UNIVERSITY OF CALIFORNIA

IRVINE

Analysis and Design of Bennett Linkages

PROJECT

Submitted in partial satisfaction of the requirements for the degree of

MASTER OF SCIENCE

in Engineering

by

Alba Perez Gracia

Project Comitee:

Professor J. Michael McCarthy, Chair

Professor James E. Bobrow

Professor Derek Dunn-Rankin

Table of Contents

List of Figures

Acknowledgements

Chapter 1

Overview

1.1. Kinematic Synthesis of Linkages

The design of geometric constraints to guide a body through a specified movement is called *kinematic synthesis*. If the geometric constraints are in the form of an open or closed chain of rigid links and joints such as a robot manipulator, the device is called a *linkage*. The kinematic synthesis of a linkage formulates the geometric constraint equations associated with the joints of the system and then solves for the dimensions of the chain that ensure its movement through a prescribed task.

This project focuses on the kinematic synthesis of spatial chains formed by a pair of revolute joints, the *RR dyad*. Planar and spherical RR dyads form the basic elements for the construction of planar and spherical linkages. Spatial RR dyads are found as elements in most all robot manipulators. However, the synthesis theory for general RR chains is theoretically challenging and has not been implemented in Engineering practice.

This project presents a new formulation of the synthesis equations for a spatial RR chain that simplifies the design process.

1.2. The Revolute-Revolute Chain

The kinematic synthesis theory for planar RR chains, in which the axes of the revolute joints are parallel, is well developed, see Hartenberg and Denavit (1964), Sandor and Erdman (1984). Similar results exist for spherical RR chains, where the axes of the joints intersect in a point, see Chiang (1988) and Ruth and McCarthy (1998). In contrast to the planar and spherical chains which depend on only one parameter, either the length or angle of the connecting links, the spatial RR chain depends on two parameters, the length of the common normal between the axes, and an angle of twist about the common normal. These are often termed the Denavit-Hartenberg parameters of the link, see Craig (1989).

The constraint equations that define the spatial RR dyad were studied by Roth (1967), who showed that for a task defined by three locations of a goal frame, there were no more than 24 RR chains. Vedkamp (1967) found that the instantaneous version of this problem, in which the task is defined by the position, velocity and acceleration of a goal frame, has two solutions. Suh (1969) showed that the finite problem has two RR chain solutions as well, and further that these chains can be assembled to form a Bennett linkage. Tsai and Roth (1973) analyzed the algebraic constraint equations and prove that the three position synthesis problem always has two solutions which form a Bennett linkage.

The Bennett Linkage is a 4R spatial closed chain. Bennett (1903) discovered the geometric relations that ensure that this chain can move freely with one degree of freedom. Research on the Bennett linkage has focused on its instantaneous kinematic geometry. See for example, Bennett (1914), Waldron (1969), Baker (1979, 1988), and Yu (1981). The synthesis theory for Bennett linkages consists of the results by Suh (1969) and Tsai and Roth (1973).

Recent work by Huang (1996) shows that the axes of the finite displacement screws generated by the coupler of a Bennett Linkage forms a geometric entity known as a *cylindroid*. The cylindroid has been studied in detail as part of the analysis of linear combinations of screws, Hunt (1978). The set of displacement screws generated by the movement of a link in a kinematic chain is known as called its *constraint manifold*, McCarthy (1990). Constraint manifolds have been shown to provide a convenient formulation for constraint equations in the kinematic synthesis of a linkage, Murray (1996). Thus, Huang has shown that axes associated with the constraint manifold of an RR chain, when constrained to move like a Bennett linkage system, forms a well-known geometric object.

1.3. Overview of the Project

The goal of this project is to apply the synthesis methodology of constraint manifolds to RR chain design. Because the displacement axes of the specified positions must lie on the constraint manifold of resulting RR chain, we see that these axes must generate the cylindroid identified by Huang. This is the insight that leads to our new geometric formulation.

Chapter Two defines the geometry of the spatial RR chain and the Bennett linkage. The analytical relationship between the input to the linkage and the coupler angle is presented. The constraint manifold for the RR chain is derived and specialized to the case of the coupler of a Bennett linkage to obtain a cylindroid.

Chapter Three examines the geometry of a cylindroid defined by the linear combination of two screw axes. In particular the goal is to determine the principal axes of the cylindroid.

Chapter Four formulates the design equations of the RR chain. For the three position problem, there are 10 equations in 10 unknowns. We use the geometry of the cylindroid generated by the specified positions to obtain special coordinates that simplify these design equations. The result are four equations in four unknowns that we solve numerically using Maple V.

Chapter Five presents the design procedure and two examples. Chapter Six presents a summary of the results and suggested further research. Three Appendices provide example worksheets from Maple V, as well as the libraries of functions implemented to generate the design equations.

Chapter 2

Kinematic Analysis of RR Chains

2.1. Introduction

In this chapter we analyze the RR chain and the Bennett linkage. We study their geometry and angular relations and derive the constraint manifold for the RR chain as the displacement screw axes given by the motion of the chain. The constraint manifold of the Bennett linkage is then related to that of the RR chain as a particular case given by the angular relation of the coupler.

2.2. The Spatial Revolute-Revolute Chain

The spatial RR chain is an open linkage consisting of two revolute joints linked by one rigid element. The position and orientation of both joint axes are arbitrary, so that the mechanism has spatial movement. Figure 2.1 shows a schematic design for the RR chain.
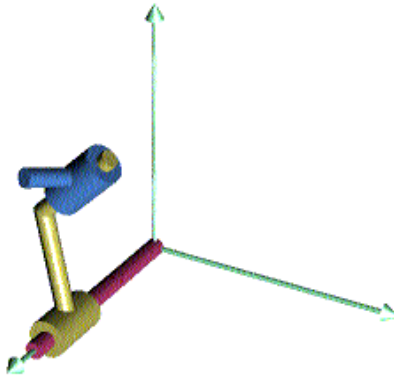
Figure 2. 1. The spatial RR chain consists of two revolute joints.

The design of RR dyads is related to the design of Bennett linkages. In fact, Tsai and Roth showed in 1973 that the two RR chains obtained when solving the 3-position synthesis problem can be linked together to form a Bennett linkage.

2.3. The Bennett Linkage

2.3.1 The geometry of the Bennett linkage

The Bennett linkage was discovered by Bennett (1903). It is a special case of a 4R spatial closed chain as showed in figure 2.2.
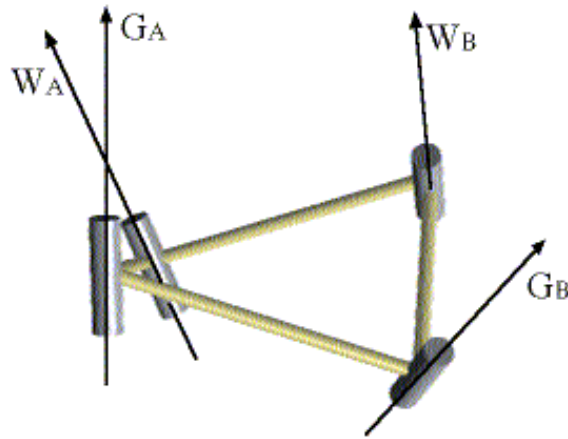
Figure 2. 2. The Bennett linkage is a movable closed chain formed by four revolute joints

In general, a 4R spatial closed chain forms a rigid structure. We can compute its degrees of freedom F using a formula for closed chains, known as Grubler's criterion:

$$F = 6(n-1) - \sum_{i=1}^{m}(6 - f_i) \qquad (2.1)$$

where n is the number of links in the system (including the base), m is the number of joints, and f_i is the freedom of the ith joint. In our case, we have $n=4$, $m=4$, and $f_i = 1$, which yields $F=-2$. This shows that a general 4R chain is overconstrained and will not move.

However, the Bennett linkage is movable with one degree of freedom. The movement is allowed due to its special geometric configuration. Bennett defined the conditions that the mechanism must satisfy to be able to move as the following:

1.   The opposite sides of the mechanism (i.e., links that are not concurrent) have the same lengths, denoted by p, r.

10

2. The angles of twist are denoted by ρ, ξ, and they are equal on opposite sides but with different sign.

3. The link lengths and link twist angles must satisfy the relation:

$$\frac{\sin \rho}{r} = \frac{\sin \xi}{p} \qquad (2.2)$$

In figure 2.3 we can see the conditions applied to a Bennett linkage.
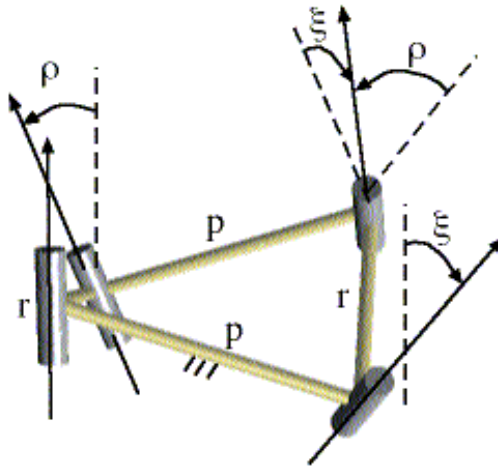
Figure 2. 3. Opposite sides of the Bennett linkage have same lengths
and opposite twist angles.

2.3.2. Properties of the Bennett linkage

The geometry of the Bennett linkage makes it a singular mechanism with interesting properties.

As we saw in equation 2.1, it is allowed to move only under a specific geometric conditions. The Bennett linkage is then overconstrained. That has some interest by itself; overconstrained mechanisms are stiffer than the regular spatial linkages, and capable of sustaining larger loads without the loss of accuracy. Also the Bennett mechanism can be folded, what may be of interest in certain applications.

The Bennett linkage is the base for bigger overconstrained mechanisms, the 6R linkages, that find application in different areas, as fluid mixing machines or aircraft land gears. Different types of 6R linkages, as Goldberg's linkages, Waldron's linkages, or

Wohlhart's linkages, are constructed combining two or three Bennett linkages, and inherit the linear properties of them in their finite displacements, as described in Huang & Sun (1998).

2.2.3. The angular relations for a Bennett linkage

The input/output angular relation for a Bennett linkage is derived in Hunt (1978) and can be easily found as follows. Consider the reference frame showed in figure 2.4.
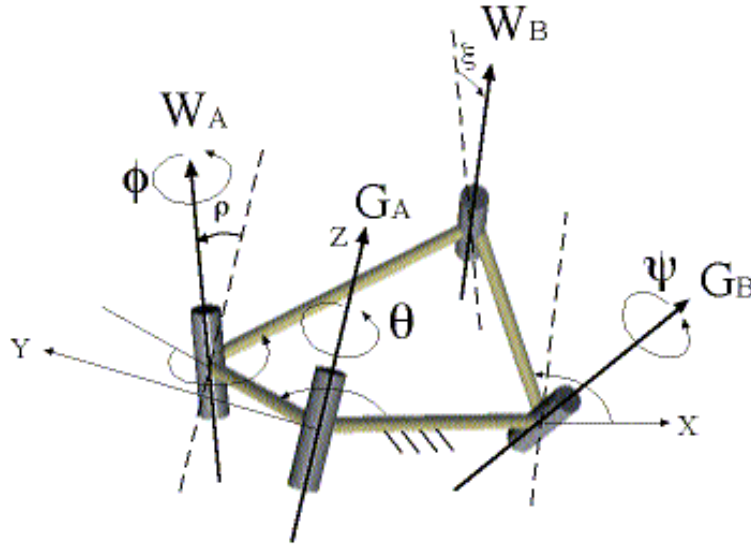
Figure 2. 4. The input angle θ, the output angle ψ, and the coupler φ

The link joining G_A and G_B is considered fixed to the reference frame. The angle θ is the input angle; it is measured from the fixed link. The angle φ about the rotation axis W_A is usually called the coupler. The output angle is ψ and it is associated to the axis G_B.

To find the relation between the output and input angle, we compute the length and orientation of the coupler joining W_A and W_B. This gives us the equation:

$$1 = \frac{p}{r}(\cos\theta - \cos\psi) + (\cos\theta\cos\psi + \sin\theta\sin\psi\cos\xi) \tag{2.3}$$

To eliminate the intermediate angle φ we use the Bennett relation of equation 2.2 and some trigonometric identities to yield:

12

$$\frac{\tan\dfrac{\psi}{2}}{\tan\dfrac{\theta}{2}} + \frac{\sin\dfrac{\xi+\rho}{2}}{\sin\dfrac{\xi-\rho}{2}} = 0 \tag{2.4}$$

An equivalent analysis using the axes W_A and G_A yields the relation between the coupler angle ϕ and the input angle θ:

$$\frac{\tan\dfrac{\phi}{2}}{\tan\dfrac{\theta}{2}} + \frac{\sin\dfrac{\xi+\rho}{2}}{\sin\dfrac{\xi-\rho}{2}} = 0 \tag{2.5}$$

This description and characterization of the Bennett linkage will be used in the following sections to relate pairs of RR chains linked together.

2.4. The Constraint Manifold of an RR Chain

When the direct kinematic equations of a spatial chain are written in dual quaternion form, they define a surface in the image space that is called the constraint manifold. It represents the positions and orientations that the chain can reach, and it is defined by the geometrical constraints imposed on the linkage.

Dual quaternions describe general transformations in space and are equivalent to the matrix description T=[A,d]. It can be proved (see Appendix 2) that the dual quaternion multiplication gives an equivalent formulation to that in matrix form for the kinematic equations.

Consider now the RR chain in Figure 2.5. The fixed axis is denoted by G and the moving axis is denoted by W.
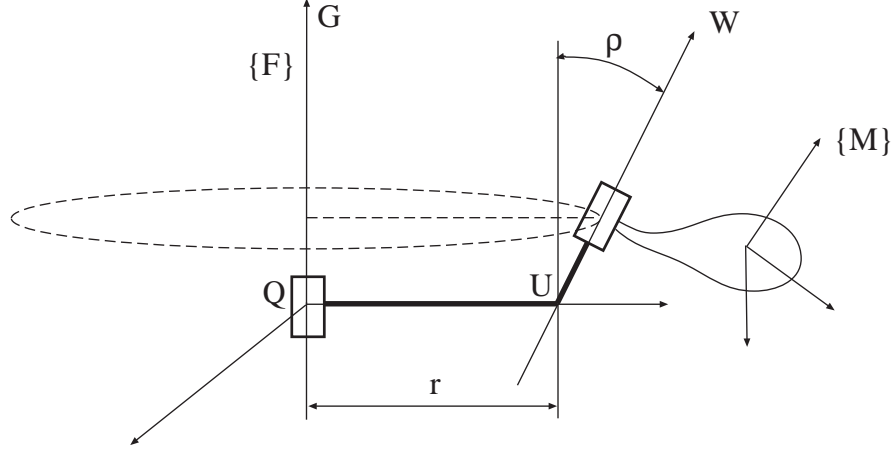
Figure 2. 5 Schematic view of the RR chain

The axis G is fixed with respect to the fixed frame {F} and the axis W is fixed in the moving frame {M}. The points Q and U are points on the lines defined by G and W respectively.

