# Optimization of Spherical Four-Bar Path Generators

Zheng Liu

(B.Eng, Tsinghua University, China, 1985)

Department of Mechanical Engineering

McGill University

Montréal, Canada

A thesis submitted to the Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

Master of Engineering

November 1988

# Abstract

In this thesis, the optimization of spherical four-bar linkages for the problem of path generation is presented In this problem, a set of points is given, and the linkage whose coupler link contains a point tracing a trajectory, called coupler curve, passing as close as possible to a given set is sought The problem is formulated as a two-layer minimization of the linkage error which is defined as the sum of the distances between the coupler curve and each point of the given set, thereby decoupling the linkage parameters from the configuration variables. Hence, the optimization procedure consists of evaluating a set of input angles, $\{\psi_k\}_1^m$, defining $m$ linkage configurations, and the linkage parameters independently. This leads to a nonlinear least-square minimization problem with equality constraints The *orthogonal-decomposition algorithm*, introduced elsewhere, is employed to solve the problem, which allows us to obtain the solution iteratively. Continuation and damping techniques are used in the numerical procedure to ensure convergence and speed up its rate. The optimization scheme is developed on a general basis and can handle the problems of $m$ prescribed points, where $m$ can be any number greater than nine Several design problems are solved by using the method and results are presented in the thesis. In addition to solving the synthesis problem, a novel criterion for mobility analysis of the spherical four-bar linkage was devised and is included in the thesis.

# Résumé

Dans cette thèse, l'auteur présente une méthode d'optimisation des mécanismes sphériques à quatre barres articulées pour le tracée de trajectoires Dans le cadre de ce problème, un ensemble de points est donné et l'on cherche le mécanisme dont la bielle contient un point traçant une trajectoire, dite la courbe de la bielle, qui passe aussi proche que possible de l'ensemble donné La formulation du problème consiste en une minimisation à deux stages imbriqués de l'erreur du mécanisme, qui est défini ici commme la somme des distances entre la courbe de la bielle et chacun des points de l'ensemble donné. Cette approche permet de découpler les paramètres cinématiques du mécanisme et les variables décrivant la configuration Ainsi, la procédure d'optimisation consiste à évaluer indépendament un ensemble d'angles d'entrée, $\{\psi_k\}_1^m$, définissant $m$ configurations, et les paramètres cinématiques du mécanisme On formule ainsi un problème de minimisation dit de moindres carrés contraint, soumis à des contraintes d'égalité. L'algorithme de décomposition orthogonale, présenté ailleurs, est utilisé pour résoudre ce probleme, ce qui conduit à une solution itérative Les techniques de continuation et d'amortissement sont également introduites afin d'accélérer la procédure et de garantir sa convergence De plus, la généralité de l'algorithme d'optimisation développée ici permet de résoudre le problème de tracée de trajectoire pour $m$ points, où $m$ peut être n'importe quel nombre supérieur à neuf Plusieurs problèmes de conception de mécanismes sont résolus à l'aide de la méthode décrite ci-haut et les résultats sont présentés dans cette thèse. Enfin, l'auteur présente dans cette thèse une nouvelle condition de mobilité du mécanisme sphérique à quatre barres articulées.

# Acknowledgements

I would like to thank professor Jorge Angeles, my research supervisor, for his guidance, encouragement and support, without which this research work could not have been completed.

I am indebted to Mr. Clément Gosselin for his helpful instructions on using the computer package implementing the orthogonal-decomposition algorithm and some graphics software on the SUN workstations of the McRCIM (McGill Research Centre for Intelligent Machines) network Additionally, he kindly help me with the translation of the abstract of the thesis Also to be thanked is Mr Ou Ma, who provided a preliminary version of linkage analysis routines on the Orcatech workstation, part of which were later transferred to the SUN workstations, debugged and modified for the computations of the Jacobian Matrix of the objective function

I would like to express my gratitude to those who help me with the English and French while producing the thesis, particularly Messrs Owen Williams and Roger Cormier Jr.. I also wish to thank all my colleagues and friends in the Robotic Mechanical System Laboritory of McRCIM and in the Department of Mechanical Engineering for sharing with me their time in the course of my research

Many thanks are due to McRCIM for all the computer facilities provided and for the pleasant research environment

The research was possible under Quebec's IRSST (Institut de recherche en santé et en sécurité du travail du Québec) Grant No. IRSS-87-06 and NSERC (Natural Science and Engineering Research Council of Canada) Research Grant No A4532. Additional funding was provided to the author through a scholarship from the People's Republic of China and a postgraduate fellowship from the Department of Mechanical Engineering.

# Contents

# List of Figures

# Claim of Originality

The author claims the originality of the idea of the optimization scheme reported in this thesis. Moreover, to the author's knowledge, the criterion for the mobility analysis of spherical four-bar linkages is original and has not been presented elsewhere.

# Chapter 1 Introduction

Modern technology has brought along a wide application of the spherical mechanism to industry. We have, in this category, solar-tracking mechanisms, rotating radar mechanisms, robotic wrists and prosthetic devices, such as artificial wrists and subtalar joints, etc. In this type of mechanisms, the motion takes place on the surface of a sphere. Of all spherical mechanisms, the spherical four-bar linkage is the simplest in structure and has received a special attention A common example of this type of mechanism is the universal joint. Many practical problems dealing with spherical four-bar linkages are related to the trajectory followed by a certain point on the coupler link, which leads to the problem of path generation The design of a solar collector, for example, which traces a solar-path, belongs to this category. Hence, the significance of developing efficient schemes for solving this type of problems becomes apparent.

The synthesis of path-generating linkages is a classical problem in applied kinematics In connection with the four-bar linkage, it requires to determine the dimensions of the linkage having the property that one of the points of its coupler link passes through certain prescribed positions Reported in this thesis is the solution of the aforementioned problem in the spherical case. In this context, all the relevant parameters of a spherical *RRRR* four-bar linkage are computed, given a set of points on the surface of the sphere on which the motion of the linkage takes place, so that one point of its coupler link, which we call coupler point, traces a path whose distance to the given points is a minimum. The problem solved here is aimed at the approximate synthesis of the linkage, the exact syn-

thesis being regarded as a particular case of this.

Linkage synthesis problems have been a subject of intensive research in kinematics. Classical problems, such as path generation, rigid-body guidance and function generation, have been extensively studied Various methods have been proposed in the literature to deal with these problems Tomáš (1968) showed that the majority of problems concerning the synthesis of mechanisms can be formulated as nonlinear programming problems. Fox and Willmert (1967) and Alizade, Novruzbekov and Sandor (1975) applied nonlinear programming techniques to solve the synthesis problem for function generation, while Rao (1979) used geometric programming to solve the same type of synthesis problems Sutherland and Karwa (1978) developed a general scheme for the synthesis of linkages for rigid-body guidance. Chiang (1984) used the concept of kinematic inversion and solved the synthesis of path generation symbolically. Other methods have been studied by different researchers, such as the geometric and algebraic methods proposed by Hartenberg and Denavit (1964), and the Monte-Carlo method (Golinski, 1970). On the whole, a variety of methods has been developed by different researchers in the past decades for dealing with various problems in the area of linkage synthesis.

However, it seems that the research has mostly focused on planar linkages. Investigations in the spherical case are relatively few. For the case of path generation of spherical four-bar linkages, although some work has been done, the arising problems are mainly solved for not more than four given points (Suh and Radcliffe, 1978) or for the coordination of three or four prescribed points and the corresponding crank rotations (Chiang, 1986). Although it has been proved theoretically that up to nine given points on a sphere can be met exactly by a spherical four-bar linkage (Kraus, 1952), finding the solutions becomes a major task, due to the highly nonlinear nature of the equations involved Up to now, no research has been reported in the literature for more than four given positions. It seems clear that, as the synthesis problems turn more complicated, the conventional geometrical and symbolic methods meant for the exact-synthesis problem are very limited and difficult to apply. Consider the problem of path generation, for example. On

the one hand, those methods can only be used for less than five prescribed positions which are to be met exactly. On the other hand, when the number of prescribed points exceeds a certain maximum, namely, nine, an overdetermined system of nonlinear equations will arise, which means that no exact solutions are possible Thus, for such problems, we must resort to numerical methods for optimization As practical path-generating problems may very likely include more than nine points, devising a general scheme capable of handing more points becomes of the utmost concern This motivates the research work reported in this thesis.

Recently, Angeles, Alivizatos and Akhras (1987) and Akhras and Angeles (1987) proposed a new method to solve approximate synthesis problems for planar linkages They applied a variable-separation technique and unconstrained nonlinear least-square optimization schemes to the problems of path generation and rigid-body guidance Their method is very efficient, since it can handle any number of given positions and orientations In seeking the same performance for spherical problems, unconstrained methods are no longer applicable, and hence, constrained least-square techniques have to be introduced While some common features between planar and spherical linkages exist, special characteristics concerning spherical linkages must be considered For example, due to the fact that all the joints and points of interest in the problem lie on the surface of a sphere, algebraic constraints on the coordinates of points defining the location of the centre of these joints are necessary. The problem is hence formulated as one of constrainted nonlinear least-square optimization. Two layers of minimization are included, each of them being considered separately. An iterative scheme is used in evaluating the design variables in the outer layer of optimization, while, in the inner layer, the optimal choice of input angle corresponding to a linkage configuration, with the coupler point lying the closest to a given point, is found. In this formulation, equality constraints are used, which eases the computational procedure to a large extent In fact, whereas the solution of least-square minimization problems, subject to continuous equality constraints, are *continuous* functions of the problem parameters, those of mathematical programming problems subject to inequality constraints are, in general, *discontinuous* functions of the problem parameters. This property has rel-

evant consequences, for it allows the introduction of a technique known as *continuation* in the solution procedure This technique, in turn, guarantees the convergence to a solution—provided that the problem has one—even if the initial guess lies far from the said solution. Moreover, the convergence rate is enhanced by the use of *damping* techniques in the iterative procedure All these techniques contribute to the efficiency of the method

The optimization scheme aims at solving problems in which a set of points are given on the surface of a sphere To this end, the problem of synthesis for path generation, given any number of points, greater than nine, can be solved However, no coordination between these points and the input angles are considered in the formulation For this type of problems, certain modifications in the formulation are necessary, which fall beyond the scope of the thesis

A Fortran77 computer package, called SPHER1, has been written for implementing the whole optimization scheme and runs on the Unix-based workstations of the McRCIM network A brief description of the computer package is given in Appendix E. Some design problems, including solar-tracking mechanisms and a linkage for driving a Geneva wheel, are solved by using the computer package. The results are reported in Chapter 6

In addition to the synthesis work mentioned above, the mobility range of spherical four-bar linkages are studied using the concept of linkage discriminant The full-mobility regions of both the input and the output links are described in the 4-D space of linkage parameters a novel form of the criterion for mobility analysis being thus proposed The details are given in Chapter 2.

# Chapter 2 Background and Terminology

## 2.1 On Spherical Four-Bar Linkages

Shown in Fig. 2.1 is a general spherical *RRRR* four-bar linkage The axes of the four revolute joints, *A*, *B*, *C* and *D*, intersect in one point *O*, the centre of the sphere on which the motion of the linkage takes place We define *AD* as the fixed link, *AB* and *CD* as the input and output links, respectively, and *BC* as the coupler link A point *P* on the coupler link is chosen as the coupler point which will generate a coupler curve when the linkage is in motion The link dimensions are given by the angle between adjacent joint axes, i e , $\alpha_i$, for $i = 1, 2, 3, 4$, as shown in Fig 2 1 Additionally, three other angles, $\alpha_5$, $\alpha_6$ and $\alpha_7$, are defined to describe the coupler point Possible combinations of the length of each link yield different types of linkages double-crank, crank-rocker, rocker-crank, or double-rocker As in the planar case, the well-known rule of Grashof can be used to identify linkage types (Hartenberg and Denavit, 1964, Chiang, 1984). Here a new method is developed for this purpose, as described below

### 2.1.1 Input-Output Analysis

Since the four-bar linkage is a one-degree-of-freedom mechanism, the input angle $\psi$, namely, the angle between the links *AB* and *AD* measured from the major circle

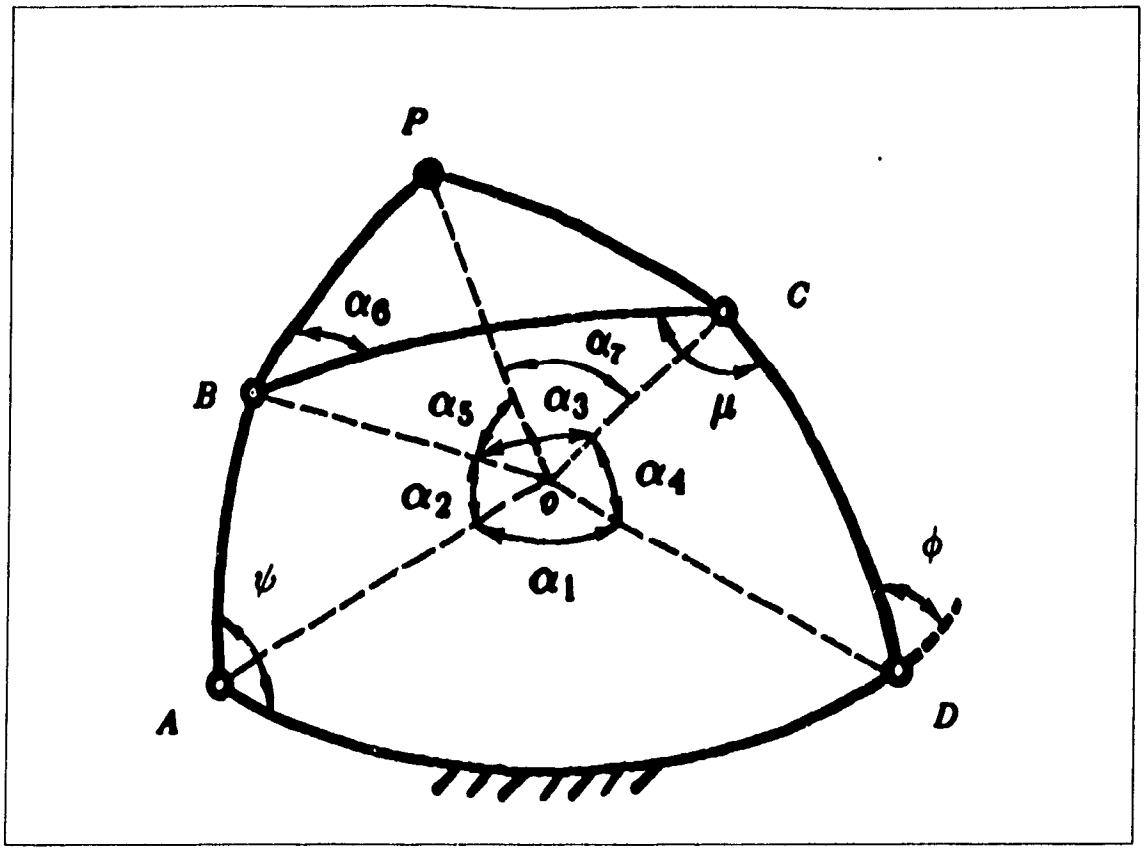


**Figure 2.1** A General Spherical RRRR Four-Bar Linkage

*AD*, defines its motion. Correspondingly, the angle between *CD* and *AD*, measured as indicated in Fig 2.1 and called $\phi$, is defined as the output angle. A distinct function, which is called the input-output function, exists that defines the relationships between the input and the output angles, as described in (Angeles, 1982, Angeles and Bernier, 1987). Here, an alternative form is used, namely,

$$f(\psi, \phi; \mathbf{k}) = k_1 + k_2 \cos\psi + k_3 \cos\psi \cos\phi - k_4 \cos\phi + \sin\psi \sin\phi = 0 \qquad (2.1)$$

where $\mathbf{k}$ is defined as the four-dimensional vector of linkage parameters, its $k_i$ components, for $i = 1, 2, 3, 4$, being defined as follows:

$$k_1 = \frac{\cos\alpha_1 \cos\alpha_2 \cos\alpha_4 - \cos\alpha_3}{\sin\alpha_2 \sin\alpha_4} \qquad (2.2a)$$

$$k_2 = \frac{\sin\alpha_1 \cos\alpha_4}{\sin\alpha_4} \qquad (2.2b)$$

$$k_3 = \cos\alpha_1 \qquad (2.2c)$$

$$k_4 = \frac{\sin\alpha_1 \cos\alpha_2}{\sin\alpha_2} \tag{2.2d}$$

The above equation, although related to that given by Angeles and Bernier (1987), bears some differences with it. The major improvement here is the avoidance of the singularities at $\alpha_2 = 90^\circ$. A possible inversion of those equations is the following.

$$\begin{aligned}
\sin\alpha_1 &= \sqrt{1-k_3^2}, & \cos\alpha_1 &= k_3 \\
\sin\alpha_2 &= \sqrt{\frac{1-k_3^2}{1+k_4^2-k_3^2}}, & \cos\alpha_2 &= \frac{k_4}{\sqrt{1+k_4^2-k_3^2}}, \\
\sin\alpha_3 &= \sqrt{1-\frac{[k_2k_3k_4-k_1(1-k_3^2)]^2}{(1-k_3^2+k_4^2)^2(1+k_1^2-k_3^2)^2}}, \\
\cos\alpha_3 &= \frac{[k_2k_3k_4-k_1(1-k_3^2)]}{(1-k_3^2+k_4^2)(1+k_1^2-k_3^2)}, \\
\sin\alpha_4 &= \sqrt{\frac{1-k_3^2}{1+k_2^2-k_3^2}}, & \cos\alpha_4 &= \frac{k_2}{\sqrt{1+k_2^2-k_3^2}}
\end{aligned} \tag{2.3}$$

where all the angles are assumed to be within the range from 0 to $\pi$.

With simple trigonometrical transformations, eq.(2.1) can also be expressed in the following form.

$$A(\psi)T^2 + 2B(\psi)T + C(\psi) = 0 \tag{2.4}$$

where

$$A(\psi) = k_1 + k_4 + (k_2 + k_3)\cos\psi \tag{2.5a}$$

$$B(\psi) = \sin\psi \tag{2.5b}$$

$$C(\psi) = k_1 - k_4 + (k_2 + k_3)\cos\psi \tag{2.5c}$$

$$T = \tan(\phi/2) \tag{2.5d}$$

The output angle in terms of the input angle is then obtained as:

$$\phi = 2\tan^{-1}\left(\frac{B(\psi) + K\sqrt{B(\psi)^2 - A(\psi)C(\psi)}}{A(\psi)}\right) \tag{2.6}$$

where $K$ is the branch index of linkage configurations It is defined as the sign of the sine of the transmission angle, $\mu$, as shown in Fig 2.1 The reason why $K$ is introduced is that branching problems may occur in the synthesis procedure That is, for a certain input angle $\psi_k$, the coupler point $P_k$ attains two conjugate positions, which correspond to the two different roots obtained from eq (2 6) when its discriminant is greater than zero. In order to avoid branch jumping, $K$ is then used to distinguishing between linkage configurations (Ma and Angeles, 1987)

Similarly, to obtain the input angle in terms of the output angle, we rewrite eq (2 1) as follows

$$D(\phi)S^2 + 2E(\phi)S + F(\phi) = 0 \tag{2.7}$$

where

$$D(\phi) = k_1 - k_2 - (k_4 + k_3)\cos\phi \tag{2.8a}$$

$$E(\phi) = \sin\phi \tag{2.8b}$$

$$F(\phi) = k_1 + k_2 - (k_4 - k_3)\cos\phi \tag{2.8c}$$

$$S = \tan(\psi/2) \tag{2.8d}$$

and an expression similar to eq (2.6) for the input angle can then be obtained.

