A virtual reality interface for the design of compliant mechanisms

Utkarsh Seth
Iowa State University

A virtual reality interface for the design of compliant mechanisms

by

Utkarsh Seth

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Human Computer Interaction

Program of Study Committee:
Judy M. Vance, Major Professor
James H. Oliver
Sunghyun Kang

Iowa State University

Ames, Iowa

2009

Table of Contents

List of figures

List of tables

Acknowledgements

Upon completing my graduate coursework, I strongly believe that I have grown academically, professionally, and personally from the support of several important individuals. First and foremost, I would like to thank Dr. Judy Vance for providing me with valuable experiences while under her tutelage. I have enjoyed all aspects of working with her and respect her drive and motivation. She has shown an immense amount of patience during my development, and I am forever indebted. I would like to express gratitude to Dr. Jim Oliver and Professor Sung for their willingness to participate on my committee and their insights concerning the research.

I would like to thank National Science Foundation for funding the graduate research. I am grateful for my family and friends for all of their love and support. They have provided me with the motivation to excel and work hard at everything in life. I am thankful to my parents, my brother and Jyoti for their assistance and dedication. I have the utmost respect for each one and wish them the very best in their future endeavors. I appreciate the entire staff at the Virtual Reality Applications Center for providing a supportive environment.

Abstract

The objective of this research is to develop an immersive interface and a design algorithm to facilitate the synthesis of compliant mechanisms from a user-centered design perspective. Compliant mechanisms are mechanical devices which produce motion or force through deflection or flexibility of their parts. Using the constraint-based method of design, the design process relies on the designer to identify the appropriate constraint sets to achieve the desired motion. Currently this ability requires considerable prior knowledge of how non-linear flexible members produce motion. As a result, the design process is based primarily on the designer's previous experience and intuition.

The contribution of this research is the creation of a user-centered methodology towards the design of compliant mechanisms where the interface guides the designer throughout the design process. This research combines a mathematical representation of the constraint-based compliant mechanism design process with an immersive interface to support active user interaction in the design process. A virtual reality (VR) immersive interface lets the user interact with the problem at hand in a natural way with hand gestures, head motion, etc. This enables the designer to input the intended motion path by simply grabbing and moving the object and letting the system decide which constraint spaces apply. The user-centered paradigm supports an approach that focuses on the designer defining the desired motion, the system generating the constraint sets, and the designer deciding which constraints to apply to complete the design. With this approach, the system produces a set of possible solutions and the designer completes the design process.

This research results in an intelligent design framework that will allow a broader group of engineers to design complex compliant mechanisms, giving them new options to draw upon when searching for design solutions to critical problems.

Chapter 1: Introduction

This chapter provides an overview of the objective and the impact of the current research. It also gives a brief introduction on compliant mechanisms which allows the reader to develop an understanding of the topic at hand.

1.1 Objective

The objective of this research is to develop an immersive interface and a design algorithm to facilitate the synthesis of compliant mechanisms from a user-centered perspective. Currently, it is difficult to design these linkages because the non-linear motion of the components is easy to determine intuitively in the early design process; therefore, successful designs are achieved by only by highly skilled and experienced designers. A three-dimensional (3D) immersive intelligent interface is proposed which provides an abstract layer between the designer and the design process enabling the designer the ability to input the intended motion path, examine potential solutions and select candidate configurations for evaluation.. A user-centered methodology towards the design of compliant mechanism is proposed which lets novice designers design a variety of compliant mechanisms where the computer takes care of the mathematics involved in the background. The approach coupled with VR technology lets the user interact with the problem at hand in a natural way with hand gestures, head motion, etc.

1.2 Compliant mechanisms

A mechanism is a mechanical device which is used to transfer or transform motion, energy or force. A rigid body mechanism consists of joints and rigid links. It is through these joints that the force or motion is transferred. The mechanism shown in Figure 1 consists of four rigid links joined with pin joints or revolute joints that result in the transfer of motion and energy.

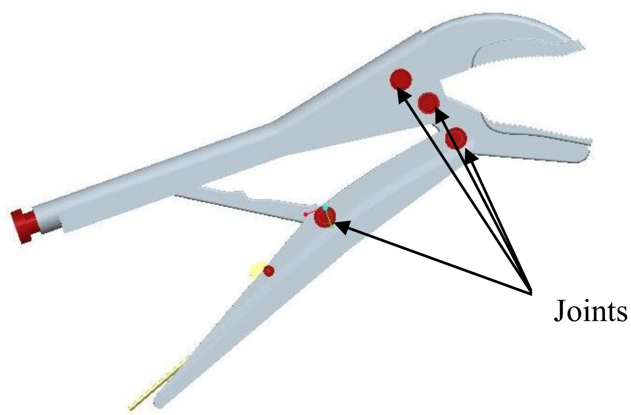


Figure 1: A rigid link mechanism

Compliant mechanisms [1], on the other hand, are mechanical devices which produce motion or force through deflection or flexibility of their parts instead of rigid links and joints. Figure 2 shows a gripper that is a compliant mechanism. While most of the gripper is made from rigid components, the very thin member which is close to the gripper end provides the compliance and deflects to produce the gripper motion when the handles of the gripper are squeezed closed. Often compliant mechanisms achieve compliance through the careful placement of these thin, compliant, geometric elements in the final design. It is the placement and configuration of the rigid members with the compliant members to achieve a desired motion and force transfer that is the challenge in designing compliant mechanisms.

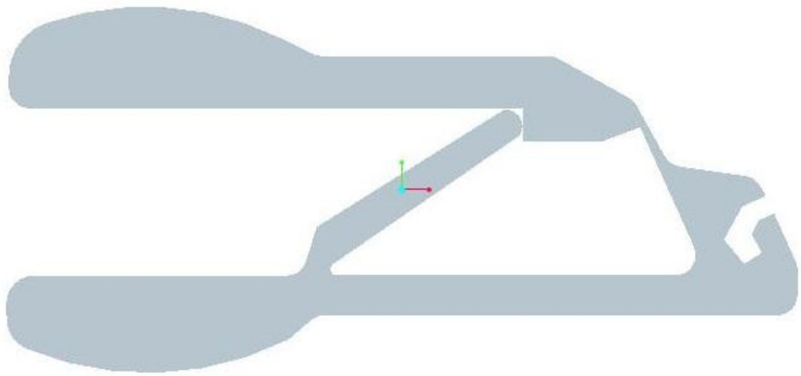


Figure 2: Compliant mechanism

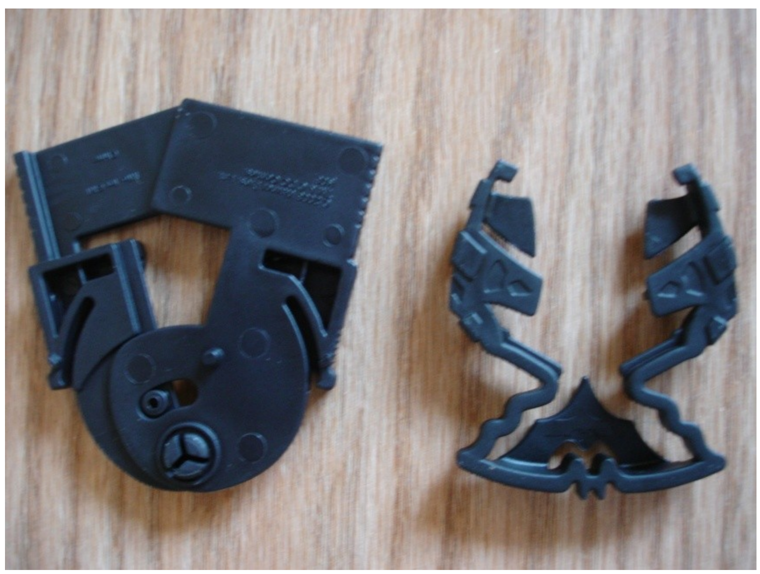


Figure 3: Rigid link (left) mechanism and compliant mechanism (right)

1.2.1 Advantages and disadvantages of compliant mechanisms

There are several advantages to using compliant mechanisms:

1. The number of parts in a compliant mechanism is largely reduced as compared to a traditional rigid body mechanism. This simplifies manufacturing and reduces assembly costs.

2. Compliant mechanisms can retain their precision for longer periods of use because there is little wear at the joints, compared to their rigid body counterparts. This characteristic makes them attractive solutions for applications in precision machinery where very accurate positioning is required.

3. The weight of the overall mechanism is most often reduced. This aspect makes them attractive solutions in applications weight reduction is of primary importance such as airplanes.

4. These mechanisms can be miniaturized easily because they are often manufactured in their final form with little or no assembly required. They are often used as actuators for other micro-mechanisms.

Compliant mechanisms store energy as strain energy much the same way that springs store energy. This method allows for controlled release of this energy to produce actuation of components when needed. One of the disadvantages of compliant mechanisms is that they are more subject to material fatigue failure since the motion and force transfer comes from deflection of the material. In addition, it is difficult to design compliant mechanisms to transmit large forces. While it is possible to get large force magnification from compliant mechanisms, the absolute forces transmitted are limited compared to traditional rigid link mechanisms. Another disadvantage is that repair of compliant mechanisms results in complete replacement where in rigid body mechanisms, a single link or joint can be repaired or replaced without replacing the entire mechanism.

1.2.2 Challenges in compliant mechanisms

The challenge in designing compliant mechanisms is in designing the linkage geometry to deflect under the applied load to provide the motion and force transfer. Large deflections introduce geometric nonlinearities into the analysis of the motion of compliant mechanisms, requiring non-linear modeling and analysis methods. An understanding of mechanism synthesis theory and non-linear mathematics is required to achieve a suitable design. Due to this, the design of such mechanisms has followed a trial and error approach in most cases. The research results presented in this thesis support the design of a user interface to aid in the design of compliant mechanisms.

