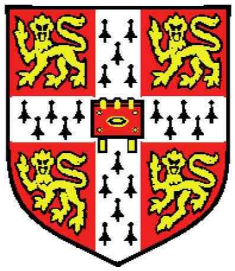
Polynomial Continuation in the Design of Deployable Structures

Andrew Viquerat

King's College



Department of Engineering

University of Cambridge

A thesis submitted for the degree of

Doctor of Philosophy

October 2011

For my parents, David and Maureen.

Acknowledgements

Most of all, I would like to thank my supervisor, Dr Simon Guest for his ability to immediately identify the core issues in any problem and to see new and exciting directions for research with ease, his willingness to hear out any and all of my ideas, no matter how implausible, and his painstaking attention to detail. My advisor, Dr Keith Seffen also deserves thanks for being a wonderful source of, and check-point for ideas.

I would like to thank the University of Cambridge, King's College Cambridge, and in particular, the King's College graduate community for providing a ready-made family away from home.

The Advanced Structures research group has provided a rich and diverse source of inspiration over the past three years. In particular I would like to thank Mark Schenk, Alex MacDonald, Richard Silveira, Richard Persaud, Mohammad-reza Golabchi and Arnaud Bonin.

I am very grateful for the financial support provided by King's College, the Cambridge Commonwealth Trust, and the University of Sydney Travelling Scholarships.

Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except where specifically indicated in the text. The research described in this dissertation was carried out in the Department of Engineering at the University of Cambridge between October 2008 and September 2011. This dissertation has not been previously submitted, in part or in whole, to any university or institution for any degree, diploma, or other qualification. This dissertation is presented in 186 pages, and contains fewer than 150 images, and fewer than 65000 words including all appendices and bibliography.

Abstract

Polynomial continuation, a branch of numerical continuation, has been applied to several primary problems in kinematic geometry. The objective of the research presented in this document was to explore the possible extensions of the application of polynomial continuation, especially in the field of deployable structure design. The power of polynomial continuation as a design tool lies in its ability to find **all** solutions of a system of polynomial equations (even positive dimensional solution sets). A linkage design problem posed in polynomial form can be made to yield every possible feasible outcome, many of which may never otherwise have been found.

Methods of polynomial continuation based design are illustrated here by way of various examples. In particular, the types of deployable structures which form planar *rings*, or *frames*, in their deployed configurations are used as design cases. Polynomial continuation is shown to be a powerful component of an equation-based design process.

A polyhedral homotopy method, particularly suited to solving problems in kinematics, was synthesised from several researchers' published continuation techniques, and augmented with modern, freely available mathematical computing algorithms. Special adaptations were made in the areas of level-k subface identification, lifting value balancing, and path-following. Techniques of forming closure/compatibility equations by direct use of symmetry, or by use of transfer matrices to enforce loop closure, were developed as appropriate for each example.

The geometry of a plane symmetric (rectangular) 6R foldable frame was examined and classified in terms of Denavit-Hartenberg Parameters. Its design parameters were then grouped into feasible and non-feasible regions, before continuation was used as a design tool; generating the design parameters required to build a foldable frame which meets certain configurational specifications.

Two further deployable ring/frame classes were then used as design cases: (a) rings which form (planar) regular polygons when deployed, and (b) rings which are doubly plane symmetric and planar when deployed. The governing equations used in the continuation design process are based on symmetry compatibility and transfer matrices respectively.

Finally, the 6, 7 and 8-link versions of N-loops were subjected to a witness set analysis, illustrating the way in which continuation can reveal the nature of the mobility of an unknown linkage.

Key features of the results are that polynomial continuation was able to provide complete sets of feasible options to a number of practical design problems, and also to reveal the nature of the mobility of a real overconstrained linkage.

General Keywords: 6R Linkage, Deployable ring, Deployable structures, Numerical continuation, Overconstrained mechanism, Polyhedral homotopy, Polynomial continuation.

Library of Congress Keywords: Continuation methods, Equations (Numerical solutions), Foldable structures, Homotopy theory, Kinematics.

CUED Library Keywords: Deployable structures, Kinematic geometry, Linkages, Polynomial continuation.

Contents

Nomenclature

Roman Symbols

a_{ij}	1) Denavit-Hartenberg bar length ($i, j = 1, \ldots, 6$) 2) element j of polynomial support i
d	witness set dimension
f	polynomial target system
g	polynomial start system
h	homotopy
$\mathbf{h}_i$	hinge vector i
J	Jacobian matrix
K	small displacement matrix of a closed loop
k_i	multiplicity of equation structure i
L	intersecting complex hyperplane
L_i	single axis rotation matrix ($i = 1, 2, 3$)
l_i	length of bar i
m_i	dim(Q_i)
M, M_i	pure translation matrix
N	number of bars in loop
n	number of variables or equations in system of polynomials
$\mathbf{n}_x$	normal vector to plane ($x = l, r$)

$\mathbf{p}_i$ hinge or point location

Q loop closure/compatibility equation

Q_i convex hull of equation i's support

R rotation matrix

r number of distinct polynomial structures present in system

R_i Denavit-Hartenberg hinge extension ($i, j = 1, \ldots, 6$)

t continuation parameter

T_{ij} transfer matrix from coordinates i to coordinates j

w_i lifting function for equation i

z_i extra variables in homotopy cascade system

Greek Symbols

α_i 1) 6-bar hinge tilt angle ($i = 1, 2$)
2) Reg.-polygon cross-section internal angles ($i = 1, 2, 3$)

α_{ij} Denavit-Hartenberg hinge angle ($i, j = 1, \ldots, 6$)

α_x hinge tilt angle ($x = a, b, c$)

β_i 6-bar hinge tilt angle ($i = 1, 2$)

δ_{xY} offset distance ($x = a, b, c$)

γ 6-bar design parameter

λ 1) number of spatial degrees of freedom
2) random complex number

ϕ general roll angle

ϕ_{ij} N-loop hinge twist between link i and j

ϕ_x hinge opening angle ($x = a, b, c$)

ψ general yaw angle

σ singular value

θ general pitch angle

θ_{ij} 6-bar hinge angle ($i, j = 1, \ldots, 6$)

Superscripts

$'$ 6-bar hinge tilt projection onto YZ plane

L geometric part of transfer matrix

Subscripts

B transfer matrix equation

g geometric part of transfer matrix

l left

R rotation matrix equation

r right

s 1) stowed location
2) 'state' part of matrix

1. Introduction

1.1 Deployable Structures

The term Deployable Structure (Gantes, 2001; Guest, 1994; Pellegrino, 2001) is used to describe a device which has at least two distinct states. A simple example is an umbrella. An umbrella has two distinct states (open and closed), and displays one of the most common design requirements of deployable structures in general, which is that in its inoperative state it is much more compact. The umbrella design in Figure 1.1 is a mechanism which consists of a single revolute/slider joint, and a number of planar sub-mechanisms (arranged radially), each with two bars connected by revolute joints. In deploying an umbrella, one introduces strain energy into the covering fabric, and then contains this energy by using a simple spring loaded catch. The energy gradient of the deployment process ensures that this state is stable. A number of typical requirements of

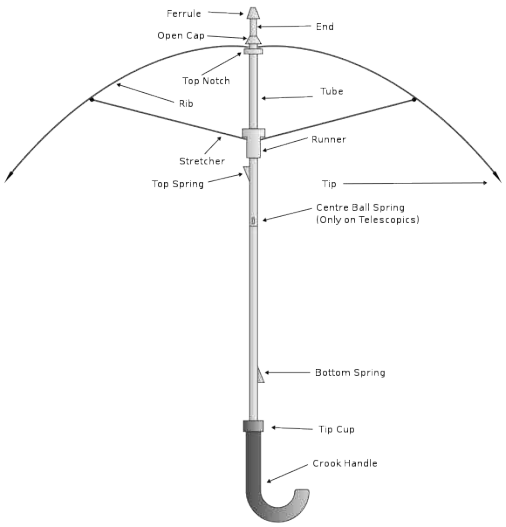


Figure 1.1: Umbrella: The most familiar deployable structure? (Licensed under the GFDL by the author)

a deployable structure are illustrated in this simple example. These include:

- structure has at least two distinct states;
  - stowed
  - deployed
- structure is typically more compact and robust in the stowed state;
- structure occupies a larger volume in the deployed state. This is usually the state in which the deployable structure is operational, although it might be more fragile;
- a smooth deployment path exists between the two states;
- the deployed state is stable (meaning the deployed shape does not need to be actively maintained);
- the structure's deployment is reliable and, sometimes, repeatable. This points to a need for simplicity in design.

Deployable structures have been used by engineers to solve problems in a variety of situations. Deployable structures have even been found to arise in nature (De Focatiis & Guest, 2002). Some of the most interesting applications are to be found in satellite design, in which deployable booms, masts, antennae, radars and solar arrays are commonplace. These deployable structures are launched in the stowed state, when volume minimisation requirements are of the greatest importance. Once in orbit, the structures can be deployed. The reliability of this deployment phase is of critical importance. Reliability can be increased by keeping a design as simple as possible, and also often by restricting any mechanisms involved to those with a single degree of freedom (SDOF). It is for this reason that a number of the linkages addressed in this document are SDOF.

Not all deployable structures are mechanisms in the traditional sense of the word. Some categories of structures are:

Telescopic Structures These are often used to deploy masts, antennae, or other devices of circular cross-section. They can be deployed using internal cables, threads, or even controlled gas expansion. See Figure 1.2.

Pantographic Structures A simple pantograph is a planar, scissor-like linkage whose motion is strictly linear. Pantographs of this type are often found attached to shaving mirrors in bathrooms. The basic linkage can be modified to form mechanisms such as the Hoberman Sphere and expandable "blob" structures (Jensen & Pellegrino, 2002, 2005), and extended into three dimensions to form pantographic

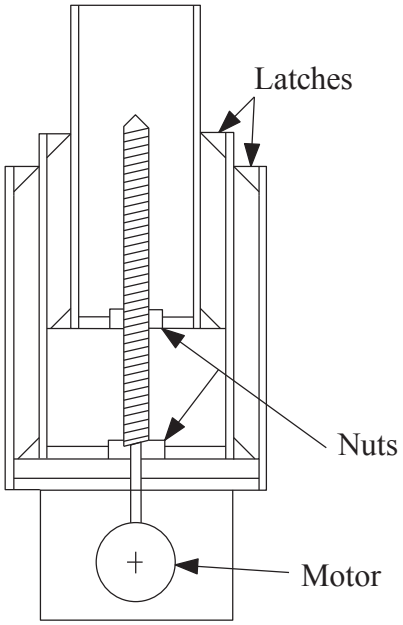


Figure 1.2: Telescopic Mast (reproduced from Pellegrino (1995)).

rings, or masts such as that shown in Figure 1.3. Further examples of pantographic structures can be found in Pellegrino & You (1993); You & Pellegrino (1997a,b). Similar base-level mechanisms called double chains have also been used to create mobile, closed loop structures (Mao *et al.*, 2009).

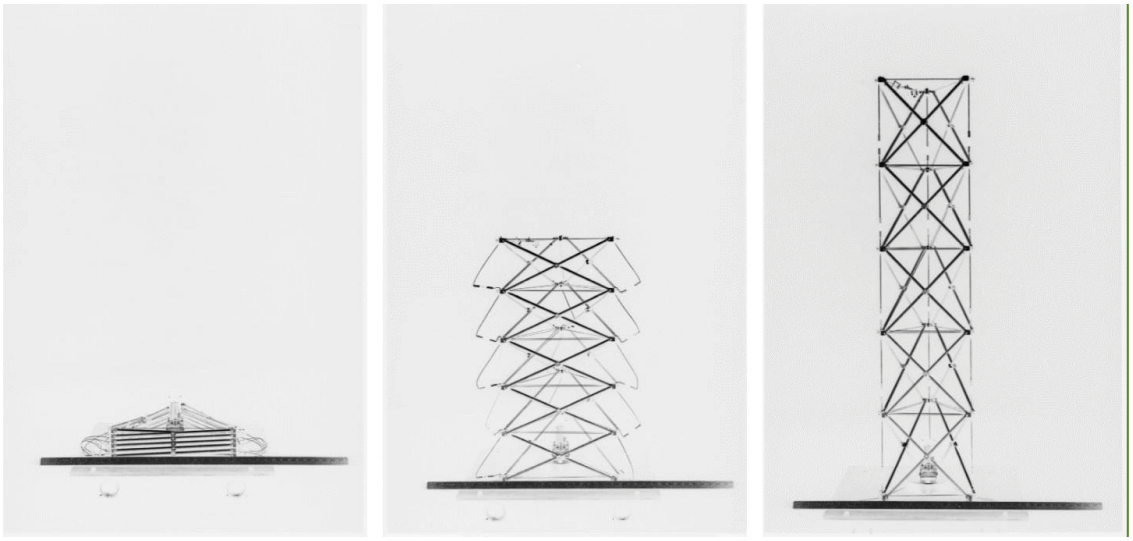


Figure 1.3: Pantographic mast developed by You.

Bistable Structures A common example of a bistable structure is a carpenter's tape

measure. Strips of metal, composite or plastic can be pre-stressed in such a way that the strip is stable when rolled up, and also when stretched out straight. A similar structure can be constructed without pre-stress by making use of composite material properties (an idea commercialised by a company called RolaTube, which makes masts using this concept). An example is shown in Figure 1.4. The dynamics of strips which use their own internal energy to deploy have been the topic of a good deal of research (Seffen & Pellegrino, 1999).

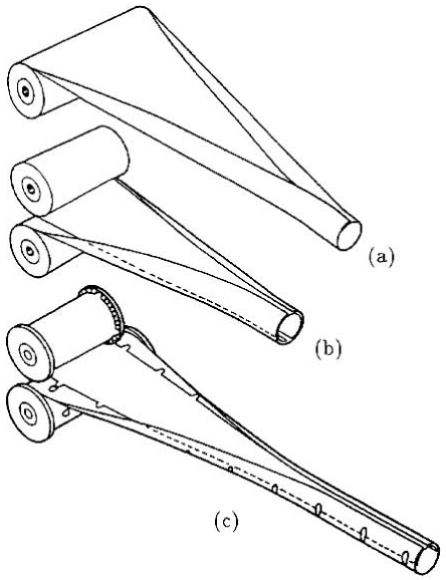


Figure 1.4: A bistable tube design. These are long thin strips which are stable both in a coiled and deployed state. (a) STEM, (b) bi-STEM, and (c) interlocking bi-STEM. The Storable Tubular Extendible Member^{TM} is produced by Northrop Grumman and has been used in space applications for over 30 years.

Inflatable Structures As the name suggests, inflatable structures (Fang *et al.*, 2004) are devices which in the stowed (deflated) state are thin membranes which can be easily and compactly stored, and can be inflated to fill a larger volume, and even take on some structural rigidity. A simple example is a balloon (Figure 1.5). A slightly more complicated example is JPL's inflatable SAR frame, shown in Figure 1.6. Sometimes inflatable structures are chemically *rigidized* once fully deployed.

Folding Frames and Rings This type of deployable structure is the primary feature of this document. The inflating structure in Figure 1.6 is technically a Frame, but cannot be said to be *folding*. Folding frames have been a focus of the Advanced Structures Laboratory at Cambridge for some time, and numerous versions have been produced there. A folding frame or Ring is typically stowed with some or all

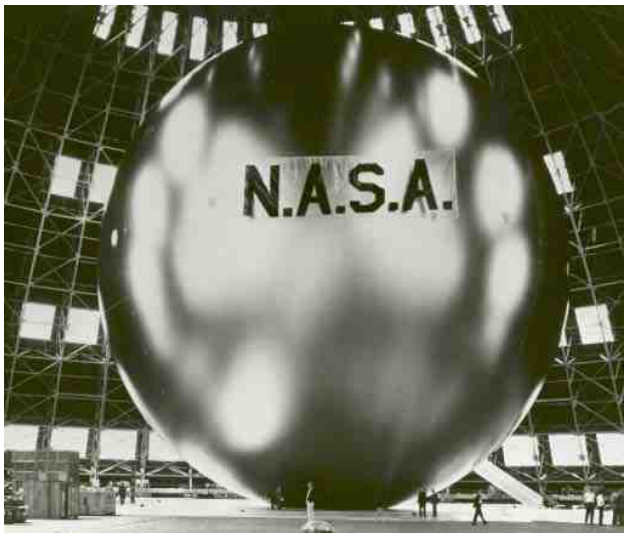


Figure 1.5: NASA's Echo II Balloon (which was rigidized when inflated).

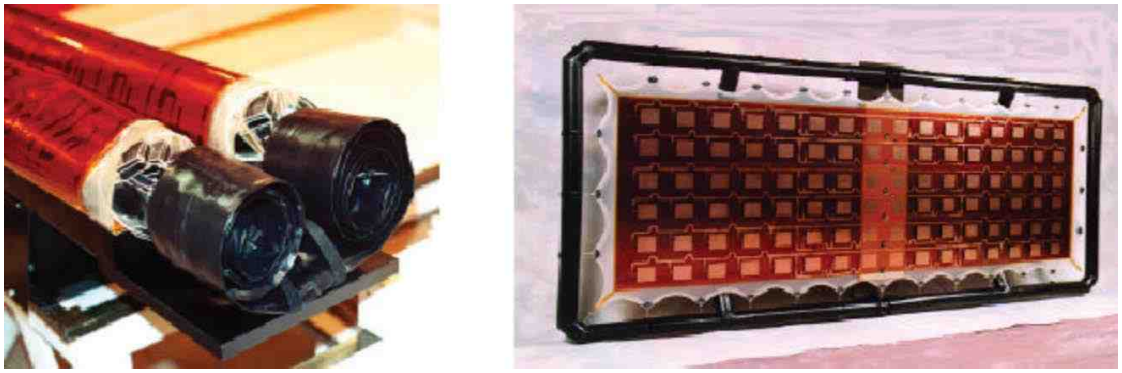


Figure 1.6: JPL's inflatable SAR frame. The flexible patch array and the inflatable booms are shown rolled up in the stowed state on the left.

bars parallel, and deployed to form a rigid supporting structure for a membrane. Frames or rings are closed loops, and are linkages which may have any number of kinematic degrees of freedom, although single degree of freedom closed loops are of particular interest. Some linkages are mobile, not because they are mechanisms in the true sense, but because they possess a Geometric Degree of Freedom (Chen *et al.*, 2004; Gan & Pellegrino, 2003; You, 2007). A subset of linkages with a geometric degree of freedom are known as Overconstrained Mechanisms (Cui & Dai, 2011; Mavroidis & Roth, 1994). An overconstrained mechanism has a degree of freedom or a mobility which arises because of some quirk of symmetry, or other special geometric property (see Section 3.2); it has a higher mobility than

that predicted by the Chebychev-Grübler-Kutzbach criterion. An example of the deployment of such an overconstrained mechanism is given in Figure 1.7. Deployable frames have been proposed for use in space applications (Chen & You, 2006a,b) (in particular panel and antenna deployment (Pellegrino *et al.*, 2000)).

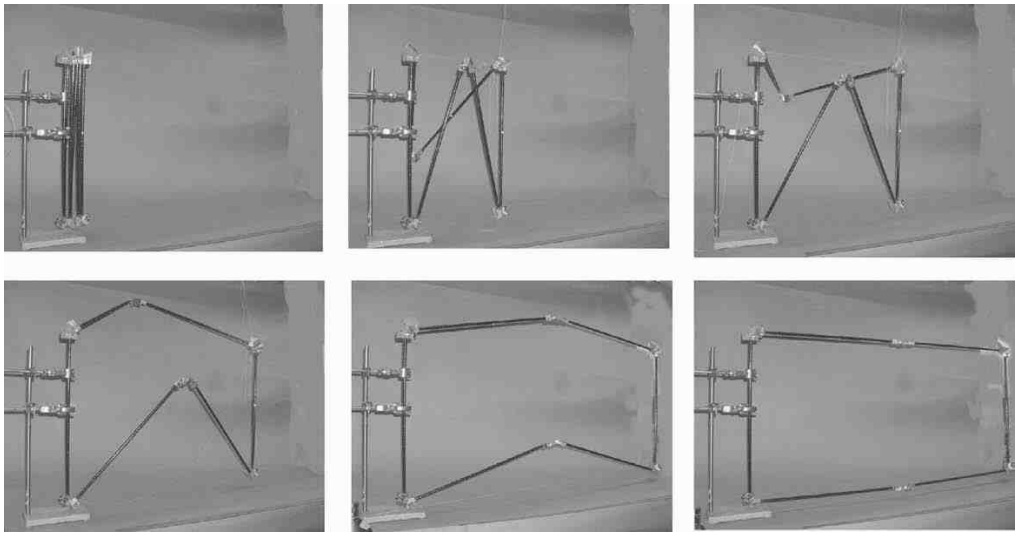


Figure 1.7: A single degree of freedom, 6-bar foldable ring deploying, from Gan & Pellegrino (2006).

The study of deployable structures spans a broad range of disciplines, and incorporates a variety of structures, however, the focus of this document is in the area of folding rings and frames.

1.2 Design Techniques

Both mechanism simulation and design are addressed in the chapters which follow. This section outlines some of the many techniques used to date to synthesise and design deployable structures.

1.2.1 Standard Kinematic Synthesis Techniques

The design of mechanisms was originally an inherently creative task. Designers would use a trial and error approach until a satisfactory linkage or mechanism was found. The need for reliable linkages which could perform a prescribed function with some accuracy increased with the onset of the industrial revolution. For some time the most sought after linkage was a planar, straight-line linkage, which could be used to guide machinery along

a linear path. James Watt developed a linkage (sometimes known as Watt's linkage, or the parallel linkage) in 1784, the central point of which traces out what is very nearly a straight line over a limited range of motion. Watt used this linkage to constrain the motion of his steam driven pistons, preventing them from binding with their cylinders. The Watt linkage is still used today in the suspension of some cars. Other linkages capable of approximating straight line motion soon appeared. Examples are the Chebyshev linkage, the Hoekens linkage (which converts rotational motion into approximate straight-line motion), and the Evans linkage (Bryant & Sangwin, 2008). In 1853, the Sarrus linkage was invented. The Sarrus linkage is a spatial linkage which converts circular motion (over a limited range) into linear motion. If manufactured very carefully, the top of the linkage traces out a perfectly straight line. It was not until several years after the invention of Watt's linkage that the first planar perfect straight-line linkage appeared. The Peaucellier-Lipkin linkage (Demaine & O'Rourke, 2007), was invented in 1864 (and again in 1871 independently), essentially by accident. Hart's linkage, another perfect straight-line mechanism, was invented in 1875, and requires only five links, compared to the Peaucellier-Lipkin's seven.

Not all linkages are required to follow a straight line. Sometimes linkages which approximate a prescribed curve are required. Given some basic specifications about desired mobility, it may be possible to methodically determine the types of mechanisms able to perform the desired task (Freudenstein & Maki, 1979; Read & Wilson, 1998), but it is almost always impossible to synthesise the exact geometry from any kind of recipe. This said, very simple mechanisms can be designed in such a way that a point on the mechanism, or a coupler link, can be made to closely follow a given algebraic curve. An atlas of 4-bar linkages was given in Hrones & Nelson (1951), which allowed the designer to manipulate an existing planar 4-bar design in such a way that it more closely followed a desired curve.

In the 20th century, the procedure for kinematic synthesis became more formalised (Hartenberg & Denavit, 1964). The three standard stages of mechanism design are type, number and dimensional synthesis. In type synthesis, one identifies the mechanical component best suited to producing the type of motion required. In number synthesis, the number of degrees of freedom is determined. In dimensional synthesis, the exact lengths/angles/sizes of the mechanism components are determined. Dimensional synthesis forms the primary topic of this document.

Until fairly recently, accurate simulation of the motion of all but the most elementary mechanisms was very difficult. The curves traced out by even the simplest linkages can still be very complicated. The simulation and design of more complicated linkages has

become possible only with the arrival of computer based numerical analysis and simulation. Prior to that, linkage synthesis involved a combination of intuition, experimentation, and luck.

1.2.2 Perfect Planar Mechanisms

Knowing that it is possible to construct linkages capable of tracing out a perfectly straight line, one might also wonder if it is possible to design linkages capable of tracing out more complicated, algebraic curves. Kempe's Universality Theorem (Hopcroft *et al.*, 1984; Kempe, 1877) states that it is possible to construct a planar mechanism such that one of the mechanism's vertices traces out a specified parametric curve, no matter how high the degree of the curve. The design of such mechanisms can even be automated (Gao *et al.*, 2001). Kempe's Universality Theorem can also be generalised to higher dimensions (Abbott & Barton, 2004). The linkages which are generated by such automated processes tend towards impracticality, with even the most simple curves requiring a great number of links. In general, the number of links required to trace an algebraic plane curve of degree n is O(n^4).

1.2.3 Optimisation

Optimisation techniques lend themselves to the task of refining an already established solution concept, rather than top level design. In linkage design, that might mean that the number of links, the types of joints, and the connectivity (the way in which the links are connected to one another) have already been decided. An optimisation step could be included to adjust the lengths of the links, the location of the linkage anchors or perhaps the dimensions of the bars to be used, to more closely match the design constraints.

Essentially all optimisation techniques require the specification of an *objective* function, which is then maximised or minimised, subject to certain constraints. Some examples of features which one might include in an objective function when designing linkages, with particular reference to deployable structures, are:

- proximity of a coupler link to a series of prescribed points (sometimes referred to as Precision Points) at the position of greatest proximity throughout the range of the linkage's motion. This is known as *point-path synthesis*, and examples are given in figures 1.8(a) and 1.8(b);
- maximum volume of space required during deployment;

- maximum stretch in a membrane mounted to a folding frame during deployment;
- maximum velocity of certain joints during deployment;
- magnitude of the smallest singular value in a linkage during deployment (this might be useful in designing a bifurcation-free mechanism).

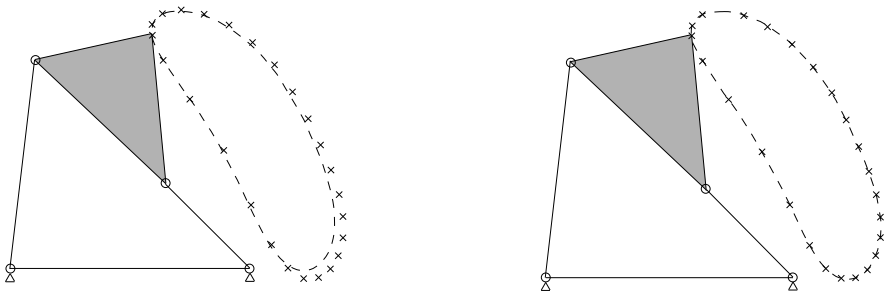
Some of the most powerful optimisation methods require computation of the gradients of the objective function with respect to the various design variables. Examples of numerical gradient-based methods are line searches, the Newton-Raphson method, and non-linear least squares (Angeles *et al.*, 1988). Sometimes, if enough care is taken, gradients can be obtained analytically, making optimisation more precise (Sancibrian *et al.*, 2004). If gradients are not readily available, one might use a quasi-Newton method (Ge *et al.*, 1999), trust regions, pattern searches (Krishnamurty & Turcic (1992) uses a Hooke-Jeeves approach to linkage design), or more exotic approaches such as an *ant-gradient search* (Diab & Smaili, 2008), or a technique known as simulated annealing (Martínez-Alfaro, 2007).

The optimisation methods mentioned above vary in the extent to which they rely on a high quality start point in design variable space. The numerical gradient-based techniques are perhaps the most sensitive to the choice of start point, but have the advantage of performing very well when in proximity to a minimum or maximum of the objective function. All optimisation methods require a point at which to begin the search for a local minimum/maximum of the objective function, which complicates the task of a linkage designer. In using optimisation, a designer will initially have to rely on his or her intuition to find a feasible start point in design variable space close enough to the optimum design to allow the optimiser to converge.

Figures 1.8(a) and 1.8(b) illustrate an example in which the dimensions of a 4-bar planar mechanism are adjusted to match a set of precision points as closely as possible (a collaborative optimisation architecture (Braun & Kroo, 1995) was used to generate the examples shown).

1.2.4 Direct Solution of Closure/Compatibility Equations

A further approach to mechanism design involves first expressing a kinematic problem as a series of Closure Equations or Compatibility Equations (which ensure the proper formation and motion of the mechanism), as well as a collection of Constraint Equations which specify something about the way in which the linkage should behave. The optimisation methods of Section 1.2.3 are typically used in situations in which there are



(a) Initial 4-bar configuration showing vertex locus.

(b) Final 4-bar configuration showing vertex locus.

Figure 1.8: An example of the optimisation of a 4-bar planar linkage to closely match a set of precision points. The lengths of the bars have been adjusted using an optimiser.

a greater number of constraints and design targets than design variables. The coupler curve-fitting example of Figures 1.8(a) and 1.8(b) illustrates a situation in which there are a great many more design targets (precision points) than there are design variables (bar lengths). An optimiser will approximate the design targets within operational limits, while obeying any constraints which have been specified. In trying to directly solve a system of equations which specify a certain design problem, one generally aims to work with a *square* system; that is, a system in which there are as many equations (constraints) as variables. That is not to say that one need always deal with a strictly square system, as under-determined systems can provide very interesting curves and manifolds in solution space which reveal a lot about the nature of a problem.

More often than not (especially in kinematics) a square system of closure, compatibility and constraint equations will have more than a single solution. In fact, a feature which makes the direct solution of closure/compatibility equations such an attractive approach to mechanism or linkage design is that **sometimes a number of different designs will satisfy the same requirements**. Using an *ad hoc* or optimisation method will most likely only locate a single solution to a kinematic problem (if any is found at all).

An example of the way in which a design procedure might flow is given below:

1. select design requirements (such as choosing a number of precision points);
2. select an appropriate type of mechanism and set the number of links and their connectivity;
3. determine which dimensions/angles are to be used as design variables;
4. construct a system of equations which describes the motion of the mechanism;

5. solve these equations for the design variables.

The final step in the procedure above involves the solution of a set of non-linear equations (general solution methods for these kinds of equations are covered in Hoffmann & Vermeer (1995); Nielsen (1997)). The approach described above does require an educated guess about the type of mechanism required to perform the task, the number of links, and the connectivity of the links (note, however, that Emiris & Moroz (2011) uses the direct solution of non-linear compatibility equations to enumerate all possible connectivities for a given linkage type, negating the need to pre-specify a connectivity).

Symbolic and Eigenvalue Techniques

A large number of non-linear equations which arise in kinematics can be converted to *polynomial form*. Once a system of non-linear equations is in polynomial form, some standard solution techniques can be applied. Such methods include the use of Gröbner Bases to solve equations exactly, using substitution to reduce a system of polynomials to a single equation (Lee & Liang, 1988) (which could then be solved as an eigenvalue problem (Chtcherba & Kapur, 2004)), and Galois Theory (Owen, 1991).

Method of Resultants

One particularly powerful technique involves the use of *resultants*. A good background on the use of resultants as they apply to kinematics is given in Nielsen (1997), while the mathematics (resultants of the Sylvester type) are covered in Chtcherba & Kapur (2004) and Sturmfels & Zelevinsky (1994). Applications of result based methods to kinematics and structural problems are given in Emiris (1994); Emiris & Mourrain (1996), and more famously in Lee & Liang (1988). At their core, resultant techniques are formalised strategies for performing elimination. Slightly different approaches exist for reducing systems of multivariate polynomials, but all involve the construction of a matrix system equivalent to the original polynomials, and the evaluation of a determinant which reduces the number of variables involved. While very powerful in the treatment of small systems, the method of resultants is difficult to formalise for all sizes and types of polynomial systems, which leads to an insufficient degree of automation for a generally applicable method.

Polynomial Continuation

Of key interest in this document is a method called Numerical Continuation, and in particular, Polynomial Continuation (Allgower & Georg, 2003; Zulehner, 1988). Continua-

tion is a numerical technique in which the solutions of a system of polynomial equations are found by way of a second, similar system of equations whose solutions are already known. By incrementally transforming the system whose solutions are known into the original 'target' system, the unknown solutions can be found. Numerical continuation methods can be applied to a variety of equation families, but polynomials have certain properties which allow for the simplification of the continuation procedure. In particular, it is possible to construct very tight bounds on the number of solutions which may theoretically arise in a given set of equations. The solutions of a system of polynomial equations with real coefficients are a mixture of *solutions at infinity*, complex conjugate pairs and real solutions (Erdos & Grunwald, 1939). It is the real solutions which are typically of greatest interest in linkage design (Luo & Dai, 2007; Wampler *et al.*, 1990) because they describe the physically realisable linkages. Note that there is no way of predicting how many of the solutions of a system of polynomials will fall into each of these categories (at infinity, complex or real) before the continuation process has been completed. Polynomial continuation has been applied to a variety of kinematic problems in the past (Soeseno, 1990; Verschelde, 2011), and constitutes the principal topic of investigation for a number of researchers. Typically, applications involve the solution of inverse kinematics (Chablat & Angeles, 2003).

Some of the major examples of the use of polynomial continuation in the solving of problems in kinematics are:

Computing camera motion from a sequence of images An inverse kinematics problem explored in the Ph.D. thesis Emiris (1994).

Predicting molecule structure An exercise in "finding energetically favourable configurations"[1] in which the shape of a molecule is determined mathematically (Emiris, 1994; Emiris & Mourrain, 1996).

Stewart platform kinematics Sometimes called a *left hand* problem, a Stewart platform is a parallel manipulator in which a static base plate is attached to a mobile top plate by six rods of variable length. The problem of finding the orientation and position of the top plate, or the lengths of the rods as a function of the top plate's orientation and position is a classic problem in non-linear equation solving. Stewart platforms have featured in a number of analyses based on continuation methods (Emiris, 1994; Raghavan & Roth, 1993; Sommese *et al.*, 2002a).

Number of assembly modes of a linkage It is possible to enumerate all the possible

[1] from Emiris & Mourrain (1996)

connectivities of a linkage, given a number of links and a number of nodes (Emiris & Moroz, 2011).

Inverse kinematics of general 6R manipulators Another classic problem in inverse kinematics frequently adressed using continuation. A typical task is to establish all possible linkage configurations given a manipulator end location and orientation (Manocha & Canny, 1994; Raghavan & Roth, 1993; Wampler & Morgan, 1993).

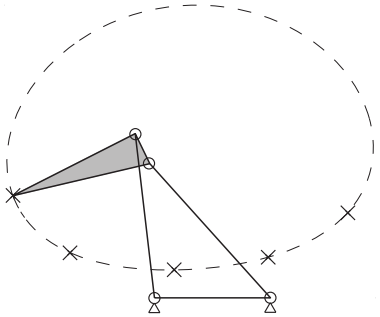
Planar 4-bar point-path synthesis This was mentioned briefly in section 1.2.3 and will be addressed in the form of an example below. 4-bar point-path synthesis was an early 'demonstrator' problem in polynomial continuation which has been considered by a number of authors (Morgan & Wampler, 1990; Subbian, 1990; Subbian & Flugrad, 1991; Wampler *et al.*, 1992).

Spatial body-guidance problem Sometimes called a Burmester problem, the body-guidance problem involves the use of inverse kinematics to design a linkage based on a sequence of pre-specified body locations and orientations (Sommese *et al.*, 2002a; Wampler *et al.*, 1990).

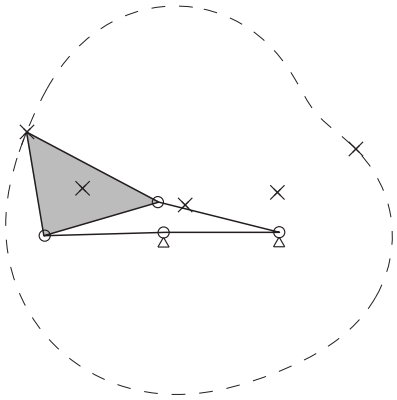
For an example of polynomial continuation as applied to a problem in kinematics, return now to the point-path synthesis for a planar 4-bar linkage. It is not specified that the coupler link need follow a particular continuous curve, but only that it pass through a collection of precision points at various stages in its range of motion. In Section 1.2.1 it was described how a curve approximating a desired path could be generated using an atlas of 4-bar designs. In Section 1.2.2 it was stated that a linkage following a given algebraic curve could always be found, but that the linkage would most likely be intractably complex. In Section 1.2.3, a precision point example involving a large number of precision points and a reasonably good start point in the design space was used to illustrate linkage optimisation. In the current point-path synthesis it is required that the coupler link of the 4-bar linkage to be designed must pass **precisely** through each of the precision points. In order for this to be possible, a square system must be constructed; so only as many precision points as there are design variables may be specified. Figures 1.9(a) to 1.9(c) show three different solutions arising from the specification of five precision points (requiring the use of five independent design variables in the form of bar lengths). The only pre-specifications for the linkage are that it must be a planar 4-bar, and that the coupler link must be opposite the anchored link. Note that the order in which the precision points are hit cannot be stipulated. The length and orientation of the anchored link are also pre-specified in this example for simplicity.

4-bar point-path synthesis is a classic application of continuation to the design of mechanisms. The solutions shown are, in fact, not the only solutions to this particular set of precision points, but they illustrate the way in which several different designs can satisfy the same requirements. Interestingly, the number of solutions arising from the specification of any number of precision points in planar 4-bar mechanism design is always a multiple of three; every mechanism appears together with its two Roberts' cognates. This example highlights the difference between an optimisation approach (Figures 1.8(a) and 1.8(b)), and a direct solution approach. No start point (initial design) was required to generate the solutions, and a designer using this technique will be presented with a number of solutions to the specified problem, some of which he or she will possibly not have conceived. Planar 4-bar design problems of this nature permit a maximum of nine design variables (four bar lengths, two coupler side lengths, two translational parameters and one rotation) for exact solution. This means that it is possible to solve exactly for nine precision points (Wampler *et al.*, 1992).

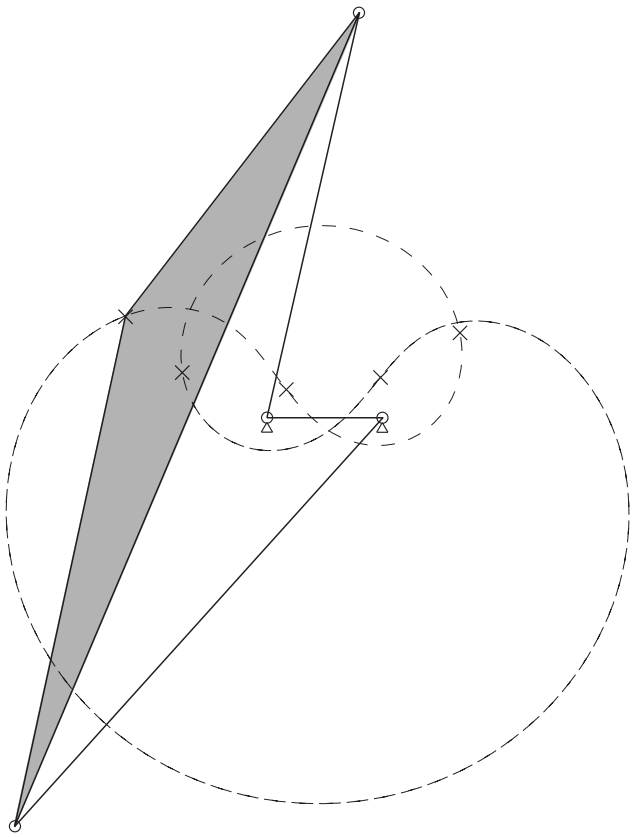
A more comprehensive introduction to polynomial continuation is given in Chapter 2.



(a) 4-bar mechanism showing vertex locus, Example 1.



(b) 4-bar mechanism showing vertex locus, Example 2. Note that solution is degenerate.



(c) 4-bar mechanism showing vertex locus, Example 3.

Figure 1.9: Examples of 4-bar linkages arising from the direct solution of their loop closure equations. The same set of precision points produced all of the examples shown.

1.3 Chapter Summary

Chapter 2: Introduction to Polynomial Continuation A branch of numerical continuation known as polynomial continuation has been applied to numerous problems in kinematics and linkage design for over twenty years. Polynomial continuation is a broadly applicable mathematical tool, however, the majority of the applications found in this and proceeding chapters will concern only the application of polynomial continuation to linkage design problems.

This chapter provides a background in the basics of continuation, including: specification of homotopies; path-following and predictor-corrector methods; homogenisation and Bézout numbers, and the generation of multihomogeneous start systems; Polyhedral Homotopy methods, including the calculation of mixed volumes, optimising lifting values and the creation of minimum volume start systems; witness sets and the reduction of systems of polynomials into their irreducible components.

At the end of the chapter, an illustrative example, in the form of a hinge design problem, is presented.

Chapter 3: 6R Plane Symmetric Frames The plane symmetric 6R foldable frame is identified as a case of the plane symmetric Bricard linkage in terms of Denavit-Hartenberg Parameters; the nature of its single degree of freedom is examined in detail by determining the exact structure of the system of equations governing its movement; a range of design parameters for building feasible mechanisms is determined numerically; and polynomial continuation is used to design frames with certain specified desirable properties.

Chapter 4: Regular-Polygonal Foldable Rings A regular-polygonal deployable ring is designed using polynomial continuation. When deployed, the ring forms a regular polygon with an even number of sides, and when stowed, forms a close packed bundle of parallel bars. The ring's motion is characterised using a pair of compatibility equations, and these equations are used to design rings whose deployment occurs in a prescribed manner. Various combinations of design variables are used to design the ring from different perspectives, illustrating the versatile way in which continuation can be used to solve real linkage problems.

Chapter 5: Doubly Symmetric 8-Bar Foldable Rings This is, in some ways, a generalisation of the design case in Chapter 4. The requirement for the deployed ring to be a regular polygon is relaxed. Rings with a double plane symmetry when deployed are required, for volume minimisation purposes, to fold up into a bundle

when stowed. Rather than a pair of compatibility equations, a matrix equation, based on a transfer matrix closure equation approach, is used to design rings which satisfy certain design specifications. In a further departure from Chapter 4, the majority of these specifications take the form of a precisely regulated stowed shape, rather than a particular deployment path.

Chapter 6: Mobility of N-Loops Chapter 3 includes an example of how Witness Sets can be used to understand the structure of the closure or compatibility equations defining the kinematics of a linkage. In this chapter, an extended example of a similar nature is given, with a greater emphasis on the implications of the structure of the equations for the mobility of the linkage.

N-loops, made up of closed loops of 90° arc shaped links are studied, with particular reference to the 6, 7 and 8-link variants. The 6-link variant was found to have the most interesting closure equation structure, with direct implications for the linkage's mobility. The degree of the irreducible components governing the mobility of the 7 and 8-link variants were determined.

2. Polynomial Continuation

Polynomial continuation is used extensively throughout this thesis. A handful of software packages are available for research purposes; most notably PHCpack (Guan & Verschelde, 2008; Verschelde, 1999), HOM4PS-2.0 (Lee *et al.*, 2008) and HOMPACK (Watson *et al.*, 1987). For maximum versatility, all software used in this thesis was written entirely by the author in Matlab, using a variety of literature sources as the basis of the underlying algorithm.

2.1 Polynomial Continuation Basics

A branch of Numerical Continuation known as Polynomial Continuation has been applied to various problems in linkage design over a period of more than two decades. Polynomial continuation is a method of finding the roots of a system of polynomials.

2.1.1 The Degree of a System of Polynomial Equations

Just as a polynomial equation has a *degree*, so too does a system of polynomials. Equation 2.1 is a 4^{th} degree polynomial in x, y and z. The degree, or order, of the whole equation is determined by the term in the equation which has the highest degree.

$$x^2yz + 2y^2 - 5z = 0 \tag{2.1}$$

The system of Equation 2.2 has degree 8, which is simply the product of the degrees of the constituent equations. According to the fundamental theorem of algebra, Equation 2.2 has **at most** eight solutions.

$$\begin{aligned} x^2yz + 2y^2 - 5z &= 0 \\ 3xy + z - 2 &= 0 \\ 2x - y + z &= 0 \end{aligned} \tag{2.2}$$

Methods to solve systems like 2.2 symbolically are available. Software packages such as Matlab, Maple and Mathematica can be used to solve simple systems exactly, but these packages falter when the degree of a system becomes too large. In addition, not all systems of polynomials can be solved symbolically, as discussed in Section 2.2 of Lamure & Michelucci (1996).

2.1.2 Types of Solutions of Polynomial Equations

There are a number of ways to classify the types of solutions which arise in polynomial systems. Firstly, a solution may be contained within the finite plane, or it may be a Solution at Infinity. Within this, solutions may be further categorised as:

- Geometrically Isolated;
- Positive Dimensional (continuous solution set which arises when a solution is not geometrically isolated, could be overlapping lines etc.);
- non-singular (determinate of Jacobian of system $\neq 0$).

Note that being non-singular implies a solution is geometrically isolated, but being geometrically isolated does not imply non-singular. A graphical representation of this is given in Figure 2.1. (in computational analysis, Singular Solutions present greater numer-

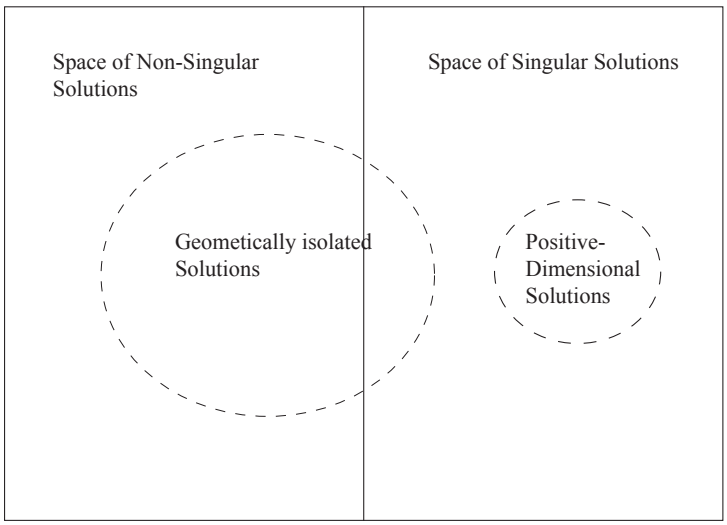


Figure 2.1: Polynomial solution/root types.

ical difficulties than non-singular, and *end games*, which deal specifically with singular solutions, have become an area of study unto themselves (Leykin *et al.*, 2000)).

Some simple examples of polynomial systems, each of which illustrates one of the solution types mentioned above, have been placed in Table 2.1. Each system has degree 2. In the first example, the line and the parabola clearly intersect at two geometrically isolated locations. Both solutions are non-singular. The second example illustrates a single (geometrically isolated, non-singular) solution in the finite plane. The second solution is said to be at infinity. The third example illustrates a geometrically isolated, but singular solution. It is said to be a solution with a multiplicity of 2. No examples of Positive Dimensional solution sets have been given, but one could be created by multiplying any of the second equations in Table 2.1 by a factor of $x^2 - y$: a second overlapping parabola with a continuum of solutions along its entire length would be introduced. More will be said about positive dimensional solution sets later in the chapter.

To understand better what is meant by a solution at infinity, consider the following system:

$$f(x,y) = \begin{bmatrix} x^2 + y^2 - 7 \\ x^2 + y^2 - 14y + 30 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{2.3}$$

Next, consider what happens when $x \to X/p$ and $y \to Y/p$, and all denominators are cleared:

$$f(X,Y,p) = \begin{bmatrix} X^2 + Y^2 - 7p^2 \\ X^2 + Y^2 - 14Yp + 30p^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

and send $p \to 0$. What is left is called the Homogenised version of f, sometimes abbreviated as f^0.

$$f^0(X,Y) = \begin{bmatrix} X^2 + Y^2 \\ X^2 + Y^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The solutions of f^0 are called the 'solutions at infinity.' Note that any scalar multiple of the solutions at infinity will also be a solution to f^0. Conventionally, one searches for solutions of the form $\{1, Y\}$ or $\{0, Y\}$. In this case, the solutions at infinity are at $\{1, \pm i\}$. The additional two solutions to this 4^{th} degree problem are at $\{\pm 0.1237, 2.6429\}$.

Not all systems have solutions at infinity, sometimes despite first appearances. Consider:

$$f(x,y) = \begin{bmatrix} 3xy - 2x^2 + 2y - 7 \\ x^2 + y^2 - 4x + 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

with

$$f^0(X,Y) = \begin{bmatrix} 3XY - 2X^2 \\ X^2 + Y^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

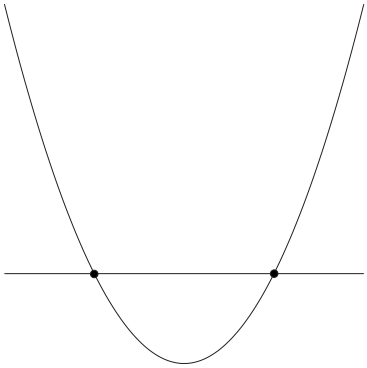
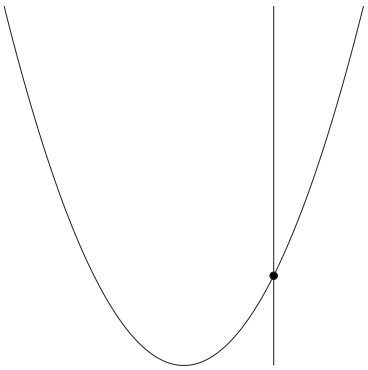
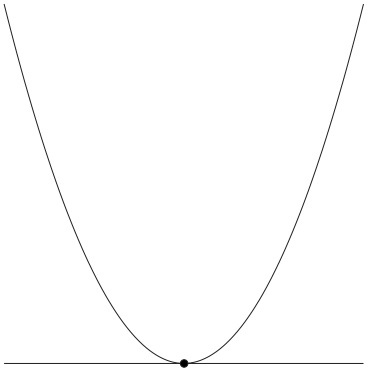
	1) $x^2 - y = 0$ $y - 1 = 0$ Two non-singular solutions in the finite plane.
	2) $x^2 - y = 0$ $x - 1 = 0$ One solution at ∞. One non-singular in finite plane.
	3) $x^2 - y = 0$ $y = 0$ One singular solution in finite plane with multiplicity 2.

Table 2.1: Simple polynomial systems.

which has only the four finite solutions $\{-0.7001 \pm 0.8908i, -1.0218 \mp 2.3539i\}$ and $\{1.6232 \pm 0.8837i, 1.6372 \pm 0.2039i\}$.

2.1.3 The 'Circle' Problem

The examples discussed above can easily be solved algebraically. To describe how continuation works, consider the following motivational problem: find the equation of a circle which passes through three prescribed points. This centuries old problem can be solved geometrically, algebraically (somewhat laborious) or via numerical optimisation if a reasonable initial guess is available. The general form of the equation describing a circle is given by 2.4.

$$(x-a)^2 + (y-b)^2 = r^2 \tag{2.4}$$

This is depicted in Figure 2.2. Once the three points p_j, have been specified, the problem

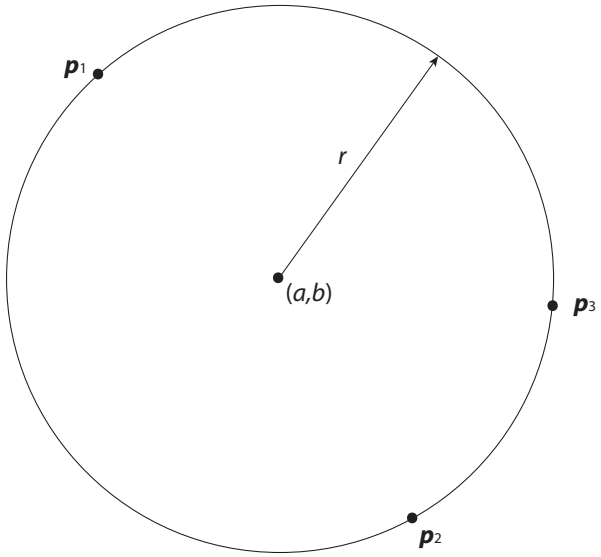


Figure 2.2: Problem description: circle.

can be written as in Equation 2.5.

$$f(a,b,r) = \left\{ \begin{array}{l} (p_{1_x} - a)^2 + (p_{1_y} - b)^2 - r^2 \\ (p_{2_x} - a)^2 + (p_{2_y} - b)^2 - r^2 \\ (p_{3_x} - a)^2 + (p_{3_y} - b)^2 - r^2 \end{array} \right\} = 0 \tag{2.5}$$

This is a degree-8 system. For now, rather than trying to solve the equations directly, consider instead the system given in Equation 2.6.

$$g(a,b,r) = \left\{ \begin{array}{r} a^2 - 1 \\ b^2 - 1 \\ r^2 - (a + 3b) \end{array} \right\} = 0 \tag{2.6}$$

This is also a degree-8 system in the same variables. g has no physical significance, and was chosen only for its simple solution structure. The solutions to Equation 2.6 can immediately be read off, and are placed in the columns of:

$$\begin{bmatrix} a \\ b \\ r \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 2 & i\sqrt{2} & -2 & -i\sqrt{2} & \sqrt{2} & i2 & -\sqrt{2} & -i2 \end{bmatrix} \tag{2.7}$$

Let $g(a,b,r)$ be known as the Start System, and $f(a,b,r)$ as the Target System. It is possible to use knowledge of the solution structure of one system of polynomial equations to determine the solutions of another. A new function, known as a Homotopy (Zulehner, 1988) is defined. A homotopy is given in its simplest form in Equation 2.8.

$$h(a,b,r) = (1-t)g(a,b,r) + tf(a,b,r) \tag{2.8}$$

As the Homotopy Parameter t is incrementally varied from 0 to 1, h transforms slowly from g into f. Starting at $t = 0$ with the known solutions of g, h is solved numerically for a, b and r at intervals between $t = 0$ and $t = 1$. At present, there are two main approaches taken to the tracking of solutions from $t = 0$ to $t = 1$ via Equation 2.8. The first of these is known as the Predictor-Corrector method, and the second as the Simplicial method (Allgower & Georg, 1980). Only the predictor-corrector method will be discussed here in detail. Coding methods for the predictor-corrector approach are well documented (Morgan, 1987; Wampler *et al.*, 1990). Note also that continuation methods can involve more than a single continuation parameter (Henderson, 2004), although only the single case is considered here.