Once the dimensions of a linkage are given, the position of the coupler point becomes a function of the input angle. As the input angle varies in its mobility range, the coupler point moves on the surface of the sphere and the coupler curve is generated.

### 2.1.2 Mobility Analysis

The mobility of linkages has been intensively studied by various researchers since Grashof first proposed a set of inequalities for its analysis in the planar case (Grashof, 1883).

which is widely known as Grashof's criteria. Different methods concerning the mobility analysis of spherical linkages, both algebraic and geometrical, have been devised in the past (Duditza and Dittrich, 1969, Soni and Harrisberg, 1967; Savage and Hall, 1970, Gupta, 1986) Here, the mobility range of the spherical four-bar linkage is analyzed based on the concept of *linkage discriminant*, which was first proposed by Angeles and Callejas (1984) and has been further studied by different researchers (Williams and Reinholtz, 1986 & 1987, Angeles and Bernier, 1987, Gosselin and Angeles, 1988) The following approach is based on the input-output function defined by eq (2.1) and is an extension to the one presented by Gosselin and Angeles (1988) Here, the new definition of **k** in eqs (2 2*a-d*) is used

Now, the discriminant of eq (2 4) is written as

$$\Delta(\cos\psi) \equiv B^2(\psi) - A(\psi)C(\psi) \tag{2.9a}$$

$$= (k_3^2 - k_2^2 - 1)\cos^2\psi + 2(-k_1k_2 - k_3k_4)\cos\psi + (1 - k_1^2 + k_4^2) \tag{2.9b}$$

where the coefficient of the quadratic term is negative definite, i.e ,

$$k_3^2 - k_2^2 - 1 = -\frac{\sin^2\alpha_1}{\sin^2\alpha_4} < 0 \tag{2.10}$$

which means that eq.(2.9*b*) represents a parabola in $\cos\psi$ with negative curvature The condition for the input link to have full mobility is that the linkage discriminant be positive for every value of $\cos\psi$, i.e., for $-1 < \cos\psi < 1$ This is equivalent to saying that $\Delta(-1) \geq 0$ and $\Delta(1) > 0$, which yields the following inequalities

$$(k_2 + k_1)^2 < (k_3 - k_4)^2 \tag{2.11a}$$

$$(k_2 - k_1)^2 < (k_3 + k_4)^2 \tag{2.11b}$$

or

$$k_1 + k_2 \leq |k_3 - k_4| \tag{2.12a}$$

$$k_1 + k_2 \geq -|k_3 - k_4| \tag{2.12b}$$

$$k_2 - k_1 \leq |k_3 + k_4| \tag{2.12c}$$

$$k_2 - k_1 \geq -|k_3 + k_4| \tag{2.12d}$$

The foregoing inequalities define the full mobility region of the input link in the 4-D space of linkage parameters. It is apparent that each of the above inequalities represents a half-space separated by a 4-D plane, once the signs of $k_3 - k_4$ and $k_3 + k_4$ are determined. The common part of the these half-spaces is the full mobility region of the linkage. If a linkage falls into this region, i.e., if all the above inequalities are met, then the linkage has an input crank, otherwise, its input link is a rocker.

For the output link, we repeat the above analysis using eq.(2.7). The new linkage discriminant is written as

$$\delta(\cos\phi) = D^2(\varphi) - E(\phi)F(\phi) \tag{2.13a}$$

$$= (k_3^2 - k_4^2 - 1)\cos^2\varphi + 2(k_1 k_4 + k_2 k_3)\cos\phi + (1 - k_1^2 + k_2^2) \tag{2.13b}$$

Again, the coefficient of the quadratic term in eq.(2.11) is negative definite, i.e.,

$$k_3^2 - k_4^2 - 1 = -\frac{\sin^2\alpha_1}{\sin^2\alpha_2} < 0 \tag{2.13b}$$

The full mobility condition of the output link is expressed as $\delta(-1) \geq 0$ and $\delta(1) \geq 0$, which leads to the following inequalities.

$$(k_1 - k_4)^2 \leq (k_2 + k_3)^2 \tag{2.15a}$$

$$(k_1 + k_4)^2 \leq (k_2 - k_3)^2 \tag{2.15b}$$

or

$$k_1 - k_4 \leq |k_2 + k_3| \tag{2.16a}$$

$$k_1 - k_4 \geq -|k_2 + k_3 \tag{2.16b}$$

$$k_1 + k_4 \leq |k_2 - k_3| \tag{2.16c}$$

$$k_1 + k_4 \geq -|k_2 - k_3 \tag{2.16d}$$

These inequalities form the full mobility region of the 4-D space of linkage parameters in which the output link is a crank. For an output crank, the linkage should be inside this

region so that all the inequalities in eqs.(2.16$a$-$d$) are satisfied Otherwise, the output link is a rocker or the linkage is unfeasible.

For identifying the linkage type, the common part of the full-mobility regions of both the input and the output links have to be found, inside of which the linkage is a double-crank, otherwise, the linkage is a crank-rocker, rocker-crank, double-rocker or unfeasible, depending on where the linkage is located in the space of linkage parameters. A diagram illustrating this relationship is given in Fig. 2.2.

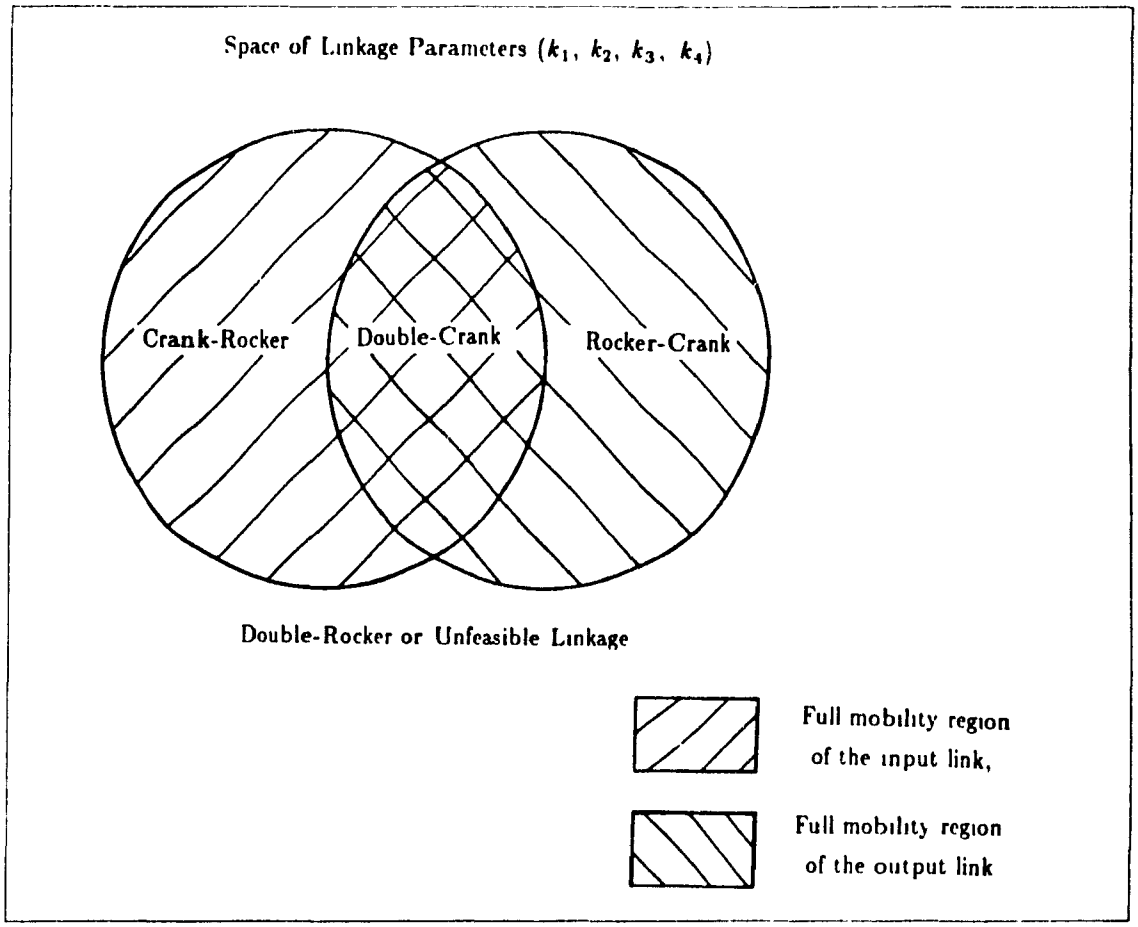


**Figure 2.2** Diagram of Linkage Type Identification

Based on the above analysis, a mobility criterion of spherical four-bar linkages can be obtained. This criterion arises directly from the inequalities (2.12$a$-$d$) and (2.16$a$-$d$). The detailed analysis can be found in Appendix A. Here, the result is presented:

The input link is a crank if the following conditions are met:

$$\min(k_3,\ k_4) \leq s \leq \max(k_3,\ k_4)$$

$$s \geq \max(k_1,\ k_2), \quad \text{if } k_3 + k_4 \geq 0, \tag{2.17a}$$

$$s < \min(k_1,\ k_2), \quad \text{if } k_3 + k_4 < 0$$

where $s = (k_1 + k_2 + k_3 + k_4)/2$ The conditions for the output link to be a crank are the following:

$$\min(k_2,\ k_3) \leq s \leq \max(k_2,\ k_3)$$

$$s \geq \max(k_1,\ k_4), \quad \text{if } k_2 + k_3 \geq 0; \tag{2.17b}$$

$$s < \min(k_1,\ k_4), \quad \text{if } k_2 + k_3 < 0$$

It is apparent that if both eqs.(2.17$a$) and (2.17$b$) are satisfied, the linkage is a double-crank; the linkage is a crank-rocker or rocker-crank if only eq.(2.17a) or (2.17b) is met; the linkage is a double-rocker or unfeasible otherwise.

## 2.2 On Constrained Least-square Optimization

The purpose of this Section is to introduce the problem of constrained least-square optimization and outline its method of solution. The introduction of this method is crucial in computing solutions of the problem at hand.

### 2.2.1 General Description

Constrained least-square optimization problems are a special case of nonlinear programming problems, which can be stated as follows:

$$\min z(\mathbf{x}) \tag{2.18}$$

subject to

$$g_i(\mathbf{x}) = 0, \quad i = 1, 2, \ldots, p \tag{2.19a}$$

and

$$h_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \ldots, q \tag{2.19b}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ is the vector of design variables, and $z(\mathbf{x})$ is the objective function to be minimized by a proper choice of $\mathbf{x}$. For least-square optimization problems, the objective function has a quadratic form, namely, $z(\mathbf{x}) = \mathbf{e}(\mathbf{x})^T \mathbf{W} \mathbf{e}(\mathbf{x})$, where $\mathbf{W}$ is a positive-definite weighting matrix. Here, the problem is expressed in a general way so that both equality and inequality constraints are included. As a rule, equality constraints are easier to handle than inequality constraints. Therefore, it is important to formulate a design problem, if possible, using only equality constraints. In fact, inequality constraints can be avoided for many cases by properly formulating the optimization problem to be solved.

There are mainly two types of method for solving constrained least-square optimization problems, namely, the unconstrained approach and the direct method. In the former, the most common way is to transform design variables so that constraints are eliminated or satisfied automatically. The problem is therefore transformed into an unconstrained one. However, real problems are usually too complex for this approach and penalty function methods are often used (Fiacco and McCormick, 1964, Zangwill, 1967), although this method sometimes has limitations in connection with equality-constrained problems. Direct methods include those that deal with the constraints directly as limiting hypersurfaces in the design space. We have, in this category, the method of feasible directions (Zoutendijk, 1960, Fox, 1971), the gradient projection method (Rosen, 1960) and the sequence of linear programming problems (Zhou and Mayne, 1985, Schittkowski, 1985).

As related to kinematic synthesis, least-square optimization techniques have been proven to be a powerful tool and have been used for solving various synthesis problems, such as those reported by Chi-Yeh (1966), Lewis and Falkenhagen (1968) and Angeles, Alivizatos and Akhras (1987). In this approach, the error between the desired output and the actual output for a certain problem is first squared, the objective function then being

defined as the sum of the resulting squares. However, for the following design stages, no single method exists which is best suited to all problems formulated in this way. Discussed in the next section is a newly developed method, the orthogonal-decomposition algorithm, which will be used to solve our problem

### 2.2.2 The Orthogonal-Decomposition Algorithm

A detailed description of the orthogonal-decomposition algorithm is given in (Angeles, Anderson and Gosselin, 1987). As a quick reference, it is briefly described here, the aspects important to our problem being emphasized.

As a direct method, the algorithm consists of solving the original nonlinear programming problem as a sequence of linear quadratic programming problems. A positive definite quadratic performance index of an $m$-dimensional nonlinear vector function $\mathbf{f}(\mathbf{x})$ is included in the problem statement, where $\mathbf{x}$ is an $n$ dimensional vector of design variables. Then, the objective function is written in the following form:

$$z(\mathbf{x}) = \frac{1}{2}\mathbf{f}(\mathbf{x})^T\mathbf{W}\mathbf{f}(\mathbf{x}) \tag{1.20}$$

where $\mathbf{W}$ is a constant positive definite $m \times m$ matrix. The design vector is subject to a set of $p$ nonlinear equality constraints represented as $\mathbf{g}(\mathbf{x}) = \mathbf{0}$. Further assumptions are made that $m$ is greater than $n - p$ and both $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ are continuous and differentiable functions of $\mathbf{x}$. An iterative scheme is used in the algorithm, in which both the objective function and constraint functions are linearized in each iteration, which allows the application of the techniques for linear quadratic programming problems. That is, the following linear quadratic programming problem is solved at each iteration:

$$\min_{\mathbf{x}} \frac{1}{2}\mathbf{e}^T\mathbf{W}\mathbf{e} \tag{1.21}$$

subject to

$$\mathbf{C}\mathbf{x} = \mathbf{d} \tag{1.22a}$$

with

$$\mathbf{e} = \mathbf{b} - \mathbf{A}\mathbf{x} \tag{1.22b}$$

where **b** and **d** are $n$- and $p$- dimensional vectors, respectively, and **A** and **C** are $m \times n$ and $p \times n$ matrices, respectively. Here, an orthogonal complement of the matrix **C** representing the equality constraints is computed efficiently so that the solution space is decomposed into two orthogonal components one lying in the nullspace of **C** and one lying in the range of $\mathbf{C}^T$ This makes the method computationally simpler than others such as the one reported by Betts (1980).

The orthogonal-decomposition algorithm is well suited for our problem, which is formulated in the same way as stated above. As will be seen in the next chapter, the objective function, as well as the constraint functions, are formed in such a way that they are both continuous and differentiable. As for the condition of $m \geq n - p$, this is actually met since we are aiming at solving the approximate synthesis problem concerning more than nine points Thus whereas $n$ and $p$ are fixed, $m$ can be any number greater than nine.

# Chapter 3 Problem Formulation

## 3.1 General Considerations

The synthesis problem under study consists of finding all the relevant parameters of a spherical four-bar linkage with a coupler point tracing a path lying a minimum distance away from a set of given points on the surface of a sphere. If $m+1$ points $\{Q_k\}_0^m$ are given, the coupler point should attain $m+1$ positions $P_k$, for $k = 0,1,2,...,m$, that lie as close as possible from the given set.

As discussed in the previous chapter, we need seven angles, $\alpha_1, \alpha_2, ..., \alpha_7$, to define the dimensions of a linkage and its coupler point. Moreover, to describe the location of the linkage, the positions of the fixed joints, $A$ and $D$, need to be known For this purpose, we can specify the orientation of a coordinate frame rigidly attached to the fixed link $AD$, which will be defined presently, in terms of the position vectors of both joints $A$ and $D$ Actually, we need eight independent parameters to uniquely define a spherical four-bar linkage, whether they are angles or Cartesian coordinates. As far as the design parameters are concerned, including angular quantities is not convenient. On the one hand, inequality constrains would have to be introduced since the angle values are bounded as:

$$0 < \alpha_i < 2\pi, \qquad i = 1,2,\ldots,7$$

and the sum of adjacent angles must not exceed $2\pi$. As discussed in Chapter 2, inequality constrains like these will no doubt increase the complexity of the problem and will render

its solutions discontinuous functions of the problem parameters. On the other hand, due to the special characteristics of spherical linkages, the length of each link can be replaced by its supplementary arc on the sphere without affecting the kinematic behaviour of the linkage. In other words, a certain spherical four-bar linkage can be defined by different values of the $\alpha$ angles In fact, for any spherical four-bar linkage, fifteen additional linkages exist, which are kinematically equivalent to it (Chiang, 1984) Therefore, other variables than angles should be defined

The problems mentioned above can be avoided if we simply use the position vectors of the four joint centres as design variables Once these four vectors are evaluated, the link lengths, or the arcs that connect adjacent joints, can be readily determined Then, the relevant $\alpha$ angles are computed from the foregoing vectors Hence, the design variables are chosen as the Cartesian coordinates of the four joint centres. We group them in a design vector $\mathbf{x}$ defined as.

$$\mathbf{x} = [\mathbf{a}^T,\ \mathbf{b}^T,\ \mathbf{c}^T,\ \mathbf{d}^T]^T \tag{3.1}$$

where $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ and $\mathbf{d}$ are the position vectors of points $A$, $B$, $C$ and $D$, respectively. i.e.,

$$\mathbf{a} = [x_A,\ y_A,\ z_A]^T \tag{3.2a}$$

$$\mathbf{b} = [x_B,\ y_B,\ z_B]^T, \tag{3.2b}$$

$$\mathbf{c} = [x_C,\ y_C,\ z_C]^T, \tag{3.2c}$$

$$\mathbf{d} = [x_D,\ y_D,\ z_D]^T \tag{3.2d}$$

The components of $\mathbf{x}$ are not independent, for the four joints are located on the surface of a sphere, say the unit sphere, and their position vectors have all a unit magnitude. These vectors are, then, constrained as follows:

$$\mathbf{a}^T\mathbf{a} = 1 \tag{3.3a}$$

$$\mathbf{b}^T\mathbf{b} = 1 \tag{3.3b}$$

$$\mathbf{c}^T\mathbf{c} = 1 \tag{3.3c}$$

$$\mathbf{d}^T\mathbf{d} = 1 \tag{3.3d}$$

As for the coupler point, one of the given points, $Q_0$, is used to define it in the 0th configuration, namely, as $P_0 \equiv Q_0$. Here, 0 denotes the reference configuration. Therefore, although $m+1$ points are given on the surface of the sphere, only $m$ points have to be included in the optimization procedure, one point being met exactly.

