 1970.
- [41] Ivanko, T. G., Scott, R. C., Love, M. H., Zink, P. S., and Weisshaar, T. A., “Validation of the Lockheed-Martin Morphing Concept with Wind Tunnel

- Testing,” *Proceedings of the 48th AIAA/ASME/ASCE/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu, Hawaii, 2007.
- [42] Weisshaar, T. A., “Morphing Aircraft Systems: Historical Perspectives and Future Challenges,” *AIAA Journal of Aircraft*, Vol. 50, No. 2, pp. 337-353, 2013.
- [43] Valasek, J., *Morphing Aerospace Vehicles and Structures*, John Wiley & Sons, West Sussex, United Kingdom, 2012.
- [44] Wroblewski, M. S., Henderson, J., and Bartley-Cho, J. D., “BAC 1-11 and MAW F-111 Control Surface Weight Estimation for SWIFT Study,” *Proceedings of the 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Austin, Texas, 2005.
- [45] Yenne, B., *The Great Warplanes of the 1990s*, Bison Books, London, United Kingdom, 1990.
- [46] Weber, L., *Fighting Jets:—America’s Frontline Fliers*, Publications International, Lincolnwood, Illinois, 1990.
- [47] Saltzman, E. J., Goecke, S. A., and Pembo, C., “Base Pressure Measurements on the XB-70 Airplane at Mach Numbers from 0.4 to 3.0,” NASA TM X-1612, 1968.
- [48] Jackson, R., *101 Great Bombers*, The Rosen Publishing Group, New York, New York, 2010.
- [49] Holle, A. A., and Judge, A. W., “Improvements in and Relating to Planes or the like for Aeroplanes,” UK Patent Specification No. 15290, 1916.

- [50] Elzey, D. M., Sofla, A. Y. N., and Wadley, H. N. G., "A Bio-Inspired, High-Authority Actuator for Shape Morphing Structures," *Proceedings of SPIE*, Vol. 5053, pp. 92-100, 2003.
- [51] Austin, F., Rossi, M. J., Nostrand, W. V., and Knowles, G., "Static Shape Control for Adaptive Wings," *AIAA Journal*, Vol. 32, No. 9, pp. 1895-1901, 1994.
- [52] Joo J. J., and Sanders, B., "Optimal Location of Distributed Actuators within an In-plane Multi-cell Morphing Mechanism," *Journal of Intelligent Material Systems and Structures*, Vol. 20, No. 4, pp. 481-492, 2009.
- [53] Strelec, J. K., Lagoudas, D. C., Khan M. A., and Yen, J., "Design and Implementation of a Shape Memory Alloy Actuated Reconfigurable Airfoil," *Journal of Intelligent Material Systems and Structures*, Vol. 14, No. 4-5, pp. 257-273, 2003.
- [54] Pinkerton, J. L., "A Feasibility Study to Control Airfoil Shape Using THUNDER," NASA TM-4767, 1997.
- [55] Smith, S. B., and Nelson, D. V., "Determination of the Aerodynamic Characteristics of the Mission Adaptive Wing," *AIAA Journal of Aircraft*, Vol. 27, No.11, pp. 950-958, 1990.
- [56] Kudva, J. N., Appa, K., Martin, C. A., Jardine, A. P., Sendekyj, G., Harris, T., McGowan, A-M., and Lake, R., "Design, Fabrication, and Testing of the DARPA / Wright Lab "Smart Wing" Wind Tunnel Model," *Proceedings of the 38th Structures, Structural Dynamics, and Materials Conference*, Kissimmee, Florida, 1997.

- [57] Elzey, D. M., Sofla, A. Y. N., and Wadley, H. N. G., "A Shape Memory-Based Multifunctional Structural Actuator Panel," *International Journal of Solids and Structures*, Vol. 42, No. 7, pp. 1943-1955, 2005.
- [58] Reich, G. W., Sanders, B., and Joo, J. J., "Development of Skins for Morphing Aircraft Applications via Topology Optimization," *Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu, Hawaii, 2007.
- [59] Murray, G, Gandhi, F., and Bakis, C., "Flexible Matrix Composite Skins for One-Dimensional Wing Morphing," *Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu, Hawaii, 2007.
- [60] Allen, L., "A Limber Future 'Smart Skin' Holds Promise for Morphing Wings and Wearable Computers," *Popular Science*, Vol. 265, No. 2, pp. 36-36, 2004.
- [61] McClung, A. J. W., Tandon, G. P., and Baur, J. W., "Deformation Rate-, Hold Time-, and Cycle-dependent Shape-memory Performance of Veriflex-E Resin," *Mechanics of Time-Dependent Materials*, Vol. 17, No. 1, pp. 39-52, 2013.
- [62] Sundaresan, V. B., and Leo, D. J., "Chemo-mechanical Model for Actuation Based on Biological Membranes," *Journal of Intelligent Material Systems and Structures*, Vol. 17, No. 10, pp. 863-870, 2006.
- [63] Barbarino, S., Bilgen, O., Ajaj, R. M., Friswell, M. I., and Inman, D., "A Review of Morphing Aircraft," *Journal of Intelligent Material Systems and Structures*, Vol. 22, No. 9, pp. 823-877, 2011.