As a practical example, assume that the intention is to solve Equation 2.5 with the (rather uninteresting) points $p_1 = \{-1, 1\}$, $p_2 = \{1, 1\}$ and $p_3 = \{1, -1\}$, which define three corners of a square. Commonsense suggests that the solution will be a circle of radius $\sqrt{2}$ centred at the origin. Running a continuation script (using Equation 2.6 as the

start system and 2.7 as the start solutions) yields the following results:

$$\begin{bmatrix} a \\ b \\ r \end{bmatrix} = \begin{bmatrix} \infty & \infty & \infty & \infty & \infty & 0 & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty & 0 & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty & \sqrt{2} & \infty & -\sqrt{2} \end{bmatrix}$$

The problem is seen to have six solutions at infinity. The remaining two are both finite, geometrically isolated non-singular solutions. Only one is physically realisable, with a negative radius being meaningless. In the continuation process, six of the paths from the finite start solutions have diverged to infinity, while two have tracked to finite, geometrically isolated points.

Polynomial continuation presents itself as an appealing tool in kinematics because it finds not just one, but **all** of the solutions to a given system (whether they are physically realisable or not). Numerical methods, such as the Newton Raphson Method are reliable, but only when the starting estimate lies within an attraction region of a desirable solution.

2.2 Predictor-Corrector Method (Path Following)

Once the $f(\mathbf{x})$ (target) and $g(\mathbf{x})$ (start) systems (which combine to give $h(\mathbf{x}, t)$) have been constructed, small increments can be made in the arguments of h to produce:

$$h(\mathbf{x} + \Delta\mathbf{x}, t + \Delta t) = h(\mathbf{x}, t) + J_x\Delta\mathbf{x} + J_t\Delta t + \cdots$$

where J_x and J_t are the Jacobians of h with respect to $\mathbf{x}$ and t respectively. Since it is assumed that h equates to zero at all points between $t = 0$ and $t = 1$, the above can be solved as in Equation 2.9, which is known as the *predictor* step.

$$\Delta\mathbf{x} = -J_x^{-1}J_t\Delta t \tag{2.9}$$

If t undergoes an incremental increase, the solutions to h can be refined by repetitively applying Equation 2.10, which is known as the *corrector* step.

$$\Delta\mathbf{x} = -J_x^{-1}h(\mathbf{x}, t) \tag{2.10}$$

A graphical representation of this approach is given in Figure 2.3.

Both the predictor and corrector steps described above can be improved: the total length of the predictor step can be controlled, and the corrector step can be forced to be

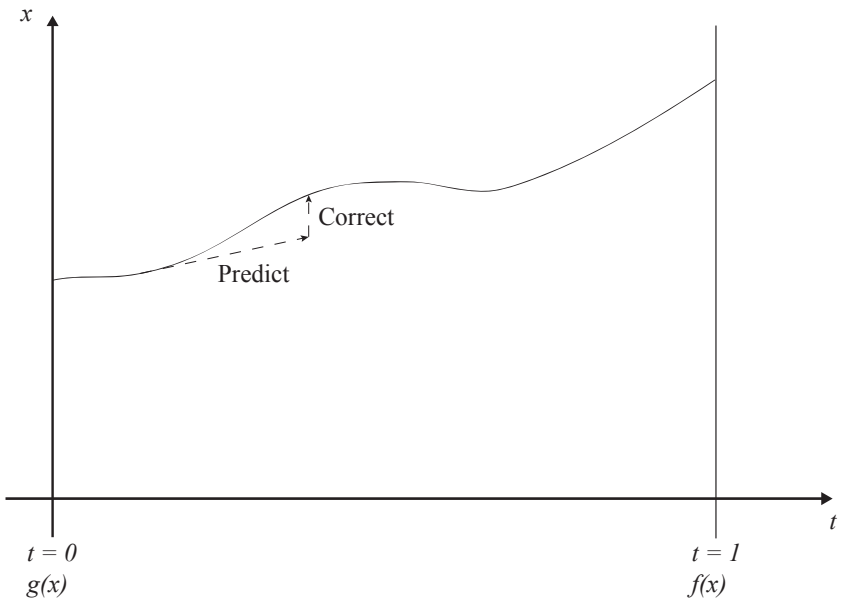


Figure 2.3: Standard path tracking.

perpendicular to the predictor step. Firstly, construct a new vector $\mathbf{z} = \begin{bmatrix}\mathbf{x}^T & t\end{bmatrix}^T$. From before:

$$\frac{d\mathbf{x}}{dt} = -J_x^{-1} J_t$$

The $\mathbf{z}$ vector is updated during the path-following process using Equation 2.11:

$$\mathbf{z}_1 = \mathbf{z}_0 + \frac{\begin{bmatrix} \frac{d\mathbf{x}}{dt} & 1 \end{bmatrix}}{\left\| \begin{bmatrix} \frac{d\mathbf{x}}{dt} & 1 \end{bmatrix} \right\|} \cdot \text{step} \tag{2.11}$$

where the size of the step can be varied. Once the predictor step has been performed, the correction can then be forced to occur along a line perpendicular to the predictor vector; the idea being that the distance to the path should be shorter in the majority of cases. The corrector step is made by repeatedly solving Equation 2.12, which ensures the orthogonality of the predictor and subsequent corrector steps.

$$\left[\begin{array}{c|c} J_x & J_t \\ \hline (\mathbf{z}_0 - \mathbf{z}_1)^T & \end{array}\right] [\mathbf{z}_{k+1} - \mathbf{z}_k] = - \begin{bmatrix} h(\mathbf{z}, t) \\ 0 \end{bmatrix} \tag{2.12}$$

A representation of the tangential prediction/orthogonal correction method is given in

Figure 2.4.

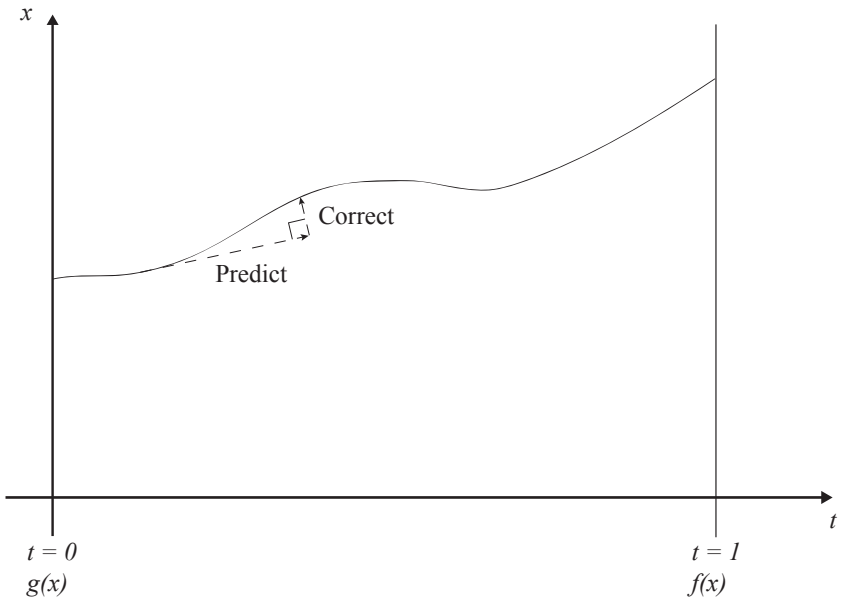


Figure 2.4: Tangential path tracking.

In practice, a correction process using a fixed value of t is usually sufficient. The method used to generate the results contained in this document is a combination of a predictor step based on Equation 2.11, and a corrector step based on Equation 2.10.

2.2.1 Step Size Control

How far to progress along the solution path with the next prediction is a key issue in efficient programming. If the step size is too small then each path will take too long to trace. If the step size is too big, then the corrector may fail to re-converge to the path. The method employed to control the step size is as follows:

1. provide an initial step size;
2. if the correction term after a certain number of iterations is small enough, then count the step as a success;
3. if a certain number of consecutive successful steps have been taken, increase the step size;
4. if the last step was considered a failure, decrease the step size;

5. ensure that the step size lies within the upper and lower allowable universal bounds.

The choice of the initial step size is very important, and dictates how likely the path follower is to experience a failure at the start.

2.2.2 Path Follower Control Parameters

The path following process as implemented in the Matlab code used throughout this document can be represented as a flow chart, included as Figure 2.5. The flow chart illustrates a number of step size controls used in path following, most of which were synthesised from other authors' publications on the topic. The combination of step size controls constitutes a unique path following algorithm, suited to the types of polynomials which arise in the study of the kinematics of deployable structures.

A number of numerical parameters can be used to define the way in which the predictor-corrector path follower of Figure 2.5 operates. These scalars can be used to manipulate the trade-off between accuracy and speed in computation. A list of the parameters and their functions is provided.

Initial step size User defined figure (usually 10^{-4}) which sets the magnitude of the first predictor step in the process. Too big a step could result in a path jump (failed re-convergence) before the iteration process has even begun.

Maximum number of steps Simply sets an upper limit on the number of prediction steps before a path following process is considered to have failed.

Minimum step size A lower limit on allowed step size. Often encountered when attempting to follow solutions to infinity. Should typically be greater than machine precision.

Maximum step size Step size is allowed to grow to a maximum of this magnitude. This limit is typically reached in the mid stages of path following $t = 0.2 - 0.8$, and to a large extent dictates the speed of the overall process. Path jumps can easily occur if this number is set too large.

Minimum successful iterations This parameter sets the number of prediction-correction steps which must occur without incident before the step size can be increased with confidence. It is typically in the range 4 to 6.

Allowable predictor vector direction change This parameter constitutes one of the main defences against path jumps. It sets the minimum allowable inner product between

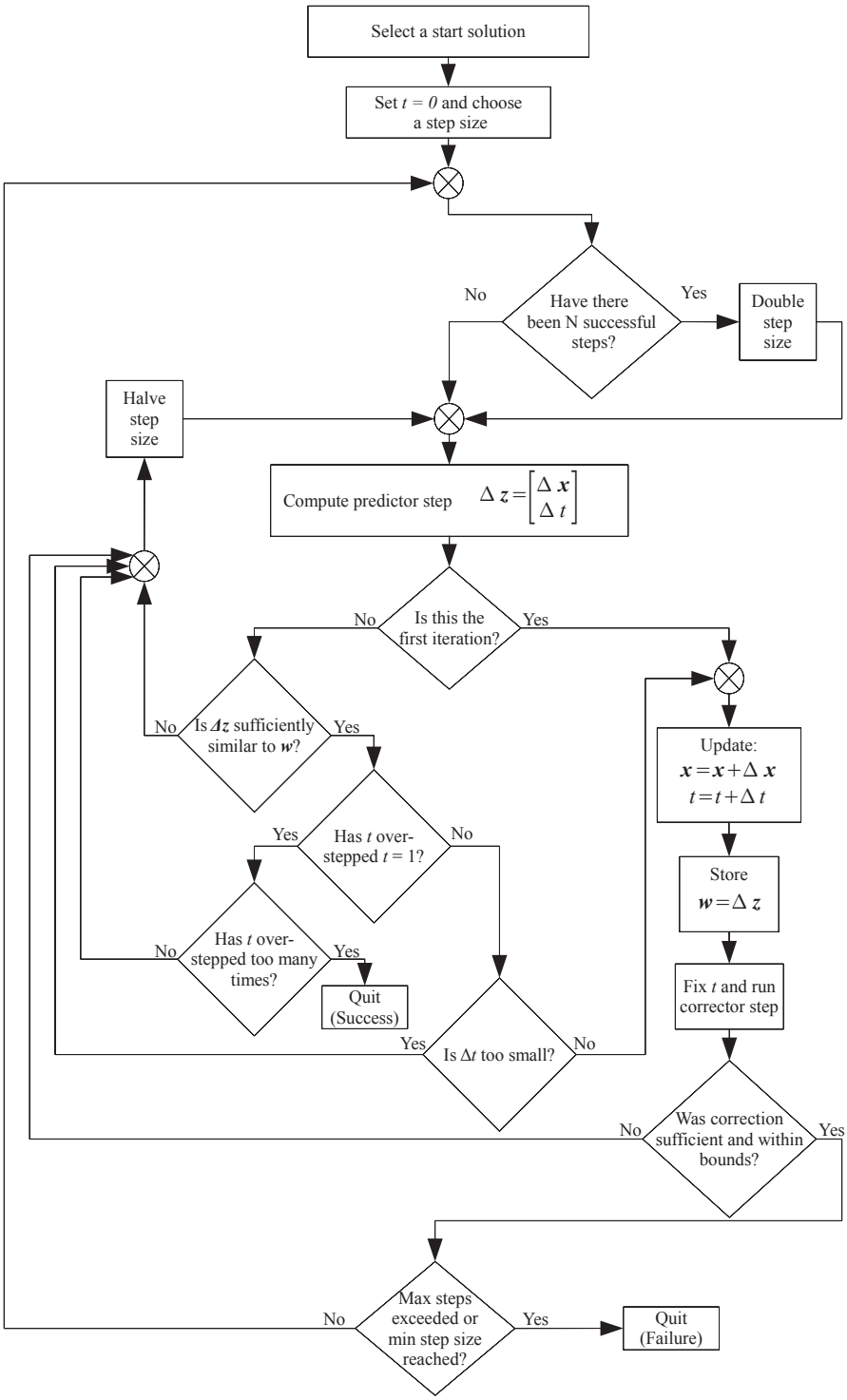


Figure 2.5: Flowchart describing the predictor-corrector method used in path following.

the current predictor vector and the previous one. Simply, each step in the path following process must roughly be in the same direction as the last. A big change in predictor direction is often a good indication that a path jump has occurred. The value is typically between 0.95 and 0.99. Regions of paths with high curvature will cause the step size to decrease, allowing the curve to be traced more accurately.

Minimum convergence requirement The set value for the magnitude of the corrector step at which the convergence loop is terminated.

Minimum function magnitude The value of the 2-norm of the homotopy output required after the corrector step before a new corrector step is allowed. It is usually around 10^{-6}. This criterion is often the first to catch solutions on their way to infinity.

Continuation parameter overstep allowance This integer specifies the number of times t should exceed 1 before a target system solution is declared to have been found. Each time t does exceed 1, the path follower returns to the previous point on the path found between 0 and 1 and the step size is halved. In this way the path follower converges slowly on $t = 1$. A value of 10 is usually sufficient for this parameter.

Allowable convergence stretch The correction phase should not shift the solution by a distance significantly greater than the magnitude of the step size for the predictor stage. If it does, then it is very likely that a path jump has occurred. This parameter is applied as a multiplication factor of the current step size, and if the corrector step shifts the solution by more than this amount then the entire predictor-corrector iteration is considered to be a failure. A value of 2 is usually effective.

Minimum t component As a path begins to diverge to infinity, the amount by which the continuation parameter (t) increases with each iteration becomes a smaller and smaller fraction of the amount by which the solution variables themselves increase. This parameter sets the minimum allowable size of this fraction, usually around 10^{-6} to 10^{-5}.

Multi-precision flag Simply a 0 or 1 which specifies whether the program should be enabled to make use of multiple precision capabilities in undertaking particularly troublesome convergence loops. For almost all purposes, standard machine precision is sufficient, and the flag is set to 0 (off).

Output type flag Another flag with value 0 or 1. A 1 (on) requests that the path follower

return every point encountered on the path in the form of a matrix. Can be very heavy on memory usage.

2.3 Homogenisation and Bézout Numbers

A simple example of homogenisation has already been seen in the discussion of solutions at infinity above. In homogenising a system of equations the hope is that some, or all, of the solutions at infinity can be removed. Homogenising a system of equations can remove solutions at infinity, reducing the number of paths unnecessarily tracked, but it does result in the introduction of new homogeneous variables (denoted as p previously). The addition of the new homogeneous variables also requires the addition of some new equations if the system as a whole is to be kept square.

The Bézout Number (Wampler *et al.*, 1990) of a system of polynomial equations can loosely be described as the number of solutions left to be found following homogenisation (after some solutions at infinity have been removed). Each type of solution has its own associated Bézout number. Non-singular, geometrically isolated solutions have a Bézout number of 1, but singular solutions may have higher Bézout numbers, which correspond to the multiplicity of the solution at that point. Summing the Bézout numbers for all the solutions to be found in a particular system gives the total Bézout number for the system. Prior to the development of polyhedral methods, n-homogenisation was used extensively in the study of kinematic equations (Dhingra *et al.*, 1994).

Consider again the system of three equations given in Equation 2.2. As a system with degree 8, it is expected that eight solutions will be present, counting those which may be at infinity and multiplicities. As an alternative to homogenising the system in the way outlined previously (known as 1-homogenising), the variables can be placed into homogeneous groups, and the system n-homogenised. Consider forming two homogeneous groups from the set $\{x, y, z\}$: $\{x\}$ and $\{y, z\}$. Let k be the number of variables in each group, so in this case: $k_1 = 1$ and $k_2 = 2$. With each homogeneous group is associated a new homogeneous variable, say p and q respectively for this example. In the original equations, set $x \rightarrow X/p$ and $y \rightarrow Y/q, z \rightarrow Z/q$. Either p or q independently going to zero is enough to define a solution at infinity. What becomes of interest now, is not the degree of the original equations in all the variables, but rather in each group. The 2-homogenisation described here has the homogeneous structure given in Table 2.2. The Bézout Number for the 2-homogenised system is slightly more difficult to calculate. Mathematically, it is the coefficient of the term $\prod_{j=1}^{m} \alpha_j^{k_j}$ in Equation 2.13 (see Wampler

Degree	Group 1	Group 2
Equation 1	2	2
Equation 2	1	1
Equation 3	1	1

Table 2.2: Homogeneous structure of Equation 2.2.

et al. (1990)):

$$\prod_{l=1}^{n}\left(\sum_{j=1}^{m} d_{jl}\alpha_j\right) \tag{2.13}$$

where n is the number of equations, m is the number of groups and d_{jl} is the degree of the l^{th} equation with respect to the j^{th} group. The variable α is a 'place-holder' for the original terms of the equations. In this example, the coefficient is 6, which is marginally smaller than the original system's 8. The benefit of n-homogenisation is that it can exclude many solutions at infinity from the outset, reducing the number of paths to be followed. Often the number of paths to be followed can be halved or more, depending on the homogeneous grouping used. Here, the modest reduction from eight to six solutions still affords some computational benefit. Once the 2-homogenisation has been performed, and a multi-homogeneous start system with six solutions constructed, six paths can be found to trace to the solutions of Equation 2.2:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -3.1164 & 2.5444 & 0.3002 & -0.6143 & -0.1140 & \infty \\ 0.5070 & 0.8211 & 1.3682 & -0.9152 & 2.6927 & \infty \\ 6.7398 & -4.2678 & 0.7677 & 0.3134 & 2.9206 & \infty \end{bmatrix}$$

To confirm that there are in fact five finite solutions, note that (one of the ways in which) Equation 2.2 can be written following elimination is:

$$57y^5 - 255y^4 + 260y^3 + 120y^2 - 256y + 80 = 0$$

which clearly has five solutions. The solutions of this Univariate polynomial can be found as the eigenvalues of the *companion matrix* (Sommese & Wampler, 2005):

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ -\frac{80}{57} & \frac{256}{57} & -\frac{120}{57} & -\frac{260}{57} & \frac{255}{57} \end{bmatrix}$$

which match those solutions found using continuation.

Homogenisation introduces an extra computational complication in the form of additional equations to account for the homogeneous variables. An equation of the form:

$$c_{j0}y_{j0} + \ldots + c_{jk_j}y_{jk_j} - 1 = 0 \; (j = 1 : m) \tag{2.14}$$

(where y are the elements of the particular group and c are complex coefficients chosen at random (Wampler *et al.*, 1990)) introduced for each homogeneous group works well. It is then possible to solve for the homogeneous variables for each group at the end of the calculations. A homogeneous variable of value zero corresponds to a solution at infinity. Recall the example of Equation 2.3, in which two solutions at infinity were discovered at $(1, \pm i)$. A tracking of the four start points of the 1-homogenised version of 2.3 leads to four finite solutions, of which two would have homogeneous variables equal to zero, and would also be a scalar multiple of $(1, \pm i)$.

2.3.1 Multi-Homogeneous Start Systems

To effectively implement the Polynomial Continuation method, it is necessary to be able to automate the generation of appropriate start systems. The homogeneous structure of a start system must be the same as that of the target system, and it must have the same number of solutions.

Return once more to the simple polynomial system of Equation 2.2, which has the homogeneous structure given in Table 2.2. A general start system can be constructed which consists of linear factors only, making solution straightforward. The example at hand would have a start system of the form given in Equation 2.15.

$$g(x, y, z) = \left\{ \begin{array}{r} (x - \alpha_1)(x - \alpha_2)(y + \tau_1 z - \beta_1)(y + \tau_2 z - \beta_2) \\ (x - \alpha_3)(y + \tau_3 z - \beta_3) \\ (x - \alpha_4)(y + \tau_4 z - \beta_4) \end{array} \right\} = 0 \tag{2.15}$$

The coefficients α, τ and β are chosen at random. The more uncorrelated the coefficients, the better. The Bézout Number for a system with this homogeneous structure is known to be six, so there must be six finite solutions to this start system. It is easy to find the roots of an equation corresponding to variables which alone constitute a particular group (group of size one). In the notation used in Equation 2.15 they are simply $\alpha_1, \alpha_2, \ldots$. To find the roots corresponding to the members of a group containing $k > 1$ members, k equations are required. In this case, group 2 has two members, so to find x, y pairs

requires a 2×2 matrix inversion. Explicitly, the solutions to 2.15 are:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} & \alpha_1 & \\ \begin{bmatrix} 1 & \tau_3 \\ 1 & \tau_4 \end{bmatrix}^{-1} & & \begin{bmatrix} \beta_3 \\ \beta_4 \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} & \alpha_3 & \\ \begin{bmatrix} 1 & \tau_1 \\ 1 & \tau_4 \end{bmatrix}^{-1} & & \begin{bmatrix} \beta_1 \\ \beta_4 \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} & \alpha_4 & \\ \begin{bmatrix} 1 & \tau_1 \\ 1 & \tau_3 \end{bmatrix}^{-1} & & \begin{bmatrix} \beta_1 \\ \beta_3 \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} & \alpha_2 & \\ \begin{bmatrix} 1 & \tau_3 \\ 1 & \tau_4 \end{bmatrix}^{-1} & & \begin{bmatrix} \beta_3 \\ \beta_4 \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} & \alpha_3 & \\ \begin{bmatrix} 1 & \tau_2 \\ 1 & \tau_4 \end{bmatrix}^{-1} & & \begin{bmatrix} \beta_2 \\ \beta_4 \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} & \alpha_4 & \\ \begin{bmatrix} 1 & \tau_2 \\ 1 & \tau_3 \end{bmatrix}^{-1} & & \begin{bmatrix} \beta_2 \\ \beta_3 \end{bmatrix} \end{bmatrix}$$

These solutions are all unique.

Before solutions can be found, it is necessary to construct a 'Solution Map' (method synthesized by the author), which describes which equations are to be used to calculate the value of a particular variable. The solution map for the case above is given in Table 2.3. If the degree of a particular equation with respect to a homogeneous group is greater

	Solution Num:	1	2	3	4	5	6
x	Equation	1	2	3	1	2	3
	Multiplicity	1	1	1	2	1	1
y	Equation	2	1	1	2	1	1
	Multiplicity	1	1	1	1	2	2
z	Equation	3	3	2	3	3	2
	Multiplicity	1	1	1	1	1	1

Table 2.3: Solution map for a degree-8 system of equations (Equation 2.15).

than one, then the corresponding equation in the start system will contain more than one factor consisting of the elements of that group. Looking at Table 2.2, it can be seen that Equation 2.2 has degree two in both group 1 and group 2. 'Multiplicity' in Table 2.3 refers to the particular occurrence of a group within the start system equation. Since groups 1 and 2 each appear in two factors in the first equation of the start system in Equation 2.15, equation 1 can be accompanied by multiplicities of 1 and 2.

2.3.2 The Homotopy

While the definition of a homotopy given in Equation 2.8 is sufficient for most applications, it is not necessarily as numerically stable as some others which could be used. Take

for example, the case that $f = -g$, for which the value $t = 1/2$ yields nothing at all. In practice, a homotopy of the form given in Equation 2.16 is used for most applications.

$$h(a, b, r) = \exp(i\theta)(1 - t)g(a, b, r) + tf(a, b, r) \tag{2.16}$$

Here, θ is a randomly chosen number. This reduces the probability of numerical difficulties arising in the homotopy effectively to zero.

If the only difference between the systems g and f is the value of the (complex) coefficients of the Monomials, a simple and very efficient form of continuation known as *Coefficient-Parameter Continuation* (Morgan & Sommese, 1989) can be used to map from one system to the other.

2.4 Polyhedral Homotopy Method

There are several ways of constructing start systems and homotopies, each with their own advantages and disadvantages. One's goal should always be to decrease the number of solution paths to be followed by pre-identifying and eliminating those which will diverge towards infinity. Often, the most efficient methods for small systems are not particularly generalisable, and the complexity of the start system and its construction outweighs the benefits to be gained by a reduction in path count for larger systems.

Ideally, one would construct a start system with a monomial structure as similar as possible to that of the target system. A start system constructed using linear terms via n-homogenisation will usually closely approximate the target structure, but almost invariably contains extra terms. An outline is provided in this section of a method which has been in development since the mid 1990's, and which arguably constitutes the best *generalisable* system for constructing start systems and homotopies. The exact method presented here is adapted from one contained in the article Li (1999), and described later in more detail in Li (2003). The reader is referred to these sources for more information.

In Bernstein (1975), it was shown than an upper bound for the number of roots of a system of Laurent polynomials (note that the indices do not need to be ≥ 0 for this to work) is provided by computing something known as the Mixed Volume. The mixed volume refers to a special way of combining the Convex Polytopes of the Supports of each equation. *Support* refers to the set of indices which appear in a given polynomial. For example, in:

$$P(x_1, x_2, x_3) = 3x_1^2x_2x_3^3 + 1.4x_2x_3^2 - 6x_1^3 + 4.3x_2^2 + 1.2x_2 - 3.2x_3 + 7$$

in the variables $\{x_1, x_2, x_3\}$, the support is:

$$Supp = \left[\begin{Bmatrix} 2 \\ 1 \\ 3 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 1 \\ 2 \end{Bmatrix}, \begin{Bmatrix} 3 \\ 0 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 2 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 1 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \right]$$

Notice that so far the coefficients of the equation have not featured. Imagine now that each element of the support defines a point in n dimensional space, where n is the dimension of the solution vectors, so each equation's support defines a collection of points. To make use of the results of Bernstein (1975) (often simply referred to as Bernstein's Theorem), one need only consider the convex hull of the collection of points formed by each support. Returning to the example above, the convex hull is:

$$\text{conv}(Supp) = Q = \left[\begin{Bmatrix} 2 \\ 1 \\ 3 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 1 \\ 2 \end{Bmatrix}, \begin{Bmatrix} 3 \\ 0 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 2 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \right]$$

which defines a convex polytope. Two or more polytopes can be combined using a Minkowski Sum, defined in Equation 2.17 (see Sommese & Wampler (2005), p139 and Konaxis (2010), p18).

$$Q_1 + Q_2 = \{q_1 + q_2 \mid q_1 \in Q_1, q_2 \in Q_2\} \tag{2.17}$$

Next, the definition of volume used here is simply that which follows geometrically.

$$\text{vol}(Q') = \frac{1}{n!}\det\left[a_1 - a_0, \ldots, a_n - a_0\right] \tag{2.18}$$

Here, Q' is a simplex with $n+1$ vertices, labelled $a_0, \ldots, a_n$. The mixed volume (Lee & Li, 2007) of a system of polytopes is defined in Equation 2.19 (where C_i^n represents the combinations of n things taken i at a time).

$$\begin{aligned} \mathcal{M}(Q_1, \ldots, Q_n) &= \text{coeff}\left(\lambda_1 \ldots \lambda_n, \text{vol}(\lambda_1 Q_1 + \ldots + \lambda_n Q_n)\right) \\ &= \sum_{i=1}^{n} (-1)^{n-i} \text{vol}\left(\sum_{j \in C_i^n} Q_j\right) \end{aligned} \tag{2.19}$$

Bernstein's theorem states that the root count of a polynomial system with supports $\text{Supp}_1, \ldots, \text{Supp}_n$ on $(\mathbb{C})^n \setminus \mathbf{0}$ is equal to the mixed volume of its supports as defined in Equation 2.19.

To get a better handle on the concept of a mixed volume, consider the polynomial system given in Equation 2.20.

$$\begin{aligned} p_1 &= ax^3y + bxy^2 + cy + 1 &&= 0 \\ p_2 &= dxy^3 + ex + 1 &&= 0 \end{aligned} \tag{2.20}$$

Here we have:

$$Q_1 = \left[\begin{Bmatrix} 3 \\ 1 \end{Bmatrix}, \begin{Bmatrix} 1 \\ 2 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \right]$$

$$Q_2 = \left[\begin{Bmatrix} 1 \\ 3 \end{Bmatrix}, \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \right]$$

$$Q_1 + Q_2 = \left[\begin{Bmatrix} 4 \\ 4 \end{Bmatrix}, \begin{Bmatrix} 4 \\ 1 \end{Bmatrix}, \begin{Bmatrix} 2 \\ 5 \end{Bmatrix}, \begin{Bmatrix} 1 \\ 4 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}, \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \right]$$

A graphical representation of the situation is given in Figure 2.6. Directly applying Equa-

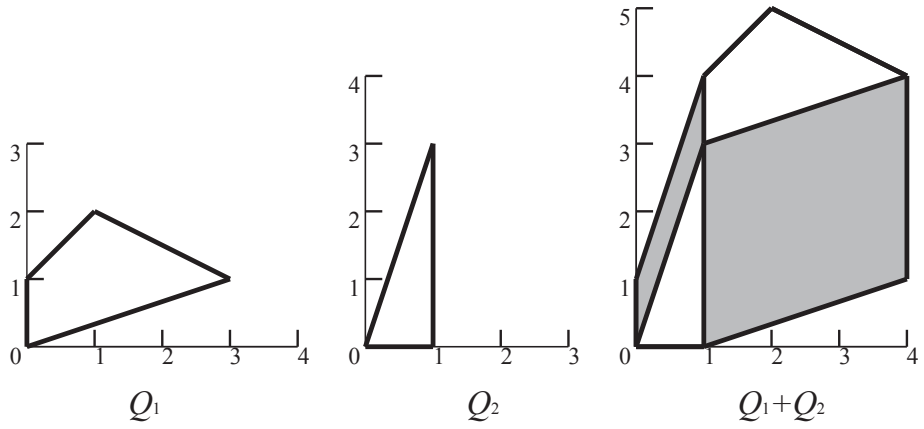


Figure 2.6: Graphical representation of mixed volume of equation 2.20.

tion 2.19 shows that this system has a mixed volume of 10. Note at this point that the Bézout number for this system is 16, while the 2-homogeneous Bézout number is 11. Already it can be seen that a polyhedral homotopy will produce the smallest number of paths to be followed. Looking at the graphical representation of $Q_1 + Q_2$, it is possible to see two greyed areas, each sharing a single edge with each of Q_1 and Q_2. These areas are known as *mixed cells*, *mixed subdivisions* or *n-subfaces*. Their areas (or volumes in higher dimensions) can be combined to give the mixed volume. In this case, the smaller

area has an area of 1, and the larger, 9, which combine to 10. Each of these mixed cells is said to *generate* a number of start solutions equal to its volume. In most cases, the way in which the polyhedra may be arranged in order to produce mixed cells is not unique, although the mixed volume will not vary. When a valid subdivision is being sought computationally, a technique to ensure that one is found is to actually introduce a $n+1^{\text{th}}$ dimension to the problem. This process is referred to as *lifting*, and the exact manner in which it is applied has been the subject of quite some research in its own right. Particularly efficient methods of generating a lifting are available if the polynomials being investigated are symmetric (Verschelde & Gatermann, 1995), but only the general lifting case will be presented below (symmetric polynomials do not frequently arise in the types of kinematic systems considered here). Briefly: the process involves taking each vertex of each Q_i, a_{ij} say, and using it as the argument of a random (real valued) Lifting Function, unique to each support. Denoting these random functions as ω_i for $i = 1, \ldots, n$, then each support is lifted by one dimension such that:

$$\bar{Q}_i = \left\{ \text{conv}(\bar{a}_{ij}) \mid \bar{a}_{ij} = \begin{bmatrix} a_{ij} \\ \omega_i(a_{ij}) \end{bmatrix} \right\} \quad \begin{matrix} i = 1, \ldots, n \\ a_{ij} \in Q_i \end{matrix} \tag{2.21}$$

Each $\bar{Q}_i$ can be used to form the Minkowski sum $\bar{Q}_i + \ldots + \bar{Q}_n$, which itself defines a convex polytope. The edges of the *lower hull* of this new polytope can be projected onto the original n dimensions, forming a valid subdivision, from which the mixed cells can be identified.

As a further example, consider once more the system of Equation 2.2. Recall that this system has a total degree (or 1-homogeneous Bézout number) of 8, and a 2-homogeneous Bézout number of 6, with a 2-homogeneous start system tracking to the five finite solutions of the target system, and one at infinity. As it happens, Equation 2.2 has a mixed volume of 5, illustrating the way in which polyhedral methods can narrow down the number of start solutions to the absolute upper bound.

The theorem of Bernstein (1975) was applied directly to the solution of polynomial systems via continuation in Huber & Sturmfels (1995). There are several methods available to construct polyhedral homotopies. At the core of each method is a search for a valid subdivision which can be used to identify the mixed cells. The most intuitive method is simply to take every single combination of polyhedra edges, and test each one for validity. This is very reliable, and works for small systems, but quickly becomes impractical in systems of dimension greater than 3 or 4, or with equations with a great many terms.

An outline of the method given in Li (1999) and Li (2003) is given below. First, use a contracted notation in which $x^a = x_1^{a_1} x_2^{a_2} \ldots x_n^{a_n}$. Each polynomial can then be

represented as:

$$p_i(x) = \sum_{a_{ij} \in Q_i} c_{ij} x^{a_{ij}} \quad i = 1, \ldots, n \tag{2.22}$$

To increase numerical reliability, it is usually better to solve for the solutions of a system with the same support structure, but with random complex coefficients first, and then perform a Coefficient Homotopy from this system to the original target. In light of this, let us deal instead with:

$$p_i(x) = \sum_{a_{ij} \in Q_i} c'_{ij} x^{a_{ij}} \quad i = 1, \ldots, n \tag{2.23}$$

where c'_{ij} are random complex coefficients. Next, introduce the continuation parameter, t, by multiplying each term by $t^{\omega_i(a_{ij})}$. Notice that at $t = 1$, $h_i(x)|_{t=1} = p_i(x)$.

$$h_i(x) = \sum_{a_{ij} \in Q_i} c'_{ij} x^{a_{ij}} t^{\omega_i(a_{ij})} \tag{2.24}$$

Now, a change of variable is performed, and in the process an unknown set of real numbers $\alpha = (\alpha_1, \ldots, \alpha_n)$ is introduced. Set:

$$x_i = y_i t^{\alpha_i} \quad i = 1, \ldots, n$$

which transforms Equation 2.24 into:

$$h_i(y) = \sum_{a_{ij} \in Q_i} c'_{ij} y^{a_{ij}} t^{\langle a_{ij}, \alpha \rangle + \omega_i(a_{ij})} \tag{2.25}$$

Finally, the continuation parameter is altered once more by introducing a further unknown, the real scalar β_i. Each equation is multiplied through by $t^{-\beta_i}$.

$$h^*_i(y) = \sum_{a_{ij} \in Q_i} c'_{ij} y^{a_{ij}} t^{\langle a_{ij}, \alpha \rangle + \omega_i(a_{ij}) - \beta_i} \tag{2.26}$$

The unknown β_i must be chosen to satisfy a very special set of criteria. In particular, if each Q_i has m_i elements, and $j = 1, \ldots, m_i$, then from each Q_i, it can be shown that there must exist a pair of elements $\{a'_i, a''_i\}$ such that:

$$\begin{aligned} &\langle a'_i, \alpha \rangle + \omega_i(a'_i) = \langle a''_i, \alpha \rangle + \omega_i(a''_i) = \beta_i \\ &\langle a'_i, \alpha \rangle + \omega_i(a'_i) \leq \langle a_{ij}, \alpha \rangle + \omega_i(a_{ij}) \quad \forall a_{ij} \in Q_i \backslash \{a'_i, a''_i\} \end{aligned} \tag{2.27}$$

Using the criteria of 2.27, Equation 2.26 can be rewritten as in Equation 2.28, which represents the final form of the polyhedral homotopy.

$$h_i^*(y) = \sum_{a_{ij} \in \left\{a_i', a_i''\right\}} c_{ij}' y^{a_{ij}} + \sum_{a_{ij} \in Q_i \setminus \left\{a_i', a_i''\right\}} c_{ij}' y^{a_{ij}} t^{\langle a_{ij}, \alpha \rangle + \omega_i(a_{ij}) - \beta_i} \tag{2.28}$$

It is most probable that more than one set of $\{\alpha, \beta_i\}$ unknowns will be found which satisfy the constraints of Equation 2.27. At this stage an analogy can be made with the geometry of the problem, as **each of these sets of $\{\alpha, \beta_i\}$ unknowns corresponds to a mixed cell**.

It can be seen that when $t = 1$, Equation 2.28 is equivalent to Equation 2.23; the original random-coefficient system of polynomials. When $t = 0$, only the first two terms of each of the n equations remain. There will be as many versions of Equation 2.28 as there are mixed cells (or $\{\alpha, \beta_i\}$ sets). Each of these $n \times 2$ *binomial* systems which arise at $t = 0$ can be solved using a relatively straightforward process to produce a number of solutions equal to the volume of the corresponding mixed cell. The solutions of each of these subdivisions can then be traced using a predictor-corrector path follower from $t = 0$ to $t = 1$, after which all the solutions of the random-coefficient system of polynomials will be known. These solutions can then be traced in a second continuation process, using a standard coefficient homotopy, to find all the solutions of the system with the original coefficients. A short summary of the process is given:

1. construct a system of polynomials with the same structure as the target system, but with random complex coefficients;
2. ensure that each support contains the origin (each equation has a constant term), for the greatest generality;
3. use a series of n lifting functions to add an $n + 1^{\text{th}}$ dimension;
4. use this lifting to determine all sets of $\{\alpha, \beta_i\}$ (and their corresponding pairs of vertices) which satisfy the constraints of Equation 2.27;
5. set $t = 0$ and determine all the solutions to each of the binomial systems arising from Equation 2.28;
6. use continuation to trace each of these solutions to those of Equation 2.23;
7. trace all of these solutions to those of the original Equation 2.22 using a coefficient homotopy.

Up till now, the method of completing step 4 in the list above has not been addressed, although it is possibly the most labour intensive stage in the process. The primary goal in this step is to find all the vertices of the convex polytope defined by the lifted indices of each equation. Two methods are given below.

2.4.1 A Simplex Minimisation Method

If a system of n polynomial equations has only $r < n$ distinct supports, then it is possible to take advantage of this fact to speed up the solving of Equation 2.27. Generally, one considers the case of $r = n$ first, but for compactness, only the general case of $1 \leq r \leq n$ will be presented here. A polynomial system with $r < n$ distinct supports is said to have Semi-Mixed Supports, while if they are all unique, to have Mixed Supports. An example of a system with semi-mixed supports is given in Section 3.6. Define the multiplicity of each type of support as k_i, where $\sum_{i=1}^{r} k_i = n$, and $1 \leq l \leq k_i$. Equation 2.28 requires a slight modification to the form of Equation 2.30. The definition in Equation 2.27 also needs to be slightly modified to allow larger groupings than pairs to satisfy its requirements (Equation 2.29).

$$\begin{aligned} C_i &= \{a_{ij} |\, \langle a_{ij}, \alpha \rangle + \omega_{ij} = \beta_i\} \quad i = 1, \ldots, r \\ \beta_i &\leq \langle a_{ij}, \alpha \rangle + \omega_i(a_{ij}) \quad \forall a_{ij} \in Q_i \backslash C_i \end{aligned} \tag{2.29}$$

$$h^*_{il}(y) = \sum_{a_{ij} \in C_i} c'_{ijl} y^{a_{ij}} + \sum_{a_{ij} \in Q_i \backslash C_i} c'_{ijl} y^{a_{ij}} t^{\langle a_{ij}, \alpha \rangle + \omega_i(a_{ij}) - \beta_i} \tag{2.30}$$

Each of the r sets C_i contains $k_i + 1$ elements. In the case that each equation's support is unique, and $r = n$, then C_i reduces to the $\{a'_i, a''_i\}$ pairs described above.

Finding Level-k Subfaces

Following the method of Li (2003), the first task is to find all points which lie on the boundary of the polyhedron:

$$\beta_i \leq \langle a_{ij}, \alpha \rangle + \omega_i(a_{ij})$$

This can be rewritten in matrix form as:

$$\begin{bmatrix} 1 & -a_{i1,1} & \dots & -a_{i1,n} \\ 1 & -a_{i2,1} & \dots & -a_{i2,n} \\ \vdots & \vdots & & \vdots \\ 1 & -a_{im_i,1} & \dots & -a_{im_i,n} \end{bmatrix} \begin{bmatrix} \beta_i \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} \leq \begin{bmatrix} \omega_i(a_{i1}) \\ \omega_i(a_{i2}) \\ \vdots \\ \omega_i(a_{im_i}) \end{bmatrix} \tag{2.31}$$

$$\text{or} \quad A\nu \leq \mathbf{b}$$

Equation 2.31 defines a polyhedron in $n+1$ dimensions. At each vertex of the polyhedron, at least $n+1$ of the inequalities will be active (become equalities). If more than $n+1$ of the inequalities are active at a vertex, the vertex is said to be *degenerate*. It is likely that some of the inequalities will never become active, and cannot be used to define any vertex (these lie inside the convex polytope defined by the lifted support). To determine which inequalities can be used to form vertices, one must solve what is usually known as a *phase 1* problem to find an initial vertex of 2.31, and use the simplex minimisation method to attempt to reach a vertex at which each inequality becomes active.

1. Construct a set of integers, $U_1 = \{1, \dots, m_i\}$ representing each of the inequalities in Equation 2.31;

2. solve a phase 1 problem to find an initial vertex where at least $n+1$ of the inequalities are active;

3. construct a new set on integers, V_1, representing these active inequalities, and set $U_1 = U_1 \backslash V_1$;

4. take an element off the top of U_1, say number j;

5. remove j from U_1: $U_1 = U_1 \backslash j$;

6. solve the problem:

$$\begin{gathered} \text{Minimise} - \beta_i + a_{ij,1}\alpha_1 + \ldots + a_{ij,n}\alpha_n \\ \text{s.t.} \\ A\nu \leq \mathbf{b} \\ \text{w.r.t. } (\beta_i, \alpha_1, \ldots, \alpha_n) \end{gathered}$$

   The objective of this minimisation is simply the j^{th} row of A. Note that in a practical implementation of this method, A should be row reduced to form an upper

triangle of zeros, and columns of zeros if it is rank deficient. During the minimisation process, various vertices will be encountered on the way to the target vertex. Each of these vertices will be defined by a set of active inequalities, which should be added to V. If the minimum value of the objective at a vertex is $-\omega_i(a_{ij})$, then j should be added to V_1;

7. set $U_1 = U_1 \backslash V_1$. If U_1 is not empty, return to step 4. If U_1 is empty, then exit; V_1 is the set of all possible active inequalities.

A vertex is defined by $n+1$ active inequalities. An edge of the polyhedron is defined by n inequalities, and the simplex method works by moving from a vertex, along one of the edges which forms this initial vertex, to another at which the function objective is smaller. If a degenerate vertex (one with more than $n+1$ active constraints) is encountered, then *all possible combinations* of n constraints, defining edges intersecting the vertex, must be considered. Each must be tested as a potential feasible exit direction from the current vertex. The treatment of degenerate vertices significantly slows the simplex minimisation process, but lifting functions which are sufficiently random tend to lead to polyhedra with non-degenerate vertices. Each vertex encountered during the simplex minimisation process has a new set of active inequalities which can immediately be added to V_1 and removed from U_1, meaning that each element of the initial U_1 does not need to be sought sequentially and independently (one of the great advantages of the simplex method).

The next stage involves finding pairs of inequalities which appear together. The process is quite similar.

1. Construct a set of pairs $U_2 = \{\{j_1, j_2\} \mid \{j_1, j_2\} = \mathrm{comb}(V_1, 2)\}$;
2. use a non-degenerate vertex encountered in the previous process as a start vertex;
3. remove all the pairs of inequalities which appear at this start vertex from U_2, and use them to define V_2;
4. take a pair $\{j_1, j_2\}$ from the top of U_2;
5. remove $\{j_1, j_2\}$ from U_2: $U_2 = U_2 \backslash \{j_1, j_2\}$;
6. solve the problem:

$$\begin{aligned} &\text{Minimise} - 2\beta_i + (a_{ij_1,1} + a_{ij_2,1})\alpha_1 + \ldots + (a_{ij_1,n} + a_{ij_2,n})\alpha_n \\ &\qquad\qquad \text{s.t.} \\ &\qquad A\nu \leq \mathbf{b} \\ &\text{w.r.t. } (\beta_i, \alpha_1, \ldots, \alpha_n) \end{aligned}$$

Again, remove any pairs of inequalities encountered at vertices during minimisation from U_2 and append them to V_2. If the minimum value of the objective is $-\omega_i(a_{ij_1}) - \omega_i(a_{ij_2})$, then add $\{j_1, j_2\}$ to V_2;

7. set $U_2 = U_2 \backslash V_2$. If U_2 is not empty, return to step 4. If U_2 is empty, then exit; V_2 is the set of all possible pairs of active inequalities.

If the equation in question has a multiplicity of one ($k_i = 1$), then stop here. If $k_i = 2$, then repeat the process above, this time looking for all possible combinations of three active constraints, given that each of the three elements must already have appeared in a pair with each of the other two elements in V_2. This can be pursued to the point of finding all possible combinations of $k_i + 1$ active constraints. It is much faster to undertake this iterative process in finding all the combinations of $k_i + 1$ active constraints than it is to set U_{k_i+1} with $\binom{m_i}{k_i+1}$ elements and search for them directly. $\binom{m_i}{k_i+1}$ can be a very large number!

So far, the r supports have been only been considered individually. Equation 2.29 defines a polyhedron which encompasses all of the supports. At this stage, it is known which combinations of k_i+1 elements from each of the r supports **could possibly** satisfy 2.29. Each of the r supports has yielded what is known as a *level-k subface*, and these k-subfaces need to be fitted together in a way which satisfies Equation 2.29, forming a level-r subface. Most combinations will not meet this requirement. It is, however, likely that more than one combination of elements for each support will work, resulting in more than one level-r subface (and hence more than one value of α). One could simply try all the combinations of $k_i + 1$ support elements (level-k subfaces) for each support in an attempt to satisfy Equation 2.29, but in most cases this would be quite slow. Instead, a further simplex minimisation step is employed to *extend* the level-k subface for the first support. Each level-r subface will correspond to a particular value of α and β. It cannot be known beforehand how many of these subfaces there will be. What is known is that each r-subface will define a volume:

$$\det\left(V(\alpha)\right) = \begin{vmatrix} a'_{11} - a'_{10} \\ \vdots \\ a'_{1k_1} - a'_{10} \\ \vdots \\ a'_{r1} - a'_{r0} \\ \vdots \\ a'_{rk_r} - a'_{r0} \end{vmatrix} \tag{2.32}$$

(note the r sets of $k_i + 1$ elements arranged into n pairs, forming a binomial start system as in Equation 2.28). The sum of each of the volumes $\det(V(\alpha))$ is equal to the mixed volume for the entire system.

Extending the Level-k Subfaces

Once level-k subfaces have been found for each support using Equation 2.31, attention is now turned to the subfaces generated by the first support. In practice, it is likely that quite a number of pairs (or sets of $k_1 + 1$ elements) will have been found which satisfy 2.31, each representing a level-k subface.

Assume that the elements $C_1 = \left\{a'_{10}, \ldots, a'_{1k_1}\right\}$ form one of the level-k subfaces for the first support. There is now a larger simplex:

$$\begin{aligned} \langle a_{1j} - a'_{10}, \alpha \rangle \leq \;& \omega_1(a'_{10}) - \omega_1(a_{1j}) \quad \forall a_{1j} \in Q_1 \backslash C_1 \\ \beta_2 - \langle a_{2j}, \alpha \rangle \leq \;& \omega_2(a_{2j}) \qquad\qquad\quad \forall a_{2j} \in Q_2 \end{aligned} \tag{2.33}$$

which can be written in a matrix form as:

$$\begin{bmatrix} 0 & a_{11,1} - a'_{10,1} & \ldots & a_{11,n} - a'_{10,n} \\ \vdots & \vdots & & \vdots \\ 0 & a_{1m_1,1} - a'_{10,1} & \ldots & a_{1m_1,n} - a'_{10,n} \\ 1 & -a_{21,1} & \ldots & -a_{21,n} \\ \vdots & \vdots & & \vdots \\ 1 & -a_{2m_2,1} & \ldots & -a_{2m_2,n} \end{bmatrix} \begin{bmatrix} \beta_2 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} \leq \begin{bmatrix} \omega_1(a'_{10}) - \omega_1(a_{11}) \\ \vdots \\ \omega_1(a'_{10}) - \omega_1(a_{1m_1}) \\ \omega_2(a_{21}) \\ \vdots \\ \omega_2(a_{2m_2}) \end{bmatrix}$$

The sequence of k_1 equations:

$$\begin{gathered} \langle a'_{11} - a'_{10}, \alpha \rangle = \omega_1(a'_{10}) - \omega_1(a'_{11}) \\ \vdots \\ \left\langle a'_{1k_1} - a'_{10}, \alpha \right\rangle = \omega_1(a'_{10}) - \omega_1(a'_{1k_1}) \end{gathered}$$

can be used to reduce the dimensionality of the problem by k_1. This means that α now only has $n - k_1$ elements.

Once again, the aim is to locate all the vertices of the simplex defined by 2.33. This time, one seeks to find pairs (or sets of $k_2 + 1$) of active inequalities in the second set of equations (pertaining to the second support). If the number of active inequalities never exceeds $k_2 + 1$ in this section, then it is said that C_1, the level-k subface for the first support, cannot be extended. Recall that there will most likely be more than one version

of C_1 to test in such a way. If one particular version of C_1 is extendible, then that $C_1 + C_2$ combination is recorded. One then seeks to extend this $C_1 + C_2$ subface by searching for a compatible C_3 using a similar simplex to Equation 2.33. Once a set of $C_1 + C_2 + \ldots + C_r$ level-r subfaces has been found which satisfy Equation 2.29, then their corresponding α's and β's can be computed, and their volumes recorded using 2.32.

It is worth noting that the task of finding the level-k subfaces for each support can be quite time consuming. Often it is better to find the level-k subface for the first support only, and then progress immediately to extending the subface using the other supports (in their entirety). The reason this is often faster is that the simplex method's speed depends heavily on the dimension of the problem. Since with each step of the subface extension process, it is possible to reduce the dimensionality of the problem by the sum of the previous k_i's, it can be beneficial to find the level-k subface for the first support only, and leave the other supports untreated. This leads to an important operational note for using the simplex minimisation method: always sort the equations and their supports from first to last **in order of increasing complexity**. In other words, place the equation/support with the fewest terms/elements first, and that with the greatest number last. In this way, by the time the subface extension process reaches the final support (with the greatest number of inequalities to scan), the dimensionality has been greatly reduced.

At the end of the subface extension process, it is possible to form a binomial system from Equation 2.30 with $t = 0$ for each of the level-r subfaces. Each r-subface generates $\det(V(\alpha))$ solutions at $t = 0$, which can be traced to $t = 1$: the solutions of the final form of the random complex coefficient target system of polynomials.

2.4.2 A More Intuitive Method

The method of solving 2.29 given in Section 2.4.1 can be very fast when implemented, even on quite large systems. It is, however, very intricate to code. It also requires a simplex minimisation code which, although commonly available 'off the shelf,' will most likely have to be written from first principles to give access to all of the parts of the process which make the overall method so efficient. If one has access to a widely available package known as *Qhull* (implemented in Matlab as `convhulln.m`) which calculates the convex hull of groups of points in n dimensions, then it is possible to find the level-r subfaces of a system of equations quite easily. The following 'bootstrap' algorithm was developed specifically for use in solving the systems which appear in the following chapters.

Assume again that there are n equations with r distinct supports of multiplicity k_i for

$i = 1, \ldots, r$ with $\sum_{i=1}^{r} k_i = n$ and $1 \leq l \leq k_i$. Consider the definition of $\bar{Q}_i$ as given in Equation 2.21, as well as the Minkowski sum of hulls as given in Equation 2.17. Also assume again that each of the r supports contains the origin. The process is as follows:

1. lift each of the r supports using random lifting functions ω_i, and construct the $\bar{Q}_i$'s;
2. construct the complete Minkowski sum of these hulls $\bar{Q}_{tot} = \bar{Q}_1 + \bar{Q}_2 + \ldots + \bar{Q}_r$. This is an iterative process, and is straightforward to implement. One can either recompute the convex hull at each step, or once at the end (meaning Qhull is only called once in the entire process). The time taken by each approach varies based on the type of problem. Either way, at the end it is important that the total Minkowski sum be a convex polytope.
3. Once the vertices of each facet of the convex polytope $\bar{Q}_{tot}$ have been found, express the polytope in terms of a system of inequalities $Bx \leq c$. This can be achieved as follows:
   (a) find a location inside $\bar{Q}_{tot}$, say $\bar{q}$;
   (b) shift $\bar{Q}_{tot}$ by distance $\bar{q}$ so that $\bar{Q}_{tot}$ now contains the origin;
   (c) let M be a $(n + 1 \times n + 1)$ matrix whose rows represent the points at each vertex of a facet of $\bar{Q}_{tot}$. Each row of B is calculated by inverting the vertices $M^{-1}\mathbf{e}$ where $\mathbf{e}$ is a $(n + 1 \times 1)$ column vector of 1's;
   (d) c is then simply calculated as $c = \mathbf{1} + B\bar{q}$.
4. the rows of B are the outward pointing normals of each of the polyhedron's facets;
5. identify the lower hull simply by looking for all normals with a negative 'ω' component;
6. normalise each of the rows of B by its ω component, producing a list of *inner normals* to the polyhedron;
7. identify which of the facets of the lower hull correspond to the original supports, and which correspond to mixed cells. This is achieved by taking a row of B, say b, and forming the product:

$$\beta_i = \langle \bar{q}_{ij}, b \rangle \quad \bar{q}_{ij} \in \bar{Q}_i$$

   If there is an i for which β_i is different for every j, then b is an inner normal to a support, and not a mixed cell.