We define the fixed axis and moving axes using the Plucker coordinates as:

$$\hat{G} = G + \varepsilon(Q \times G)$$
$$\hat{W} = W + \varepsilon(U \times W) \qquad (2.6)$$

where the symbol ε is the dual unit, which has the property that it square to zero, see McCarthy (1990).

We now define the dual quaternions that represent the rotation about G by an angle θ and about W by an angle ϕ as

$$QG = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\hat{G}$$
$$QW = \cos\frac{\phi}{2} + \sin\frac{\phi}{2}\hat{W} \qquad (2.7)$$

Quaternion multiplication of QG and QB yields the resultant displacement screw as the dual quaterion:

$$(\cos\frac{\hat{\gamma}}{2} + \sin\frac{\hat{\gamma}}{2}\hat{C}) = (\cos\frac{\theta}{2} + \sin\frac{\theta}{2}\hat{G})(\cos\frac{\phi}{2} + \sin\frac{\phi}{2}\hat{W}) \qquad (2.8)$$

14

where the Plucker coordinates for the resultant screw axis are:

$$\hat{C} = C + \varepsilon(c \times C) \tag{2.9}$$

and the rotation angle γ about this axis and the translation g along it, are defined by:

$$\cos\frac{\hat{\gamma}}{2} = \cos\frac{\gamma}{2} - \varepsilon\frac{g}{2}\sin\frac{\gamma}{2}$$
$$\sin\frac{\hat{\gamma}}{2} = \sin\frac{\gamma}{2} + \varepsilon\frac{g}{2}\cos\frac{\gamma}{2} \tag{2.10}$$

For G and W the associated dual number is just a real number, due to the fact that they are revolute joints.

Expanding equation 2.8 we obtain equations 2.11 and 2.12, which express the dual number and dual vector relations:

$$\cos\frac{\hat{\gamma}}{2} = \cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\hat{G}\cdot\hat{W} \tag{2.11}$$

$$\sin\frac{\hat{\gamma}}{2}\hat{C} = \cos\frac{\theta}{2}\sin\frac{\phi}{2}\hat{W} + \sin\frac{\theta}{2}\cos\frac{\phi}{2}\hat{G} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}\hat{G}\times\hat{W} \tag{2.12}$$

All possible screws defining displacements for the RR chain will be given by the screw axes C with rotation and translation γ and g. The set of displacement screws generated for all values θ and ϕ forms the constraint manifold of the RR chain. The set of axes C given by equation was defined by Hernandez as the *screw surface* of the chain.

2.5. Constraint Manifold of a Bennett Mechanism

2.5.1. The Bennett linkage as two RR dyads

The Bennett mechanism can be seen as two RR chains linked together. Consider figure 2.4. We define the RR chains formed by G_A, W_A and G_B, W_B.

The link joining both fixed axes is fixed itself; the coupler will be the angle of the moving axis, and now we have a relation between the values of rotation about the fixed

axis and the rotation about the moving axis. Both angles can not vary freely, but they are related and this will influence the constraint manifold that we obtain.

2.5.2. The constraint manifold of the coupler

The expression of the constraint manifold is as in equation 2.8; but now the angles are related by equation 2.5.

Huang (1996) showed that the screw axes that we obtain in varying the angle θ of the fixed axis form a cubic ruled surface called cylindroid. He obtained this result as the intersection of the constraint manifolds of two RR chains, since the movement of the Bennett linkage must be compatible with its two RR dyads. In figure 2.6. we can see the aspect of the cylindroid.
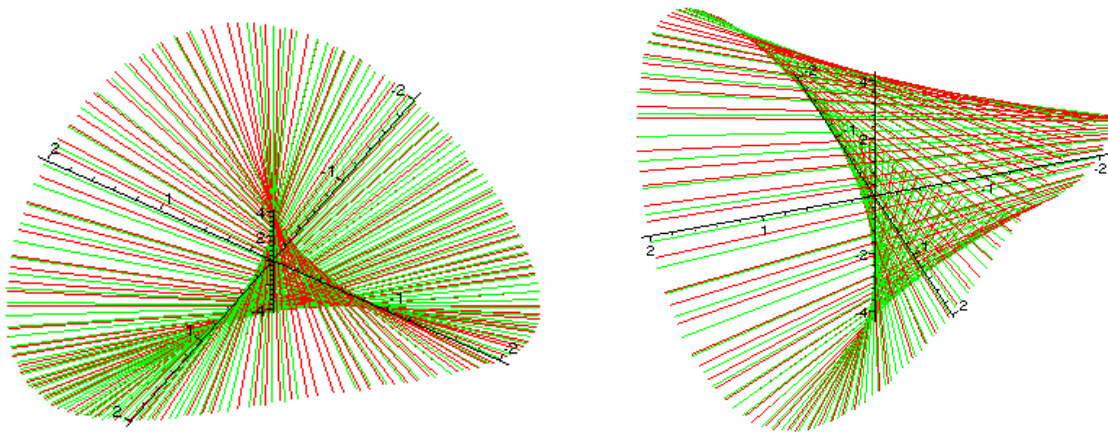
Figure 2. 6. The cylindroid is generated as the constraint manifold
of the Bennett linkage

2.6. Summary

In this chapter we performed the analysis of the RR chain and defined and analyzed the Bennett linkage, stating the relation between both chains.

The constraint manifold of the RR chain is generated by the rotations about the moving and fixed axes with angles ϕ and θ respectively; for a general RR chain, no relation exists between both angles.

The coupler of a Bennett linkage can be seen as an RR chain in which a condition exists to relate the input and the coupler angle. We derived that condition as a function of the geometry of the linkage. Using this relation, the constraint manifold of the coupler of the Bennett linkage was defined as a particular case of the constraint manifold of the RR chain, when instead of two free angles we have only one. Previous results allowed us to identify the constraint manifold of the Bennett linkage as a cubic ruled surface called *cylindroid*.

Chapter 3

Screw Systems of Dimension 2

3.1. Introduction

A screw system is spanned by the linear combination of n independent screws. It can be proved that the screw system is a vector subspace for the dual vectors over the field of the real numbers.

The screw systems are classified according to the number of independent generator screws.   The one-system consists of all multiples of a given screw, and it is immediate to see that it contains all screws with same direction and position but different magnitude.

The two-system of screws contains every linear combination of two independent screws. Along this chapter we will see that the two-system is the locus of all possible screw axes for a Bennett mechanism, and hence it is the case that we will study in detail.

3.2. The 2-screw System

Consider the two independent general screws:

$$S_A = a(1 + \varepsilon P_A)\hat{s}_A = a(1 + \varepsilon P_A)(s_A + \varepsilon(q_A \times s_A))$$
$$S_B = b(1 + \varepsilon P_B)\hat{s}_B = b(1 + \varepsilon P_B)(s_B + \varepsilon(q_B \times s_B)) \tag{3.1}$$

where s_A and s_B are the unit direction vectors along the line, q_A and q_B are points on S_A and S_B, P_A and P_B are the pitches, and a, b are the magnitudes of both screws. The screws of the 2-system, of magnitude F and screw axis C, are a combination of S_A and S_B as we observe in the following equation:

$$F(1 + \varepsilon P)C = a(1 + \varepsilon P_A)\hat{s}_A + b(1 + \varepsilon P_B)\hat{s}_B \tag{3.2}$$

where a and b take values on the real numbers.

The set of lines generated as a linear combination of S_A and S_B forms the cubic ruled surface called the cylindroid (Hunt (1978)). All the screws of the system will intersect the line perpendicular to S_A and S_B at right angles. The Figure 3.1 shows the cylindroid.
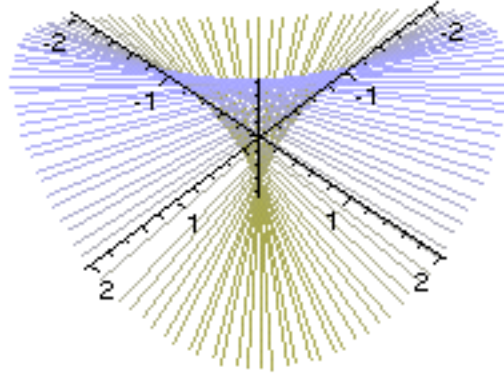
Figure 3. 1: The linear combination of two screws
generates a cylindroid

3.3. Characterization of the Cylindroid

3.3.1. Relation among orientation angle, distance and pitch for the screws

In order to characterize the cylindroid we will assign coordinates to the screws. Let {X,Y,Z} a reference frame. The generator screw axes of S_A and S_B, and the resulting screw axis C can be expressed in the reference frame as:

$$\hat{s}_A = \cos\hat{\theta}_A I + \sin\hat{\theta}_A J$$
$$\hat{s}_B = \cos\hat{\theta}_B I + \sin\hat{\theta}_B J \qquad (3.3)$$
$$C = \cos\hat{\theta} I + \sin\hat{\theta} J$$

where I is the line along the X-axis and J is the line along the Y-axis. The axis Z is chosen so that it is the perpendicular line to S_A and S_B, and the dual numbers are defined as:

19

$$\cos\hat{\theta} = \cos\theta - \varepsilon z \sin\theta$$
$$\sin\hat{\theta} = \sin\theta + \varepsilon z \cos\theta$$

(3.4)

Substituting the values of the expression 3.4 into 3.2 and separating the components of the real and dual parts, we obtain the system of equations:

$$F(1+\varepsilon P)(\cos\hat{\theta}I + \sin\hat{\theta}J) = a(1+\varepsilon P_A)(\cos\hat{\theta}_A I + \sin\hat{\theta}_A J) + b(1+\varepsilon P_B)(\cos\hat{\theta}_B I + \sin\hat{\theta}_B J)$$

$$\begin{cases} F\cos\theta = a\cos\theta_A + b\cos\theta_B \\ F\sin\theta = a\sin\theta_A + b\sin\theta_B \\ F(P\cos\theta - z\sin\theta) = a(P_A\cos\theta_A - z_A\sin\theta_A) + b(P_B\cos\theta_B - z_B\sin\theta_B) \\ F(P\sin\theta + z\cos\theta) = a(P_A\sin\theta_A + z_A\cos\theta_A) + b(P_B\sin\theta_B + z_B\cos\theta_B) \end{cases}$$

(3.5)

We can arrange these equations as two systems. Using the first two equations, we obtain for the real part:

$$F\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \begin{bmatrix} \cos\theta_A & \cos\theta_B \\ \sin\theta_A & \sin\theta_B \end{bmatrix} \cdot \begin{bmatrix} a \\ b \end{bmatrix}$$

(3.6)

and the last two equations, corresponding to the dual part, can be combined to yield:

$$F\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} P \\ z \end{bmatrix} = \begin{bmatrix} P_A\cos\theta_A - z_A\sin\theta_A & P_B\cos\theta_B - z_B\sin\theta_B \\ P_A\sin\theta_A + z_A\cos\theta_A & P_B\sin\theta_B + z_B\cos\theta_B \end{bmatrix} \cdot \begin{bmatrix} a \\ b \end{bmatrix}$$

(3.7)

The solution of the first system gives us the ratio of magnitude for the screws, a/b or a/F and b/F:

$$\frac{a}{F} = \frac{\sin(\theta_B - \theta)}{\sin(\theta_B - \theta_A)}$$
$$\frac{b}{F} = \frac{\sin(\theta - \theta_A)}{\sin(\theta_B - \theta_A)}$$

(3.8)

For the second system, we can substitute in equation 3.7 the value of the magnitudes given by equation 3.8 and solve for the pitch and distance of the resultant screw: The result of this operation yields:

$$F\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} P \\ z \end{bmatrix} = \frac{F}{\sin(\theta_B - \theta_A)}\begin{bmatrix} P_A\cos\theta_A - z_A\sin\theta_A & P_B\cos\theta_B - z_B\sin\theta_B \\ P_A\sin\theta_A + z_A\cos\theta_A & P_B\sin\theta_B + z_B\cos\theta_B \end{bmatrix}\begin{bmatrix} \sin\theta_B & -\cos\theta_B \\ -\sin\theta_A & \cos\theta_A \end{bmatrix}\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \quad (3.9)$$

We can eliminate the magnitude F from this system. Although we can solve this linear system in general, the result is easier to obtain if we move the frame such that the first screw S_a defines the X axis of the frame and Z is again in the perpendicular line to S_A and S_B. The Y axis is define such that we have a direct frame. Notice that now $\theta_A=0$ and $z_A=0$. We also introduce here the notation –that will be used in the general case- for the angle and distance between the screws: we define $\delta=\theta_B-\theta_A$ and $d=z_B-z_A$. With this notation, the linear system in equation 3.9 simplifies to:

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} P \\ z \end{bmatrix} = \frac{1}{\sin\delta}\begin{bmatrix} P_A\sin\delta & (P_B - P_A)\cos\delta - d\sin\delta \\ 0 & P_B\sin\delta + d\cos\delta \end{bmatrix}\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \quad (3.10)$$

And solving in 3.10 for P and z we obtain:

$$\begin{bmatrix} P \\ z \end{bmatrix} = \begin{bmatrix} (P_B + d\cot\delta)\sin^2\theta + ((P_B - P_A)\cot\delta - d)\cos\theta\sin\theta + P_A\cos^2\theta \\ ((P_B - P_A)\cot\delta + d)\sin^2\theta + ((P_B - P_A) + d\cot\delta)\cos\theta\sin\theta \end{bmatrix} \quad (3.11)$$

Equation 3.11 gives us the values of pitch and distance from the origin as a function of the orientation angle and some constant values depending on the initial screws. To simplify this expression, we can introduce the substitutions as in Hunt (1978) defined by:

$$\frac{d - (P_B - P_A)\cot\delta}{2} = A\sin\sigma$$
$$\frac{d\cot\delta + (P_B - P_A)}{2} = A\cos\sigma \quad (3.12)$$

The angle σ introduced in equation 3.12 has a geometrical meaning in the cylindroid. The value $\sigma/2$ gives the orientation of the principal axes of the cylindroid about its common normal. Due to the importance of the principal axes in our design problem, we will devote next sections to define and characterize them.