- [64] Thill, C., Etches, J., Bond, I., Potter, K., and Weaver, P, "Morphing Skins," *The Aeronautical Journal*, Vol. 112, No. 1129, pp. 117-139, 2008.
- [65] Wlezien, R. W., Horner, G. C., McGowan, A-M. R., Padula, S. L., Scott, M. A., Silcox, R. J., and Harrison, J. S., "The Aircraft Morphing Program," *Proceedings of SPIE*, Vol. 3326, pp. 176-187, 1998.
- [66] McGowan, A-M. R., Cox, D. E., Lazos, B. S., Waszak, M. R., Raney, D. L., Siohci, E. J., and Pao, S. P., "Biologically-Inspired Technologies in NASA's Morphing Project," *Proceedings of SPIE*, Vol. 5051, pp. 1-13, 2003.
- [67] Amprikidis, M., and Cooper, J. E., "Development of Smart Spars for Active Aeroelastic Structures," *Proceedings of the 44th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Norfolk, Virginia, 2003.
- [68] Gough, V. E., "Contribution to Discussion to Papers on Research in Automobile Stability on Control and in Tyre Performance," *Proceedings of the Institution of Mechanical Engineers: Auto Division*, Vol. 171, pp. 392-395, 1956.
- [69] Stewart, D., "A Platform with six Degrees of Freedom," *Proceedings of the Institution of Mechanical Engineers*, Vol. 180, pp. 371-386, 1965.
- [70] Miura, K., and Furuya, H., "Variable Geometry Truss and Its Application to Deployable Truss and Space Crane Arm," *Acta Astronautica*, Vol. 12, No. 7-8, pp. 599-607, 1985.
- [71] Hughes, P. C., Sincarsin, W. G., and Carroll, K. A., "Trussarm - A Variable Geometry Truss Manipulator," *Journal of Intelligent Material Systems and Structures*, Vol. 2, No. 2, pp. 148-160, 1991.

- [72] Naccarato, F., and Hughes, P. C., “Inverse Kinematics of Variable Geometry Truss Manipulators,” *Journal of Robotic Systems*, Vol. 8, No. 2, pp. 249-266, 1991.
- [73] Finistauri, A. D., Xi, F., and Walsh, P., “Discretization Method for the Development of a Modular Morphing Wing,” *AIAA Journal of Aircraft*, Vol. 49, No. 1, pp. 116-125, 2012.
- [74] Moosavian, A., Xi, F., and Hashemi, S. M., “Design and Motion Control of Fully Variable Morphing Wings,” *AIAA Journal of Aircraft*, Vol. 50, No. 4, pp. 1189-1201, 2013.
- [75] Moosavian, A., Xi, F., and Hashemi, S. M., “Optimal Configuration Design for the Variable Geometry Wing-Box,” *AIAA Journal of Aircraft*, Vol. 51, No. 3, pp. 811-823, 2014.
- [76] Ramrakhiani, D. S., Lesieutre, G. A., Frecker, M., Bharti, S., “Aircraft Structural Morphing Using Tendon-Actuated Compliant Cellular Trusses,” *AIAA Journal of Aircraft*, Vol. 42, No. 6, pp. 1615-1621, 2005.
- [77] Inoyama, D., Sanders, B. P., and Joo, J. J., “Topology Synthesis of Distributed Actuation Systems for Morphing Wing Structures,” *AIAA Journal of Aircraft*, Vol. 44, No. 4, pp. 1205-1213, 2007.
- [78] Moored, K. W., and Bart-Smith, H., “The Analysis of Tensegrity Structures for the Design of a Morphing Wing,” *ASME Journal of Applied Mechanics*, Vol. 74, No. 4, pp. 668-676, 2007.
- [79] Steeves, C. A., Timpano, K. H., Maxwell, P. T., Martinelli, L., and Miles, R. B., “Design and Manufacture of a Morphing Structure for a Shape-Adaptive