1.3 Thesis overview

Chapter 2 discusses the state-of-the-art in compliant mechanism design and virtual reality. **Chapter 3** focuses on the methodology adopted to solve the problem at hand from a user-centered perspective. It highlights how the approach focuses on the mechanism designer by freeing him/her from the burden of complex mathematical calculations and from the need of hands on experience with compliant mechanism design. It also introduces a proposed step by step design process. **Chapter 4** presents a case study and shows how the method could be adopted to solve problems. **Chapter 5** summarizes the research and gives a road map for future work which needs to be done to expand the project scope.

Chapter 2: Background

2.1 Compliant mechanism design methods

In the mechanism design field much research has been done on applying computational techniques for the synthesis of compliant mechanisms to achieve a pre-decided user defined motion. The most often used approaches in the area are the pseudo rigid body model approach [2] and topological synthesis [3-8]. In the pseudo rigid body model a rigid body analysis method is used in the analysis of compliant mechanisms. This approach models a compliant mechanism as a rigid body which allows the rigid body theories and methodologies to be applied on compliant mechanisms [9-10]. As the approach is not directly related to the design of compliant mechanisms, validation and verification of the result becomes important.

The topological synthesis method relies on optimization methods to arrive at an optimum structural topology to achieve specified motion requirements. It models the mechanisms as a series of several link members of different sizes which together perform the desired motion. In both of the above methods prior experience and mechanism design knowledge is needed for successful completion of the design.

The third approach, on which this research is based, is the constraint-based design approach. This approach was introduced by Maxwell [11]. It is based upon the concept that the motion of a given body is controlled by the position and orientation of constraints applied to it at any given instance. This approach comes in useful as it helps the designer in visualizing motions and ultimately designing the desired mechanism. Recently this approach was revisited by Blanding [12] and by Hale, Awtar and Slocum at MIT [13 – 14]. The constraint-based design approach lies on the basic premise of constraints and freedom.

2.2 Maxwell's theory

The contribution of James Clerk Maxwell [15] forms of the basis of the constraint-based approach. He proposed a simple mathematical formula which stated that a non-redundant constraint when applied to a body removes one degree of freedom from it. A constraint restricts the motion of a body in a particular direction. The equation could be written as (Eqn.1) where C is the number of constraints and DOF is the degree of freedom

$$\mathbf{6 - C = DOF} \tag{1}$$

Equation (1) states that as a designer applies non-redundant constraints on a body, those many degrees of freedom are removed. For a free body in space, there are six degrees of freedom; i.e. three translations and three rotations. This is represented in Figure 4 where the straight arrows represent translation and the twisted ones represent rotation.

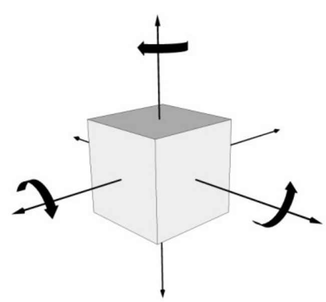


Figure 4: Object's 6 DOF in space

2.3 Blanding's theory

Douglas E. Blanding's research [16] is also very essential in the theory of constraints and how the object behaves when a constraint is applied. He presented theories which explained the use of constraints, exact constraints, over-constraints, and the theory of instant centers. All of these theories feed in to the basic concept of the constraint-based approach. We will discuss some of these theories in this section.

2.3.1 Constraint

A body is said to be constrained when a physical connection is made between the object and a reference body such that the number of degrees of freedom of the object gets reduced. A constraint, applied to a body, has infinite compliance perpendicular to the constraint's line of action and infinite stiffness along its line of action. Figure 5 illustrates the concept of a single constraint.

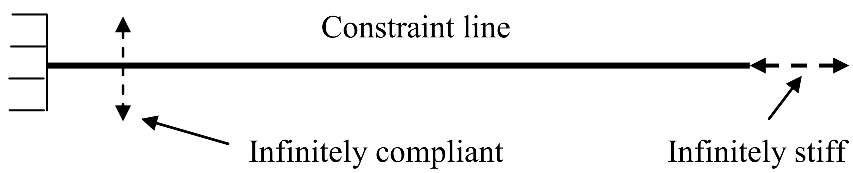


Figure 5: Constraint

The number of reduced degrees of freedom is defined as the degree-of-constraint (doc) of the physical connection. A constraint that eliminates the translation along a line is called a translational constraint. A constraint that eliminates the rotation about a line is called a rotational constraint. A constraint, when applied to a body, is functionally equivalent to any other constraint acting upon the same body along the same constraint line.

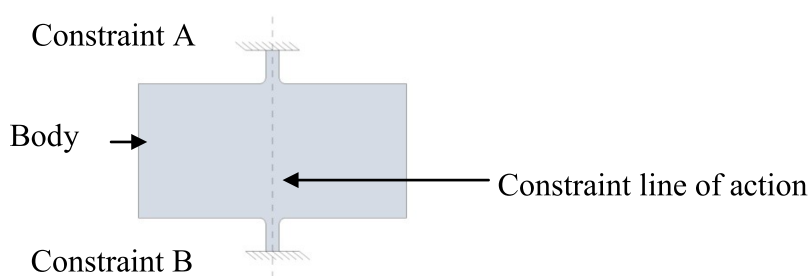


Figure 6: Functionally equivalent constraints

Figure 6 shows two constraints, Constraint A and Constraint B, which are functionally equivalent as they act along the same constraint line. In this arrangement, one of the constraints is redundant. This situation is referred to as being *over-constrained*. A body is said to be *over-constrained* when two or more constraints acting on the body control the same degree of freedom.

2.3.2 Instant center of rotation

Using the theory of instant centers Blanding stated that rotation can be achieved by placing two non-collinear constraints on the body. The body will rotate about an axis which is perpendicular to the plane containing the two constraints and at the location of the intersection of the two constraints. This is known as the "instant" center (Figure 7) and all points in the body will rotate about this center when examined during a very short "instantaneous" time.

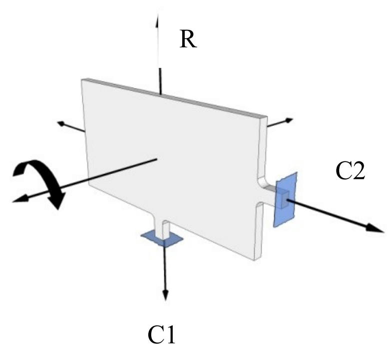


Figure 7: Instant center of rotation

Blanding also observed that a translation could be represented as a rotation about a point located at infinity as shown in Figure 8. He stated that pure translation of a body in a plane could be represented as rotations about axes located at infinity with respect to the body.

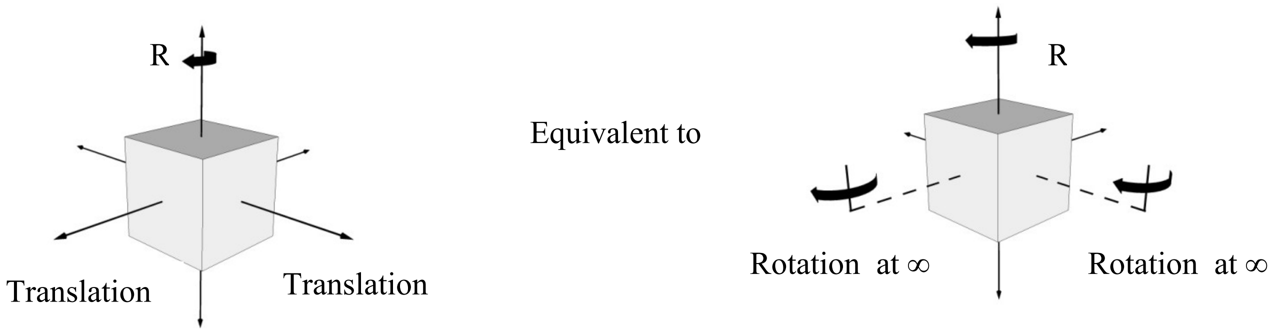


Figure 8: Translation equals rotation at infinity

Blanding proposed a "Rule of complimentary patterns" which defined a relationship between an object's degree of freedom and the constraints applied to it. It states:

> When a pattern of constraint lines is applied to two bodies, there is a resultant and complementary pattern of freedom lines which exist between them. Every freedom line intersects every constraint line. Given one or the other of these patterns containing *n* lines without redundancy, the complementary pattern will contain *6-n* lines.

Blanding gives an example of the above rule as shown in Figure 9. The figure shows a pattern of four constraints (C1 – C4) applied on a body. The rule of complementary patterns explains that four constraints reduce the number of degrees of freedom by four leaving only two remaining degrees of freedom. The particular orientation and location of the constraints as applied to the body in Figure 9 result in allowing free movement of only two independent rotations. The two rotations will lie on the plane of C1, C2 and C3 and will intersect C4. It is to be noted that the two rotational degrees of freedom as drawn are not unique and could be selected from any two independent lines which lie in the plane of the top surface of the object and intersect at the indicated corner. Any pair of intersecting rotational degrees of freedom is equivalent to any other pair intersecting at the same point and lying in the same plane.