When $\mathbf{x}$ is given a certain value in the optimization procedure, finding the generated coupler curve based on the chosen coupler point becomes crucial in solving the problem because these trajectory points, attaining the minimum error with the corresponding points in the given set $\{Q_k\}_1^m$, are needed for the following minimization steps. To ease the formulation, three coordinate systems, with the same origin $O$, are defined as shown in Fig. 3.1. The transformation of coordinates between each two frames is discussed in the next section.

## 3.2 Transformations Between Coordinate Frames

Now, reference is made to Fig. 3.1. The three coordinate frames are assigned in the following way. The design parameter $\mathbf{x}$, the coupler point $P$ and the given set $\{Q_k\}_1^m$ are defined in the first coordinate frame. The second frame is defined as follows. $X_2$ passes through $A$ and $Y_2$ lies in the plane determined by the position vectors of joints $A$ and $D$. The third coordinate system is defined in a similar way. $X_3$ passes through joint $B$ and $Y_3$ lies on the plane determined by the position vectors of joints $B$ and $C$. In doing so, the coupler point can be easily located in the third coordinate frame and, through linear transformations, expressed in the first coordinate frame.

The rotation matrices specifying the coordinate transformations between each two coordinate frames are next defined. For each two adjacent coordinate frames $i$ and $i+1$, a matrix, $\mathbf{Q}_i$, exists, which specifies the rotation between them (Angeles, 1988). We can write this relationship as follows

$$[\mathbf{p}]_i = \mathbf{Q}_i[\mathbf{p}]_{i+1}$$

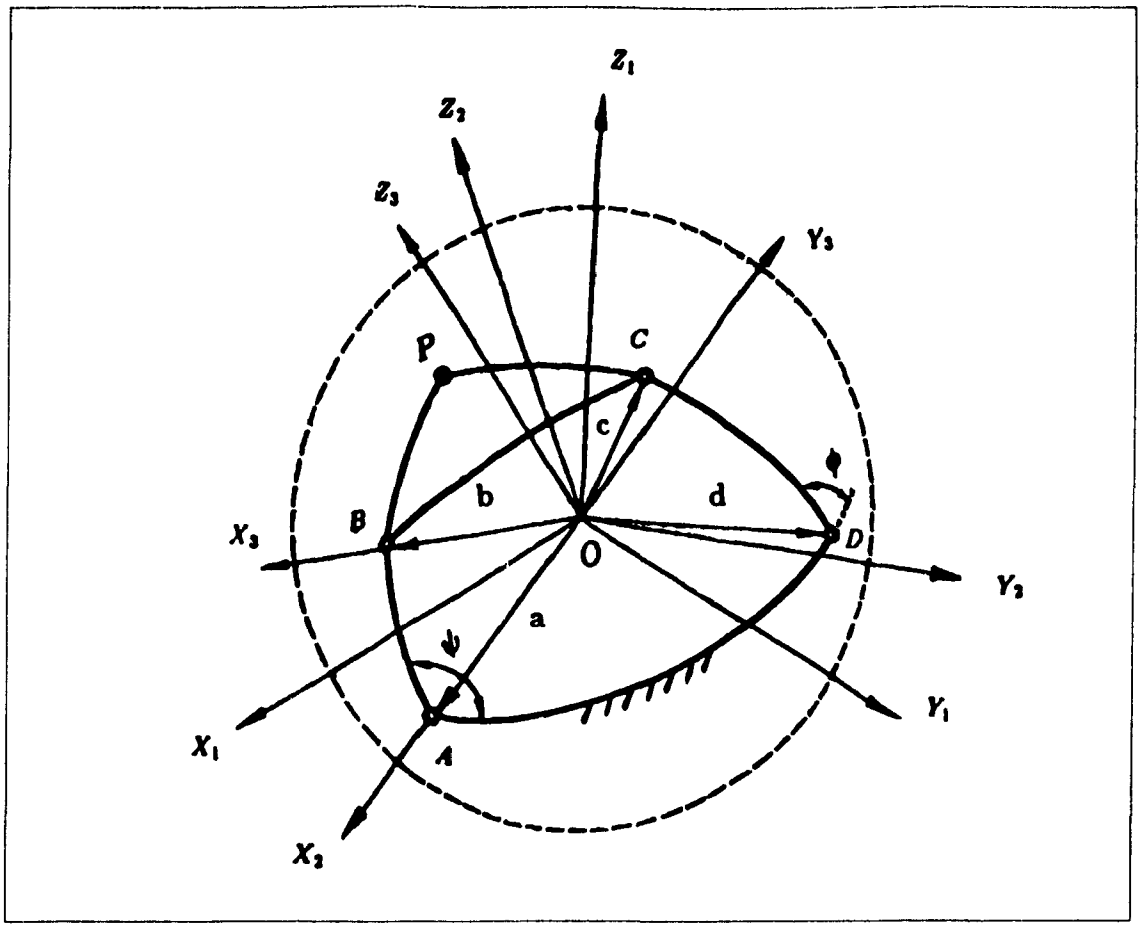


**Figure 3.1** Coordinate Systems Assigned to the Spherical Four-Bar Linkage

Now, let $\mathbf{i}_i$, $\mathbf{j}_i$ and $\mathbf{k}_i$ denote the unit vectors parallel to coordinate axes $X_i$, $Y_i$ and $Z_i$, for $i = 1,2,3$, respectively. The matrix transforming coordinates from the second coordinate system to the first one, expressed in the first coordinate system, is then defined as

$$[\mathbf{Q}_1]_1 = \begin{bmatrix} \mathbf{i}_2 \cdot \mathbf{i}_1 & \mathbf{j}_2 \cdot \mathbf{i}_1 & \mathbf{k}_2 \cdot \mathbf{i}_1 \\ \mathbf{i}_2 \cdot \mathbf{j}_1 & \mathbf{j}_2 \cdot \mathbf{j}_1 & \mathbf{k}_2 \cdot \mathbf{j}_1 \\ \mathbf{i}_2 \cdot \mathbf{k}_1 & \mathbf{j}_2 \cdot \mathbf{k}_1 & \mathbf{k}_2 \cdot \mathbf{k}_1 \end{bmatrix} \tag{3.4}$$

and the matrix transforming coordinates from the third coordinate system to the second one, expressed in the second coordinate system, is defined as:

$$[\mathbf{Q}_2]_2 = \begin{bmatrix} \mathbf{i}_3 \cdot \mathbf{i}_2 & \mathbf{j}_3 \cdot \mathbf{i}_2 & \mathbf{k}_3 \cdot \mathbf{i}_2 \\ \mathbf{i}_3 \cdot \mathbf{j}_2 & \mathbf{j}_3 \cdot \mathbf{j}_2 & \mathbf{k}_3 \cdot \mathbf{j}_2 \\ \mathbf{i}_3 \cdot \mathbf{k}_2 & \mathbf{j}_3 \cdot \mathbf{k}_2 & \mathbf{k}_3 \cdot \mathbf{k}_2 \end{bmatrix} \tag{3.5}$$

For brevity, $[\mathbf{Q}_1]_1$ and $[\mathbf{Q}_2]_2$ will be written as $\mathbf{Q}_1$ and $\mathbf{Q}_2$, respectively, in what follows. Moreover, If $[\mathbf{p}]_3$ denotes the array containing the Cartesian coordinates of the

coupler point in the third coordinate system, then that containing its coordinates in the first coordinate frame can be written as:

$$[\mathbf{p}]_1 = \mathbf{Q}_1\mathbf{Q}_2[\mathbf{p}]_3 \tag{3.6}$$

Obviously, $[\mathbf{p}]_1$ is a function of both $\psi$ and $\mathbf{x}$. For a given linkage, $[\mathbf{p}]_1$ will depend only on the input angle $\psi$. As the input angle $\psi$ varies within its mobility range, the coupler curve is generated. So, eq (3.6) is actually the equation of the coupler curve of a spherical four-bar linkage.

## 3.3 Setting Up the Optimization Procedure

From the coupler curve generated based on eq.(3.6), with the coupler point defined as $P_0 \equiv Q_0$, we can always find a set of points $\{P_k\}_1^m$ which lie the closest from the given set $\{Q_k\}_1^m$. The sum of the distances between the two sets $\{P_k\}_1^m$ and $\{Q_k\}_1^m$ is henceforth denoted as the linkage error. Moreover, in general, the dimensions of the linkage can be corrected so that the linkage error becomes smaller. This forms the basic idea in solving the optimization problem at hand. In fact, the whole synthesis procedure consists of two layers of optimization. Suppose we have defined a function $z(\mathbf{x}, \psi)$ representing the linkage error. We will first fix the dimensions of the linkage by setting $\mathbf{x} = \mathbf{x}^r$ and minimize its error over the input angle, $\psi$, i.e.,

$$\min_{\psi} z(\mathbf{x}^r, \psi) \tag{3.7}$$

which will produce a set of input angles $\psi_k$, for $k = 1, 2, ...m$, corresponding to the configurations under which the coupler point lies closest to the given points. For convenience, the set $\{\psi_k\}_1^m$ is grouped in an $m$-dimensional vector $\mathbf{t}_\psi$. Next, the outer layer of optimization will be performed on the linkage error over $\mathbf{x}$ as follows:

$$\min_{\mathbf{x}} z(\mathbf{x}, \mathbf{t}_\psi) \tag{3.8a}$$

subject to

$$\mathbf{g}(\mathbf{x}) = \mathbf{0} \tag{3.8b}$$

where $\mathbf{g}(\mathbf{x})$ is a set of constraints to be defined in Chapter 4. A new $\mathbf{x}$ is then computed so that the linkage error decreases.

Thus, we can accumulate the above two optimization layers and state the whole procedure as follows:

$$\min_{\mathbf{x}} \min_{\mathbf{t}_{\psi}} z(\mathbf{x}, \mathbf{t}_{\psi}) \tag{3.9a}$$

subject to

$$\mathbf{g}(\mathbf{x}) = \mathbf{0} \tag{3.9b}$$

Since we will write $z(\mathbf{x}, \mathbf{t}_{\psi})$ in quadratic form, eqs.(3.9$a$) and (3.9$b$) define a least-square optimization problem with equality constraints. The two layers of optimization involved will be treated separately, using different numerical schemes, the details of which are discussed in the following chapters.

# Chapter 4 Optimization Procedure

## 4.1 Introduction

In this chapter, the optimization procedure for the synthesis problem is introduced. As mentioned in the previous chapters, the whole procedure consists of two layers of optimizations and they are dealt with separately. In the inner layer, the objective function representing the linkage error is minimized over the input angle $\psi$, with linkage dimensions fixed for a given iteration of the outer layer of optimization. This requires the solution of a set of *nonlinear equations*. The output of this is the set of input angles defining the linkage configurations under which the coupler point attains the closest position from a given point. The outer layer, on the other hand, minimizes the linkage error over the design vector, $\mathbf{x}$, which involves an iterative scheme. The correction to $\mathbf{x}$ is computed in each iteration so that the linkage error becomes smaller and smaller until the convergence criterion, as yet to be derived, is met. The orthogonal-decomposition algorithm, as introduced in Chapter 2, is used to deal with this layer of optimization. Numerical techniques enhancing the convergence of the procedure, *namely, damping and continuation*, are introduced, as discussed in detail in the next chapter. As the Jacobian matrix of the objective function plays a very important role in the optimization, its evaluation is given special attention and is included in this chapter.

## 4.2 The Inner Optimization Loop—Computation of the Input Angles $\{\psi_k\}_1^m$

Given the dimensions of a spherical four-bar linkage defined by the design vector **x** at certain iteration, along with the coupler point $P_0$ in the initial configuration, the set $\{\psi_k\}_1^m$ is to be determined

Let the position vector of the $k$th given point $Q_k$ be $\mathbf{q}_k$ and the associated closest trajectory point be $\mathbf{p}_k$, which is to be found via angle $\psi_k$. Here, all vectors are represented in the first coordinate system For simplicity, the subscript 1 is dropped Their components are written as follows.

$$\mathbf{q}_k = \begin{bmatrix} \xi_k \\ \eta_k \\ \varsigma_k \end{bmatrix}, \qquad \mathbf{p}_k(\psi) = \begin{bmatrix} x_k(\psi) \\ y_k(\psi) \\ z_k(\psi) \end{bmatrix} \tag{4.1}$$

From eq.(3.6), we can write $\mathbf{p}_k$ in the following form:

$$\mathbf{p}_k(\psi) = \mathbf{Q_1 Q_2 [p]}_3\Big|_{\psi=\psi_k} \tag{4.2}$$

Now, the error between $P_n$ and $Q_k$, $\mathbf{e}_k$, associated with a given configuration $k$, is expressed as follows.*

$$\mathbf{e}_k = \mathbf{p}_k(\psi) - \mathbf{q}_k = \begin{bmatrix} x_k - \xi_k \\ y_k - \eta_k \\ z_k - \varsigma_k \end{bmatrix} \tag{4.3}$$

and the Euclidean norm of the the error, $d_k$, is defined as the distance between points $P_k$ and $Q_k$ Thus,

$$d_k(\psi) = \|\mathbf{e}_k(\psi)\| = \sqrt{(x_k - \xi_k)^2 + (y_k - \eta_k)^2 + (z_k - \varsigma_k)^2} \tag{4.4}$$

We now define function $D_k(\psi)$ as one half of the square of the Euclidean norm of the error, i.e.,

$$D_k(\psi) = \frac{1}{2}\mathbf{e}_k{}^T\mathbf{e}_k = \frac{1}{2}[\mathbf{p}_k(\psi) - \mathbf{q}_k]^T[\mathbf{p}_k(\psi) - \mathbf{q}_k] \tag{4.5}$$

* For brevity, the argument $\psi$ of $x_k$, $y_k$ and $z_k$ is omitted henceforth

Furthermore, considering the special geometry of spherical linkages defined on a unit sphere, eq.(4.5) can be rewritten as follows.

$$D_k(\psi) = 1 - \mathbf{q}_k \cdot \mathbf{p}_k(\psi) \tag{4.6}$$

The problem is thus reduced to minimizing $D_k(\psi)$ over $\psi$. In order for function $D_k(\psi)$ to reach a minimum, its derivative with respect to the input angle $\psi$ must vanish, i.e.,

$$h_k(\psi) = D_k'(\psi) = -\mathbf{q}_k \cdot \frac{d\mathbf{p}_k(\psi)}{d\psi}\Big|_{\psi=\psi_k} = 0 \tag{4.7}$$

Upon substitution of eq.(4.2) into eq.(4.7) and noticing that only $\mathbf{Q}_2$ is a function of $\psi$, we obtain the following equation:

$$h_k(\psi) = -\mathbf{q}_k^T \cdot \left[\mathbf{Q}_1 \frac{d\mathbf{Q}_2}{d\psi}[\mathbf{p}]_3\right]_{\psi=\psi_k} = 0, \qquad k = 1, 2, \ldots, m \tag{4.8}$$

Equation (4.8) is called the *configuration normality condition*. It consists of $m$ equations for $k = 1, 2, \ldots m$. Their roots will give the set $\{\psi_k\}_1^m$. In other words, the $m$ equations in eq.(4.8) define $\psi_k$, for $i = 1, 2, \ldots, m$. In order to solve these equations for $\psi_k$, various methods can be employed. Among them, the secant method (Forsythe, Malcolm and Moler, 1977) is more convenient for this case, since $h_k'(\psi)$ is not required in this method. However, sometimes the roots may lie very close to dead points of the coupler trajectory and the secant method will fail to solve the problem. The golden section search method (Brent, 1973) will take over in this case, as discussed in detail in (Ma and Angeles, 1987).

## 4.3 The Outer Optimization Loop—Evaluating the Linkage Parameters

With the set $\{\psi_k\}_1^m$ found for a given design vector $\mathbf{x}^r$, a new vector $\mathbf{x}^{r+1}$ is sought so that the linkage error will decrease. The said procedure is performed iteratively: each time, a better $\mathbf{x}$ is found, until no further improvement is possible.

Now we define two $3m$ dimensional vectors $\mathbf{p}$ and $\mathbf{q}$ as follows:

$$\mathbf{p} = [\mathbf{p}_1^T,\ \mathbf{p}_2^T,\ \ldots, \mathbf{p}_m^T]^T \tag{4.9}$$

$$\mathbf{q} = [\mathbf{q}_1^T,\ \mathbf{q}_2^T,\ \ldots,\ \mathbf{q}_m^T]^T \tag{4.10}$$

The linkage error is therefore expressed as.

$$\mathbf{f}(\mathbf{x}) = \mathbf{p} - \mathbf{q} \tag{4.11}$$

and the performance index of the optimization problem is next defined as.

$$z(\mathbf{x}) = \frac{1}{2}\mathbf{f}(\mathbf{x})^T \mathbf{W} \mathbf{f}(\mathbf{x}) \tag{4.12}$$

where W is a positive-definite constant weighting matrix that accounts for normalization and allows the assignment of different relevance to each trajectory point. For example, if W is chosen as

$$\mathbf{W} = \frac{1}{3m}\mathbf{1} \tag{4.13}$$

where 1 is the $3m \times 3m$ identity matrix, then $z(\mathbf{x})$ in eq.(4.12) represents the RMS value of the position errors of all the given points

Since the components of the design vector $\mathbf{x}$ are not independent, as discussed in Chapter 3, geometric constraints have to be imposed on them for completing the optimization problem. Equations (3.3$a$-$d$) are used for this purpose. The constraint function is hence written as follows

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} \mathbf{a}^T\mathbf{a} - 1 \\ \mathbf{b}^T\mathbf{b} - 1 \\ \mathbf{c}^T\mathbf{c} - 1 \\ \mathbf{d}^T\mathbf{d} - 1 \end{bmatrix} = \mathbf{0} \tag{4.14}$$

Up to now, the optimization problem is fully defined by eqs.(4.11)-(4.14). Notice that the number of unknowns in these equations is $12 + 3m$, namely, the Cartesian coordinates of the four joints of the linkage grouped in the design vector $\mathbf{x}$ and the Cartesian coordinates of the $m$ points corresponding to the set $\{\psi_k\}_1^m$ and grouped in vector

**p**. However, only two components of the Cartesian coordinate of the point on the surface of the sphere are independent, and hence, the total number of independent unknowns in these equations is $12 + 3m - 4 - m = 8 + 2m$. Moreover, if we improve the condition that the linkage error $\mathbf{f}(\mathbf{x})$, as defined in eq.(4.11), vanish, then we establish $3m$ equations. If the number of independent unknowns is to equal the number of equations, then the system can be solved exactly. Thus, if $8 + 2m = 3m$, we have $m = 8$, which means that up to 9 positions can be met with zero error under this formulation. If $m > 9$, an overdetermined system arises and no exact solutions exist. Hence, optimization methods are adopted to find the approximate solutions to the problem.