8. the remaining rows of B are now of the form $b_p = \{\alpha_p, 1\}$, where p refers to a particular mixed cell. Each row b_p will have $k_i + 1$ values of j which generate identical values of $\beta_i = \langle \bar{q}_{ij}, b_p \rangle$ in each support i. The identification of the mixed cells is complete.

Note that if a facet has more vertices than there are dimensions, Qhull will divide the facet up into smaller subfacets. Imagine the square base of a four-sided pyramid in three dimensions; QHull will divide the square base up into two equal triangles, and output six distinct facets. An unmentioned, but important part of the process above, is to re-combine these divided facets to capture the true number of vertices. Mixed cells usually have more vertices than there are dimensions.

2.4.3 Optimising the Lifting Values to Improve Numerical Stability

Tracking the solution paths which arise using the polyhedral method involves implementing Equation 2.28 numerically. The power to which the continuation parameter t is raised has a great impact on the numerical stability of the path following process. If the exponent ($\langle a_{ij}, \alpha \rangle + \omega_i(a_{ij}) - \beta_i$ term) is too small, then tracking precision will be lost where t is close to 0. Similarly, if the exponent is too large, then tracking precision will be lost as $t \to 1$. *Dynamic Lifting* (Mizutani *et al.*, 2007; Verschelde *et al.*, 1996) is one technique which can be applied to try to keep these exponents in a reasonable range, but is not practical beyond a certain mixed volume. Another technique which involves *balancing* the lifting values (Gao *et al.*, 2000) tries to make the best of a pre-existing subdivision which was most likely calculated using a random lifting. In Gao *et al.* (2000), a constant multiple is applied to all of the existing lifting values to ensure no exponents are smaller than a certain threshold value. A form of optimisation is then applied to the lifting values with the goal of reducing the magnitude of the largest exponents while preserving the structure of the subdivision. A slight variation on this was performed here. First, a constant multiple was applied to the entire lifting set to bring the largest exponent within a certain upper bound. Next, a form of optimisation was applied to the smallest exponents to raise them to a sufficient height. The following simple line-search based lifting value optimiser was written specifically for use in solving the systems which appear in the following chapters.

Imagine that for a particular set of equations, and a particular set of random lifting values, the subdivision is found to consist of S level-r subfaces. For each of the r equation supports, S pairs (or $k_i + 1$ sets) of support terms are formed. Some of these support terms will appear in more than one of the pairs. Form a collection of these terms for each

equation as:

$$\Phi_i = \bigcup_{s=1}^{S} \{a_i', a_i''\}_s \quad i = 1, \ldots, r$$

The objective function to be minimised can then be written as:

$$\text{Obj} = \sum_{i=1}^{r} \sum_{s=1}^{S} \sum_{a_{ij} \in Q_i \setminus \Phi_i} -\log\left(\langle a_{ij}, \alpha_s \rangle + \omega_i(a_{ij}) - \beta_{i_s}\right) \tag{2.34}$$

The variables used in the optimisation are the heights:

$$\omega_{ij} = \{\omega_i(a_{ij}) | a_{ij} \in \Phi_i\}$$

The effect of the logarithm is to heavily penalise exponents of t which are too close to zero. Using a simple gradient-based line search method is usually sufficient, as it leads to a convergence within a reasonable number of iterations. To increase numerical accuracy close to zero, the complex step method was used in computing gradients (Martins *et al.*, 2003; Ridout, 2009). Those exponents which are close to zero are pushed upwards in the process. The magnitude of the smallest exponent can usually be expected to at least triple using this method. It was discovered that a minimum exponent value of 0.35 sufficed for numerical stability. A lesser sensitivity to large exponent values was observed. Anything up to an exponent value of 20 was observed to cause little or no numerical instability.

2.5 Witness Sets

Sometimes the solutions to a system of polynomial equations are not all geometrically isolated (see fig. 2.1). Take the example:

$$f(\mathbf{x}) = \begin{cases} (x_1^2 + x_2^2 - 1)(3x_1^2 + x_2) \\ (x_1^2 + x_2^2 - 1)(x_1 - x_2) \end{cases} = \mathbf{0} \tag{2.35}$$

Clearly, there is a trivial solution at $\{0, 0\}$, as well as a Zero Dimensional Solution at $\{-1/3, -1/3\}$. Systems of polynomial equations can have positive dimensional solutions as well. The 'dimension' can be thought of as the number of parameters it would take to describe the line or surface of solutions in polynomial form. The example above has a solution set of dimension 1, which is a curve. In this case, it is simply the unit circle. The maximum dimension of any solution set is $d = n - 1$, where n is the number of variables (Verschelde, 2009). A direct application of polynomial continuation to this

problem will produce a number of singular and non-singular solutions, since all positive dimensional solutions are singular, and the continuation process will identify some points on the unit circle. Without prior intuition about where the solutions to this problem might fall, and of what type they might be, this blind application of continuation would not reveal much about the structure of the equations. Any singular solution found cannot be guaranteed to belong to a positive dimensional set, nor can it be determined what the dimension of that set might be, nor which other solutions might share that set. One further problem with this approach is that the Newton's method endgames used to locate solutions at the end of a continuation run do not work particularly well on singular solutions (convergence is less reliable, and limited to a linear rather than quadratic rate).

A good approach for determining the structure of a system of polynomial equations, **as well as finding the solutions themselves**, is to append a number of intersecting complex hyperplanes to the original problem (Sommese & Verschelde, 2000; Verschelde, 2009). Each new hyperplane brings with it a new variable. The number of hyperplanes required is equal to the dimension of the solution set being examined. A 1D solution set, as in the example above, would require the addition of a single hyperplane, and a single new variable. In general, the modified equations look like:

$$f'(\mathbf{x},\mathbf{z}) = \begin{cases} f_1(\mathbf{x}) + \sum_{j=1}^{d} \lambda_{1,j} z_j \\ \vdots \\ f_n(\mathbf{x}) + \sum_{j=1}^{d} \lambda_{n,j} z_j \\ a_1 + a_{1,1}x_1 + \cdots + a_{1,n}x_n + z_1 \\ \vdots \\ a_d + a_{d,1}x_1 + \cdots + a_{d,n}x_n + z_d \end{cases} \tag{2.36}$$

where d is the dimension of the solution set being scanned. The $\mathbf{z}$ terms are the introduced variables, and the $a's$ and $\lambda's$ are random complex numbers. These hyperplanes will intersect the positive dimensional solution set, and define a number of non-singular solutions on it. The number of these solutions corresponds to the degree of the positive dimensional solution set. Positive dimensional solutions are identified by looking for solutions to 2.36 with $\mathbf{z} = \mathbf{0}$. These solutions are said to form a Witness Set.

Once the solutions with $\mathbf{z} = \mathbf{0}$ have been identified, they are put aside and labelled as the witness set for the dimension d. If $d \geq 1$, then the number of hyperplanes still included in the problem can be reduced to gain access to lower dimension witness sets. This process is generally referred to as a Cascade of Homotopies. The cascade process is repeated for each value of d until $d = 0$ is reached. At each stage of the process, all

solutions with $\mathbf{z} = \mathbf{0}$ are labelled as witness sets for that particular level (value of d). The leftover solutions at the end are the zero-dimensional solutions which would have been identified using a direct application of continuation. To reduce the dimensionality, an homotopy of the form in Equation 2.37 is constructed and tracked from $t = 0 \rightarrow 1$.

$$h'(\mathbf{x}, \mathbf{z}) = \begin{cases} f_1(\mathbf{x}) + \sum_{j=1}^{d-1} \lambda_{1,j} z_j + (1-t)\lambda_{1,d} z_d \\ \vdots \\ f_n(\mathbf{x}) + \sum_{j=1}^{d-1} \lambda_{n,j} z_j + (1-t)\lambda_{n,d} z_d \\ a_1 + a_{1,1}x_1 + \cdots + a_{1,n}x_n + z_1 \\ \vdots \\ (1-t)(a_d + a_{d,1}x_1 + \cdots + a_{d,n}x_n) + z_d \end{cases} \tag{2.37}$$

During this process, the d^{th} intersecting hyperplane and the d^{th} z variable are removed from the system of equations.

There are two points worthy of note at this stage:

1. Not all solutions with $\mathbf{z} = \mathbf{0}$ are necessarily members of a witness set of the dimension currently under investigation. It is possible that such a solution actually lies on a higher dimensional solution set. If all higher dimension witness sets have already been identified, it is possible to determine if a point is actually a member of one of these higher sets using a *membership test*. This is why, when no prior information about a system of equations is available, it is prudent to begin with $d = n - 1$. An indication that a point is actually a member of a higher dimensional solution set is that the point is singular.

2. Members of a witness set for a particular dimension are not necessarily members of the same Irreducible Component. This could be the case if there are two or more separate curves/manifolds in the solution space at the same dimensionality. It is possible to determine which elements of a witness set are members of the same irreducible component using a process called Monodromy (Sommese & Wampler, 2005), discussed in Section 2.5.1.

An illustration of what a membership test involves is given in Figure 2.7. In this illustration, a single hyperplane L intersects two irreducible components at a total of five locations. Assume that these five points constitute the entire witness set for this dimension. The point x has been located as part of the cascade process, in a lower dimension. To determine if point x is actually a member of the witness set just described, plane L is translated such that it passes through x. If x was a member of the witness set (and hence

lay on one of the two irreducible components pictured), then one of the five witness set points would be tracked to the exact location of x as the intersecting plane was moved.

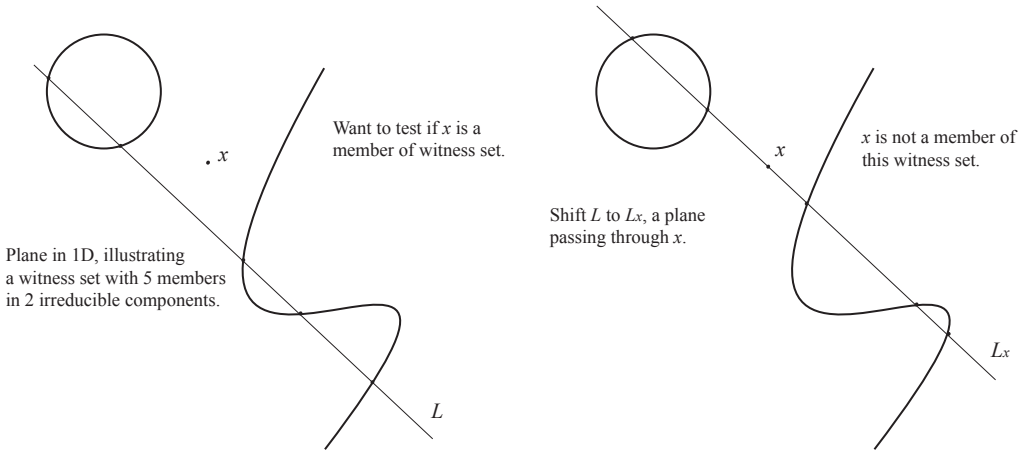


Figure 2.7: Illustration of membership test for a single point.

By way of an example, the decomposition of equation system 2.35 will be described in detail (a more involved example is given on p2028 of Sommese *et al.* (2001a)). Since only two variables are involved, start with $d = n - 1 = 1$. This involves the introduction of a single new variable z, and a single intersecting plane. With the introduction of the new variable, the system in terms of $\{x_1, x_2, z\}$ has a mixed volume of 12. Solving this system shows that four of the solutions go to infinity, six have $z \neq 0$ and two have $z = 0$. It can immediately be said that the witness set of dimension 1 has two members. This is not surprising, as the unit circle is a second order curve in one parameter. The six remaining solutions are then subjected to a dimension reduction homotopy, or cascade (Equation 2.37), which brings d to zero. This second homotopy shows that one solution goes to infinity, while five remain finite. Applying a membership test to these five remaining points shows that three actually lie on the unit circle, leaving the expected two solutions $\{0, 0\}$ and $\{-1/3, -1/3\}$. This is illustrated graphically in Figure 2.8.

2.5.1 Monodromy

Should a system of polynomial equations possess a positive dimensional solution set, then the solution of this system, augmented with an intersecting complex hyperplane (as in Equation 2.36), will consist of a number of witness points. In the case of Figure 2.7, five witness set points, lying on two irreducible components, are shown. Initially, it

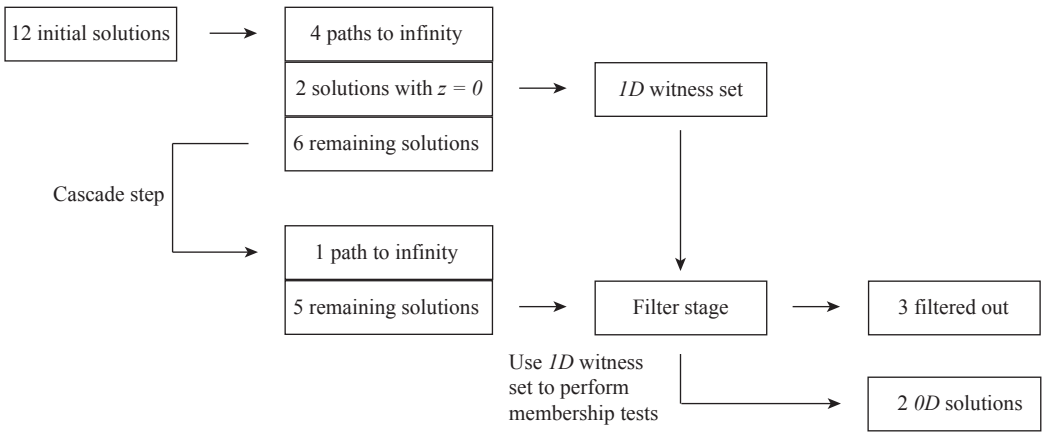


Figure 2.8: Reduction process for system of polynomial equations (Equation 2.35).

will not be known how many irreducible components are present, nor how many witness points lie on each. One way of determining which points lie on which irreducible component is to sample the irreducible components at a sufficient number of locations (perhaps by incrementally moving intersecting hyperplanes and tracking individual points) to construct a polynomial of the same degree as the irreducible component. This polynomial can then be used to directly test membership of other points (Sommese *et al.*, 2001a).

Another way to determine the finer structure of a system of equations is to employ a process known as *monodromy* (Sommese *et al.*, 2001b). This involves tracking witness points from one intersecting plane to another random plane, before permuting the members of each irreducible component by changing the complex coefficient often introduced to a linear homotopy to increase numerical accuracy in solution tracking (essentially the factor θ in Equation 2.16). These permuted solutions are then tracked back to the original intersecting plane. Members of the same irreducible component will permute amongst themselves only, which allows these components to be identified. If these witness points happen to be singular (if they have multiplicity > 1), they are more difficult to track along their irreducible components. Methods do exist for treating singular points of this type (Sommese *et al.*, 2002b), but these will not be covered here as no such points were discovered in the kinematic equations considered.

2.6 Solution to a Hinge Closure Problem

To illustrate the applicability of polynomial continuation to a simple mechanical problem, a planar 4-bar mechanism will be examined. This mechanism must be able to fold completely flat, with all of its bars collinear (a condition which requires that the sum of the lengths of two adjacent sides must be equal to the sum of the lengths of the other two sides). The layout is shown in Figures 2.9(a) and 2.9(b). This mechanism is actually a special type of 'hinge' which is symmetric about CD (a concept proposed by the author and Dr Simon Guest for connecting adjacent SAR panels in an articulated fashion following a visit to EADS Astrium in Portsmouth). The ends of two panels can be attached to the AB links, allowing the two panels to be parallel but separated by a distance of $2p$ when in the configuration pictured in Figure 2.9(a), and collinear when the hinge is folded flat. The hinge provides a way of attaching one panel to another while keeping the active surface (which faces inwards in the closed configuration) free from attachments. It can be shown that the only constraints necessary for operation are:

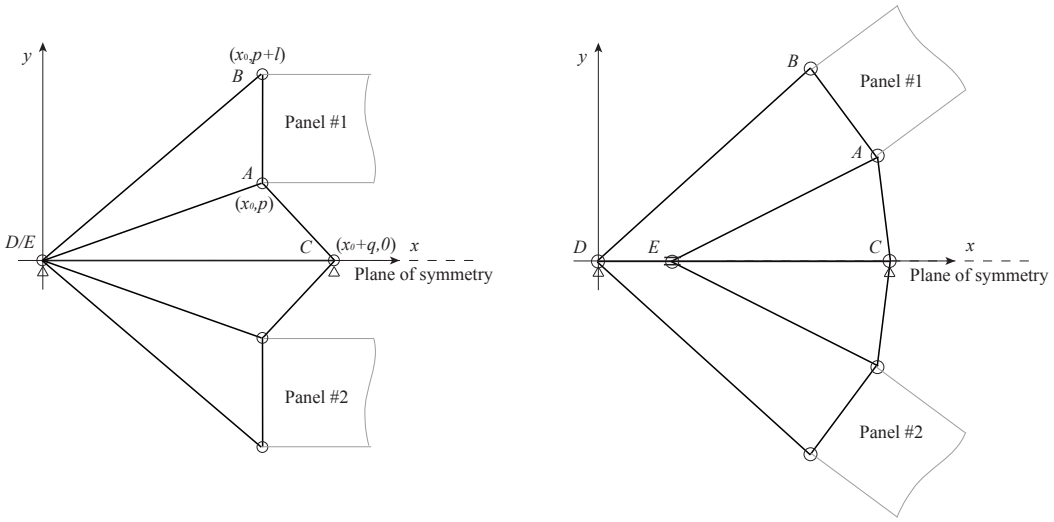
$$\begin{aligned} AC + CD &= AB + BD \\ AC &\leq AE \end{aligned} \tag{2.38}$$

The use of the second of Equations 2.38 can be avoided through careful choice of the parameter q (pictured). The aim of the design procedure is to specify p and l based on the required geometry, choose a value for q, and then solve for the lengths CD and BD. There are two unknowns in this problem, and as things stand, only one equation. In addition to $AC + CD = AB + BD$, equations can be written which relate to the lengths AC, BD and CD. These are:

$$\begin{aligned} AC^2 &= (x_0 - CD)^2 + p^2 \\ BD^2 &= x_0^2 + (p + l)^2 \\ CD &= x_0 + q \end{aligned}$$

The complete set of equations is given in 2.39.

$$\begin{aligned} AC + CD - l - BD &= 0 \\ AC^2 - (x_0 - CD)^2 - p^2 &= 0 \\ BD^2 - x_0^2 - (p + l)^2 &= 0 \\ CD - x_0 - q &= 0 \end{aligned} \tag{2.39}$$



(a) Initial layout of hinge.

(b) Hinge during operation. Notice that joint E is a combination slider/revolute.

Figure 2.9: Hinge design for continuation analysis.

This system has a total degree of four. It is good practice to remove any linear equations, if possible, by substitution. After the first and last equations of 2.39 have been removed, what is left is:

$$\begin{aligned}(BD + l - q - x_0)^2 - (q^2 + p^2) &= 0 \\ BD^2 - x_0^2 - (p + l)^2 &= 0\end{aligned} \tag{2.40}$$

which is written in terms of the unknowns BD and x_0. These equations still have a total degree of four, and also a mixed volume of four. Introducing a standard set of parameters which are known to work:

$$\begin{aligned}q &= 0.2 \\ p &= 0.2 \\ l &= 0.3\end{aligned}$$

gives the two finite solutions:

$$\begin{bmatrix} BD \\ x_0 \end{bmatrix} = \begin{bmatrix} -0.5179 \\ -0.1351 \end{bmatrix}, \begin{bmatrix} 0.7751 \\ 0.5922 \end{bmatrix}$$

Clearly the first solution is not physically realisable, but the second is. The remaining two solutions are at infinity. It is easy to see why the solution structure is as it is by looking at the curves representing Equations 2.40. This is shown in Figure 2.10. The two finite solutions are clearly visible, with the other two off at infinity.

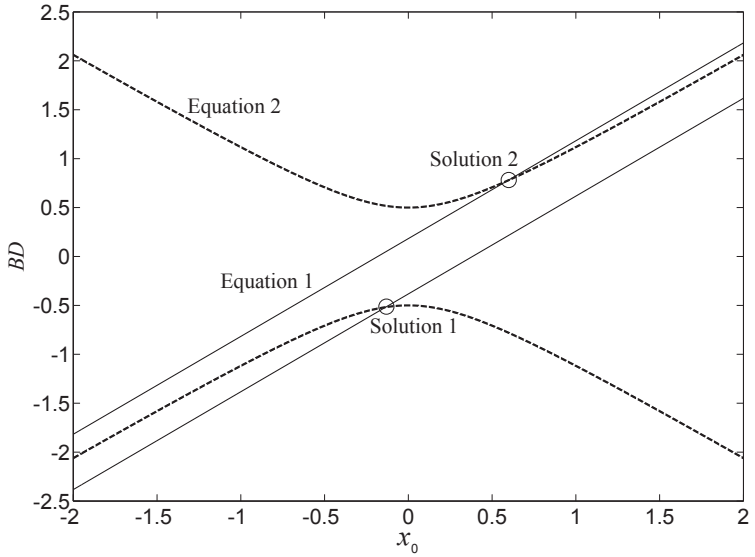


Figure 2.10: Solution paths for simple hinge design showing two solution locations.

While this example is simple, it demonstrates that the method of Polynomial Continuation can reduce the complexity of finding solutions to kinematic problems.

3. Plane Symmetric Bricard 6R Foldable Frames

In this chapter, a particular form of plane symmetric 6-bar folding linkage will be examined in some detail. Several different types of foldable frames, which in their deployed configurations form (often regular) polygons, have appeared in literature in the past 40 years. An early example appears in Crawford *et al.* (1975), in which an even number of bars are linked together in such a way that they can be folded into a tight bundle, and unfolded to form a regular polygon (in the 6-bar case of this, a three-fold symmetric linkage results (Chen *et al.*, 2004), while in the 4-bar case, a Bennett linkage is formed (Chen & You, 2006a,b)). 4-bar foldable frames have also been extensively examined recently (Chen, 2003; Chen & You, 2006a; Gan & Pellegrino, 2003). In Pellegrino *et al.* (2000), a new family of 6-bar foldable frames was proposed. A two-fold symmetric member of this family has been proposed as a support for a solar blanket, and its kinematics examined numerically (Gan & Pellegrino, 2005, 2006). Recently, this two-fold symmetric 6R foldable frame was identified as a special line *and* plane symmetric Bricard linkage (Chen & You, 2009). This particular variant does, however, suffer from problems with bifurcations (although certain designs avoid this). If one of the two planes of symmetry is removed, a mobile 6R frame which experiences fewer problems with bifurcations remains. An example is shown in Figure 3.1. The kinematics of this singly symmetric 6R foldable ring were studied recently in Hutt (2007).

This chapter seeks to extend the results of the previous work by: first identifying the frame as a plane symmetric Bricard linkage (Bricard, 1897); examining the nature of its mobility using a cascade of homotopies (Sommese & Verschelde, 2000) to identify positive dimensional solution sets; determining a range of design parameters for building feasible mechanisms; deriving a closed form expression for the linkage's kinematics; and finally employing polynomial continuation in an attempt to design a family of singly symmetric 6R foldable rings with certain desirable practical properties.

Sections 3.2-3.4 contain some analysis which is not directly applicable to the design

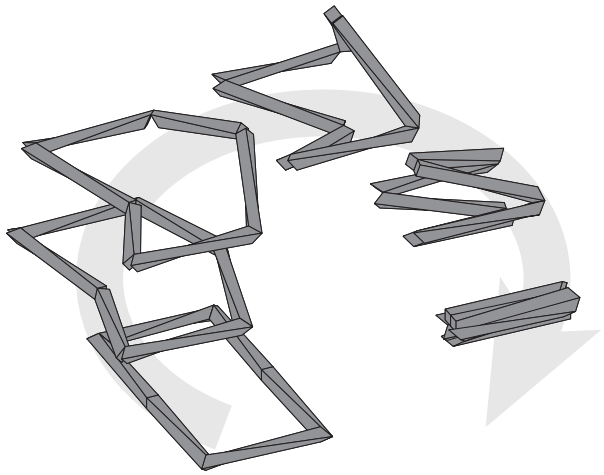


Figure 3.1: Folding process for a linkage with parameters $\alpha_1 = \pi/4, \alpha_2 = -\pi/4$ and $\gamma = \pi/2$. Each bar is shown as a twisted prismatic bar with square cross-section.

of linkages using continuation, but which nonetheless adds some important breadth of understanding of the plane symmetric 6R foldable frame. It is hoped that these sections will allow for a more informed interpretation of the results on the design process in Section 3.6. This kind of extended analysis will be omitted from the following chapters.

3.1 Frame Geometry and Definitions

A representation of the general plane symmetric Bricard linkage is given in figure 3.2. In this chapter, a special class of this linkage, which can be constructed using straight links forming a rectangle when deployed, is considered. The two-fold symmetric ring of Chen & You (2009) has two different bar lengths (l_1 and l_2), and the bars are all tilted from the vertical by a single angle μ. Square cross-sectioned, prismatic (i.e. untwisted) bars were used. By contrast, the 6R linkage considered in this chapter has all six bar lengths the same (l), and four separate bar tilt angles ($\alpha_1', \alpha_2', \beta_1$ and β_2), introducing a requirement that, if the bars have a square cross-section, some of the bars must be twisted in order to match the prescribed tilt angles at each end. The deployed linkage is a rectangular frame of breadth twice its height. A simplified diagram of the deployed linkage with all design parameters labelled is given in Figure 3.3, while a representation of the physical linkage is given in Figure 3.4, in which the twists in the bars are clearly visible. While a square cross-section is not required to construct the linkage, it does aid in visualising the bar twist.

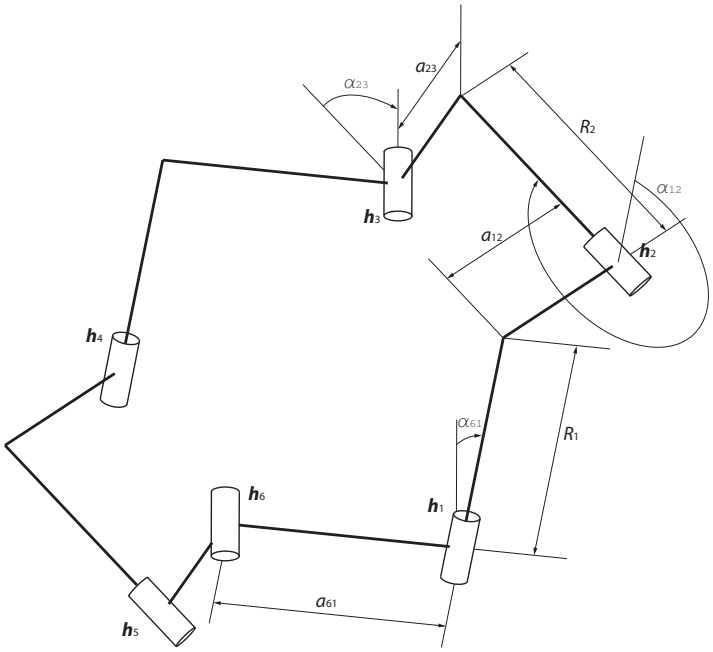


Figure 3.2: Plane symmetric Bricard linkage (case illustrated is from linkage with $\alpha_1 = \pi/4, \alpha_2 = -\pi/4$ and $\gamma = \pi/2$).

Because the linkage has only a single degree of freedom, its opening and closing (deployment/stowing) can be driven using any one of its six hinges. The plane symmetry means that the motion of two of the six hinges (referred to here as θ_{61} and θ_{12}) will be exactly replicated (in hinges θ_{45} and θ_{34} respectively). To track how the linkage moves from its deployed to stowed position, one can plot the hinge angle of one of these two hinges against the other. The relationship between the angle of opening of each of these two hinges is not simple, but can be found algebraically (see Section 3.5).

In linkage design, one adjusts a collection of design variables (lengths, angles, thicknesses etc.) such that the linkage moves in a way which is *as close as possible* to a desired pattern. This closeness can be specified in a number of ways. It might be that a certain part of the linkage must trace out a given curve in space during its unfolding. It might be that a part of the linkage never exceeds a given velocity, or that a joint never subtends more than a given angle. In the design of the plane-symmetric 6-bar linkage, one appropriate design criterion which suggests itself is the specification of a special relationship between θ_{61} and θ_{12} during the deployment phase. In a way, this is like specifying Precision Points, but in angle space. In the current chapter, three design variables (identified later) are used to specify the linkage. In order to construct a square system of equations

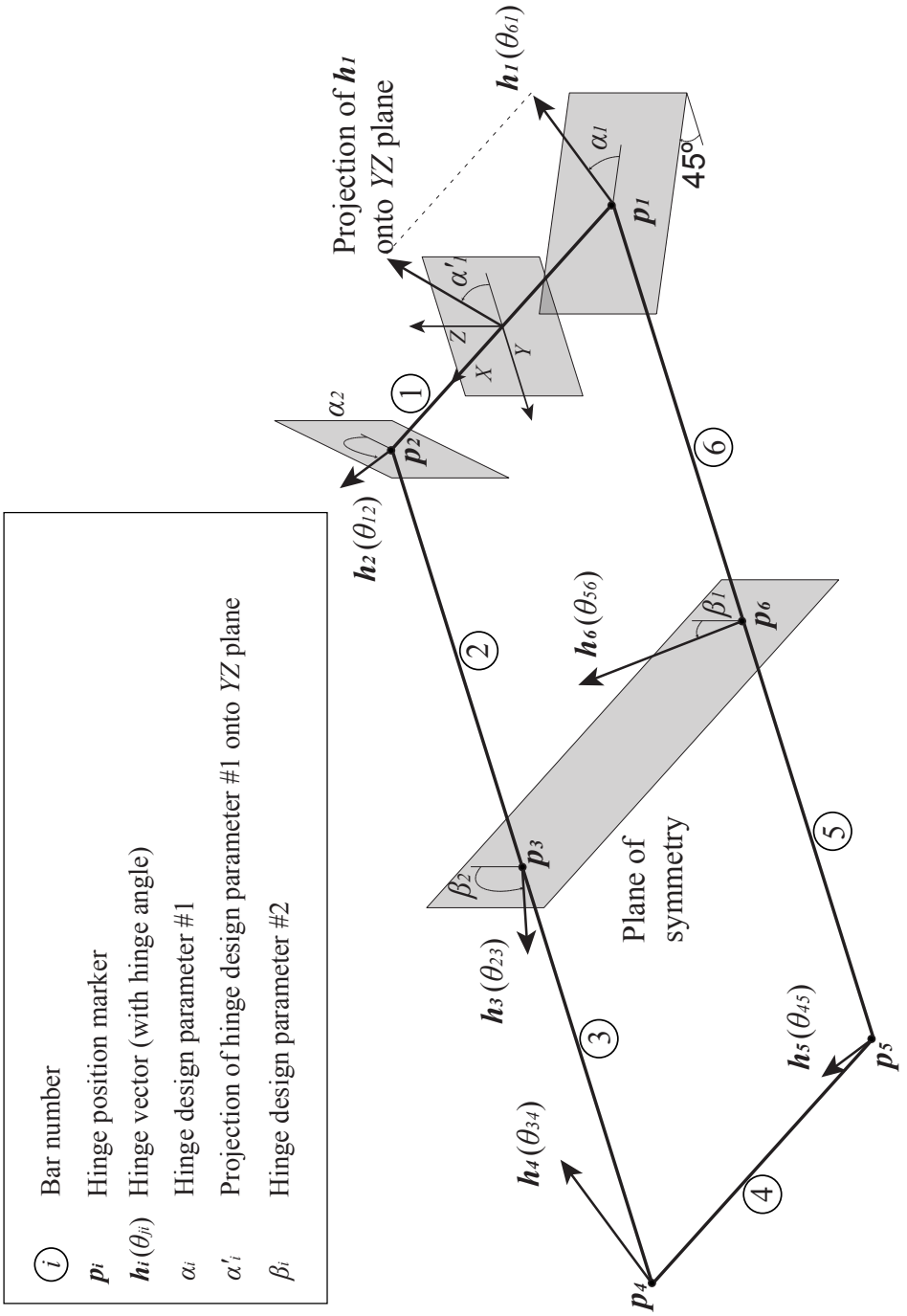


Figure 3.3: Design variables of the singly-symmetric 6R linkage with plane of symmetry shown. The global axes are indicated in the centre of bar 1. This is a special subset of the Bricard plane symmetric case in which all the hinges lie in the same plane in the 'deployed' configuration.

one can specify three $\{\theta_{61}, \theta_{12}\}$ pairs, and use these to form three closure equations to solve. In this way, three precision points can be used to guide the linkage from the stowed to the deployed position along a desirable path. This will be covered in more detail in Section 3.6.

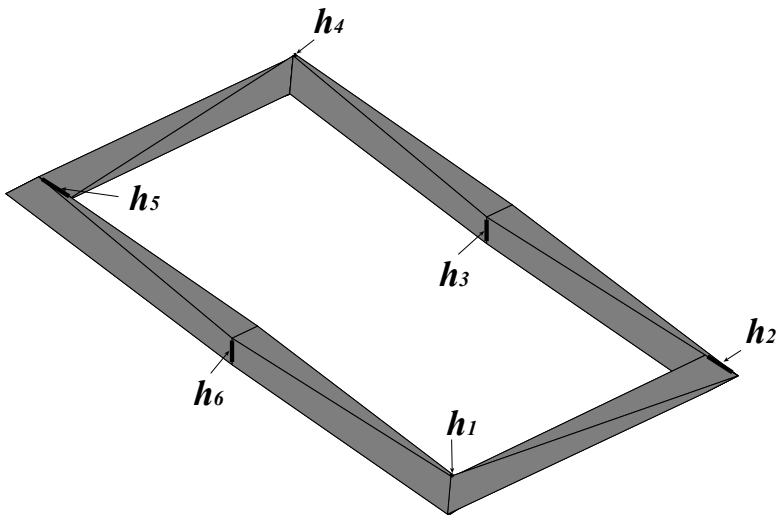


Figure 3.4: Example of singly-symmetric 6R foldable linkage with visible bar twist for $\alpha_1 = \pi/4, \alpha_2 = -\pi/4$ and $\gamma = \pi/2$. Notice that even the offsets in the hinge locations caused by the finite thickness of the bars does not cause the linkage to behave significantly differently.

There are six hinges (labelled $\mathbf{h}_1 - \mathbf{h}_6$), each with a single rotational degree of freedom, connecting the bars in a closed loop. The plane of symmetry is preserved through the folding motion. This plane is parallel to the XZ plane in the fully deployed/open configuration, shown in Figure 3.3. The plane contains the points $\mathbf{p}_6$ and $\mathbf{p}_3$ ($\mathbf{p}_i$ are position vectors), and the vectors $\mathbf{h}_6$ and $\mathbf{h}_3$, which are inclined to the Z axis by angles β_1 and β_2 respectively. Also when deployed, hinges $\mathbf{h}_1$ and $\mathbf{h}_2$ lie in planes rotated from the YZ plane by 45° about the Z axis. The angles these hinges form to the horizontal can be specified in two important ways. When constructing physical models of the plane-symmetric 6-bar, the most intuitive form is obtained by taking the projection of the hinges onto the YZ plane, and considering the angle formed between that projection and the XY plane, labelled here as α_1' and α_2'. This projection is shown in Figure 3.3. This definition of hinge inclination to the vertical is more intuitive as it specifies the tilt angle that a square cross-sectioned bar would need to form with the horizontal before cuts at 45° are made. However, future mathematical results are simplified by directly taking the angle between the XY plane and the hinge vectors, written as α_1 and α_2 here. Relationships between

the two α definitions can be constructed as:

$$\begin{aligned}
\tan\alpha &= \frac{1}{\sqrt{2}}\tan\alpha' \\
\Rightarrow \sin\alpha &= \frac{\sin\alpha'}{\sqrt{\sin^2\alpha' + 2\cos^2\alpha'}} \\
\Rightarrow \sin^2\alpha &= \frac{\sin^2\alpha'}{1+\cos^2\alpha'} \\
\cos^2\alpha &= \frac{2\cos^2\alpha'}{1+\cos^2\alpha'}
\end{aligned}$$

At present, there are four angular parameters (either $\{\alpha_1', \alpha_2', \beta_1, \beta_2\}$ or $\{\alpha_1, \alpha_2, \beta_1, \beta_2\}$) which define the linkage. In order for all the bars to be parallel when fully stowed, the hinge vectors $\mathbf{h}_6$ and $\mathbf{h}_3$ must also be parallel in the fully folded configuration since they must be perpendicular to both the normal to the plane of symmetry, and the ends of the bars to which they are attached. In general, $\mathbf{h}_6$ and $\mathbf{h}_3$ will not be parallel when folded for a given choice of β_1 and β_2. The parallel condition can be enforced by specifying a simple linear relationship:

$$\begin{aligned}
\beta_1 &= 2\alpha_1' - \gamma \\
\beta_2 &= \pi - 2\alpha_2' + \gamma
\end{aligned} \tag{3.1}$$

where an extra variable, γ, has been introduced to replace β_1 and β_2. Also note that it is the (projected) α_1' and α_2' definitions which have been used here. Three variables ($\{\alpha_1', \alpha_2', \gamma\}$ or equivalently $\{\alpha_1, \alpha_2, \gamma\}$) remain to specify the linkage.

It can be shown that hinges 1 and 2 have maximum opening angles, as described in Equation 3.2.

$$\begin{aligned}
\theta_{61max} &= \pi - 2\tan^{-1}\left[\frac{\tan\ \alpha_1'}{\sqrt{2+\tan^2\alpha_1'}}\right] = \pi - \cos^{-1}(\cos^2\alpha_1') \\
\theta_{12max} &= \pi - 2\tan^{-1}\left[\frac{\tan\ \alpha_2'}{\sqrt{2+\tan^2\alpha_2'}}\right] = \pi - \cos^{-1}(\cos^2\alpha_2')
\end{aligned} \tag{3.2}$$

At these maximum angles, bars 6 and 2 are both colinear with bar 1, indicating full folding of the linkage. These definitions will be useful for simulation purposes later.

3.2 Expression in Terms of Denavit-Hartenberg Parameters

A good rule of thumb for determining the number of degrees of freedom (DOF) in a mechanism, known as the Kutzbach Criterion, is given in Equation 3.3 (to be found in Freudenstein & Maki (1979)).

$$F = \lambda(l - j - 1) + \sum_{i=1}^{j} f_i \tag{3.3}$$

Here, F is the number of degrees of freedom, l is the number of links present, j is the number of binary joints, f_i is the number of degrees of freedom in the i^{th} joint (eg. 3 for ball joint, 1 for hinge) and λ is the number of spatial degrees of freedom available to the mechanism (3 for planar mechanisms, and 6 for 3D). A joint connecting $n > 2$ links is counted as $(n - 1)$ binary joints. Equation 3.3 is only a general indication of the degrees of freedom a mechanism is likely to possess, as it often happens that a mechanism may in fact have a greater number of degrees of freedom because of certain special geometric properties. Phillips (1984) states this as "All geometric specialities, whether consciously introduced by the designer or occurring accidentally, give rise to special extra mobilities M_s: *these always increase the apparent mobility* M_a *of mechanism, never decrease it*." These geometric properties are very difficult to generalise, and will not be addressed in detail here. For further reading on this topic, see both Phillips (1984) and Phillips (1990), which form an extended work dealing primarily with the identification and classification of these geometric degrees of freedom studied from a screw theory perspective.

A blind application of Equation 3.3 indicates that the minimum number of links required to be arranged in a loop using single DOF joints for a positive mobility to arise is seven. In general, this is true, and seven bars arranged to form a loop using hinges will produce a linkage with a mobility of one. Several loops possessing fewer bars but with positive mobility due to special geometric properties have been identified. The Bennett Linkage is a mobile 4-bar loop (Chen & You, 2005). The Bennett Linkage can be used to generate a 5-bar loop known as the Goldberg 5R linkage (Chen, 2003). There are several known mobile 6-bar loops, which can be synthesised in a number of ways (including combining Goldberg linkages (Chen & You, 2007)). Other forms of foldable 6-bar linkages have been discovered more recently (Baker, 2005, 2006; Chen & You, 2008). Arguably the most famous of the 6-bar linkages are the Bricard Linkages (Baker, 1980), of which there are six subtypes, being the general Line Symmetric case, the gen-

eral plane-symmetric case, the trihedral case, the line-symmetric octahedral case, the plane-symmetric octahedral case and the doubly collapsible octahedral case. The 6-bar linkage of this chapter happens to be the plane-symmetric Bricard case. A representation of this case, with the Denavit-Hartenberg parameters labelled, is given in Figure 3.2. The linkage is symmetric about the plane passing through hinges 6 and 3. The defining parameters are:

$$\begin{aligned} &a_{61} = a_{65}\ ,\ a_{12} = a_{54}\ ,\ a_{23} = a_{43} \\ &\alpha_{61} + \alpha_{65} = \alpha_{12} + \alpha_{54} = \alpha_{23} + \alpha_{43} = \pi \\ &R_6 = R_3 = 0\ ,\ R_1 = R_5\ ,\ R_2 = R_4 \end{aligned} \tag{3.4}$$

It is possible to derive a relationship between the design variables α_1, α_2, β_1 and β_2 of the singly symmetric 6R foldable ring, and the Denavit-Hartenberg parameters. This is achieved by solving a series of simple linear equations which arise from the linkage's geometry. Start with the link between hinges 6 and 1. The Bricard linkage joints must lie on a line which passes through $\mathbf{p}_i(0)$ and is parallel to $\hat{\mathbf{h}}_i(0)$, where the notation (0) represents positions and vectors in the deployed configuration, and $\hat{\mathbf{h}}$ represents a unit vector:

$$\begin{aligned} \mathbf{p}_6 &= \mathbf{p}_6(0) + t_{l6}\hat{\mathbf{h}}_6(0) \\ \mathbf{p}_1 &= \mathbf{p}_1(0) + t_{l1}\hat{\mathbf{h}}_1(0) \end{aligned}$$

Here, t_{l6} and t_{l1} are unknown scaling factors. It is apparent that $R_6 = 0$, but also that a_{61} is non-zero. a_{61} is perpendicular to both $\hat{\mathbf{h}}_6(0)$ and $\hat{\mathbf{h}}_1(0)$, so define a new vector of unit length as:

$$\mathbf{a}'_{61} = \hat{\mathbf{h}}_6(0) \times \hat{\mathbf{h}}_1(0)$$

The length of a_{61} is unknown, so a third scaling factor, t_{a61}, is introduced. The governing equation is given in 3.5.

$$\begin{aligned} \mathbf{p}_6 + t_{a61}\mathbf{a}'_{61} &= \mathbf{p}_1 \\ \Rightarrow \mathbf{p}_6(0) + t_{l6}\hat{\mathbf{h}}_6(0) + t_{a61}\mathbf{a}'_{61} &= \mathbf{p}_1(0) + t_{l1}\hat{\mathbf{h}}_1(0) \end{aligned} \tag{3.5}$$

This can easily be solved to give t_{l6}, t_{l1} and t_{a61}, and hence $\mathbf{p}_6$, $\mathbf{p}_1$ and $a_{61} = |t_{a61}\mathbf{a}'_{61}|$.

Moving on to the next link (between hinges 1 and 2), define:

$$\mathbf{p}_2 = \mathbf{p}_2(0) + t_{l2}\hat{\mathbf{h}}_2(0)$$

as well as:

$$\mathbf{a}'_{12} = \hat{\mathbf{h}}_1(0) \times \hat{\mathbf{h}}_2(0)$$

Since R_1 is non-zero, introduce the unknown t_{R1} to the governing Equation 3.6.

$$\begin{aligned} \mathbf{p}_1 + t_{R1}\hat{\mathbf{h}}_1(0) &= \mathbf{p}_2 + t_{a12}\mathbf{a}'_{12} \\ \Rightarrow \mathbf{p}_1 + t_{R1}\hat{\mathbf{h}}_1(0) &= \mathbf{p}_2(0) + t_{l2}\hat{\mathbf{h}}_2(0) + t_{a12}\mathbf{a}'_{12} \end{aligned} \tag{3.6}$$

From this t_{l2}, t_{R1} and t_{a12} can be found, which in turn gives $\mathbf{p}_2$, $R_1 = t_{R1}$ and $a_{12} = |t_{a12}\mathbf{a}'_{12}|$.

Finally, consider the link between hinges 2 and 3 by defining:

$$\mathbf{p}_3 = \mathbf{p}_3(0) + t_{l3}\hat{\mathbf{h}}_3(0)$$

as well as:

$$\mathbf{a}'_{23} = \hat{\mathbf{h}}_2(0) \times \hat{\mathbf{h}}_3(0)$$

and Equation 3.7.

$$\begin{aligned} \mathbf{p}_2 + t_{R2}\hat{\mathbf{h}}_2(0) &= \mathbf{p}_3 + t_{a23}\mathbf{a}'_{23} \\ \Rightarrow \mathbf{p}_2 + t_{R2}\hat{\mathbf{h}}_2(0) &= \mathbf{p}_3(0) + t_{l3}\hat{\mathbf{h}}_3(0) + t_{a23}\mathbf{a}'_{23} \end{aligned} \tag{3.7}$$

From this t_{l3}, t_{R2} and t_{a23} can be found, which in turn gives $\mathbf{p}_3$, $R_2 = t_{R2}$ and $a_{23} = |t_{a23}\mathbf{a}'_{23}|$.

The important parameters are:

$$\frac{a_{61}}{l} = \sqrt{1 - \frac{4\cos^2(\alpha_1)}{2\sqrt{2}\sin(2\alpha_1)\sin(2\beta_1) + (3\cos(2\alpha_1) - 1)\cos(2\beta_1) + \cos(2\alpha_1) + 5}}$$

$$\frac{a_{12}}{l} = 2\sqrt{\frac{\sin^2(\alpha_1 - \alpha_2)}{2\cos(2\alpha_1) + 4\sin^2(\alpha_1)\cos(2\alpha_2) + 6}}$$

$$\frac{a_{23}}{l} = \sqrt{1 - \frac{4\cos^2(\alpha_2)}{-2\sqrt{2}\sin(2\alpha_2)\sin(2\beta_2) + (3\cos(2\alpha_2) - 1)\cos(2\beta_2) + \cos(2\alpha_2) + 5}}$$

and

$$\frac{R_1}{l} = \sqrt{2}\left(\frac{4\cos(\alpha_1)}{2\sqrt{2}\sin(2\alpha_1)\sin(2\beta_1) + (3\cos(2\alpha_1) - 1)\cos(2\beta_1) + \cos(2\alpha_1) + 5} \cdots - \frac{2(\cos(\alpha_1) + \sin(\alpha_1)\sin(\alpha_2)\cos(\alpha_2))}{\cos(2\alpha_1) + 2\sin^2(\alpha_1)\cos(2\alpha_2) + 3}\right)$$

$$\frac{R_2}{l} = \sqrt{2}\sec(\alpha_2) \cdot \left(-\frac{4\cos^2(\alpha_2)}{-2\sqrt{2}\sin(2\alpha_2)\sin(2\beta_2) + (3\cos(2\alpha_2) - 1)\cos(2\beta_2) + \cos(2\alpha_2) + 5} \cdots + \frac{\sin(2\alpha_1)\sin(2\alpha_2) + 4\cos(2\alpha_2) + 4\sin^2(\alpha_2)\cos(2\alpha_1) + 8}{2\cos(2\alpha_1) + 4\sin^2(\alpha_1)\cos(2\alpha_2) + 6} - 1\right)$$

The twist angles can be simply determined as:

$$\begin{aligned}\alpha_{61} &= \cos^{-1}\left(\langle\hat{\mathbf{h}}_6(0), \hat{\mathbf{h}}_1(0)\rangle\right)\\ \alpha_{12} &= \cos^{-1}\left(\langle\hat{\mathbf{h}}_1(0), \hat{\mathbf{h}}_2(0)\rangle\right)\\ \alpha_{23} &= \cos^{-1}\left(\langle\hat{\mathbf{h}}_2(0), \hat{\mathbf{h}}_3(0)\rangle\right)\end{aligned}$$

As a numerical example, consider the case of $\alpha_1 = \pi/4, \alpha_2 = -\pi/4$, and $\beta_1 = \beta_2 = 0$ ($\gamma = \pi/2$). This can be written in terms of Denavit-Hartenberg parameters for the plane symmetric case:

$$\begin{array}{ccc}\frac{a_{61}}{l} = \frac{1}{\sqrt{2}} & \frac{a_{12}}{l} = \sqrt{\frac{2}{3}} & \frac{a_{23}}{l} = \frac{1}{\sqrt{2}}\\ & \frac{R_1}{l} = \frac{2}{3} & \frac{R_2}{l} = -\frac{2}{3}\\ \alpha_{61} = \frac{\pi}{4} & \alpha_{12} = \frac{2\pi}{3} & \alpha_{23} = \frac{3\pi}{4}\end{array}$$

Figure 3.2 represents this case. A real model with the same parameters is shown during its folding process in Figure 3.6.

3.3 Examination of Mobility using a Cascade of Homotopies

It is possible to construct a set of loop closure equations by defining coordinate systems attached to each link and then deriving the *transfer matrices* which describe the transformation from one coordinate system to the next. This method is illustrated in Chen & You (2009) and Gan & Pellegrino (2006), where it is used to simulate the motion of closed loop linkages, and to study their bifurcations. A transfer matrix is typically 4×4, and consists of a 3×3 rotation matrix, say R, and a 3×1 translation vector, say $\mathbf{v}$. These parts are arranged as:

$$T = \begin{bmatrix} & R & & \mathbf{v}\\ 0 & 0 & 0 & 1\end{bmatrix}$$

If a coordinate system is attached to the end of a link in a linkage, then a transfer matrix can be used to rotate and translate it to the location of the coordinate system attached to an adjacent link in a single operation. Repeating this operation around a linkage which is also a closed loop will eventually lead back to the original link. Mathematically, this can be expressed as:

$$F = T_{61}T_{56}T_{45}T_{34}T_{23}T_{12} - I = 0$$

where T_{ab} defines the transfer between the coordinate system attached to link a to that attached to b. The coordinate system at each joint is aligned so that the z-axis is aligned with the hinge axis. Before each translation, the x-axis is rotated such that it points along the current bar towards the next joint. As there are single degree of freedom connections between the links, each transfer matrix can be separated into a part which deals only with rotation about the z-axis, L_3, and a part which relates to the unchanging geometry of the link, T^L_{ab} (which shifts the coordinate system from the closest end of one hinge to the closest end of the next), and $T_{ab} = T^L_{ab} L_3(\theta_{ab})$. The equations above then become:

$$F = T^L_{61} L_3(\theta_{61}) T^L_{56} L_3(\theta_{56}) T^L_{45} L_3(\theta_{45}) T^L_{34} L_3(\theta_{34}) T^L_{23} L_3(\theta_{23}) T^L_{12} L_3(\theta_{12}) - I = 0 \quad (3.8)$$

Explicitly, the transfer matrices for each of the six links of the plane symmetric 6R foldable frame are:

$$\begin{aligned}
T^L_{12} &= L_1(\beta_2) M L_3(\pi/4) L_1(\alpha_2 - \pi/2) \\
T^L_{23} &= L_1(\pi/2 - \alpha_2) L_3(\pi/4) M L_1(-\beta_2) \\
T^L_{34} &= L_1(\pi/2 - \alpha_1) L_3(\pi/4) M L_3(\pi/4) L_1(\alpha_2 - \pi/2) \\
T^L_{45} &= L_1(-\beta_1) M L_3(\pi/4) L_1(\alpha_1 - \pi/2) \\
T^L_{56} &= L_1(\pi/2 - \alpha_1) L_3(\pi/4) M(\pi/4) L_1(\beta_1) \\
T^L_{61} &= L_1(\pi/2 - \alpha_2) L_3(\pi/4) M L_3(\pi/4) L_1(\alpha_1 - \pi/2)
\end{aligned} \quad (3.9)$$

where L_1 is a rotation about the x-axis, and M defines a translation of length l along each link:

$$M = \begin{bmatrix} 1 & 0 & 0 & -l \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The single off-diagonal entry in matrix M has a negative sign because the effect of applying M to a point in space is to rewrite that point as a location **in the basis of the new, translated coordinate system**. Shifting a coordinate system in the positive x-direction requires the inclusion of a negative term in the $(1, 4)$ location of matrix M. The matrix Equation 3.8 can be separated into six individual equations which together ensure loop closure. As in Gan & Pellegrino (2006), the strictly upper triangular part of this matrix equation provides the six independent scalar equations necessary to form a square

system:

$$\mathbf{F}(\theta_{12},\theta_{23},\theta_{34},\theta_{45},\theta_{56},\theta_{61}) = \left\{ \begin{array}{c} F_{1,2} \\ F_{1,3} \\ F_{1,4} \\ F_{2,3} \\ F_{2,4} \\ F_{3,4} \end{array} \right\} = \mathbf{0} \tag{3.10}$$

Once the loop closure equations have been formed, the motion of the linkage can be simulated using a type of predictor-corrector approach detailed in Gan & Pellegrino (2006) and Chen & You (2009). One particularly useful by-product of this method is a matrix whose singular values can be used to examine the linkage's mobility at each point of the unfolding process.