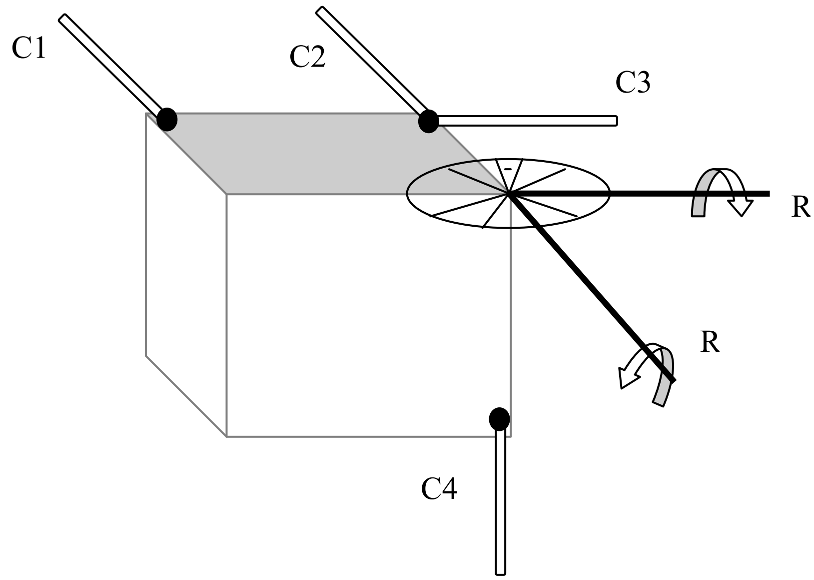


Figure 9: Disk of radial lines

Blanding also asserted that two parallel lines in space intersect at infinity. Figure 10 shows two lines with constant distance (d) between them. It is shown that they would intersect at a point in infinity.

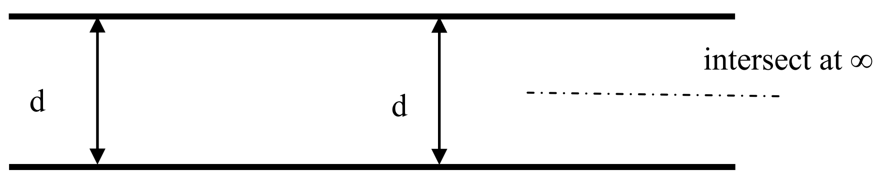


Figure 10: Parallel lines intersect at infinity

With the help of the above theory, Blanding gave another example for the rule of complementary patterns. In Figure 11, the object is constrained by four constraints oriented in a different pattern. All the four Cs are parallel to each other. Once again, the number of degrees of freedom has been reduced by four constraints resulting in only two degrees of freedom for the object. In this configuration, the two degrees of freedom are again two rotations but they consist of parallel lines, all of which lie in the same plane. The two lines of rotation intersect C4 and are parallel to the other Cs. Applying the rule of complimentary patterns, it can be seen that two rotations are not unique and could consist of any two independent lines from the infinite plane of parallel lines.

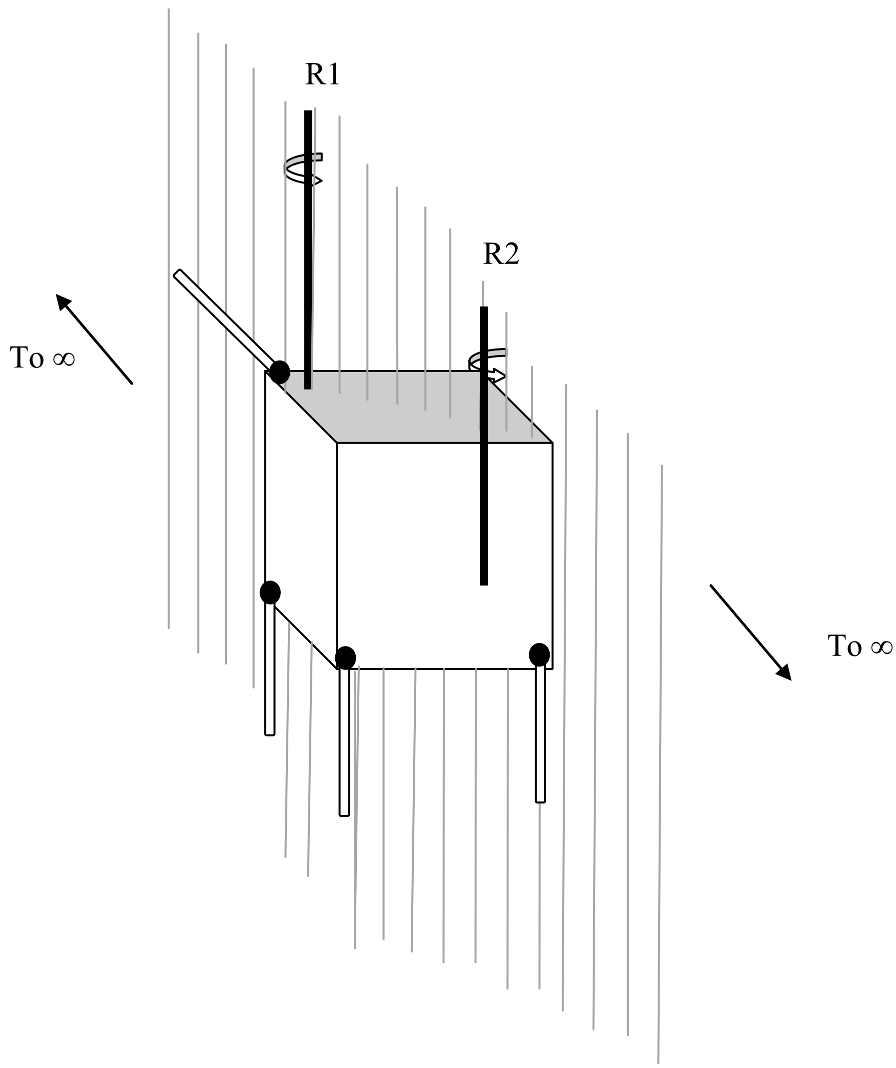


Figure 11: Plane of parallel lines

2.4 Freedom space and constraint space

The freedom space or freedom topology represents the allowable motion in space. The constraint space represents the restricted motions in space. Hopkins [17] extended Blanding's theory to come up with a series of geometric representations of possible freedom and constraint spaces. The design method based on these geometric representations is known as

FACT (Freedom and Constraint-based Topologies). This section reviews that work and its significance.

FACT deals with the different constraint and freedom spaces by dividing them into different CASEs and TYPEs. The CASE in the FACT method defines the number of constraints applied on the body. For e.g. CASE # 1 denotes one constraint which means if the body is classified as a CASE 1, it has five degrees of freedom as only one constraint is applied. If the user has defined two independent motions, the CASE number (according to Maxwell's theory) becomes four. The TYPEs within a CASE outline all possible ways to combine and orient constraint lines to produce the desired degrees of freedom. For every constraint space produced, there is a specific freedom space associated with it.

2.4.1 CASE 1 TYPE 1

As there are six degrees of freedom for a body in space, there were six CASEs defined. The TYPEs within each case encompass all of the possible ways each CASE can be implemented. For example CASE 1 (where one constraint could be applied on a body) had just one TYPE, as there is only one way in which one degree of freedom of a body could be blocked. A single constraint line applied to a body in space would produce the exact same freedom space as any other constraint applied to the body anywhere else. The freedom space produced might be different for differently oriented constraint lines but the basic motion that placing one constraint on a body would produce would be the same. The example is illustrated in Figure 12.

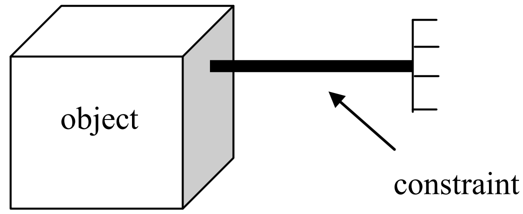


Figure 12: Object with one constraint

Applying one constraint in a body results in five degrees of freedom.

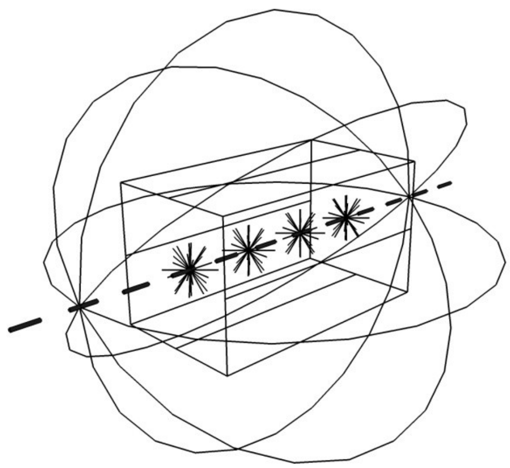


Figure 13: Freedom space for CASE 1 TYPE 1

The freedom space illustrated in Figure 13 consists of three independent freedom configurations which combine to form a single freedom space for CASE 1 TYPE 1. The innermost geometric representations shown in Figure 13 are the spherical sets, each of which consists of an infinite set of infinitely long lines, all intersecting at the same point. The center point of each of these spherical sets lie on the single constraint line as defined in CASE 1. The next set of freedom lines consists of an infinite set of infinitely long lines all parallel to the single constraint. We have drawn this set of four perpendicular planes forming a square enclosure that is aligned with the single constraint line. The final set of freedom lines consists of a set of infinite hoops which intersect the single constraint line at infinity. We have drawn these as a set of co-axial circles which intersect each other at points on the constraint line. In addition to this geometric representation of the freedom space, in this particular CASE, the space also consists of screws with non-zero finite pitch values. These screw representations will be explained later in the thesis.

Hopkins and Culpepper have devised constraint and freedom set pairs for all of the six CASEs with different number of TYPEs in each. The FACT method is based on using a visual method to identify freedom and constraint spaces along with calculations in the design process.

2.5 Screw theory

In order to develop an immersive environment for user-centered compliant mechanism design, both visual representation and mathematical formulation is essential. The FACT method provides an excellent visual representation of the freedom and constraint spaces. We turned to screw theory to provide the mathematical underpinnings of the design approach. Screw theory is widely used in mechanism synthesis because it provides a compact mathematical formulation for motion in three-dimensional space. In this research screw theory can be used to describe both the constraint spaces and the freedom spaces of compliant mechanism design.

2.5.1 Introduction

In a rigid body motion, any displacement can be described using a screw. A screw mathematically represents rotation about a line in space and a translation about that line. This line is known as the screw axis. Screw motion can be defined using four parameters – the three components of a direction vector and the angle rotated about the screw axis.