The orthogonal-decomposition algorithm, as briefly outlined in Chapter 2, is employed to solve the foregoing optimization problem. To this end, we start from a given initial guess $\mathbf{x}^0$, and the sequence $\mathbf{x}^1$, $\mathbf{x}^2$, , ... $\mathbf{x}^k$, $\mathbf{x}^{k+1}$,.. is then generated as:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta\mathbf{x}^k \tag{4.15}$$

where the increment $\Delta\mathbf{x}^k$ is computed in each iteration as the solution of the following constrained linear least-square optimization problem.

$$\min_{\Delta\mathbf{x}^k} \frac{1}{2}(\mathbf{e}^k)^T \mathbf{W} \mathbf{e}^k \tag{4.16}$$

subject to

$$\mathbf{G}^k \Delta\mathbf{x}^k = -\mathbf{g}(\mathbf{x}^k) \tag{4.17}$$

with $\mathbf{e}^k$ defined as

$$\mathbf{e}^k = -\mathbf{f}(\mathbf{x}^k) - \mathbf{F}^k \Delta\mathbf{x}^k \tag{4.18}$$

and $\mathbf{F}^k$ and $\mathbf{G}^k$ represent the Jacobian matrices $\mathbf{F}$ and $\mathbf{G}$ of functions $\mathbf{f}$ and $\mathbf{g}$, with respect to $\mathbf{x}$, evaluated at $\mathbf{x} = \mathbf{x}^k$.

The foregoing scheme is repeated until the following convergence criterion, discussed in detail in (Angeles, Anderson and Gosselin, 1987), is met at the solution:

$$\|\Delta\mathbf{x}^k\| \leq \epsilon \tag{4.19}$$

where $\epsilon$ is a prescribed tolerance.

To stabilize the above numerical procedure and ensure a fast convergence rate, damping techniques are used in every iteration step. Continuation is also introduced to guarantee its convergence. The details of these techniques are discussed in Chapter 5.

## 4.4 The Jacobian Evaluation

In using the orthogonal-decomposition algorithm, the computation of the Jacobian matrices of both functions $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ are required. Let these matrices be denoted by $\mathbf{F}(\mathbf{x})$ and $\mathbf{G}(\mathbf{x})$.

From eq.(4.11), we have

$$\mathbf{F}(\mathbf{x}^k) = \frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}}|_{\psi=\psi_k} = \frac{d\mathbf{p}}{d\mathbf{x}}|_{\psi=\psi_k} \tag{4.20}$$

Moreover, from eqs (3.6) and (4.9), vector $\mathbf{p}$ can be written in the following form:

$$\mathbf{p} = \begin{bmatrix} [\mathbf{p}_1]_1 \\ [\mathbf{p}_2]_1 \\ \cdot \\ [\mathbf{p}_m]_1 \end{bmatrix} \tag{4.21}$$

Differentiating both sides of eq.(4.21) with respect to $\mathbf{x}$, the following is obtained:

$$\mathbf{F}(\mathbf{x}) = \begin{bmatrix} d[\mathbf{p}_1]_1/d\mathbf{x} \\ d[\mathbf{p}_2]_1/d\mathbf{x} \\ \vdots \\ d[\mathbf{p}_m]_1/d\mathbf{x} \end{bmatrix} \tag{4.22}$$

Based on eq.(3.6), the $k$th submatrix in eq.(4.22) is written as

$$\frac{d[\mathbf{p}_k]_1}{d\mathbf{x}} = \left[\frac{d\mathbf{Q}_1}{d\mathbf{x}}\mathbf{Q}_2[\mathbf{p}]_3 + \mathbf{Q}_1\frac{d\mathbf{Q}_2}{d\mathbf{x}}[\mathbf{p}]_3 + \mathbf{Q}_1\mathbf{Q}_2\frac{d[\mathbf{p}]_3}{d\mathbf{x}}\right]_{\psi=\psi_k} \tag{4.23}$$

Now, it is clear that $\mathbf{Q_1}$ is a function of $\mathbf{x}$ and $\alpha_1$; $\mathbf{Q_2}$ is a function of $\alpha_1$, $\alpha_2$, $\alpha_3$, $\alpha_4$, $\psi$ and $\phi$; and $[\mathbf{p}]_3$ is a function of only $\alpha_5$ and $\alpha_6$. Here, in order to enhance the numerical stability of the problem, instead of using angles $\alpha_i$, for $i = 1,2,...,7$, their sine and cosine functions are used in computing the right hand side of eq.(4.23). Hence, the three derivatives in eq.(4.23) are computed as follows:

$$\frac{d\mathbf{Q_1}}{d\mathbf{x}} = \frac{\partial \mathbf{Q_1}}{\partial \mathbf{x}} + \frac{\partial \mathbf{Q_1}}{\partial c_1}\frac{dc_1}{d\mathbf{x}} + \frac{\partial \mathbf{Q_1}}{\partial s_1}\frac{ds_1}{d\mathbf{x}} \tag{4.24}$$

$$\frac{d\mathbf{Q_2}}{d\mathbf{x}} = \sum_{i=1}^{4}\left(\frac{\partial \mathbf{Q_2}}{\partial c_i}\frac{dc_i}{d\mathbf{x}} + \frac{\partial \mathbf{Q_2}}{\partial s_i}\frac{ds_i}{d\mathbf{x}}\right) + \frac{\partial \mathbf{Q_2}}{\partial \psi}\frac{d\psi}{d\mathbf{x}} + \frac{\partial \mathbf{Q_2}}{\partial \phi}\frac{d\phi}{d\mathbf{x}} \tag{4.25}$$

$$\frac{d[\mathbf{p}]_3}{d\mathbf{x}} = \sum_{i=5}^{6}\left(\frac{\partial [\mathbf{p}]_3}{\partial c_i}\frac{dc_i}{d\mathbf{x}} + \frac{\partial [\mathbf{p}]_3}{\partial s_i}\frac{ds_i}{d\mathbf{x}}\right) \tag{4.26}$$

where $c_i \equiv \cos\alpha_i$ and $s_i \equiv \sin\alpha_i$, for $i = 1,2,...7$.

From the geometry of the general spherical four-bar linkage, we can readily express the sine and cosine functions of the $\alpha$ angles in terms of the dot and cross products of the position vectors of two adjacent joints, thereby enabling the computation of $dc_i/d\mathbf{x}$ and $ds_i/d\mathbf{x}$. In computing $d\psi/d\mathbf{x}$ and $d\phi/d\mathbf{x}$, we resort to both the normality condition, eq.(4.8), and the input-output function, eq.(2.1), which can be written in the following forms:

$$h(\psi,\ \phi,\ \{c_i\}_1^6,\ \{s_i\}_1^6,\ \mathbf{x}) = 0 \tag{4.27}$$

$$f(\psi,\ \phi,\ \{c_i\}_1^4,\ \{s_i\}_1^4) = 0 \tag{4.28}$$

Differentiating both sides of eqs.(4.27) and (4.28) with respect to $\mathbf{x}$, the following, is obtained:

$$\frac{\partial h}{\partial \psi}\frac{d\psi}{d\mathbf{x}} + \frac{\partial h}{\partial \phi}\frac{d\phi}{d\mathbf{x}} + \sum_{i=1}^{6}\left(\frac{\partial h}{\partial c_i}\frac{dc_i}{d\mathbf{x}} + \frac{\partial h}{\partial s_i}\frac{ds_i}{d\mathbf{x}}\right) + \frac{\partial h}{\partial \mathbf{x}} = 0 \tag{4.29}$$

$$\frac{\partial f}{\partial \psi}\frac{d\psi}{d\mathbf{x}} + \frac{\partial f}{\partial \phi}\frac{\partial \phi}{d\mathbf{x}} + \sum_{i=1}^{4}\left(\frac{\partial f}{\partial c_i}\frac{dc_i}{d\mathbf{x}} + \frac{\partial f}{\partial s_i}\frac{ds_i}{d\mathbf{x}}\right) = 0 \tag{4.30}$$

Next, eqs.(4.29) and (4.30) are written in matrix form as

$$\begin{bmatrix} a_{11}\mathbf{1} & a_{12}\mathbf{1} \\ a_{21}\mathbf{1} & a_{22}\mathbf{1} \end{bmatrix} \begin{bmatrix} d\psi/d\mathbf{x} \\ d\phi/d\mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \tag{4.31}$$

where

$$a_{11} = \frac{\partial h}{\partial \psi}, \quad a_{12} = \frac{\partial h}{\partial \phi}, \quad a_{21} = \frac{\partial f}{\partial \psi}, \quad a_{22} = \frac{\partial f}{\partial \phi} \tag{4.32}$$

and

$$\mathbf{b}_1 = -\frac{\partial h}{\partial \mathbf{x}} - \sum_{i=1}^{6} \left( \frac{\partial h}{\partial c_i} \frac{dc_i}{d\mathbf{x}} + \frac{\partial h}{\partial s_i} \frac{ds_i}{d\mathbf{x}} \right) \tag{4.33a}$$

$$\mathbf{b}_2 = -\sum_{i=1}^{4} \left( \frac{\partial f}{\partial c_i} \frac{dc_i}{d\mathbf{x}} + \frac{\partial f}{\partial s_i} \frac{ds_i}{d\mathbf{x}} \right) \tag{4.33b}$$

with $\mathbf{1}$ defined as the $12 \times 12$ identity matrix

From eq.(4.31), we obtain

$$\begin{bmatrix} d\psi/d\mathbf{x} \\ d\phi/d\mathbf{x} \end{bmatrix} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22}\mathbf{1} & -a_{12}\mathbf{1} \\ -a_{21}\mathbf{1} & a_{11}\mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \tag{4.35}$$

which yields

$$\frac{d\psi}{d\mathbf{x}} = \frac{a_{22}\mathbf{b}_1 - a_{12}\mathbf{b}_2}{a_{11}a_{22} - a_{12}a_{21}} \tag{4.36a}$$

$$\frac{d\phi}{d\mathbf{x}} = \frac{a_{11}\mathbf{b}_2 - a_{21}\mathbf{b}_1}{a_{11}a_{22} - a_{12}a_{21}} \tag{4.36b}$$

The detailed procedure of the above computations, as well as the computation concerning the Jacobian matrix of the constraint function, $\mathbf{G}(\mathbf{x}) = d\mathbf{g}(\mathbf{x})/d\mathbf{x}$, are included in Appendices C and D

# Chapter 5 Convergence Enhancement

## 5.1 Introduction

As pointed out in Chapter 2, various numerical schemes have been devised to solve least-square optimization problems with equality constraints. In this thesis, a newly-developed scheme, the orthogonal-decomposition algorithm, is employed. Nonlinear least-square optimization problems with equality constraints can be solved very efficiently with the orthogonal-decomposition algorithm Moreover, the algorithm converges very fast if the Jacobian matrices of both the objective function and constraint function are well-conditioned and the initial guess is reasonably close to the solution However, the said Jacobian matrices can sometimes become ill-conditioned, which will lead to divergence. This is a situation to be taken into account. Here, to measure the conditioning of these Jacobian matrices, the concept of condition number of a matrix is recalled (Golub and Van Loan, 1983), which measures the amplification of the roundoff error in solving a system of linear equations associated with that matrix. A well-conditioned matrix has a small condition number close to 1. If the condition number of the Jacobian matrix is high with respect to the roundoff error of the data, convergence will not be achieved unless suitable numerical means are used. In our case, the techniques of damping and continuation are applied. Damping is performed within each iteration by reducing the computed correction by a certain amount so that the procedure is stablized and attains a high convergence rate. Continuation, on the other hand, is used if an initial guess lying reasonably close to the solution is difficult

to estimate. It is applied to the whole procedure in such a way that the final solution is obtained step by step along a certain path from an initial guess, quite arbitrary With these techniques added, the problem is solved safely No doubt, these techniques are essential to the success of the aforementioned method, as will be discussed in this chapter.

## 5.2 Damping Techniques

Sometimes in the optimization procedure, the condition number of the Jacobian matrix of the objective function may become relatively high at certain iterations. This is the case when some of the joint centres move too far away from the surface of the sphere after the correction in certain iteration, or some $w_k$ values define a configuration close to a dead point. If this happens, the roundoff error becomes inadmissibly large and the computed correction to the design vector, $\Delta\mathbf{x}$, is meaningless This may lead to the violation of the constraints and some of the joints may go further apart from the surface of the unit sphere in the following iterations The procedure will finally diverge. Means to cope with this problem rely on damping techniques, which have been proved to be a useful tool (Dahlquist and Björck 1974; Angeles, 1982). We first reduce the correction to a certain amount to make the procedure stable, and then go to the next iteration This is achieved as follows. Instead of using eq.(4 15) directly, we use the following.

$$\mathbf{x}^{k+1} = \mathbf{x}^k + d\Delta\mathbf{x}^k \tag{5.1}$$

where $d$ is a scalar between 0 and 1, called the *damping factor*, which serves to reduce the magnitude of $\Delta\mathbf{x}^k$. When one damping is not enough to stablize the procedure, further dampings are performed until convergence is achieved Using damping techniques here becomes essential in the optimization procedure. Indeed, it keeps the procedure more stable and maintains a higher convergence rate. However, if the condition number of the Jacobian matrix becomes very high, damping will fail to work. In this case, the initial guess has to be reselected. To prevent the joints from going too far from the surface of the sphere, we also include a normalization step in each iteration to keep the magnitude of the position vectors of joint centres close to 1; we found that this provision enhances the convergence of

the procedure. Of course, the value of the damping coefficient is problem dependent. When the convergence behaviour is not so favourable, the damping value need to be changed, since the procedure is very sensitive to it. We can always achieve a fastest convergent rate, or a minimum number of iterations for obtaining a solution, with a proper choice of this value However, there are no general rules for obtaining this value In many cases, it is found through trial and error

In addition to a well-conditioned Jacobian matrix, an initial guess lying reasonably close to the solution is also important for the problem to converge Since, in general, an initial guess close to the solution is not available, a technique called *continuation* is introduced, the details of which are discussed in the next section

## 5.3 Continuation

Due to the difficulty in locating a suitable initial guess which lies close enough to the solution, continuation is necessary. Continuation is a powerful means to guide an initial guess step by step to the final solution, as has been widely recognized in numerical computations (Wasserstrom, 1973, Avila, 1974, Richter and DeCarlo, 1983). It works for problems whose solutions are continuous functions of the parameters of the problem. In our case, we propose the following scheme:

For the given set $\{Q_k\}_1^m$, we first choose an arbitrary initial guess $\mathbf{x}^0$ under which, together with the coupler point, $P_0 \equiv Q_0$, a coupler curve is generated on the surface of the sphere, and the corresponding trajectory points $\{P_k\}_1^m$, lying the closest to the relevant points in the given set $\{Q_k\}_1^m$, are found Now, we connect points $Q_k$ and $P_k$, for $k = 1, 2, \ldots, m$, with a major circular arc and compute the points $R_{ki}$, for $i = 1, 2, \ldots, l$, which divide the arc into $l$ parts, as shown in Fig. 5.1. Then, we start the numerical procedure from the current initial guess with $\{R_{k1}\}_1^m$ defined as the given points Since points $\{R_{k1}\}_1^m$ are very close to the corresponding points in the coupler curve associated with the initial linkage, if we choose $l$ large enough—say $l = 5$—the

procedure will converge to a solution. Then, using this solution as the initial guess of the next continuation step, we repeat the procedure and it will soon reach the next solution associated with the set $\{R_{k2}\}_1^m$. In this way, the procedure goes from one continuation step to the next one, until the original set $\{Q_k\}_1^m$ is met.