The plane-symmetric 6-bar ring can be described as an Overconstrained Mechanism, or as possessing a geometric degree of freedom. This implies that it does not satisfy the Kutzbach Criterion (Freudenstein & Maki, 1979), which must be due to some special feature of the linkage's geometry, in this case its symmetry. It is possible to determine mathematically if a linkage is likely to possess a geometric degree of freedom, and if so, of what degree, and in how many disconnected sets, by using the method of witness sets described in Sommese *et al.* (2001a), Sommese & Verschelde (2000) and introduced in Section 2.5. The method can be used to find curves of solutions through the solution space.

The loop closure equation can be simplified by assuming that hinge angles reflected in the plane of symmetry will be equal, as:

$$\begin{aligned} \theta_{12} &= \theta_{34} \\ \theta_{45} &= \theta_{61} \end{aligned}$$

Finally, the maximum degree of the resulting polynomial closure equations can be reduced by rearrangement into the form of Equation 3.11.

$$\begin{aligned} F' =& L_3(\theta_{61})T_{34}^L L_3(\theta_{12})T_{23}^L L_3(\theta_{12})T_{12}^L - \ldots \\ & (T_{45}^L)^{-1} L_3(-\theta_{56})(T_{56}^L)^{-1} L_3(-\theta_{61})(T_{61}^L)^{-1} L_3(-\theta_{12}) = \mathbf{0} \end{aligned} \tag{3.11}$$

which is written in terms of the four remaining unknowns $\{\theta_{12},\theta_{23},\theta_{56},\theta_{61}\}$. Equation 3.11 can be written in pure polynomial form by replacing the trigonometric functions of the four unknowns with new variables by setting $\cos(\theta_{ij}) = C_{ij}$ and $\sin(\theta_{ij}) = S_{ij}$. Four elements from the strictly upper triangular part of Equation 3.11 can be chosen, and

augmented with standard trigonometric identities relating the new variables, to construct a system of equations of the form:

$$\left.\begin{array}{l} C_{12}^2 + S_{12}^2 - 1 \\ C_{23}^2 + S_{23}^2 - 1 \\ C_{56}^2 + S_{56}^2 - 1 \\ C_{61}^2 + S_{61}^2 - 1 \\ F'_{1,2} \\ F'_{1,3} \\ F'_{2,3} \\ F'_{3,4} \end{array}\right\} = \mathbf{0} \tag{3.12}$$

Equation 3.12 defines a relationship between each of the hinge angles. It is a system of polynomial equations in which the coefficients are written in terms of the 6-bar design parameters $\alpha_1, \alpha_2, \beta_1, \beta_2$ and the bar length l. It has a Mixed Volume (Gao & Li, 2003) of 176, which means there is a tight upper bound of 176 on the number of finite solutions. If this is, in fact, an over-constrained mechanism, then it can be expected that positive dimensional solution sets will be present. A positive dimensional solution set is a continuum of solutions which may exist on a line, plane, or higher dimensional manifold. Often, more than one curve or manifold of solutions will be present in a system of equations. If these curves do not intersect one-another at any point, they are known as Irreducible Components. The appearance of positive dimensional solution sets in the closure equations of a linkage is a sign that the linkage may actually be mobile. The dimensionality of the solution set corresponds to the degree of freedom of the linkage. For example, if a one-dimensional solution set is found, then it is possible that the linkage will be mobile with a single degree of freedom. Since Equation 3.12 contains four unknowns, it is possible that there could be as high as a $n - 1 = 3$-dimensional solution set. It is known *a priori*, however, that no three or two-dimensional solution sets are present. After introducing a single, random complex intersecting hyperplane into the equations (increasing the mixed volume to 400), and solving the resulting system using standard continuation methods, it was discovered that Equation 3.12 has a one-dimensional solution set whose witness set contains 28 elements. This first step also generated a further 246 solutions which do not lie on a one-dimensional curve of solutions. To determine which of these extra solutions are actually zero-dimensional geometrically isolated points, a *cascade* step is required (Sommese & Verschelde, 2000). The cascade involves gradually removing the intersecting plane, introduced earlier, from the equations by way of a new homotopy. Using this method a zero-dimensional solution set, also with 28 members, was found. The mem-

bers of the witness set were all found to belong to the same irreducible component which must, therefore, represent the single mobile path known to exist in the real linkage (an algebraic curve of degree 28 in the solution space). The cascade process is represented in Figure 3.5.

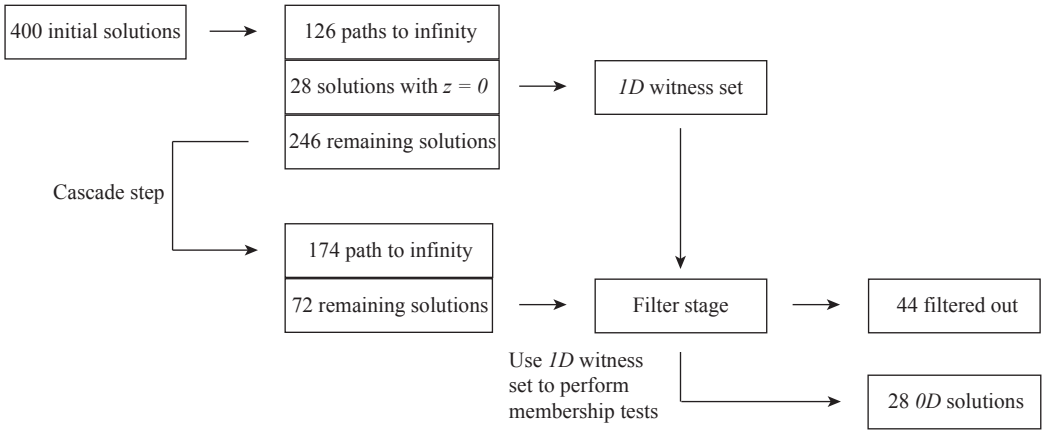


Figure 3.5: Homotopy cascade used to determine 1D and 0D witness sets for 6R linkage.

It has been shown that the plane-symmetric 6-bar linkage has a single irreducible component in dimension-1, which has 28 members when the closure equations are posed as above. Since it is known that this linkage has a single geometric degree of freedom, it can be stated that this irreducible component of degree 28 is responsible for its mobility. The fact that a linkage's closure equations contain a positive dimensional solution set does not, in general, prove that a linkage will actually possess any mobility, but suggests that it may be possible. To prove that a linkage does have a degree of freedom would require showing that there is at least one irreducible component in purely real space, and that at least part of this component lies in a feasible region of the linkage's parameters.

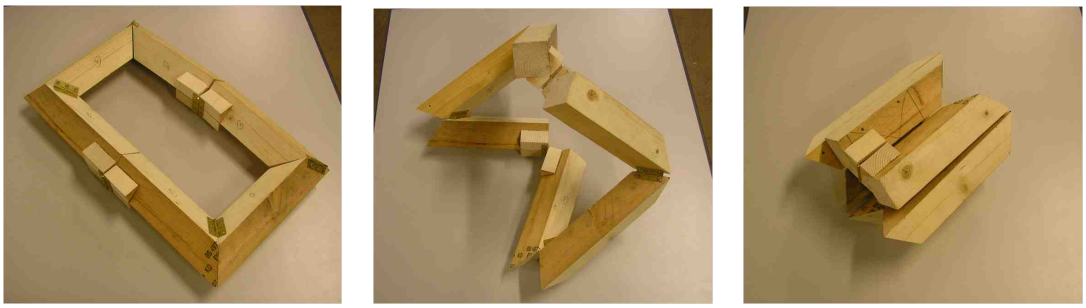


Figure 3.6: The folding of a wooden model with $\alpha_1' = \pi/4,\ \alpha_2' = -\pi/4$, and hence $\alpha_1 = \tan^{-1}(1/\sqrt{2}),\ \alpha_2 = -\tan^{-1}(1/\sqrt{2})$.

3.4 Finding a Feasible Region of Design Parameters

Only certain combinations of the three design parameters $\{\alpha_1, \alpha_2, \gamma\}$ (as described in Section 3.1) will lead to a linkage which behaves in a way which is likely to be desirable. The most fundamental of the requirements of a linkage of the type under consideration (plane symmetric Bricard foldable frame) are:

- linkage does not bifurcate at any point;
- linkage unfolds continuously and smoothly from closed to open configuration;
- hinge angles are allowed to lie only in the range spanned by the same hinge's angle when deployed, and the angle when stowed;
- no two bars intersect during the unfolding process.

The satisfaction of the first three points ensures the satisfaction of the fourth, and so this fourth point is not considered here. At present it is not clear why this automatic satisfaction of the fourth point occurs in practice.

Since there are only three design variables, one can be held constant, and a 2-dimensional plot constructed depicting a *feasible* space in terms of the other two. Holding γ constant, a set of feasibility *contour* lines can be plotted, as in Figure 3.7. The γ dependence of the feasible region is indicated in Figure 3.8.

Points of note about these feasibility graphs are:

- there is always a plane of symmetry defined by $\alpha_1 = \alpha_2$;
- the range of values permissible for α_1 and α_2 is $[-\pi/2, \pi/2]$ (outside this range proper linkage closure does not occur);
- the range of values permissible for γ is $(-\pi, \pi]$;
- plots for the range $\gamma \in (-\pi, 0]$ can be obtained by using $\gamma = \gamma_0 - \pi$, where γ_0 is in the range $(0, \pi]$.

The loop closure equations (Equation 3.10) were used as the basis of construction for each of the contour lines in Figure 3.7. Many combinations of design variables will lead to a linkage which becomes singular somewhere in its range of movement. These combinations are marked as regular dashed lines. Some of the dashed lines represent singularities which occur outside the standard stowed-deployed range of motion (often a configuration the linkage could only reach if it passed through itself), but they are included for completeness. Singularities in the closure equations are of interest primarily

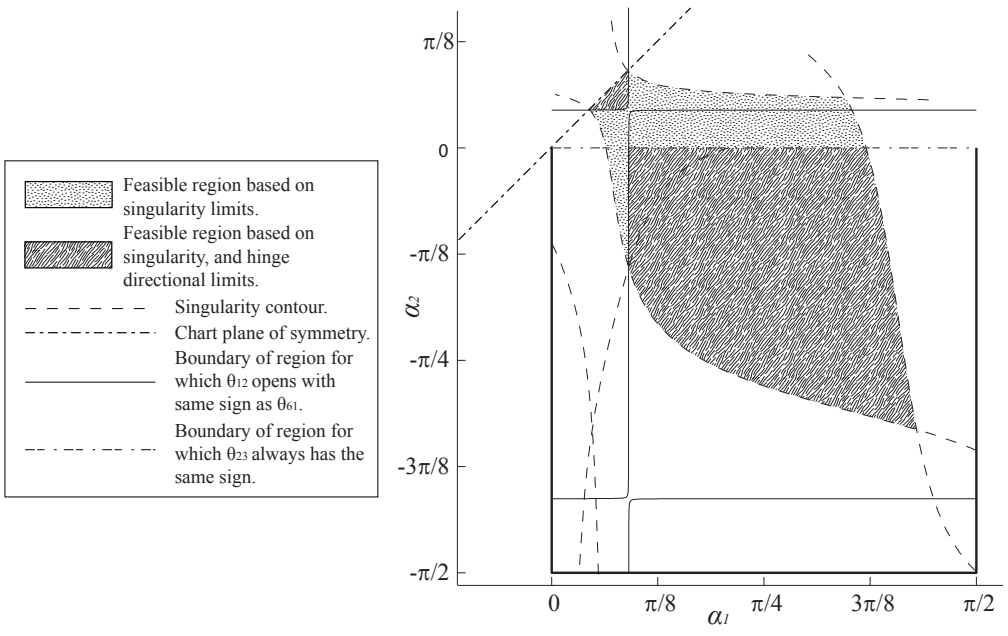


Figure 3.7: Feasibility map for $\gamma = 5\pi/8$.

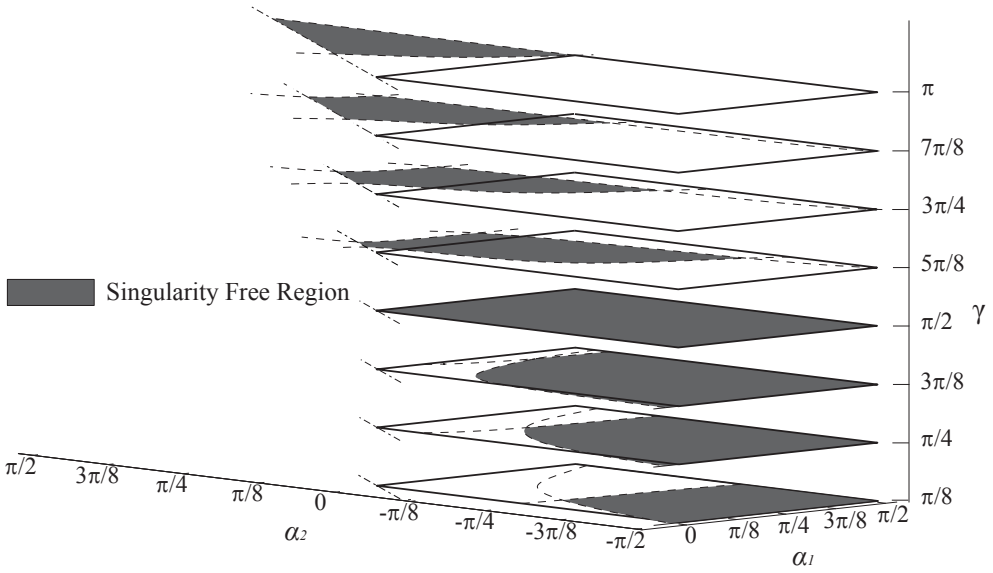


Figure 3.8: Variance of singularity free region with γ.

because they indicate that a bifurcation could occur at the point at which the linkage suddenly gains an extra degree of freedom. The singularity contours of key interest are those which mark a boundary between linkage designs which move continuously from stowed to deployed, and those which do not. If the combination of variables which produce a linkage which becomes singular right at the perimeter of this range can be determined, then a region of design variable space which produces feasible linkages can be bounded. Because of the highly singular nature of the loop closure equations at a bifurcation/singularity point (Gosselin & Angeles, 1990), standard predictor-corrector methods were not suitable for following the singular paths through the design parameter space. Instead, a method which involved a (variable size) predictor step based on the previous step, and a subsequent unconstrained minimisation was used. The objective function has the form:

$$F = \sigma_5 + |\mathbf{F}|$$

where σ_5 is the fifth singular value of the Jacobian of the loop closure equations (zero at a singularity), and $|\mathbf{F}|$ is the norm of the closure equations themselves (Equation 3.10). Note that it is possible to find the linkage's singular values by constructing:

$$K \begin{bmatrix} \Delta\theta_{12} \\ \Delta\theta_{23} \\ \Delta\theta_{34} \\ \Delta\theta_{45} \\ \Delta\theta_{56} \\ \Delta\theta_{61} \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta\theta_x \\ \Delta\theta_y \\ \Delta\theta_z \end{bmatrix}$$

where

$$K = \begin{bmatrix} \mathbf{p}_1 \times \mathbf{h}_1 & \mathbf{p}_2 \times \mathbf{h}_2 & \mathbf{p}_3 \times \mathbf{h}_3 & \mathbf{p}_4 \times \mathbf{h}_4 & \mathbf{p}_5 \times \mathbf{h}_5 & \mathbf{p}_6 \times \mathbf{h}_6 \\ \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{h}_3 & \mathbf{h}_4 & \mathbf{h}_5 & \mathbf{h}_6 \end{bmatrix}$$

The singular values of K can be used in just the same way as those of the Jacobian of Equation 3.10. Here, $[\Delta x, \Delta y, \Delta z, \Delta\theta_x, \Delta\theta_y, \Delta\theta_z]^T$ is the resultant twist of the Bricard 6R plane symmetric linkage.

The other contour lines, defining the boundaries of design for linkages with mono-directional hinges, were found using path followers which use a predictor step in the direction of the null space of the Jacobian at the previous point, and a corrector based simply on Newton's method.

A more extensive collection of feasibility graphs is included as an appendix.

3.5 A Two-Variable Compatibility Equation

A loop closure equation based on each of the linkage's six hinge angles was derived in Section 3.3. In this section, a similar equation is derived, but only in terms of two of the hinge angles. This equation will be referred to as a Compatibility Equation, as it ensures the compatibility of each half of the ring at the plane of symmetry. The derivation of the compatibility equation is based on the assumption that the linkage is always symmetric about the plane defined by hinges $\mathbf{h}_6$ and $\mathbf{h}_3$. A compact way of representing this symmetry is:

$$Q = [(\mathbf{p}_6 - \mathbf{p}_3) \times \mathbf{h}_6] \cdot \mathbf{h}_3 = 0 \tag{3.13}$$

This equation can be written entirely in terms of the variable hinge angles θ_{61} and θ_{12} (as well as the fixed design parameters α_1, α_2 and γ). If one of the hinge angles, say θ_{61}, is nominated as the driving, or input angle, then Equation 3.13 can be used to find θ_{12} in terms of θ_{61}. The other four hinge angles can then be found.

If the bar lengths are labelled simply as l, then the hinge locations **in the deployed configuration** can be written as:

$$\mathbf{p}_1(0) = \begin{bmatrix} -\frac{l}{2} \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{p}_2(0) = \begin{bmatrix} \frac{l}{2} \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{p}_6(0) = \begin{bmatrix} -\frac{l}{2} \\ l \\ 0 \end{bmatrix} \quad \mathbf{p}_3(0) = \begin{bmatrix} \frac{l}{2} \\ l \\ 0 \end{bmatrix}$$

where the midpoint of $\mathbf{p}_1$ and $\mathbf{p}_2$ has been taken as the origin. Furthermore, hinge unit vectors can be determined:

$$\hat{\mathbf{h}}_6(0) = \begin{bmatrix} \sin\beta_1 \\ 0 \\ \cos\beta_1 \end{bmatrix} \quad \hat{\mathbf{h}}_3(0) = \begin{bmatrix} \sin\beta_2 \\ 0 \\ \cos\beta_2 \end{bmatrix}$$

Finally, the alternate hinge angle definitions α_1 and α_2 can be used to define:

$$\hat{\mathbf{h}}_1 = \begin{bmatrix} -\frac{1}{\sqrt{2}}\cos\alpha_1 \\ -\frac{1}{\sqrt{2}}\cos\alpha_1 \\ \sin\alpha_1 \end{bmatrix} \quad \hat{\mathbf{h}}_2 = \begin{bmatrix} \frac{1}{\sqrt{2}}\cos\alpha_2 \\ -\frac{1}{\sqrt{2}}\cos\alpha_2 \\ \sin\alpha_2 \end{bmatrix}$$

In the following it is critical that $\mathbf{h}_1$ and $\mathbf{h}_2$ be expressed as unit vectors. The alternative definitions α_1 and α_2 have been used here because **they allow $\mathbf{h}_1$ and $\mathbf{h}_2$ to be expressed as unit vectors in the current axes without the use of a normalising factor dependent on α_1' or α_2'.** This greatly simplifies the process of solving the compatibility equations

later on.

If the locations of hinges 1 and 2 ($\mathbf{p}_1$ and $\mathbf{p}_2$) are held fixed in space, then the locations of hinges 6 and 3 ($\mathbf{p}_6$ and $\mathbf{p}_3$) can be found by rotating their locations in the deployed configuration about hinge vectors $\hat{\mathbf{h}}_1$ and $\hat{\mathbf{h}}_2$. To rotate a vector $\mathbf{v}$ about a (unit) axis $\mathbf{w}$ by an angle θ:

$$\mathbf{v}' = (\mathbf{v}\cdot\mathbf{w})\mathbf{w} + (\mathbf{v} - (\mathbf{v}\cdot\mathbf{w})\mathbf{w})\cos\theta + \mathbf{v}\times\mathbf{w}\sin\theta \tag{3.14}$$

If the axis $\mathbf{w}$ passes through a point $\mathbf{p}$, and the substitution $\mathbf{u} = \mathbf{v} - \mathbf{p}$ is made, then:

$$\mathbf{v}' = (\mathbf{u}\cdot\mathbf{w})\mathbf{w} + (\mathbf{u} - (\mathbf{u}\cdot\mathbf{w})\mathbf{w})\cos\theta + \mathbf{u}\times\mathbf{w}\sin\theta + \mathbf{p} \tag{3.15}$$

Applying Equation 3.15 to the current problem for $\mathbf{p}_1$ and $\mathbf{p}_2$ gives:

$$\begin{aligned}\mathbf{p}_6 &= (\mathbf{u}_1\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1 + (\mathbf{u}_1 - (\mathbf{u}_1\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1)\cos\theta_{61} + \mathbf{u}_1\times\hat{\mathbf{h}}_1\sin\theta_{61} + \mathbf{p}_1(0)\\ \mathbf{p}_3 &= (\mathbf{u}_2\cdot\hat{\mathbf{h}}_2)\hat{\mathbf{h}}_2 + (\mathbf{u}_2 - (\mathbf{u}_2\cdot\hat{\mathbf{h}}_2)\hat{\mathbf{h}}_2)\cos\theta_{12} + \mathbf{u}_2\times\hat{\mathbf{h}}_2\sin\theta_{12} + \mathbf{p}_2(0)\end{aligned} \tag{3.16}$$

where $\mathbf{u}_1 = \mathbf{p}_6(0) - \mathbf{p}_1(0)$ and $\mathbf{u}_2 = \mathbf{p}_3(0) - \mathbf{p}_2(0)$. Also introduced are the hinge angles θ_{61} and θ_{12} which each start at zero, and indicate how much hinges 1 and 2 have 'opened' during the process of linkage folding. Hinge vectors $\mathbf{h}_6$ and $\mathbf{h}_3$ also change during the folding/unfolding process. Applying Equation 3.14 to the problem of finding the hinge orientations gives:

$$\begin{aligned}\hat{\mathbf{h}}_6 &= (\hat{\mathbf{h}}_6(0)\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1 + (\hat{\mathbf{h}}_6(0) - (\hat{\mathbf{h}}_6(0)\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1)\cos\theta_{61} + \hat{\mathbf{h}}_6(0)\times\hat{\mathbf{h}}_1\sin\theta_{61}\\ \hat{\mathbf{h}}_3 &= (\hat{\mathbf{h}}_3(0)\cdot\hat{\mathbf{h}}_2)\hat{\mathbf{h}}_2 + (\hat{\mathbf{h}}_3(0) - (\hat{\mathbf{h}}_3(0)\cdot\hat{\mathbf{h}}_2)\hat{\mathbf{h}}_2)\cos\theta_{12} + \hat{\mathbf{h}}_3(0)\times\hat{\mathbf{h}}_2\sin\theta_{12}\end{aligned} \tag{3.17}$$

Using Equation 3.1, is is possible to re-write the definitions for $\hat{\mathbf{h}}_6$ and $\hat{\mathbf{h}}_3$ as:

$$\hat{\mathbf{h}}_6(0) = \begin{bmatrix} 2\sin\alpha_1'\cos\alpha_1'\cos\gamma - \sin\gamma(\cos^2\alpha_1' - \sin^2\alpha_1') \\ 0 \\ \cos\gamma(\cos^2\alpha_1' - \sin^2\alpha_1') + 2\sin\alpha_1'\cos\alpha_1'\sin\gamma \end{bmatrix}$$

$$\hat{\mathbf{h}}_3(0) = \begin{bmatrix} 2\sin\alpha_2'\cos\alpha_2'\cos\gamma - \sin\gamma(\cos^2\alpha_2' - \sin^2\alpha_2') \\ 0 \\ \cos\gamma(\sin^2\alpha_2' - \cos^2\alpha_2') - 2\sin\alpha_2'\cos\alpha_2'\sin\gamma \end{bmatrix}$$

Now to be consistent, these definitions need to be re-written in terms of α_1 and α_2.

$$\hat{\mathbf{h}}_6(0) = \begin{bmatrix} 2\sqrt{2}\sin\alpha_1\cos\alpha_1\cos\gamma - \sin\gamma(\cos^2\alpha_1 - 2\sin^2\alpha_1) \\ 0 \\ \cos\gamma(\cos^2\alpha_1 - 2\sin^2\alpha_1) + 2\sqrt{2}\sin\alpha_1\cos\alpha_1\sin\gamma \end{bmatrix} / \left(1+\sin^2\alpha_1\right)$$
$$\hat{\mathbf{h}}_3(0) = \begin{bmatrix} 2\sqrt{2}\sin\alpha_2\cos\alpha_2\cos\gamma - \sin\gamma(\cos^2\alpha_2 - 2\sin^2\alpha_2) \\ 0 \\ \cos\gamma(2\sin^2\alpha_2 - \cos^2\alpha_2) - 2\sqrt{2}\sin\alpha_2\cos\alpha_2\sin\gamma \end{bmatrix} / \left(1+\sin^2\alpha_2\right) \tag{3.18}$$

If Equation 3.18 is substituted into 3.17, and then combined with 3.16 in the compatibility equation (3.13), the final form is achieved. Worthy of note is that the $(1+\sin^2\alpha)$ terms in the denominator of 3.18 can be multiplied out, leaving a pure polynomial form of the compatibility equation.

It is also possible to use Equation 3.13 to simulate the motion of the loop. Assume that the bar connecting hinges 1 and 2 is fixed in space (i.e. the plane of symmetry moves). Since the position of hinge 6 is being rotated about $\hat{\mathbf{h}}_1$, Equation 3.17 can be used to decompose $\mathbf{h}_6$:

$$\mathbf{a}_6 = (\hat{\mathbf{h}}_6(0)\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1$$
$$\mathbf{b}_6 = (\hat{\mathbf{h}}_6(0) - (\hat{\mathbf{h}}_6(0)\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1)$$
$$\mathbf{c}_6 = \hat{\mathbf{h}}_6(0)\times\hat{\mathbf{h}}_1$$

and thus

$$\hat{\mathbf{h}}_6 = \mathbf{a}_6 + \mathbf{b}_6\cos\theta_{61} + \mathbf{c}_6\sin\theta_{61}$$
$$\hat{\mathbf{h}}_3 = \mathbf{a}_3 + \mathbf{b}_3\cos\theta_{12} + \mathbf{c}_3\sin\theta_{12}$$

The positions of the hinges can also be parameterised as:

$$\mathbf{q}_6 = (\mathbf{u}_1\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1 + \mathbf{p}_1(0)$$
$$\mathbf{r}_6 = (\mathbf{u}_1 - (\mathbf{u}_1\cdot\hat{\mathbf{h}}_1)\hat{\mathbf{h}}_1)$$
$$\mathbf{s}_6 = \mathbf{u}_1\times\hat{\mathbf{h}}_1$$

leading to:

$$\mathbf{p}_6 = \mathbf{q}_6 + \mathbf{r}_6\cos\theta_{61} + \mathbf{s}_6\sin\theta_{61}$$
$$\mathbf{p}_3 = \mathbf{q}_3 + \mathbf{r}_3\cos\theta_{12} + \mathbf{s}_3\sin\theta_{12}$$

Substituting these forms into Equation 3.13, and collecting trigonometric terms leads to:

$$\eta_0 + \eta_1\cos\theta_{12} + \eta_2\sin\theta_{12} + \eta_3\sin\theta_{12}\cos\theta_{12} + \eta_4\cos^2\theta_{12} + \eta_5\sin^2\theta_{12} = 0 \tag{3.19}$$

where η_i can be expressed explicitly in terms of θ_{61} as well as $\mathbf{a}_6, \mathbf{b}_6 \ldots \mathbf{q}_6, \mathbf{r}_6 \ldots$. The process can just as easily be reversed to write Equation 3.19 in terms of θ_{61} with $\mathbf{a}_4, \mathbf{b}_4 \ldots \mathbf{q}_4, \mathbf{r}_4 \ldots$ found by choosing a value of θ_{12}. It can be shown that in fact:

$$\eta_3 = 0$$
$$\eta_4 = \eta_5$$

If:

$$l = \frac{\eta_1\eta_2 \pm (\eta_0 + \eta_4)\sqrt{\eta_1^2 + \eta_2^2 - (\eta_0 + \eta_4)^2}}{(\eta_0 - \eta_2 + \eta_4)(\eta_0 + \eta_2 + \eta_4)}$$

then Equation 3.19 has the solution:

$$\theta_{12} = \sin^{-1}\left(\frac{l}{\sqrt{1 + l^2}}\right) \tag{3.20}$$

Equation 3.20 can be used to find the positions and orientations of hinges 6 and 3. Using this information, it is possible to reflect in the plane defined by hinges 6 and 3 to find the positions and orientations of hinges 4 and 5. Hinge angles θ_{56} and θ_{23} can be found easily in terms of the angles between the plane of symmetry and the bars connecting hinges 1 and 6, and 2 and 3. An example of how the hinge angles θ_{61} and θ_{12} vary with respect to one-another during deployment is given in Figure 3.9.

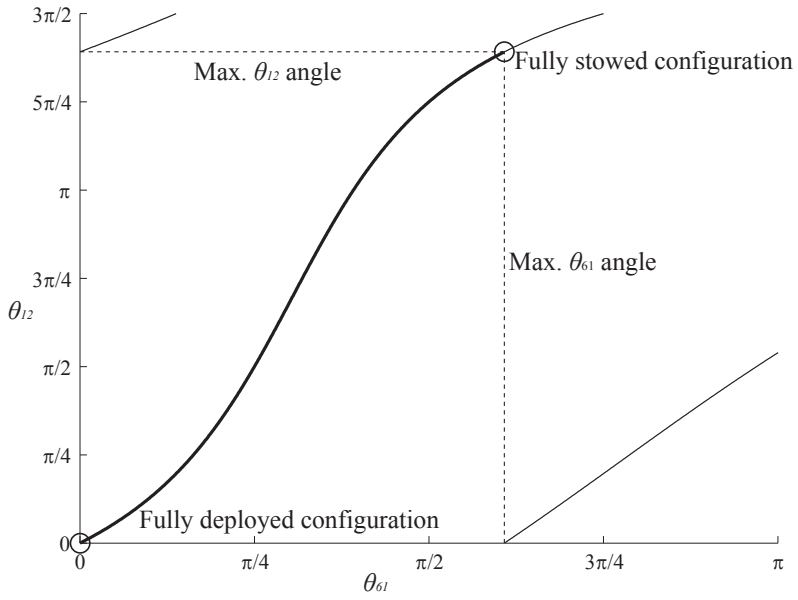


Figure 3.9: Angle evolution during the deployment of a linkage with parameters $\alpha_1 = \pi/4$, $\alpha_2 = -\pi/4$ and $\gamma = \pi/2$, derived from equation 3.20.

It is also possible to simulate the motion of closed loops numerically (Chen & You, 2009; Gan & Pellegrino, 2006), although this will not be discussed here. One advantage of simulating the motion of linkages numerically is that this can provide more ready access to information about likely bifurcations (Gan & Pellegrino, 2005; Kumar & Pellegrino, 2000) using singular value decomposition (other techniques, such as catastrophe theory (Lengyel & You, 2004), can also be useful in the identification of bifurcation points).

3.6 Using a Polyhedral Homotopy to Design the Plane Symmetric Bricard 6R Foldable Frame

Once an expression for the compatibility equation in terms of α_1, α_2 and γ has been obtained, Equation 3.13 can be written as a pure polynomial in terms of new variables (as in Section 3.3) replacing the set of sines and cosines $\{C_{\alpha_1}, S_{\alpha_1}, C_{\alpha_2}, S_{\alpha_2}, C_\gamma, S_\gamma\}$. The polynomial itself is quite long, and in its fully expanded form consists of 389 distinct terms, prohibiting any manual manipulation, and making its explicit representation difficult. The equation can be formed as a black-box function, with inputs $\mathbf{x} = \{C_{\alpha_1}, S_{\alpha_1}, C_{\alpha_2}, S_{\alpha_2}, C_\gamma, S_\gamma\}$, and extra parameters $\{\theta_{61}, \theta_{12}, l\}$. The compatibility equation can be written as:

$$Q\left(\mathbf{x}, \theta_{61}, \theta_{12}, l\right) = 0$$

If the function evaluates to zero, then the equation is satisfied.

The compatibility equation for the plane symmetric rectangular 6-bar linkage has been derived in terms of three variables, α_1, α_2 and γ. All three are angles, and are considered to be the only design variables for the linkages for the purposes of this section.

Because of the nature of Equation 3.1, it is implicit that the linkage will start as a rectangular structure with length twice its width, and then fold into a compact configuration in which all the bars are parallel. This is arguably the most desirable characteristic of the linkage should one intend to use it as a deployable structure. However, one might like to exercise greater control over the way in which the linkage opens and folds. This might be desirable in order to minimise the stretch on a flexible sheet attached to the linkage, or perhaps to confine the dimensions of the linkage to a particular three dimensional envelope during its opening. The complexity of the equations involved in describing the motion of the linkage means that only numerical optimisation techniques lend themselves to any attempt to modify the linkage's parameters of motion. It is possible, however, to 'guide' the linkage on its way from deployed to stowed and vice versa. Since the linkage

has only a single degree of freedom, the specification of the angle between any two adjacent bars is sufficient to completely describe the state of the linkage. It is not possible, in general, to specify exactly what any of the other angles in the linkage should be at the same time, as this over-determines the system. One can, however, specify the values of more than one hinge angle in the linkage during the opening and closing process **at a discrete number of positions**. For each of these positions, one of the design variables must be freed up in order to keep the system determined. Since there are only three design variables, a maximum of three discrete positions may be specified on the linkage's path.

An appropriate way to specify positions along the opening and closing path is to use the hinge angles θ_{61} and θ_{12} in matched pairs $\{\theta_{61i}, \theta_{12i}\}$ for $i = 1, 2, 3$ to establish a set of angular *precision points* through which the linkage must pass. Note that it is not possible to specify the order in which the precision points are encountered during an opening/closing run. It is also not possible to determine whether the linkage will self-intersect during opening/closing. This can only be investigated using simulation.

In choosing to consider the sines and cosines of the variables instead of the variables themselves, it has become necessary to introduce a set of equations to compensate for the increase in the number of polynomial variables. This can be achieved in a number of ways, but a simple way is given in Equation 3.21, preserving polynomial form.

$$\begin{aligned} C_{\alpha_1}^2 + S_{\alpha_1}^2 - 1 &= 0 \\ C_{\alpha_2}^2 + S_{\alpha_2}^2 - 1 &= 0 \\ C_{\gamma}^2 + S_{\gamma}^2 - 1 &= 0 \end{aligned} \tag{3.21}$$

Design problems can now be solved using the compatibility equation. The particular type of problem considered here involves using pre-specified parameter sets $\{\theta_{61_i}, \theta_{12_i}\}$ and then solving for a $\{C_{\alpha_1}, S_{\alpha_1}, C_{\alpha_2}, S_{\alpha_2}, C_{\gamma}, S_{\gamma}\}$ set which satisfies the equations. The full equation set is given in Equation 3.22.

$$F = \begin{Bmatrix} Q\left(\mathbf{x}, \theta_{611}, \theta_{121}, l\right) \\ Q\left(\mathbf{x}, \theta_{612}, \theta_{122}, l\right) \\ Q\left(\mathbf{x}, \theta_{613}, \theta_{123}, l\right) \\ C_{\alpha_1}^2 + S_{\alpha_1}^2 - 1 \\ C_{\alpha_2}^2 + S_{\alpha_2}^2 - 1 \\ C_{\gamma}^2 + S_{\gamma}^2 - 1 \end{Bmatrix} = \mathbf{0} \tag{3.22}$$

Equation 3.22 is all that is required to form a polyhedral homotopy of the form de-

scribed in Section 2.4. It is a system of polynomial equations in six variables. Each of the first three equations of 3.22 contains the same set of 389 monomials, which means they have an identical support with 389 elements. Mathematica can be used to analyse the full equations and arrange the supports into matrix form. The mixed volume of the supports is 2352, meaning that the system has at most this many solutions. It is worth noting that if the variables are placed into homogeneous groups $[\{C_{\alpha_1}, S_{\alpha_1}\}, \{C_{\alpha_2}, S_{\alpha_2}\}, \{C_{\gamma}, S_{\gamma}\}]$, the system has a Bézout number of 2400. The proximity of the Bézout number and the mixed volume is due to the breadth of monomials present in the first three equations of the target system. It was found that, in this case, the polyhedral homotopy method exhibited greater numerical stability than multi-homogenisation methods. Because of this, a polyhedral homotopy was used to solve the system.

Since the system of equations in 3.22 contains more than one equation with the same structure, that is, equations with the same polynomial structure but different coefficients, special polyhedral methods were used to simplify the process of constructing a start system for solving the target problem. The system is said to have Semi-Mixed Supports (systems in which every support is unique are said to have Mixed Supports). The first three equations, representing the compatibility equation but with different coefficients, are treated as a single equation, but with multiplicity three. The convex hull of the support of the first three equations (Q_1) contains only 102 elements; a significant reduction on 389. It is these 102 which are dealt with directly when forming the polyhedral homotopy. In the notation of Li (2003), the target equations of 3.22 have $n = 6$ with $r = 4 : k_1 = 3, k_2 = 1, k_3 = 1, k_4 = 1$. That is to say, the system is written in terms of six unknowns, and there are six equations, but the first three have the same polynomial structure, leaving four distinct polynomial types. Also, $\dim(Q_1) = m_1 = 102$, $\dim(Q_2) = m_2 = 3$, $\dim(Q_3) = m_3 = 3$ and $\dim(Q_4) = m_4 = 3$. Recall that in practice, it is better to order the system of supports in terms of size from smallest to largest. Q_1, being the largest support, was placed last when forming the polyhedral homotopy. Computing a binomial start system for the 6-bar linkage can take half an hour on a 3.16GHz Intel processor with the ordering of Equation 3.22, but only 8 seconds after the re-ordering.

Using continuation to follow the 2352 start solutions from the binomial start system to the random coefficient version of the target system results in a full complement of non-singular finite solutions to track to the real coefficient system. This second continuation process leaves only $\sim$500 non-singular finite solutions. These are the solutions of key interest.

3.6.1 Examples of Solution Runs

Some examples for essentially randomly chosen precision points are given below. In each of the continuation runs below, the number of non-singular finite solutions was in the region of 500. What is of interest is how many of those solutions are real. In each of the following examples, the number of real solutions is given, along with the number of these which were found to be geometrically meaningful and distinct.

In the first example in Table 3.1, only one solution progresses smoothly from deployed to stowed (this is common), and in this case it has the design parameters:

$$\begin{aligned} \alpha_1 &= -\pi/4 \\ \alpha_2 &= \pi/4 \\ \gamma &= -\pi/2 \end{aligned}$$

This result is not particularly surprising as the precision points were taken from a simulation of a linkage with these design variables.

In Table 3.1, Example #2, four real solutions appeared again, however, this time none were found to move smoothly from the deployed to stowed configurations.

The marked precision points of the example in Table 3.1, Example # 3, produced seven distinct real solutions, although again, only one moves along a smooth path from open to closed. It has the design variables:

$$\begin{aligned} \alpha_1 &= -1.358 \\ \alpha_2 &= 0.5871 \\ \gamma &= -2.421 \end{aligned}$$

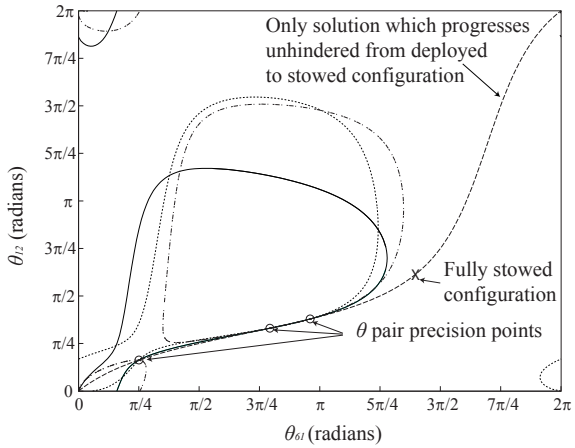
The results for the precision points specified in Table 3.1, Example # 4 also show no practically desirable solutions. This example differs from the others in that the search was for a set of hinge angles not monotonically increasing in θ_{12} with respect to θ_{61}. It is possible that no smooth solutions can be found for such a case.

It is important to remember that the theory of polynomial continuation guarantees that **all** solutions satisfying the precision point constraints will be found. The solution sets given here for these particular examples can be said with confidence to be complete.

Summary of Triple Precision Point Examples

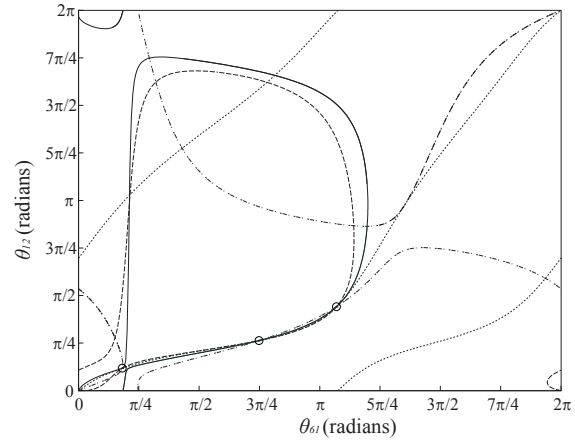
Example # 1

θ pairs	$\left\{ \begin{array}{c} \theta_{61} \\ \theta_{12} \end{array} \right\} = \left\{ \begin{array}{c} 0.787 \\ 0.512 \end{array} \right\}, \left\{ \begin{array}{c} 2.49 \\ 1.04 \end{array} \right\}, \left\{ \begin{array}{c} 3.02 \\ 1.19 \end{array} \right\}$
No. of real solutions found	40
No. of indep. real solutions	4



Example # 2

θ pairs	$\left\{ \begin{array}{c} \theta_{61} \\ \theta_{12} \end{array} \right\} = \left\{ \begin{array}{c} 0.571 \\ 0.370 \end{array} \right\}, \left\{ \begin{array}{c} 2.35 \\ 0.826 \end{array} \right\}, \left\{ \begin{array}{c} 3.36 \\ 1.39 \end{array} \right\}$
No. of real solutions found	32
No. of indep. real solutions	4

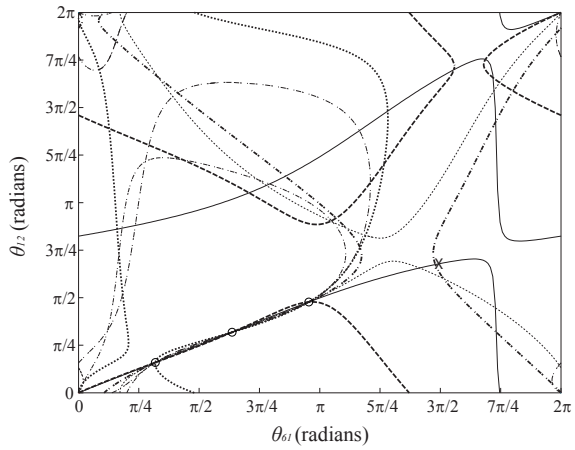


...Continued on next page

– continued from previous page

Example # 3

θ pairs	$\left\{ \begin{array}{c} \theta_{61} \\ \theta_{12} \end{array} \right\} = \left\{ \begin{array}{c} 1 \\ 0.5 \end{array} \right\}, \left\{ \begin{array}{c} 2 \\ 1 \end{array} \right\}, \left\{ \begin{array}{c} 3 \\ 1.5 \end{array} \right\}$
No. of real solutions found	44
No. of indep. real solutions	7



Example # 4

θ pairs	$\left\{ \begin{array}{c} \theta_{61} \\ \theta_{12} \end{array} \right\} = \left\{ \begin{array}{c} 1 \\ 0.5 \end{array} \right\}, \left\{ \begin{array}{c} 2 \\ 1.2 \end{array} \right\}, \left\{ \begin{array}{c} 3 \\ 0.7 \end{array} \right\}$
No. of real solutions found	28
No. of indep. real solutions	3

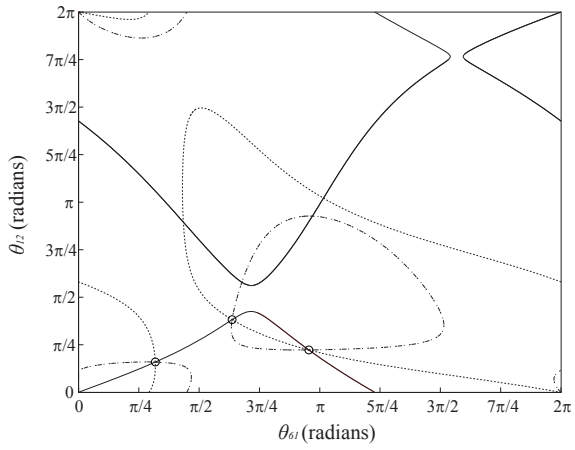


Table 3.1: Angle paths for triple precision point examples. Each curve in each example represents a different linkage design. The θ pair precision points are marked, as is the location at which a feasible design is fully stowed. Note that the combination of θ_{61} and θ_{12} for which a linkage is said to be stowed will be different for each design. In the examples presented, at most one design was found to actually reach the stowed configuration.

3.7 Deployment of the 6R Foldable Frame

The deployment mechanisms for most feasible designs of the plane symmetric 6R foldable frame are likely to be quite simple. The linkage has only a single degree of freedom, and it is possible that the entire deployment could be driven by applying a torque at a single hinge. More likely some redundancy will be included, in the form of spring loaded, damped hinges installed at more than one location around the frame. For example, driving hinges 1,2,4 and 5 would, in most cases, lead to very reliable deployment. Including this redundancy would result in some additional stresses being introduced into the frame during deployment, and also in the deployed configuration. In this situation an issue for consideration would be the avoidance of excessive distortion of the deployed frame's shape. Damping would also need to be sufficient to negate unwanted impulses being applied to any device to which the frame is attached, as well as any vibration produced as bars lock into their deployed positions.

Figure 3.9 shows an example of the relationship between angles θ_{61} and θ_{12} during deployment. Table 3.1 also contains a number of examples. In every case of a feasible frame design encountered to date, θ_{12} is a monotonic function of θ_{61}, and vice versa (note that in Table 3.1, Example #4, the only case in which the θ pairs were not monotonically increasing, no feasible designs were found). This suggests that in most cases, applying a constant torque to hinges 1 and 2 (and 4 and 5) is sufficient to drive a frame from the stowed to deployed configurations.

3.8 Conclusion

The mobility of the singly symmetric 6R foldable ring over-constrained mechanism has been shown to manifest itself in the linkage's closure equations as a single irreducible component in the one-dimensional solution set. A set of 'feasibility maps' showing the regions in parameter space in which the 6R foldable ring exhibits desirable characteristics has been produced. Also, a method of designing such rings by specifying angular precision points has been demonstrated. It is hoped that these techniques, together, will provide a useful and practical way of designing singly symmetric 6R foldable rings.

More work needs to be done in examining exactly what happens to a membrane attached to the frame during deployment. A particularly fragile membrane could tear (most likely near the corner hinges) if overextended, or become too crumpled in the stowed configuration. The exact method of deployment is also an area which has been neglected to date. Which of the hinges need to be driven, and how a sufficient torque could be applied

at those hinges, are issues which require much more thought and, most likely, experimentation.

In considering the design of a practical folding frame, one would also need to consider the dynamics of the deployment; the rigidity of the frame both during deployment, and when fully deployed; the moments necessary to deploy the frame at each hinge location; the velocities of each of the hinges during deployment; and the volume of the region of space required in order for the frame to deploy uninterrupted. For space applications, the stiffness and vibration characteristics are particularly important, especially given that a deployable structure is likely to constitute a large portion of the satellite/spacecraft, and could affect its dynamics significantly.

4. Regular-Polygonal Foldable Rings

The type of linkage considered in this chapter is a cyclic and equilateral ring (that is, it displays rotational symmetry about a line passing through the centre of the ring), generally referred to as a regular polygon. Its purpose is most likely to be the provision of a frame upon which to stretch a membrane, such as a flexible solar array, or a radar. It could also be used to support a reflecting antenna (as analysed in Tibert (2002)); and in such a circumstance might be referred to as a *hoop-column* antenna (see Figure 4.5).

Much like the precision points specified as part of the point-path synthesis described in Section 1.2.4, and the internal mechanism angle relationships prescribed in Section 3.6, the design of deployable folding rings in this chapter uses a set of angular 'precision points' to attempt to steer the linkage along a certain path during its deployment. The prescribed precision points in this chapter are not internal angles, but rather are orientation, or Euler, angles used to describe the attitude of the bars of the ring as the ring deploys. This type of angle specification is a powerful tool, in that it allows a designer to select the exact shape of the ring at a limited number of positions during deployment. This process involves solving some sometimes quite large systems of equations for several design variables at a time. It is possible that a number of different ring designs could satisfy the constraints set by the designer; a situation which manifests itself in the appearance of multiple solutions to the afore-mentioned equations. In general, it cannot be known *a priori* how many, if any, solutions exist to a given problem, but the use of numerical continuation can guarantee that every one of these solutions will be found.

4.1 Existing Designs

Two similar structures: a deployable ring antenna designed by the Harris corporation in the early 1980's, and a large deployable *spoked wheel* designed by Astro Research corporation in the mid 1970's, provide good examples of how deployable rings can be used to solve volume minimisation problems, especially in space applications. Both Harris, and Astro Research prepared their designs as part of contracting work for NASA. Both

designs also provide good insight into the ways in which such frames can be deployed and held rigid in the deployed configuration.

4.1.1 A Hoop-Column Antenna

The hoop-column antenna design, proposed and constructed by Harris, illustrates one of the primary uses of deployable rings; a framework for a large antenna. Because dish antennas require a reflective mesh surface with a specific shape, a deployable antenna will consist not only of a central column and circumferential hoop (the element of primary interest in this chapter), but also a pretensioning mechanism (usually attached to the underside of the mesh), in this case called *surface shaping cords*.

Most sources detailing the development of deployable rings, or hoop-column antennas, suggest some form of column/cable-release mechanism to generate controllable, stable deployment. Quite often, the central column is a telescopic or deployable mast itself. An early concept of how such a structure might deploy is shown in Figure 4.1, while a top view of a later 15 metre hoop-column design is shown in Figure 4.2. This particular model was constructed as a proof of concept for a later design to measure 100 metres in diameter. This design is interesting as, during deployment, it seems all joints are constrained to lie in a cylindrical manifold of slowly increasing radius (see Figure 4.3). Extra controllability is achieved by a complicated combination of pulley systems between joints, and active and passive hinges.

4.1.2 A Spoked Wheel to Deploy Large Surfaces

Deployable rings can also be used to support membranes which do not require any curvature. This includes flexible solar arrays, and phased array antennas. Two similar designs for deployable solar arrays were proposed, first in a conference (Crawford *et al.*, 1974), and then in a report (Crawford *et al.*, 1975). Both designs form a rigid regular polyhedron when fully deployed, upon which a flexible solar array could be mounted. When stowed, both designs fold into a compact bundle. The so called ‘two-hinge’ design receives most attention in the body of the text of the report, but a basic description of the dynamics of a variant in which each bar is joined to its neighbour using only a single hinge is given in the appendix. In the two-hinge case, adjacent bars are connected to one another by an intermediate block with one hinge at each end, leading to the ‘two-hinge’ description. A representation of the two hinge design is shown in Figure 4.4.

The one-hinge version poses a more complex design problem, as compatibility constraints are tighter when only one hinge is used to join each of the bars in the ring. Only

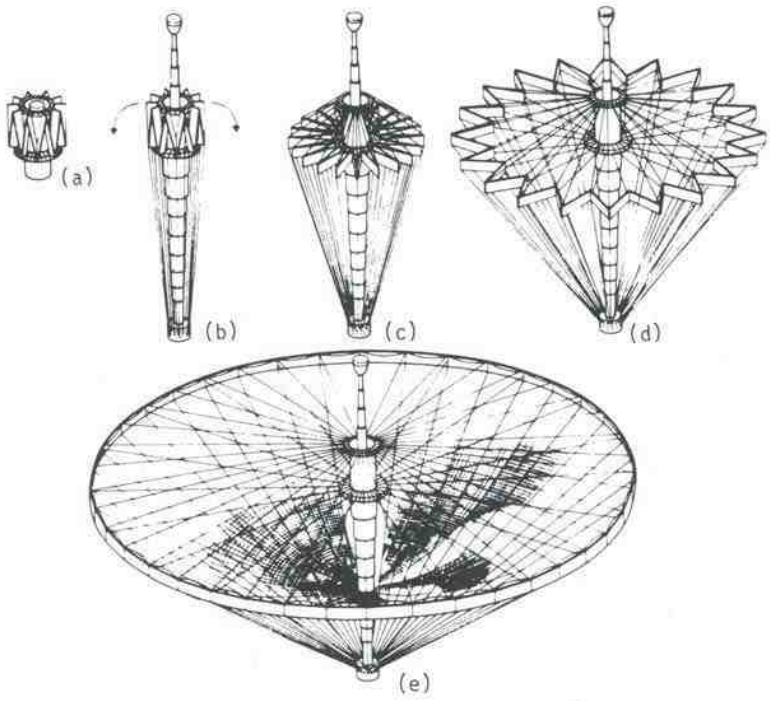


Figure 4.1: A deployable hoop-column concept, developed by the Harris Corporation. Notice how the bars making up the outer ring structure are stored vertically, but rotate into the horizontal plane before deploying, staying in the horizontal plane the whole time. Reproduced from Tibert (2002).