2.5.2 Twists & Wrench

There are two kinds of screws. The first kind is known as the "twist" and represents the kinematics of the motion of a body in three-dimensional space. These kinematics involve the velocities of the body - linear and angular - about the axis of translation, and the relationship between the two known as the pitch. The pitch represents the ratio of the linear velocity to the angular velocity. The second kind of screw is known as the "wrench" which represents the constraints/forces/torques acting on the body. These two concepts are often known as duality [18] in kinematics and statics.

The twist, $\mathbf{T}^{\wedge}$**,** is formed by a pair of three dimensional vectors, $\mathbf{\Omega}$ and $\mathbf{V}$ written as (Eqn.2)

$$\boldsymbol{T}^{\wedge} = (\boldsymbol{\Omega} \mid \boldsymbol{V}) = (\boldsymbol{\omega s} \mid \boldsymbol{c} \times \boldsymbol{\omega s} + \boldsymbol{v s}) = (\boldsymbol{\omega s} \mid \boldsymbol{c} \times \boldsymbol{\omega s} + \boldsymbol{p \omega s}) \tag{2}$$

Where

Ω : angular velocity

V: linear velocity

s: vector denoting the direction of the twist axis

c: point on the twist axis

ω: magnitude of angular velocity along the axis

$\boldsymbol{v}$: partial linear velocity along the axis

$\boldsymbol{p}$: pitch

As special cases, a pure rotation and a pure translation in space are represented by a twist of zero pitch and a twist of infinite pitch respectively, written as (Eqn. 3-4):

$$\boldsymbol{R}: \quad \boldsymbol{T}^{\wedge} = (\boldsymbol{\omega s} \ : \ \boldsymbol{c} \times \boldsymbol{\omega s})\, \boldsymbol{p} = \boldsymbol{0} \tag{3}$$

$$\boldsymbol{T}: \quad \boldsymbol{T}^{\wedge} = (\boldsymbol{0} \ : \ \boldsymbol{v s})\, \boldsymbol{p} = \infty \tag{4}$$

Similarly, wrenches represent moments or forces. These are written as a force F and a couple M acting on rigid bodies. It could be written as (Eqn. 5)

$$\boldsymbol{W}^{\wedge} = (\boldsymbol{F} \mid \boldsymbol{M}) = (\boldsymbol{fu} \mid \boldsymbol{r} \times \boldsymbol{fu} + \boldsymbol{mu}) = (\boldsymbol{fu} \mid \boldsymbol{r} \times \boldsymbol{fu} + \boldsymbol{qfu}) \tag{5}$$

Where

W$^{\wedge}$ = Wrench

u: direction of the wrench axis

r: point on the wrench axis

f: magnitude of the force applied

m: magnitude of the partial moment along the axis

q: pitch ***q*** = *m* / *f*

As special cases, a pure force and a pure couple are represented as a wrench of zero and infinite pitch respectively (Eqn. 6-7).

$$(\boldsymbol{q} = \boldsymbol{0}): \boldsymbol{W}^{\wedge} = (\boldsymbol{fu} \mid \boldsymbol{r} \times \boldsymbol{fu}) \tag{6}$$

$$(\boldsymbol{q} = \infty): \boldsymbol{W}^{\wedge} = (\boldsymbol{0} \mid \boldsymbol{mu}) \tag{7}$$

2.5.3 Virtual power

The virtual power of a wrench **W**$^{\wedge}$ acting on a moving body with motion **T**$^{\wedge}$ is given by the reciprocal product of the wrench and the twist. It is written as (Eqn. 8)

$$\textbf{Virtual power} = \mathbf{T}^{\wedge} \circ \mathbf{W}^{\wedge} = \mathbf{F} \square \mathbf{V} + \mathbf{M} \square \boldsymbol{\Omega} \tag{8}$$

$$= [\,(fv + m\omega)(s \cdot u) + f\omega\,(c - r) \cdot (s \times u)]$$

$$= [\,(fv + m\omega)cos\alpha - f\omega sin\alpha]$$

A twist and a wrench are said to be reciprocal when the reciprocal product is zero. Reciprocal product is considered a linear operation on either twist or wrench separately. For instance, the reciprocal product of a twist **T**$^{\wedge}$ with a linear combination of two wrenches $\mathbf{W}^{\wedge}{}_{\mathbf{1}}$ and $\mathbf{W}^{\wedge}{}_{\mathbf{2}}$ can be expressed the linear combination of the reciprocal product of $\mathbf{T}^{\wedge}$ with each of the two wrenches, that is (Eqn. 9),

$$\boldsymbol{T} \bullet (a\, \boldsymbol{W}^{\wedge}_1 + b\, \boldsymbol{W}^{\wedge}_2) = a\, \boldsymbol{T}^{\wedge} \bullet \boldsymbol{W}^{\wedge}_1 + b\, \boldsymbol{T}^{\wedge} \bullet \boldsymbol{W}^{\wedge}_2 \tag{9}$$

Where coefficients ***a*** and ***b*** are arbitrary constants.

Irrespective to the magnitude of force *f* and angular velocity ω, the virtual power is zero if the following condition is satisfied (Eqn. 10),

$$\boldsymbol{p} + \boldsymbol{q} = \boldsymbol{a} \tan\boldsymbol{\alpha} \tag{10}$$

where *a* is the perpendicular distance between the twist and the wrench and α is the angle between the twist and the wrench.

The following statements can be developed:

a) When two pitches have the same magnitude but with opposite sign p = -q (including the case p = q = 0), the two screws are reciprocal if either a = 0 or sin α=0. This situation occurs when the twist axis and wrench axis are coplanar, i.e., intersecting or parallel to each other.

b) If one pitch is zero, e.g. q = 0, then the condition $\boldsymbol{p} + \boldsymbol{q} = \boldsymbol{a} \tan \boldsymbol{\alpha}$ becomes p = a tan α. And if the other pitch is infinite p = ∞ the two screw axes must be perpendicular, i.e. α = 90° in order to be reciprocal.

c) If both pitches are infinite, the screws are always reciprocal. The twist represents pure translation and the wrench represents a pure couple. In other words, a pure couple does no work on a pure translation.

d) If the two screw axes are perpendicular then cos α = 0, and their reciprocity is independent of their pitches. This can only occur when a = 0, therefore the axes intersect.

e) A screw is reciprocal to itself if its pitch is either zero or infinite.

2.6 Virtual reality

Virtual reality is a technology which lets users interact with computer generated three dimensional environments. It is used in a wide variety of applications which support a user's direct interaction with geometry, CAD images, and environments as a whole. Users can interact with digital images using hardware devices like 3D mice, joysticks, haptic devices, wands, etc. Some common VR technology used for 3D dimensional displays are the HMDs (Head Mounted Displays) and the CAVE. A HMD is a helmet worn by the user where separate left and right eye images are projected on a CRT or LCD located inside the helmet to generate stereo images. The helmet is also connected to a position tracker which tracks the user's motion and adjusts the user's viewpoint in space. The CAVE environment supports visual displays by locating large projection screens in a surround screen environment. Figure 14 shows a user in a CAVE environment.

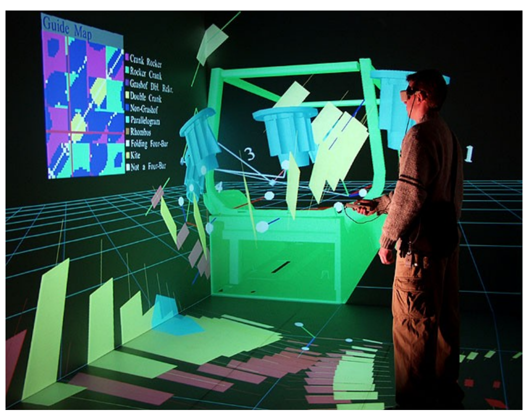


Figure 14: Virtual environment

Compliant mechanisms, are potential applications suitable for such environments due to their sheer three dimensional nature. Design of a 3D mechanism in a 2D environment such as a computer monitor, mouse and keyboard is difficult due to the need to define and validate the

3D motion of the mechanism. We wish to give the user a natural way of interacting with the mechanism which could be attained by the use of such immersive environments.

There have been efforts in designing such mechanisms via computer generated programs. A computer based tool known as CoMeT (Compliant Mechanism Tool) was developed by Kim and Culpepper [19] at the Massachusetts Institute of Technology. It used a GUI based approach with a MATLAB-based computational engine to aid the mechanism design process. These design steps include sketching the problem, defining the mechanism, specifying operation parameters and analyzing the resulting solution. While CoMeT proved to be a good tool, there were several drawbacks. It used a 2D interface to design a 3D mechanism which by its nature involved unintuitive interactions. It becomes difficult to make design alternatives while dealing with spatial mechanisms because the designer has to visualize and interact in 3D. The method was not user-centered and a novice mechanism designer could not use it efficiently.

A need for a 3D environment is of utmost importance which CoMeT lacked. Virtual Reality Technology provides such an alternative to the traditional 2D computer interface. It allows the user to walk around the mechanism (as in real life), and interact with it by performing actions in 3D space. This is attained through the use of position trackers, position tracked input devices and stereo display. Much research has been done in using VR as an immersive environment for the synthesis and design of mechanisms.

One of the first attempts in spherical mechanism synthesis to use Virtual Reality as a design interface was done by Osborn and Vance in 1995 [20]. Subsequent efforts included the development of VEMECS by Kraal and Vance [21] and ISIS by Furlong and Vance (1999)

[22]. In 2002 research expanded to explore the design of spatial mechanisms and VRSpatial was created [23].

None of the above mentioned efforts focused on compliant mechanism synthesis.

This research focuses on novice as well as experienced designers by giving them a natural way of interaction through VR technology and also a user-centered design paradigm for the design of compliant mechanisms.