The expression of σ as a function of the initial screw parameters is given by:

$$\tan\sigma = \frac{d\sin\delta - (P_B - P_A)\cos\delta}{d\cos\delta + (P_B - P_A)\sin\delta} \quad (3.13)$$

21

And the expressions for the pitch and distance are finally, after applying the substitution in 3.12 and some trigonometric identities:

$$P = P_A + A\cos\sigma - A\cos(2\theta - \sigma)$$
$$z = A\sin\sigma + A\sin(2\theta - \sigma)$$

(3.14)

Or transforming equation 3.13 back to the initial and more general reference frame,

$$P = P_A + A\cos\sigma - A\cos(2(\theta - \theta_A) - \sigma)$$
$$z = z_A + A\sin\sigma + A\sin(2(\theta - \theta_A) - \sigma)$$

(3.15)

3.3.2. The distribution of pitches

We observe in equation 3.15 that both P and z are sinusoidal functions of the angle θ. The maximum and minimum values for the pitch and distance are given by the following expressions:

$$P_{max} = P_A + A(\cos\sigma + 1)$$
$$P_{min} = P_A + A(\cos\sigma - 1)$$
$$z_{max} = z_A + A(\sin\sigma + 1)$$
$$z_{min} = z_A + A(\sin\sigma - 1)$$

(3.16)

and from them we can find that the sum and difference of the extrema pitches are two invariants of the cylindroid, because they only depend of the parameters of the generator screws:

$$P_{max} + P_{min} = (P_A + P_B) + d\cot\delta$$
$$P_{max} - P_{min} = \frac{\sqrt{(P_B - P_A)^2 + d^2}}{\sin\delta}$$

(3.17)

We can go further in the characterization of the cylindroid based on pitch and distance. If we combine the equations for pitch and distance in 3.15, we obtain the expression:

$$(P - P_A)^2 + (z - z_A)^2 = 2A^2$$

(3.18)

which shows that there are couples of screws with same value of pitch but located at different distances. There are one single value of distance only for the screws with P_{max} and P_{min} and one single screw only at a distances z_{max} and z_{min}.

22

3.3.3. The principal axes of the cylindroid

In this section we will characterize the principal axes of the cylindroid. The principal axes are the result of the eigenvalue problem for the system. It can be seen that the principal axes are located in the middle of the cylindroid and that they are the only screws located at right angles one each other. The principal axes are also the screws with maximum and minimum pitch. All these characteristics make them very useful as a reference frame to define the cylindroid itself.

Previous to find the principal axes we observe that we have two screws for each value of z. From equation 3.15 we can solve for θ to yield:

$$\theta = \theta_A + \frac{1}{2}(\sigma + \arcsin(\frac{z - z_A}{A} - \sin\sigma)) \tag{3.19}$$

if we note the arc sine as α, we see that the two solutions are given by α and π-α, hence we have two values of orientation for each value of distance z. The two angles will differ 90 degrees only when the arcsine is either 0 or π. For this we need:

$$z_0 = z_A + A\sin\sigma = \frac{1}{2}\left[(z_A + z_B) - (P_B - P_A)\cot\delta\right] \tag{3.20}$$

And we see easily that z_0 is located in the middle of the cylindroid, as:

$$\frac{z_{max} + z_{min}}{2} = z_A + A\sin\sigma = z_0 \tag{3.21}$$

Notice that the magnitude z_0 locates the principal axes along the z-axis with respect to the original reference frame. The magnitudes z_0 and σ suffice to characterize the principal axes for the cylindroid.

We can also compute the value of z corresponding to the maximum and minimum pitches: P_{max} corresponds to $2(\theta-\theta_A)-\sigma=0, \pi$, and that gives us also the value of z_0. The following figure shows the location of the principal axes in the cylindroid.
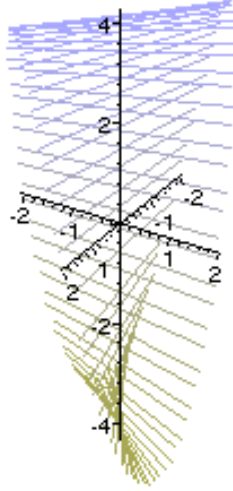
Figure 3. 2. The principal axes are located in the middle of
the cylindroid at right angles

When the principal axes are chosen as the reference frame, the cylindroid appears easier to describe. From the equations for the pitch and distance, if we substitute $z_A=z_B=0$, $\theta_A=0$, $\theta_B=90°$, $\sigma=0$, we obtain the relation between distance and angle:

$$z = \frac{1}{2}(P_\beta - P_\alpha)\sin 2\theta$$

(3.22)

and for two screws of the cylindroid it will hold:

$$\frac{z_1}{z_2} = \frac{\sin 2\theta_1}{\sin 2\theta_2}$$

(3.23)

expression that we can use to characterize the cylindroid, once we know its principal axes.

It is also possible to find the algebraic equation for the cylindroid making substitution for the angles, as:

$$z(x^2 + y^2) - (P_\alpha - P_\alpha)xy = 0 \tag{3.24}$$

The principal axes can be also derived directly from the linear system of equations as the solution of the eigenvalue problem. This derivation follows the approach of Ian A. Parkin (1997).

We start with the equation of the linear combination of the initial screws. In particular we can describe the principal axes as:

$$HX = a_X S_A + b_X S_B$$
$$HY = a_Y S_A + b_Y S_B \tag{3.25}$$

Or, in matrix terms:

$$[HX \quad HY] = [S_A \quad S_B]\cdot\begin{bmatrix} a_X & a_Y \\ b_X & b_Y \end{bmatrix} \tag{3.26}$$

We call K_X the vector of coefficients corresponding to HX, and K_Y for HY.

The principal axes will be located at right angles one each other and they will intersect. The dual dot product of HX and HY can be expressed using this knowledge and separating real and dual part as:

$$\begin{bmatrix} HX^T \\ HY^T \end{bmatrix}\cdot[HX \quad HY] = \tag{3.27}$$

$$\begin{cases} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} K_X^T \\ K_Y^T \end{bmatrix}\begin{bmatrix} s_A^2 & s_A s_B \\ s_A s_B & s_B^2 \end{bmatrix}\cdot[K_X \quad K_Y] \\ \begin{bmatrix} 2P_X & 0 \\ 0 & 2P_Y \end{bmatrix} = \begin{bmatrix} K_X^T \\ K_Y^T \end{bmatrix}\cdot\begin{bmatrix} 2s_A w_A & s_A w_B + s_B w_A \\ s_A w_B + s_B w_A & 2s_B w_B \end{bmatrix}\cdot[K_X \quad K_Y] \end{cases}$$

And combining these two systems for K_X and K_Y separately, we arrive to the final expression, where $2P_X$ is the eigenvalue for HX and $2P_Y$ is the eigenvalue for HY:

$$[K_X \quad K_Y] \cdot \begin{bmatrix} 2P_X & 0 \\ 0 & 2P_Y \end{bmatrix} = \begin{bmatrix} s_A{}^2 & s_A s_B \\ s_A s_B & s_B{}^2 \end{bmatrix}^{-1} \cdot \begin{bmatrix} 2s_A w_A & s_A w_B + s_B w_A \\ s_A w_B + s_B w_A & 2s_B w_B \end{bmatrix} \cdot [K_X \quad K_Y] \qquad (3.28)$$

Expanding this system and solving for the eigenvalues we arrive to the same expression of the pitches for the principal axes, equivalent to equation 3.17. The value of the principal axes can be given as a ratio of the coefficients a_X/b_X.

3.4. Summary

Along this chapter we defined and characterized the cylindroid. The cylindroid, which first appeared as the constraint manifold of the Bennett linkage, turns out to be the locus of the linear combination of two arbitrary screws. This relates the design of Bennett mechanisms and the 2-position synthesis of RR chains, defined by the two screws of the relative displacements.

The principal axes of the cylindroid are two screw axes located in the center of the cylindroid at right angles, and they appeared as the solution of the eigenvalue problem for the cylindroid. These two axes and the common normal to all screw axes form a reference frame that simplifies the expression of the cylindroid and will be useful in further chapters. The principal axes were defined by an angle $\sigma/2$ and a distance z_0 along the common normal; the formulas for these parameters as a function of the initial screws were found.

Chapter 4

Design of RR Chains

4.1. Introduction

For the kinematic design of RR dyads, we will focuse in the three position synthesis, following the previous results by Tsai and Roth (1973). They used the equivalent screw triangle formulation to obtain a set of 4(n-1) equations plus two extra conditions. After some analytic work they reduced the equations to two cubic polynomials, which had only two meaningful solutions. They stated that the two solutions can be combined to form a Bennett mechanism.

Figure 4.1. shows the basic RR chain. Although in the figure we can see the Z-axis coincident with the fixed axis G, the initial data will be given in some different coordinate frame and we will need to locate the point Q on G with respect to those axes.

Figure 4. 1. The direction and position of G and W define
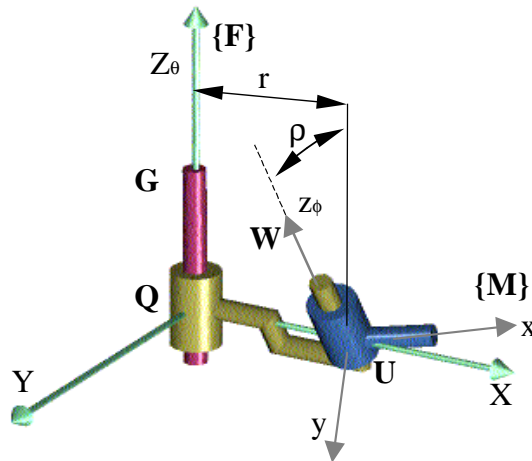
In this chapter we will find the equations for the design of RR chains. The knowledge of the constraint manifold will help us to simplify considerably the design problem with respect to previous design procedures.

4.2. The Statement of the Problem

4.2.1. Specifying the task

Consider we are given three arbitrary positions (location + orientation) in space, say P^1, P^2, P^3. The standard description of these positions is in matrix form, specifying a coordinate transformation from the fixed frame –attached to the fixed axis- to the moving frame –attached itself to the moving axis-. We also can describe the transformation as a screw S_i whose axis is the rotation axis and its magnitude and pitch give the angle of rotation and translation (see Appendix 1).

Without lost of generality we can take the first position as the reference position, and rewrite the positions 2, 3, .. m as relative to 1. In doing so, the initial data for designing the RR chain reduces to the screws of the relative displacements S_{1i}.

4.2.2. Design variables

To design the RR dyad we need to know the location and orientation of both the fixed and moving axes. We use 5 parameters to define a unitary line in space: 2 parameters for the direction of the joint axis (the third coordinate is related to them by the fact the vector is unitary) and 3 coordinates to define one point lying on the line. This sums a total of 10 unknowns for the RR chain.

4.2.3. Constraint conditions

The constraint conditions for the RR chain can be expressed in different ways and lead to equations with different degree of complexity.

Basically we can impose two constraints due to the geometry of the linkage. The first one is showed in equation 4.1. and states that for any position of the mechanism, the distance and orientation between the fixed and moving axes must be hold constant to the values:

$$\hat{G} \cdot \hat{W}_i = \cos\rho - \varepsilon r \sin\rho \quad , i = 2...n \tag{4.1}$$

as defined in Figure 4.1. This constraint is not exclusive of the RR chain, and can be found also when solving the design problem for the spherical RR chain or the spatial CC chain.

Thus the second condition must contain the specific information about the restriction to revolute joints in both axes. As we will see in the following sections, this can be expressed as some orthogonality conditions between G, W and the screw axis of the relative displacement S_{1i}.

4.3. Constraints for 2-position Synthesis

The 2-position synthesis problem determines the set of RR dyads that can reach two given arbitrary positions in space, P^1 and P^2. The positions define a relative displacement $T_{12}=[A_{12},d_{12}]$, or a relative screw S_{12}.

For the RR dyad, the axis G is fixed in the displacement from 1 to 2, while the axis W rotates around it. It can be proved (see Appendix 1) that for two positions of the line W, it holds:

$$G \cdot (W^2 - W^1) = 0 \tag{4.2}$$

where the superscript indicates the position of the moving axis W. In the same way we define the relation:

$$S_{12} \cdot (W^2 - W^1) = 0 \tag{4.3}$$

In order to simplify the calculations, we will define the following frame (see Figure 4.2): S_{12} will define the Z-axis direction, the normal line N to S_{12} and G will define the Y-axis and the X-axis will form a direct frame. To set the position of G is enough to give the values (d,δ) of distance and angle with respect to S_{12} along the common normal line N.
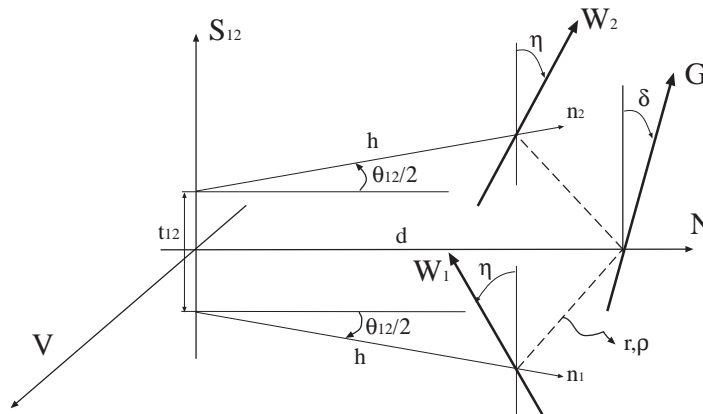
Figure 4. 2. In the displacement S_{12}, W^1 moves to position W^2

In this reference frame, the coordinates of the lines will be:

$$\hat{G} = \cos\hat{\delta}S + \sin\hat{\delta}V = \begin{bmatrix} \sin\delta \\ 0 \\ \cos\delta \end{bmatrix} + \varepsilon \begin{bmatrix} d\cos\delta \\ 0 \\ -d\sin\delta \end{bmatrix} \tag{4.4}$$

$$\hat{W}^1 = \cos\hat{\eta}S + \sin\hat{\eta}(\cos\frac{\hat{\theta}_{12}}{2}V - \sin\frac{\hat{\theta}_{12}}{2}N) = \begin{bmatrix} \sin\eta\cos\frac{\theta_{12}}{2} \\ -\sin\eta\sin\frac{\theta_{12}}{2} \\ \cos\eta \end{bmatrix} + \varepsilon \begin{bmatrix} h\cos\eta\cos\frac{\theta_{12}}{2} - \frac{t_{12}}{2}\sin\eta\sin\frac{\theta_{12}}{2} \\ -h\cos\eta\sin\frac{\theta_{12}}{2} - \frac{t_{12}}{2}\sin\eta\cos\frac{\theta_{12}}{2} \\ -h\sin\eta \end{bmatrix}$$

We can impose the constraints to the problem by looking at its geometry. This constraints, imposed for the two position problem, are valid in general and will be applied later to solve the three position problem.