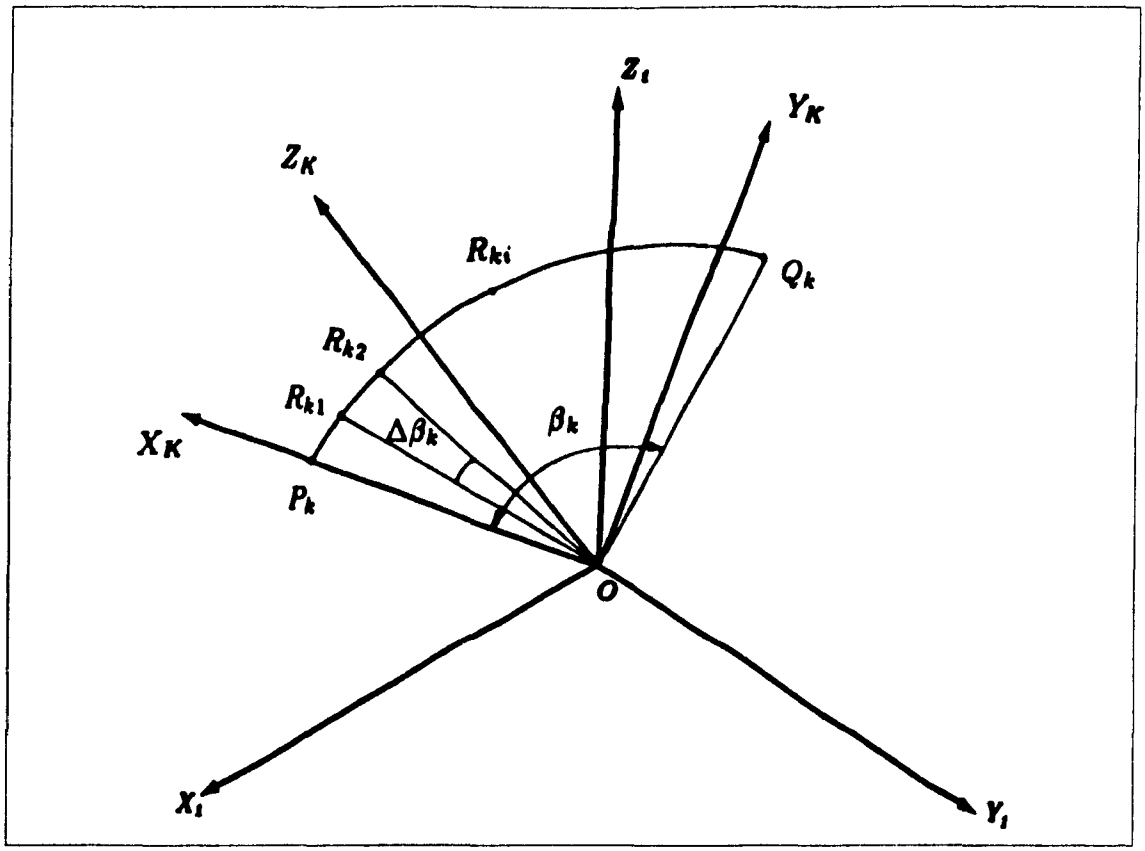


**Figure 5.1** Continuation Scheme

For simplicity, we divide the arc into equal parts and use the following scheme to compute the dividing points: First, a new coordinate system labeled $K$, with origin at $O$, is assigned to describe each arc connecting $P_k$ and $Q_k$, for $k = 1, 2, \ldots, m$, as shown in Fig. 5.1. The $X_k$ axis passes through $P_k$, whereas the $Y_k$ axis lies on the plane determined by points $P_k$, $Q_k$ and $O$. Hence, in the $K$ coordinate system, the arc connecting $P_k$ and $Q_k$ becomes a part of the following circle:

$$\begin{cases} x_k^2 + y_k^2 = 0 \\ z_k = 0 \end{cases}$$

If $\mathbf{r}_{ki}$ denotes the position vector of $R_{ki}$, we have

$$[\mathbf{r}_{ki}]_K = [\cos(i\Delta\beta_k),\ \sin(i\Delta\beta_k),\ 0]^T, \tag{5.2}$$

$$k = 1,2,\ \ldots,m$$

$$i = 1,2,\ldots,l$$

where

$$\beta_k = \cos^{-1}(\mathbf{p}_k \cdot \mathbf{q}_k) \tag{5.3}$$

$$\Delta\beta_k = \frac{\beta_k}{l} \tag{5.4}$$

To obtain an expression for $\mathbf{r}_{ki}$ in the first coordinate system, in which $\mathbf{p}_k$ and $\mathbf{q}_k$ are defined, the rotation matrix $\mathbf{R}$, which transforms coordinates from the $K$ coordinate system to the first coordinate system, is next computed.

$$\mathbf{R} = [\mathbf{p}_k,\ \mathbf{e},\ \mathbf{f}] \tag{5.5}$$

where

$$\mathbf{f} = \frac{\mathbf{p}_k \times \mathbf{q}_k}{\|\mathbf{p}_k \times \mathbf{q}_k\|} \tag{5.6}$$

$$\mathbf{e} = \mathbf{f} \times \mathbf{p}_k \tag{5.7}$$

and hence, we have

$$[\mathbf{r}_{ki}]_1 = \mathbf{R}[\mathbf{r}_{ki}]_K \tag{5.8}$$

thereby completing the computation of the dividing points.

The number of dividing points depend on the problem and the initial guess. It is always advisable to adjust the initial guess to make this number smaller, since involving too many continuation steps will lead to a computational procedure unnecessarily expensive

# Chapter 6 Examples

Based on the optimization scheme described in previous chapters, a general Fortran77 computer package, SPHER1, was developed, along with graphics software for visualizing the spherical four-bar linkage in a 3-D representation on Unix-based workstations. Several examples regarding the design of spherical four-bar mechanisms were solved with SPHER1 and their results are reported in this chapter. In solving practical problems with the aid of this package, the first step is to give a proper description of the problem in the context of synthesis of spherical four-bar path generators. That is, define the set of given trajectory points according to the problem and represent the linkage with the Cartesian coordinates of its four joints in a certain coordinate system. Then, with the help of the graphics facility, choose a suitable initial guess to start the numerical computation and, at the same time, plan the continuation procedure for the problem. Having done all this, the problem can be solved satisfactorily. Following the steps mentioned above, four design problems are solved, the results of which are presented next.

## 6.1 A Circular-Path-Tracking Spherical Four-Bar Linkage

This is originally a trial problem for testing the computer package. Actually, it is not necessary to use a four-bar linkage to trace a circular path, since a single link will do. The purposes of introducing this example is to outline the optimization procedure with a real problem and to discuss an interesting result. The circular path used in this example

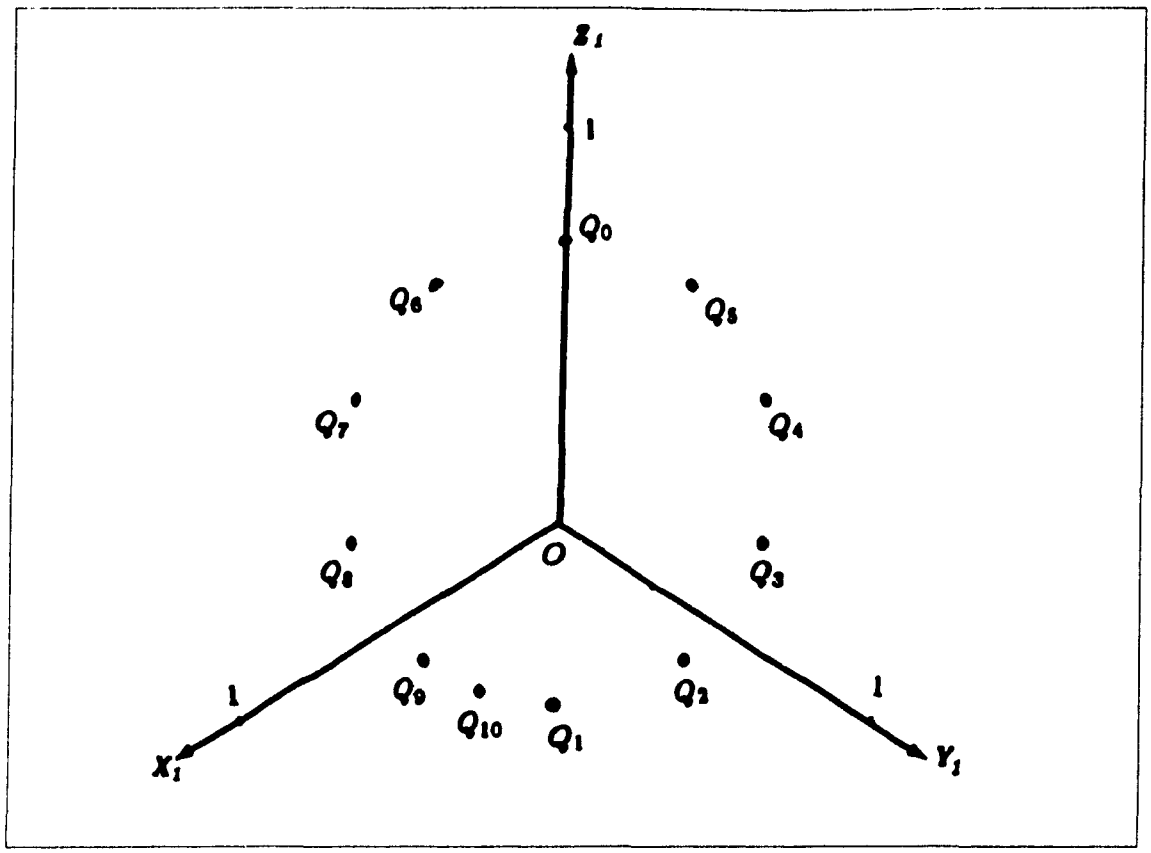


**Figure 6.1a** Points Selected from the Circular Path

is defined by the following two surfaces:

$$x^2 + y^2 + z^2 = 1$$

$$x + y + z = 4\sqrt{3}/5$$

where the circle has a radius of 0.8. Eleven points are selected from the circle as the given points, their coordinates being given as follows (Fig. 6.1a):

$Q_0$ : (0.1, 0.1, 0.989949),

$Q_1$ : (0.7, 0.7, 0.141421),

$Q_2$ : (0.393329, 0.892081, 0.222449),

$Q_3$ : (0.0892059, 0.892081, 0.434581),

$Q_4$ : (−0.096204, 0.710749, 0.696790),

$Q_5$ : (−0.092081, 0.406671, 0.908922),

$$Q_6 : (0.406671,\ -0,092081,\ 0,908922),$$

$$Q_7 : (0.710794,\ -0.096204,\ 0.696790),$$

$$Q_8 : (0.896204,\ 0.0892059,\ 0.434581),$$

$$Q_9 \quad (0.892081,\ 0\ 393329,\ 0.222449),$$

$$Q_{10} : (0.827015,\ 0.536801,\ 0.167001)$$

where the coupler point in its initial configuration is $P_0 \equiv Q_0$ Next, the initial guess is chosen Here, to avoid involving too many continuation steps in the procedure, an initial guess lying reasonably close to the solution is desired This is achieved by trial and error with the help of the graphics facility Different linkages were tried and adjusted until the one with the coupler curve comparatively close to the given points is found. In this example, a suitable initial guess found in this way is given as follows (Fig 6 1b)

$$\mathbf{a}^0 = [0.5491,\ 0\ 3936,\ 0.7373]^T$$

$$\mathbf{b}^0 = [0.3186,\ 0.0216,\ 0.99476]^T$$

$$\mathbf{c}^0 = [0.047,\ 0.4288,\ 0.9022]^T$$

$$\mathbf{d}^0 = [0.404,\ 0.56,\ 0.7237]^T$$

A three-step continuation scheme is used for the example to ensure convergence The solutions at the end of each continuation step are shown in Figs (6 1c) and (6 1d)

The optimum linkage obtained is defined by the following position vectors of joint centres (Fig. 6.1e)

$$\mathbf{a} = [0.539957,\ 0.561799,\ 0.626760]^T$$

$$\mathbf{b} = [0.305536,\ 0.00317,\ 0.952175]^T$$

$$\mathbf{c} = [0.09804,\ 0.416301,\ 0.903924]^T$$

$$\mathbf{d} = [0.516004,\ 0.534261,\ 0.669556]^T$$

Here the average number of iterations within each continuation step is around 15, the final solution containing a linkage error of the order of $10^{-4}$.

The interesting issue here is that, in the solution, joints $A$ and $D$ lie very close to each other. It is also found that, if we impose an even smaller tolerance on the procedure

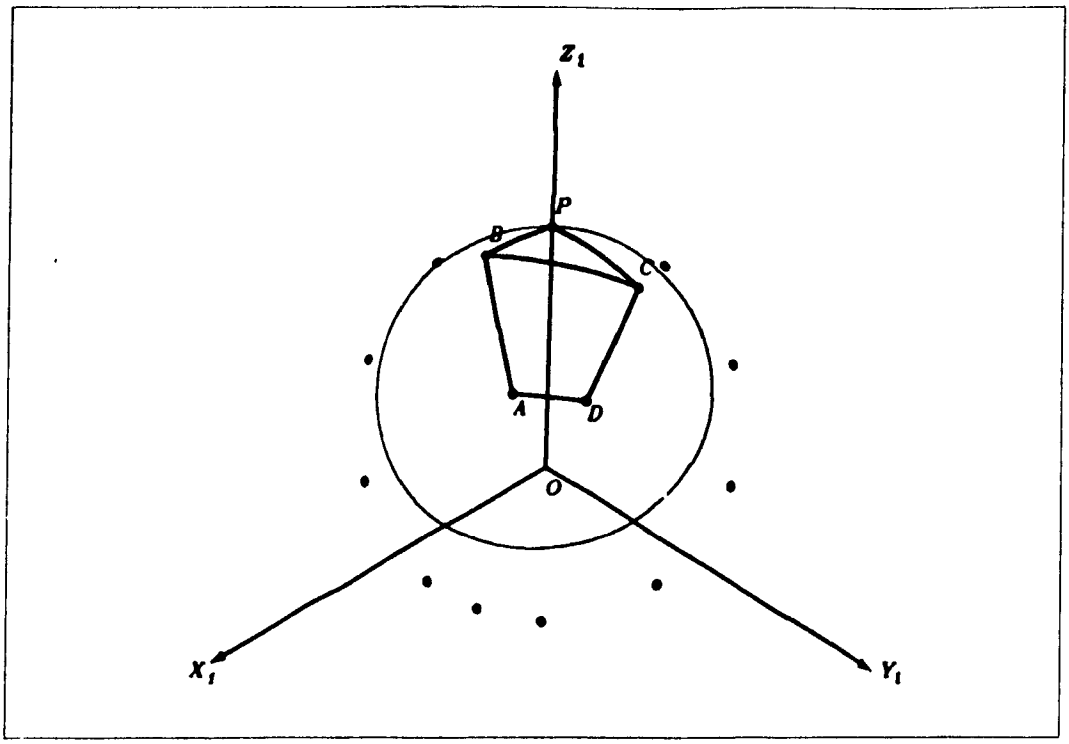


**Figure 6.1b** Initial Guess

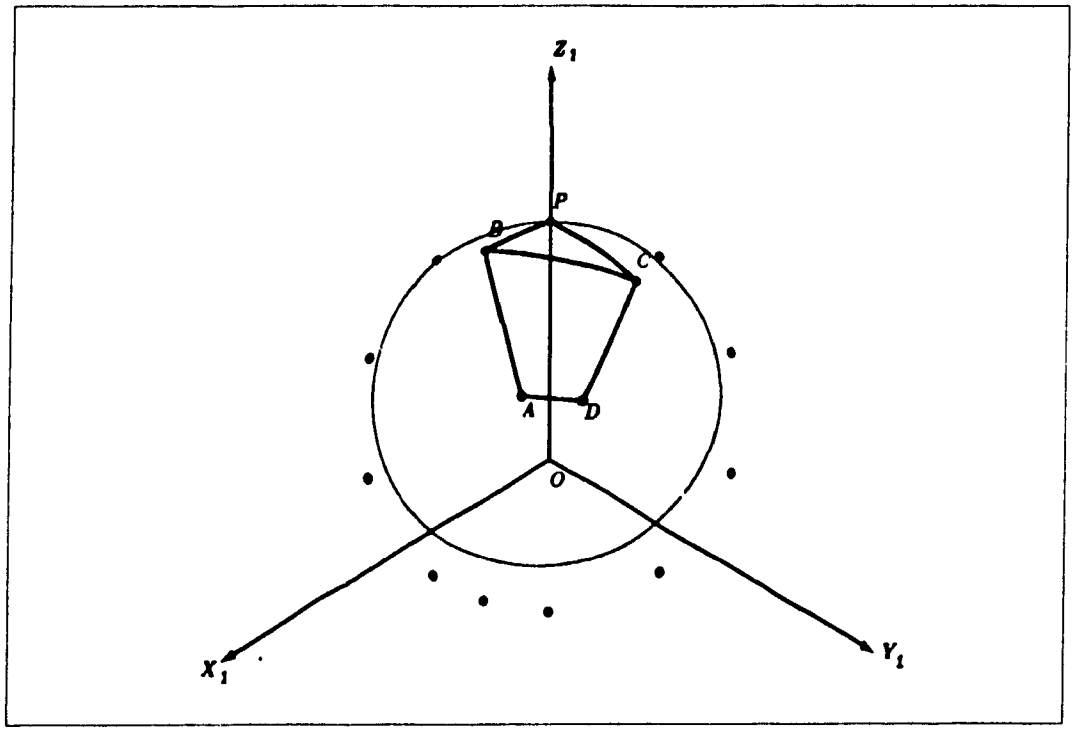


**Figure 6.1c** Solution at the End of the First Continuation Step

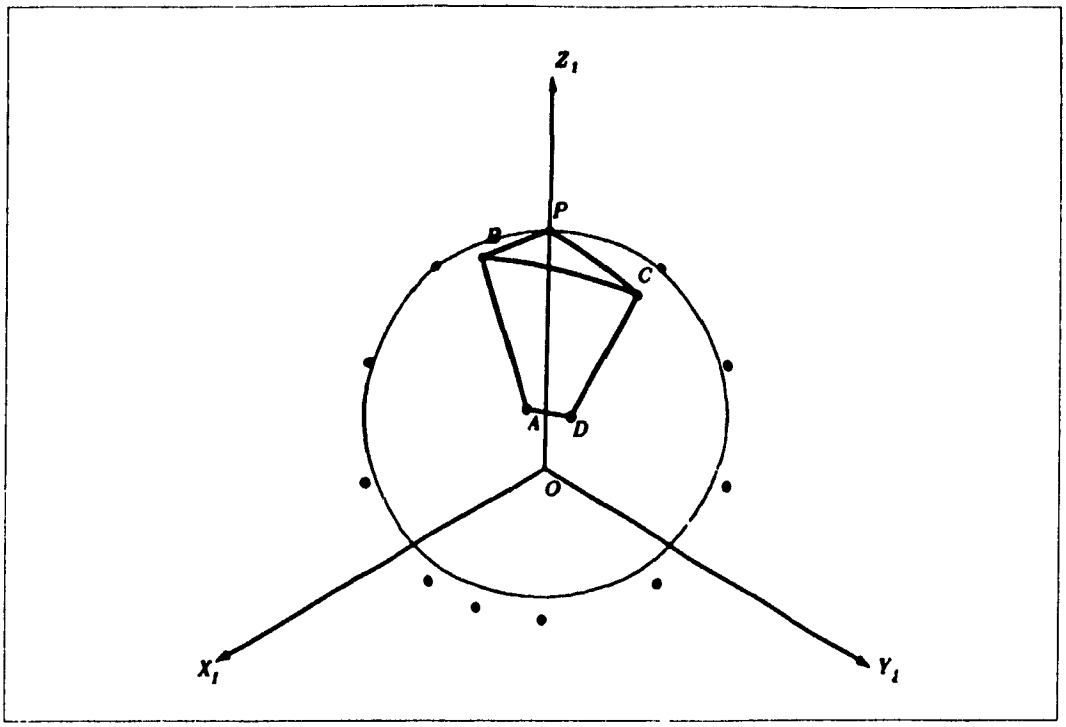


**Figure 6.1d** Solution at the End of the Second Continuation Step

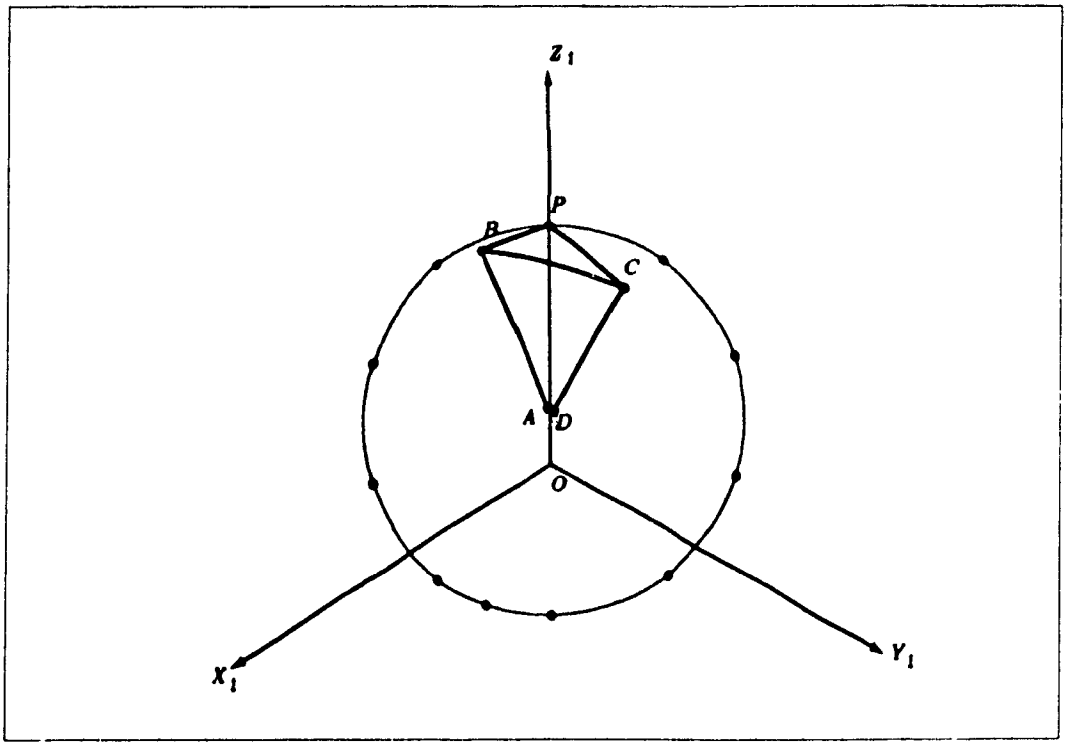


**Figure 6.1e** Final Solution

and allow more iterations. $A$ and $D$ will become closer and closer. In the limit, they meet at one point, namely, the centre of the circle. Then, the length of the fixed link vanishes and the four-bar linkage collapses into one single rigid-body which rotates about an axis passing through the centre of the circle This is reasonable, since this is a case in which the given points are met exactly by merely rotating a single rigid-body about a fixed axis, as pointed out at the beginning of the example

## 6.2 The Design of Solar-Tracking Mechanisms

### 6.2.1 A Spherical Mechanism Tracing a Solar Path on the Summer Solstice

The celestial sphere can be considered centred at a point $O$ on the surface of the earth. As the sun traverses the celestial sphere from sunrise to sunset, it describes a trajectory which can be projected onto a unit sphere centred at $O$. For a spherical mechanism on the surface of the earth, the centre of its sphere can be considered to be at $O$ as well. Our design of the solar-tracking mechanism takes place on the unit sphere.