2.7 Application of screw theory in constraint-based design approach

Constraint-based compliant mechanism design has proven to be a viable approach to use to design compliant mechanisms; however, the scope of the constraint-based approach has been limited to designers with extensive prior hands-on experience. The design rules which the approach presents are very subjective in nature which poses limitations on the part of the designer to design and optimize compliant mechanisms.

Ball [24] was the first to formulate screw theory in a systematic way. Hunt [25] and Phillips [26, 27] later developed the geometrical and mathematical representation of screws and screw systems. They used the screw theory for the synthesis and analysis of mechanisms. Since then, screw theory has also been applied topology synthesis [28].

2.7.1 Motions as twists

As stated earlier, constraints and freedom are the key concepts of the constraint-based design approach. To tie this to screw theory, we note that any motion (degree of freedom) of a body could be represented as a twist with a pure rotation or a pure translation. Any free body in space has six degrees of freedom – three rotations and three translations.

These six motions can be represented by 6×1 vectors as outlined in Table 1.

Table 1: Movement and twist representation

Movement	Twist Vector
X Rotation	$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$
Y Rotation	$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$
Z Rotation	$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$
X Translation	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$
Y Translation	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$
Z Translation	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$

The work of Su [25] et al. is of significance in the application of screw theory to freedom and constraint spaces. He has presented a screw theory based approach to represent freedom spaces in a twist vector representation. He states that a freedom space could be defined by a twist matrix given by (Eqn. 11)

$$\Pi_T = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_f \end{bmatrix} = \begin{bmatrix} \Omega_1 & \vdots & V_1 \\ \Omega_2 & \vdots & V_2 \\ \vdots & \vdots & \vdots \\ \Omega_f & \vdots & V_f \end{bmatrix} \tag{11}$$

Where

f: dimension of the freedom space

T_j: Basis twist that span the freedom space

Su also states that any motion in the freedom space can be denoted by a linear combination of the basis twists given by (Eqn. 12)

$$T^{\wedge} = \sum_{j=1}^{f} k_j T_j \tag{12}$$

Where

k_j: arbitrary constant

If the rank of the matrix is less than *f*, twists are said to be redundant which means that some twists can be written as the linear combination of others.

As an example, the parallel line freedom space introduced by Blanding can be written as two parallel rotational twists (Eqn. 13-14)

$$\boldsymbol{T_1} = (\boldsymbol{\Omega} : \boldsymbol{c_1} \times \boldsymbol{\Omega}) \tag{13}$$

$$\boldsymbol{T_2} = (\boldsymbol{\Omega} : \boldsymbol{c_2} \times \boldsymbol{\Omega}) \tag{14}$$

Because of the linearity property of twists, any parallel line can be represented by a linear combination of these two twists (Eqn. 15):

$$\boldsymbol{T} = \boldsymbol{k_1 T_1} + \boldsymbol{k_2 T_2} = ((\boldsymbol{k_1} + \boldsymbol{k_2})\boldsymbol{\Omega} : (\boldsymbol{k_1 c_1} + \boldsymbol{k_2 c_2}) \times \boldsymbol{\Omega} \tag{15}$$

The approach taken in this research is based on the user-centered design paradigm. This system focuses on the user as he/she defines the desired motion and selects from a set of potential solutions to pick the final design configuration. The interface incorporates several interface design principles and theories to support the user in the design process. The next chapter introduces the methodology and solution to the problem.

Chapter 3: Methodology

This chapter introduces the methodology behind solving the compliant mechanism design process from a user-centered perspective. The approach as explained in the following sections shows how the design process could focus on the designer by abstracting the mathematical process in the background to support design decision making.

3.1 Screw theory

As explained above, screw theory provides a mathematical approach for defining motions and eventually freedom spaces as twist vector representations. Desired motions must be defined mathematically as twists and wrenches. Although much work has been done to provide a mathematical approach to the design of compliant mechanism, this approach is not user-centered as it requires a deep understanding of complex mathematics to define the problem before reaching a final solution.

3.2 FACT

The FACT theory gives a detailed geometrical representation of all the freedom spaces and their associated constraint spaces. The theory is very helpful at the end of the design process once it is determined in which of the CASE and TYPE the user motion falls into. The issue with the FACT design method lies when the user is in the process of defining the motion and deciding which CASE and TYPE it falls into. The approach asks the designer to follow a visual based process where a user has to analyze the defined motion visually and see what freedom or constraint space it falls into. This approach is not easy for novice designers or the ones who don’t have much experience with the approach. The designer needs to know the intricacies as to what the freedom and constraint spaces mean and how have they been developed, before he/she could go forward to decide one of them for a given motion.

3.3. Proposed approach

A gap exists in the current design process such that the two existing approaches to designing compliant mechanisms have different advantages and disadvantages, but neither of them are integrated or follow a user-centered approach. The proposed approach tries to bridge this gap by using both the above methodologies and providing the user a 3D immersive interface to design compliant mechanisms. The interface helps the user during the design process through its intuitive UI. This approach supports novice designers to enter the compliant mechanism design domain as it abstracts the complex mathematical calculations from screw theory and provides an environment where the designer both defines the desired motion and selects from alternate solutions to pick the final design. The user is presented with an immersive interface where he/she defines the desired motion by simply grabbing and moving the object or by selecting one of the pre-defined presets. All the mathematical calculations are done by the system and the algorithm selects the appropriate constraint space. Once the constraint space is determined, it is displayed in front of the user to allow the user to explore multiple design solutions within the feasible design space.

The method begins with the establishment of a catalogue of the twist vector representations of all the freedom spaces which is pre-calculated and stored. The freedom spaces define the allowable motions for a body. Next, the user defines a three dimensional motion and the twist vector representations of the motion is calculated. At this point, the system contains the twist vector representations of the desired motion and also the description of the freedom spaces.

The algorithm checks for linear independence between the desired motion and the catalog of freedom spaces. When linear independence is achieved, the system identifies the appropriate CASE and TYPE for this design. The system identifies and displays the corresponding constraint space in front of the user to let the user explore the set of potential constraints and iterate to the final solution.

As noted above, the method requires a catalogue of the twist vector representations of all the freedom spaces defined in the FACT theory. This catalogue should be defined before the design process starts. For the proof of concept, we have developed the catalogue for CASE 3 TYPE 1, 4 and 5 freedom spaces. This thesis presents a demonstration of this method with a case study using a ball joint example. The following section explains how to come up with a twist vector catalogue of the freedom spaces.

It is to be significantly noted that the freedom spaces, as explained by Hopkins, are not associated with any coordinate system. They show no information about their orientation and location in space. To come up with a twist vector of them, we need to put the freedom spaces in a coordinate system. The freedom spaces could be attached to the coordinate system in three different ways where they could lie in any of the x-y, y-z or z-x planes. Therefore, there would be three twist representations of each freedom space.

The following section explains freedom space TYPE 1, 4 and 5 within CASE 3 with their diagrammatic representation. Analysis of individual freedom spaces is done and the possible motions they represent are calculated mathematically in a twist vector form. The diagrams shown in red represent the freedom spaces. The constraint spaces will be shown in blue.

3.3.1 CASE 3 TYPE 1

The freedom space for CASE 3 TYPE 1 is displayed in Figure 15. CASE 3 is defined by having three constraints and three degrees of freedom. This particular type in CASE 3, i.e. TYPE 1, allows two rotational degrees of freedom along two axes lying in a plane and perpendicular to each other and one translational degree of freedom perpendicular to the plane. This freedom space is represented geometrically by a hoop and a square. The hoop representation allows one translational degree of freedom along the line passing perpendicular to the plane of the hoop and through its center. The square shape represents rotation in two directions: around an axis parallel to either side of the square.

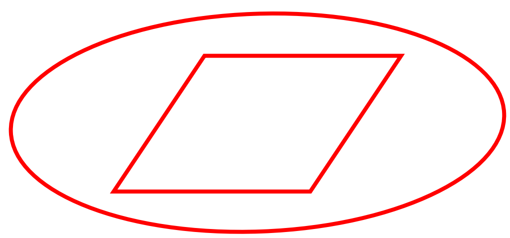
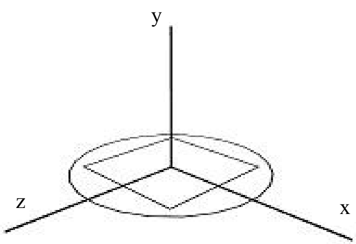
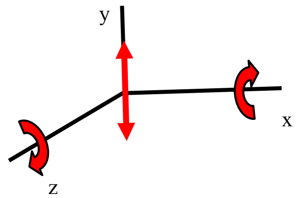
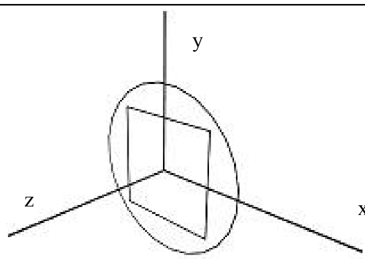
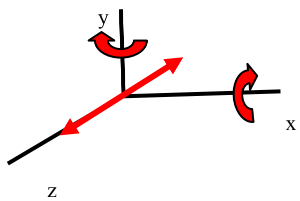
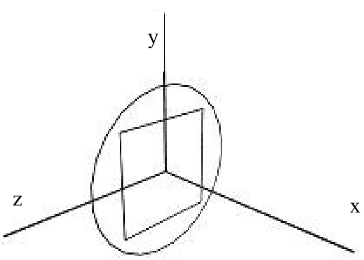
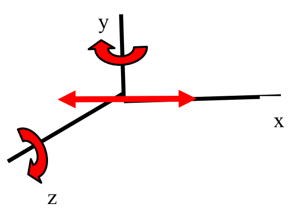


Figure 15: Original freedom space, CASE 3 TYPE 1

In order to couple the geometric representation with the screw mathematics, we need to place the geometric freedom space within a coordinate system in order to define the twist vectors. Table 2 shows the three orientations, the allowed motions and the resultant twist representations for the CASE 3 TYPE 1 freedom space.