Consider the plane defined in figure 4.2 by the lines N and the perpendicular line to G and W^1. This plane is perpendicular to G. The dot product of the direction vector G with any point of this plane must be zero. If we compute this dot product for the intersection point of n_1 with W^1 we have the expression:

$$G \cdot X = 0$$

$$(\cos\delta S + \sin\delta V)\cdot(-\frac{t_{12}}{2}S + h\cos\frac{\theta_{12}}{2}N + h\sin\frac{\theta_{12}}{2}V) = 0 \tag{4.5}$$

which yields the condition:

$$h = \frac{\frac{t_{12}}{2}\cos\delta}{\sin\delta\sin\frac{\theta_{12}}{2}} \tag{4.6}$$

Again we can define the planeperpendicular to W^1 and impose the condition of belonging to this plane for the intersection point of N and G, and we obtain:

$$W^1 \cdot (X + \frac{t_{12}}{2}S) = 0$$

$$(\cos\eta S + \sin\eta\cos\frac{\theta_{12}}{2}V - \sin\eta\sin\frac{\theta_{12}}{2}N)\cdot(dN + \frac{t_{12}}{2}S) = 0 \tag{4.7}$$

that can be simplified to:

$$\frac{\cos\eta}{\sin\eta} = \frac{d\sin\frac{\theta_{12}}{2}}{\frac{t_{12}}{2}}$$
(4.8)

In Figure 4.3 we can see the planes which define the constraints of equations 4.6 and 4.8.
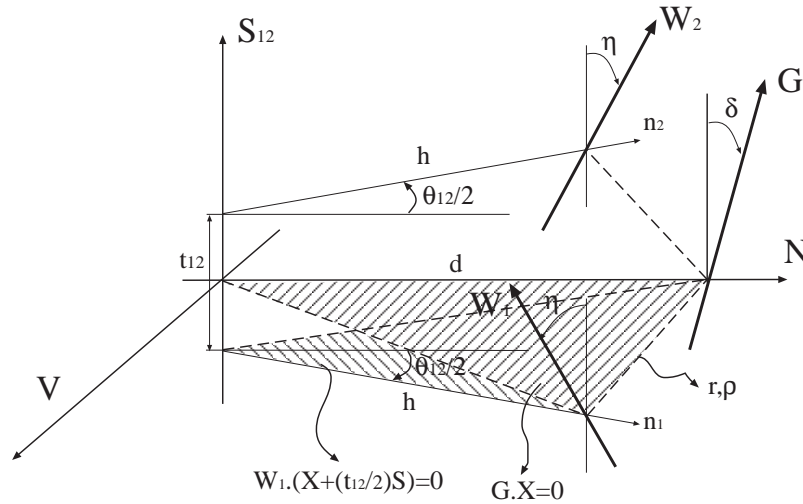
Figure 4. 3. Orthogonality conditions relate G and W to
the screw S_{12}

Observe that the RR chain is totally defined if we know the four parameters d, δ, h, η. We can locate the normal line N arbitrarly around the screw axis due to the symmetry of the problem. Looking at the geometry of the dyad we obtain two equations relating these four magnitudes, what means that there are two free parameters to define in the two-position problem, plus the arbitrarity for the location of N. One interpretation of this is: for the 2-position problem, we can choose the position and orientation for the fixed joint and then solve for the moving joint to get its position and orientation as defined in the figure.

Therefore, the 2-position problem gives a finite solution for one axis, once we define the other. This result agrees with the dimension of the algebraic constraint equations

31

problem. We can identify these parameters as following: define the position for the line G (point, 3 parameters). Define N perpendicular to S_{12} and passing through this point. Now G has to lie in the plane perpendicular to N. One more parameter will define the angle of G with respect to S_{12}.

Finally, we can combine the two conditions to obtain the relation:

$$\frac{d}{\tan \delta} = \frac{h}{\tan \eta}$$

$$(4.9)$$

4.4. The Design Equations

Consider the RR chain of Figure 4.4
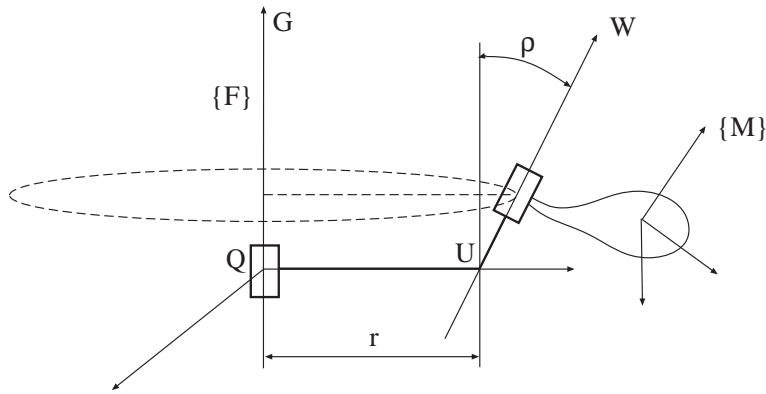
Figure 4. 4. The values of r and ρ must be hold constant
along the movement

4.4.1. The direction equation

The constraint condition in equation 4.1 yields, when we express the moving axes in the moving frame and subtract the first position from the rest,

$$\hat{G}^T \cdot \left(\left[\hat{T}_{1i} - I \right] \hat{W}^1 \right) = (0,0)$$

$$(4.10)$$

We can decompose equation 4.10 in the real and dual part using the Plucker coordinates:

$$\begin{bmatrix} G^T & (Q \times G)^T \end{bmatrix} \cdot \left(\begin{bmatrix} A_{1i} - I & 0 \\ DA_{1i} & A_{1i} - I \end{bmatrix} \begin{bmatrix} W^1 \\ U \times W^1 \end{bmatrix} \right) = (0,0)$$

$$(4.11)$$

In the above system the direction equation is:

$$G^T[A_{1i} - I]W^1 = 0 \qquad (4.12)$$

which appears frequently in the synthesis problem. It corresponds to the solution for the spherical RR chain and gives the relation between directions of the joint axes. It gives a set of bilinear equations that we can solve imposing existence of nontrivial solution.

If we use the matrix form of the dual scalar product in equation 4.11 we obtain:

$$\begin{bmatrix} G^T & (Q \times G)^T \end{bmatrix} \cdot \begin{bmatrix} DA_{1i} & A_{1i} - I \\ A_{1i} - I & 0 \end{bmatrix} \cdot \begin{bmatrix} W^1 \\ U \times W^1 \end{bmatrix} = 0 \qquad (4.13)$$

We will not use this form of the constraint, but rather we will impose some other constraints that can also be derived directly from the geometry of the problem.

4.4.2. The position and rotation equations

The next constraint equations are defined as follows: consider an arbitrary point c_{1i} on the screw axis of the displacement in figure 4.3. Form the line joining the point c_{1i} and the point on the fixed axis Q. If we subtract the component in the direction S_{1i}, the line is perpendicular to S_{1i}. Now we impose it must be perpendicular to the direction vector G. If we do the same for W, we obtain the equations:

$$G^T[I - S_{1i}S_{1i}{}^T](Q - c_{1i}) = 0$$
$$W^T[I - S_{1i}S_{1i}{}^T](U - c_{1i}) = 0 \qquad (4.14)$$

which basically contain the information about the restriction to rotational motion; the points Q and U must be fixed in the fixed and moving frame respectively.

The last conditions will be derived from the geometry of the movement about the screw axis of the displacement. The translation along S_{1i} is defined as t_{1i}. As the axis G is fixed during the movement, the moving axis W must be located as showing in figure 4.3. in its initial position. That can be expressed as:

$$\frac{t_{1i}}{2} = S_{1i}{}^T \cdot (Q - U) \qquad (4.15)$$

33

In order to simplify the results, we impose the additional condition that the points Q, U will be chosen on the perpendicular line to G and W. This adds the two extra conditions:

$$G^T(Q-U) = 0$$
$$W^T(Q-U) = 0$$

(4.16)

4.4.3. Summary of the equations

Summarizing our results, we have 4(n-1)+2 equations:

$$\begin{cases} G^T[A_{1i} - I]W = 0 \\ G^T[I - S_{1i}S_{1i}{}^T](Q - c_{1i}) = 0 \\ W^T[I - S_{1i}S_{1i}{}^T](U - c_{1i}) = 0 \\ \dfrac{t_{1i}}{2} = S_{1i}{}^T \cdot (Q - U) \end{cases}$$
$$G^T(Q-U) = 0$$
$$W^T(Q-U) = 0$$

(4.17)

Recall that the number of unknowns was equal to 10. Hence, for n=3 positions we obtain 10 quadratic equations in 10 unknowns. We can find a finite number of solutions; in the general case, this can be so high as 2^{10} solutions.

However, we know that there is only 2 meaningful solutions plus a little set of trivial solutions. Due to the symmetry of the problem, the number of solutions must reduce drastically. In the following sections we will formulate the constraint equations in a new coordinate frame that allows the simplification of this result.

4.5. Yu's Coordinates

In 1981, Yu introduced a new reference frame for the Bennett linkage. He used the known fact that the four joints of the Bennett linkage pass through the vertices of a tetrahedron. In Figure 4.5 we can see the Bennett linkage attached to the tetrahedron.
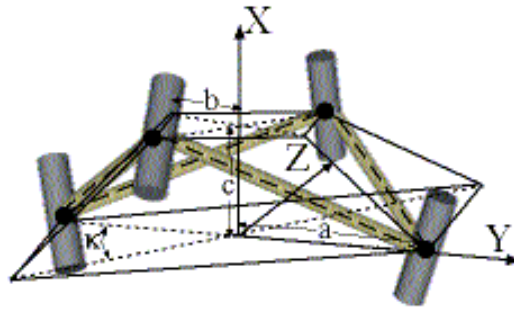
Figure 4. 5. The joints of a Bennett linkage pass through
the vertices of a tetrahedron

The origin of the coordinate frame is located on the base of the tetrahedron, and the directions for X, Y, Z are as shown in Figure 4.5. Notice that the tetrahedron is completely defined by specifying the four parameters c (height of the tetrahedron), a (half the length of the main diagonal), b(half the length of the upper diagonal), and the angle kappa between the diagonals for the base.

4.6. Modified Yu's Coordinates

We introduce a new reference frame based on Yu's Coordinates, that we will denote Modified Yu's Coordinates. This new frame is also based on the tetrahedron, but, as can be seen in Figure 4.6, the origin of the axes is now moved to the center of the tetrahedron with the Z-axis in the vertical direction.
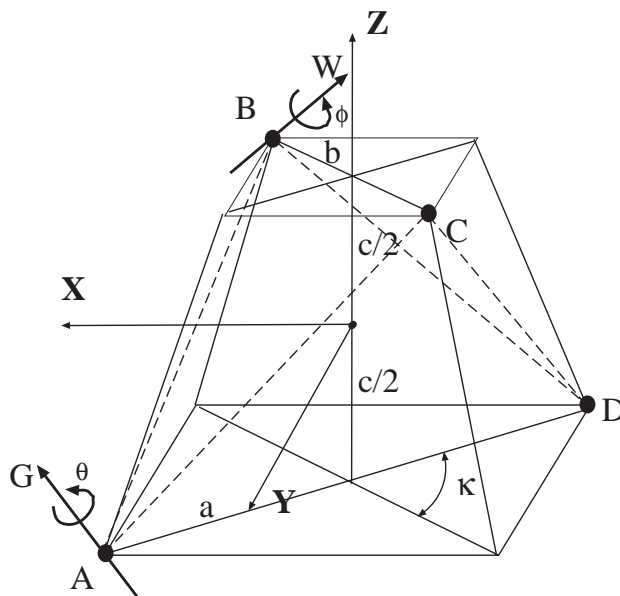
Figure 4. 6. In the Modified Yu's Coordinates, the origin of the
reference frame is located in the center of the tetrahedron

35

The X-axis and Y-axis pass through the center of the faces of the tetrahedron. The points A, B, C, D are points on the joint axes.

With this formulation, the joint axis directions can be easily found as just the cross product of adjacent links. Doing so, the direction G would be the cross product of the vector AC cross with the vector AB.

To solve the design equations expressed in this coordinate frame is especially simple because, as we are going to see, it matches the principal axes frame of the cylindroid generated by the movement of the Bennett linkage.

The expressions of the fixed and moving axes in this new reference frame are:

$$
\hat{G} = \begin{bmatrix} 2bc\sin\dfrac{\kappa}{2} \\[2ex] 2bc\cos\dfrac{\kappa}{2} \\[2ex] 4ab\cos\dfrac{\kappa}{2}\sin\dfrac{\kappa}{2} \end{bmatrix} + \varepsilon \begin{bmatrix} b\cos\dfrac{\kappa}{2}(4a^2\sin^2\dfrac{\kappa}{2}+c^2) \\[2ex] -b\sin\dfrac{\kappa}{2}(4a^2\cos^2\dfrac{\kappa}{2}+c^2) \\[2ex] 2abc(\cos^2\dfrac{\kappa}{2}-\sin^2\dfrac{\kappa}{2}) \end{bmatrix} \tag{4.18}
$$

$$
\hat{W} = \begin{bmatrix} -2ac\sin\dfrac{\kappa}{2} \\[2ex] 2ac\cos\dfrac{\kappa}{2} \\[2ex] 4ab\cos\dfrac{\kappa}{2}\sin\dfrac{\kappa}{2} \end{bmatrix} + \varepsilon \begin{bmatrix} -a\cos\dfrac{\kappa}{2}(4b^2\sin^2\dfrac{\kappa}{2}+c^2) \\[2ex] -a\sin\dfrac{\kappa}{2}(4b^2\cos^2\dfrac{\kappa}{2}+c^2) \\[2ex] 2abc(\cos^2\dfrac{\kappa}{2}-\sin^2\dfrac{\kappa}{2}) \end{bmatrix} \tag{4.19}
$$

It is important to point out here that the expression of the RR chain in these coordinates is going to simplify considerably the problem. The reason for this is that the new coordinates contain already part of the information of the constraints that we are willing to impose.

As the Modified Yu's coordinates coincide with the principal axes of its cylindroid, the Z-axis will be perpendicular to all screws of the cylindroid. Then we can locate our two screws S_{12} and S_{13} in some position intersecting Z at right angles, let say at a distance d_i and an angle δ_i. The points E_1 and F_1 in figure 4.7 are the intersection of the screw S_{12} with the common normal to the lines G and W respectively. Same can be drawn for S_{13} to obtain the points E_2 and F_2, but was not included in the drawing for clarity.
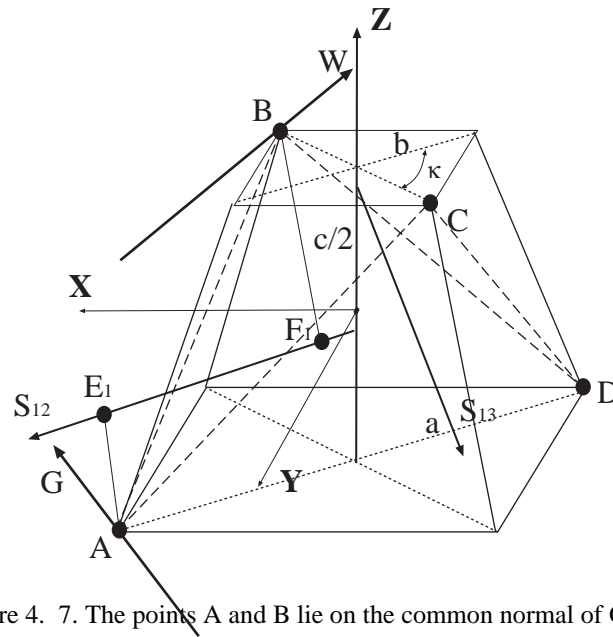
Figure 4. 7. The points A and B lie on the common normal of G and W to S_{12}

We can show that the points A and B lie already in the common normal to G and S_{12}. If we compute the dot product and impose it to be equal to zero, $G.(A-E_1)=0$, we obtain the following condition:

$$\begin{bmatrix} \sin 2\delta_1 & 2d_1 \\ \sin 2\delta_2 & 2d_2 \end{bmatrix} \begin{bmatrix} c \\ \sin\kappa \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \implies \frac{\sin 2\delta_1}{\sin 2\delta_2} = \frac{d_1}{d_2} \tag{4.20}$$

Imposing the existence of nontrivial solutions to this system, we arrive to the above relation between orientation and distance for the screw axes. This formula is identical to the equation 3.23 for the relation between distance and orientation of the cylindroid expressed in the principal axes frame.