The solar path is generated by using the following equations (Iqbal, 1983):

$$\sin\theta = \sin\varphi \sin\delta + \cos\varphi \cos\delta \cos\omega \tag{6.1}$$

$$\cos\gamma = \frac{\sin\theta \sin\varphi - \sin\delta}{\cos\theta \cos\varphi} \tag{6.2}$$

$$\sin\gamma = \frac{cos\delta \sin\omega}{\cos\theta} \tag{6.3}$$

where: $\theta$ is the solar altitude angle; $\gamma$ is the solar azimuth angle; $\varphi$ is the latitude of point $O$ on the earth; $\delta$ is the declination, and $\omega$ is the hour angle, which at noon is zero and in the morning, positive The solar path used in this example is based on Montreal (latitude of 45° north) on the day of the summer solstice ($\delta = 23.5°$). The original data obtained from the above equations are expressed in latitude and azimuth angles in a spherical coordinate

system. Here, to fit our method, they are transformed into a Cartesian coordinate system defined as follows: The $X_1$ axis is directed to the south; the $Y_1$ axis is directed to the east and the $Z_1$ axis is directed to the Zenith. Fourteen points are selected from the solar path as shown in Fig. 6.2a, starting from 8:30 a.m and ending at 3:00 p.m for every half an hour.

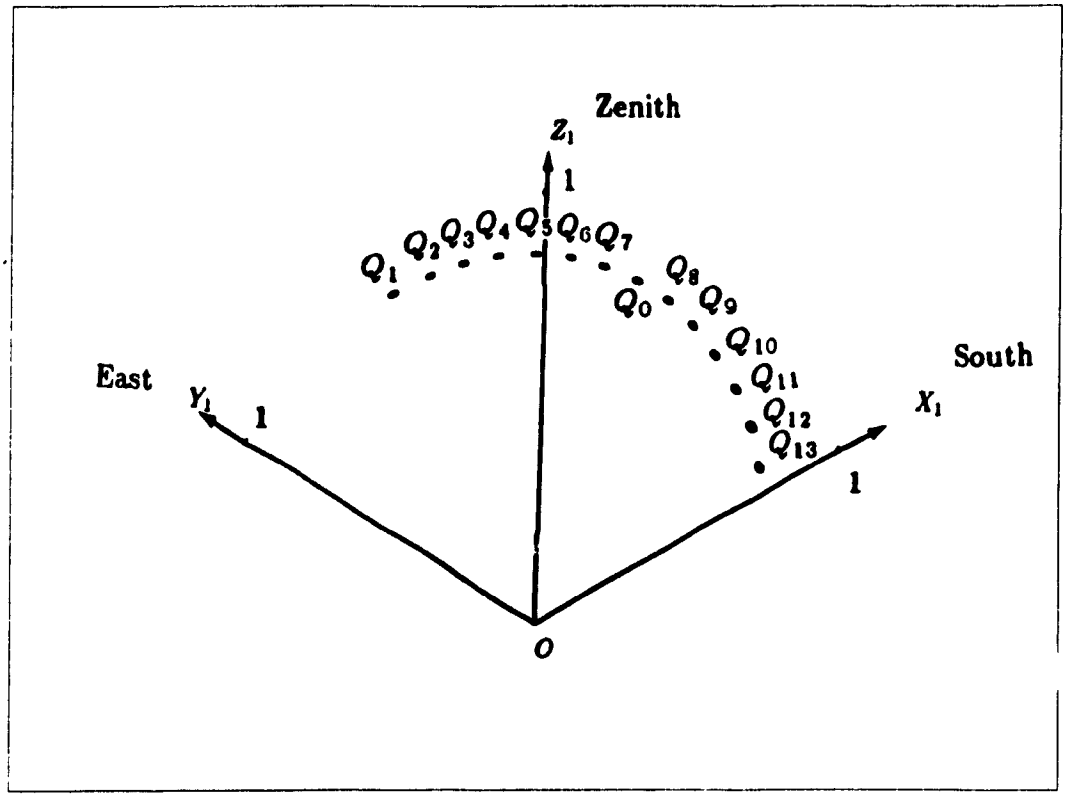


**Figure 6.2a** Points Selected from the Solar Path of Summer Solstice

One point, namely, the point which represents the solar position at noon, is chosen as the coupler point in its initial configuration, i.e., $P_0 \equiv Q_0$, and the remaining points form the given set, $\{Q_k\}_1^{13}$. They are given by the following coordinates:

$Q_0$ : (0.366501, 0, 0.930418),

$Q_1$ : (0.112799, 0.727553, 0.676715),

$Q_2$ : (0.176518, 0.648459, 0.740488),

$Q_3$ : (0.232499, 0.558271, 0.796416),

$Q_4$ : (0.279624, 0.458530, 0.843541),

$Q_5$ : (0.317140, 0.350944, 0.881057),

$Q_6$ : (0.344406, 0.237353, 0.908322),

$Q_7$ . (0.360954, 0.119701, 0.924870).

$Q_8$ · (0.360954, - 0.119701. 0.924870),

$Q_9$ . (0.344406, −0.237353. 0.908322),

$Q_{10}$ . (0.317140, −0.350944, 0.881057),

$Q_{11}$ (0.279624, −0 458530, 0.843541),

$Q_{12}$ : (0.232499, −0 558271. 0.796416).

$Q_{13}$ : (0.176518, −0.648459, 0 740488)

The initial guess is chosen next in the same way as we did in the last example. We then have the following (Fig. 6.2b):

$$\mathbf{a}^0 = [-0.75,\ 0.23,\ 0.62]^T$$

$$\mathbf{b}^0 = [0.13,\ 0.33,\ 0.935]^T$$

$$\mathbf{c}^0 = [0.1,\ -0.42,\ 0.902]^T$$

$$\mathbf{d}^0 = [-0.68,\ -0.12,\ 0.7233]^T$$

To ensure the procedure to converge to a solution, a three-step continuation scheme is used, under which the procedure converges satisfactorily with an average of about 30 iterations per continuation step. Shown in Figs (6.2c) and (6.2d) are the solutions at the end of each continuation step The linkage error at the final solution was found to be the order of $10^{-3}$. The position vectors of the four joints of the optimum linkage are (Fig. 6 2e):

$$\mathbf{a} = [-0.751365,\ 0.027818,\ 0.659298]^T$$

$$\mathbf{b} = [0.135741,\ 0.332738,\ 0.933199]^T$$

$$\mathbf{c} = [0.095161,\ -0.408915,\ 0.907597]^T$$

$$\mathbf{d} = [-0.685186,\ -0.072465,\ 0.724754]^T$$

and the $\alpha$ angles are hence given as follows:

$$\alpha_1 = 8^\circ, \quad \alpha_2 = 58.5^\circ, \quad \alpha_3 = 43.6^\circ, \quad \alpha_4 = 51.5^\circ$$

$$\alpha_5 = 23.4^\circ, \quad \alpha_6 = 38.3^\circ, \quad \alpha_7 = 28.4^\circ$$

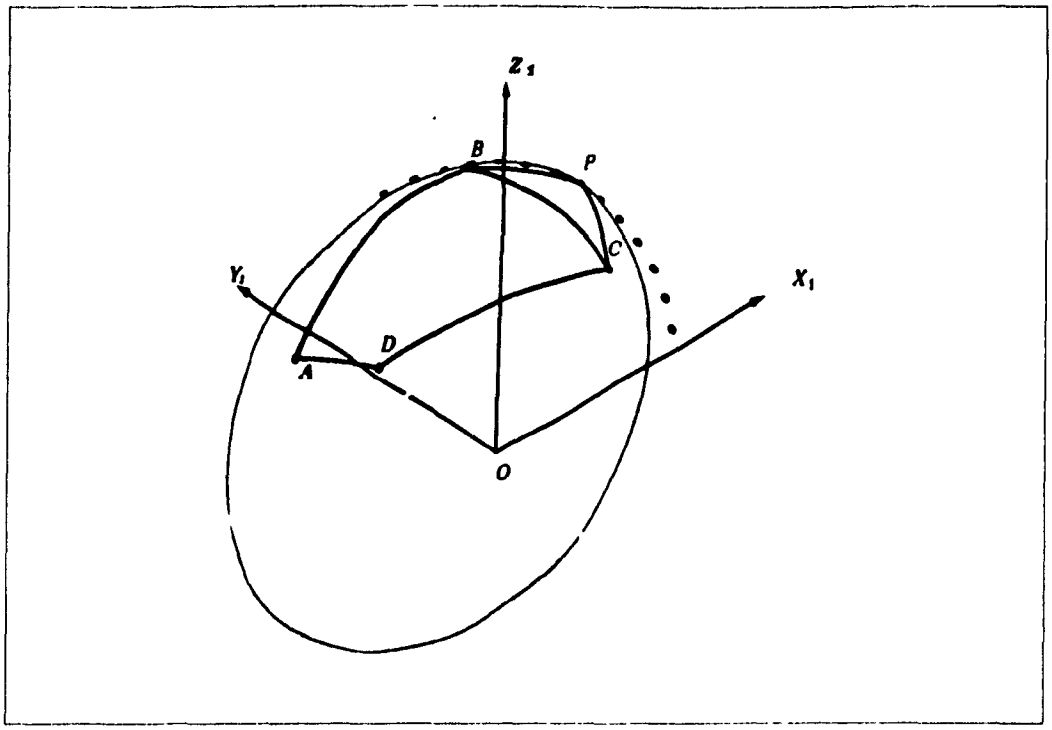


**Figure 6.2b** Initial Guess

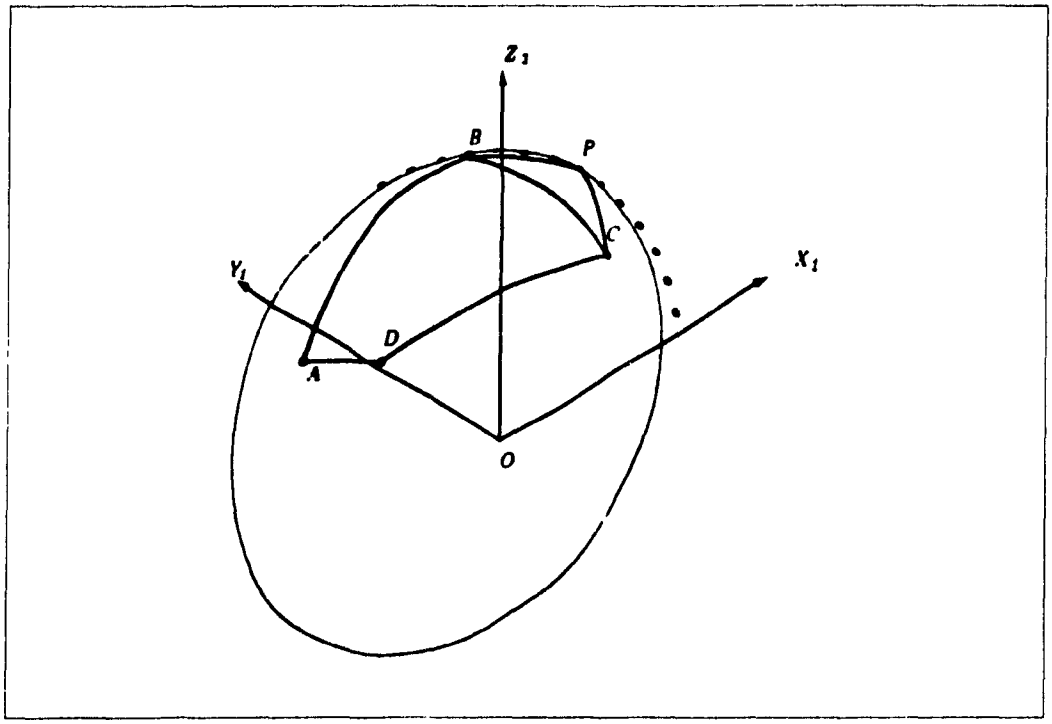


**Figure 6.2c** Solution at the End of the First Continuation Step

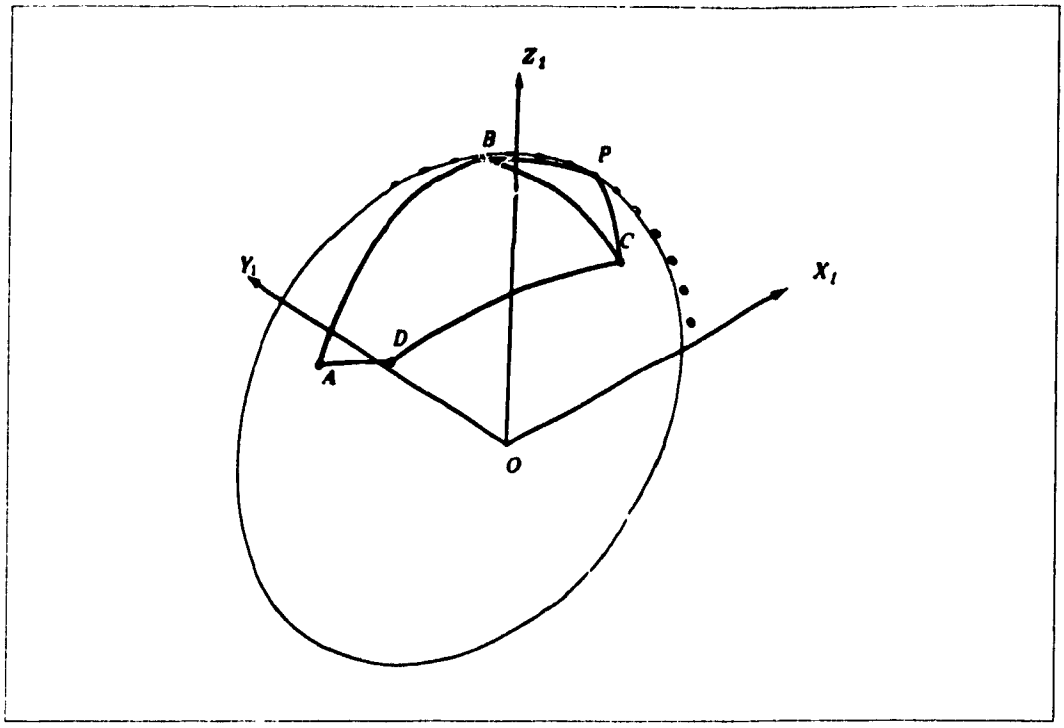


**Figure 6.2d** Solution at the End of the Second Continuation Step

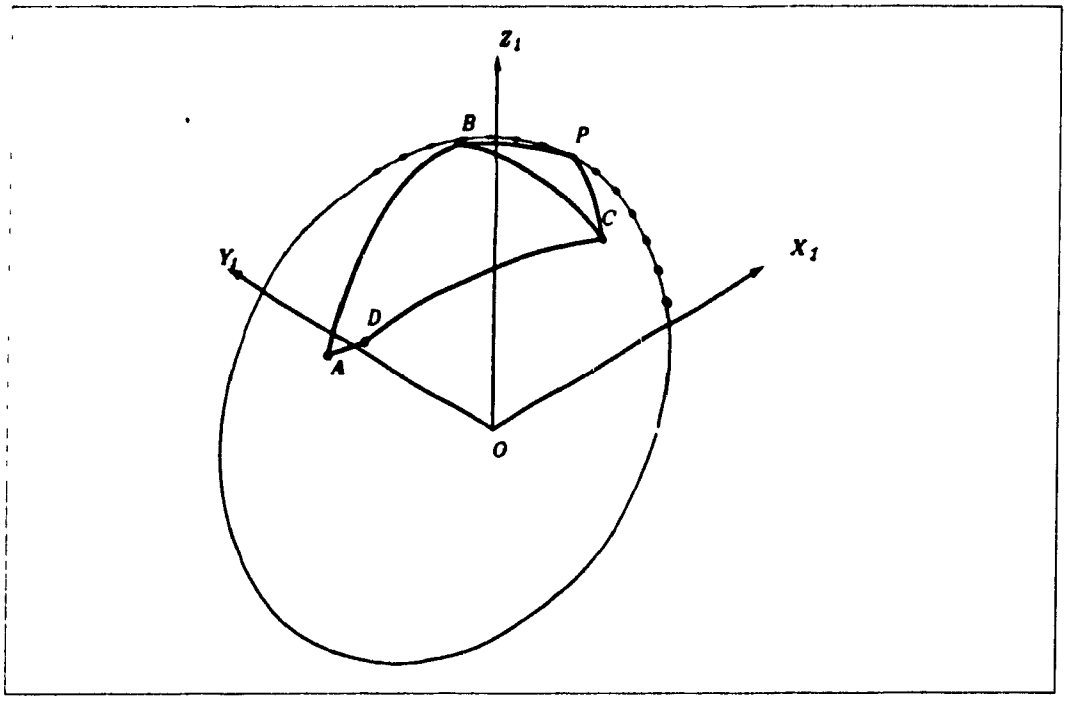


**Figure 6.2e** Final Solution

### 6.2.2 A Spherical Mechanism Tracing a Solar Path on the Winter Solstice

Again, we use eqs.(6 1)-(6.3) to compute the solar path in Montreal with $\delta = -23.5^{\circ}$, for the winter solstice The data used in this example are selected from the solar path, also starting from 8.30 a m and ending at 3·00 p.m for every half an hour, the coordinates of the set $\{Q_k\}_0^{13}$ being as follows (Fig 6 3a)

$$
\begin{aligned}
Q_0 &\quad (0.930418,\ 0,\ 0.366501),\\
Q_1 &\quad (0.676715,\ 0.727553,\ 0.112799),\\
Q_2 &\quad (0.740488,\ 0.648459,\ 0.176518),\\
Q_3 &\quad (0.796416,\ 0.558271,\ 0.232499),\\
Q_4 &\quad (0.843541,\ 0.458530,\ 0.279624),\\
Q_5 &\quad (0.881057,\ 0.350944,\ 0.317140),\\
Q_6 &\quad (0.908322,\ 0.237353,\ 0.344406),\\
Q_7 &\quad (0.924870,\ 0.119701,\ 0.360954),\\
Q_8 &\quad (0.924870,\ -0.119701,\ 0.360954),\\
Q_9 &\quad (0.908322,\ 0.237353,\ 0.344406),\\
Q_{10} &\quad (0.881057,\ -0.350944,\ 0.317140),\\
Q_{11} &\quad (0.843541,\ -0\ 458530,\ 0.279624),\\
Q_{12} &\quad (0.796416,\ 0.558271,\ 0.232499),\\
Q_{13} &\quad (0.740488,\ -0.648459,\ 0.176518)
\end{aligned}
$$

where $P_0 \equiv Q_0$, the solar position at noon, is the coupler point in its initial configuration. A suitable initial guess was obtained and is given as follows (Fig. 6.3b).