Table 2: CASE 3 TYPE 1 with three orientations

Freedom space in three orientations	Allowed motions	Twist representations
		$\begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$
		$\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$
		$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$

3.3.2 CASE 3 TYPE 4

The CASE 3 TYPE 4 freedom space is represented as shown below. CASE 3 again consists of three constraints and allows three degrees of freedom. TYPE 4 consists of three rotational degrees of freedom along an arbitrary set of axes and no rotations. It is represented geometrically as a set of infinite lines all intersecting in one point (Fig. 16).

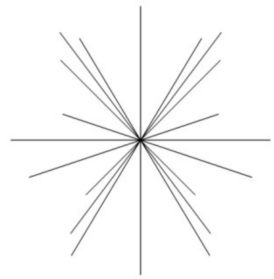


Figure 16: Original freedom space, CASE 3 TYPE 4

This freedom space is axially symmetric so that it does not have any one orientation. Therefore, a coordinate system can be attached in any orientation and it would produce the same twist representation (Figure 17). The twist vector representation is shown in Equation 16.

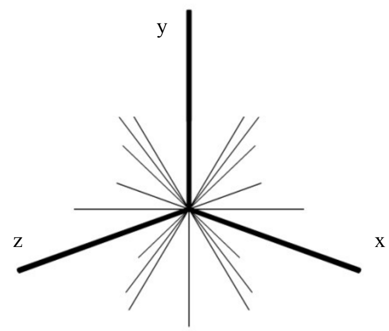


Figure 17: Freedom space with axis, CASE 3 TYPE 4

$$\text{Twist vector representation} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{16}$$

3.3.3 CASE 3 TYPE 5

The CASE 3 TYPE 5 freedom space once again allows three degrees of freedom (CASE 3). TYPE 5 allows two translations and one rotation. It is represented geometrically in Figure 18 and consists of a series of hoops, from which two translations can be produced and an infinite set of parallel lines representing the one rotational degree of freedom. Table 3 presents the various twist vectors associated with CASE 3 TYPE 5.

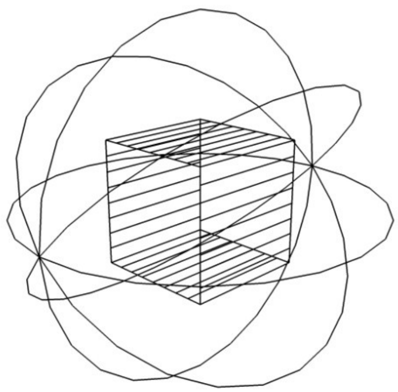


Figure 18: Original freedom space, CASE 3 TYPE 5

Table 3: CASE 3 TYPE 5 with three orientations

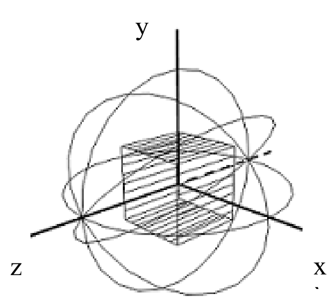
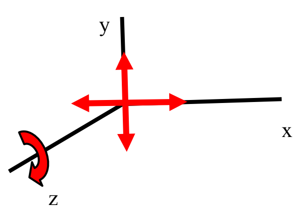
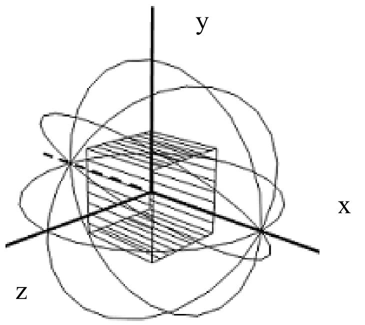
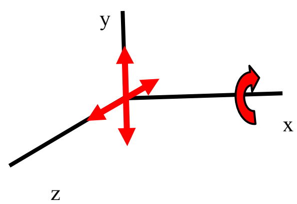
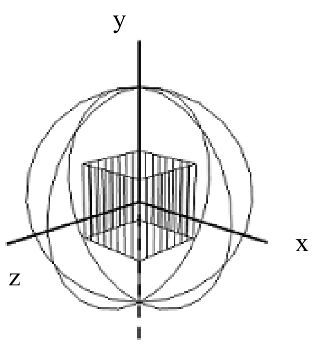
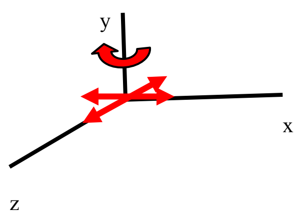
Freedom space in three different orientations	Allowed motions	Twist representations
		$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$

Table 3. (continued)

Freedom space in three different orientations	Allowed motions	Twist representations
		$\begin{bmatrix}1\\0\\0\\0\\1\\1\end{bmatrix}$
		$\begin{bmatrix}0\\1\\0\\1\\0\\1\end{bmatrix}$

At this point we have the user's motion in a twist vector form and also the freedom space representations in twist vector form.

Once the system knows how many independent motions the user wants, the CASE number is automatically known. The CASE in the FACT (Freedom and Constraint-based Topologies) method defines the number of constraints applied on the body. For e.g. CASE # 1 denotes one constraint which means if the motion falls into CASE 1, it has five degrees of freedom as only one constraint is applied. From the above example, if the user has defined two independent motions, the CASE number becomes CASE 4.

3.4 Proposed solution

Now that the relationship between screw theory and compliant mechanism design has been developed, a tool can be developed for the design and analysis of compliant mechanisms. The solution proposed here follows a user-centered design paradigm where several interface design principles and theories have been followed to give an intuitive GUI to the user.

3.4.1 Scenario

The entire design process is outlined in Figure 19. A user gets into the immersive virtual reality environment and sees a virtual object in front. The user has a pre-defined goal for the desired motion path. The user grabs the object with a 3D input device and defines the desired motion by rotating or translating the object. She/he marks every independent motion by explicitly telling the system about each one of them. This task is supported through the use of a floating menu in the immersive environment. Once the user is finished with defining the path, he/she selects the “Finish” option in the menu to let the system do the processing. The system then identifies the appropriate constraint spaces, which support the defined motion path. The user selects appropriate constraints from the space (guided by design principles) which then results in a final mechanism design.

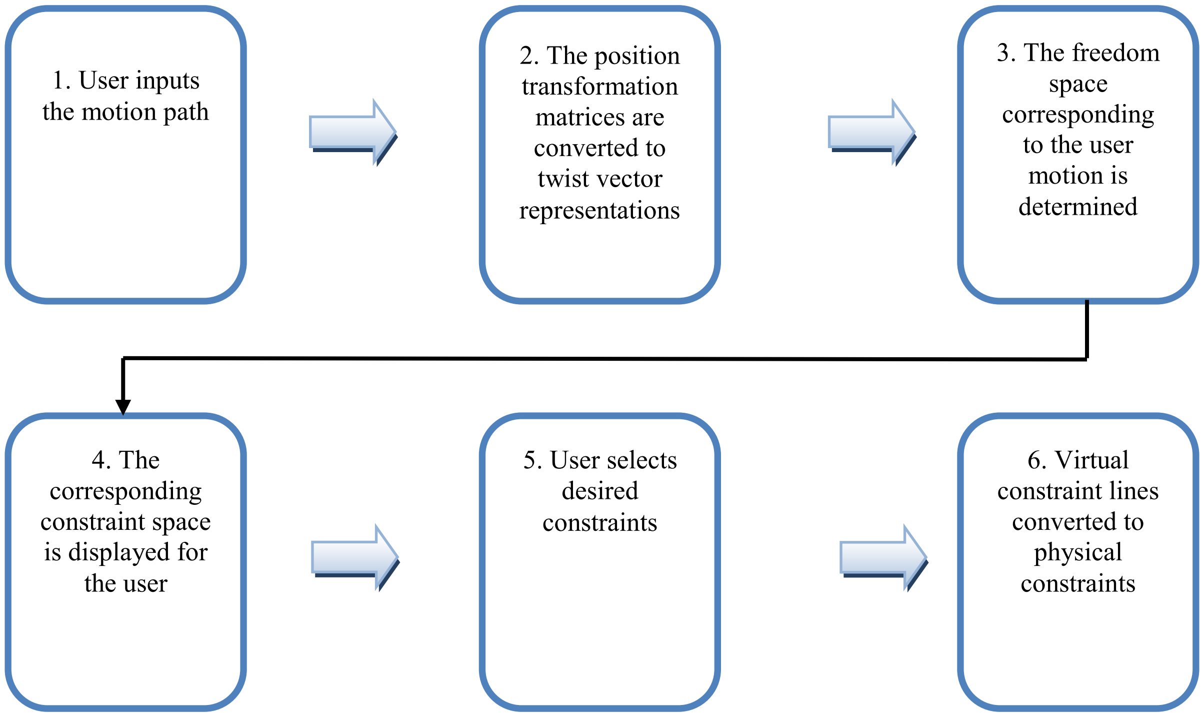


Figure 19: Proposed method

3.5 Steps explained

This section contains more detailed description of the design steps.

3.5.1 Step 1: User defines motion

The user grabs the object and locates it to a position by translating or rotating it. This defines the first motion. The user makes a menu selection to declare the first independent motion. As the user does that, the object snaps back to the original position to let the user start from the beginning in case he/she wishes to define another motion. Once the user is finished, the system has n + 1 number of matrices ('n' number of positions and 1 starting position where n ≤ 6).

Once the user is finished with defining the motion path, the relative transformation matrices for the motions have been defined. These matrices are 4 x 4 GMTL (Generic Math Template Library) matrices. The user interaction in this step is natural as he/she uses a 3D interaction device to define the object motion.