4.7. Design Equations in Modified Yu's Coordinates

4.7.1. New formulation of the design equations

With the new coordinate frame, we only need four parameters to define position and orientation of the RR chain. Hence we only need four constraint equations, that is, two constraint equations applied to the screws S_{12} and S_{13}. The rest of the constraints that we found in section 4.4 hold because of the geometrical meaning of the new reference frame.

The non-trivial constraints capture the information that we deduced in the 2-position synthesis about the rotation and traslation along S_{i2}. The angle between the directions of

37

E_1A and F_1B in figure 4.7 must be half of the rotation angle. And the distance between E_1 and F_1 on S_{12} must be half of the translation. Same conditions hold for S_{13} The new formulation yields:

$$\tan\frac{\theta_{12}}{2} = \frac{G \cdot (S_{12} \times W^1)}{(S_{12} \times G) \cdot (S_{12} \times W^1)}$$

$$\tan\frac{\theta_{13}}{2} = \frac{G \cdot (S_{13} \times W^1)}{(S_{13} \times G) \cdot (S_{13} \times W^1)} \qquad (4.21)$$

$$\frac{t_{12}}{2} = S_{12} \cdot (Q - U)$$

$$\frac{t_{13}}{2} = S_{13} \cdot (Q - U) \qquad (4.22)$$

where the first condition can be proved that is equivalent to the direction equation in section 4.4.1.

4.7.2. The equations in Modified Yu's Coordinates

The following equations are just the coordinate substitution in 4.21 and 4.22. A simplified expression is needed in order to use them in a closed form.

The first two equations correspond to equation 4.21, and the last two correspond to 4.22.

$$algeq1 := \tan\left(\frac{1}{2}\theta 12\right) + \left(8\,c\,b^2\sin(\delta 1)\cos\left(\frac{1}{2}\kappa\right)a - 8\cos\left(\frac{1}{2}\kappa\right)^3 c\,b^2\sin(\delta 1)\,a - 8\,c\,b^2\cos\left(\frac{1}{2}\kappa\right)^2\cos(\delta 1)\,a\sin\left(\frac{1}{2}\kappa\right)\right.$$
$$\left. + 8\,b\cos\left(\frac{1}{2}\kappa\right)^2 a^2\sin\left(\frac{1}{2}\kappa\right)\cos(\delta 1)\,c + 8\,b\cos\left(\frac{1}{2}\kappa\right)a^2\sin(\delta 1)\,c - 8\cos\left(\frac{1}{2}\kappa\right)^3 c\sin(\delta 1)\,a^2\,b\right) \Big/ \Big($$
$$\left. -16\,b^2\cos\left(\frac{1}{2}\kappa\right)^2 a^2 + 16\,b^2\cos\left(\frac{1}{2}\kappa\right)^4 a^2 + 4\,c^2\,b\,a - 4\,c^2\,b\,a\cos\left(\frac{1}{2}\kappa\right)^2 - 4\,c^2\,b\,a\cos(\delta 1)^2\right)$$

$$algeq2 := \tan\left(\frac{1}{2}\theta 13\right) + \left(8\,c\,b^2\sin(\delta 2)\cos\left(\frac{1}{2}\kappa\right)a - 8\,c\,b^2\sin(\delta 2)\cos\left(\frac{1}{2}\kappa\right)^3 a - 8\,c\,b^2\cos\left(\frac{1}{2}\kappa\right)^2\cos(\delta 2)\,a\sin\left(\frac{1}{2}\kappa\right)\right.$$
$$\left. + 8\,b\cos\left(\frac{1}{2}\kappa\right)^2 a^2\sin\left(\frac{1}{2}\kappa\right)\cos(\delta 2)\,c + 8\,b\cos\left(\frac{1}{2}\kappa\right)a^2\sin(\delta 2)\,c - 8\,b\cos\left(\frac{1}{2}\kappa\right)^3 a^2\sin(\delta 2)\,c\right) \Big/ \Big($$
$$\left. -16\,b^2\cos\left(\frac{1}{2}\kappa\right)^2 a^2 + 16\,b^2\cos\left(\frac{1}{2}\kappa\right)^4 a^2 + 4\,c^2\,b\,a - 4\,c^2\,b\,a\cos(\delta 2)^2 - 4\,c^2\,b\,a\cos\left(\frac{1}{2}\kappa\right)^2\right)$$

$$algeq3 := \frac{1}{2}t12 - \cos(\delta 1)\left(-b\cos\left(\frac{1}{2}\kappa\right) + a\cos\left(\frac{1}{2}\kappa\right)\right) - \sin(\delta 1)\left(b\sin\left(\frac{1}{2}\kappa\right) + a\sin\left(\frac{1}{2}\kappa\right)\right)$$

$$algeq4 := \frac{1}{2}t13 - \cos(\delta 2)\left(-b\cos\left(\frac{1}{2}\kappa\right) + a\cos\left(\frac{1}{2}\kappa\right)\right) - \sin(\delta 2)\left(b\sin\left(\frac{1}{2}\kappa\right) + a\sin\left(\frac{1}{2}\kappa\right)\right)$$

4.8. Summary

In this chapter we found the equations necessary to design an RR chain; we called them the *design equations*. We saw in the statement of the problem that the general derivation yields 10 quadratic equations in 10 unknowns, being this equations fairly complicated to solve.

To reduce the complexity of the equations, we used the relations between the axes of the RR chain and the screw of the relative displacement. The screw axes S_{12} and S_{13} have simple expressions in the principal axes frame of the cylindroid that they generate. To locate the fixed axis G and moving axis W in that reference frame, we introduce a new coordinate system.

Yu's Coordinates (Yu (1981)) locate the reference frame in the basis of the tetrahedron associated to the Bennett linkage. This gives a particularly simple expression for the location of G and W, with only 4 parameters to determine. We defined the Modified Yu's Coordinates, which use the same tetrahedron structure but now they are coincident with the principal axes frame. We showed that, using this convention, some of the constraints are included in the coordinates and only four equations in four unknowns must be solved.

Chapter 5

The Design Procedure

5.1. Introduction

In this chapter we summarize the different results that we saw along the previous chapters and emphasize the relation between them, to see how they influence the design procedure.

The design procedure will be explained after that and some numerical examples will be provided. The first example is extracted from Tsai and Roth (1973); in this example we will see that our results match with theirs. In both examples we include the plots of the complete designed Bennett linkage and the tetrahedron that it generates.

5.2. The Design Procedure

5.2.1. The cylindroid generated by 3 positions

The screws S_{12} and S_{13} generate a cylindroid. The cylindroid contains all the screw axes of the displacements of the R-R chain given S_{12} and S_{13}. Using the formulas 3.13 and 3.20 we can compute the cylindroid's principal axes. This is a necessary information, because we want to express the initial screws S_{12} and S_{13} in the principal axes reference frame.

5.2. 2. Expression of the R-R chain in Modified Yu's Coordinates

The 10 initial unknowns, necessary to position two axes in space, are reduced to 4 parameters with the use of Modified Yu's Coordinates, shown in figure 4.6. This coordinates coincide with the principal axes of the cylindroid generated in the movement of the associated Bennett linkage. In equations 4.18 and 4.19 we can find the expressions of the lines G and W as a function of the parameters a, b, c, kappa.

5.2.3. Design equations

Having both the screws of the displacements and the unknown joint axes expressed in principal axes coordinates, we can define the geometric relation between them. Out of the initial 10 equations necessary to express all constraints of the problem, the use of modified Yu's coordinates reduces them to 2(n-1) equations (4 equations for 3 positions). The definition of the joint axes G and W as the cross product of successive links, and the position of the axes in the corners of a tetrahedron, forces part of the geometric constraints to be included in the coordinate expressions of G and W.

The final set of design equations are shown in expressions 4.21 and 4.22. Solving for this equations is by now performed by numeric methods, but their simplicity allows us to believe that we can arrive to simple closed algebraic form for them.

This set of equations gives two solutions, that correspond to both R-R dyads of the Bennett linkage. The two solutions are found applying the design equations to both sets of R-R dyads, defined by the corners of the tetrahedron.

5.3. Examples

5.3.1. Tsai and Roth Example

Tsai and Roth define the following input data:

	Direction			Location			Rotation	Translation
ij	S_x	S_y	S_z	c_x	c_y	c_z	θ_{ij}	t_{ij}
12	0	0	1	0	0	0	40°	0.8
13	$\sin 30°$	0	$\cos 30°$	0	1	0	70°	0.6

The solutions were obtained using the Maple V.5 worksheet shown in the Appendix 3. The main results of the procedure are indicated below.

First we compute the parameters of the principal axes:

$$rzx := 1.080710362$$

$$rsigma := 1.114282655$$

The principal axes are located at an angle rsigma/2 and a distance r_{zx} from the original axes, being the Z-axis in the common perpendicular of the screws.

Now we express the screws S_{12} and S_{13} in the principal axes reference frame:

$$rpS12 := \begin{bmatrix} .8487701344 & -.5714385788 \\ -.5287620059 & -.9172746792 \\ 0 & 0 \end{bmatrix}$$

$$rpS13 := \begin{bmatrix} .9994375015 & -.00270672389 \\ -.0335362626 & -.0806649625 \\ 0 & 0 \end{bmatrix}$$

Solving the design equations we obtain the values for the parameters a, b, c, kappa:

$$afinal := -1.220134870$$
$$bfinal := .9271547677$$
$$cfinal := .6453985164$$
$$kappafinal := 3.412931041$$

And the moving and fixed axes in the principal reference frame are:

$$Gnumer := \begin{bmatrix} .8838278702 & -.5853506912 \\ -.1206493541 & -.3598007854 \\ .4519867580 & 1.048569486 \end{bmatrix}$$

$$Wnumer := \begin{bmatrix} .9246409849 & -.3708108723 \\ .1262206607 & .3434393781 \\ .3593151722 & .8335795652 \end{bmatrix}$$

Tsai and Roth give the following results, once in principal axes coordinates:

$$Gw1 := \begin{bmatrix} .8838441454 & -.5853679806 \\ -.1206960508 & -.3598259112 \\ .4520 & 1.04854977 \end{bmatrix} \qquad Gr1 := \begin{bmatrix} .9246439890 & -.3707712872 \\ .1261630037 & .3434271509 \\ .3593 & .83357540 \end{bmatrix}$$

Figure 6.1 shows the 2 dyads and the initial screws, in the principal axes frame.
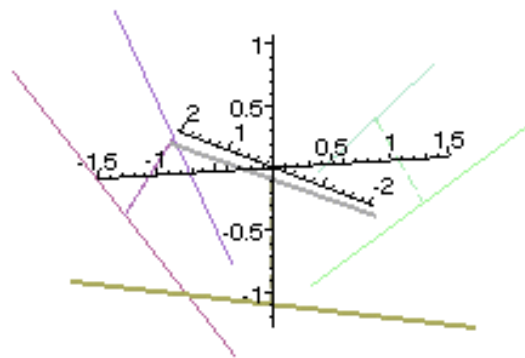
Figure 6.1.

In figure 6.2he we can see more clearly the symmetry of the Bennett linkage designed:
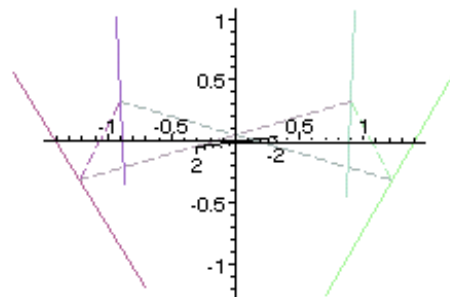
And figure 6.3 shows the l sdyads and the cylindroid of all possible screw axes.

5.3.2. Another examp'˄
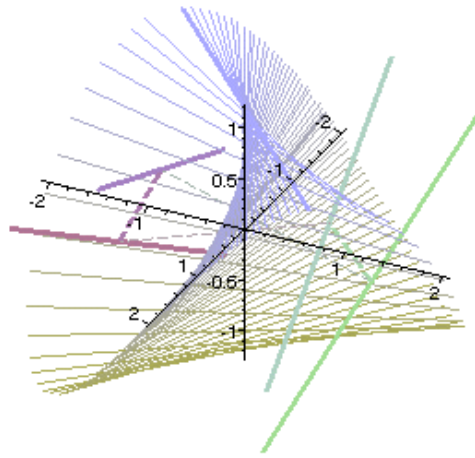
Figure 6. 3

The initial data for the problem is given in the table:

	Direction			Location			Rotation	Translation
ij	S_x	S_y	S_z	c_x	c_y	c_z	θ_{ij}	t_{ij}
12	1	0	0	0	0	0	30°	0.5
13	sin 60°	0	cos 60°	0	0	2.5	90°	0.2

We obtain the values for both esidyads (G-W and H-V):

$$Hnumer := \begin{bmatrix} -.9131683735 & .4201804756 \\ .2245301925 & .5905586297 \\ -.3401613043 & -.7381712014 \end{bmatrix}$$

$$Vnumer := \begin{bmatrix} -.9318927758 & .3261113071 \\ -.2291341557 & -.5774197530 \\ -.2811999160 & -.6102213189 \end{bmatrix}$$

$$Gnumer := \begin{bmatrix} .9318927758 & -.3261113072 \\ .2291341558 & .5774197533 \\ -.2811999161 & -.6102213190 \end{bmatrix}$$

$$Wnumer := \begin{bmatrix} .9131683746 & -.4201804762 \\ -.2245301927 & -.5905586303 \\ -.3401613048 & -.7381712026 \end{bmatrix}$$

With the values for a, b, c, kappa and the principal axes angle (rsigma) and distance (rzx):

bfinal := .8860050887   *rsigma* := 1.368832673
afinal := -.7177139893   *rzx* := 1.490470054
kappafinal := -3.623787928
cfinal := -1.135819936

On figure 6.4 we can see the Bennett mechanism obtained:
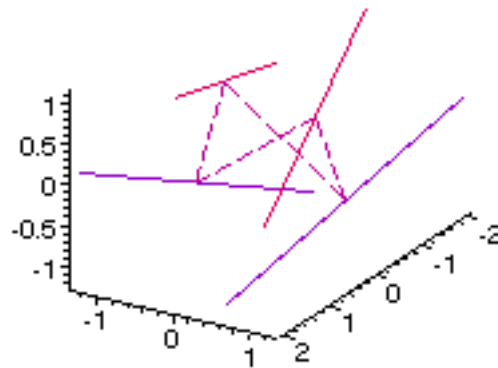
Figure 6. 4
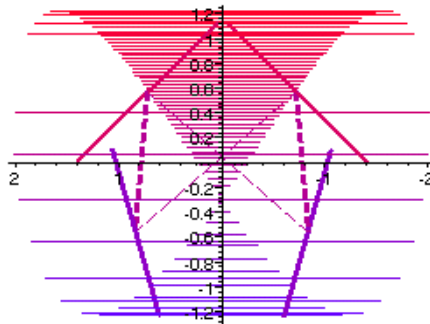
And previous figure shows the mechanism with the cylindroid generated in its displacement.