- Supersonic Wind Tunnel Nozzle,” *ASME Journal of Applied Mechanics*, Vol. 76, No. 3, pp. 031012, 2009
- [80] Kota, S., Hetrick, J., and Saggere, L., “Design of Variable Stiffness Spar,” Aerospace Structure Information and Analysis Center (ASIAC) TR-97-01, 1997.
- [81] Mattioni, F., Gatto, A., Weaver, P. M., Friswell, M. I., and Potter, K. D., “The Application of Residual Stress Tailoring of Snap-through Composites for Variable Sweep Wings,” *Proceedings of the 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Newport, Rhode Island, 2006.
- [82] Xi, F., Xu, Y., and Xiong, G., “Design and Analysis of a Re-configurable Parallel Robot,” *Mechanism and Machine Theory*, Vol. 41, pp. 191-211, 2006.
- [83] Gan, D., Dai, J. S., and Liao, O., “Constraint Analysis on Mobility Change of a Novel Metamorphic Parallel Mechanism,” *Mechanism and Machine Theory*, Vol. 45, No. 12, pp. 1864-1876, 2010.
- [84] Aghili, F., and Parsa, K., “A Reconfigurable Robot with Lockable Cylindrical Joints,” *IEEE Transactions on Robotics*, Vol. 25, No. 4, pp. 785-797, 2009.
- [85] Bi, Z. M., and Kang, B., “Enhancement of Adaptability of Parallel Kinematic Machines with an Adjustable Platform,” *ASME Journal of Manufacturing Science and Engineering*, Vol. 132, No. 6, 2010.
- [86] Bi, Z. M., “Development and Control of a 5-axis Reconfigurable Machine Tool,” *Journal of Robotics*, Vol. 2011, 2011.

- [87] Xi, F., Li, Y., and Wang, H., "Module-Based Method for Design and Analysis of Reconfigurable Parallel Robots," *Frontiers of Mechanical Engineering*, Vol. 6, No. 2, pp. 151-159, 2011.
- [88] Grosch, P., Di Gregorio, R., Lopez, J., and Thomas, F., "Motion Planning for a Novel Reconfigurable Parallel Manipulator with Lockable Revolute Joints," *Proceedings of the 2010 IEEE International Conference on Robotic and Automation*, Anchorage, Alaska, 2010.
- [89] Pierrot F., "Parallel Mechanisms and Redundancy," *Proceedings of the 1st International Colloquium of the Collaborative Research Centre 562*, Braunschweig, Germany, 2002.
- [90] Moosavian, A., and Xi, F., "Design and Analysis of Reconfigurable Parallel Robots with Enhanced Stiffness," *Mechanism and Machine Theory*, Vol. 77, pp. 92-110, 2014.
- [91] Wang, J., and Gosselin, C. M., "Kinematic Analysis and Design of Kinematically Redundant Parallel Mechanisms," *ASME Journal of Mechanical Design*, Vol. 126, No. 1, pp. 109-118, 2004.
- [92] Mohamed, M. G., and Gosselin, C. M., "Design and Analysis of Kinematically Redundant Parallel Manipulators with Configurable Platforms," *IEEE Transactions on Robotics*, Vol. 21, No. 3, pp. 277-287, 2005.
- [93] Müller, A., "Redundant Actuation of Parallel Manipulators," in *Parallel Manipulators, Towards New Applications* (Edited by H. Wu), pp. 87-108, I-Tech Education and Publishing, Vienna, Austria, 2008.

- [94] O'Brien, J. F., and Wen, J. T., "Redundant Actuation for Improving Kinematic Manipulability," *Proceedings of the 1999 IEEE International Conference on Robotic and Automation*, Detroit, Michigan, 1999.
- [95] Park, F. C., and Kim, J. W., "Manipulability and Singularity Analysis of Multiple Robot Systems: A Geometric Approach," *Proceedings of the 1998 IEEE International Conference on Robotic and Automation*, Leuven, Belgium, 1998.
- [96] Bonev, I. A., Ryu, J., Kim, S-G., and Lee, S-K., "A Closed-form Solution to the Direct Kinematics of Nearly General Parallel Manipulators with Optimally Located Three Linear Extra Sensors," *IEEE Transactions on Robotics and Automation*, Vol. 17, No. 2, pp. 148-156, 2001.
- [97] Kutzbach, K., "Mechanische Leitungsverzweigung, ihre Gesetze und Anwendungen," *Maschinenbau Der Betrieb*, Vol. 8, No. 21, pp. 710-716, 1929.
- [98] Yang, H., Krut, S., Baradat, C., and Pierrot, F., "Locomotion Approach of REMORA: a Reconfigurable Mobile Robot for Manufacturing Applications," *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, California, 2011.
- [99] Bergerman, M., and Xu, Y., "Optimal Control of Manipulators with Any Number of Passive Joints," *Journal of Robotic Systems*, Vol. 15, No. 3, pp. 115-129, 1998.
- [100] Angeles, J., *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*, Springer, New York, New York, 2007.