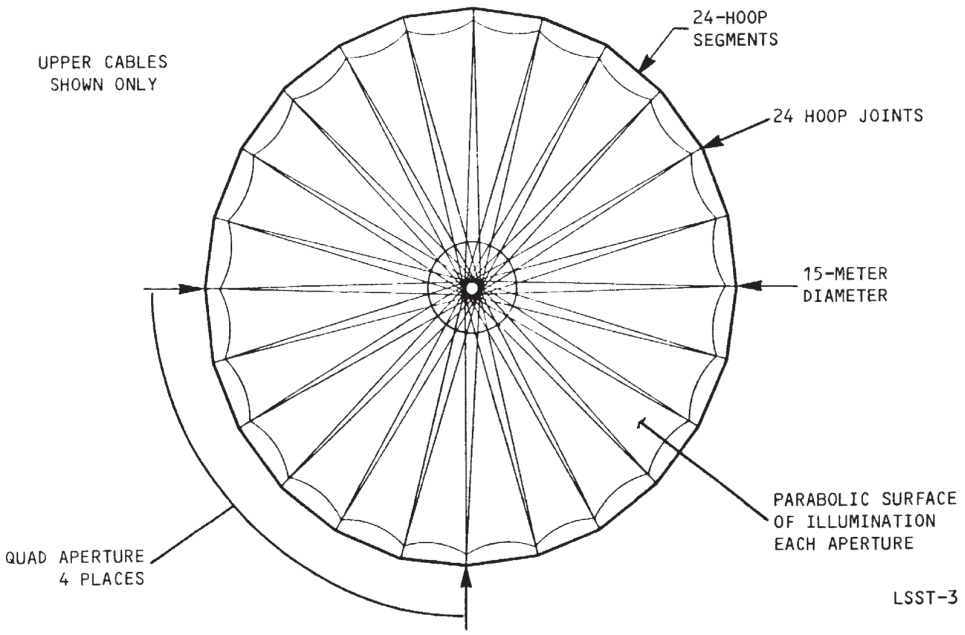


Figure 4.2: Top view of a 15 metre hoop-column design, from Harris-Corporation (1986).

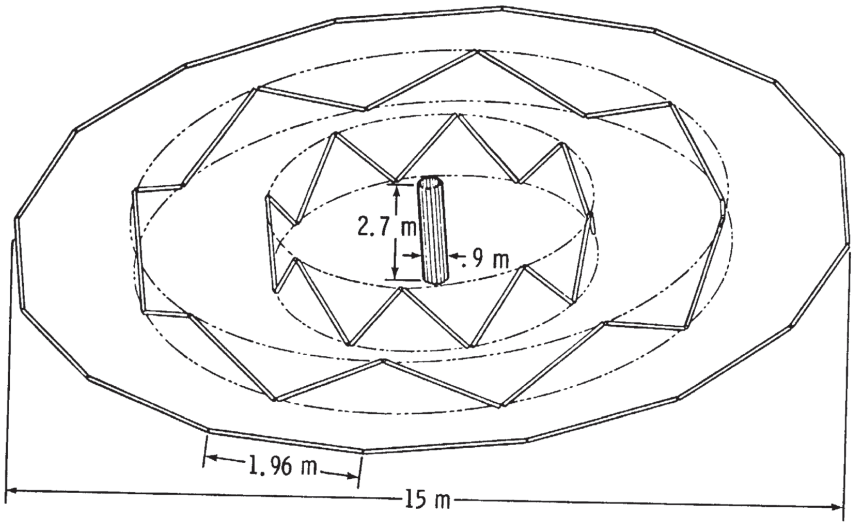


Figure 4.3: Deployment process of a hoop-column ring, from Harris-Corporation (1986).

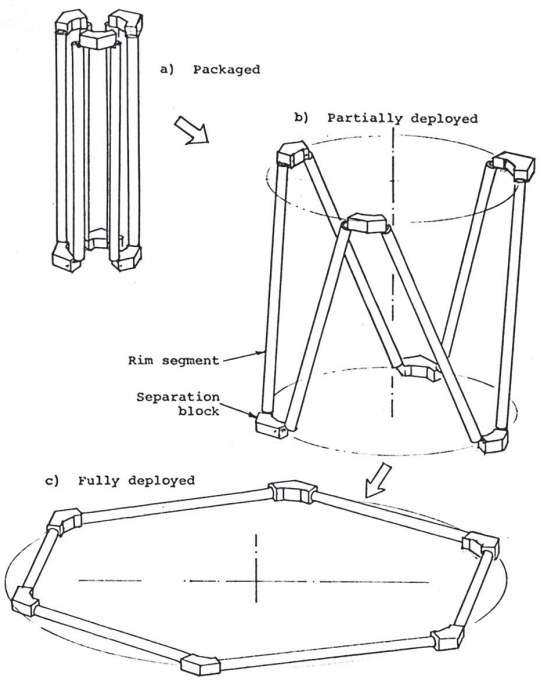


Figure 4.4: A representation of the two hinge spoked wheel design (from Crawford *et al.* (1975)).

an even number of bars is permitted, say N. N is quite often greater than seven, indicating that several degrees of freedom will be present in the ring. In Crawford *et al.* (1975) it was proposed that the deployment be controlled by use of a number of cables, or 'spokes', to constrain these extra degrees of freedom. A real ring whose deployment

is controlled by a system of cables is shown in figure 4.5. The deployment of a 10-bar

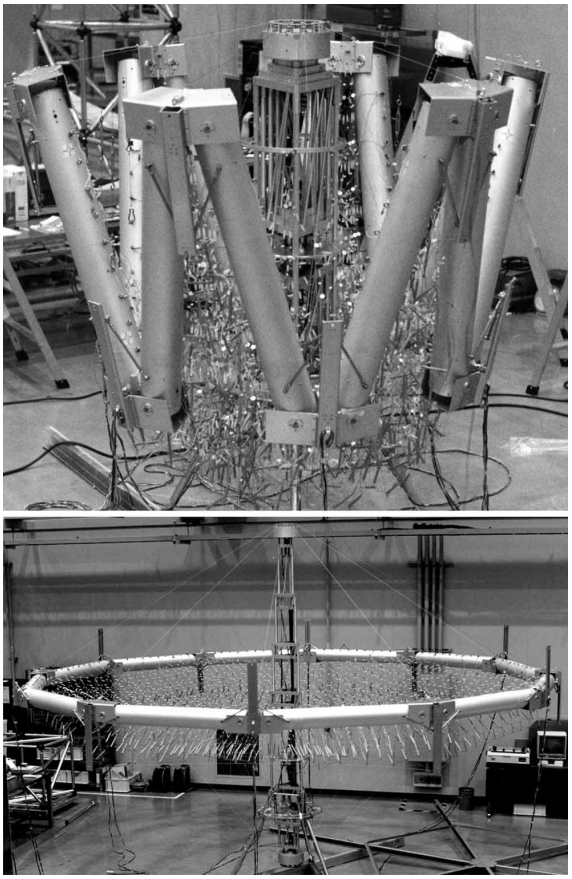


Figure 4.5: A deployable antenna (courtesy of the Toshiba Corporation and Sergio Pellegrino) of the type suggested by Crawford *et al*. This is the two-hinge version of the deployable ring.

single-hinge deployable ring is shown in Figure 4.6.

Because any deployable ring with more than seven bars will have more than one degree of freedom, it is necessary to control the deployment using some additional mechanism. Figure 4.7 shows a fully deployed 8-bar deployable ring supported by a spoked wheel structure. The same ring, partially deployed, is shown in Figure 4.8.

4.2 Ring Geometry and Compatibility Equations

For the remainder of the chapter, the focus will be on single hinge (or NR, where N is the number of elements of the ring) versions of deployable rings, much like those presented in the second example of Section 4.1. Because of its simplicity, the single hinge ring has the potential to exhibit greater reliability. It also presents a more interesting design problem

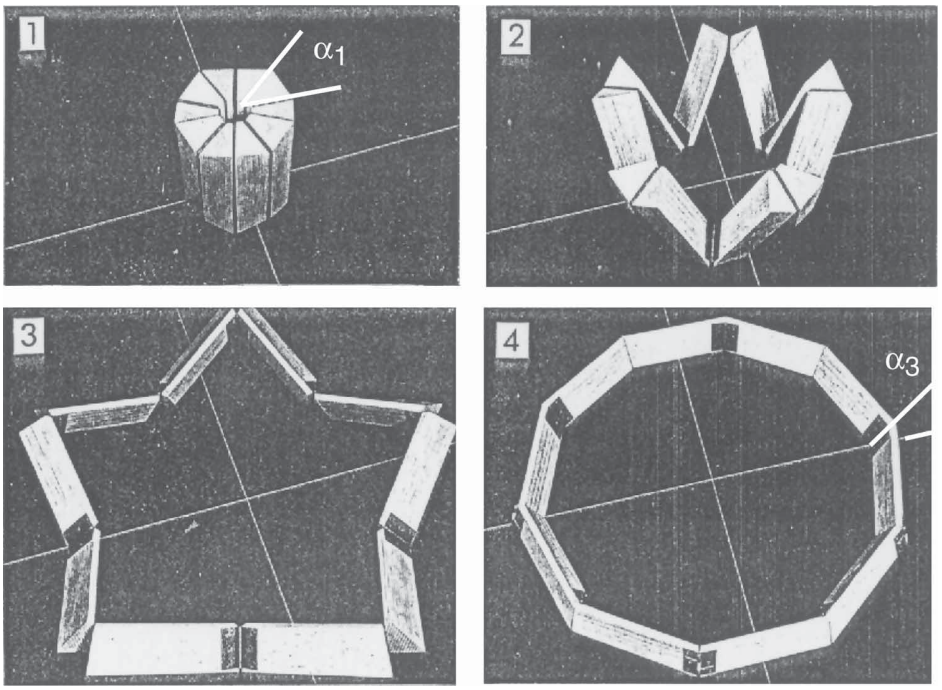


Figure 4.6: The deployment of a 10-bar single-hinge ring (from Crawford *et al.* (1975)). The multiple degrees of freedom are not apparent here as the deployment has been carefully controlled, and the linkage supported by the table underneath.

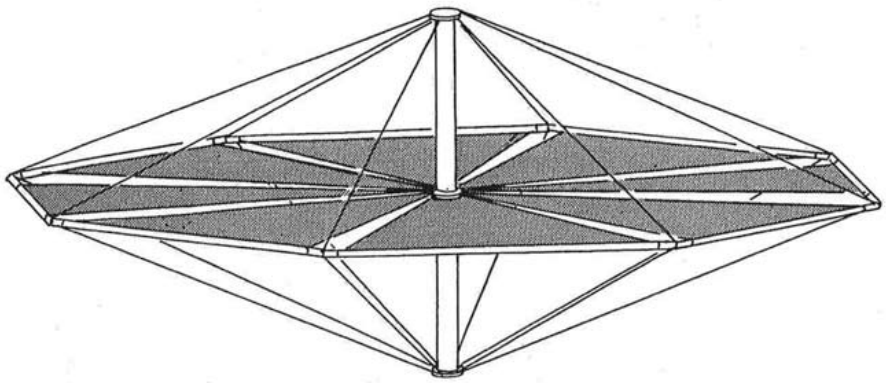


Figure 4.7: A fully deployed 8-bar deployable ring (from Crawford *et al.* (1974)). The ring is held rigid by the *spoked wheel* structure to which it is attached.

as the number of design variables is quite limited, naturally steering the search for designs in the direction of more elegant solutions. It is possible, however, that the versatility of the deployable ring will be limited by the restriction to NR linkages only, with these types

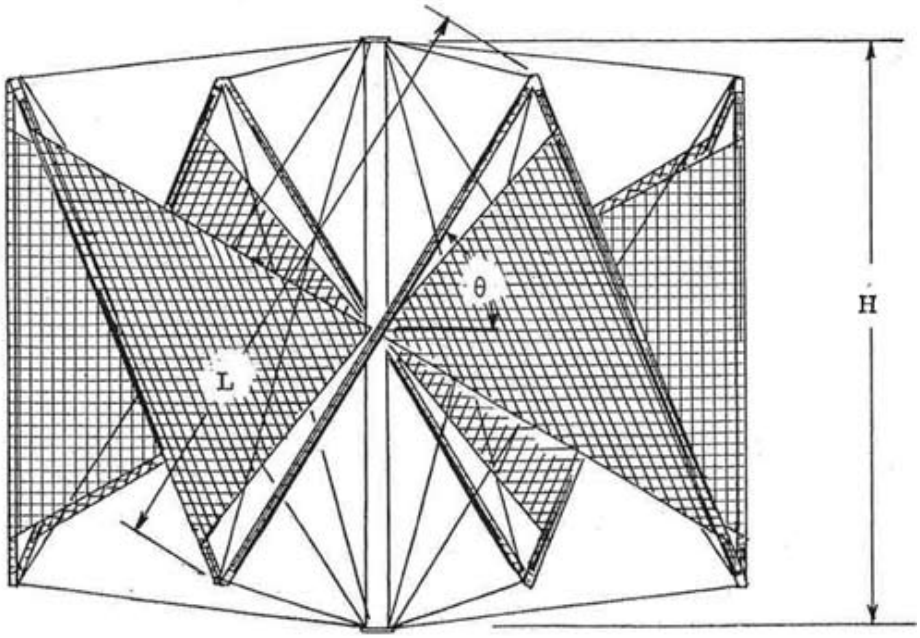


Figure 4.8: A partially deployed 8-bar deployable ring (from Crawford *et al.* (1974)). The solar cell gores visible here unravel from the rim segments as the ring undergoes deployment. The source document contains a calculation of the tension arising in the gore as a result of deployment.

lending themselves more to the support of flat membranes than to contoured surfaces.

4.2.1 Original Derivation of Ring Geometry Based on Triangular Cross-Sectioned Bars

The single-hinge NR version of the regular-polygonal deployable ring can be designed in quite an elegant manner by making the bar cross-section an isosceles triangle. In Figure 4.6, the bars are observed to fit very closely together in the first image because the angle marked α_1 is set to be:

$$\alpha_1 = \frac{360^\circ}{N}$$

More detailed geometry is given in Figure 4.9. Once the angle α_1 has been set, the other two internal angles can be found as:

$$\alpha_2 = 90^\circ - \frac{\alpha_1}{2}$$

The ends of each of the bars are cut at an angle of $180^\circ/N$ so that they fit together in the deployed position. It can be shown that, at the end faces of each bar, the double internal

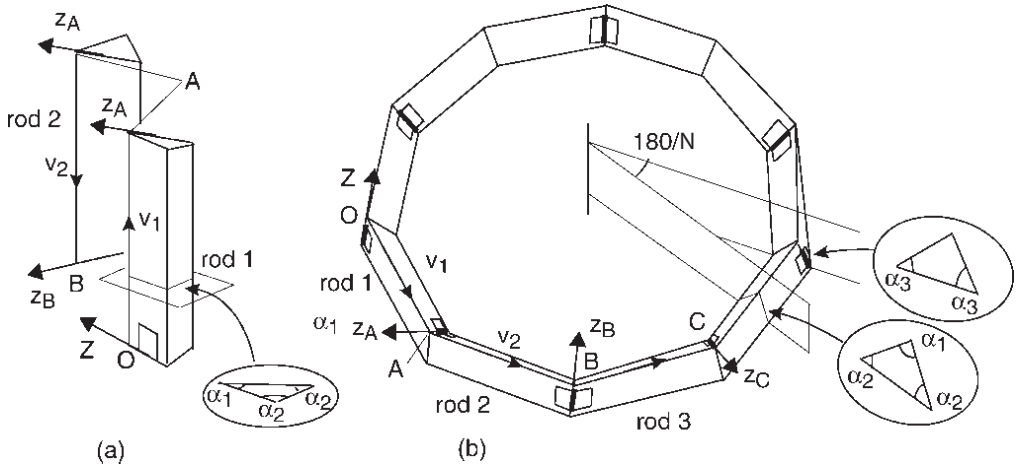


Figure 4.9: Geometry of single-hinge NR foldable ring with triangular cross-sectioned bars (from Gan & Pellegrino (2006)).

angles are:

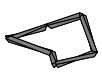
$$\alpha_3 = \tan^{-1}\left(\frac{\cos^2(\alpha_1/2)}{\sin(\alpha_1/2)}\right)$$

The hinges are mounted on the inside and outside upper faces of the ring alternatively.

The aim of the subsequent sections is to present a more comprehensive process for design, based on the assumption that the cross section of the bars is either circular, or irrelevant. The particularly neat and compact stowed configuration of the triangular cross-sectioned bar design is sacrificed for greater control over the deployment path.

4.2.2 Expression of the Hinge Vectors in Local Bar Coordinates

A top view of the one-hinge regular-polygonal deployable ring (sometimes referred to as a Hedgepeth Ring) is given in Figure 4.10. The highlighted bar also has its hinges shown as $\mathbf{h}_l$ on the left and $\mathbf{h}_r$ on the right, where the subscripts l and r denote 'left' and 'right' respectively. To understand the motion of the entire ring, one need only focus on any individual bar, referred to as the 'design' bar from now on. This bar can then be reflected $N-1$ times about the indicated planes of symmetry to obtain the rest of the ring. It is necessary that the hinge vectors of this design bar (and hence those of all the other bars) remain in the planes shown during deployment. A front view of the highlighted (design) bar is shown in Figure 4.11. The hinges are fixed to the end of each bar. An obvious question is: is it always possible to find a bar orientation such that the hinges at the ends of the bar lie exactly in each of the two planes shown in Figure 4.11, and if this is possible, how many different orientations which achieve this are there? It



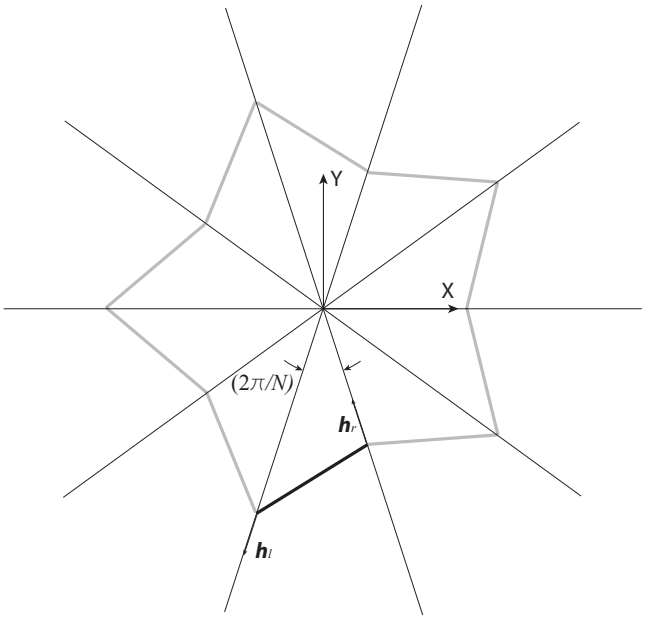


Figure 4.10: Top view of regular-polygonal foldable ring (in mid-deployment) with $N = 10$. Global coordinates are indicated.

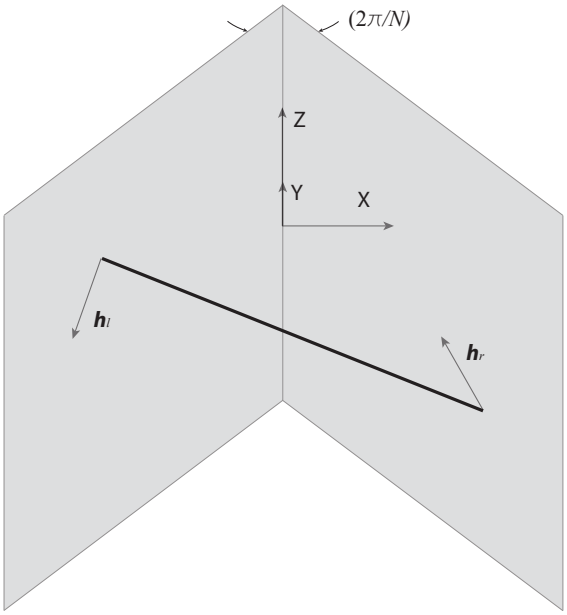


Figure 4.11: Front view of individual bar in regular-polygonal ring with global coordinates indicated.

is not immediately clear that there are not an infinite number of orientations. The benefit of using polynomial continuation in this design problem is that it can be guaranteed that every one of the possible solutions will be found (including positive dimensional sets

indicating that there are an infinite number), so they can be enumerated with confidence.

The next stage is to determine an appropriate set of design variables. Attention is restricted to the highlighted bar shown in Figures 4.10 and 4.11. The first parameter one might consider is the bar length, but this can be neglected without any loss of generality. To allow for maximum generality, the hinges attached to each end of the design bar may have any orientation with respect to the local coordinates of the bar. Figure 4.12 shows how the hinge vectors are defined in the local coordinates of the bar. Each hinge is

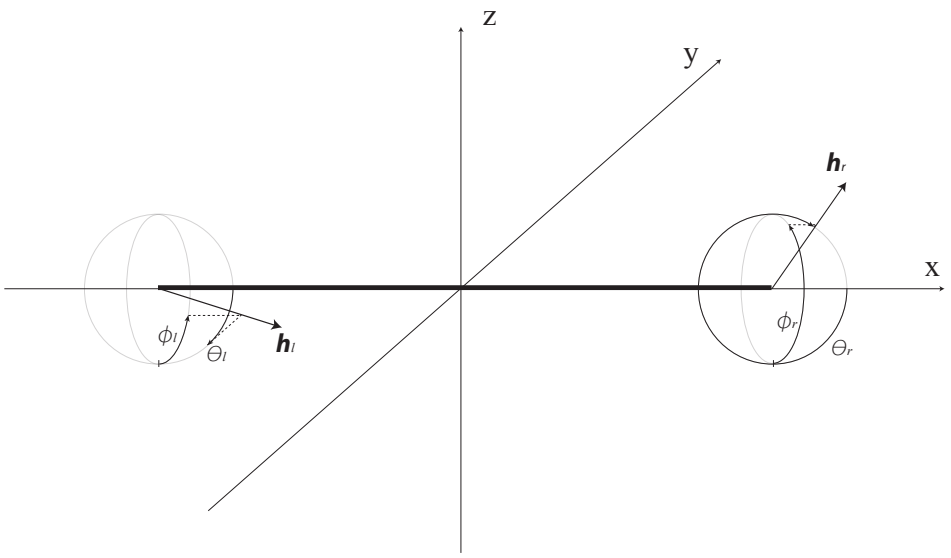


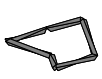
Figure 4.12: Regular-polygonal ring bar shown in local coordinates with hinge definitions.

assigned a 'longitude' ϕ, and a 'lattitude' θ. Using l and r again to denote the left and right hand ends of the bar; the four design variables are $\{\phi_l, \theta_l, \phi_r, \theta_r\}$. The left and right hinges can thus be constructed as unit vectors (in local coordinates attached to the bar):

$$\mathbf{h}_l = \begin{bmatrix} \cos(\theta_l) \\ \sin(\theta_l)\sin(\phi_l) \\ -\sin(\theta_l)\cos(\phi_l) \end{bmatrix} \quad \mathbf{h}_r = \begin{bmatrix} \cos(\theta_r) \\ \sin(\theta_r)\sin(\phi_r) \\ -\sin(\theta_r)\cos(\phi_r) \end{bmatrix} \tag{4.1}$$

4.2.3 Constructing the Two Compatibility Equations

In this section, a system of equations is derived which constrains each link to remain within its own segment, and also requires its orientation to be such that the hinge vectors



attached to each end of the link always lie within the two partitioning planes of symmetry. In this way, such a system enforces the rotational symmetry of the complete ring. In practice, these constraints would be administered by a carefully controlled system of cables, or other such mechanism. The term *compatibility* is used because the purpose of the equation is to enforce certain constraints at planes of symmetry, which then ensures that each link is compatible with the next (across the plane of symmetry).

As the ring is deployed, each bar undergoes a rotation. Define a rotation matrix R in terms of the standard single-axis rotation matrices:

$$L_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}$$

$$L_2 = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}$$

$$L_3 = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R = (L_1 L_2 L_3)^T$$

If the bar undergoes a rotation R, then so do the hinge vectors at each of its ends. Now assume that the two planes shown in Figure 4.11 have normal vectors $\mathbf{n}_l$ on the left and $\mathbf{n}_r$ on the right. It is now possible to write the two governing equations for the motion of the bar (based on the fact that each hinge must remain in its respective plane) as:

$$\begin{aligned} \langle R\mathbf{h}_l \, , \, \mathbf{n}_l \rangle &= 0 \\ \langle R\mathbf{h}_r \, , \, \mathbf{n}_r \rangle &= 0 \end{aligned} \tag{4.2}$$

Together, these equations form the compatibility equations for the ring. As depicted, the plane normals will have the form:

$$\mathbf{n}_l = \begin{bmatrix} \cos(\frac{\pi}{N}) \\ -\sin(\frac{\pi}{N}) \\ 0 \end{bmatrix} \quad \mathbf{n}_r = \begin{bmatrix} \cos(\frac{\pi}{N}) \\ \sin(\frac{\pi}{N}) \\ 0 \end{bmatrix}$$

in global coordinates.

The two relationships in Equation 4.2 are written in terms of three variables; a roll

(ϕ), a pitch (θ) and a yaw (ψ).

4.3 Using the Compatibility Equations to Design the Ring

The goal in this section is to arrive at a reliable design method for a deployable folding ring. Some design criteria which might naturally arise are:

- ring folds into a compact bundle with all bars parallel;
- ring opens out into a regular polygon with number of sides equal to the number of bars (such that it might support a circular sheet of photovoltaic cells, radar patches etc.);
- the orientation of the bars during deployment is pre-specified at several points;
- the bars do not self intersect during deployment.

All but the last of these points can be addressed using continuation to design the ring.

The basic details of the mechanism layout (number of bars, connectivity etc.) have essentially been set by choosing to focus on one-hinge regular-polygonal deployable rings only. Next, some features of the mechanism which can be varied to alter the kinematics (the design variables) need to be identified (in this case the four hinge orientation angles $\{\phi_l, \theta_l, \phi_r, \theta_r\}$). Following this, the most appropriate way to use continuation in the design process must be chosen. In this chapter, it was decided to specify a number of link attitudes (equal to the number of design variables) at various stages in the ring's deployment. A set of compatibility equations describing the linkage in the desired attitudes was then constructed, and solved for the design variables which are required to force the mechanism to attain these positions. The outline of the process is:

1. determine geometric constraints on bars/links in the ring (eg: planes of symmetry);
2. identify the plausible design variables;
3. derive the compatibility equation(s);
4. find the polyhedral structure of the compatibility equation(s), and establish a start system with the same structure for use in continuation;
5. determine appropriate inputs to the compatibility equations to achieve the required design criteria.

The compatibility equations (4.2) are written in terms of ϕ, θ and ψ, as well as the four design variables ϕ_l, θ_l, ϕ_r and θ_r. In this case there are two fundamental compatibility equations, three input (orientation) angles and four design variables. There are a number of approaches which may be taken to using the compatibility equations. The main guiding principle is that four independent equations must be formed in order to solve for all four of the design variables. Three different methods will be explored in this section:

1. use the two compatibility equations in their current form, and use two sets of input angles to form two pairs of the compatibility equations, resulting in the required four equations;

2. combine the two compatibility equations into one by eliminating the ψ input angle. This allows for four sets of the remaining inputs $\{\theta, \phi\}$ to be used to generate the required four versions of the reduced single compatibility equation;

3. combine the two compatibility equations into one by eliminating the ϕ input angle. A mixture of $\{\theta, \phi\}$ and $\{\theta, \psi\}$ type reduced compatibility equations is used to design the ring.

4.3.1 Using the Original Form of the Compatibility Equations

It is possible to choose two sets of orientation (Euler) angles $\{\phi_1, \theta_1, \psi_1\}$ and $\{\phi_2, \theta_2, \psi_2\}$, and then use polynomial continuation to arrive at a set of design variables which describe a bar/ring system which will pass through the prescribed orientations at some point during the ring's deployment. This process is similar, in a way, to that applied to the point-path synthesis problem described in Section 1.2.4, except that in this case, sets of pre-specified bar orientations take the place of precision points. It is also similar to the internal-angle precision point example of Section 3.6, except that the angular precision points in this chapter are specified in global coordinates.

As is common when using continuation, the sines and cosines of the design variable angles are used instead of the angles themselves. This doubles the number of unknowns, and requires the addition of equations of the form $C_x^2 + S_x^2 - 1 = 0$. The full set of

equations is given below.

$$F = \left\{ \begin{array}{l} \langle R(\phi_1,\theta_1,\psi_1)\mathbf{h}_l(C_{\phi_l},S_{\phi_l},C_{\theta_l},S_{\theta_l},C_{\phi_r},S_{\phi_r},C_{\theta_r},S_{\theta_r})\ ,\ \mathbf{n}_l(N)\rangle \\ \langle R(\phi_2,\theta_2,\psi_2)\mathbf{h}_l(C_{\phi_l},S_{\phi_l},C_{\theta_l},S_{\theta_l},C_{\phi_r},S_{\phi_r},C_{\theta_r},S_{\theta_r})\ ,\ \mathbf{n}_l(N)\rangle \\ \langle R(\phi_1,\theta_1,\psi_1)\mathbf{h}_r(C_{\phi_l},S_{\phi_l},C_{\theta_l},S_{\theta_l},C_{\phi_r},S_{\phi_r},C_{\theta_r},S_{\theta_r})\ ,\ \mathbf{n}_r(N)\rangle \\ \langle R(\phi_2,\theta_2,\psi_2)\mathbf{h}_r(C_{\phi_l},S_{\phi_l},C_{\theta_l},S_{\theta_l},C_{\phi_r},S_{\phi_r},C_{\theta_r},S_{\theta_r})\ ,\ \mathbf{n}_r(N)\rangle \\ C_{\phi_l}^2 + S_{\phi_l}^2 - 1 \\ C_{\theta_l}^2 + S_{\theta_l}^2 - 1 \\ C_{\phi_r}^2 + S_{\phi_r}^2 - 1 \\ C_{\theta_r}^2 + S_{\theta_r}^2 - 1 \end{array} \right\} = \mathbf{0}$$

This system has a total degree of $2^8 = 256$ in the eight unknowns. However, it has a mixed volume of only 16.

Without knowing any more about the system, it is now possible to design a ring which will fold up into a tight bundle, and deploy into the shape of a regular polygon, as in Figure 4.13. There are two critical phases to the deployment; fully deployed and

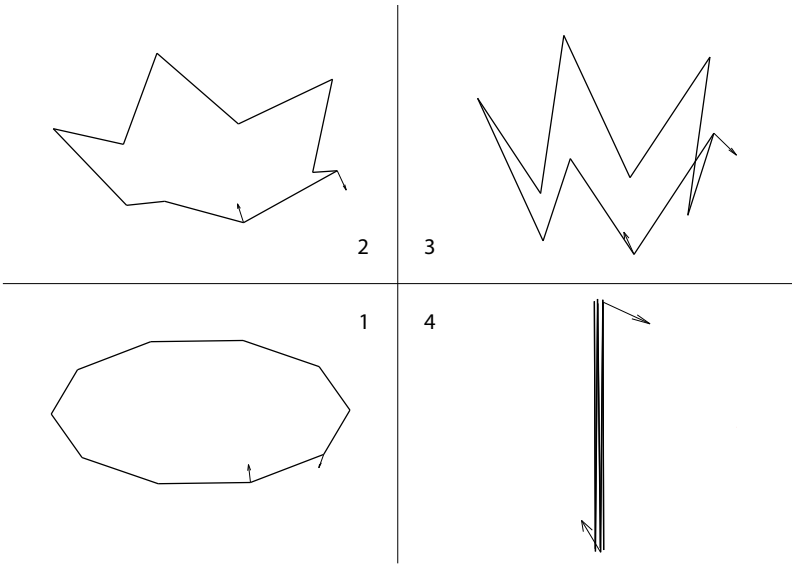


Figure 4.13: Deployment/stowing sequence of a 10-sided regular-polygonal ring (hinge vectors of design bar shown).

fully stowed (depicted as positions 1 and 4 in Figure 4.13). At each of these phases, the 'design' bar (which then gets reflected several times to form the rest of the ring) has a particular attitude. When it is deployed, it has a pitch (θ) of $0°$, and a yaw (ψ) of $0°$. The roll (ϕ) is irrelevant. When the bar is stowed, it has a pitch of $90°$, and both roll and yaw

are irrelevant. As an example, choose the following set of angles (in radians):

$$\begin{bmatrix} \phi_1 \\ \theta_1 \\ \psi_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; \quad \begin{bmatrix} \phi_2 \\ \theta_2 \\ \psi_2 \end{bmatrix} = \begin{bmatrix} \frac{\pi}{2} \\ \frac{\pi}{2} \\ 0 \end{bmatrix} \tag{4.3}$$

This set gives sixteen real solutions, which all reduce to the same hinge vectors. One solution is:

$$\begin{bmatrix} \phi_l \\ \theta_l \\ \phi_r \\ \theta_r \end{bmatrix} = \begin{bmatrix} -18^\circ \\ -84.3^\circ \\ -162^\circ \\ -95.7^\circ \end{bmatrix} \Rightarrow \mathbf{h}_l = \begin{bmatrix} 0.0999 \\ 0.3075 \\ 0.9463 \end{bmatrix}, \ \mathbf{h}_r = \begin{bmatrix} -0.0999 \\ 0.3075 \\ -0.9463 \end{bmatrix}$$

Notice the obvious symmetry in the hinge vectors here, with the magnitude of each component being equal left and right, and only the sign changing. As it happens, these are the same hinge vectors which arise using the design method described in Section 4.2.1. Since this is the only solution to appear in the continuation process, it can be said that the isosceles triangle based design is the only one which satisfies these particular deployed and stowed configuration requirements. The attitude of the design bar during closure is shown in Figure 4.14, while a sequence of the ring moving from the deployed to stowed configuration is given in Figure 4.15.

It is, of course, possible to choose other sets of input angles, but the set described above illustrates how continuation can be used quite simply to design a practical ring. As a further example, consider the set of input target angles:

$$\begin{bmatrix} \phi_1 \\ \theta_1 \\ \psi_1 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.2 \\ 0.25 \end{bmatrix}; \quad \begin{bmatrix} \phi_2 \\ \theta_2 \\ \psi_2 \end{bmatrix} = \begin{bmatrix} 0.9 \\ 1.3 \\ 0.7 \end{bmatrix} \tag{4.4}$$

The angular progression for the ring designed with these inputs is shown in Figure 4.16. Notice that the ring never reaches the fully stowed ($\theta = \pi/2$) nor fully deployed ($\theta = 0$) configurations. This example provides a good insight into the nature of the design problem. When using a similar angle/orientation-based design approach in Section 3.6, the rectangular shape of the deployed linkage was **implicit in the compatibility equation**, that is: setting the driven angle θ_{61} to zero results in a system for which $\theta_{12} = 0$ (the dependent angle) is always a solution. Bar pitch (θ) is usually used as the driven input in this chapter. Setting $\theta = 0$ does not guarantee that the yaw angle, ψ, will also be zero,

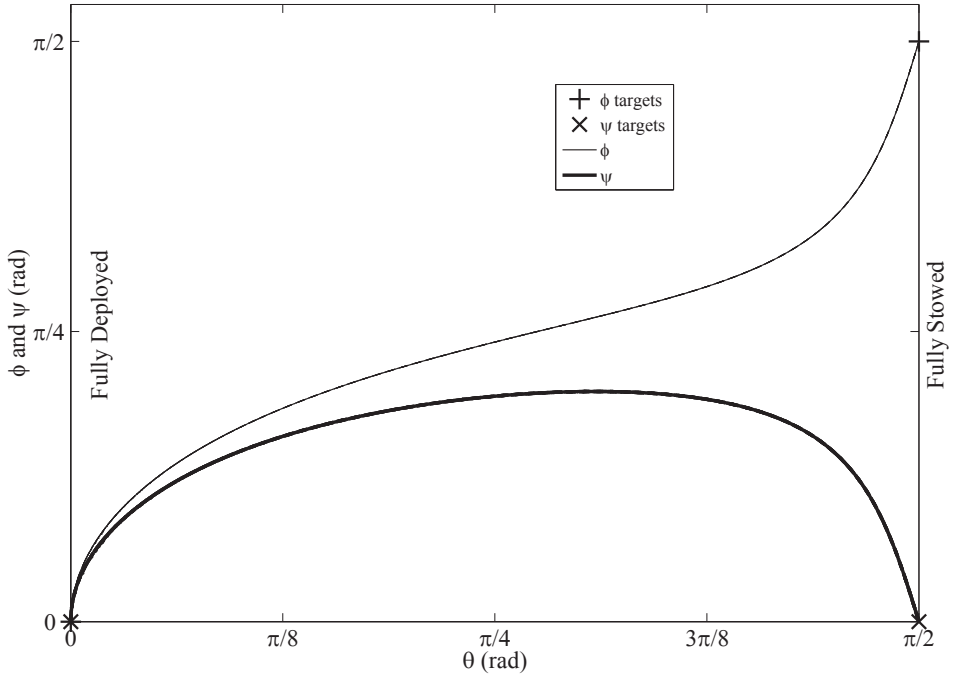


Figure 4.14: Example of a ring design using the original compatibility equations ($N = 10$), and the angle targets given in Equation 4.3.

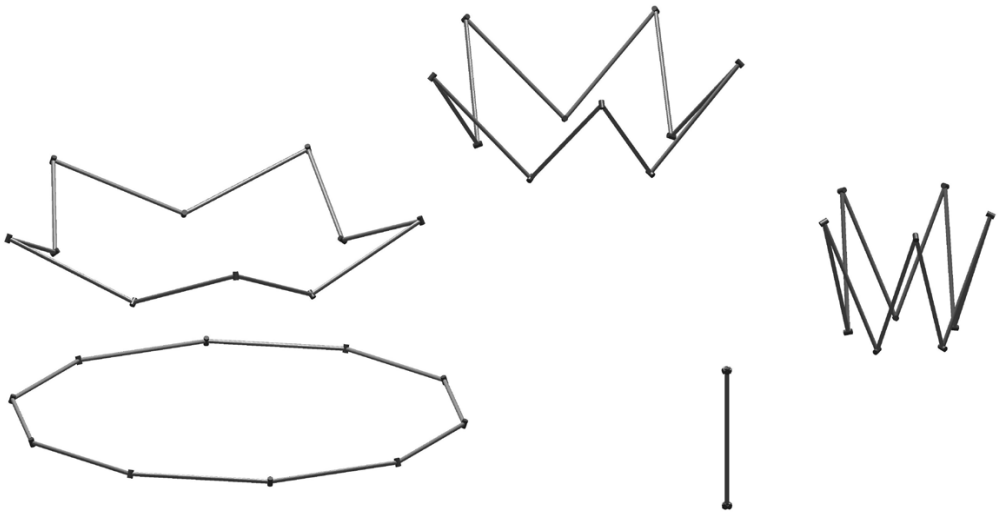


Figure 4.15: First example of a fully feasible ring moving from deployed to stowed positions (clockwise from bottom left). This is identical to the ring which would result from the design methods of Section 4.2.1. It is based on the targets of Equation 4.3, and its angular progression is shown in Figure 4.14.

and since $\theta = \psi = 0$ is a necessary condition for the regular-polygonal foldable ring to be fully deployed, such a relationship **must be specified explicitly** as part of the design process. This is why the targets of Equation 4.3 produce a feasible ring design, while

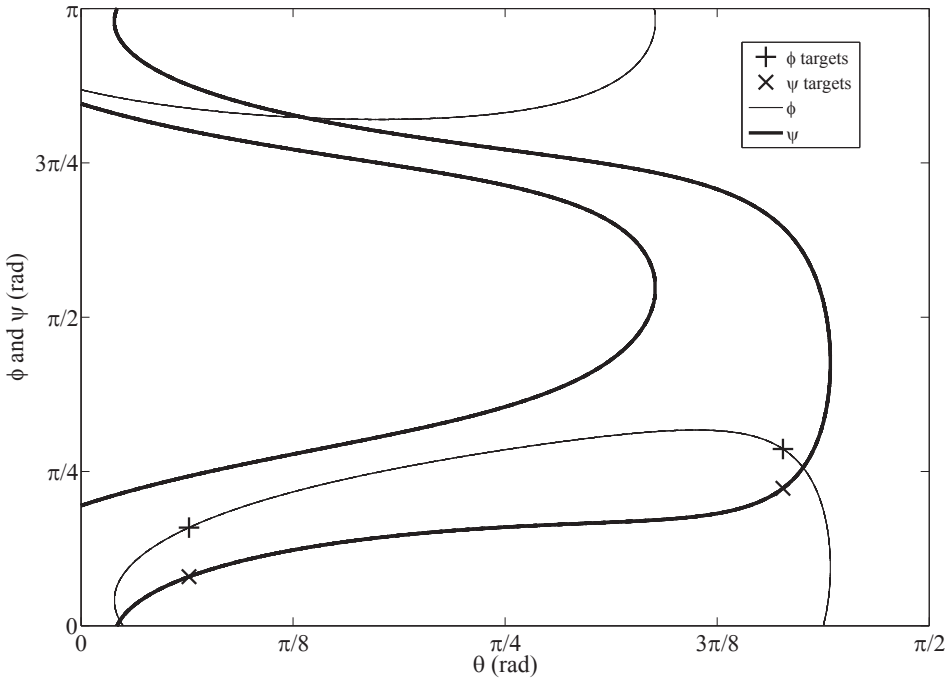


Figure 4.16: Second example of a ring design using the original compatibility equations ($N = 10$), and the angle targets given in Equation 4.4.

those of Equation 4.4 do not.

4.3.2 Using $\{\phi\ ,\ \theta\}$ to Specify Bar Orientation

Sometimes, it is beneficial to reduce the description of the motion of a deployable structure to a single compatibility equation, as this reduces the amount of information which must be specified *a priori* in a design using continuation. In this section, it is shown that only two of the three input angles need be specified to define a target point, leaving the third free. To combine the two compatibility equations into one, one of the three variables; ϕ, θ or ψ must be eliminated. Although the design of a feasible deployable ring using the methods described here must involve the explicit specification of the deployed state constraint $\theta = \psi = 0$, it is nonetheless interesting to observe what happens when the dependence on ψ is removed altogether.

Assume that $C_x = \cos(x)$ and $S_x = \sin(x)$. The equations in 4.2 can be written as:

$$\begin{bmatrix} m_{l_1} & m_{l_2} \\ m_{r_1} & m_{r_2} \end{bmatrix} \begin{bmatrix} C_\psi \\ S_\psi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

where:

$$\begin{bmatrix} m_{l_1} & m_{l_2} \\ m_{r_1} & m_{r_2} \end{bmatrix} = M(\phi, \theta, \phi_l, \theta_l, \phi_r, \theta_r)$$

For this system to have a solution, then

$$\tilde{f} \equiv \det(M) = m_{l_1} m_{r_2} - m_{l_2} m_{r_1} = 0 \tag{4.5}$$

(this particular elimination technique was taken from Crawford *et al.* (1975)). Equation 4.5 is a single compatibility equation for the system, and is written completely in terms of the angular inputs ϕ and θ, as well as the design variables $\{\phi_l, \theta_l, \phi_r, \theta_r\}$. The equation has 9 terms. This equation can be used to define a necessary relationship between ϕ and θ at a point in the bar's motion, without specifying anything about the third angular input, ψ.

Once again, eight equations form the continuation target set. Four sets of pairs of input angles can be selected: $\{\phi_i, \theta_i\}$ $i = 1, \ldots, 4$. These sets form the target system as:

$$F = \left\{ \begin{array}{l} \tilde{f}\left(\{\phi_1, \theta_1\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ \tilde{f}\left(\{\phi_2, \theta_2\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ \tilde{f}\left(\{\phi_3, \theta_3\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ \tilde{f}\left(\{\phi_4, \theta_4\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ C_{\phi_l}^2 + S_{\phi_l}^2 - 1 \\ C_{\theta_l}^2 + S_{\theta_l}^2 - 1 \\ C_{\phi_r}^2 + S_{\phi_r}^2 - 1 \\ C_{\theta_r}^2 + S_{\theta_r}^2 - 1 \end{array} \right\} = \mathbf{0}$$

This system has a total degree of $4^4.2^4 = 4096$, but a mixed volume of only 96, indicating that the vast majority of the solutions are at infinity. Typically, all 96 solutions of the target system are non-singular and geometrically isolated.

As an example, consider the target angles (again in radians):

$$\begin{bmatrix} \phi_1 \\ \theta_1 \end{bmatrix} = \begin{bmatrix} 0.05 \\ 0.1 \end{bmatrix}; \begin{bmatrix} \phi_2 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.3 \end{bmatrix}; \begin{bmatrix} \phi_3 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 0.3 \\ 0.7 \end{bmatrix}; \begin{bmatrix} \phi_4 \\ \theta_4 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1.2 \end{bmatrix} \tag{4.6}$$

which produces the result shown in Figure 4.17. Notice that in this case there are two separate ϕ paths which hit all of the input angle targets, although neither design could be said to be feasible.

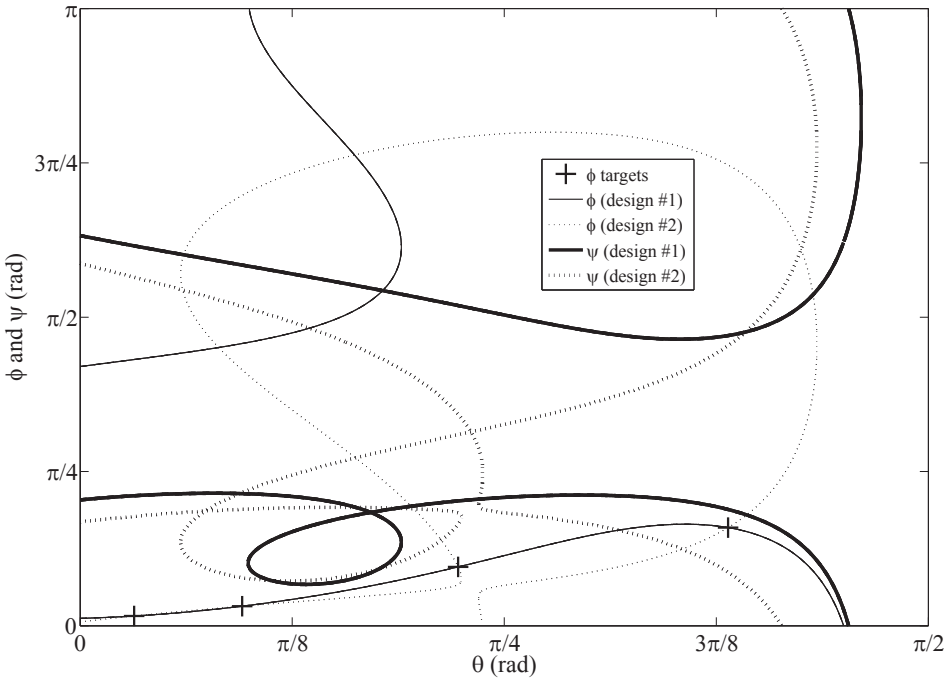


Figure 4.17: Example of system designed with four $\{\phi, \theta\}$ specifications ($N = 10$), and the angle targets given in Equation 4.6.

Next, alter the target angles slightly to include one target pair at $\theta = \pi/2$:

$$\begin{bmatrix} \phi_1 \\ \theta_1 \end{bmatrix} = \begin{bmatrix} 0.05 \\ 0.1 \end{bmatrix}; \begin{bmatrix} \phi_2 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.3 \end{bmatrix}; \begin{bmatrix} \phi_3 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 0.3 \\ 0.7 \end{bmatrix}; \begin{bmatrix} \phi_4 \\ \theta_4 \end{bmatrix} = \begin{bmatrix} 0.5 \\ \frac{\pi}{2} \end{bmatrix} \tag{4.7}$$

with the result shown in Figure 4.18. In this case there are actually four separate ϕ paths which hit the first three targets, but then none hit the fourth at $\theta = \pi/2$. This illustrates an interesting feature. It happens that:

$$\left.\frac{\partial \tilde{f}}{\partial \phi}\right|_{\theta=\frac{\pi}{2}} = 0$$

This means that the specification of any value of ϕ is irrelevant when $\theta = \pi/2$ as the equation exhibits no sensitivity to ϕ. Any value of ϕ_4 in the example above will produce the same results as those shown in Figure 4.18. The benefit of still including an equation with a target pair at $\theta = \pi/2$ is that it encourages real solution paths which travel all the way from $\theta = 0$ to $\theta = \pi/2$, producing a ring which can be fully stowed. If it is not required that a ring be fully stowable, then a target at $\theta = \pi/2$ is not necessary.

Because the reduction from two compatibility equations to one involved the elimination of the yaw orientation (ψ) in this section, all control over this variable was relin-

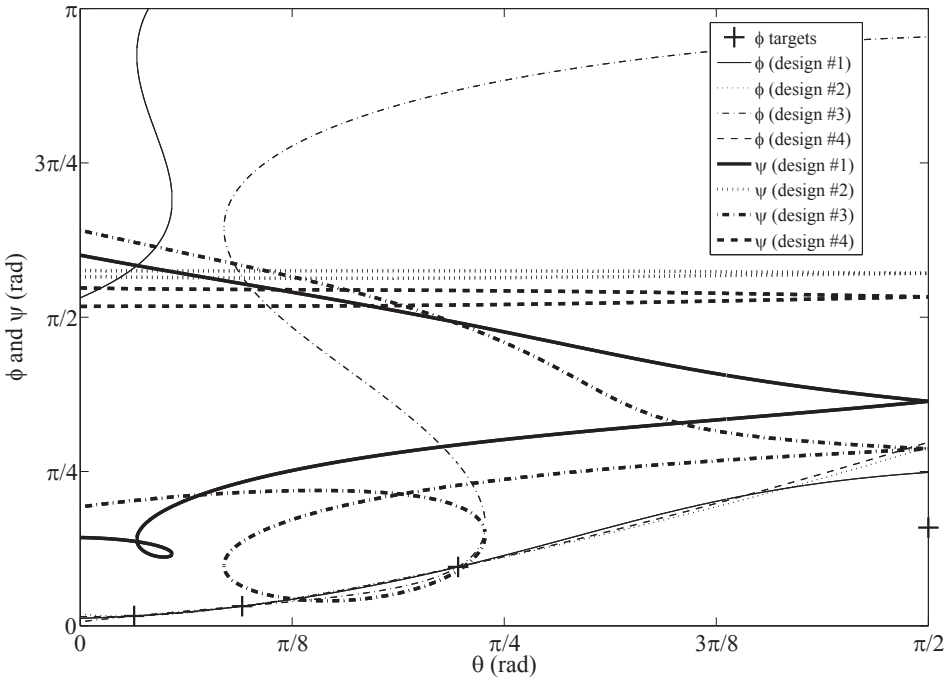
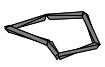


Figure 4.18: Second example of system designed with four $\{\phi, \theta\}$ specifications ($N = 10$), and the angle targets given in Equation 4.7.

quished. Using the remaining two variables, it was possible to generate a number of ring designs which satisfied the four $\{\phi, \theta\}$ precision points, and even to ensure that some of these designs would stow properly by specifying a precision point at $\theta = \pi/2$, but none were found to deploy to the proper shape. To ensure both proper deployment and closure, as well as specifying a number of angular precision points between these two configurations, the system of equations used in this section needs to be modified slightly.

4.3.3 Using $\{\theta\ ,\ \psi\}$ to Specify Bar Orientation

In this section, the two original compatibility equations are combined by way of the elimination of the variable ϕ. The power of this reduction is that it allows the yaw angle to be specified at a particular pitch angle. This is particularly useful for ensuring that the ring forms a regular polygon when fully deployed by enforcing a zero yaw angle at zero pitch (fully deployed). The elimination of the angle ϕ from the governing Equations 4.2 is a little more involved than the elimination of ψ, but demonstrates nicely the way in which new unknowns can be introduced to pose a compatibility equation in pure polynomial form.



Start with the left hinge constraint, and condense its notation as follows:

$$\langle R\mathbf{h}_l \ , \ \mathbf{n}_l \rangle = w_{l_0} + w_{l_1} C_\phi + w_{l_2} S_\phi = 0$$

The w_{l_i} terms represent combinations of trigonometric functions of the input angles θ and ψ, and the design variables. The explicit expressions are omitted for brevity. Next, make the substitution $C_\phi = m$ and $S_\phi = \sqrt{1-m^2}$, which after rearranging gives:

$$(w_{l_1}^2 + w_{l_2}^2)m^2 + 2w_{l_0}w_{l_1}m + w_{l_0}^2 - w_{l_2}^2 = \zeta_2 m^2 + \zeta_1 m + \zeta_0 = 0$$

where

$$\begin{aligned} \zeta_2 &= w_{l_1}^2 + w_{l_2}^2 \\ \zeta_1 &= 2w_{l_0}w_{l_1} \\ \zeta_0 &= w_{l_0}^2 - w_{l_2}^2 \end{aligned}$$

Similar treatment of the equation for the right hinge leads to a second quadratic in m of the form:

$$\xi_2 m^2 + \xi_1 m + \xi_0 = 0$$

Solving each quadratic for m and equating the results in:

$$\frac{-\zeta_1 \pm \sqrt{\zeta_1^2 - 4\zeta_0\zeta_2}}{2\zeta_2} = \frac{-\xi_1 \pm \sqrt{\xi_1^2 - 4\xi_0\xi_2}}{2\xi_2} \tag{4.8}$$

At this stage, introduce the new unknowns:

$$\begin{aligned} \Gamma^2 &= \zeta_1^2 - 4\zeta_0\zeta_2 \\ \Omega^2 &= \xi_1^2 - 4\xi_0\xi_2 \end{aligned}$$

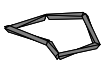
If these new variables are substituted into Equation 4.8, then the compatibility equation system can be written as:

$$\breve{f} \equiv \xi_2 (\Gamma - \zeta_1) - \zeta_2 (\Omega - \xi_1) = 0$$

and

$$\begin{aligned} \Gamma^2 - \zeta_1^2 + 4\zeta_0\zeta_2 &= 0 \\ \Omega^2 - \xi_1^2 + 4\xi_0\xi_2 &= 0 \end{aligned}$$

which are all pure polynomials written completely in terms of the angular inputs ψ and θ, as well as the design variables $\{\phi_l, \theta_l, \phi_r, \theta_r\}$.