$$
\begin{aligned}
\mathbf{a}^0 &= [0.62,\ 0.23,\ -0.75]^T\\
\mathbf{b}^0 &= [0.935,\ 0.33,\ 0.13]^T\\
\mathbf{c}^0 &= [0.902,\ -0.42,\ 0.1]^T\\
\mathbf{d}^0 &= [0.7233,\ 0.12,\ -0.68]^T
\end{aligned}
$$

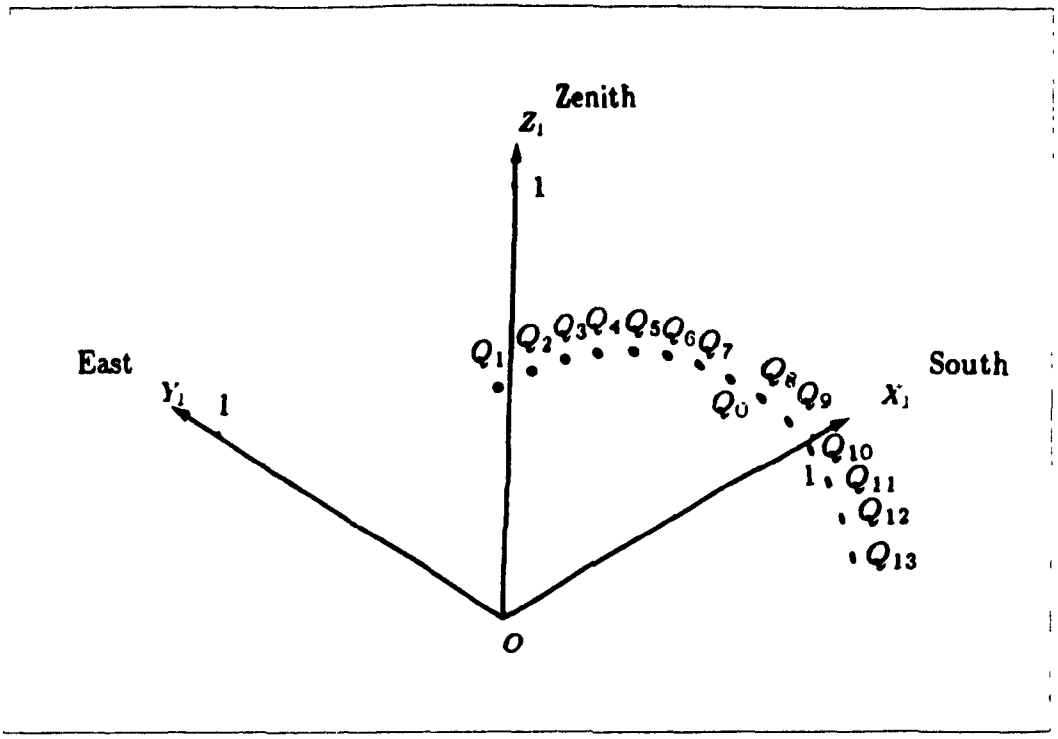


**Figure 6.3a** Points Selected from the Solar Path of Winter Solstice

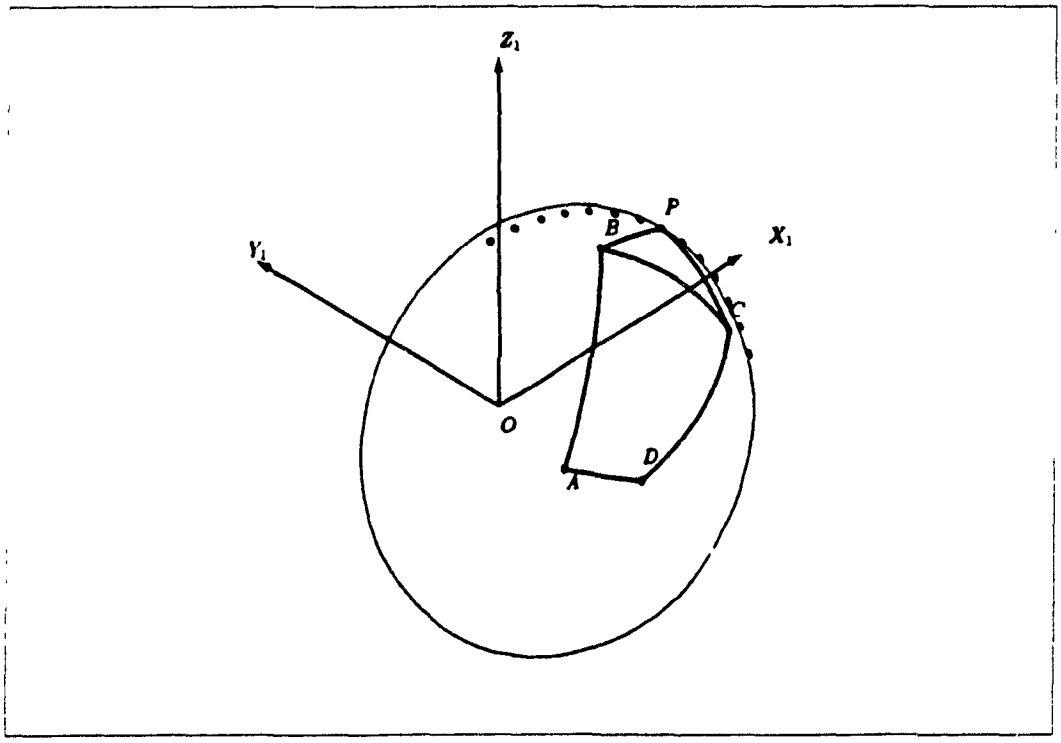


**Figure 6.3b** Initial Guess

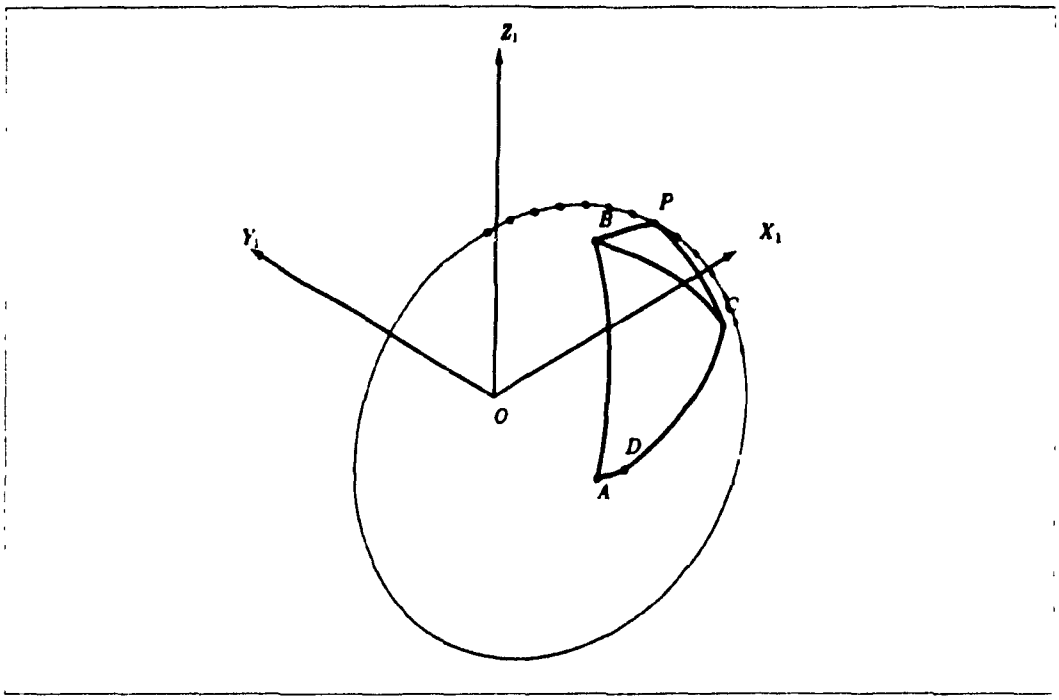


**Figure 6.3c** Final Solution

With a three-step continuation similar to the one in the last example, the solution to the problem is readily obtained as follows (Fig. 6.3c).

$$\mathbf{a} = [0.659294,\ 0.027820,\ -0.751364]^T$$

$$\mathbf{b} = [0.933195,\ 0.332741,\ 0.135757]^T$$

$$\mathbf{c} = [0.907591,\ -0.408922,\ 0.095191]^T$$

$$\mathbf{d} = [0.724764,\ -0.072462,\ -0.685176]^T$$

at which the linkage error was found to be the order of $10^{-3}$. Then, we have the dimensions of the optimal linkage as follows

$$\alpha_1 = 8^\circ, \quad \alpha_2 = 58.6^\circ, \quad \alpha_3 = 43.6^\circ, \quad \alpha_4 = 51.5^\circ$$

$$\alpha_5 = 23.4^\circ, \quad \alpha_6 = 38.3^\circ, \quad \alpha_7 = 28.4^\circ$$

Obviously, the dimensions—the $\alpha$ angles—of this linkage are identical to those of the optimum linkage obtained in last example. The reason for this is clear if we notice the fact that every $Q_k$ point of the solar path in the summer solstice and its counterpart in the winter solstice are related in such a way that their $x$ and $z$ coordinates are exchanged.

while their $y$ coordinate remains the same. In other words, one trajectory can be obtained from other by a simple linear transformation The matrix representing the transformation is, in fact, the following:

$$\mathbf{Q}_{su'} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

which is clearly a reflection about the plane $x + z = 0$. It is easy to see that the solutions of **a**, **b**, **c** and **d** in these two examples also bear this relationship. Therefore, the angles between each two joint vectors are preserved, which is just a consequence of the fact that one set is a reflection of the other This can be a useful conclusion in the design of solar-tracking mechanisms

## 6.3 A Four-Bar Geneva-Wheel Driving Mechanism

A Geneva mechanism is a very common means of producing intermittent motion. Sometimes it is driven by a four-bar linkage instead of a crank for reducing the inertia force and the consequent wear on the sides of the slot, since a jump in acceleration occurs upon engagement and disengagement (Shigley, 1959, Pazouki and Rees Jones, 1982) In using a four-bar linkage as the driver, the trajectory traced by the pin which is placed at the coupler point of the linkage can be carefully planned to fit the smoothness requirements in the wheel motion, thereby eliminating the problem of jerk discontinuities (Dijksman, 1966) The linkage is then designed as a spherical four-bar path generator, based on the required coupler curve In this example, a spherical four-bar linkage to drive a four-sloted spherical Geneva mechanism (Fig 6 4) is to be designed The acceleration of the Geneva wheel is required to follow a sine function, the engagement and disengagement points of the pin located at 0 and $2\pi$, respectively, so that the jumps in acceleration at these two points are eliminated

First of all, the trajectory to be followed by the pin under the above requirements is determined We set up a spherical coordinate system and assume that the pin moves on

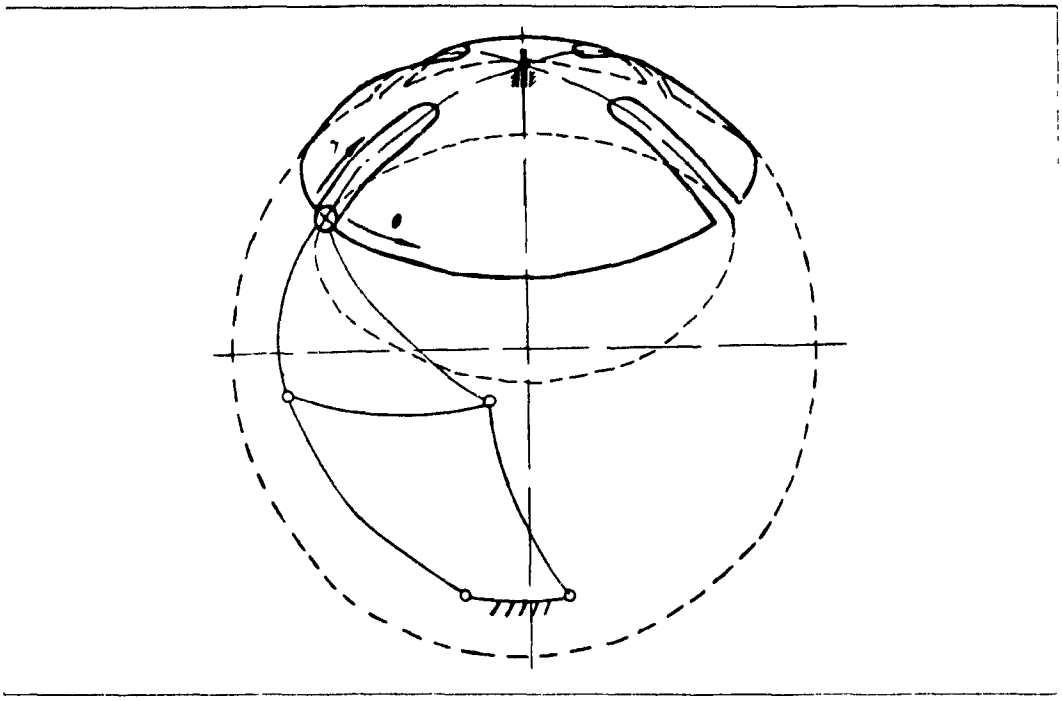

**Figure 6.4** Spherical Geneva Wheel Driven by A Four-Bar Linkage

a unit sphere The motion of the pin can then be defined by the following two coordinates the angular displacement of the Genava wheel, $\theta$, and the relative displacement of the pin with respect to the wheel, $\gamma$, as shown in Fig 6 4 The equations of motion $\theta = \theta(t)$ and $\gamma = \gamma(t)$ are next determined As required, we write $\dot{\theta}$ as

$$\theta = C_1 \sin(2\pi \frac{t}{t_0}) \tag{6.4}$$

where $C_1$ is a constant to be determined from the problem's conditions, $t_0$ is the time spanned from engagement until disengagement The angular velocity and displacement of the wheel are thus derived as

$$\theta = \int \ddot{\theta} dt = -C_1 \frac{t_0}{2\pi} \cos(2\pi \frac{t}{t_0}) + C_2 \tag{6.5}$$

$$\theta = \int \dot{\theta} dt = -C_1 \frac{t_0^2}{4\pi^2} \sin(2\pi \frac{t}{t_0}) + C_2 t + C_3 \tag{6.6}$$

with the following boundary conditions:

$$\text{for } t = 0, \quad \theta = 0, \quad \dot{\theta} = 0;$$

$$\text{for } t = t_0, \quad \theta = \frac{\pi}{2}, \quad \dot{\theta} = 0.$$

Hence we obtain

$$C_1 = \frac{\pi^2}{t_0^2}, \quad C_2 = \frac{\pi}{2t_0}, \quad C_3 = 0 \tag{6.7}$$

Upon substitution of eq (6 7) into eq (6 6), we have

$$\theta = -\frac{1}{4}\sin(2\pi\tau) + \frac{\pi}{2}\tau \tag{6.8}$$

where $\tau \equiv t/t_0$, represents the normalized time We have then $0 \leq \tau \leq 1$. Equation(6.8) thus defines the motion of the wheel Given in (Fig 6.5a) is the plot of $\theta$ vs $\tau$.