45

Chapter 6

Conclusions and Future Research

This project develops a new methodology for the design of spatial RR chains which combines the equivalent screw triangle methods of Tsai and Roth (1973) with the geometric properties of the Bennett cylindroid studies by Huang (1996). Tsai and Roth obtained 10 quadratic equations in 10 unknowns for the three position synthesis of a spatial RR chain which they proved have two unique nontrivial solutions that can be combined to form a Bennett linkage. Huang showed that the constraint manifold of a Bennett linkage is a cylindroid. We use the properties of the cylindroid to simplify the formulation of Tsai and Roth's design equations.

In our approach, we recognize that the two relative displacement screws obtained from the three goal position must lie on the cylindroid associated with the Bennett linkage obtained from the solutions of the design equations. We compute the principal axes of the cylindroid generated by these two screws, which must correspond to the cylindroid of the design solution. In this principal axis frame, we introduce a special set of coordinates, related to those used by Yu, that we call "modified Yu's coordinates." These coordinates automatically satisfy the geometric constraints of the Bennett linkage.

Using the modified Yu's coordinates, the design equations for the RR chain become four equations in four unknowns. Numerical solution of these equations yields the desired spatial RR chain. These equations seem to be simpler than the previous results and may provide a convenient algebraic solution to this design problem.

A design procedure for spatial RR chains has been developed and implemented in Maple V.5. Two examples are provided. The first reproduces the results of Tsai and Roth as a check of the computation, and the second example demonstrates the versatility of the design procedure. In order to facilitate the computations required in this design procedure, library functions were developed to perform operations with dual vectors and quaternions, which are listed in Appendices 1 and 2. Worksheets that demonstrate the computations for the analysis and design of RR chains and Bennett linkages are provided in Appendix 3.

Future research will focus on the development of an interactive graphics algorithm for spatial RR chain synthesis together with Bennett linkage synthesis. An interesting theoretical challenge is the reduction of the four design equations to a single equation in a single unknown. This may provide more efficiency in any design algorithm.

References

Craig, John: Introduction to Robotics. Addison-Wesley, 1989

Huang, C.: The Cylindroid Associated with Finite Motions of the Bennett Mechanism. . *Proceedings of the ASME Design Engeineering Technical Conferences*, 1996

Huang, C. and Sun, C.C.: An investigation of screw systems in the finite displacements of Bennett-based 6R linkages. *Proceedings of the ASME Design Engeineering Technical Conferences*, 1998

Hunt, K.H.: Kinematic Geometry of Mechanism. Clarendon Press, 1978

McCarthy, J.M.: Introduction to Theoretical Kinematics. The MIT Press, 1990

McCarthy, J.M.: Spatial Mechanism Design. UCI, 1997

Parkin, Ian A.: Finding the Principal Axes of Screw Systems. *Proceedings of the ASME Design Engeineering Technical Conferences*, 1997

Suh, C.H., Radcliffe, C.W.: Kinematics and Mechanisms Design. John Wiley & Sons, 1978

Tsai, L.W. and Roth, B.: A Note on the Design of Revolte-Revolute Cranks. *Mechanisms and Machine Theory*, 1973, Vol.8, pp 23-31

Tsai, L.W.: Design of Open-Loop Chains for Rigid-Body Guidance. University Microfilms, 1972

Ward, J.P.: Quaternions and Cayley Numbers. Kluwer Academic Publishers, 1997

Appendix 1

Procedures for the Algebra of Dual Vectors

Procedures for the algebra of dual vectors

The following procedures were implemented on the software Maple V5.5 to perform the algebraic operations with dual vectors. The procedures automatize all the common calculations with dual vectors, dual numbers, and their application to kinematics. The following is a description of the operations performed by the procedures. After it we include a list with the procedures.

Definition of dual vectors and dual numbers

Briefly, a geometric vision of dual vectors define them as the element formed by two 3 dimensional vectors, the first one specifying one direction in space and the second one accounting for the "momentum" of the line with respect to the coordinate frame, or more simply for the position of the direction vector in space.

$$S = \vec{v} + \varepsilon \vec{w} = k\vec{s} + \varepsilon(\vec{c} \times k\vec{s} + pk\vec{s})$$

The magnitude of the direction vector v is called the magnitude of the dual vector, and it is related to the rotation about the direction vector. The scalar p is called the pitch, and it is related to the displacement along the direction vector. The vector c is a point on the line defined by the dual vector. The dual vector is called line if the pitch is zero, and screw if it is not zero. The dual vectors can be seen geometrically as lines in space with some values of rotation and displacement along them.

From a more algebraic point of view, we can define the dual vectors as 3-dimensional vectors formed by dual numbers. The dual numbers are ordered pairs of real numbers (a,b) that can be written as a+εb, where $\varepsilon^2=0$. The algebra of the dual numbers has the same properties that the algebra of the 2x2 matrices of the form:

$$A = \begin{bmatrix} a & 0 \\ b & a \end{bmatrix}$$

For the algebraic manipulation of dual vectors in Maple V5.5, we accept three ways to specify the dual vectors. The most common and useful is to define them as 3x2 matrices, each column being one of the component vectors. It is also useful to be able to work with the parameter ε, and we allow the definition of the dual vector as v+Ew, being $E^2=0$.

And finally we can define the dual vectors as a 6x1 vector; in this case, both component vectors are written sequentially.

To simplify the procedures, we convert those formats to the 3x2 matrix format to work internally.

The algebra of the dual vectors

We constructed the algebra of the dual vectors on the matrix algebra. The dual vectors form a vector space over the dual numbers in which we can define the common operations sum, product by scalar, dot product and cross product.

For the addition of dual vectors we use the Maple package linalg. The product of dual numbers and dual vectors follows the natural way for ordered pairs with the condition $\varepsilon^2 = 0$, i.e.

$$(a + \varepsilon b)(\vec{v} + \varepsilon \vec{w}) = a\vec{v} + \varepsilon(a\vec{w} + b\vec{v})$$

The product of two dual numbers follows a similar rule:

$$(a + \varepsilon b)(c + \varepsilon d) = ac + \varepsilon(ad + bc)$$

To compute the inverse of a dual number, recall the equivalence between dual numbers and certain group of 2x2 matrices. Provided a is not equal to zero,

$$(a + \varepsilon b)^{-1} = (\frac{1}{a} + \varepsilon \frac{-b}{a^2})$$

The dot product of two screws gives a scalar dual number. It is defined as the usual component-by-component product. It leads to the following expression as a function of the vector dot product.

$$(\vec{v}_1 + \varepsilon \vec{w}_1) \cdot (\vec{v}_2 + \varepsilon \vec{w}_2) = \vec{v}_1 \cdot \vec{v}_2 + \varepsilon(\vec{v}_1 \cdot \vec{w}_2 + \vec{w}_1 \cdot \vec{v}_2)$$

The cross product can also be defined based on the vector cross product.

$$(\vec{v}_1 + \varepsilon \vec{w}_1) \times (\vec{v}_2 + \varepsilon \vec{w}_2) = \vec{v}_1 \times \vec{v}_2 + \varepsilon(\vec{v}_1 \times \vec{w}_2 + \vec{w}_1 \times \vec{v}_2)$$

Some properties of dual vectors

The specific application of dual vectors to identify geometry in space allows us to define a series of useful properties. Magnitudes such as angle and distance between lines or concepts such as perpendicularity have to be well identified and easy to compute.

Let us compute first the common normal to two general screws. Recall the cross product formula applied to the general screw:

$$
\begin{bmatrix} k_1\vec{s}_1 \\ \vec{c}_1 \times k_1\vec{s}_1 + p_1 k_1\vec{s}_1 \end{bmatrix} \times \begin{bmatrix} k_2\vec{s}_2 \\ \vec{c}_2 \times k_2\vec{s}_2 + p_2 k_2\vec{s}_2 \end{bmatrix} =
$$

$$
= (k_1 + \varepsilon k_1 p_1)(\vec{s}_1 + \varepsilon(\vec{c}_1 \times \vec{s}_1)) \times (k_2 + \varepsilon k_2 p_2)(\vec{s}_2 + \varepsilon(\vec{c}_2 \times \vec{s}_2)) =
$$

$$
= \left(k_1 k_2 + \varepsilon(k_1 k_2 p_1 + k_1 k_2 p_2) \right)\left(\vec{s}_1 \times \vec{s}_2 + \varepsilon\left[\vec{s}_1 \times (\vec{c}_2 \times \vec{s}_2) + (\vec{c}_1 \times \vec{s}_1) \times \vec{s}_2 \right] \right) =
$$

$$
= K_1 K_2 \left(\vec{s}_1 \times \vec{s}_2 + \varepsilon\left[(\vec{s}_1 \cdot \vec{s}_2)(\vec{c}_2 - \vec{c}_1) + (\vec{s}_2 \cdot \vec{c}_1)\vec{s}_1 - (\vec{s}_1 \cdot \vec{c}_2)\vec{s}_2 \right] \right)
$$

The cross product of the unit direction vectors will give the normal direction n multiplied by the sine of the angle between them, δ. The first term under ε is the cosine of the angle δ in the direction of c_2-c_1, that is dn, being d the distance between screws. The last two terms combine after including a new point c as:

$$
S_1 \times S_2 = K_1 K_2 \left(\sin\delta \vec{n} + \varepsilon(d\cos\delta \vec{n} + \sin\delta(\vec{c} \times \vec{n})) \right)
$$

We can simplify the notation if we define the dual number:

$$
\sin\hat{\delta} = \sin\delta + \varepsilon d\cos\delta
$$

Then the equation becomes

$$
S_1 \times S_2 = K_1 K_2 \sin\hat{\delta}(\vec{n} + \varepsilon(\vec{c} \times \vec{n}))
$$

Summarizing, the cross product of dual vectors gives us the normal line to both screws with the dual magnitude containing all information about distance and angle between screws.

The angle and distance between screws can be obtained also by looking at the dot product.

$$S_1 \cdot S_2 = k_1\vec{s}_1 \cdot k_2\vec{s}_2 + \varepsilon((\vec{c}_1 \times k_1\vec{s}_1 + k_1 p_1\vec{s}_1) \cdot \vec{s}_2 + (\vec{c}_2 \times k_2\vec{s}_2 + k_2 p_2\vec{s}_2) \cdot \vec{s}_1) =$$
$$= K_1 K_2 (\cos\delta + \varepsilon(-d\sin\delta))$$

Sometimes is useful to work only with screws such that their dual magnitude K=(1,0). This corresponds to the case of having a unit direction vector (and equivalently, no rotation angle) and no displacement along the line, i.e., pitch equal to zero. In this case, a screw is called a (unit) line and it has this aspect:

$$L = \vec{s} + \varepsilon(\vec{c} \times \vec{s})$$

where the direction vector does not have to be unitary in general. From this formula we can see that for a line, it has to be true that s.(cxs)=0. When we have a screw representing a rotation in space, its corresponding unit line is called the axis of the screw.

Because a screw can represent a general transformation (rotation plus displacement) in space.

In a general case, we can define the screw S_{12} associated with the relative transformation T_{12}. The screw contains all the information about the spatial displacement. It can be viewed as a rotation about the screw axis plus a translation along the direction of the screw axis. The rotation and translation are related to the magnitude and pitch of the screw.

To find the screw we impose the condition that it must be an eigenvector for the transformation matrix. We already know that the rotation matrix has an eigenvalue equal to 1.

$$\left[\hat{T}_{12} - I\right] S = 0$$

Developing this equation we get:

$$\left[A_{12} - I\right] s = 0$$
$$\left[D_{12} A_{12}\right] s - \left[A_{12} - I\right] v = 0$$

Where A_{12} is the rotation matrix and D_{12} is the skew-symmetric matrix defined by the displacement vector d_{12}.

Then the screw axis of the displacement is $S_{12}=(s,cxs)$, with s unit vector and c point on the line defined from the second equation, using Cayley's formula and the fact that s is an eigenvector of A_{12}.

$$c = s \times v = \frac{1}{2\tan\dfrac{\theta_{12}}{2}}(-S + \tan\frac{\theta_{12}}{2}S^2)t_{12}$$

where θ_{12} is the rotation angle and t_{12} is the amount of translation along s.

Under this situation, we can notice a general fact. If x is transformed to X with T_{12}, we can easily see that the following expression is always true:

$$S_{12} \cdot (X - x) = (0,0) = (k,kp)(\cos\alpha, -a\sin\alpha)$$

The axes of all screws X-x intersect the screw axis S_{12} at right angles. We will also use the Rodrigus equation for screws given by:

$$X - x = \tan\frac{\theta_{12}}{2}S_{12} \times (X + x)$$

In the following figure we can see the basic geometry of these equations. The screw x moves about the screw axis S_{12} up to the second position, ilustrated by the screw X. The rotation angle θ_{12} and translation t_{12} can be divided by two. We define the frame $\{S_{12},N,V\}$ where N is the line that bisects the rotation angle and passes through the
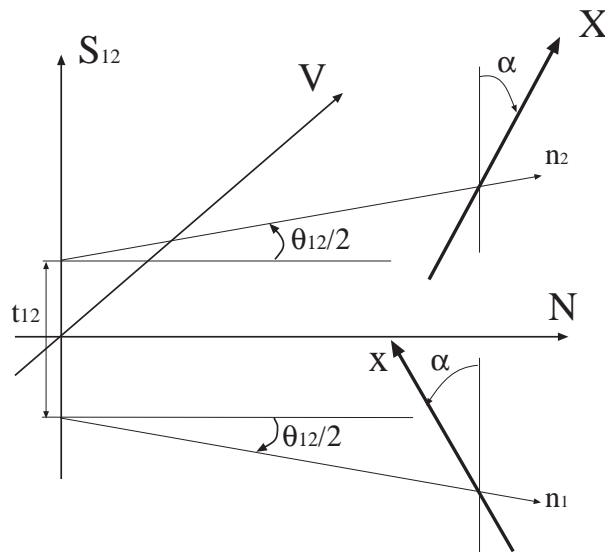
medium point of the displacement, and V=S$_{12}$xN.

Summarizing, a screw can always be interpreted as a displacement in space, with its screw axis defining the axis of rotation. The rotation angle and displacement can be related to the dual magnitude of the screw by the following formula: given a screw S of dual magnitude K=(k,kp),

$$K = \tan\frac{\hat{\phi}}{2} = \tan\frac{\phi}{2} + \varepsilon\frac{d}{2\cos^2\frac{\phi}{2}}$$

The Maple V.5 procedures

The following table summarizes the contain of the library "dualvectors", and the routines are attached after it.