Clearly it is going to be more computationally expensive to specify design precision points in terms of $\{\theta, \psi\}$ pairs than $\{\phi, \theta\}$ pairs, as the number of design variables, and hence equations, will increase by two for each pair. As a compromise, a system of target equations which are a combination of the $\{\phi, \theta\}$ and $\{\theta, \psi\}$ based compatibility equations will be used:

$$F = \left\{ \begin{array}{l} \tilde{f}\left(\{\phi_1, \theta_1\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ \tilde{f}\left(\{\phi_2, \theta_2\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ \tilde{f}\left(\{\phi_3, \theta_3\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ \breve{f}\left(\{\psi_1, \theta_4\}, \{C_{\phi_l}, S_{\phi_l}, C_{\theta_l}, S_{\theta_l}, C_{\phi_r}, S_{\phi_r}, C_{\theta_r}, S_{\theta_r}\}\right) \\ \Gamma^2 - \zeta_1^2 + 4\zeta_0\zeta_2 \\ \Omega^2 - \xi_1^2 + 4\xi_0\xi_2 \\ C_{\phi_l}^2 + S_{\phi_l}^2 - 1 \\ C_{\theta_l}^2 + S_{\theta_l}^2 - 1 \\ C_{\phi_r}^2 + S_{\phi_r}^2 - 1 \\ C_{\theta_r}^2 + S_{\theta_r}^2 - 1 \end{array} \right\} = \mathbf{0}$$

This system has an impressive total degree of $4^3.7.8.8.2^4 = 458752$, but a mixed volume of only 768. Using this system, it is possible to specify what the yaw angle should be at one particular pitch angle, as well as what the roll should be at a further three pitches. This gives a very large degree of control over the motion of the ring.

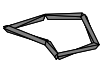
As an example, consider the target angles:

$$\begin{bmatrix} \phi_1 \\ \theta_1 \end{bmatrix} = \begin{bmatrix} \frac{3\pi}{16} \\ \frac{3\pi}{8} \end{bmatrix}; \begin{bmatrix} \phi_2 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} \frac{\pi}{8} \\ \frac{\pi}{4} \end{bmatrix}; \begin{bmatrix} \phi_3 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} \frac{\pi}{16} \\ \frac{\pi}{8} \end{bmatrix}; \begin{bmatrix} \psi_1 \\ \theta_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.9}$$

This generates the paths shown in Figure 4.19. One of the two separate designs shown here which hit the various targets, actually gets quite close to full closure at $\theta = \pi/2$ by chance, but just fails to achieve full closure. Notice, however, that the ψ paths pass through $\psi = 0$ at full deployment, which is very likely to be a desirable design criterion for the ring. To actually ensure full closure, one of the target equations needs to specify $\theta = \pi/2$, as in:

$$\begin{bmatrix} \phi_1 \\ \theta_1 \end{bmatrix} = \begin{bmatrix} - \\ \frac{\pi}{2} \end{bmatrix}; \begin{bmatrix} \phi_2 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0.5 \\ \frac{\pi}{3} \end{bmatrix}; \begin{bmatrix} \phi_3 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 0.25 \\ \frac{\pi}{6} \end{bmatrix}; \begin{bmatrix} \psi_1 \\ \theta_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.10}$$

which results in the two separate ring designs shown in Figure 4.20. While both designs



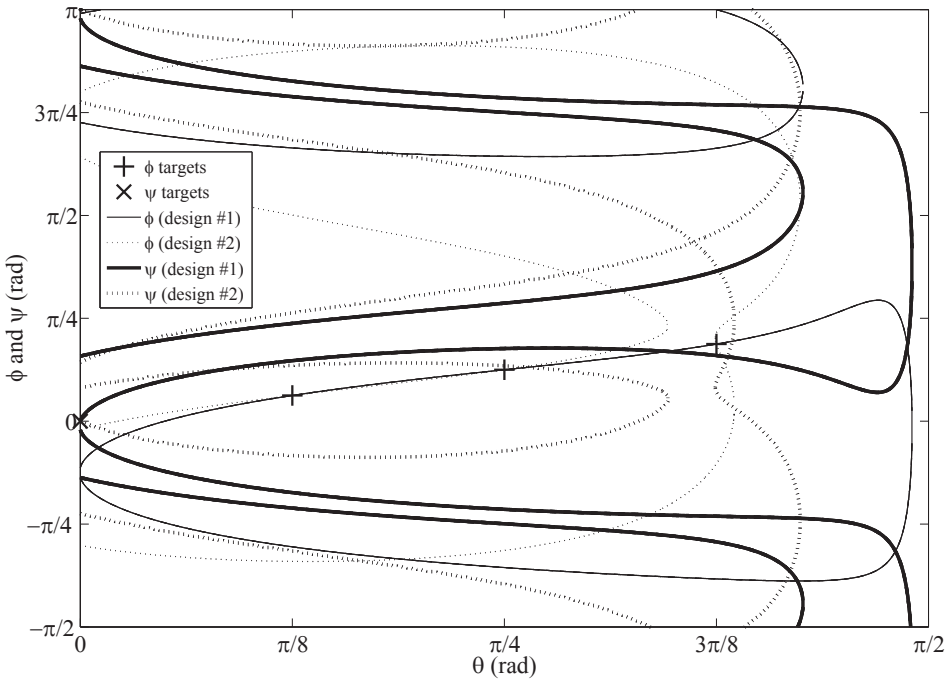
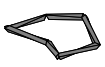


Figure 4.19: Example of system designed with three $\{\phi, \theta\}$ specifications and one $\{\theta, \psi\}$ ($N = 10$), and the angle targets given in Equation 4.9.

reach a fully stowed position with $\theta = \pi/2$, only one of the two designs starts with $\psi = 0$ in the deployed position (labelled as design # 1), so only one completely satisfies the design requirements. This feasible design is represented in a sequence showing the ring moving from the deployed to stowed position in Figure 4.21. The ring represented in Figure 4.21 has the hinge vectors:

$$\begin{bmatrix} \phi_l \\ \theta_l \\ \phi_r \\ \theta_r \end{bmatrix} = \begin{bmatrix} 136.78^\circ \\ -96.80^\circ \\ 172.78^\circ \\ -94.64^\circ \end{bmatrix} \Rightarrow \mathbf{h}_l = \begin{bmatrix} -0.1184 \\ -0.6800 \\ -0.7236 \end{bmatrix}, \ \mathbf{h}_r = \begin{bmatrix} -0.0809 \\ -0.1253 \\ -0.9888 \end{bmatrix}$$

Once again, the value of ϕ_1 specified is irrelevant since $\tilde{f}$ is completely insensitive to ϕ when $\theta = \pi/2$. In this case, what has been designed is a completely functional ring which has all bars parallel in its fully stowed state, and forms a regular polygon when deployed.



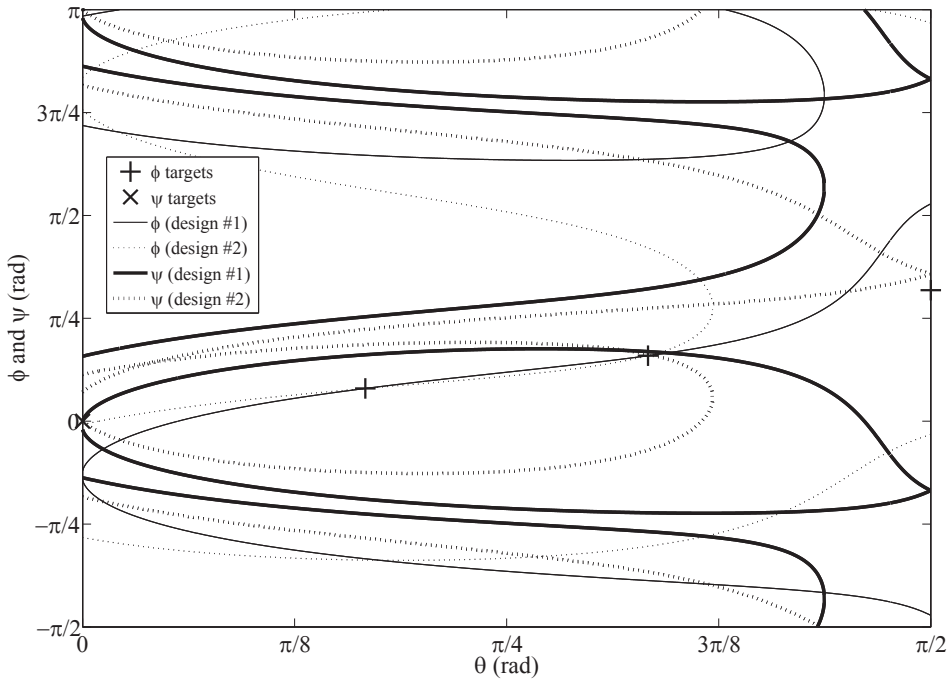


Figure 4.20: Second example of system designed with three $\{\phi, \theta\}$ specifications and one $\{\theta, \psi\}$ ($N = 10$), and the angle targets given in Equation 4.10.

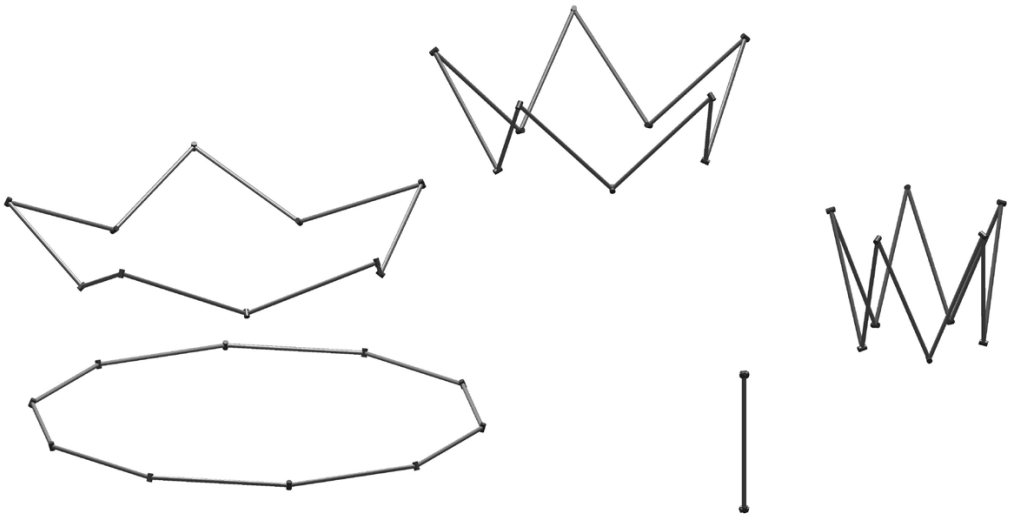
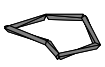


Figure 4.21: Feasible ring design for target set of Equation 4.10. Its angular progression is labelled as design # 1 in Figure 4.20.

4.4 Simulating the Ring's Motion

A number of ways of specifying bar attitudes during deployment (neglecting $\{\phi, \psi\}$ pair specification for now) have been established. It is now necessary to be able to simulate the motion of the bar (and hence the entire ring) accurately in order to verify any results



which might be obtained using a polynomial continuation design approach. Any one of the three rotation angles $\{\phi, \theta, \psi\}$ could be chosen as an input, with the other two being solved by way of the two Equations 4.2. Selecting the pitch angle θ as the input happens to simplify the simulation process.

The two equations in 4.2 can be written as:

$$\begin{aligned}(a_l + b_l C_\phi + c_l S_\phi)\, C_\psi + (d_l + e_l C_\phi + f_l S_\phi)\, S_\psi &= 0 \\ (a_r + b_r C_\phi + c_r S_\phi)\, C_\psi + (d_r + e_r C_\phi + f_r S_\phi)\, S_\psi &= 0\end{aligned}$$

where specifically:

$$\begin{aligned}a_l &= C_\theta C_{\theta_l} C_\delta \\ b_l &= C_{\phi_l} S_\theta S_{\theta_l} C_\delta - S_{\theta_l} S_{\phi_l} S_\delta \\ c_l &= S_\theta S_{\theta_l} S_{\phi_l} C_\delta + C_{\phi_l} S_{\theta_l} S_\delta \\ d_l &= C_\theta C_{\theta_l} S_\delta \\ e_l &= S_{\theta_l} S_{\phi_l} C_\delta + C_{\phi_l} S_\theta S_{\theta_l} S_\delta \\ f_l &= -C_{\phi_l} S_{\theta_l} C_\delta + S_\theta S_{\theta_l} S_{\phi_l} S_\delta\end{aligned}$$

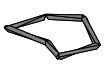
$$\begin{aligned}a_r &= C_\theta C_{\theta_r} C_\delta \\ b_r &= C_{\phi_r} S_\theta S_{\theta_r} C_\delta + S_{\theta_r} S_{\phi_r} S_\delta \\ c_r &= S_\theta S_{\theta_r} S_{\phi_r} C_\delta - C_{\phi_r} S_{\theta_r} S_\delta \\ d_r &= -C_\theta C_{\theta_r} S_\delta \\ e_r &= S_{\theta_r} S_{\phi_r} C_\delta - C_{\phi_r} S_\theta S_{\theta_r} S_\delta \\ f_r &= -C_{\phi_r} S_{\theta_r} C_\delta - S_\theta S_{\theta_r} S_{\phi_r} S_\delta\end{aligned}$$

This can be written in matrix form as:

$$\begin{bmatrix} a_l + b_l C_\phi + c_l S_\phi & d_l + e_l C_\phi + f_l S_\phi \\ a_r + b_r C_\phi + c_r S_\phi & d_r + e_r C_\phi + f_r S_\phi \end{bmatrix} \begin{bmatrix} C_\psi \\ S_\psi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.11}$$

Again, the determinant of the matrix above must be zero in order for a solution to exist. The determinant of the matrix can be written as:

$$\eta_0 + \eta_1 C_\phi + \eta_2 S_\phi + \eta_3 C_\phi^2 + \eta_4 S_\phi^2 + \eta_5 C_\phi S_\phi = 0 \tag{4.12}$$



where specifically:

$$\begin{aligned}
\eta_0 &= a_l d_r - a_r d_l \\
\eta_1 &= -d_l b_r + d_r b_l - a_r e_l + a_l e_r \\
\eta_2 &= -c_r d_l + c_l d_r - a_r f_l + a_l f_r \\
\eta_3 &= b_l e_r - b_r e_l \\
\eta_4 &= c_l f_r - c_r f_l \\
\eta_5 &= c_l e_r - c_r e_l + b_l f_r - b_r f_l
\end{aligned}$$

Once again, use the substitution $C_\phi = m$ and $S_\phi = \sqrt{1-m^2}$ in Equation 4.12 to get:

$$-(\eta_2 + \eta_5 m)\sqrt{1-m^2} = (\eta_3 - \eta_4)m^2 + \eta_1 m + \eta_0 + \eta_4$$

Square both sides of this equation, and rearrange into the form:

$$k_4 m^4 + k_3 m^3 + k_2 m^2 + k_1 m + k_0 = 0 \tag{4.13}$$

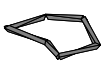
where specifically:

$$\begin{aligned}
k_4 &= (\eta_3 - \eta_4)^2 + \eta_5^2 \\
k_3 &= 2\eta_1(\eta_3 - \eta_4) + 2\eta_2\eta_5 \\
k_2 &= \eta_1^2 + \eta_2^2 + 2(\eta_0 + \eta_4)(\eta_3 - \eta_4) - \eta_5^2 \\
k_1 &= 2\eta_0\eta_1 + 2\eta_1\eta_4 + 2\eta_2\eta_5 \\
k_0 &= (\eta_0 + \eta_4)^2 - \eta_2^2
\end{aligned}$$

Equation 4.13 is a standard quartic equation in m, and closed form expressions exist for each of its four solutions. Since $k_i = f(\theta\ ,\ \phi_l\ ,\ \theta_l\ ,\ \phi_r\ ,\ \theta_r)$ for $i = 0, \ldots, 4$, it is possible to specify a value of θ as the input, then solve for $m = \cos(\phi)$. Once ϕ has been found, ψ can be determined using Equation 4.11. The complete orientation vector $[\phi, \theta, \psi]^T$ can be found in this way.

4.5 Conclusions

Polynomial continuation has been shown to be a reliable method of designing deployable spatial rings to meet certain geometric criteria. Fundamentally, the design process involves the construction of relevant compatibility equations for the basic configuration of the ring, and then solving these equations for the design variables required to allow the



ring to exhibit the desired kinematic properties. Solving these compatibility equations by standard numerical techniques is unlikely to reveal a real solution, let alone the full complement of solutions. By using continuation it is possible to guarantee that every satisfactory combination of design variables has been found.

A mathematical description of the motion of the foldable ring was obtained by focussing on an individual bar of the ring, and applying rotation matrices to this bar (in the Euler angles ϕ, θ and ψ) to derive an expression for the bar's attitude. The problem then became one of finding the mathematical relationships which would ensure that each end of the bar just touched the two closest $N/2$ dividing planes of symmetry, and that the hinges attached to each of these ends lay within these planes. Only one bar need ever be considered, as this individual bar is simply reflected $N - 1$ times around a loop to construct the rest of the ring. Two fundamental compatibility equations were found, and it was shown that these equations could be used directly to design the ring. It was also shown that the two compatibility equations could be combined via the elimination of one of the three input angles. Two different combinations ($\{\phi, \theta\}$ and $\{\theta, \psi\}$) of input angle based compatibility equation were demonstrated, as well as a target system constructed using combinations of these compatibility equations. It was found that in order to design a feasible ring, it is necessary to specify the relationship $\theta = \psi = 0$ (full deployment), as well as $\theta = \pi/2$ with ϕ, ψ free (bars parallel when stowed), all within the same system of equations. Target systems based on different combinations of compatibility equations are possible, but the mixed volume of these systems can become quite large. For example, a target system based on two $\{\phi, \theta\}$, and two $\{\theta, \psi\}$ equations (two $\tilde{f}$ and two $\breve{f}$ equations) has a mixed volume of 3840 in twelve unknowns, which is a considerable, although not impossible, computational problem.

Finally, it was shown that it is possible to simulate the motion of a regular-polygonal foldable ring by way of closed form relationships between the orientation angles of the ring's bars. The pitch angle, θ, is the best candidate for the input angle, and ϕ and ψ can be found explicitly using standard formulae for the solution of quartic polynomials.

5. Doubly Symmetric 8-Bar Foldable Rings

In Crawford *et al.* (1975) a type of deployable ring, or loop, was proposed which can be stored as a collection of parallel bars in a bundle, and deployed to form a regular polygon upon which can be stretched a radar, reflector, solar panel or other flexible sheet. Two different types were proposed; one in which each bar is connected to the next by two hinges attached to a small hub containing a gearing mechanism to keep the rate of deployment of each of the two hinges the same, and a second type in which only a single hinge is used to connect each pair of bars. The number of bars is always even, and for number of bars $n \geq 8$ a great number of degrees of freedom can be present. The design of the single-hinge type was considered in detail in Chapter 4.

In this chapter, the design of a similar type of loop is considered, except that the requirement for the deployed shape to be a regular polygon is removed. This was done in an attempt to explore the versatility of deployable frames of this type; meaning, those which are stored with their bars parallel to a central axis, and deploy to form a planar loop, orthogonal to a central axis. To constrain the difficulty of the design problem, two perpendicular planes of symmetry were introduced, with their intersection defining the central axis. The two-fold plane symmetry means that the number of bars will always be a multiple of four. A representation of a typical folding process is given in Figure 5.1. Only the 8-bar case is examined here in detail, with the 4-bar case being a Bennett linkage, and the 12-bar and above cases forming a good area for further research. Two different variants of the 8-bar are considered; being one in which the vertices of the loop in the deployed configuration are positioned arbitrarily, and another in which the vertices are constrained to lie on a rectangle.

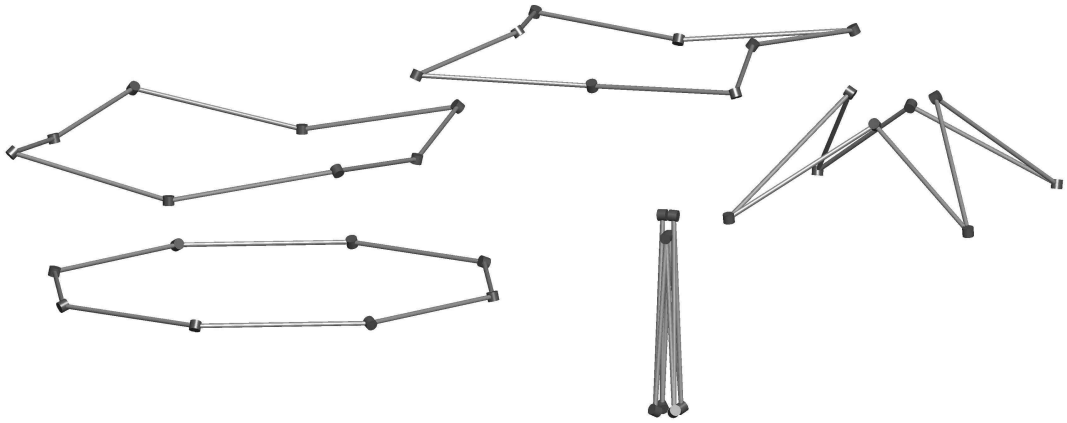


Figure 5.1: A doubly-symmetric 8-bar loop undergoing folding (clockwise from bottom left).

5.1 Arbitrarily Positioned Vertices

The design parameters for this ring are shown in Figures 5.2 and 5.3, the latter being a top view showing the locations of the vertices in XY coordinates (capitalisation denoting global coordinates). Single hinges, each with a single rotational degree of freedom, connect each bar to the next. The only constraints placed on the positioning of the vertices are that hinge a must lie on the X axis, meaning $Y_a = 0$, and hinge c must lie on the Y axis, meaning $X_c = 0$. This leaves four initial positional parameters for the loop, $\{X_a, X_b, Y_b, Y_c\}$.

In Chapter 4, the motion of regular-polygonal foldable loops was defined and simulated by way of two compatibility equations. Each of these equations was derived by imposing the constraint that the hinges at the end of each bar must lie within two static planes. This kind of constraint would not work for the doubly symmetric ring. In this section, the defining equations for a doubly symmetric 8-bar foldable ring are derived by using a series of transfer matrices, similar to those used in Section 3.3 (to define the closure of the 6-bar rings), and also in Gan & Pellegrino (2006). The entire loop can be designed by focusing on the first XY plane quadrant (as shown in Figure 5.3), and reflecting the result in the XZ and YZ planes to generate the rest of the ring. Setting the locations of the vertices (a, b and c) automatically defines the bar lengths l_1 and l_2, as well as the angles γ_a, γ_b and γ_c. The parameters $\alpha_a, \alpha_b, \alpha_c$ and β, and the hinge rotations ϕ_a, ϕ_b and ϕ_c remain to be determined. The lengths can be found simply by:

$$l_1 = \|b - a\| \quad l_2 = \|c - b\|$$

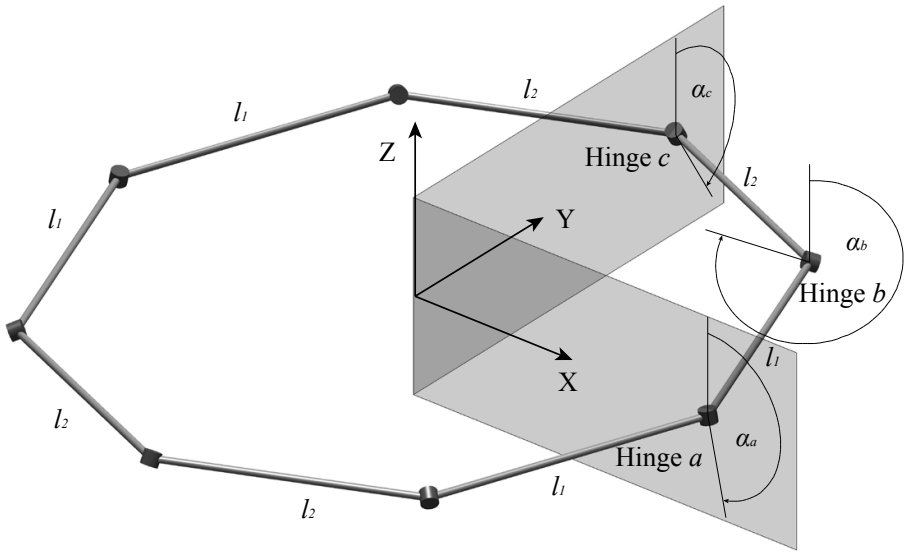


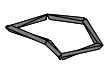
Figure 5.2: The doubly symmetric 8-bar foldable loop in the deployed configuration. Hinge inclinations to the vertical are shown. The two perpendicular planes of symmetry are shown bounding the first XY quadrant. Bar lengths l_1 and l_2 are also labelled.

5.1.1 A Loop Closure Equation

In order to construct the defining loop closure equation for the linkage, first establish some rotation matrices:

$$L_2(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \qquad L_3(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$L_{2B}(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad L_{3B}(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 & 0 \\ \sin(\psi) & \cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



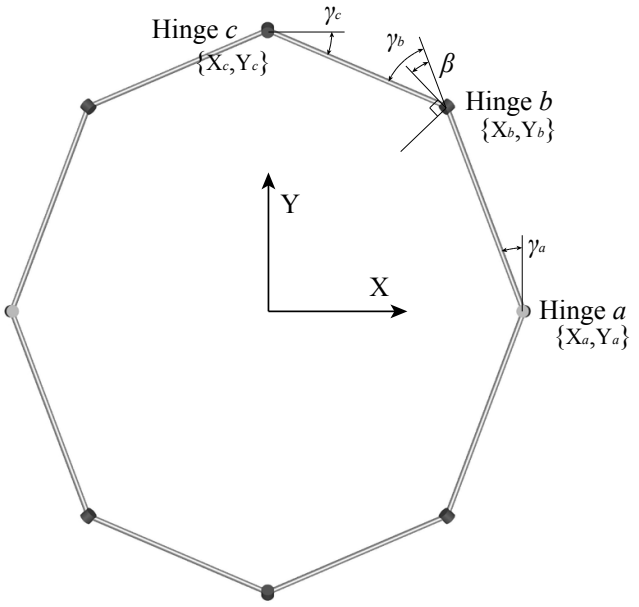


Figure 5.3: The doubly symmetric 8-bar foldable loop in the deployed configuration. Hinge XY plane locations are shown.

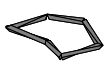
and some pure translation matrices:

$$M_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad M_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Now imagine a local coordinate system attached to hinge a, with its local x and z axes lying in the global XZ plane in the deployed configuration (though local and global axes are not necessarily aligned). To express a location at hinge c in the frame attached to hinge a, the following matrix transformation can be used:

$$\begin{aligned} L_B = L_{3B}(\phi_a)L_{2B}(-\alpha_a)L_{3B}(\gamma_a)M_1L_{3B}(\beta)L_{2B}(\alpha_b)L_{3B}(\phi_b)L_{2B}(-\alpha_b) \\ \ldots L_{3B}(\gamma_b - \beta)M_2L_{3B}(\gamma_c)L_{2B}(\alpha_c)L_{3B}(\phi_c) \end{aligned} \tag{5.1}$$

In this process, the local y axis is kept aligned with each bar during a translation step, and the local z axis is kept aligned with each hinge vector during a rotation about that hinge.



A rotation only version of Equation 5.1 is useful, and is given in Equation 5.2.

$$\begin{aligned} L_R = L_3(\phi_a)L_2(-\alpha_a)L_3(\gamma_a)L_3(\beta)L_2(\alpha_b)L_3(\phi_b)L_2(-\alpha_b) \\ \ldots L_3(\gamma_b - \beta)L_3(\gamma_c)L_2(\alpha_c)L_3(\phi_c) \end{aligned} \tag{5.2}$$

To maintain the two-fold symmetry of the ring, it is necessary to ensure that hinge vectors at a and c lie in each of the planes indicated in Figure 5.2 at all times. One way of enforcing this condition is to stipulate that the local y axes (at a and c) must be perpendicular to one another at all times (since the hinge vectors at a and c are defined as being coincident with the z axis of the local coordinate system at each end). This forces both the x and z axes of the local coordinate system attached to c to lie in the same plane from the deployed to stowed position. In this way, the **global Y axis is defined by the local y axis at a**, while the **global X axis is defined by the (negative of the) local y axis at c**. The global Z can be found simply as $X \times Y$. The governing equation for the system can be given in the form of Equation 5.3, which is expressed in terms of the local coordinates at a.

$$\left\langle L_R \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right\rangle = 0 \tag{5.3}$$

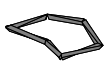
While this is not a 'closure' equation in the truest sense of the word, the use of transfer matrices in its derivation is sufficiently reminiscent of the loop closure techniques of Section 3.3 for the name to be loosely applied.

At present, the hinge b vector is initially specified using two parameters; α_b and β. In the design of a loop which is capable of folding up into a tight bundle, hinge b must be able to rotate such that bars 1 (length l_1) and 2 (length l_2) are parallel and coincident. It can be shown that for this to occur, it is necessary that $\beta = \gamma_b/2$. Using this, β can be removed from the preceding equations, to give:

$$\begin{aligned} L_B = L_{3B}(\phi_a)L_{2B}(-\alpha_a)L_{3B}(\gamma_a)M_1L_{3B}(\gamma_b/2)L_{2B}(\alpha_b)L_{3B}(\phi_b)L_{2B}(-\alpha_b) \\ \ldots L_{3B}(\gamma_b/2)M_2L_{3B}(\gamma_c)L_{2B}(\alpha_c)L_{3B}(\phi_c) \\ L_R = L_3(\phi_a)L_2(-\alpha_a)L_3(\gamma_a)L_3(\gamma_b/2)L_2(\alpha_b)L_3(\phi_b)L_2(-\alpha_b) \\ \ldots L_3(\gamma_b/2)L_3(\gamma_c)L_2(\alpha_c)L_3(\phi_c) \end{aligned}$$

5.1.2 Constraint Equations and Design Variables

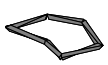
At this stage it is appropriate to begin to enumerate the design variables, and to consider some constraints which might be imposed on the system. It is necessary that the number



of each of these be equal to form a square system of polynomial equations. The parameters $\gamma_a, \gamma_b, \gamma_c, l_1$ and l_2 are all determined directly by the location of the three hinges a, b and c in the fully deployed configuration in the XY plane. The remaining unknowns are the three hinge inclinations α_a, α_b and α_c, which will serve as the design variables in the continuation design process. Some thought can now be given to what characteristics a practical foldable 8-bar loop should possess.

- Deployed shape:
  - deployed shape is directly specified by setting vertex locations in 1^{st} quadrant;
  - it is considered likely that the deployed shape will be a Convex Polygon for most purposes, so only shapes of this type will be studied here.
- Folding motion of loop:
  - to avoid bar collisions, each hinge must remain entirely within its own quadrant at all times;
  - hinge b should follow as simple a path as possible from deployed to stowed to minimise stress on any flexible sheet which might be attached to the frame;
  - since the linkage has, by its nature, multiple degrees of freedom, some of the hinges will have to be specially driven/guided;
  - the relationship between, and control of, these driven hinges should be as simple as possible.
- Stowed shape:
  - it is likely that any practical 8-bar loop of this type will not be required to fold into as tightly packed a bundle as possible, i.e. with bars stowed perfectly parallel;
  - some space should be left in the centre of the stowed linkage to accommodate a central deployment assistance mechanism, or a packed flexible sheet;
  - a top view of a stowed linkage is given in Figure 5.4, showing the dimensions of the internal space.

In Section 3.6, a plane symmetric $6R$ foldable frame was designed using numerical continuation. This was accomplished by defining sets of matched internal angles which acted as 'precision points' at various stages throughout the linkage's deployment. In Section 4.3, angles were again used to specify precision points, except that rather than



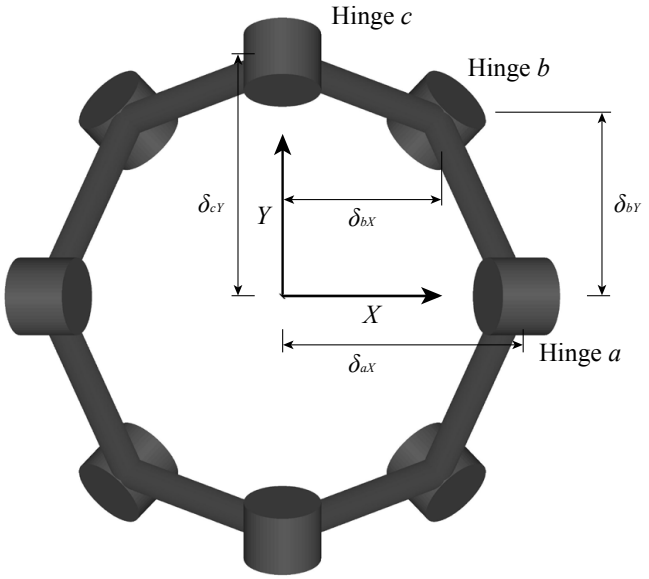
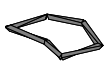


Figure 5.4: Top view of 8-bar linkage in stowed configuration.

internal angles, Euler angles were used to directly prescribe bar orientations in global coordinates. In a departure from the precision point design methods of previous chapters, the doubly symmetric 8-bar foldable ring in this chapter will be designed by way of exact specification of the linkage's stowed shape. Constraint equations will be written in terms of distances rather than angles (although the design variables themselves will again be angles). By choosing to focus only on the stowed shape one relinquishes all control over the deployment path. Choice of deployment path is relegated to the second, more subjective stage of the continuation design process in which once chooses designs from amongst the continuation derived results.

To achieve the stowed configuration shown in Figure 5.4, it is necessary to introduce three further unknowns in the form of the hinge angles in the stowed position, ϕ_{as}, ϕ_{bs} and ϕ_{cs}. This makes a total of six unknowns; $\{\phi_{as}, \phi_{bs}, \phi_{cs}, \alpha_a, \alpha_b, \alpha_c\}$. At present, there are no constraint equations, and one closure equation in the form of Equation 5.3.



Construct the following transfer matrix definitions:

$$\text{a)} \quad L_{B1} = L_{3B}(-\phi_{cs})L_{2B}(-\alpha_c)L_{3B}(-\gamma_c)M_2^{-1}L_{3B}(-\gamma_b/2)L_{2B}(\alpha_b)L_{3B}(-\phi_{bs}) \ldots L_{2B}(-\alpha_b)L_{3B}(-\gamma_b/2)M_1^{-1}$$

$$\text{b)} \quad L_{B2} = L_{3B}(\phi_{as})L_{2B}(-\alpha_a)L_{3B}(\gamma_a)M_1$$

$$\text{c)} \quad L_{B3} = L_{3B}(-\phi_{cs})L_{2B}(-\alpha_c)L_{3B}(-\gamma_c)M_2^{-1}$$

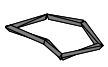
$$\text{d)} \quad L_{B4} = L_{3B}(\phi_{as})L_{2B}(-\alpha_a)L_{3B}(\gamma_a)M_1L_{3B}(\gamma_b/2)L_{2B}(\alpha_b)L_{3B}(\phi_{bs})L_{2B}(-\alpha_b) \ldots L_{3B}(\gamma_b/2)M_2$$

$$\text{e)} \quad L_R = L_3(\phi_{as})L_2(-\alpha_a)L_3(\gamma_a)L_3(\gamma_b/2)L_2(\alpha_b)L_3(\phi_{bs})L_2(-\alpha_b)L_3(\gamma_b/2)L_3(\gamma_c) \ldots L_2(\alpha_c)L_3(\phi_{cs})$$

and from this, a set of constraint equations can be formed, as in Equation 5.4.

$$\begin{aligned}
&\text{a)} \quad \left\langle L_{B1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right\rangle + \delta_{aX} = 0 \\
&\text{b)} \quad \left\langle L_{B2} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right\rangle - \delta_{bY} = 0 \\
&\text{c)} \quad \frac{Y_b}{X_b} \left\langle L_{B3} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right\rangle + \delta_{bY} = 0 \\
&\text{d)} \quad \left\langle L_{B4} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right\rangle - \delta_{cY} = 0 \\
&\text{e)} \quad \left\langle L_R \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right\rangle = 0
\end{aligned} \tag{5.4}$$

Equation 5.4a is expressed in the coordinates of the local axes attached to hinge c, and sets the distance hinge a should sit away from the origin in the X direction in the stowed



configuration. Equation 5.4b is expressed in the coordinates of the local axes attached to hinge a, and sets the distance hinge b should sit away from the X axis when stowed. Equation 5.4c is in local hinge c coordinates, and sets the distance hinge b sits away from the Y axis when stowed. The Y_b/X_b factor included here ensures that the position of hinge b in the deployed and stowed configurations, and the origin, are collinear. The inclusion of the Y_b/X_b factor also means that the offset δ_{bX} does not need to be specified explicitly. Equation 5.4d is in local hinge a coordinates, and sets the offset of hinge c from the X axis when stowed. Equation 5.4e is, of course, the definition equation of 5.3, but is included again here for completeness.

The four constraint and single closure equations given above are sufficient to specify a foldable linkage with several desirable characteristics. These five equations, together with the six design variables $\{\phi_{as}, \phi_{bs}, \phi_{cs}, \alpha_a, \alpha_b, \alpha_c\}$, form an under-determined system which could be used to design a family of linkage designs. As an alternative, an additional constraint is introduced which simply has the form:

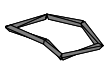
$$\phi_c = \phi_a$$

Stipulating that the hinge angles at a and c must be the same at all times may have some practical advantages. If these hinges were, say, identical spring loaded hinges with enough damping to cause them to expand at the same rate, the linkage would behave as though it had a single degree of freedom, and expand to be rigid. The four hinge b's could be left free to rotate. This new equality reduces the number of design variables to five, resulting in a square system of equations.

The form of Equation 5.4c, along with the constraint $\phi_c = \phi_a$, has an interesting consequence in terms of the motion of hinge b during deployment. It is the equivalent, mathematically, to including a further constraint equation requiring the projection of hinge b onto the XY plane to remain on the same line (formed by the hinge b location when deployed, and the origin) at all times during deployment (hinge b expands radially away from the centre during deployment). This equivalence can be observed by actually constructing such a constraint equation, and noting that its gradient with respect to $\{\phi_{a/c}, \phi_b, \alpha_a, \alpha_b, \alpha_c\}$ is always spanned by any four of the five rows of the Jacobian formed by the other five equations.

5.1.3 Continuation Results

Equation set 5.4 forms the basis of the equations used in the continuation process. As usual, trigonometric terms are replaced with new variables, doubling the number of un-



knowns. The unknowns are now $\{C_{\phi_{as}}, S_{\phi_{as}}, C_{\phi_{bs}}, S_{\phi_{bs}}, C_{\alpha_a}, S_{\alpha_a}, C_{\alpha_b}, S_{\alpha_b}, C_{\alpha_c}, S_{\alpha_c}\}$. Five new equations of the form:

$$\left.\begin{array}{l} C_{\phi_{as}}^2 + S_{\phi_{as}}^2 - 1 \\ C_{\phi_{bs}}^2 + S_{\phi_{bs}}^2 - 1 \\ C_{\alpha_a}^2 + S_{\alpha_a}^2 - 1 \\ C_{\alpha_b}^2 + S_{\alpha_b}^2 - 1 \\ C_{\alpha_c}^2 + S_{\alpha_c}^2 - 1 \end{array}\right\} = \mathbf{0}$$

are appended to the set of five in 5.4. The vertex locations, such as those in Table 5.1, are used to generate the coefficients for equation set 5.4, which is now a system of pure polynomials in terms of the ten unknowns. This system of ten equations in ten unknowns has a total degree of 22400, but a mixed volume of only 512. The system can then be solved using the polyhedral homotopy methods of Chapter 2.

A polynomial system with the same structure as 5.4, but with random complex coefficients, can be solved first and then used as the start system for each real valued design problem of the same type. Because polyhedral homotopy methods provide such a tight upper bound on the number of solutions to a system, the start system with random complex coefficients will have only finite solutions, the number of which is exactly equal to the mixed volume of the system. Any real valued system will only ever have the same number, or fewer solutions. To solve a real (or possibly another complex) valued system, one can construct a coefficient homotopy from the initial complex valued start system to the real or complex target. Constructing a coefficient homotopy from one real target system to another will not guarantee that all solutions will be found.

After using polyhedral homotopy methods to construct a random complex coefficient start system with structure identical to Equations 5.4, some design cases were examined using a coefficient homotopy. The results are discussed below.

Design Case # 1

To test the design method, some random vertex locations were selected, with the proviso that the deployed linkage be a convex polygon with $l_1 \approx l_2$, and vertices approximating a regular polygon for the first case. The initial parameters for this first example are given in Table 5.1. The particular value of the δ_{bY} offset was chosen in this case because it causes hinge b to be situated a distance of 0.15 units radially from the Z axis (the same distance as hinges a and c). This system yielded surprisingly few geometrically isolated solutions. Four distinct real designs were found, with their deployment paths shown in

Parameter	Value
Hinge a location	{1,0}
Hinge b location	{0.7,0.8}
Hinge c location	{0,1.1}
δ_{aX}	0.15
δ_{bY}	0.1129
δ_{cY}	0.15

Table 5.1: Initial parameters for 8-bar design example # 1.

Figure 5.5. Only one of the finite, real solutions (labelled as design # 1) was found not

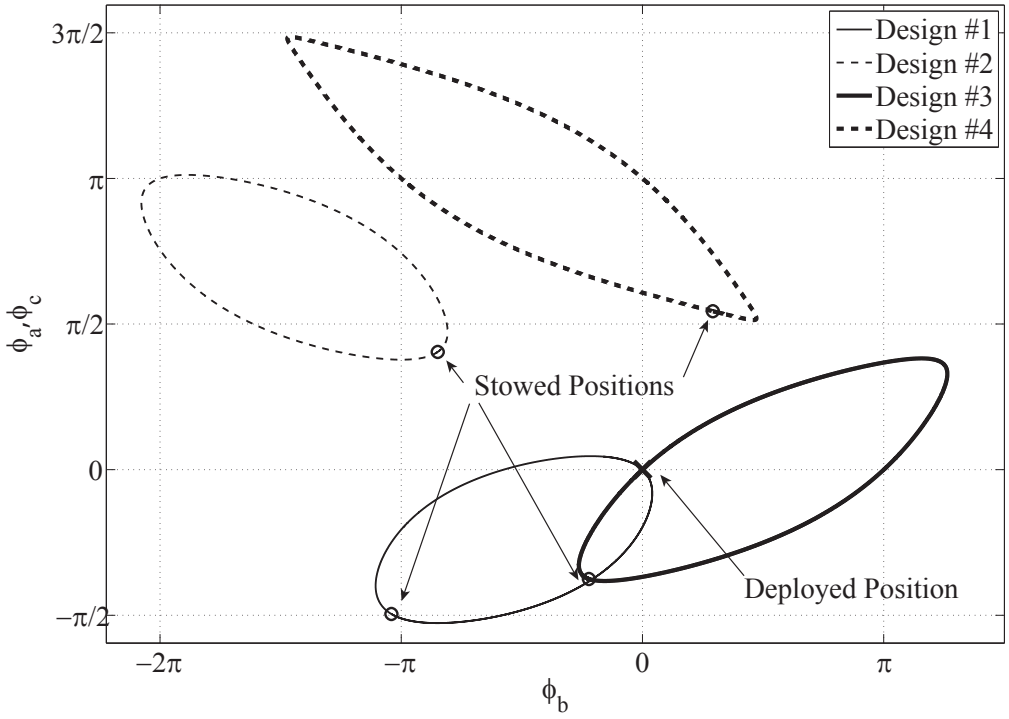


Figure 5.5: Internal angles for doubly symmetric 8-bar foldable ring, example # 1. All simulations were started at the ring stowed locations. Only design # 1 was found to progress satisfactorily from the stowed to deployed configuration.

only to satisfy the constraints, but to operate in a totally satisfactory manner. This entails all hinges opening smoothly from stowed to deployed, and all hinges staying within their quadrant at all times during deployment. Some designs are *degenerate*, and do not have uninterrupted paths from the stowed to the deployed configurations (such as designs #2 and #4), while others can move continuously from stowed to deployed, but have to self-intersect in order to do so. As expected, the vertical projection of hinge b onto the XY plane followed the line described by $Y = Y_b/X_b.X$, with only the height above this plane varying. A summary of the details about the single feasible design found is given in Table

5.2, along with a detail of the deployment path.

Although design # 1 does meet almost all the design criteria laid out earlier in the chapter, it is not quite ideal. This design was chosen over the other possible three because it deploys in a relatively straightforward manner, without any of the hinges crossing the X or Y axes during deployment. This said, hinge c comes quite close (to within a distance of 0.1 units) of the X axis during deployment, meaning the design would probably be rejected in practice. To find a truly acceptable design with these deployed vertex locations, the relationship $\phi_c = \phi_a$ would need to be relaxed, or at least modified. A flowchart of the linkage's folding from the deployed to stowed position is given in Figure 5.6.

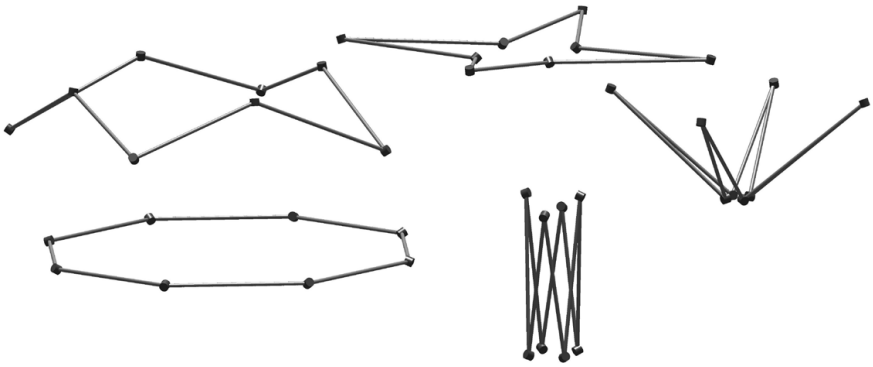


Figure 5.6: Folding of 8-bar example # 1, design # 1; clockwise from bottom left. Notice how close the c hinges come to one another in the second to last image.

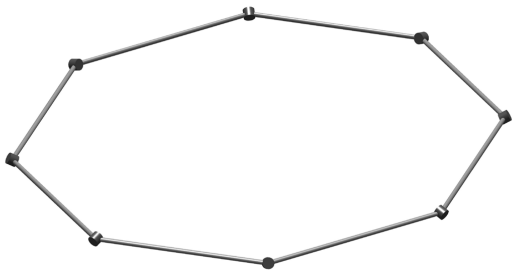
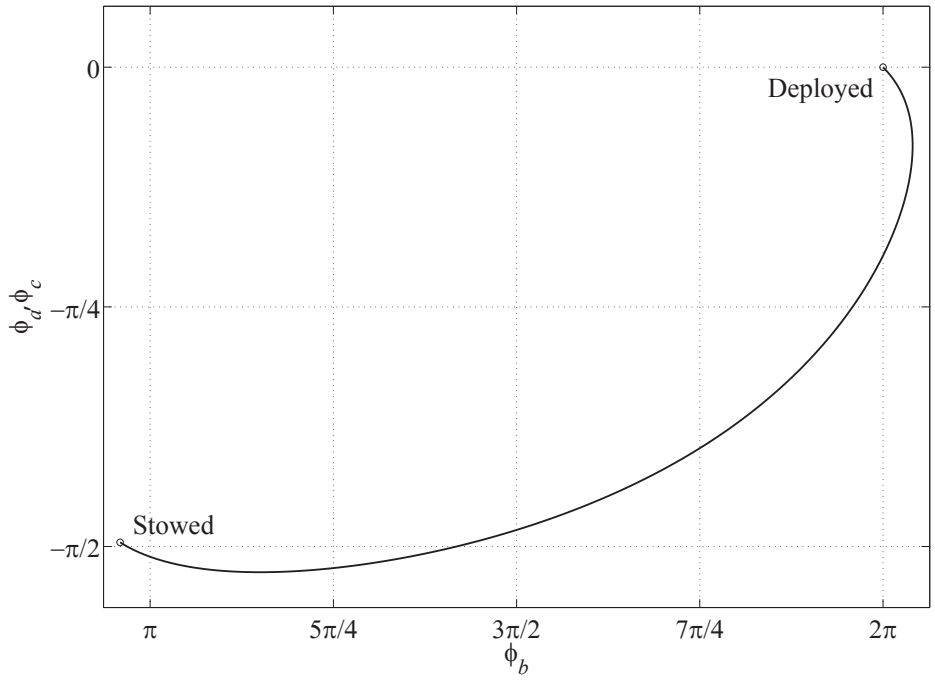
Solution Structure	
No. of real solutions found	32
No. of geom. distinguishable solutions	4
Geometry of Solution	
α_a	-1.2238 rad
α_b	1.0641 rad
α_c	-1.2686 rad
Deployed Loop	**Stowed Loop**
	
Angle Progression Plot	
	

Table 5.2: Data for 8-bar foldable ring, example # 1, design # 1.

Design Case # 2

For this example, vertex locations were chosen such that lengths l_1 and l_2 were significantly different from one another. This was to test the feasibility of the design procedure for loops which differ appreciably from a regular polyhedron. The initial parameters are given in Table 5.3. Again, four different real solutions were found, and their deployment

Parameter	Value
Hinge a location	{1,0}
Hinge b location	{0.8,0.4}
Hinge c location	{0,0.8}
δ_{aX}	0.15
δ_{bY}	0.06708
δ_{cY}	0.15

Table 5.3: Initial parameters for 8-bar design example # 2.

paths are shown in Figure 5.7. Once again also, only a single feasible solution was found

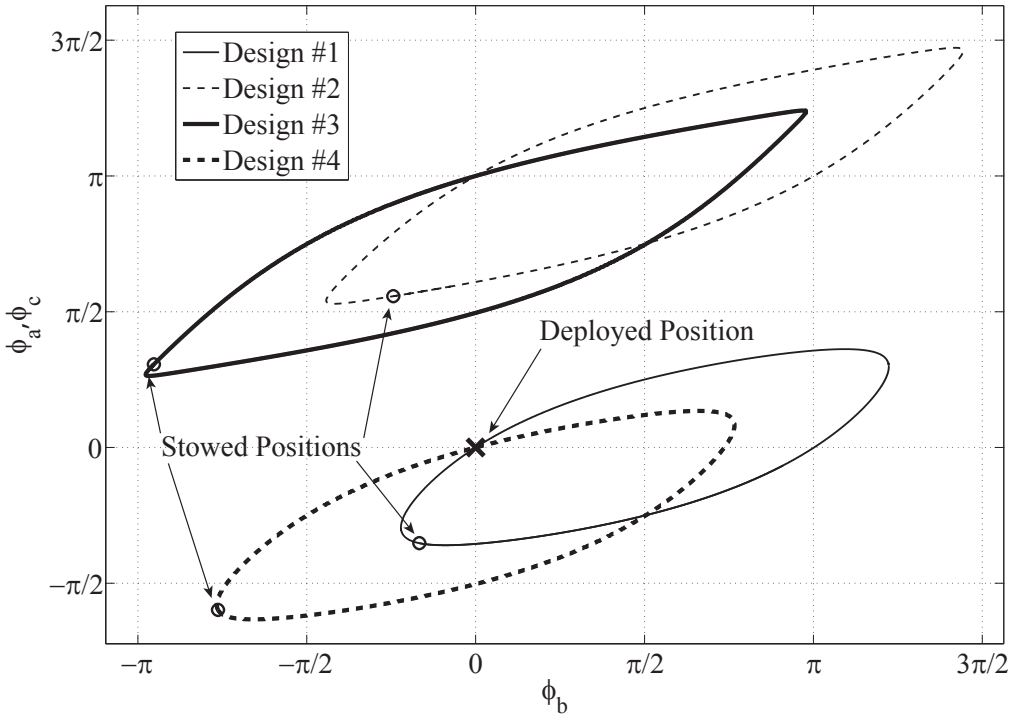


Figure 5.7: Internal angles for doubly symmetric 8-bar foldable ring, example # 2. All simulations were started at the ring stowed locations. Only design # 4 was found to progress satisfactorily from the stowed to deployed configuration.

(labelled as design # 4), the details of which are given in Table 5.4. Hinge c was found

not to venture too close to the X axis during deployment in this case, and a quite satisfactory linkage could most likely be constructed using the parameters listed here. The deployment is shown in Figure 5.8.

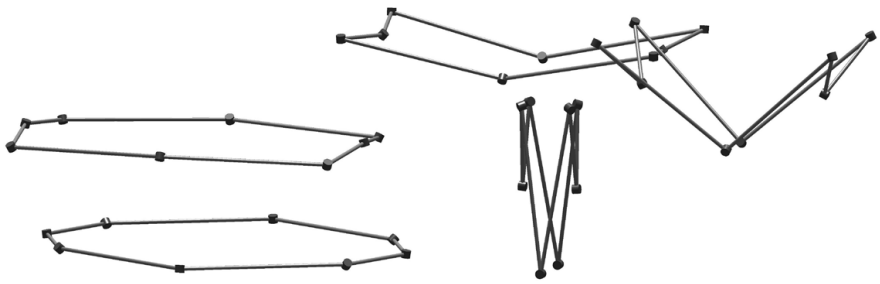


Figure 5.8: Folding of 8-bar example # 2, design # 4; clockwise from bottom left. In this case, the c hinges remain a safe distance away from one another during folding/deployment.