- [101] Ma, O., and Angeles, J., “Architecture Singularities of Platform Manipulators,” *Proceedings of 1991 IEEE International Conference on Robotic and Automation*, Sacramento, California, 1991.
- [102] Grosch, P., Di Gregorio, R., and Thomas, F., “A One-motor Full-mobility 6-PUS Manipulator,” *ROMANSY 18, Robot Design, Dynamics, and Control, CISM International Centre for Mechanical Sciences*, Vol. 524, pp. 49-56, 2010.
- [103] Ball, R. S., *A Treatise on the Theory of Screws*, Cambridge University Press, Cambridge, United Kingdom, 1900.
- [104] Gosselin, C. M., “Stiffness Mapping for Parallel Manipulators,” *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 3, pp. 377-382, 1990.
- [105] Xi, F., Zhang, D., Mechefske, C., and Lang, S. Y. T., “Global Kinetostatic Modeling of Tripod-based Parallel Kinematic Machines,” *Mechanism and Machine Theory*, Vol. 39, No. 4, pp. 357-377, 2004.
- [106] Tahmasebi, F., and Tsai, L-W., “Jacobian and Stiffness Analysis of a Novel Class of Six-DOF Parallel Minimanipulators,” University of Maryland, Maryland, TR-1992-84, 1992.
- [107] Zhang, D., and Gosselin, C. M., “Kinetostatic Modeling of N-DOF Parallel Mechanisms with a Passive Constraining Leg and Prismatic Actuators,” *ASME Journal of Mechanical Design*, Vol. 123, pp. 375-381, 2001.
- [108] Maxwell, C., *Scientific Papers*, Vol. 2, Dover Publications, New York, New York, pp. 175-177, 1965.
- [109] Mitchell, A. G. M., “The Limits of Economy of Material in Frame Structures,” *Philosophical Magazine, Series 6*, Vol. 8, No. 47, pp. 589-595, 1904.

- [110] Schmit, L. A., and Farshi, B., "Some Approximation Concepts for Structural Synthesis," *AIAA Journal*, Vol. 12, No. 5, pp. 692-699, 1974.
- [111] Moosavian, A., Hashemi, S. M., and Xi, F., "Structural Optimization of a Primary Flight Control Assembly," *International Review of Aerospace Engineering*, Vol. 3, No. 4, pp. 192-207, 2010.
- [112] Stanford, B., Beran, P., and Kobayashi, M., "Simultaneous Topology Optimization of Membrane Wings and Their Compliant Flapping Mechanisms," *AIAA Journal*, Vol. 51, No. 6, pp.1431-1441, 2013.
- [113] Inoyama, D., Sanders, B. P., and Joo, J. J., "Topology Optimization Approach for the Determination of the Multiple-Configuration Morphing Wing Structure," *AIAA Journal of Aircraft*, Vol. 45, No. 6, pp. 1853-1862, 2008.
- [114] Maute, K., and Reich, G. W., "Integrated Multidisciplinary Topology Optimization Approach to Adaptive Wing Design," *AIAA Journal of Aircraft*, Vol. 43, No. 1, pp. 253-263, 2006.
- [115] Menon, C., Verthey, R., Markót, M. C., and Parenti-Castelli, V., "Geometrical Optimization of Parallel Mechanisms Based on Natural Frequency Evaluation: Application to a Spherical Mechanism for Future Space Applications," *IEEE Transactions on Robotics*, Vol. 25, No. 1, pp. 12-24, 2009.
- [116] Lee, J., Duffy, J., and Hunt, K. H., "A Practical Quality Index Based on the Octahedral Manipulator," *The International Journal of Robotics Research*, Vol. 17, No. 10, pp. 1081-1090, 1998.
- [117] Fattah, A., and Ghasemi, A. M. H., "Isotropic Design of Spatial Parallel Manipulators," *The International Journal of Robotics Research*, Vol. 21, No. 9, pp. 811-824, 2002.

- [118] Fung, Y. C., *An Introduction to the Theory of Aeroelasticity*, Dover Publications, New York, New York, 2002.
- [119] Harris, C. D., “NASA Supercritical Airfoils, A Matrix of Family-Related Airfoils,” NASA TP-2969, 1990.
- [120] Kwon, Y. W., and Bang, H., *The Finite Element Method Using MATLAB*, CRC Press, Boca Raton, Florida, 2000.
- [121] Federal Aviation Administration, “Federal Aviation Regulations - Part 25,” <<http://www.faa.gov>> [cited Jan 2014].