Operation	Procedure	Description
Identify geometry	LINENORM	Intersection points on the common normal between two screws
	ANGDIST	Gives angle + distance between screws
	NORMALINE	Normal line to two spatial screws
	SCREWFRAME	Orthogonal components of a line
	DUALMAG	Dual magnitude of a screw (k,kp)
	ISLINE	Checks if a given screw is a line
	AXISCREW	Computes the screw axis (unitary line corresponding to a screw)
Operations	DUVEMULT	Product of dual numbers and dual vectors.
	DUMULT	Product of two dual numbers

	DUALINV	Inverts dual numbers
	SCREWDOT	Dot product of two screws
	SCREWCROSS	Cross product of two screws
	SCREWTRANS	Coordinate transformation for screws
Kinematic properties	SCREWAXIS	Given a 4x4 transformation matrix, computes the screw axis and its associated rotation + displacement.
	SCREWAXIS2	Finds the screw of a relative displacement
Graphic representation	LINEPLOT	Represents a line in space
	LINEPLOT2	Plots a line from a given point on it.
	LINEPLOT3	Plots sets of one, two or three lines with their common normals.
Format	FORMAT	Pass from any admitted format to the 2-columns matrix format for screws.
	FORMATN	Pass from epsilon format to vector format for dual numbers
	PASSTOE	Passes a screw from matrix form to epsilon form.
	PASSTOVEC	Passes a screw from matrix form to long vector form

```
#
##<SHAREFILE=dualvectors/dualvectors.mpl>
##<DESCRIBE>
##     This package performs operations with dual vectors
##     We define dual vectors as 3x2-dim arrays
##     or 6x1-dim arrays or as u+Ev, where E is such that E^2=0.
#
#########################################################
#     LINENORM
#     INTERSECTION POINTS ON THE COMMON NORMAL BETWEEN TWO LINES/SCREWS
#
#     Input two lines/screws, output a 3x2 array, first column int.
point with first screw,
#     second column int. point with second screw.(Att, c and d can be
the same point!)
#
linenorm:=proc(S,L)
local s,sc,v,w,t,p,q,n,t1,t2,c,d,points,S2,L2;
#Normalize expressions
S2:=format(S);
L2:=format(L);
#Take column vectors
s:=col(S2,1);
v:=col(S2,2);
w:=col(L2,1);
t:=col(L2,2);
#Points on the screws
p:=crossprod(s,v)/dotprod(s,s,'orthogonal');
q:=crossprod(w,t)/dotprod(w,w,'orthogonal');
#Unit normal vector
n:=crossprod(s,w);
#Points c,d on the common normal line
t1:=dotprod(crossprod(q-
p,w),n,'orthogonal')/dotprod(crossprod(s,w),n,'orthogonal');
t2:=dotprod(crossprod(q-
p,s),n,'orthogonal')/dotprod(crossprod(s,w),n,'orthogonal');
c:=evalm(p+t1*s);
d:=evalm(q+t2*w);
points:=concat(c,d);
eval(points)
end:
#
###################################################
#     ANGDIST
#     ANGLE + DISTANCE BETWEEN SCREWS
#
#     Input two screws,output dual 2-dim vector with (theta,r) (theta
in radians)
#
angdist:=proc(S,L)
local s,p,S2,L2,w,v,q,t,d1,d2,dotp,aux1,aux2,aux3;
#For lines: Given the line as two vectors, find the point.
#S=(s,pxs), L=(w,qxw)
#Identify structure
S2:=format(S);
L2:=format(L);
#Procedure
```

```
s:=col(S2,1);
v:=col(S2,2);
p:=crossprod(s,v)/dotprod(s,s,'orthogonal');
w:=col(L2,1);
t:=col(L2,2);
q:=crossprod(w,t)/dotprod(w,w,'orthogonal');
d1:=dotprod(s,w,'orthogonal');
d2:=dotprod(s,crossprod(q,w),'orthogonal')+dotprod(crossprod(p,s),w,'or
thogonal');
#Case they are lines or screws
if dotprod(s,v,'orthogonal')=0 and dotprod(w,t,'orthogonal')=0 then
      if  d1=1  or d1=-1 then dotp:=arccos(d1); RETURN(`Parallel lines,
cannot determine r`);
      else
      dotp:=vector(2,[arccos(d1),-d2/sqrt(1-d1^2)]);
      fi;
else
aux1:=axiscrew(S2);
aux2:=axiscrew(L2);
aux3:=screwdot(aux1,aux2);
dotp:=vector(2,[arccos(d1/aux3[1]),-d2/(aux3[1]*sqrt(1-
d1^2/aux3[1]^2))]);
fi;
eval(dotp)
end:
#
###############################################
#      NORMALINE
#      NORMAL LINE TO TWO SCREWS
#
#      Input two LINES, output the 3x2-dim normal LINE N
#
normaline:=proc(S,L)
local s,p,w,v,q,t,c1,c2,crossp,S2,L2,crossp2;
#S =(s,pxs),  L=(w,qxw)
S2:=format(S);
L2:=format(L);
s:=col(S2,1);
v:=col(S2,2);
p:=crossprod(s,v)/dotprod(s,s,'orthogonal');
w:=col(L2,1);
t:=col(L2,2);
q:=crossprod(w,t)/dotprod(w,w,'orthogonal');
c1:=crossprod(s,w);
c2:=crossprod(s,crossprod(q,w))+crossprod(crossprod(p,s),w);
crossp:=concat(c1,c2);
crossp2:=axiscrew(crossp);
eval(crossp2)
end:
#
###############################################
#      SCREWFRAME
#      ORTHOGONAL COMPONENTS OF A LINE
#
#      Input  lines  S,L,  being  S  the  reference  line.  Output  the  ref.
frame
#      V,N,S  + the 2-dim dual
```

```
#      number such that L=cos(thetå).S+sin(thetå).V, thetå=(theta,d)
#      The coordinate frame is V(=NxS),N(common normal to S,L),S
#
screwframe:=proc(S,L)
l   o   c   a   l                           p   o   i   n   t   s   ,
n,N,u,V,dualcos,dualsin,cosang,sinang,theta,d,dualangle,coordi,S2,L2,s,
v,p,t,w,q;
S2:=format(S);
L2:=format(L);
s:=col(S2,1);
v:=col(S2,2);
p:=crossprod(s,v)/dotprod(s,s,'orthogonal');
w:=col(L2,1);
t:=col(L2,2);
q:=crossprod(w,t)/dotprod(w,w,'orthogonal');
points:=linenorm(S2,L2);
n:=crossprod(s,w);
#Line N=(n,cxn)
N:=concat(n,col(points,1));
#Line V=(u,cxu)
u:=crossprod(n,s);
V:=concat(u,col(points,1));
#Coordinates of L in terms of V,N,S
#We note that L can be expressed as a combination of V,S
#To do that, we need the angle between L,S (theta+d)
#dualcos=cos(thetå),dualsin=sin(thetå)
#dualcos:=[cos(theta),-d*sin(theta)];
#dualsin:=[sin(theta),d*cos(theta)];
dualcos:=screwdot(S2,L2);
cosang:=dualcos[1];
sinang:=sqrt(1-cosang^2);
theta:=simplify(arccos(cosang),assume=RealRange(0,Pi));
d:=-dualcos[2]/sin(theta);
dualangle:=[theta,d];
coordi:={V,N,S,dualangle};
eval(coordi)
end:
#
################################################
#      AXISCREW
#      AXIS OF A SCREW
#
#      Input a screw, output  the 3x2-dim line axis of the screw
#
axiscrew:=proc(W)
local w,v,c,k,L,W2;
W2:=format(W);
w:=col(W2,1);
v:=col(W2,2);
c:=crossprod(w,v)/dotprod(w,w,'orthogonal');
#k,p magnitude and pitch of the screw. The line is unit vector
k:=sqrt(dotprod(w,w,'orthogonal'),symbolic);
L:=concat(w/k,crossprod(c,w/k));
eval(L)
end:
#
#########################################
```

```
#       DUALMAG
#       DUAL MAGNITUDE OF A SCREW
#
#       Input a screw, output the dual number (k,kp)
#
dualmag:=proc(W)
local w,v,k,p,du,W2;
W2:=format(W);
w:=col(W2,1);
v:=col(W2,2);
k:=sqrt(dotprod(w,w,'orthogonal'),symbolic);
p:=dotprod(w,v,'orthogonal')/dotprod(w,w,'orthogonal');
if p=0 then RETURN(`The screw is a line`); fi;
du:=[k,k*p];
eval(du)
end:
#
########################################
#      DUVEMULT
#      PRODUCT OF DUAL NUMBERS AND DUAL VECTORS
#
# Input a screw and a dual number. Output a 3x2-dim screw.
#
duvemult:=proc(L,k2)
local s,c,w,v1,v2,V,L2,k;
L2:=format(L);
k:=formatn(k2);
s:=col(L2,1);
w:=col(L2,2);
v1:=evalm(k[1]*s);
v2:=evalm(k[1]*w+k[2]*s);
V:=concat(v1,v2);
eval(V)
end:
#
########################################
#      DUMULT
#      PRODUCT OF DUAL NUMBERS
#
#      Input two 2-dim dual numbers, output a 2-dim dual number
#
dumult:=proc(k1,k2)
local p;
p:=[k1[1]*k2[1],k1[1]*k2[2]+k1[2]*k2[1]];
eval(p)
end:
#
#
######################################
#      DUALINV
######################################
#      Routine to invert dual numbers
#      Input a dual number, output the inverse dual number
#
dualinv:=proc(d)
local df,dinv;
with(dualvectors):
```

```
df:=formatn(d);
dinv:=vector(2,[1/df[1],-df[2]/(df[1])^2]);
eval(dinv)
end:
#
#
###########################################
#     SCREWDOT
#     DOT PRODUCT OF TWO SCREWS
#
#     Input two 3x2-dim screws, output a 2-dim dual number
#
screwdot:=proc(S,L)
local s,w,v,t,a,b,dotp,S2,L2;
S2:=format(S);
L2:=format(L);
s:=col(S2,1);
v:=col(S2,2);
t:=col(L2,1);
w:=col(L2,2);
a:=dotprod(s,t,'orthogonal');
b:=dotprod(s,w,'orthogonal')+dotprod(v,t,'orthogonal');
dotp:=vector(2,[a,b]);
eval(dotp)
end:
#
###################################
#     SCREWCROSS
#     CROSS PRODUCT OF TWO SCREWS
#
#     Input two 3x2-dim screws, output a 3x2-dim screw
#
screwcross:=proc(S,L)
local s,w,v,t,c1,c2,crossp,S2,L2;
S2:=format(S);
L2:=format(L);
s:=col(S2,1);
v:=col(S2,2);
t:=col(L2,1);
w:=col(L2,2);
c1:=crossprod(s,t);
c2:=crossprod(s,w)+crossprod(v,t);
crossp:=concat(c1,c2);
eval(crossp)
end:
#
#
####################################
#     SCREWAXIS
#     ###############################
#     Given a transformation defined by rotation matrix + displacement
vector,
#     find the screw axis and its associated angle and displacement.
####################################
#
screwaxis:=proc()
local A,d,Id,B,b,S,s,v,c,phi,di,Saxis,mag,ans;
```

```
with(linalg): with(dualvectors):
Id:=array(identity,1..3,1..3);
if nargs=1 and type(args[1],array) and coldim(args[1])=4 and
rowdim(args[1])=4 then
A:=submatrix(args[1],1..3,1..3);
d:=vector([col(args[1],4)[1],col(args[1],4)[2],col(args[1],4)[3]]);

elif nargs=2 and type(args[1],array) and type(args[2],array) and
coldim(args[1])=3 and rowdim(args[1])=3 and vectdim(args[2])=3 then
A:=args[1];
d:=args[2];
else RETURN(`Invalid dimensions`);
fi;
B:=evalm((A-Id) &* inverse(A+Id));
b:=vector(3,[B[3,2],B[1,3],B[2,1]]);
s:=evalm(b/sqrt(b[1]^2+b[2]^2+b[3]^3,symbolic));
S:=matrix(3,3,[0,-s[3],s[2],s[3],0,-s[1],-s[2],s[1],0]);
phi:=2*arctan(sqrt(b[1]^2+b[2]^2+b[3]^3,symbolic));
di:=dotprod(d,s,orthogonal);
v:=evalm(1/(2*tan(phi/2))*(S &* S + tan(phi/2) * S) &* d);
c:=crossprod(s,v);
Saxis:=concat(s,crossprod(c,s));
mag:=vector(2,[phi,di]);
ans:=[Saxis,mag];
eval(ans)
end:
#
#
#
##################################
#     SCREWAXIS2
##################################
#     Given two screws of two spatial transformations,
#     find the screw axis(phi,d) of the relative displacement S12
##################################
#
screwaxis2:=proc()
local
Sa,Sb,sa,sb,va,vb,tana2,tanb2,duala,dualb,da,db,saxisa,saxisb,Sc,sc,tan
c2,phic,dc,du,ans;
with(linalg): with(dualvectors):
if nargs=2 and coldim(args[1])=2 and rowdim(args[1])=3 and
coldim(args[2])=2 and rowdim(args[2])=3 then
Sa:=args[1];
Sb:=args[2];
sa:=col(Sa,1);
va:=col(Sa,2);
sb:=col(Sb,1);
vb:=col(Sb,2);
# Find angle and magnitude of the screws (i dont need it, but...)
duala:=screwdot(Sa,Sa);
dualb:=screwdot(Sb,Sb);
tana2:=sqrt(dotprod(sa,sa,orthogonal),symbolic);
da:=2*dotprod(sa,va,orthogonal)/(tana2*(tana2^2+1));
tanb2:=sqrt(dotprod(sb,sb,orthogonal),symbolic);
db:=2*dotprod(sb,vb,orthogonal)/(tanb2*(tanb2^2+1));
saxisa:=axiscrew(Sa);
```

```
saxisb:=axiscrew(Sb);
# Relative screw
Sc:=evalm(duvemult(evalm(Sb+Sa+screwcross(Sb,Sa)),dualinv(evalm(vector(
2,[1,0])-screwdot(Sb,Sa)))));
#Screw axis
sc:=axiscrew(Sc);
#Angle and displacement
tanc2:=sqrt(dotprod(col(Sc,1),col(Sc,1),orthogonal),symbolic);
phic:=2*arctan(tanc2);
dc:=2*dotprod(col(Sc,1),col(Sc,2),orthogonal)/(tanc2*(tanc2^2+1));
du:=vector(2,[phic,dc]);
ans:=[sc,du];
eval(ans)
end:
#
#
################################
#      SCREWTRANS
################################
#      Routine to transform coordinates for screws.
#      Given the transformation matrix, multiplies by the dual vector
################################
#
screwtrans:=proc(M,S)
local S2,d,A,w,v,wa,va;
S2:=format(S);
w:=col(S2,1);
v:=col(S2,2);
A:=submatrix(M,1..3,1..3);
d:=vector(3,[M[1,4],M[2,4],M[3,4]]);
wa:=evalm(A &* w);
va:=evalm(crossprod(d,evalm(A &* w))+A &* v);
ans:=concat(wa,va);
eval(ans)
end:
#
#
#
###################################
#      FORMAT
#      PASS TO 2-COL FORMAT FOR SCREWS
#
#      input a screw S in any format, output a screw S in format 3x2
matrix
#
format:=proc(S)
local s,v,S2;
#what kind of screws are these? Is it a 6x1 vector?
if type(S,vector)=true then
if vectdim(S)=6 then
S2:=array(1..3,1..2,[[S[1],S[4]],[S[2],S[5]],[S[3],S[6]]]);
fi;
#If the screws are given as two vectors
elif type(S,matrix)=true then
if coldim(S)=2 and rowdim(S)=3 then
S2:=S;
fi;
```