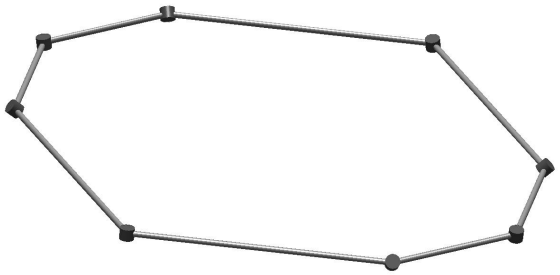
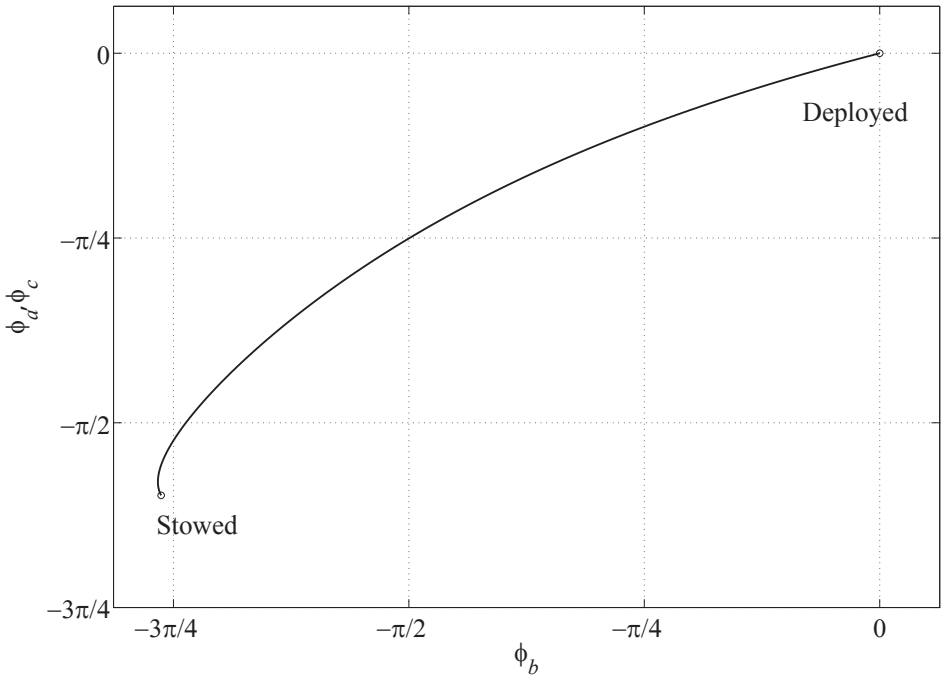
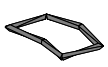
Solution Structure	
No. of real solutions found	32
No. of geom. distinguishable solutions	4
Geometry of Solution	
α_a	0.1400 rad
α_b	-2.2821 rad
α_c	-0.1400 rad
Deployed Loop	**Stowed Loop**
	
Angle Progression Plot	
	

Table 5.4: Data for 8-bar foldable ring, example # 2, design # 4.



5.2 Rectangular 8-Bar

One particularly interesting subset of the 8-bar foldable loop is the one for which:

$$\gamma_a = 0$$
$$\gamma_b = \pi/2$$
$$\gamma_c = 0$$

This describes a linkage which is rectangular in the deployed configuration. The extra degree of freedom present in the 8-bar rectangular linkage (as opposed to the 6-bar version, see Chapter 3) may allow for extra design constraints to be considered. A greater range of aspect ratios can also be achieved without the self intersection problems which can arise with the 6-bar rectangular linkage.

A rectangular linkage can be designed using polynomial continuation in much the same way as a linkage with arbitrarily positioned vertices. The only difference is that the new special values of the γ angles cause Equation 5.4b to become equivalent to 5.4c. This reduces the total number of equations to four. There are still five unknowns ($\{\phi_{as/cs}, \phi_{bs}, \alpha_a, \alpha_b, \alpha_c\}$), however, which means a new relationship needs to be introduced to square up the system. It is also possible, of course, that this under-determined system could be used to find a family of ring designs lying along the curve of solutions to the four equations in five unknowns. This will not be attempted here. Rather, a simple yet practically desirable new relationship between the unknowns will be sought.

5.2.1 Continuation Results

To overcome the lack of constraint equations, a new relationship of the form $\phi_{bs} = \pm 2\phi_{as}$ was introduced. The rationale behind this is that the hinge angles ϕ_a and ϕ_c actually only constitute 'half' angles, in that they span the angle from one bar to a central plane of symmetry. In the construction of a physical model, a hinge which opened to twice that value would be required to link one bar to the next. It was hypothesised that if hinge b could be set to open to twice the value of the two (equal) half-angles of hinges a and c, then perhaps only **a single type of hinge**, designed to open only as far as a certain fixed angle, would be required to construct the entire loop. This reduced 4×4 system of polynomials has a mixed volume of only 96. An illustrative example of the design process for a rectangular 8-bar linkage is given below.

Design Case # 3

The initial design parameters for a rectangular loop are given in Table 5.5.

Parameter	Value
Hinge a location	{1.1,0}
Hinge b location	{1.1,0.7}
Hinge c location	{0,0.7}
δ_{aX}	0.1
δ_{bY}	0.0537
δ_{cY}	0.1

Table 5.5: Initial parameters for 8-bar design example # 3.

This time, only sixteen real solutions were found, but again, four of them were geometrically distinct. The four deployment paths are shown in Figure 5.9. Note that, on this occasion, the deployed configuration was used as the start point for all the simulations depicted. Two of the rings were found to have their deployed and stowed locations

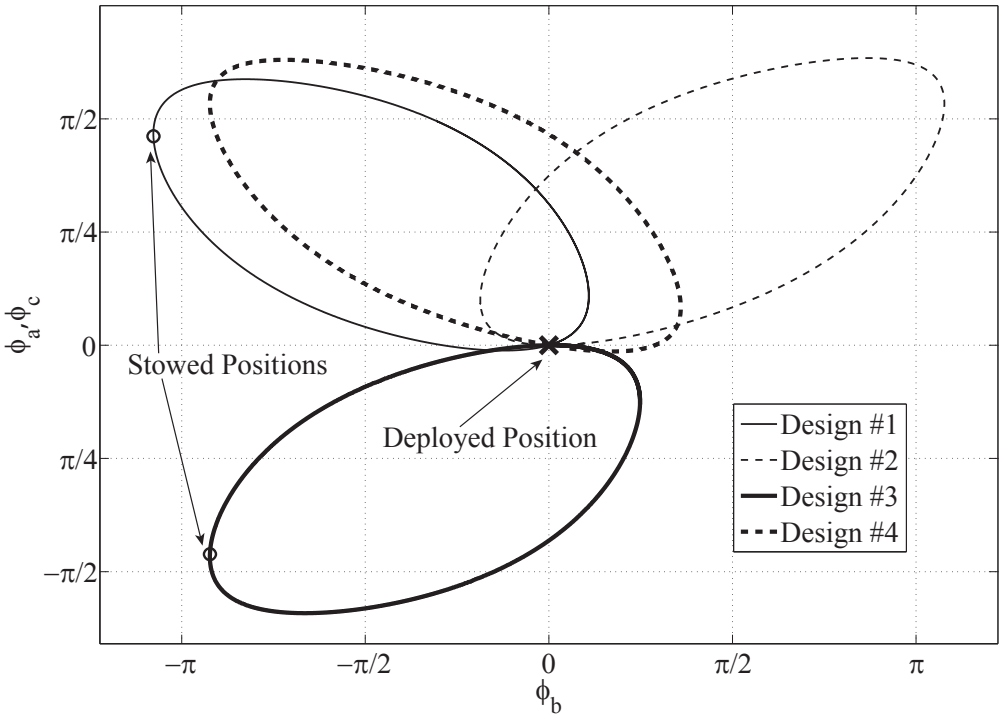
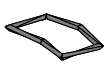


Figure 5.9: Internal angles for doubly symmetric 8-bar foldable ring, example # 3. All simulations were started at the ring deployed locations. Only design # 1 was found to progress satisfactorily from the deployed to stowed configuration.

connected by an uninterrupted deployment path. Of these, only one (design # 1) behaves



entirely satisfactorily. The details of its design are given in Table 5.6. What is immediately apparent is that $\phi_{bs} \neq \pm 2\phi_{as}$, since when deployed, both angles are zero, and when stowed:

$$\begin{aligned} \phi_{as} &= 1.4500 \text{ rad} \\ \phi_{bs} &= -3.3832 \text{ rad} \\ \cos(\phi_{bs}) &= \cos(2\phi_{as}) \\ \sin(\phi_{bs}) &= \sin(2\phi_{as}) \end{aligned}$$

This is not surprising, as the specification $\phi_{bs} = 2\phi_{as}$ has the equivalent effect on the equations of setting:

$$\begin{aligned} \cos(\phi_{bs}) &= \cos^2(\phi_{as}) - \sin^2(\phi_{as}) \\ \sin(\phi_{bs}) &= 2\sin(\phi_{as})\cos(\phi_{as}) \end{aligned}$$

As it happens, no feasible solutions satisfying $\phi_{bs} = \pm 2\phi_{as}$ were found for this rectangular linkage. The best result, this considered, is shown in Table 5.6. The folding process is shown in Figure 5.10. It is possible that a rectangular frame with a different aspect ratio might be deployable with $\phi_{bs} = \pm 2\phi_{as}$, but none has been found to date.

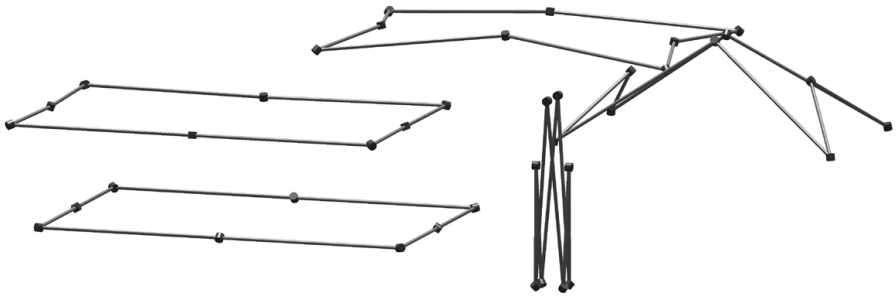
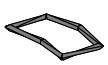


Figure 5.10: Folding of 8-bar example # 3, design # 1; clockwise from bottom left. In this case, the c hinges remain a safe distance away from one another during folding/deployment.



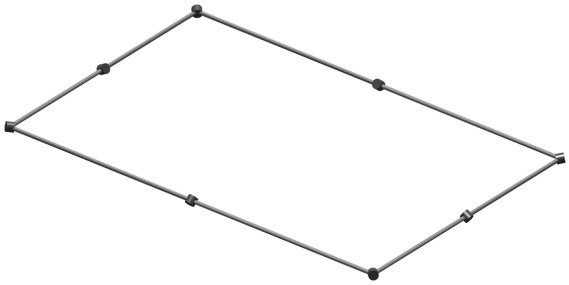
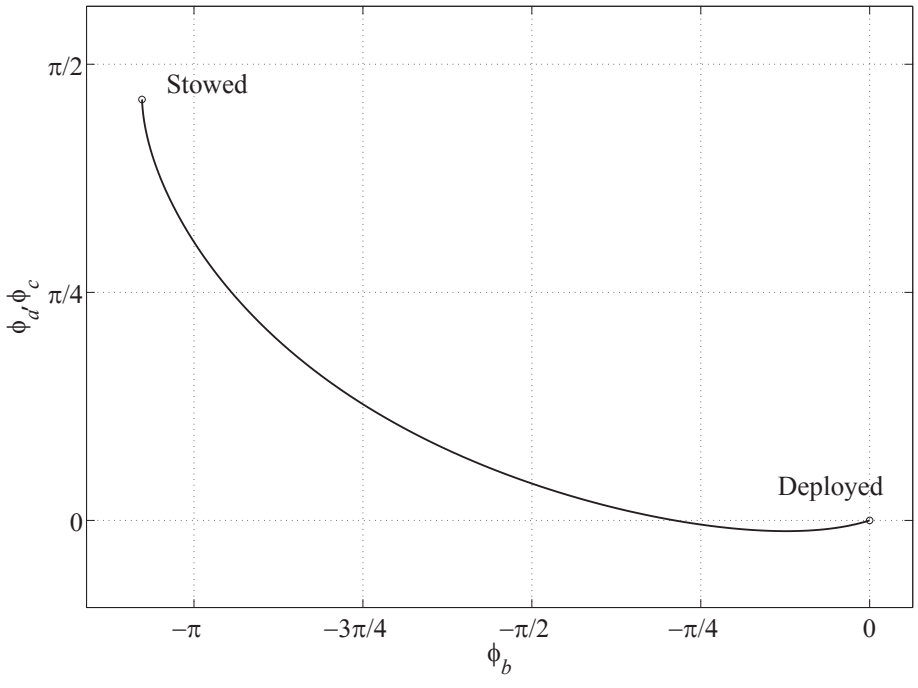
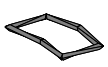
Solution Structure	
No. of real solutions found	16
No. of geom. distinguishable solutions	4
Geometry of Solution	
α_a	-1.0613 rad
α_b	-1.7585 rad
α_c	-0.8795 rad
Deployed Loop	**Stowed Loop**
	
Angle Progression Plot	
	

Table 5.6: Data for 8-bar foldable ring, example # 3, design # 1.



5.3 Deployment of 8-Bar Rings

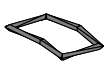
Rather than use a 4, 5 or 6-bar linkage, a designer might choose to use an 8-bar linkage as the basis of a deployable ring because of the greater range of deployed shapes possible, and because of the greater control which can be exercised over the way in which the frame deploys. However, some extra complications need to considered. Many deployable frames based on 4, 5 or 6-bar linkages have only a single degree of freedom, and hence have pre-determined deployment paths which can often be easily followed simply by applying sufficient moments at one or more of the hinges (Laloi, 1999; Pellegrino *et al.*, 2000). By their nature, 8-bar spatial linkages (arranged in a loop) will have at least two degrees of freedom. It is important that the extra benefit derived from the increased versatility in shape outweigh the increase in complexity of the deployment system. The double symmetry of the frames considered in this chapter means that hinge b is replicated four times around the loop, while hinges a and c are replicated twice each. If (as was the case in all examples given above), hinge angles ϕ_a and ϕ_c are set to be equal, then this leaves two distinct hinge angles present in the loop. At this stage, the best method of deploying a doubly symmetric 8-bar frame is unknown, although it is possible that a set of four identical, damped, spring-loaded hinges attached to either hinges b or a, c might suffice.

The charts displaying angles ϕ_a, ϕ_c vs. ϕ_b provide some valuable information about the deployability of a given 8-bar design. If hinge b is to be the driven hinge, then ϕ_b needs to be a monotonic function of ϕ_a, ϕ_c, and if a, c is to be the driven hinge, then ϕ_a, ϕ_c must be a monotonic function of ϕ_b over the deployment range.

Consider the case of example #1, design # 1. Looking at the angular progression at the bottom of Table 5.2, it can be seen that the linkage would not fully deploy if hinges a and c were designated as the driven hinges ($\phi_a,\ \phi_c$ do not increase/decrease monotonically). In fact, the same problem would be encountered with hinge b selected as the driven hinge. This design could clearly not be deployed with passive actuators such as springs, or if it was, it could not be guaranteed that the deployment would follow anything like the prescribed deployment path.

Next consider example # 2, design # 4, whose deployment path is shown at the bottom of Table 5.4. Notice that this time, $\phi_a,\ \phi_c$ do increase/decrease monotonically, meaning that spring loaded hinges could, most likely, be used to deploy the ring without causing it to deviate from its prescribed deployment path.

Finally, consider example # 3, design # 1, whose deployment path is shown at the bottom of Table 5.6. This time ϕ_b increases/decreases monotonically with $\phi_a,\ \phi_c$, but the



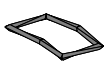
converse is not true. Hinge b presents itself as the only candidate for a driven hinge.

5.4 Conclusion

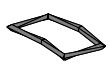
The deployable rings in Chapter 4 are regular polygons in their deployed state. The study of the doubly symmetric foldable rings in this chapter was intended as an extension by way of an exploration into the design of foldable rings which form non-regular polygons in their deployed state. It is conceivable that not all deployable membranes might require a regular polygon shaped frame (approximating a circle). The aim of this chapter was also to illustrate a different way of specifying design constraints in the process of linkage design using polynomial continuation. An 8-bar version of the doubly symmetric foldable ring was used for this purpose. As an alternative to the use of angular precision points to specify the way in which a ring deploys, a series of dimensions, prescribing an exact stowed shape, was introduced in the form of constraint equations. The deployed shape was directly specified in all cases, and the exact dimensions of the stowed ring were used to generate the coefficients for the polynomials (whose monomials were written in terms of the unknown design variables).

Two different deployed shapes, whose vertices were essentially randomly located (but still forming a convex polygon), were used as design examples. Each yielded four distinct real solutions, but only one which could be considered feasible in that (a) no self-intersection was observed, and (b) a single connected path existed between the deployed and stowed configurations. A rectangular deployed shape was also considered, and a simplified version of the compatibility/constraint equation system introduced. The rectangular design was also found to yield four distinct real solutions, and a single feasible design.

The three design examples included in this chapter together constitute a compelling case for the use of polynomial continuation in the linkage design process. In each case **more than one set of design parameters (solutions) were found to satisfy the imposed constraints, leaving the designer with a number of valid options**. Different criteria can be used to select which of the feasible solutions best fits the design objective. In this case, the design priority which guided the solution selection was avoidance of self intersection during deployment. Choosing among the solutions determined via the continuation process, is to some extent a matter of personal judgement. Once a designer has determined which features a linkage absolutely must possess, and has found, via continuation, all the combinations of design parameters which will satisfy those constraints, he/she is free to exercise his/her discretion in selecting the best.



Issues concerning the deployment of 8-bar doubly symmetric foldable frames were discussed, with particular reference to the three design cases considered earlier. Deployment of such rings using passive actuators may not be possible in situations in which deployment paths require hinge angles which do not vary monotonically.



6. Mobility of 6,7 & 8-Link N-Loops

This chapter marks a slight departure from the theme of preceding chapters. The focus in this final chapter is not the development of polynomial continuation-based design tools as applicable to practical foldable rings, but rather the use of continuation to gain a greater understanding, at a theoretical level, of the closure equations of mobile loops. This topic was introduced briefly in Chapter 3, Section 3.3, but is covered here in more detail.

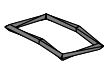
The examples used in this chapter are a specific type of closed loop called *N-loops* (where N is the number of links present). Some quite simple examples of these loops exhibit some interesting characteristics from the point of view of mobility. These characteristics can be explored in a very hands-on way by using a commercially available type of toy whose components resemble those of N-loops when connected in a ring (see next section). As a kind of shorthand, the 'N' in N-loops will occasionally be replaced by a numeral.

In this chapter, polynomial continuation is used to probe the nature of the equations describing N-loops. In the first instance, the 6-link N-loop is examined, yielding arguably the most interesting results. The existence of both rigid and mobile conformations is suggested by the structure of the 6-loop's closure equations, enabling a mathematical understanding of the linkage's mobility. 7 and 8-loops are then examined, focussing only on the positive dimensional solutions which are known to exist in their closure equations.

6.1 'Tangle' N-Loops

The toy, 'Tangle Creations,' is "a series of 90° curves, connected and able to pivot at each joint."[1] More specifically, it is a collection of quarter arcs, with each piece able to connect to another end-to-end, allowing a single (longitudinal) rotational degree of freedom between them. The links can be connected end-to-end to form a closed loop. The most simple example of such a closed loop consists of four individual pieces connected to form a rigid circle. A basic application of the Kutzbach criterion (Equation 3.3) suggests

[1] https://www.tanglecreations.com

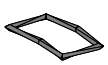


that a closed loop of links requires at least seven elements (in the spatial case) in order to display any mobility (seven links connected in a loop using seven single degree of freedom joints). A reasonable question is: given that other forms of mobile 6-bar linkages exist, is it possible to construct a mobile loop of Tangle pieces using only six, or fewer, elements? As it happens, there are two different ways in which six Tangle pieces can be connected together into a loop. One of the conformations is rigid, as predicted by the Kutzbach criterion, while the other possesses a geometric degree of freedom, and is mobile.

Recently, a version of the Grübler criterion was used to count the kinematic degrees of freedom in N-loops (Guest & Fowler, 2010). Porta *et al.* (2007) produced a complete map of the conformation space for similar (molecular) loops by manipulating distance constraints into Cayley-Menger determinants, and then finding their roots using numerical techniques. In this chapter the mobility of N-loops is considered again, but the problem of finding and predicting mobilities which might occur is approached from a different, and entirely numerical, perspective. Each link is assumed to be identical, and the key dimensions of each link can be set to unity without loss of generality (in preparation for numerical analysis). At its core, the analysis involves the solution of a system of closure equations for the linkage. The unknowns are the twist angles between each of the links, necessary to ensure loop closure. Two main categories of solutions can be expected; geometrically isolated (sometimes called zero-dimensional) points, and positive dimensional solutions sets. Positive dimensional solutions indicate that a continuum of solutions which satisfy the closure equations exists, which perhaps also indicates an internal mobility in the physical linkage. The structure of the closure equations for the 6-link Tangle loop is of particular interest because the loop is known to possess a certain amount of mobility, depending on the exact conformation.

6.2 6-Link N-Loops

A Tangle piece is quite simple to describe geometrically. Figure 6.1 shows a Tangle piece in plan view. Each element is connected socket 1 to socket 2 in a loop. Closed linkages of this nature can be described succinctly using an arrangement of transfer matrices. A transfer matrix is used to move between coordinate systems attached to each of the links (both a rotation and a translation); see Sections 3.3 and 5.1.1, as well as Chen & You (2009); Gan & Pellegrino (2006). Position vectors are augmented with a 1: $[x, y, z, 1]^T$. Transfer matrices of this nature can be expressed as a combination of geometric and 'state' components. One way of expressing the rotation between one link and the next



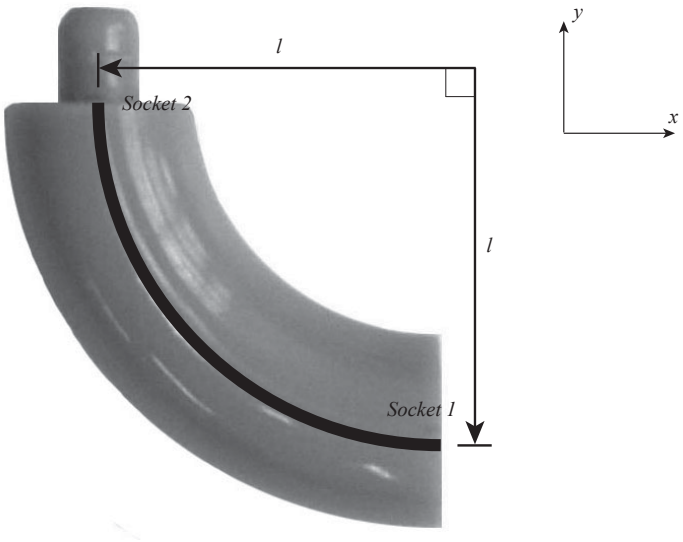


Figure 6.1: Tangle piece in plan view.

about the x-axis is a simple rotation matrix:

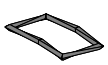
$$T_s = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_\phi & S_\phi & 0 \\ 0 & -S_\phi & C_\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This is the 'state' part of the matrix, and can be applied in local coordinates at socket 1 (Figure 6.1) to describe a rotation between the current link and the next. The angle ϕ determines the magnitude and sign of the rotation. The geometric part of the transfer matrix is fixed, and performs a rotation of $90°$ about the z-axis, as well as two translations of magnitude l.

$$T_g = \begin{bmatrix} 0 & 1 & 0 & -l \\ -1 & 0 & 0 & l \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The two parts can be combined to form the complete transfer matrix:

$$T = T_g T_s = \begin{bmatrix} 0 & C_\phi & S_\phi & -l \\ -1 & 0 & 0 & l \\ 0 & -S_\phi & C_\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.1}$$



A loop closure equation can be written as:

$$F = T_{61}T_{56}T_{45}T_{34}T_{23}T_{12} - I = 0$$

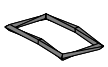
which is in terms of the six rotations $\{\phi_{12}, \phi_{23}, \phi_{34}, \phi_{45}, \phi_{56}, \phi_{61}\}$. In the solution of the closure equations, no prior assumptions about how the loop is to be formed need be made. The strictly upper triangular part of F provides six independent equations, which can be used to form a square system.

$$\left.\begin{matrix} F_{1,2} \\ F_{1,3} \\ F_{1,4} \\ F_{2,3} \\ F_{2,4} \\ F_{3,4} \end{matrix}\right\} = \mathbf{0}$$

The (real part of the) solution space of these equations is equivalent to the Conformational Space of the Tangle linkage. Quite often, the vast majority of solutions to an unhomogenised system of polynomial equations are *solutions at infinity*. Before applying continuation, there is no way to determine which of the start system's solutions will track to solutions at infinity in the target system. One way to reduce the number of paths which diverge to infinity is to reduce the degree, or order, of the target (and hence also the start) system. This can be achieved by way of elimination (Univariate polynomials have only solutions in the finite plane), but in this case a reduction in degree can be achieved simply by dividing the closure equation, placing three terms on each side:

$$F' = T_{34}T_{23}T_{12} - (T_{61}T_{56}T_{45})^{-1} = 0$$

The maximum degree of each term in F' is now three. The final step in constructing a system of equations appropriate for use with continuation is to express each equation in terms of the sines and cosines of the rotation angles, rather than the angles themselves. This doubles the number of unknowns, and the number of equations. The new variables are assigned as $C_{\phi_{ij}} = \cos(\phi_{ij})$ and $S_{\phi_{ij}} = \sin(\phi_{ij})$. The final system is given in Equation



6.2.

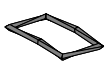
$$\left.\begin{array}{l} F'_{1,2} \\ F'_{1,3} \\ F'_{1,4} \\ F'_{2,3} \\ F'_{2,4} \\ F'_{3,4} \\ C^2_{\phi_{12}} + S^2_{\phi_{12}} - 1 \\ C^2_{\phi_{23}} + S^2_{\phi_{23}} - 1 \\ C^2_{\phi_{34}} + S^2_{\phi_{34}} - 1 \\ C^2_{\phi_{45}} + S^2_{\phi_{45}} - 1 \\ C^2_{\phi_{56}} + S^2_{\phi_{56}} - 1 \\ C^2_{\phi_{61}} + S^2_{\phi_{61}} - 1 \end{array}\right\} = \mathbf{0} \tag{6.2}$$

This system has a total degree of $486 \times 2^6 = 31104$, but a mixed volume of only 1472. The system could be solved directly using polyhedral methods, locating all the zero-dimensional solutions, but this would not reveal much about the structure of the equations except, perhaps, for hints of the existence of higher dimensional solution sets in the form of patterns of singular solutions.

(a) 6-Loop Mobile Conformation. (b) 6-Loop Rigid Conformation.

Figure 6.2: The two (mobile and rigid) 6-loops (from Guest & Fowler (2010)).

One method of probing the structure of a system of polynomial equations, and in particular, identifying higher dimensional solution sets, is the method of Witness Sets. This method was introduced in Section 2.5, and used in Section 3.3 to gain a greater



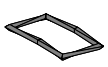
understanding of the mobility of the plane symmetric 6R foldable ring. The construction of witness sets by a cascade of homotopies is a top-down process; starting with a search for elements on the highest possible dimensional set, and working downwards. To take the most general approach would involve introducing $n - 1 = 11$ hyperplanes and new variables, and implementing 11 cascade homotopies, one after the other. This would result in an initial system in 23 unknowns; a formidable computational task. Instead, some *a posteriori* knowledge is used to rule out the existence of solutions of dimension 2 or greater, since none are observed in practice, and no unusual singular solutions appear during the search for 1 and 0 dimensional sets. This being the case, a single hyperplane and a single extra unknown are appended to system 6.2. This has no affect on the total degree of the system, but does raise the mixed volume to 4352. While 4352 is a large number of solution paths to track, the task is not an impossible one.

6.3 Solution Structure of 6-Loop Equations

The first step in solving the combined system of Equation 6.2 and the extra hyperplane is to construct and solve a random complex coefficient system with the same polynomial structure. This provides a start system with the full complement of 4352 equations which can be reused in the solution of different target systems. The next step is to use the random complex coefficient start system to solve the target system directly using a coefficient homotopy. Equation 6.2 is written in terms of the twelve original unknowns, as well as a length parameter l. Without loss of generality, this length can be set to 1.

6.3.1 Solutions of Dimension 1

Using a start system with the full complement of 4352 equations to solve the target system for the dimension 1 case leads to 2206 finite solutions. Of these, twelve satisfy the condition $z = 0$, where z is the new introduced variable which only becomes zero when a solution of dimensionality > 0 has been found. This immediately suggests that it might be possible to construct a closed loop linkage of six Tangle pieces which has some internal mobility. Looking more closely at the solutions, in can be observed that each satisfies the relationship $x_i = x_{i+3} \;\; i = 1, \ldots, 6$ (addition modulo 6). This suggests some relevance to the rotational symmetry which the mobile form of the six element N-loop is known to actually possess. So far it can only be stated that the full witness set consists of twelve members. It also needs to be determined if these twelve solutions form part of the same irreducible component, or whether there are a number of separate components. This



is achieved by using a method called Monodromy (see Section 2.5.1). In this case, all twelve solutions were observed to permute amongst themselves during the monodromy process, meaning that only a single positive dimensional solution set (dimension 1) exists, and that it has degree 12.

Table 6.1 provides a list of some of the (real) points which lie on the continuum of solutions of degree 12 (and dimension 1). Figure 6.2(a) illustrates a real model of a mobile 6-loop.

Point #	1	2	3	4	5	6
ϕ_{12}	0	$-\frac{\pi}{2}$	$\frac{\pi}{2}$	0	$\frac{\pi}{2}$	$-\frac{\pi}{2}$
ϕ_{23}	$\frac{\pi}{2}$	0	$-\frac{\pi}{2}$	$-\frac{\pi}{2}$	0	$\frac{\pi}{2}$
ϕ_{34}	$-\frac{\pi}{2}$	$\frac{\pi}{2}$	0	$\frac{\pi}{2}$	$-\frac{\pi}{2}$	0
ϕ_{45}	0	$-\frac{\pi}{2}$	$\frac{\pi}{2}$	0	$\frac{\pi}{2}$	-$\frac{\pi}{2}$
ϕ_{56}	$\frac{\pi}{2}$	0	$-\frac{\pi}{2}$	$-\frac{\pi}{2}$	0	$\frac{\pi}{2}$
ϕ_{61}	$-\frac{\pi}{2}$	$\frac{\pi}{2}$	0	$\frac{\pi}{2}$	-$\frac{\pi}{2}$	0

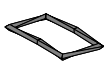
Table 6.1: A selection of points on the 6-bar N-loop 1-dimensional irreducible component. Each of these points is observable as a real configuration of the 6-bar N-loop (as well as the continuum of solutions between each point).

6.3.2 Solutions of Dimension 0

Once the full witness set for the solutions of dimension 1 has been found, a cascade homotopy can be used to reduce the dimension of the remaining 2194 solutions. In this process, 1994 of these solutions diverged to infinity. This left 200 solutions in the twelve original unknowns. Of these 200 solutions, 32 were identified as being members of the 1-dimensional irreducible component, and hence removed. This left 168 true zero dimensional (geometrically isolated) solutions. The vast majority of these 168 are complex and/or non-physical singular. In fact, only two non-singular real solutions remain, and these are given in Table 6.2. Up to rotational symmetry, these two solutions represent the one rigid conformation observed to exist in the 6-link N-loop. Figure 6.2(b) illustrates a real model of a rigid 6-loop. A graphical representation of the cascade process is given in Figure 6.3.

6.4 7 & 8-Loops and Their Mobility

While it may have been unexpected that the 6-loop should have a mobile conformation, it comes as no surprise that the 7 and 8-loops are both mobile. A direct application of



Solution #	1	2
ϕ_{12}	$\frac{\pi}{2}$	$-\frac{\pi}{2}$
ϕ_{23}	$-\frac{\pi}{2}$	$\frac{\pi}{2}$
ϕ_{34}	$\frac{\pi}{2}$	$-\frac{\pi}{2}$
ϕ_{45}	$-\frac{\pi}{2}$	$\frac{\pi}{2}$
ϕ_{56}	$\frac{\pi}{2}$	$-\frac{\pi}{2}$
ϕ_{61}	$-\frac{\pi}{2}$	$\frac{\pi}{2}$

Table 6.2: The two non-singular real zero-dimensional solutions to the 6-loop equations. These two solutions describe the only observed rigid conformation of the 6-loop, up to rotational symmetry.

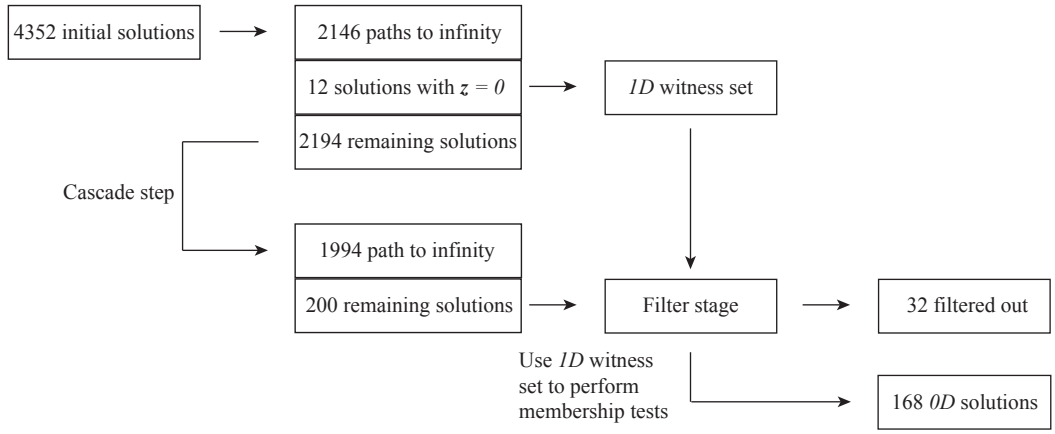
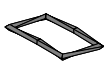


Figure 6.3: Reduction process for 6-Loop closure equations (Equation 6.2).

the Kutzbach criterion suggests that the 8-loop should possess two degrees of freedom, and the 7-link, one. This is indeed the case. Neither the 7 nor 8-loop is known to possess a rigid conformation, removing the need to perform a full homotopy cascade on each of the loops' closure equations to fully examine their nature. Something, however, can be said about the positive dimensional solution sets which are present in each loop's closure equations, and which give rise to their mobility.

6.4.1 The Degree of the 8-Loop's Mobility

To construct a set of closure equations for the 8-loop, a similar set of transfer matrices to those in Section 6.2 are used, with the addition of two extra transfer matrices to account



for the two extra links:

$$\begin{aligned} F = \quad & T_{81}T_{78}T_{67}T_{56}T_{45}T_{34}T_{23}T_{12} - I = 0 \\ \Rightarrow F' = \quad & T_{45}T_{34}T_{23}T_{12} - (T_{81}T_{78}T_{67}T_{56})^{-1} = 0 \end{aligned}$$

Once again, only the upper six off-diagonal entries contribute:

$$\left.\begin{matrix} F'_{1,2} \\ F'_{1,3} \\ F'_{1,4} \\ F'_{2,3} \\ F'_{2,4} \\ F'_{3,4} \end{matrix}\right\} = \mathbf{0}$$

This leaves a system in eight unknowns and six equations. The under-determined nature of the system immediately suggests that mobility could be present, and that there will be (in general) two degrees of freedom in the loop's motion. Two intersecting hyperplanes are added, resulting in a determined system. To calculate the degree of the positive dimensional solution sets present, it is necessary to enumerate the locations in which the hyperplanes are intersected by the solution paths, while accounting for any multiplicities (although the use of complex coefficients in the random intersecting hyperplanes generally obviates the need to consider multiplicities). Only one intersection point on each irreducible component is required to generate all the others. By using monodromy to expand the sets representative of each irreducible component, it is possible to generate the full witness set. The process may begin with any subset of intersection points which contains at least one member on each component.

It has been suggested[1] that the 8-loop may possess two 'lobes' of feasible configuration sets, connected only by their closures at the point at which the 8-loop forms a double circular ring. This raises questions about whether there may be two separate irreducible components in the dimension-2 solution set, with one representing each of the 'lobes'. To test this, two configurations, each thought to exist in different solution lobes, were generated. A 'crown' configuration, in which the twist angle between each link differs from the next only in sign, was used for this purpose. An illustration of the crown is given in Figure 6.4. The 'right way up' crown is given by:

$$\phi_{i(i+1)} = \pm \arccos(1 - \sqrt{2}) \approx \pm 114^\circ \quad i = 1, \ldots, 8$$

[1] via a personal communication, R. Connelly.

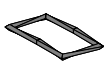


Figure 6.4: Crown configuration of the 8-loop (from Guest & Fowler (2010)).

and the 'upside down crown' is given by:

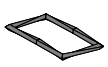
$$\phi_{i(i+1)} = \mp \arccos(1 - \sqrt{2}) \approx \mp 114^\circ \quad i = 1, \ldots, 8$$

For each of these configurations, two complex-valued hyperplanes which passed through these (real) points were found, and the process of monodromy employed to generate full sets representing the irreducible components associated with each. Both the 'right way up' and the 'upside down' irreducible components were found to contain 48 points, and a quick mapping from one set to the other revealed that they were in fact the same set. It is postulated, therefore, that a single dimension-2 irreducible component (of degree 48), is responsible for all the mobility of the 8-loop.

6.4.2 The Degree of the 7-Loop's Mobility

If seven Tangle pieces are connected in a loop, then without any prior knowledge it could be assumed that some mobility, most likely a single degree of freedom, will be present. In fact, two separate single degree of freedom conformations exist. One is dubbed the 'boat' conformation, and appears in Figure 6.5(a). The other, named the 'chair' conformation, appears in Figure 6.5(b).

One obvious question which arises is: do the two observed mobile conformations lie on the same irreducible component, or do they form two distinct parts of the one-



(a) 7-loop boat conformation. (b) 7-loop chair conformation.

Figure 6.5: 7-loop examples (from Guest & Fowler (2010)).

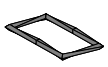
dimensional solution set? Once again, a set of closure equations of the form

$$F = T_{71}T_{67}T_{56}T_{45}T_{34}T_{23}T_{12} - I = 0$$
$$\Rightarrow F' = T_{34}T_{23}T_{12} - (T_{71}T_{67}T_{56}T_{45})^{-1} = 0$$

is constructed, and this time a single random complex coefficient intersecting hyperplane is added. Both the chair and boat conformations at some point in their motion satisfy the relationships

$$\phi_{12} = 0$$
$$\phi_{23} = -\phi_{71}$$
$$\phi_{34} = -\phi_{67}$$
$$\phi_{45} = -\phi_{56}$$

This point can be used as a seed to generate the rest of the intersection points on the irreducible component using monodromy. Specifically, for the boat conformation, the



initial angles are:

$$\begin{aligned}
\phi_{12} &= 0 \\
\phi_{23} &= \cos^{-1}\left(\frac{19\sqrt{2}-27}{16\sqrt{2}-22}\right) &&\approx 102^\circ \\
\phi_{34} &= -\cos^{-1}\left(\sqrt{\frac{1}{7}(9-4\sqrt{2})}\right) &&\approx -46^\circ \\
\phi_{45} &= -\cos^{-1}\left(1-\frac{1}{\sqrt{2}}\right) &&\approx -73^\circ
\end{aligned}$$

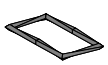
while for the chair, the initial angles are:

$$\begin{aligned}
\phi_{12} &= 0 \\
\phi_{23} &= \cos^{-1}\left(\frac{19\sqrt{2}-27}{16\sqrt{2}-22}\right) &&\approx 102^\circ \\
\phi_{34} &= -\cos^{-1}\left(-\sqrt{\frac{1}{7}(9-4\sqrt{2})}\right) &&\approx -134^\circ \\
\phi_{45} &= \cos^{-1}\left(1-\frac{1}{\sqrt{2}}\right) &&\approx 73^\circ
\end{aligned}$$

These symmetry points were found to generate witness sets with 112 elements each for both the boat and chair forms. Further analysis revealed that each of these sets mapped onto the other, and hence that both real conformations lie on the same irreducible component of degree 112. This indicates that **the two conformations are connected in complex space, but not in real**, since it is not possible to transform a real 7-loop from the boat to the chair form without disconnecting the links. The degree of this component is quite large, and indicates the high complexity of the motion of the 7-loop.

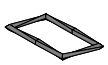
6.5 Conclusions

It was possible to use polynomial continuation (applied to the closure equations of the 6-link N-loop) to identify features of the closure equations which are responsible for the 6-loop's unusual mobility properties. The closure equations for the 6-link N-loop was found to have a dimension-1 solution set of degree 12, which is responsible for its mobility when connected in the mobile conformation. The closure equations were also found to have a dimension-zero solution set with 168 solutions, only two of which are real. These two real solutions represent the same rigid conformation up to rotational



symmetry.

Monodromy was used to identify a single (dimension-2) irreducible component of degree 48 which is responsible for the mobility of the 8-link N-loop. This process was also used to identify a single (dimension-1) irreducible component responsible for the 7-link N-loop's mobility in both the 'chair' and 'boat' forms.

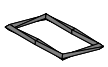


7. Conclusions

The usefulness and versatility of numerical continuation, and in particular polynomial continuation, in the design of deployable rings and frames has been illustrated by way of a number of case studies.

On a more fundamental level, the design of deployable rings has been approached from the viewpoint that an in-depth understanding of the closure/compatibility equations used to mathematically describe the behaviour of the underlying linkage is of critical importance. Often, more than one equation structure can be used to model linkage behaviour, with some some structures being more appropriate than others, depending on the exact case (compare the symmetry based compatibility equations used in Chapter 4 to the transfer matrix based loop closure equations used in Chapter 5 to model a quite similar style of ring). If one is to attempt the design of a linkage by direct solution of a system of defining equations, in particular using polynomial continuation, careful consideration must be given to the type of closure/compatibility equations used. For example, in Chapter 3, a transfer matrix based closure equation was found to be useful for understanding the mobility of the 6-link plane symmetric foldable ring, but a more simple compatibility equation based only on two of the ring's six hinge angles was found to be more appropriate for designing the ring using continuation because it was written in terms of fewer variables and could be used to define the ring's shape at a greater number of positions during deployment. Ideally, the closure/compatibility equations will be arranged in such a way that the mixed volume of the complete system of design equations is minimised (in Chapter 6, the total degree of the closure equations was reduced by splitting the transfer matrices across two terms). A fundamental understanding of the defining equations is also necessary when it comes to introducing design parameters in the form of constraint equations. One can optimise the continuation based design process by constructing constraint equations involving only those design variables which are absolutely necessary, and which can be arranged in such a way as to minimise the mixed volume of the design equations.

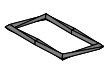
In addition to illustrating how polynomial continuation can be used in the design



process of deployable rings, it has also been shown that variations on the continuation method (in the form of cascades of homotopies, and monodromy processes) can be used to get a real understanding of the mathematical nature of a deployable linkage's mobility. In Chapter 6 it was shown that closure equations could be used (without the construction of a real 6-link N-loop) to reveal the existence of two different types of conformation; one mobile and one rigid. This method has the potential to reveal conformations which had not previously been considered in a linkage, and can also be used to determine with absolute certainty that no other conformations, besides those already known, exist.

Polynomial continuation has applications, not only in kinematics, but also in geometry, chemistry, and many other technical fields. Chapter 2 provides an introduction to polynomial equations, and how they arise in the study of geometry and kinematics. The types of solutions which arise are examined, and categorised. The concepts of start and target systems are introduced, and the path-following techniques used in numerical continuation are described in some detail. Two main methods of reducing the number of solution paths to follow are considered. The first is multi-homogenisation, and techniques for using this method are presented, along with an automated method of generating start systems. The second is the polyhedral homotopy method, and again, a practical guide to its use as a solver is given. Readers wishing to learn more about the mathematical basis of the polyhedral method will find Li (1999) and Huber & Sturmfels (1995) to be useful sources. The depth of content in Chapter 2 is thought to be sufficient for a basic understanding of the processes involved in using polynomial continuation in the design of deployable linkages.

Chapter 3 focuses on a type of plane symmetric 6-bar linkage which has been proposed as a reliable, single degree of freedom deployable frame/ring for space applications. The rectangular version of the linkage, with width twice its length, has three to four design parameters (depending on whether the new variable γ is introduced or not) pertaining to the exact orientation of the hinges joining each of the bars. The linkage is a Bricard plane symmetric case, and its Denavit-Hartenberg parameters can be written in terms of the four angular design parameters of the 6-bar foldable ring. The three design parameter version of the ring was subjected to a design process using polynomial continuation by way of a simple compatibility equation for the loop. Further work required on this topic includes; an extensive study of possible deployment mechanisms, focusing on reliability; an analysis of the stretch in a membrane attached to the frame as it deploys (could be a critical design factor if the attached membrane is particularly fragile); and an analysis of the stiffness and vibration properties of the frame. Stiffness and vibration properties are of particular importance in space applications.

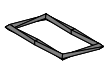


The subject of Chapter 4 is a particular type of deployable ring with N bars, and rotational symmetry of order N/2. An inventory of existing deployable rings with rotational symmetry is given, of which the type of ring first proposed in Crawford *et al.* (1975) (in particular the variant in which only one hinge is used to attach each bar to the next in a loop) is identified as the main focus for the rest of the chapter. The majority of the chapter is concerned with a continuation-based design process, much like that in Chapter 3. Again, a compatibility equation (making use of the symmetry of the ring to simplify expressions) is used to design the ring by means of a series of angular precision points, except that this time, the angles are global bar orientations rather than internal hinge angles. A greater variety of design options is also presented, with various combinations of the design variables used to illustrate the range of controls a designer can bring to bear on the problem. A method of simulation of the ring's motion based on the compatibility equations is also provided. Future work on this topic could include a scan of the feasible space of the design variables (as in Section 3.4), as well as an examination of deployment techniques. The multiple degrees of freedom present in rings of the type presented in this chapter make reliable deployment a more challenging problem.

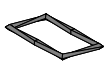
In Chapter 5, the focus is again on deployable rings, except that in this case consideration is given to rings whose deployed shape is not necessarily a regular polygon. For computational simplicity, the number of bars is restricted to eight. By way of contrast with the two preceding chapters, a system of transfer equations is used as the basis of the polynomial continuation design process. Symmetry is still used to limit the size of the expressions. Also different is the way in which constraints are placed on the system. Rather than a series of angular precision points, a series of distance-based constraints is used to specify the stowed shape of the ring precisely. This illustrates the usefulness of continuation in designing a deployable structure to meet strict practical requirements.

Finally, Chapter 6 presents a slightly less practical, but nonetheless interesting, application of polynomial continuation to the field of linkage design. Continuation can be used to enumerate the possible configurations for a linkage, once a connectivity has actually been specified. This analysis is performed on the 6-link variant of the N-loop. The equation structures of the 7 and 8-link versions of the N-loop are scanned for positive dimensional solution sets, and witness sets are generated for the irreducible components responsible for the loops' mobility.

The power of polynomial continuation as a design tool has been illustrated by way of several examples. Continuation allows a linkage (or deployable structure) designer to break the design process down into two key stages. First, he/she decides which features are essential for the linkage's purpose. These features can be represented as mathemati-



cal constraints, which are applied during the continuation process to yield a collection of design options, all of which meet these constraints. If the linkage closure/compatibility equations, and the constraint equations have been posed correctly, then the use of continuation guarantees that **every possible design option satisfying the imposed constraints will be found**. In the second stage of the process the designer can confidently exercise his/her judgement in deciding which of the feasible options best suits the application under consideration (see sections 5.1 and 5.2). Continuation reduces uncertainty, and allows a designer to be at once objective and subjective in deriving the best solution to the problem at hand.