3.5.2 Step 2: Twist vectors calculated

Once the system has the transformation matrices, they are converted to their twist vector representation.

3.5.3 Step 3: Freedom space determined

After the completion of the second step, the system has the twist representations of the individual motions defined by the user. The next step involves determining the CASE and TYPE of the freedom space in which this user defined motion falls into. The CASE number is determined from the number of desired independent motions. Once the CASE number is known, the next step is to determine the TYPE within that CASE.

a. There are two ways in which the user motions could be matched to an appropriate freedom space. If the user motions are defined along orthogonal axes, a simple comparison of the generated twist vector with the stored twist vectors in the catalog would identify the TYPE.

b. If the user motions are not defined along orthogonal axes, then the generated twist vector would have to be checked for linear independence with all other twist vector representations in the specific CASE to determine the correct freedom space.

3.5.4 Step 4: Constraint space displayed

Once the appropriate freedom space is determined, the corresponding constraint space is displayed. A database of all the freedom spaces and the constraint spaces is already known by the system. Those constraint spaces are determined by Blanding's rule of Complimentary patterns. The constraint space is displayed as an overlay on the object. This gives the user the understanding of the constraint space with respect to the object and makes the selection of constraints clear

3.5.5 Step 5: User selects constraints

The user now sees the constraint space overlaid on the object. The user selects $6 - n$ constraints from the constraint space where n is the number of motions defined by the user. As the user selects the virtual constraint lines, the color of the line changes to indicate it has been selected. The user selects "Done" from the menu once he/she is satisfied from the selection. Because there are generally sets of infinite numbers of constraint lines in each geometric constraint space set, the software provides the user with the ability to select from the displayed constraints or any of the infinite number of consistent constraints.

The ability of the user to select the final constraints from the complete viable constraint set allows the user to apply design knowledge to the process. There are several considerations which the designer should keep in mind while picking constraints from the constraint space. Some of them as explained by Blanding, are explained below.

- **Avoid over-constraints**: When the condition of over-constraint occurs, two or more constraints control the same degree of freedom of a body thus resulting in redundant constraints.

- **Angle between two constraints**: When selecting two constraints, according to Blanding, the designer should not be concerned about the angle between the two constraints as long as their point of intersection remains the same. Any two constraints lines are functionally equivalent to any other pair in the same plane whose constraint lines intersect at the same point.
- **Parallel constraints**: When the designer needs to apply two constraints parallel to each other in a plane, any two parallel constraint lines from that plane could be picked and they would be functionally equivalent. The condition of over-constraint should definitely be avoided.
- **Designing for thermal expansion**: A designer should also keep in mind the expansion of the material which could happen as a result of thermal expansion. The design should be such that the constraints applied on the body still hold true in the event of thermal expansion of the body.

3.5.6 Step 6: Final constraints appear

As the user selects "Done" from the menu option, the constraint space disappears. The constraint lines selected by the user are displayed as the final design constraints.

Chapter 4: Case study

This chapter demonstrates the theory behind designing a compliant mechanism as applied to a spherical ball-joint example.

Ball joints are useful elements in mechanism design. They allow full rotations around an infinite number of axes, but no rotations. Ball joint consists of a stud and a socket enclosed in the casing. The joint has a protective casing which prevents dirt from entering the joint. A typical example of a ball joint is shown in Figure 20.

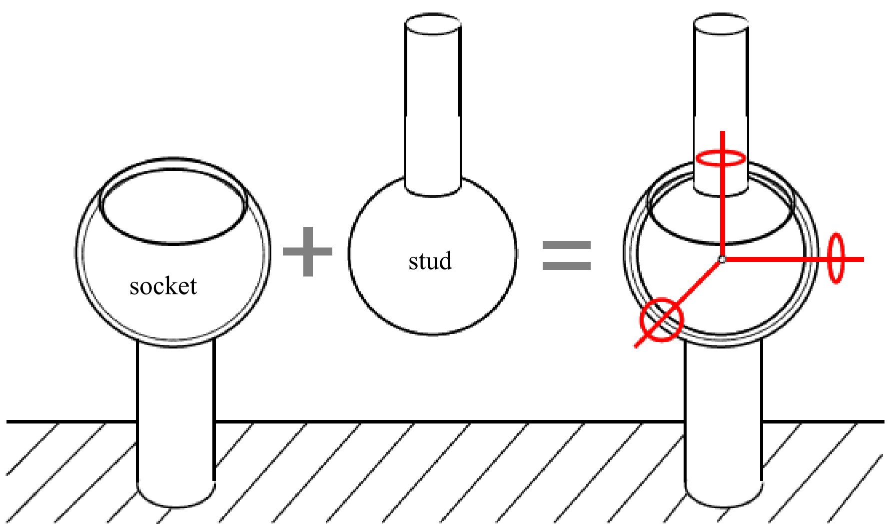


Figure 20: A ball joint

In this example, the user defines rotation motions in three independent axes and the system automatically comes up with the corresponding constraint spaces.

4.1 Step 1

The user grabs the object and defines rotational motions along each of the three axis. After every rotation motion, the object snaps back to its original position to let the user start to define the next motion.

4.2 Step 2

The 4 x 4 rotation matrices are converted to twist vector representation, which is a 6 x 1 vector. The twist vector stores the velocity of a body. It has six components: three of to define the linear velocity and three to define the angular velocity.

The equivalent twist representations of the above matrices are listed in Table 4.

Table 4: Orthogonal rotation and their twist vector representations

Rotations along axis	Twist vector representations
Rotation along X axis	$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$
Rotation along Y axis	$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$
Rotation along Z axis	$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

The twist vectors are combined into a resultant twist vector (Eqn. 17):

$$T_R = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{17}$$

Where T_R is the resultant twist vector

4.3 Step 3

The software identifies this as a CASE 3 situation since three motions were input. The next step is to identify the TYPE within CASE 3. The match is found by comparing the twist representation of the user motion with those of the freedom spaces.

The table below shows how the comparison will take place and the results associated with it. Note that each instance of a type represents a difference coordinate system location.

Table 5: CASE 3 TYPE matching

User motion	Type instance	Result
$\begin{bmatrix} \mathbf{1} \\ \mathbf{1} \\ \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$ EQUALS	Type 1 Instance 1 $\begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$	FALSE
	Type 1 Instance 2 $\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$	FALSE

Table 5. (continued)

User motion		Type instance	Result
$\begin{bmatrix} \mathbf{1} \\ \mathbf{1} \\ \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$	EQUALS	Type 1 Instance 3 $\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	FALSE
		Type 4 Instance 1 $\begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	**TRUE**
		Type 5 Instance 1 $\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$	FALSE
		Type 5 Instance 2 $\begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$	FALSE
		Type 5 Instance 3 $\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$	FALSE

It is to be noted that the algorithm will not go beyond TYPE 4 Instance 1 as a correct match is found at that point.

4.4 Step 4

Once the exact TYPE is found, the constraint space corresponding to the freedom space is displayed for the user to manipulate. The corresponding constraint spaces for CASE 3 TYPE

4 is shown in Figure 21. The lines in blue represent constraint lines and the lines in red represent freedom lines.

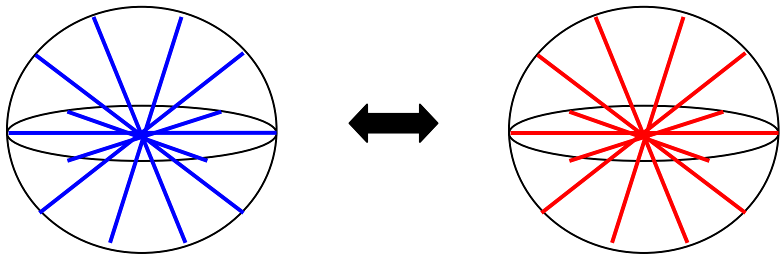


Figure 21: Constraint and freedom space for CASE 3 TYPE 4

Figure 22 shows the constraint space located in the coordinate system and attached to the object to be manipulated.

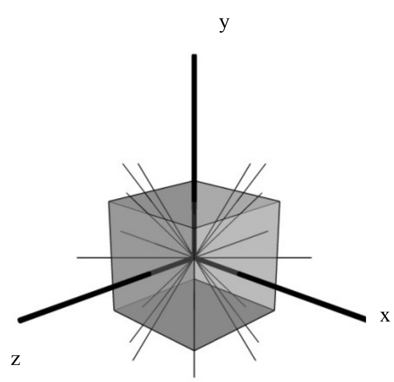


Figure 22: Constraint space applied on the shaded object

4.5 Step 5

Once the constraint space is displayed, the user select from the desired constraints to define the final solution. The system prompts the user to select no more or less than three constraints as this motion belongs to CASE 3. As the user selects the desired three constraints, the system gives a feedback to the user by changing the color of those constraints (orange in the diagram below). Once the user is satisfied with the selection, he/she could click "Done" from the Menu.

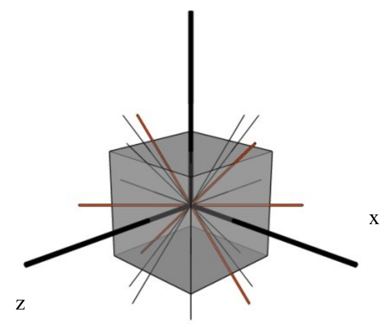


Figure 23: Constraints selected while the object is displayed

4.6 Step 6

One the user clicks on "Done" from the menu; the constraint lines selected from the space above are turned into the final design constraints.