63

```
#If the dualvector is given as a+E*b,
elif whattype(S)=`+` and op(1,op(2,S))=E then
s:=op(1,S);
v:=op(2,op(2,S));
S2:=concat(s,v);
else RETURN(`Invalid screw dimension`)
fi;
eval(S2)
end:
#
#
####################################
#     FORMATN
#     PASS TO VECTOR FORMAT FOR DUAL NUMBERS
#
#     input a dual number in any format, output a 2-vector
#
formatn:=proc(k)
local k2;
if type(k,vector)=true then
if vectdim(k)=2 then
k2:=k;
fi;
elif whattype(k)=`+` and op(1,op(2,k))=E then
k2:=array([op(1,k),op(2,op(2,k))]);
else RETURN(`Invalid  dual number`)
fi;
eval(k2)
end:
#
####################################
#     PASSTOE
#     PASS FROM MATRIX MODE TO EPSILON MODE FOR SCREWS
#
#     input a matrix form screw, output a a+Eb form screw
#
passtoE:=proc(W)
local W2;
if type(W,matrix)=true and coldim(W)=2 and rowdim(W)=3 then
W2:=col(W,1)+E*col(W,2);
else RETURN(`Wrong dimensions`);
fi;
eval(W2)
end:
#
#####################################
#     PASSTOVEC
#     PASS FROM MATRIX MODE TO VECTOR MODE FOR SCREWS
#
# input a screw, output a 6-vector screw
#
passtovec:=proc(W)
local W2;
if type(W,matrix)=true and coldim(W)=2 and rowdim(W)=3 then
W2:=array([W[1,1],W[2,1],W[3,1],W[1,2],W[2,2],W[3,2]]);
else RETURN(`Wrong dimensions`);
fi;
```

```
eval(W2)
end:
#
#
###################################
#     ISLINE
#     IDENTIFICATION OF LINES
#
#     input a line, output a true/false variable
#
isline:=proc(W)
local s,v,W2,d;
W2:=format(W);
s:=col(W2,1);
v:=col(W2,2);
d:=dotprod(s,v,'orthogonal');
if d=0 then RETURN(true);
else RETURN(false);
fi;
end:
#
###################################
#     LINEPLOT
#     PLOT OF LINES IN 3D
#
# represents 3d lines based on two points of the line
#     to plot more than one line, store them in variables
#     and make display(l1,...,ln);
#
lineplot:=proc(L)
local L2,s,v,p,p2,draw;
L2:=format(L);
s:=col(L2,1);
v:=col(L2,2);
with(plots);
p:=crossprod(s,v)/dotprod(s,s,'orthogonal');
p2:=p+2*s;
draw:=pointplot3d({evalm(p),evalm(p2)},connect=true,axes=normal);
eval(draw)
end:
#
#
###################################
#     LINEPLOT2
#     PLOTS OF LINES IN 3D FROM A GIVEN POINT
#
#     represents 3d lines taking a given point of the line
#     as the center of the plotted line. Input line and point,
#     output the plot itself.
#
lineplot2:=proc(L,point)
local L2,s,p1,p2,draw;
L2:=format(L);
s:=col(L2,1);
with(plots):
p1:=point-2*s;
p2:=point+2*s;
```

```
draw:=pointplot3d({evalm(p1),evalm(p2)},connect=true,axes=normal);
eval(draw)
end:
#
#
###################################
# LINEPLOT3
# PLOTS OF 1,2 OR 3 LINES IN 3D WITH THEIR COMMON NORMALS
#
#      Input 3d lines, output the plot of the lines plus their common
#      normal lines (dashed). It computes the intersection points and
#      prints the lines from them.
#
lineplot3:=proc()
local
SF1,SF2,SF3,points12,points13,points23,plot1,plot2,plot3,plot4,plot5,pl
ot6,ans;
with(dualvectors):
with(plots):
if nargs=1 then ans:=lineplot(args[1]);
elif nargs=2 and type(args[1],array) and type(args[2],array) then
      SF1:=format(args[1]);
      SF2:=format(args[2]);
if type(SF1,array)=false or type(SF2,array)=false then
RETURN(`Invalid screw dimensions`);
else
      points12:=linenorm(SF1,SF2);
      plot1:=lineplot2(SF1,col(points12,1));
      plot2:=lineplot2(SF2,col(points12,2));
plot3:=pointplot3d({evalm(col(points12,1)),evalm(col(points12,2))},conn
ect=true,axes=normal,linestyle=3);
      ans:=display([plot1,plot2,plot3],axes=frame);
fi;
elif nargs=3 then
      SF1:=format(args[1]);
      SF2:=format(args[2]);
      SF3:=format(args[3]);
if type(SF1,array)=false or type(SF2,array)=false then
RETURN(`Invalid screw dimensions`);
else
points12:=linenorm(SF1,SF2);
plot1:=lineplot2(SF1,col(points12,1));
plot2:=lineplot2(SF2,col(points12,2));
plot3:=pointplot3d({evalm(col(points12,1)),evalm(col(points12,2))},conn
ect=true,axes=frame,linestyle=3);
points13:=linenorm(SF1,SF3);
plot4:=lineplot2(SF3,col(points13,2));
plot5:=pointplot3d({evalm(col(points13,1)),evalm(col(points13,2))},conn
ect=true,axes=frame,linestyle=3);
points23:=linenorm(SF2,SF3);
plot6:=pointplot3d({evalm(col(points23,1)),evalm(col(points23,2))},conn
ect=true,axes=normal,linestyle=3);
ans:=display([plot1,plot2,plot3,plot4,plot5,plot6],axes=frame);
      fi;
else RETURN(`Wrong number of parameters`)
fi;
eval(ans)
```

```
end:
#
#
#
#
read     "Cinema:Applications:Maple     V     Release
5:share:dualvectors:dualvectors.msl";
#savelib('linenorm','angdist','normaline','screwframe',\
'axiscrew','dualmag','duvemult','dumult',
'dualinv','screwdot','screwcross',\
'screwaxis', 'screwaxis2','screwtrans','format','formatn','passtoE',\
'passtovec','isline','lineplot','lineplot2','lineplot3','_Share',\
"dualvectors.m");
#quit
```

Appendix 2

Preocedures for the Algebra of Quaternions

Procedures for the algebra of quaternions

The following procedures were implemented to be used with the software Maple V.5. They collect the basic operations to use when working with quaternions.

The algebra of quaternions

A quaternion is a generalization of a complex number. As the complex numbers describe rotations in the plane, the quaternions are the minimum-dimensional element to describe rotations in the three-dimensional space. Algebraically, we can consider quaternions as ordered pairs of complex numbers [Ward].

$$Q = 1.a + ib + jc + kd$$

where a, b, c, d are real numbers. We call a the point. The unit vectors i, j, k form a vector space.

The algebra of quaternions has the following properties:

$$Q = Q' \Leftrightarrow a = a', b = b', c = c', d = d'$$
$$Addition: Q + Q' = (a + a') + (b + b')i + (c + c')j + (d + d')k = Q' + Q$$
$$Mult. \, by \, scalar: sQ = sa + sbi + scj + sdk$$
$$Associative$$
$$Distributive$$

The scalar product of the algebra of quaternions allows us to define the conjugate of a quaternion:

$$\overline{Q} = a - bi - cj - dk$$

And the norm of the quaternion as:

$$N_Q = a^2 + b^2 + c^2 + d^2$$

The angle associated with the quaternion is defined as:

$$\cos\theta = \frac{a}{\sqrt{N_Q}}$$

$$\sin\theta = \frac{\sqrt{b^2 + c^2 + d^2}}{\sqrt{N_Q}}$$

We can specify the quaternion in polar form

$$Q = \sqrt{N_Q}(\cos\theta + \hat{q}\sin\theta)$$

where q is a unit vector.

The quaternionic multiplication is the main use of quaternions in kinematics. We will associate rotations in space with multiplications of quaternions, so that the problem can be restricted to complex number products.

The operation is defined in its natural way, with the following rules:

$$i^2 = j^2 = k^2 = -1$$
$$ij = k$$
$$ji = -k$$

Thus, if we call S the scalar part and V the vector part of the quaternion, we have:

$$QQ' = (S + V)(S' + V') = SS' - V \cdot V' + S'V + SV' + V \times V'$$

with the usual dot and cross product of vectors.

Notice as the first important property that the multiplication of quaternions is not commutative. This coincides with the behavior of the rotations in space. However, this operation keeps the associativity and commutativity. From the definition, we can deduce easily the properties:

$$S_{QP} = S_Q S_P - V_Q \cdot V_P = S_{PQ}$$
$$N_Q = Q\overline{Q} = S_Q^2 - N_{V_Q}$$
$$\overline{QP} = \overline{P}\,\overline{Q}$$
$$N_{PQ} = N_P N_Q$$

If a quaternion has a non-zero norm, we can define the inverse quaternion as:

$$Q^{-1} = \frac{\overline{Q}}{N_Q}$$

With the properties:

$$QQ^{-1} = Q^{-1}Q = 1$$

$$N_{P/Q} = \frac{N_P}{N_Q}$$

$$(PQ)^{-1} = Q^{-1}P^{-1}$$

$$QP - PQ = V_Q \times V_P - V_P \times V_Q$$

The quaternion representation of a rotation matrix

Given a rotation matrix, we can express it as a function of the Euler parameters. Using the Cayley's formula we arrive to the expression:

$$[A] = \begin{bmatrix} c_0^2 + c_1^2 - c_2^2 - c_3^2 & 2(c_1c_2 - c_0c_3) & 2(c_1c_3 + c_0c_2) \\ 2(c_1c_2 + c_0c_3) & c_0^2 - c_1^2 + c_2^2 - c_3^2 & 2(c_2c_3 - c_0c_1) \\ 2(c_1c_3 - c_0c_2) & 2(c_2c_3 + c_0c_1) & c_0^2 - c_1^2 - c_2^2 + c_3^2 \end{bmatrix}$$

Where the c's are the components of the quaternion, defined as:

$$\vec{c} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_0 \end{bmatrix} = \begin{bmatrix} s_x \sin\frac{\phi}{2} \\ s_y \sin\frac{\phi}{2} \\ s_z \sin\frac{\phi}{2} \\ \cos\frac{\phi}{2} \end{bmatrix}$$

This vector is composed of the rotation axis and angle (vector b of Cayley's formula) plus an additional angle term to make it unitary.

Consider the rotation transformation given by X=[A]x. We can express any vector in quaternion form, making the scalar part equal to zero. Then we can use the quaternionic multiplication to prove the relation

$$Qx\overline{Q} \leftrightarrow [A]x$$

We can operate with quaternions as we do it with matrices; the rotation described by A_2A_1 (two consecutive rotations) corresponds to the quaternion Q_2Q_1, and the inverse rotation A^{-1} is equivalent to that of the conjugate quaternion.

The quaternion of a basic rotation about the axis with direction angles α,β,γ and rotation angle $\phi/2$ is:

$$Q = \begin{bmatrix} \cos\alpha\sin\dfrac{\phi}{2} \\[2ex] \cos\beta\sin\dfrac{\phi}{2} \\[2ex] \cos\gamma\sin\dfrac{\phi}{2} \\[2ex] \cos\dfrac{\phi}{2} \end{bmatrix}$$

The procedures for quaternions

The routines can be found in the library "quaternion" for Maple V.5. The following table summarizes its content, and the routines are attached after this.

Procedure	Description
QUADD	Quaternion addition
QUSMULT	Quaternion multiplication by scalar
QUMULT	Quaternion multiplication
QUCONJ	Conjugate quaternion
QUNORM	Quaternion norm
QUANGLE	Angle of the quaternion
QUINV	Inverse quaternion

```
#
##<SHAREFILE=quaternion/quaternion.mpl>
##<DESCRIBE>
##    This   package   performs   quaternion   operations:   addition,
multiplication
##    by scalar, multiplication, conjugation, norm, inverse.
##    We define quaternions as 4-dim arrays.  Later we will define
##    quaternions as a type of variables (using a+bI+cJ+dK).
#
###############################
##    QUATERNION ADDITION
###############################
#
quadd:=proc(p,q)
local r;
r:=array(1..4,[p[1]+q[1],p[2]+q[2],p[3]+q[3],p[4]+q[4]]);
eval(r)
end:
#
###############################
##    QUATERNION MULTIPLICATION BY SCALAR
###############################
#
qusmult:=proc(k,q)
local p;
p:=array(1..4,[k*q[1],k*q[2],k*q[3],k*q[4]]);
#alias (qs* = qusmult):
eval(p)
end:
#
###############################
##    QUATERNION MULTIPLICATION
###############################
#
qumult:=proc(p,q)
local r,a,b,c,d;
a:=p[1]*q[1]-p[2]*q[2]-p[3]*q[3]-p[4]*q[4]:
b:=p[1]*q[2]+q[1]*p[2]+p[3]*q[4]-p[4]*q[3]:
c:=p[1]*q[3]+q[1]*p[3]+p[4]*q[2]-p[2]*q[4]:
d:=p[1]*q[4]+p[4]*q[1]+p[2]*q[3]-p[3]*q[2]:
r:=array(1..4,[a,b,c,d]);
eval(r)
end:
#
###############################
##    CONJUGATE QUATERNION
###############################
#
quconj:=proc(q)
local p;
p:=array(1..4,[q[1],-q[2],-q[3],-q[4]]);
eval(p)
end:
#
###############################
##    QUATERNION NORM
###############################
```

```
#
qunorm:=proc(q)
local n;
n:=q[1]^2+q[2]^2+q[3]^2+q[4]^2;
eval(n)
end:
#
##############################
##     ANGLE OF THE QUATERNION
##############################
#
quangle:=proc(q)
local theta,n;
n:=qunorm(q):
theta:=arccos(q[1]/sqrt(n));
eval(theta)
end:
#
##############################
## INVERSE QUATERNION
##############################
#
quinv:=proc(q)
local p,n;
n:=qunorm(q):
if n=0 then
RETURN(lprint(`This quaternion has not inverse`))
fi;
p:=quconj(q)/n;
eval(p)
end:
#

##
read "quaternion.msl";
#savelib('quadd','qusmult','qumult','quconj',\
'qunorm','quangle','quinv','_Share',"quaternion.m");
#quit
```

Appendix 3

Maple Worksheets for the Analysis and Design of RR Chains