Glossary

Bézout Number A measure of the number of solutions of a system of polynomial equations based on an n-Homogenised version of those equations (includes multiplicities). It is less than or equal to the total degree, but greater than or equal to the Mixed Volume. 30

Cascade of Homotopies A series of homotopies used in the process of finding Witness Set members at various dimensionalities. 49

Closure Equation An equation, or set of equations, which is used to ensure that a linkage is connected in a loop. 9

Coefficient Homotopy A form of Homotopy in which only the coefficients of a system of polynomial equations are functions of the Homotopy Parameter. 38

Compatibility Equation An equation, or set of equations, which is used to ensure that one or more parts of a linkage are connected in a way which is physically realisable and compatible. 9, 73

Conformational Space The space of feasible assemblies of a linkage. Often used to describe assemblies of molecules. 138

Constraint Equation An equation, or set of equations, which is used to enforce design constraints or realisable limits mathematically. 9

Convex Polygon A two-dimensional version of a convex polytope. 117

Convex Polytope Otherwise known as a convex polyhedron; it is an *n* dimensional shape whose vertices define a convex set of points. 34

Denavit-Hartenberg A minimal line (four parameter) representation in robotics first proposed by Denavit and Hartenberg. 16, 63

Deployable Structure A structure capable of a large change in size, usually with a smaller stowed configuration, and a larger deployed configuration. 1

Frame A linkage which has a rigid configuration which defines a particular shape. It is often used as a mount for membranes. 4

Geometric Degree of Freedom A mobility or degree of freedom not accounted for by the Kutzbach Criterion. It usually arises in a linkage as a result of some special symmetry, or conformation. 5, 154

Geometrically Isolated A solution to a system of equations which has no other solutions in the immediate vicinity. 19

Homogenised A homogeneous polynomial has monomials which all have the same total degree. A polynomial can be homogenised via the addition of new homogeneous variables. 20, 152

Homotopy For functions g and f: $X \to Y$, a homotopy is a "family of continuous functions $h(t) : X \to Y$ for $t \in [0, 1]$"[1] such that $h(0) = g$ and $h(1) = f$ and the map $t \mapsto h(t)$ is continuous from $[0, 1]$. 23, 152–154

Homotopy Parameter The homotopy parameter is the variable $t \mapsto h(t)$ which drives the Homotopy. 23, 152

Irreducible Component A connected continuum of solutions existing at a given dimension in solution space. 50, 68, 154

Lifting Function A function applied to each of the elements of the supports of a system of polynomial equations to *lift* the supports up a dimension. 37

Line Symmetric In two dimensions is usually used to refer to a reflection across a line (2D version of plane symmetry). In three dimensions can be used to refer to a rotational symmetry about a line, or axis. 62

Minkowski Sum The result of adding every element of one set to every element of another. 35

Mixed Supports A collection of Supports for a system of equations, each support being unique. 40, 79

[1] http://en.wikipedia.org/wiki/Homotopy

Mixed Volume The combined area/volume of all the level-r subfaces for a subdivision of a set of lifted supports. Practically, it provides an upper bound on the number of solutions to a given system of polynomial equations. 34, 68, 152, 154

Monodromy Literally means to "move around singly", and pertains to the movement of various objects around a singularity. The process is used here to check the membership of Irreducible Components. 50, 141

Monomial A product of powers of the variables of a system of equations. Sometimes also used to refer to the individual terms of a polynomials (products of powers of variables and their coefficients). 34

Numerical Continuation A mathematical technique for finding solutions of a system of equations. Solutions of a known system of equations are tracked numerically into those of the unknown system. 11, 18, 154, 155

Overconstrained Mechanism A subset of linkages with a Geometric Degree of Freedom. 5, 67

Polyhedral Homotopy A form of Homotopy which is based on a lifted support subdivision. It is used to find a full complement of solutions to a system of polynomial equations, with a number of start solutions equal to the Mixed Volume. 16

Polynomial Continuation The same as Numerical Continuation, applied specifically to polynomial systems. 11, 18

Positive Dimensional A positive dimensional solution set is a continuum of non-geometrically isolated solutions which exists on a line, or manifold in the solution space. 19, 20

Precision Point A point, either in physical space, or in the design parameter space of a linkage, which acts as a design target for the linkage. 8, 58

Ring A closed loop deployable linkage. 4

Semi-Mixed Supports A collection of Supports for a system of equations with some, but not all of the supports being unique. 40, 79

Singular Solution A solution of a system of equations which defines a location at which the system's Jacobian is indeterminate. 19

Solution at Infinity A solution of a non-homogeneous system of polynomial equations which does not exist in the finite plane. 19

Start System In Numerical Continuation, a start system forms the start point for a homotopy. It is usually a system of equations with a similar structure as the Target System, but with known solutions. 23

Support A representation (as a set of vectors) of the powers of each of the variables in each of the monomials in a polynomial equation. 34, 153, 154

Target System In Numerical Continuation, a target system forms the end point for a homotopy. It is usually a system of equations with unknown solutions. 23, 155

Univariate A univariate polynomial is written in terms of a single variable only. 31, 138

Witness Set A witness set is a collection of points on a positive dimensional solution set of a particular dimension. A full witness set contains as many elements as the degree of the positive dimensional solution set. 17, 49, 139, 152

Zero Dimensional Solution Another way of saying that a solution is geometrically isolated. 48

References

ABBOTT, T. & BARTON, R. (2004). Generalizations of Kempe's universality theorem. Manuscript. 8

ALLGOWER, E.L. & GEORG, K. (1980). Simplicial and continuation methods for approximating fixed points and solutions to systems of equations. *SIAM Review*, **22**, 28–85. 23

ALLGOWER, E.L. & GEORG, K. (2003). *Introduction to Numerical Continuation Methods*. Society for Industrial and Applied Mathematics (SIAM). 11

ANGELES, J., ALIVIZATOS, A. & AKHRAS, R. (1988). An unconstrained nonlinear least-square method of optimization of RRRR planar path generators. *Mechanism and Machine Theory*, **23**, 343–353. 9

BAKER, J. (1980). An analysis of the Bricard linkages. *Mechanism and Machine Theory*, **15**, 267–286. 62

BAKER, J.E. (2005). A curious new family of overconstrained six-bars. *ASME Journal of Mechanical Design*, **127**, 602–606. 62

BAKER, J.E. (2006). On generating a class of foldable six-bar spatial linkages. *ASME Journal of Mechanical Design*, **128**, 374–784. 62

BERNSTEIN, D.N. (1975). The number of roots of a system of equations. *Functional Analysis and its Applications*, **9**, 183–185. 34, 35, 37

BRAUN, R.D. & KROO, I.M. (1995). Development and application of the collaborative optimization architecture in a multidisciplinary design environment. Tech. rep., NASA Langley (NASA-95-iciam.rdb). 9

BRICARD, R. (1897). Mémoire sur la théorie de l'octaèdre articulé. *Journal de mathématiques pures et appliquées*, **3**, 113–148. 56

BRYANT, J. & SANGWIN, C. (2008). *How Round Is Your Circle?*. Princeton University Press. 7

CHABLAT, D. & ANGELES, J. (2003). The computation of all 4R serial spherical wrists with an isotropic architecture. *Journal of Mechanical Design*, **125**, 275–280. 12

CHEN, Y. (2003). *Design of Structural Mechanisms*. Ph.D. thesis, University of Oxford. 56, 62

CHEN, Y. & YOU, Z. (2005). Mobile assemblies based on the Bennett linkage. *Proceedings of The Royal Society London A*, **461**, 1229–1245. 62

CHEN, Y. & YOU, Z. (2006a). Square deployable frames for space applications. part 1: theory. *Journal of Aerospace Engineering*, **220**, 347–354. 6, 56

CHEN, Y. & YOU, Z. (2006b). Square deployable frames for space applications. part 2: realization. *Journal of Aerospace Engineering*, **221**, 37–45. 6, 56

CHEN, Y. & YOU, Z. (2007). Spatial 6R linkages based on the combination of two Goldberg 5R linkages. *Mechanism and Machine Theory*, **42**, 1484–1498. 62

CHEN, Y. & YOU, Z. (2008). An extended Myard linkage and its derived 6R linkages. *Journal of Mechanical Design*, **130**, 052301,1–052301,8. 62

CHEN, Y. & YOU, Z. (2009). Two-fold symmetrical 6R foldable frame and its bifurcations. *International Journal of Solids and Structures*, **46**, 4504–4514. 56, 57, 65, 67, 77, 136

CHEN, Y., YOU, Z. & TARNAI, T. (2004). Threefold-symmetric Bricard linkages for deployable structures. *International Journal of Solids and Structures*, **42**, 2287–2301. 5, 56

CHTCHERBA, A.D. & KAPUR, D. (2004). Constructing Sylvester-type resultant matrices using the Dixon formulation. *Journal of Symbolic Computation*, **38**, 777–814. 11

CRAWFORD, R.F., HEDGEPETH, J.M. & PREISWERK, P.R. (1974). Spoked wheels to deploy large surfaces in space: weight estimates for solar arrays. In *Canadian Aeronautics and Space Institute and American Institute of Aeronautics and Astronautics, Joint Meeting, Toronto, Canada; United States.*. 86, 90, 91

CRAWFORD, R.F., HEDGEPETH, J.M. & PREISWERK, P.R. (1975). Spoked wheels to deploy large surfaces in space-weight estimates for solar arrays. Tech. Rep. NASA-CR-2347; ARC-R-1004, NASA. 56, 86, 88, 90, 102, 112, 150

CUI, L. & DAI, J. (2011). Axis constraint analysis and its resultant 6R double-centered overconstrained mechanisms. *Journal of Mechanisms and Robotics*, **3**. 5

DE FOCATIIS, D.S.A. & GUEST, S.D. (2002). Depolyable membranes designed from folding tree leaves. *Philosophical Transactions of The Royal Society A*, **360**, 227–238. 2

DEMAINE, E.D. & O'ROURKE, J. (2007). *Geometric Folding Algorithms. Linkages, Origami, Polyhedra.*. Cambridge University Press. 7

DHINGRA, A.K., CHENG, J.C. & KOHLI, D. (1994). Synthesis of six-link, slider-crank and four-link mechanisms for function, path and motion generation using homotopy with m-homogenization. *Journal of Mechanical Design (ASME)*, **116**, 1122–1131. 30

DIAB, N. & SMAILI, A. (2008). Optimum exact/approximate point synthesis of planar mechanisms. *Mechanism and Machine Theory*, **43**, 1610–1624. 9

EMIRIS, I.Z. (1994). *Sparse Elimination and Applications in Kinematics*. Ph.D. thesis, University of California at Berkeley. 11, 12

EMIRIS, I.Z. & MOROZ, G. (2011). The assembly modes of rigid 11-bar linkages. In *International Federation for the Promotion of Mechanism and Machine Science*. 11, 13

EMIRIS, I.Z. & MOURRAIN, B. (1996). Polynomial system solving and the case of the six-atom molecule. Tech. Rep. 3075, INRIA. 11, 12

ERDOS, P. & GRUNWALD, T. (1939). On polynomials with only real roots. *The Annals of Mathematics*, **40**, 537–548. 12

FANG, H., LOU, M., HUANG, J., QUIJANO, U. & PELAEZ, G. (2004). Development of a 7-meter inflatable reflectarray antenna. In *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference, Palm Springs, CA., USA.*. 4

FREUDENSTEIN, F. & MAKI, E.R. (1979). The creation of mechanisms according to kinematic structure and function. *Environment and Planning B*, **6**, 375–391. 7, 62, 67

GAN, W.W. & PELLEGRINO, S. (2003). Closed-loop deployable structures. In *44th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*. 5, 56

GAN, W.W. & PELLEGRINO, S. (2005). Kinematic bifurcations of closed-loop deployable frames. In *Proceedings of the 5th International Conference on Computation of Shell and Spatial Structures*. 56, 77

GAN, W.W. & PELLEGRINO, S. (2006). Numerical approach to the kinematic analysis of deployable structures forming a closed loop. *Journal of Mechanical Engineering Science*, **220**, 1045–1056. 6, 56, 65, 66, 67, 77, 92, 113, 136

GANTES, C.J. (2001). *Deployable Structures: Analysis and Design*. Southampton : WIT Press. 1

GAO, T. & LI, T.Y. (2003). Mixed volume computation for semi-mixed systems. *Discrete and Computational Geometry*, **29**, 257–277. 68

GAO, T., LI, T.Y., VERSCHELDE, J. & WU, M. (2000). Balancing the lifting values to improve the numerical stability of polyhedral homotopy continuation methods. *Applied Mathematics and Computation*, **114**, 233–247. 47

GAO, X.S., ZHU, C.C., CHOU, S.C. & GE, J.X. (2001). Automated generation of Kempe linkages for algebraic curves and surfaces. *Mechanism and Machine Theory*, **36**, 1019–1033. 8

GE, J.X., CHOU, S.C. & GAO, X.S. (1999). Geometric constraint satisfaction using optimization methods. *Computer-Aided Design*, **31**, 867–879. 9

GOSSELIN, C. & ANGELES, J. (1990). Singularity analysis of closed kinematic chains. *IEEE Transactions on Robotics and Automation*, **6**, 281–290. 72

GUAN, Y. & VERSCHELDE, J. (2008). *Software for Algebraic Geometry*, vol. 148. Springer New York. 18

GUEST, S.D. (1994). *Deployable Structures: Concepts and Analysis*. Ph.D. thesis, University of Cambridge. 1

GUEST, S.D. & FOWLER, P.W. (2010). Mobility of 'N-Loops': bodies cyclically connected by intersecting revolute hinges. *Proceedings of The Royal Society A*, **466**, 63–77. 136, 139, 144, 145

HARRIS-CORPORATION (1986). Development of the 15 meter diameter hoop column antenna. Tech. rep., NASA Langley Research Center. 87, 88

HARTENBERG, R. & DENAVIT, J. (1964). *Kinematic Synthesis of Linkages*. McGraw-Hill Book Company. 7

HENDERSON, M.E. (2004). Multiparameter continuation methods. http://www.research.ibm.com/people/h/henderson/Continuation/ContinuationMethods.html. 23

HOFFMANN, C. & VERMEER, P.J. (1995). Geometric constraint solving in **R**2 and **R**3. In *World Scientific, Singapore*. 11

HOPCROFT, J., JOSEPH, D. & WHITESIDES, S. (1984). Movement problems for 2-dimensional linkages. *SIAM Journal on Computing*, **13**, 610–629. 8

HRONES, J.A. & NELSON, G.L. (1951). *Analysis of the four-bar linkage; its application to the synthesis of mechanisms*. The Technology Press, The Massachusetts Institute of Technology. 7

HUBER, B. & STURMFELS, B. (1995). A polyhedral method for solving sparse polynomial systems. *Mathematics of Computing*, **64**, 1541–1555. 37, 149

HUTT, T. (2007). Deployable rings. Tech. rep., University of Cambridge. 56

JENSEN, F. & PELLEGRINO, S. (2002). Expandable structures formed by hinged plates. In *Fifth International Conference on Space Structures*. 2

JENSEN, F. & PELLEGRINO, S. (2005). Expandable "blob" structures. *Journal of the International Association for Shell and Spatial Structures*, **46**, 151–158. 2

KEMPE, A.B. (1877). How to draw a straight line; a lecture on linkages. MacMillan and Co. (Now available online through Project Gutenberg). 8

KONAXIS, C. (2010). *Algebraic Algorithms for Polynomial System Solving and Applications*. Ph.D. thesis, National and Kapodistrian University of Athens. 35

KRISHNAMURTY, S. & TURCIC, D.A. (1992). Optimal synthesis of mechanisms using nonlinear goal programming techniques. *Mechanism and Machine Theory*, **27**, 599–612. 9

KUMAR, P. & PELLEGRINO, S. (2000). Computation of kinematic paths and bifurcation points. *International Journal of Solids and Structures*, **37**, 7003–7027. 77

LALOI, N. (1999). Analytical and experimental study of a new type of hinges. Tech. rep., Deployable Structures Laboratory, Dept. of Engineering, University of Cambridge. 132

LAMURE, H. & MICHELUCCI, D. (1996). Solving geometric constraints by homotopy. *IEEE Transactions on Visualization and Computer Graphics*, **2**, 28–34. 19

LEE, H.Y. & LIANG, C.G. (1988). Displacement analysis of the general spatial 7-link 7R mechanism. *Mechanism and Machine Theory*, **23**, 219–226. 11

LEE, T.L. & LI, T.Y. (2007). Mixed volume computation, a revisit. 35

LEE, T.L., LI, T.Y. & TSAI, C.H. (2008). HOM4PS-2.0: a software package for solving polynomial systems by the polyhedral homotopy continuation method. *Computing*, **83**, 109–133. 18

LENGYEL, A. & YOU, Z. (2004). Bifurcations of SDOF mechanisms using catastrophe theory. *International Journal of Solids and Structures*, **41**, 559–568. 77

LEYKIN, A., VERSCHELDE, J. & ZHAO, A. (2000). Evaluation of Jacobian matrices for Newton's method with deflation to approximate isolated singular solutions of polynomial systems. 20

LI, T.Y. (1999). Solving polynomial systems by polyhedral homotopies. *Taiwanese Journal of Mathematics*, **3**, 251–279. 34, 37, 149

LI, T.Y. (2003). *Handbook of Numerical Analysis*, vol. XI, chap. Numerical Solution of Polynomial Systems by Homotopy Continuation Methods, 209–304. North-Holland. 34, 37, 40, 79

LUO, Z. & DAI, J. (2007). Patterned bootstrap: A new method that gives efficiency for some precision position synthesis problems. *Journal of Mechanical Design*, **129**, 173–183. 12

MANOCHA, D. & CANNY, J.F. (1994). Efficient inverse kinematics for general 6R manipulators. *IEEE Transactions on Robotics and Automation*, **10**, 648–657. 13

MAO, D., LUO, Y. & YOU, Z. (2009). Planar closed loop double chain linkages. *Mechanism and Machine Theory*, **44**, 850–859. 3

MARTÍNEZ-ALFARO, H. (2007). *Advances in Metaheuristics for Hard Optimization*, chap. Four-Bar Mechanism Synthesis for *n* Desired Path Points Using Simulated Annealing, 23–37. Springer Berlin Heidelberg. 9

MARTINS, J.R.R.A., STURDZA, P. & ALONSO, J.J. (2003). The complex-step derivative approximation. *ACM Transactions on Mathematical Software*, **29**, 245–262. 48

MAVROIDIS, C. & ROTH, B. (1994). Analysis and synthesis of overconstrained mechanisms. In *Proceeding of the 1994 ASME Design Technical Conference, Minneapolis, MI, September*, 115–133. 5

MIZUTANI, T., TAKEDA, A. & KOJIMA, M. (2007). Dynamic enumeration of all mixed cells. *Discrete and Computational Geometry*, **37**, 351–367. 47

MORGAN, A. (1987). *Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems*. Prentice-Hall, inc. 23

MORGAN, A. & WAMPLER, C. (1990). Solving a planar four-bar design problem using continuation. *Journal of Mechanical Design*, **112**, 544–550. 13

MORGAN, A.P. & SOMMESE, A.J. (1989). Coefficient-parameter polynomial continuation. *Applied Mathematics and Computation*, **29**, 123–160. 34

NIELSEN, J. (1997). *Solving Sets of Nonlinear Equations for the Design and Analysis of Mechanical Systems*. Ph.D. thesis, Dept. Of Mechanical Engineering, Leland Stanford Junior University. 11

OWEN, J.C. (1991). Algebraic solution for geometry from dimensional constraints. In *ACM Symposium on Solid and Physical Modeling*. 11

PELLEGRINO, S. (1995). Large retractable appendages in spacecraft. *Journal of Spacecraft and Rockets*, **32**, 1006–1014. 3

PELLEGRINO, S., ed. (2001). *Deployable Structures*. Springer-Verlag Wien New York. 1

PELLEGRINO, S. & YOU, Z. (1993). Foldable ring structures. In G. Parke & C. Howard, eds., *Space Structures 4: The Fourth International Conference on Space Structures*, vol. 1. 3

PELLEGRINO, S., GREEN, C., GUEST, S.D. & WATT, A. (2000). SAR advanced deployable structure. Tech. rep., University of Cambridge Department of Engineering. 6, 56, 132

PHILLIPS, J. (1984). *Freedom in Machinery*, vol. 1. Cambridge University Press. 62

PHILLIPS, J. (1990). *Freedom in Machinery*, vol. 2. Cambridge University Press. 62

PORTA, J., ROS, L., THOMAS, F., CORCHO, F., CANTÓ, J. & PÉREZ, J. (2007). Complete maps of molecular-loop conformational spaces. *Journal of Computational Chemistry*, **28**, 2170–2189. 136

RAGHAVAN, M. & ROTH, B. (1993). Inverse kinematics of the general 6R manipulator and related linkages. *Journal of Mechanical Design*, **115**, 502–508. 12, 13

READ, R.C. & WILSON, R.J. (1998). *An Atlas of Graphs*. Oxford University Press. 7

RIDOUT, M.S. (2009). Statistical applications of the complex-step method of numerical differentiation. *The American Statistician*, **63**, 66–74. 48

SANCIBRIAN, R., VIADERO, F., GARCÍA, P. & FERNÁNDEZ, A. (2004). Gradient-based optimization of path synthesis problems in planar mechanisms. *Mechanism and Machine Theory*, **39**, 839–856. 9

SEFFEN, K.A. & PELLEGRINO, S. (1999). Deployment dynamics of tape springs. *Proceedings of The Royal Society London A*, **455**, 1003–1048. 4

SOESENO, J. (1990). *Applications of Homotopy to Kinematic Synthesis and Analysis*. Master's thesis, University of Wisconsin-Milwaukee. 12

SOMMESE, A.J. & VERSCHELDE, J. (2000). Numerical homotopies to compute generic points on positive dimensional algebraic sets. *Journal of Complexity*, **16**, 572–602. 49, 56, 67, 68

SOMMESE, A.J. & WAMPLER, C.W. (2005). *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*. World Scientific. 31, 35, 50

SOMMESE, A.J., VERSCHELDE, J. & WAMPLER, C.W. (2001a). Numerical decomposition of the solution sets of polynomial systems into irreducible components. *SIAM Journal of Numerical Analysis*, **38**, 2022–2046. 51, 52, 67

SOMMESE, A.J., VERSCHELDE, J. & WAMPLER, C.W. (2001b). Using monodromy to decompose solution sets of polynomial systems into irreducible components. In C. Ciliberto, F. Hirzebruch, R. Miranda & M. Teicher, eds., *Application of Algebraic Geometry to Coding Theory, Physics and Computation*, 297–315, NATO, Kluwer Academic Publishers. 52

SOMMESE, A.J., VERSCHELDE, J. & WAMPLER, C.W. (2002a). Advances in polynomial continuation for solving problems in kinematics. In *Proceedings of DETC'02, ASME Design Engineering Technical Conferences*. 12, 13

SOMMESE, A.J., VERSCHELDE, J. & WAMPLER, C.W. (2002b). *Algebraic Geometry*, chap. A method for tracking singular paths with application to the numerical irreducible decomposition, 329–345. de Gruyter. 52

STURMFELS, B. & ZELEVINSKY, A. (1994). Multigraded resultants of Sylvester type. *Journal of Algebra*, **163**, 115–127. 11

SUBBIAN, T. (1990). *Use of Continuation Methods for Kinematic Synthesis and Analysis*. Ph.D. thesis, Iowa State University, Ames, IA. 13

SUBBIAN, T. & FLUGRAD, D.R. (1991). Four-bar path generation synthesis by a continuation method. *Journal of Mechanical Design*, **113**, 63–69. 13

TIBERT, G. (2002). *Deployable Tensegrity Structures for Space Applications*. Ph.D. thesis, Royal Institute of Technology, Department of Mechanics, Stockholm. 85, 87

VERSCHELDE, J. (1999). Algorithm 795: PHCpack: a general-purpose solver for polynomial systems by homotopy continuation. *ACM Transactions on Mathematical Software (TOMS)*, **25**, 251–276. 18

VERSCHELDE, J. (2009). Witness sets. Lecture, UIC Dept. of Math, Stat and CS. 48, 49

VERSCHELDE, J. (2011). Database of polynomial systems. `http://www.math.uic.edu/~jan/`. 12

VERSCHELDE, J. & GATERMANN, K. (1995). Symmetric Newton polytopes for solving sparse polynomial systems. *Advances in Applied Mathematics*, **16**, 95–127. 37

VERSCHELDE, J., GATERMANN, K. & COOLS, R. (1996). Mixed volume computation by dynamic lifting applied to polynomial system solving. *Discrete and Computational Geometry*, **16**, 69–112. 47

WAMPLER, C.W. & MORGAN, A.P. (1993). Solving the kinematics of general 6R manipulators using polynomial continuation. *Institute of Mathematics and its Applications Conference Series*, **41**, 57. 13

WAMPLER, C.W., MORGAN, A.P. & SOMMESE, A.J. (1990). Numerical continuation methods for solving polynomial systems arising in kinematics. *Journal of Mechanical Design*, **112**, 59–68. 12, 13, 23, 30, 32

WAMPLER, C.W., MORGAN, A.P. & SOMMESE, A.J. (1992). Complete solution of the nine-point path synthesis problem for four-bar linkages. *Journal of Mechanical Design*, **114**, 153–159. 13, 14

WATSON, L.T., BILLUPS, S.C. & MORGAN, A.P. (1987). Algorithm 652: HOMPACK: a suite of codes for globally convergent homotopy algorithms. *ACM Transactions on Mathematical Software (TOMS)*, **13**, 281–310. 18

YOU, Z. (2007). Motion structures extend their reach. *Materials Today*, **10**, 52–57. 5

YOU, Z. & PELLEGRINO, S. (1997a). Cable-stiffened pantographic deployable structures part 2: Mesh reflector. *AIAA Journal*, **35**, 1348–1355. 3

YOU, Z. & PELLEGRINO, S. (1997b). Foldable bar structures. *International Journal of Solids and Structures*, **34**, 1825–1847. 3

ZULEHNER, W. (1988). A simple homotopy method for determining all isolated solutions to polynomial systems. *Mathematics of Computation*, **50**, 167–177. 11, 23

A. Feasibility Charts for Plane-Symmetric 6-Bar Linkage

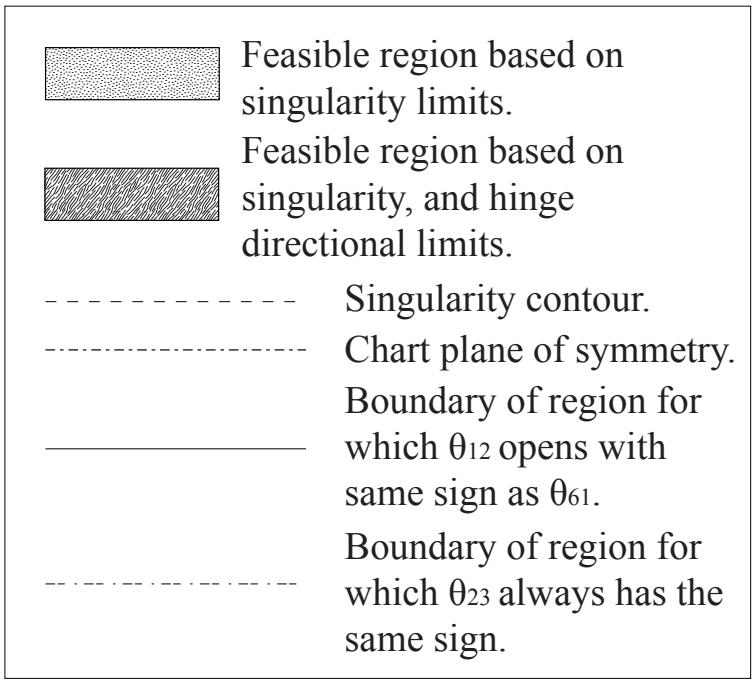


Figure A.1: Key to feasibility charts for plane-symmetric 6-bar linkage

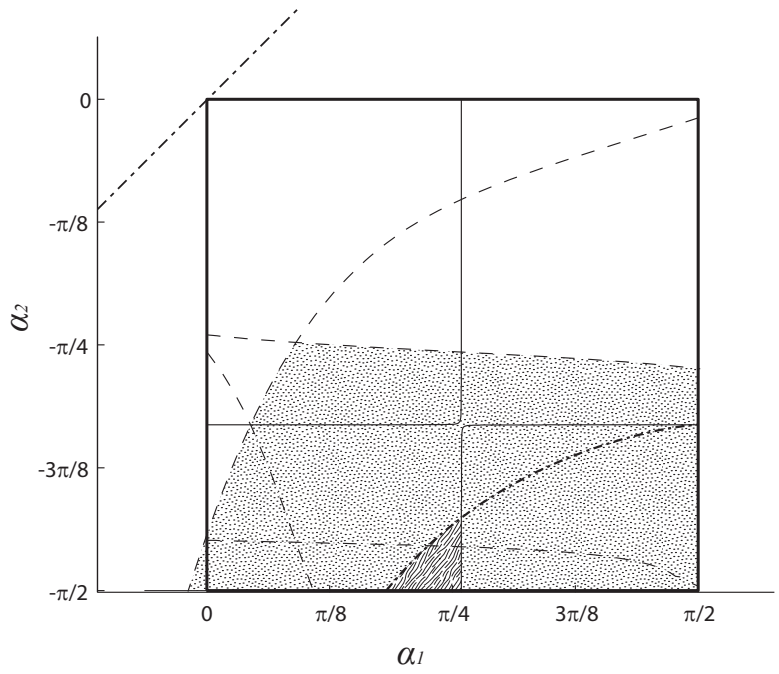


Figure A.2: Feasibility chart for $\gamma = \pi/8$

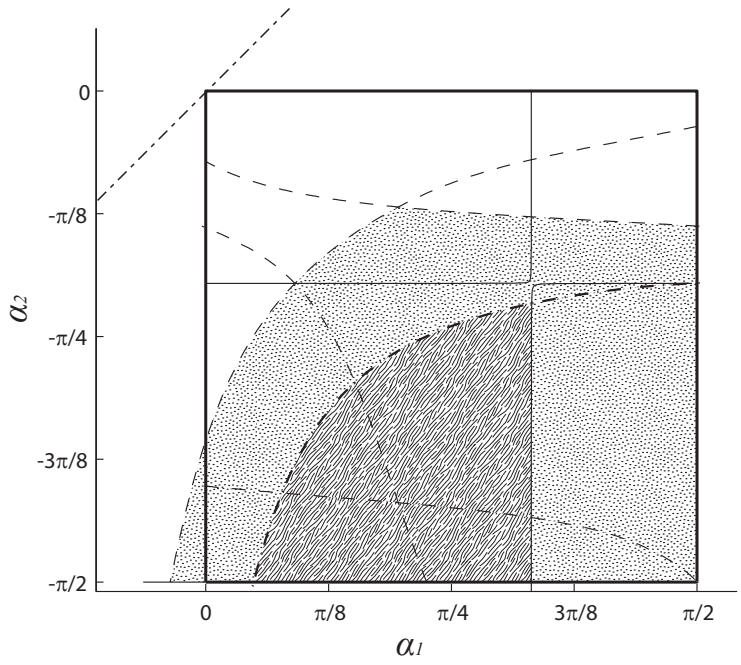


Figure A.3: Feasibility chart for $\gamma = \pi/4$

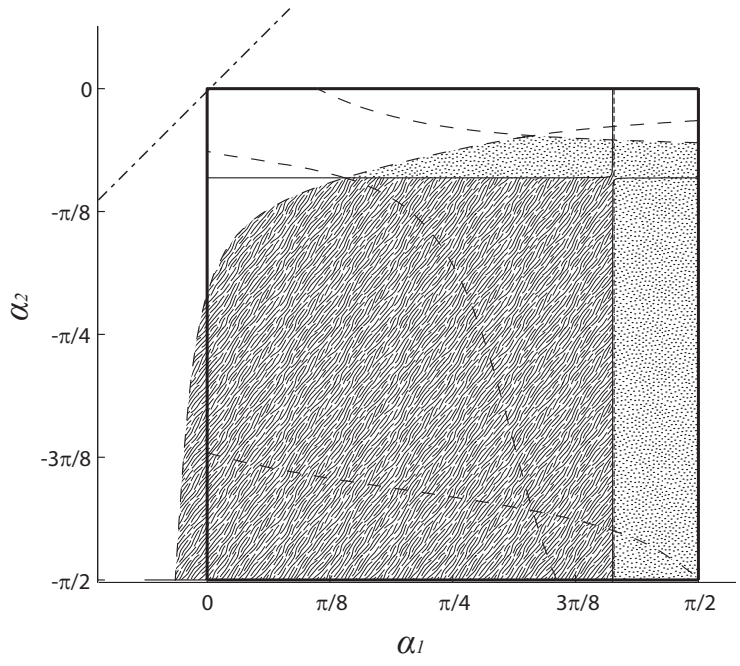


Figure A.4: Feasibility chart for $\gamma = 3\pi/8$

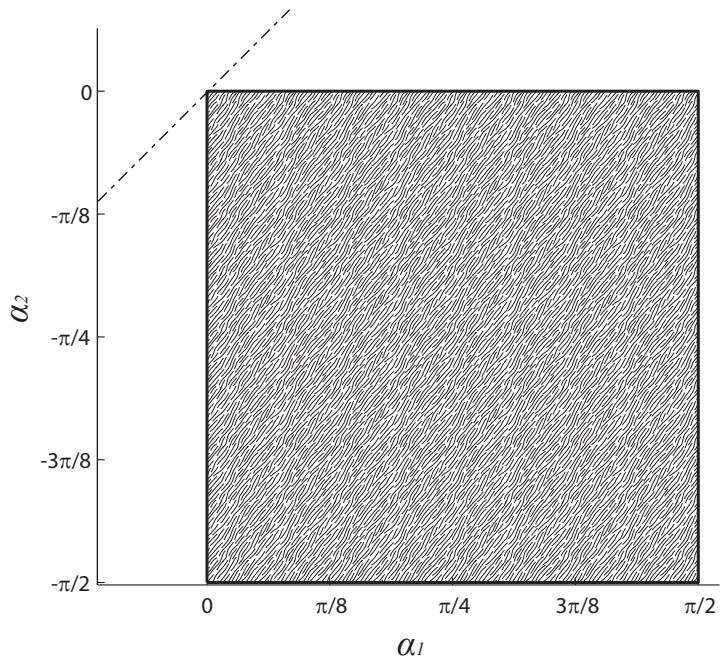


Figure A.5: Feasibility chart for $\gamma = \pi/2$

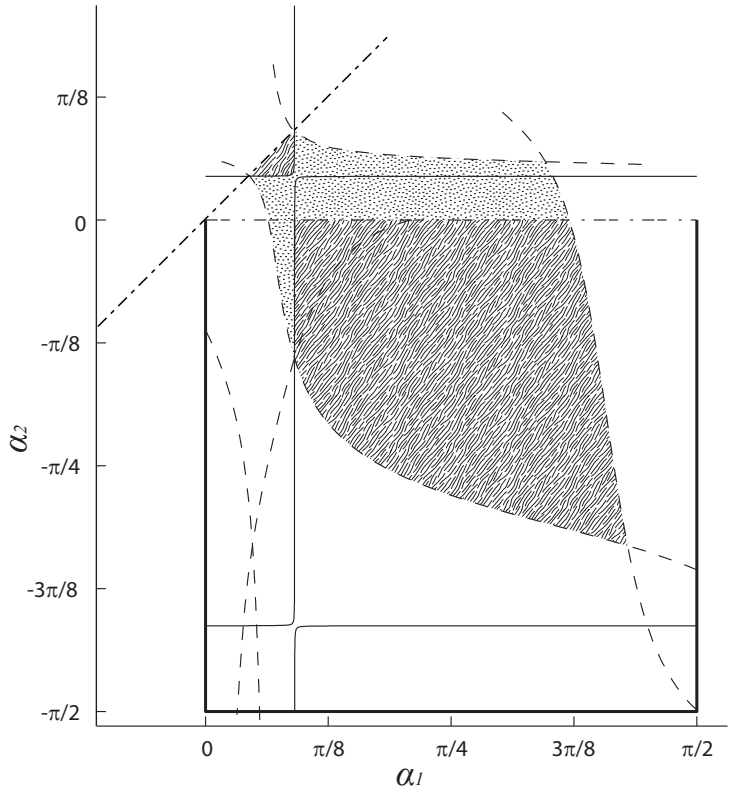


Figure A.6: Feasibility chart for $\gamma = 5\pi/8$

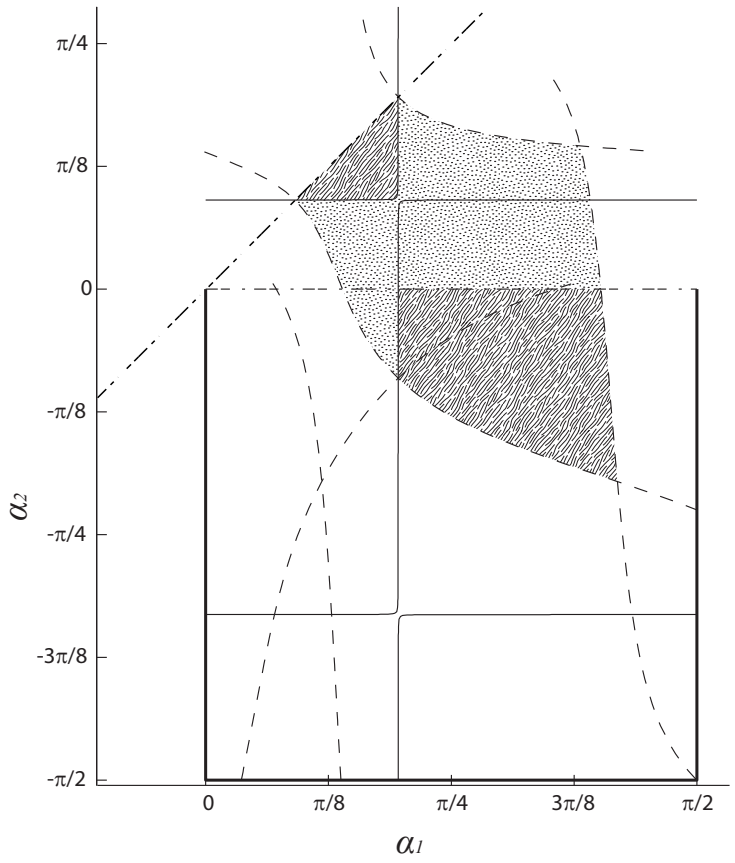


Figure A.7: Feasibility chart for $\gamma = 3\pi/4$

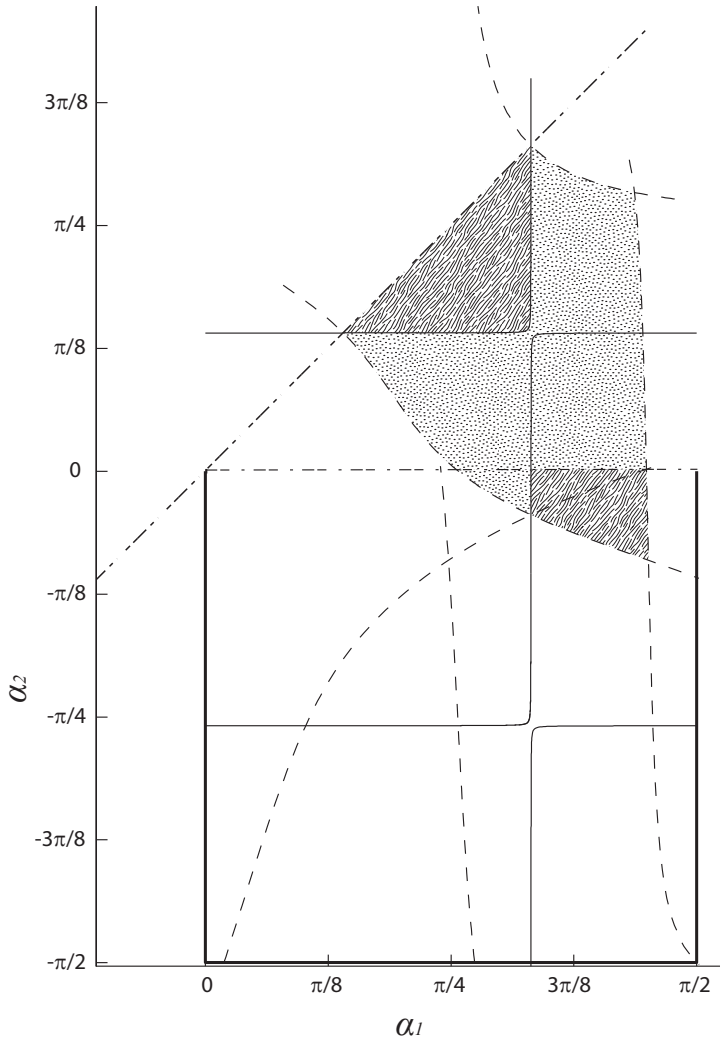


Figure A.8: Feasibility chart for $\gamma = 7\pi/8$

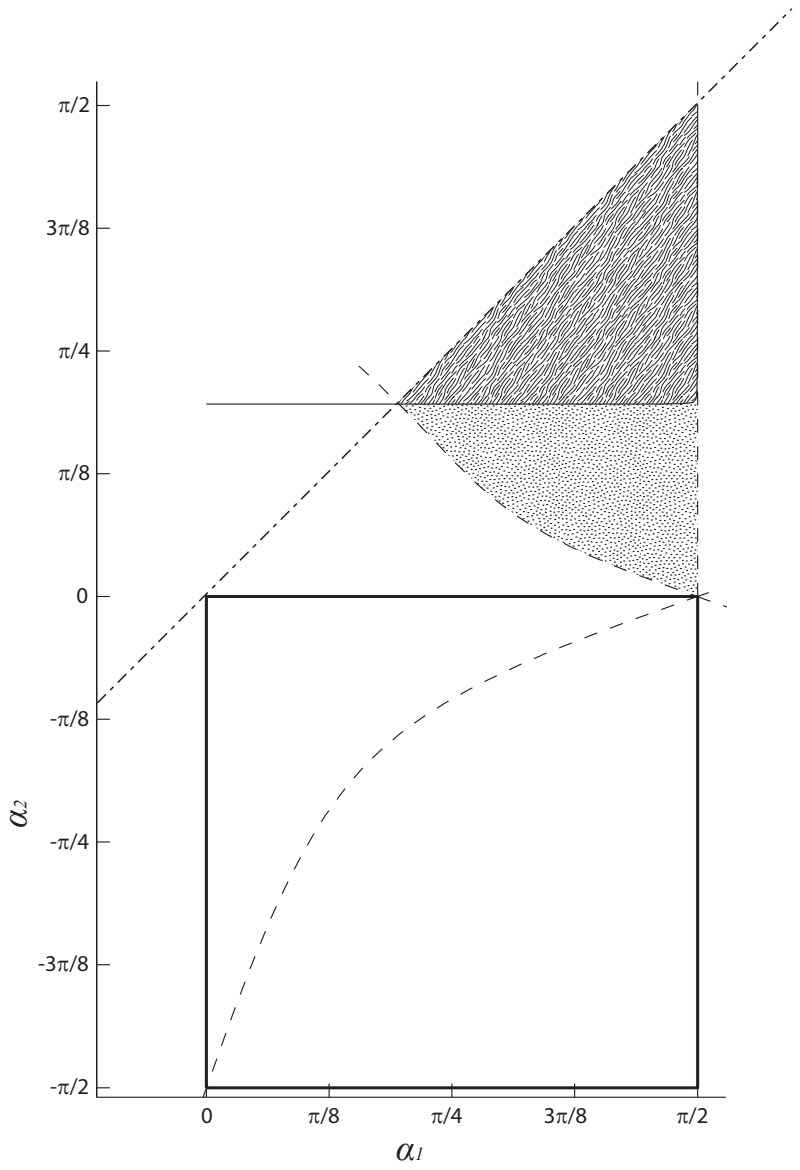


Figure A.9: Feasibility chart for $\gamma = \pi$

B. Software Guide

The polyhedral homotopy software was written entirely in Matlab. Matlab provides a flexible programming environment, and an extensive library of functions.

B.1 Using Software to Find Dimension-Zero Solutions of a System of Polynomial Equations

The first step in using any of the software utilised in solving nearly all of the problems included in this document is to express the terms of the equations in a compact, and immediately useful notation. As an example, consider equation system 2.2. This system consists of three equations in three unknowns, and is therefore the kind of square system one would expect to have a number of dimension-zero, geometrically isolated solutions. The support and coefficient list for the system must be expressed in matrix format, which in this case looks like:

$$\text{i)} \quad \text{Support:} \begin{bmatrix} 0 & 0 & 0 \\ 2 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{Coefficients:} \begin{bmatrix} 0 \\ 1 \\ 2 \\ -5 \end{bmatrix}$$

$$\text{ii)} \quad \text{Support:} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{Coefficients:} \begin{bmatrix} -2 \\ 3 \\ 1 \end{bmatrix}$$

$$\text{iii)} \quad \text{Support:} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{Coefficients:} \begin{bmatrix} 0 \\ 2 \\ -1 \\ 1 \end{bmatrix}$$

where i, ii and iii represent each of the three equations. Each row represents a term of the equation, and each column represents a variable. Note that a constant term (all zero row) has been included in each support. The support arrays are arranged into a structure called `S` (with `.support` extension), while the target system coefficients are arranged into a structure called `TargStruc` (with `.coeff` extension). These two structures are the only inputs required to fully solve the target system of polynomial equations.

Quite often, target systems contain hundreds of monomials, and the supports and target coefficients can only be reliably organised into arrays via an automated process.

The Mathematica commands Coefficient and CoefficientList were found to be quite useful in identifying the monomials of target systems, and extracting the coefficients themselves.

The following is a step-by-step practical guide to the solving of a square system of polynomial equations.

1. Determine the supports for each equation, and the coefficients of the target system. This can be done manually, or by some automated process. The results must be exported to the Matlab file `SupportConstruction.m`.

   - In the case of Equation 2.2, the first support would be entered as:
     `>> S(1).support = [0,0,0;2,1,1;0,2,0;0,0,1]`
     while the target coefficients for the first equation are entered as:
     `>> TargStruc(1).coeff = [0;1;2;-5]`
     and similarly for the subsequent equations.
   - Semi-mixed supports are handled by including the multiplicity of each support type in the `SuppRep` vector. For example, a system of five equations in which the first two supports are identical would be entered as:
     `>> SuppRep = [2;1;1;1]`
   - Running `SupportConstruction.m` loads `S`, `TargStruc` and `SuppRep` to the workspace.
   - `TargStruc` must be saved as a data file, for example:
     `>> save TargProblemEg1 TargStruc`

2. Lifting values must now be applied to the supports, and a valid subdivision determined. This is performed by running the function:
   `>> [SUpdated,KSub,yMat,maxheight,minheight] = ...`
   `PolyStartSimplexMixed(S,SuppRep)`

   - `maxheight` gives the largest power of t to appear in Equation 2.28, while `minheight` gives the smallest. Both should lie in a range of approximately 0.35 - 8 for numerical stability.
   - If the t-powers lie outside the desirable range, the heights can be shifted as a group via a simple multiplication of all the lifting values, or the spread of values can be decreased by running:
     `>> [SUpdated,KSub] = LiftingOpt(SUpdated,KSub,SuppRep)`
     as many times as necessary (which operates on the lifting values).
   - Following an optimisation run, the values of `yMat` need to be recalculated using:
     `>> yMat = yMatCalc(SUpdated,SuppRep,KSub)`
   - Finally, the data structures must be saved in a file, for example:
     `>> save StartHomotProblemEg1 SUpdated SuppRep KSub yMat`

3. The homotopy of Equation 2.28 must now be carried out, and this is done by running the function:

```
>> [XFinalMat,xmatbigMat,tmatMat,FailStats, ...
ReplaceSolLocations] = PolyDriverExacGrad(runsubs,OldFailStats)
```

- Before running this function, `PolyDriverExactGrad.m` must be edited to ensure that it loads the data for the correct start system. In the example under consideration, `StartHomotProblemEg1.mat` should be loaded.
- The input `runsubs` tells the solver which of the subfaces listed in `KSub` to run. To generate a complete start system, all subfaces must be run.
- The output, `XFinalMat`, contains a list of solutions to a start system which can be used to find the full set of solutions to any target system with the same polynomial structure. All its solutions should be finite, and the number of solutions will be equal to the mixed volume of the system (because its coefficients were randomly chosen complex numbers).
- Before using the start system whose solutions are listed in `XFinalMat`, the matrix should be renamed:
  `>> xStart = XFinalMat`
  and saved in a new data file:
  `>> save PolyStartProblemEg1 xStart`

4. To solve the target problem, run the function:
   ```
   >> [xmatout,xmatoutact,xmatbig,Jacobvec,tmat, ...
   FailStats] = CoeffDriverExactGrad(runsols)
   ```
   - Before running this function, `CoeffDriverExactGrad.m` must be edited to ensure that it loads the data for the correct start, and target systems. In the example under consideration, `StartHomotProblemEg1.mat` should be loaded for the start system information, `PolyStartProblemEg1.mat` for the start system solutions and `TargProblemEg1.mat` for the target coefficients.
   - The input `runsols` tells the solver which of the start solutions to pass to the path follower. Generally, one wishes to run all the start solutions.
   - The output, `xmatout`, contains the solutions to the target system. Paths which were found to diverge, or fail in some other way appear simply as a column of `NaN`s.

Main Data Structures

Structure/Function	Sub-Elements		Purpose
`S`	`.support`	→	Support in array form for each equation
`SUpdated`	`.support`	→	Support in array form for each equation
	`.k`	→	Number of terms in support
	`.w`	→	Lifting value for each term
	`.coeff`	→	Random complex coefficients
`TargStruc`	`.coeff`	→	Coefficients for target system
`SuppRep`	-	→	Vector containing equation multiplicities
`KSub`	`.subfaces`	→	Support pairs for each subface
	`.alpha`	→	α vector for each subface
	`.betavec`	→	β values for each subface
`yMat`	`.startsols`	→	Solutions to binomial start system
`xStart`	-	→	All solutions to homotopy in Equation 2.28
`xmatout`	-	→	All solutions to target system

Polyhedral Homotopy Function List

Function	Action	Inputs	Outputs

`SupportConstruction.m`	A script into which information about the target polynomial's support and coefficients can be entered.	-	`S`
			`TargStruc`
			`SuppRep`

`PolyStartSimplexMixed.m`	Function assigns random lifting to each support member, and computes a valid subdivision. The α and β vectors of Section 2.4 are computed, as well as the initial solutions based on each of the support base pairs.	`S`
		`SuppRep`
			`SUpdated`
			`KSub`
			`yMat`

`LiftingOpt.m`	Modifies `SupRepp` and `KSub` by implementing the method of Section 2.4.3 to increase the hight of the smallest t-power, and thus increase the numerical reliability of the following stages.	`SUpdated`
		`KSub`
		`SuppRep`
			`SUpdated`
			`KSub`

`yMatCalc.m`	Called by `PolyStartSimplexMixed.m` to compute the solutions to the t-independent support pairs. Needs to be called again following a lifting value modification, usually after an application of `LiftingOpt.m`.	`SUpdated`
		`KSub`
		`SuppRep`
			`yMat`

`PolyDriverExacGrad.m`	Implements the homotopy of Equation 2.28.	`SUpdated`
		`KSub`
		`yMat`
		`SuppRep`
			`xStart`

`StandardPFPolyhedralExact.m`	Is called directly by `PolyDriverExacGrad.m`, and performs the base level path following for each of the initial solutions in `yMat`.	`SUpdated`
		`KSub`
		`yMat`
		`SuppRep`
			`xStart`

Coefficient Homotopy Function List

Function	Action	Inputs	Outputs

`CoeffDriverExactGrad.m`	Performs a coefficient homotopy, converting the random complex coefficient start system (whose solutions were computed by `PolyDriverExacGrad.m`) into the target system with the same polynomial structure, and coefficients specified in `SupportConstruction.m`.	`SUpdated`
		`SuppRep`
		`xStart`
		`TargStruc`
			`xmatout`

`StandardPFCoeffExact.m`	Standard path following function called directly by `CoeffDriverExactGrad.m`.	`SUpdated`
		`SuppRep`
		`xStart`
		`TargStruc`
			`xmatout`

B.2 Using Software to Find Higher Dimensional Solutions of a System of Polynomial Equations

The suite of functions written to examine the structure of systems of polynomial equations makes use of many of the same components as the functions in Section B.1. However, they were written to be used in a slightly more automated way, with the user specifying the names of some data files in the first instance (within the function `SuppComp_Step_1.m`), and then simply running a series of other function in a specified order.

The following is a step-by-step practical guide to the solving of a square system of polynomial equations, which one suspects of having positive dimensional solution sets.

1. Construct the target system function, and name it `TargFun.m`. This will look like:
   `>> F = TargFun(x,ExtParams)`
   - The input should be a column vector containing the n variables;
   - The output should be a column vector containing the evaluated polynomials.
2. The function `SuppComp_Step_1.m` loads the target system information into the workspace. Run this as:
   `>> [S,SuppRep,HyperCoeff0,FileNames] = SuppComp_Step_1(d)`
   - First, manually enter the support data into the function's script. This is done in the same way as the dimension-zero case.
   - No target system coefficients need be specified, as an external target function is used.
   - All the data files used in the following process must be named in this function file. The names are stored in a structure called `FileNames`, passed to all subsequent functions.
   - The vector `SuppRep` must be a column of ones. No shorthand for mixed supports is permitted in the witness set construction process.
   - The only explicit input to `SuppComp_Step_1.m` is the parameter d, which specifies how many new variables/complex hyperplanes should be appended to the target system. This is the top dimension to be searched. Setting $d = 0$ reduces the problem to a search for zero dimensional solutions, as in Section B.1.
3. Next, run the subdivision finder. This looks like:
   `>> [SUpdated,KSub,yMat,maxheight,minheight] = ...`
   `PolyStartSimplexMixed_Step_2(S,SuppRep,FileNames)`
   This behaves in just the same way as its $0D$ counterpart, except that the start system data files are automatically saved using the names specified in `SuppComp_Step_1.m`.
4. The lifting value optimiser:
   `>> [SUpdated,KSub] = LiftingOpt_Step_3(SUpdated,KSub,SuppRep)`
   also behaves much like its $0D$ counterpart.

- Once a lifting value optimisation has been performed, the solutions in `yMat` will have to be recalculated by running:
  `>> yMat = yMatCalc(S,SuppRep,KSub)`
  then re-saving the start polynomial data using:
  `>> save(FileNames.StartSys,'SUpdated','SuppRep','KSub', ... 'yMat')`

5. The top level augmented start system can be completely solved by running:
   `>> [XFinalMat,xmatbigMat,tmatMat,FailStats, ... ReplaceSolLocations] = DriverPolyhedral_Step_4(runsubs, ... FileNames,OldFailStats)`

   - Once again, `runsubs` specifies the subfaces to be run.
   - `XFinalMat` contains the solutions to the augmented (top level) start system. The solutions are automatically saved in a data file for use in solving the target system later.

6. To solve the top level augmented target system, run:
   `>> [xmatout,xmatoutact,xmatbig,Jacobvec,tmat, ... FailStats] = DriverCoeff_Step_5(runsols,FileNames)`

   - The output, `xmatout`, contains the solutions to the top level target system (augmented with d random complex hyperplanes and d new variables, z. This output is automatically saved using the name specified in `SuppComp_Step_1.m`.
   - Solutions which form part of the witness set for dimension d can already be recognised by their z variables being zero.

7. Next, run:
   `>> [nonsols,highersols,Wout] = WitFilt_Step_ia(xmatout,[], ... FileNames)`

   - This function automatically checks for solutions of dimension d, and places them in the structure `Wout(1).set`.
   - Any other finite solutions are returned in the matrix `nonsols`.

8. The next step is to use a cascade process to reduce the dimension of the target system. This is performed by running:
   `>> [xmatout,xmatoutact,xmatbig,Jacobvec,tmat, ... FailStats] = DriverDimReducer_Step_ib(nonsols,FileNames)`

   - The points in `nonsols` are fed into a path follower which reduces the target system to a dimension of $d - 1$.
   - The output, which over-writes the previous `xmatout`, contains the solutions to this dimension $d - 1$ system.
   - Elements of the witness set for $d - 1$ are identifiable by having zero z components.

9. The function `WitFilt_Step_ia.m` is run again, except this time it acts as a filter stage, searching for solutions to the $d-1$ dimension equations which are actually members of the dimension d witness set:
   `>> [nonsols,highersols,Wout] = WitFilt_Step_ia(xmatout, ... Wout,FileNames)`

   - The structure `Wout` is used to provide the witness set points to perform the filter step.
   - The new-found elements of the witness set of dimension $d-1$ are appended to the witness set structure as `Wout(2).set`.

10. Functions `WitFilt_Step_ia.m` and `DriverDimReducer_Step_ib` are run alternatively until dimension zero is reached. `Wout` will contain $d+1$ sub-elements at the end.

Main Data Structures

Structure/Function	Sub-Elements		Purpose
`S`	`.support`	→	Support in array form for each equation
`FileNames`	`.RandCoeffs`	→	Name of file containing random hyperplane coefficients
	`.StartSys`	→	Name of file containing `SUpdated`, `KSub` and `SuppRep`
	`.PolySys`	→	Name of file containing `xStart`
	`.Output`	→	Name of file containing `xmatout`
`SuppRep`	-	→	Vector containing equation multiplicities
`SUpdated`	`.support`	→	Support in array form for each equation
	`.k`	→	Number of terms in support
	`.w`	→	Lifting value for each term
	`.coeff`	→	Random complex coefficients
`KSub`	`.subfaces`	→	Support pairs for each subface
	`.alpha`	→	α vector for each subface
	`.betavec`	→	β values for each subface
`yMat`	`.startsols`	→	Solutions to binomial start system
`xStart`	-	→	All solutions to homotopy in Equation 2.28
`xmatout`	-	→	All solutions to initial augmented target system, or the result of a dimension reducing cascade step
`nonsols`	-	→	Those results of a homotopy run which are determined not to lie on the dimension currently being examined ($z \neq 0$). These results are passed on to the cascade step
`Wout`	`.set`	→	Structure built up over cascade process containing all solutions found to exist at each dimension examined

Witness Set Construction Function List

Function	Action	Inputs	Outputs
SuppComp_Step_1.m	The supports for the target equations are entered directly into the file, as well as the multiplicity for each support. The user specifies a dimension (d) at which to begin the search for witness points, and the function appends an appropriate number of intersecting complex hyperplanes and new variables (as per Equation 2.36).	d	S SuppRep
PolyStartSimplexMixed_Step_2.m	Function assigns random lifting to each support member, and computes a valid subdivision. The α and β vectors of Section 2.4 are computed, as well as the initial solutions based on each of the support base pairs.	S SuppRep FileNames	SUpdated KSub yMat
LiftingOpt_Step_3.m	Modifies SupRepp and KSub by implementing the method of Section 2.4.3 to increase the hight of the smallest t-power, and thus increase the numerical reliability of the following stages.	SUpdated KSub SuppRep	SUpdated KSub
yMatCalc.m	Called by PolyStartSimplexMixed_Step_2.m to compute the solutions to the t-independent support pairs. Needs to be called again following a lifting value modification, usually after an application of LiftingOpt_Step_3.m.	SUpdated KSub SuppRep	yMat
DriverPolyhedral_Step_4.m	Implements the homotopy of Equation 2.28.	SUpdated KSub yMat SuppRep FileNames	xStart
StandardPFPolyhedralExact.m	Is called directly by DriverPolyhedral_Step_4.m and performs the base level path following for each of the initial solutions in yMat.	SUpdated KSub yMat SuppRep	xStart

...Continued on next page

– continued from previous page

Function	Action	Inputs	Outputs
DriverCoeff_Step_5.m	Solves the homotopy system of Equation 2.36.	SUpdated SuppRep xStart FileNames	xmatout
StandardPFCoeff.m	Standard path following function called directly by DriverCoeff_Step_5.m.	SUpdated SuppRep xStart	xmatout
TargFun.m	A function called directly by StandardPFCoeff.m. It contains the 'black-box' version of the original target equations. The user must manually insert the target system into this function. Unlike the dimension-zero process, gradients are taken by a finite difference method.	-	-
WitFilt_Step_ia.m	If this is the top dimension run, the function simply identifies solutions with $z = 0$, places them in Wout and puts the rest in nonsols. On subsequent calls, the function checks each solution for membership of higher dimensional solution sets using existing members of Wout, and removes them if necessary.	xmatout Wout FileNames	nonsols Wout
DriverDimReducer_Step_ib-.m	Implements the homotopy of Equation 2.37 to reduce the dimension by one. This function is called in a successive iteration process with WitFilt_Step_ia.m until dimension zero is reached.	nonsols FileNames	xmatout
DriverCoeff_Monodromy.m	Takes a particular dimension's witness set (or a component thereof) and maps it onto a new hyperplane, before mapping back to the original hyperplane. The output can be checked for permutations (which indicate membership of the same irreducible component).	Wout.set FileNames	xmatout