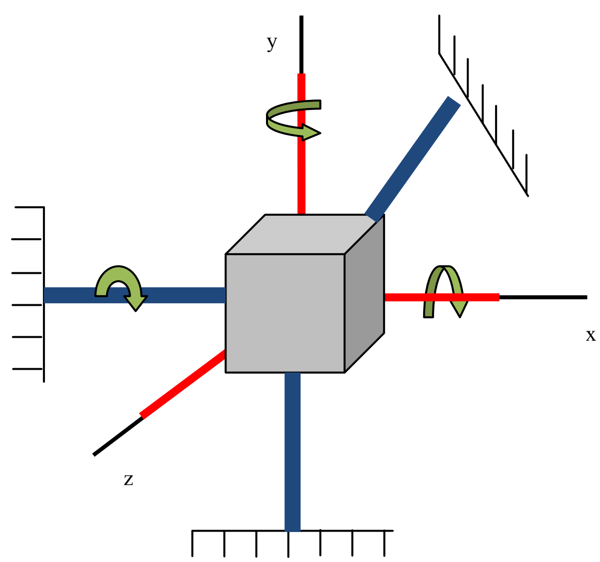


Figure 24: Final mechanism

The final mechanism is shown in Figure 24 and consists of a design with three constraints that limit all translational motion and provide for free rotation about any axis. This design has achieved the initial design objectives.

Chapter 5: Conclusion and Future work

Currently, significant experience is required to design compliant mechanisms using the constraint-based methods because of the non-intuitive motion of the compliant members. The motions made by the compliant members of the mechanism are non linear and therefore are difficult to understand intuitively. Using the current design method, significant knowledge and hands-on experience is required on the part of the designer to successfully design such a mechanism.

The approach presented in this thesis combines two methods of compliant mechanism design within an immersive VR interface. The FACT approach, developed at the Precision engineering lab at MIT, is a constraint-based approach which provides a geometric representation of all the constraint and freedom spaces. This approach requires the user to have a deep understanding of the geometrical representations of the spaces before applying them to the design process. The mathematics of screw theory provides a mathematical background for the design of compliant mechanisms that is independent of the visual design representation. According to this approach freedom spaces are represented mathematically using twist and wrench vectors. Here the user solves mathematical equations which are removed from any visual representation.

The approach integrates the above methodologies and provides a user-centered approach for the design process. A VR immersive interface is presented to the user which gives a natural way to design 3D mechanisms. Before the design process starts, a catalogue of twist vector representations of all the freedom spaces is prepared. When the user defines the desired motion, the system calculates the twist vectors for the motions and matches them with the catalogue to come up with the appropriate constraint and freedom space. The user is then free to select the individual constraints from the possible set to complete the design process. This research results in an intelligent design framework that will allow a broader group of engineers to design complex compliant mechanisms, giving them new options to draw upon when searching for design solutions to critical problems. The user-centered strategy followed in this research is novel in that it combines purely visual representations with mathematical representations and allows the designer the freedom to select from the solution set to arrive at the final design. This method uses the power of mathematics combined with visual and interactive methods to support compliant mechanism design. The research supports novel mechanism solutions for manufacturing and product design which have fewer movable joints, are more robust, and are easily scaled to meet the needs of micro-products.

A case study was presented which demonstrates the proposed approach. The six step design process presents a detailed description of how to proceed with the mechanism design. Although the proof of concept has been presented, much work still needs to be done to expand the scope. As of now, only the user motions which belong to CASE 3 TYPE 1, 4 and 5 will be recognized by the software.

There are several promising areas for future work:

1. **Catalogue expansion**: The current catalogue of freedom spaces contains twist vector representations for CASE 3 TYPE 1, 4 and 5. This restricts the user motion to be identified only when it falls within those categories. In the future, the catalogue could be expanded so that it has the twist vector representations of all the CASEs and TYPEs in order to accommodate any arbitrary input motion.
2. **Support for motions not defined at orthogonal axis**: The current implementation does not support motions which are not defined on orthogonal axis. A linear

independence check needs to be placed when the motions are not defined along orthogonal axis so that the correct freedom spaces are calculated in accordance to the defined user motion.

3. **Verify motion**: The current system does not verify that the compliant mechanism moves along the direction defined by the user. Mathematically we are confident that the results can be verified, however, there is no visual feedback to the user. Once the mechanism has been designed and the final design constraints appear, there should be a way where the user could verify the object's motion. Deflection calculations based on material properties are needed to verify the design.

In conclusion, this research resulted in a promising method to support user-centered design of compliant mechanisms through the combination of mathematical and geometric representations and an immersive virtual design environment. Future work will result in even greater impact of this design tool on the designer's ability to design compliant mechanisms.

References

[1] Howell, L. L., Compliant Mechanisms. John Wiley & Sons, Inc., New York, 2001

[2] Howell, L. L., and Midha, A., 1995. "Parametric deflection approximations for end-loaded, large deflection beams in compliant mechanisms". ASME Journal of Mechanical Design, 117(1), pp. 156–165.

[3] Ananthasuresh, G. K., Kota, S., Kikuchi, N. "Strategies for systematic synthesis of compliant MEMS". Proceedings of the 1994 ASME Winter Annual Meeting, November 1994 ASME Winter Annual Meeting, Dynamics Systems and Control, Chicago, IL, DSC-Vol. 55-2, pp. 677-686.

[4] Frecker, M. I., Ananthasuresh, G. K., Nishiwaki, S.,Kikuchi, N., and Kota, S., 1997. "Topological synthesis of compliant mechanisms using multi-criteria optimization". ASME Journal of Mechanical Design, 119(2), pp. 238–245.

[5] Sigmund, O., 1997. "On the design of compliant mechanisms using topology optimization". Mechanics of Structures and Machines, 25(4), pp. 493–524.

[6] Wang, M. Y., Chen, S., Wang, X., and Mei, Y., 2005. "Design of multimaterial compliant mechanisms using level-set methods". ASME Journal of Mechanical Design, 127(5), pp. 941–956.

[7] Hull, P. V., and Canfield, S., 2006. "Optimal synthesis of compliant mechanisms using subdivision and commercial fea". ASME Journal of Mechanical Design, 128(2), pp. 337–348.

[8] Zhou, H., and Ting, K.-L., 2005. "Topological synthesis of compliant mechanisms using spanning tree theory". ASME Journal of Mechanical Design, 127(4), pp. 753–759.

[9] Howell, L. L., and Midha, A., 1996. "A loop-closure theory for the analysis and synthesis of compliant mechanisms". ASME Journal of Mechanical Design, 118(1), pp. 121–125.

[10] Su, H.-J., and McCarthy, J. M., 2006. "A polynomial homotopy formulation of the inverse static analysis of planar compliant mechanisms". ASME Journal of Mechanical Design, 128(4), pp. 776–786.

[11] Maxwell, J. C., 1890. General considerations concerning scientific apparatus. Dover Press.

[12] Blanding, D. L., 1999. Exact Constraint: Machine Design Using Kinematic Processing. ASME Press, New York, NY.

[13] Hale, L. C., 1999. “Principles and Techniques for Designing Precision Machines”. PhD Thesis, MIT, Cambridge, MA.

[14] Awtar, S., and Slocum, A. H., 2007. “Constraint-based design of parallel kinematic xy flexure mechanisms”. ASME Journal of Mechanical Design, 129(8), pp. 816– 830.

[15] Maxwell, J. C., “General considerations concerning scientific apparatus” in ‘The Scientific Papers of James Clerk Maxwell’ ed W. D. Niven, Dover Press, 1890. Reprinted from the Handbook to the Special Loan Collection of Scientific Apparatus, 1876.

[16] Blanding, D. L., Exact Constraint: Machine Design Using Kinematic Principles. ASME Press, New York, 1999.

[17] Hopkins J.B., “Design of parallel flexure systems via Freedom and Constraint topologies (FACT)”, PhD Thesis, MIT Cambridge, MA.

[18] Shai, O., and Pennock, G. R., 2006. “A study of the duality between planar kinematics and statics”. ASME Journal of Mechanical Design, 128(3), pp. 587–598

[19] Culpepper, Martin L. and Soohyung Kim, “Framework and Design Synthesis Tool Used to Generate, Evaluate and Optimize Compliant Mechanism Concepts for Research and Education Activities.” Proceedings of the 2004 ASME Design Engineering Technical Conference and Computers and Information in Engineering Conference. Salt Lake City, UT, pp.1583-1588, DETC2004- 57606.

[20] Osborn, S. W. and Vance, J.M., A Virtual Environment for Synthesizing Spherical Four-Bar Mechanisms, Proceedings of the 1995 Design Engineering Technical Conferences, Boston, MA, DE-83:885-892, September 1995.

[21] Kraal, Juliet C., and Vance, Judy M., "VEMECS: A virtual reality interface for spherical mechanism design," Journal of Engineering Design, 2001, vol. 12, no. 3, pp. 245-254.

[22] Furlong, Todd J., Vance, Judy M., and Larochelle, Pierre M., “Spherical mechanism synthesis in virtual reality,” ASME Journal of Mechanical Design, Vol. 121, No. 4, December 1999, pp. 515-520.

[23] Vance, Judy M. Larochelle, Pierre M.; Dorozhkin, Denis "VRSpatial: Designing spatial mechanisms using virtual reality," ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Montreal, Canada, September 29 - October 2, 2002, DETC2002/MECH-34377.

[24] Ball, R. S., 1998. The Theory of Screws. Cambridge University Press, Cambridge, England. Originally published in 1876 and revised by the author in 1900, now reprinted with an introduction by H. Lipkin and J. Duffy.

[25] Hunt, K. H., 1978. Kinematic Geometry of Mechanisms. Oxford University Press, New York, NY.

[26] Phillips, J., 1984. Freedom in Machinery. Volume 1, Introducing Screw Theory. Cambridge University Press, Cambridge, U.K.

[27] Phillips, J., 1990. Freedom in Machinery. Volume 2, Screw Theory Exemplified. Cambridge University Press, Cambridge, U.K.

[28] Kong, X., and Gosselin, C. M., 2004. "Type synthesis of 3-dof translational parallel manipulators based on screw theory". ASME Journal of Mechanical Design, 126(1), pp. 83–92.