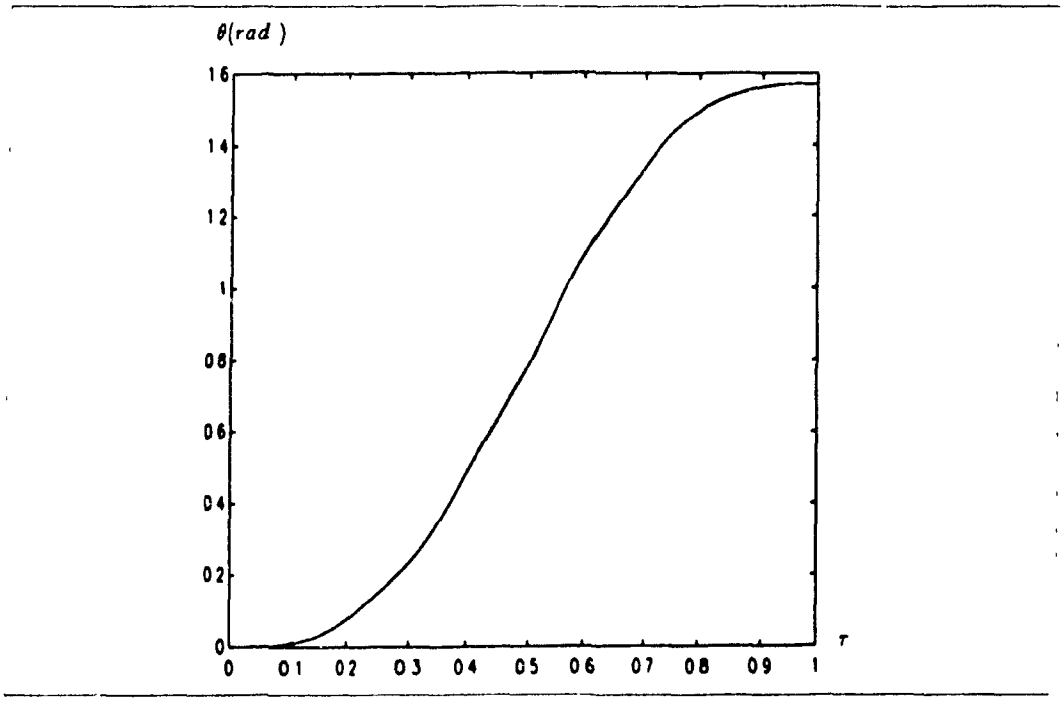


**Figure 6.5a** Plots of $\theta$ vs normalized time

Since no particular requirements are imposed on the relative motion of the pin with respect to the wheel, $\gamma$, it may be arbitrary as long as it does not exceed the length of each slot and attains the same values upon both engagement and disengagement In our case, $\gamma$ is chosen as

$$\gamma = -1.5\tau^2 + 1.5\tau + C \tag{6.9}$$

where constant $C$ depends on the length of the slots. For this example, $C = 53.13^\circ = 0.9273$ rad The $\gamma$ vs $\tau$ plot is given in Fig 6 5b.

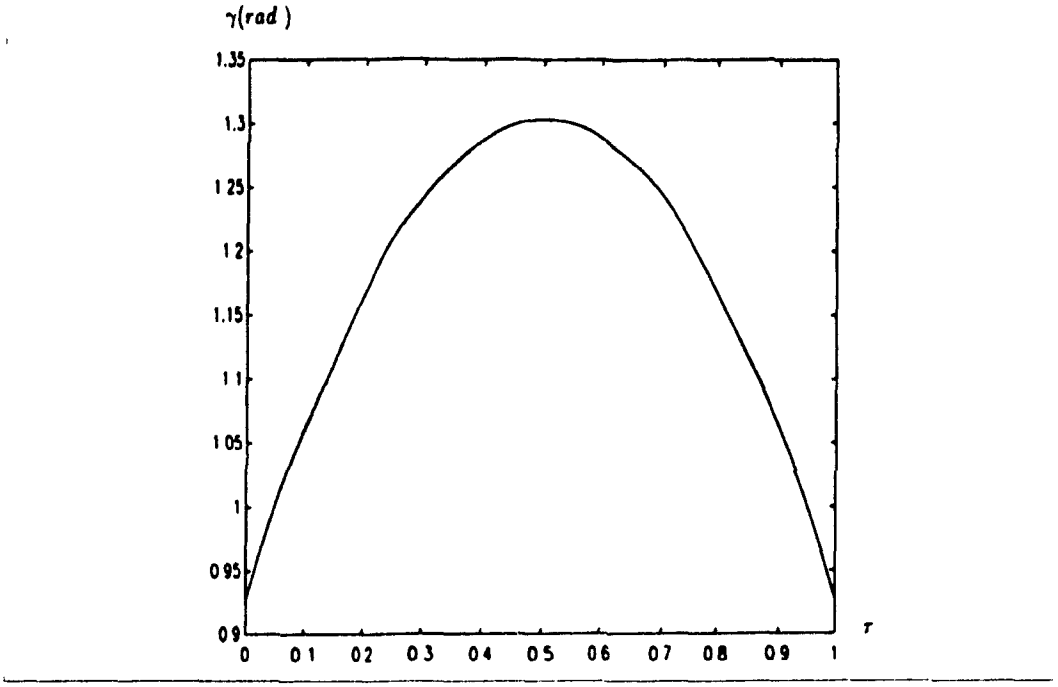


**Figure 6.5b** Plot of $\gamma$ vs normalized time

From eqs (6.8) and (6.9), the coupler trajectory is computed. After transforming it into Cartesian coordinates, the following are selected as the given trajectory points (Fig. 6.6a)

$Q_0$ : (0.1875895, 0.1875895, 0.9641682),

$Q_1$ (0.6, 0, 0.8).

$Q_2$ : (0.5082041, 0.0026555, 0.8612326),

$Q_3$ (0.4090282, 0.0231497, 0.9122280),

$Q_4$ . (0.3282382, 0.0645749, 0.9423851),

$Q_5$ (0.2605160, 0.1181456, 0.9582135),

$Q_6$ (0.1181456, 0.2605160, 0.9582135),

$Q_7$ : (0.0645749, 0.3282382, 0.9423851),

$Q_8$ : (0.0231497, 0.4090282, 0.9122280),

$Q_9$ : (0.0026555, 0.5082041, 0.8612326),

$Q_{10}$ : (0, 0.6, 0.8)

of which $P_0 \equiv Q_0$ is the coupler point in the initial configuration. The optimization starts

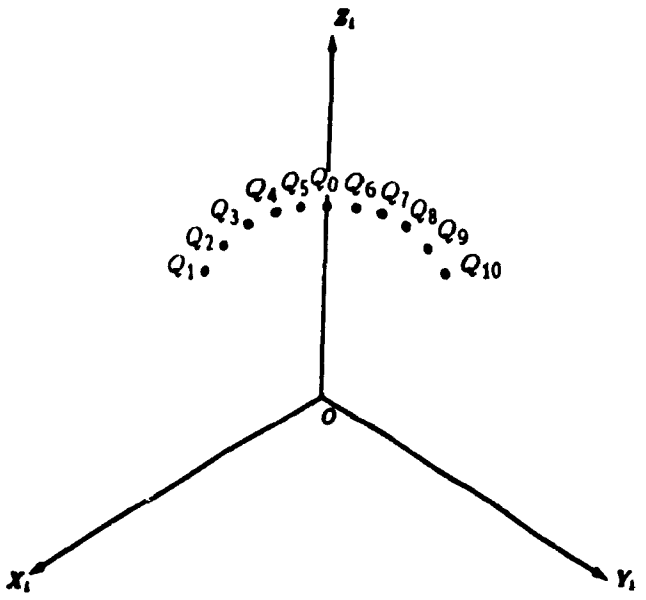


**Figure 6.6a** Points Selected from the Trajectory followed by the Pin

from the following initial guess (Fig 6.6b):

$$\mathbf{a}^0 = [0.69,\ 0.59,\ 0.4192]^T,$$

$$\mathbf{b}^0 = [0.52,\ 0.2,\ 0.83042]^T,$$

$$\mathbf{c}^0 = [0.2,\ 0.52,\ 0.83042]^T,$$

$$\mathbf{d}^0 = [0.59,\ 0.69,\ 0\,4192]^T$$

Again, a three-step continuation is applied to the procedure, which yields the optimum linkage (Fig 6.6c). The position vectors of the four joint-centre of the linkage are given as:

$$\mathbf{a} = [0.63081,\ 0.47113,\ 0.61653]^T,$$

$$\mathbf{b} = [0.46102,\ 0.18003,\ 0.86894]^T,$$

$$\mathbf{c} = [0.18101,\ 0.45984,\ 0.86936]^T,$$

$$\mathbf{d} = [0\,47052,\ 0.63101,\ 0.61680]^T$$

the linkage error being of the order of $10^{-3}$. Here, the average number of iterations within each continuation step is around 35. For the solution found, the dimensions of the linkage

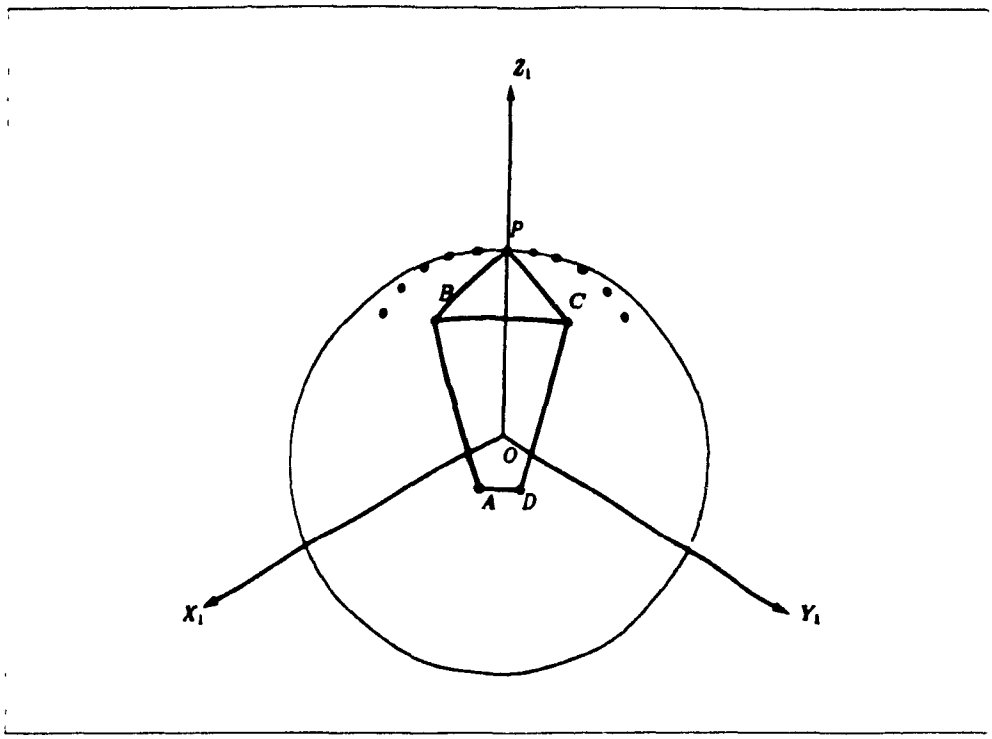


**Figure 6.6b** Initial Guess

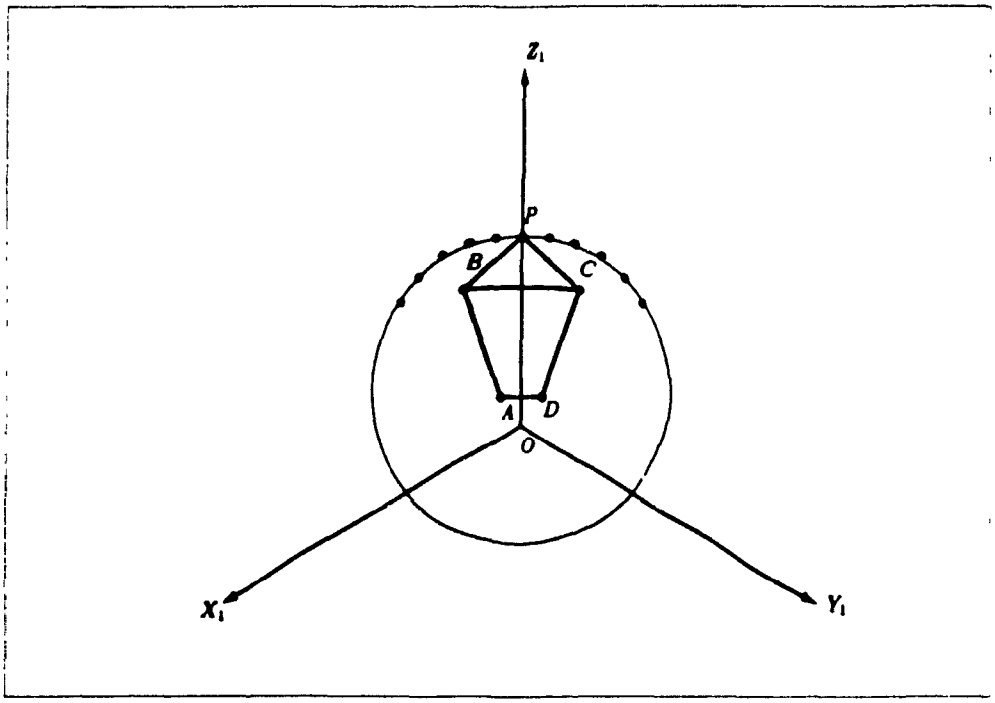


**Figure 6.6c** Final Solution

are computed as

$$\alpha_1 = 13.12^\circ, \quad \alpha_2 = 24.30^\circ, \quad \alpha_3 = 22.83^\circ, \quad \alpha_4 = 24.27^\circ,$$
$$\alpha_5 = 36.76^\circ, \quad \alpha_6 = 47.29^\circ, \quad \alpha_7 = 36.74^\circ$$

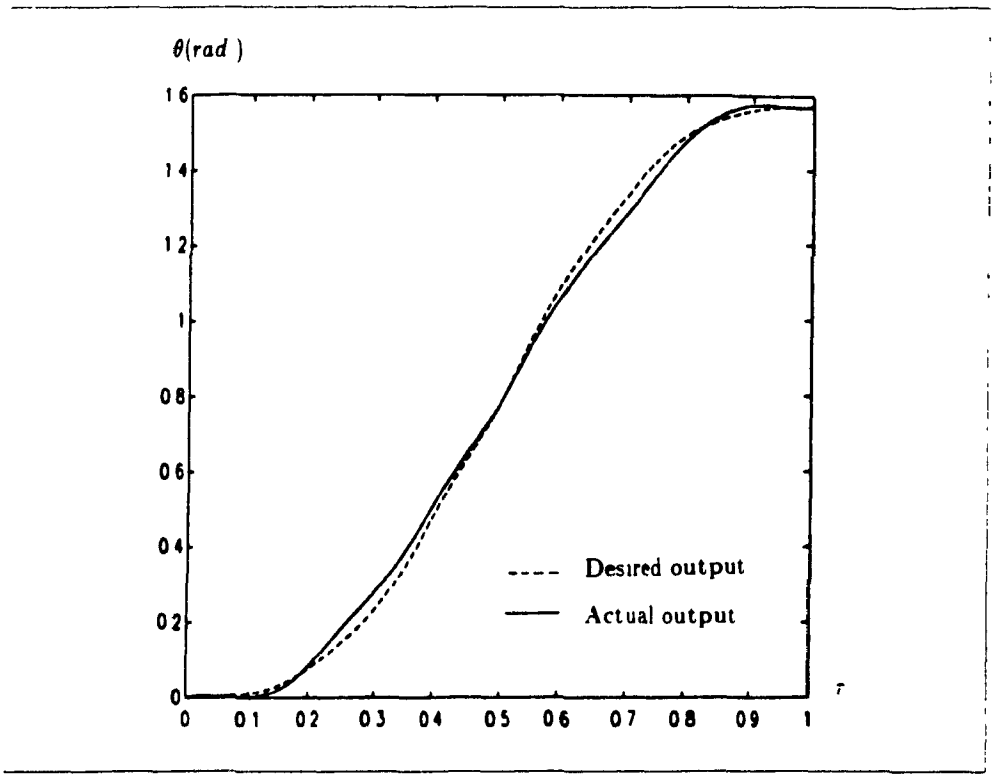


**Figure 6.7** The Actual Output in Wheel Motion Compared with the Desired Output

It is apparent that the optimum linkage is symmetric to the plane $x + y = 0$, which is not surprising because the given points, as well as the initial linkage, are both symmetric with respect to the same plane

Now, from Fig. 6.7, we can look at the error in the motion of the wheel when it is driven by the four-bar linkage obtained above. We can see that the two curves meet upon engagement and disengagement, i.e., at $\tau = 0$ and $\tau = 1$, which means that zero accelerations are achieved at these two points. Also to be noticed is the fact that the two curves intersect at $\tau = 0.5$, which actually corresponds to the case that the coupler point passes through $Q_0$, the point being met exactly

# Chapter 7 Concluding Remarks

In summary, a constrained least-square optimization scheme was developed to deal with the problems of synthesis of spherical four-bar path generators. The problem is formulated as two-layer optimization procedures, in which the linkage parameters and the configuration variables, namely, a certain set of input angles, are evaluated separately in each layer Since no upper bound of given points is imposed on the formulation, the method can handle the synthesis problem subject to any number of given points on the surface of the sphere on which the four-bar linkage is defined, as long as this number is greater than nine. Thus, the method aims at solving the approximate synthesis problem of path generation. The orthogonal-decomposition algorithm was employed here, which allowed us to obtain the solution iteratively from an initial guess In order to use the said numerical scheme, the computation of the Jacobian matrix associated with the objective function becomes a major task and has been discussed fully in Chapter 4. The behaviour of the Jacobian matrix, which affects the convergence of the procedure, is assessed by resorting to the concept of condition number. Damping techniques are used to stabilize the iterative procedure and speed up its convergence rate. Moreover, as an important step to ensure the procedure to converge to a solution, continuation is applied. A general computer package, SPHER1, was developed for the whole optimization scheme. Numerical examples concerning the design of spherical mechanisms were introduced and solved in Chapter 6. The results show that the method is efficient for the optimization of spherical four-bar path generators.

Here, a few remarks are added in concluding the thesis:

1. The optimization scheme presented here deals with problems in which only the given trajectory points are specified For solving problems which have special requirements, the method has to be modified correspondingly in the formulation For example, in some practical problems, certain link lengths or positions of certain joints need to be specified In this case, we can simply include them in the constraint function $\mathbf{g}(\mathbf{x})$ and compute its Jacobian $\mathbf{G}(\mathbf{x})$ accordingly By the same token, the coordination between the given points and the input angles can be incorporated, although this case is still under development

2 Obviously, the solution obtained by using the proposed scheme produces a local minimum The solution depends on the nature of the problem and the initial guess chosen. It is possible that, for the same problem, different initial guesses lead to different solutions, which correspond to different linkage errors This may be the base for finding the global minima, whereas how to obtain the global minima is still a topic of further research So far, no research has been reported in the literature on path-generating linkage dealing with the computation of the global minimum.

3 From the examples in Chapter 6, it is clear that the optimum linkage reflects some characteristics of the set of given points. In section 6.3, for example, the property of symmetry in the given points leads to the same property in the optimal linkage obtained, the transformation relationship in the given sets of the two examples in section 6 2 also appears in the final solution. All these facts suggest the existence of *similarity* in the synthesis procedure, which makes sense both geometrically and logically Actually, this is a fact we can exploit in chosing the initial guess, judging the correctness of the solutions, or extending results from a known solution

4. When $m$ is given exactly as nine, which is the case allowing for *exact synthe-*

*sis*, the procedure, which is heavily dependent on the orthogonal-decomposition algorithm, can still be used. However, in this case, the system of nonlinear equations arising from the zeroing of the error become *stiff* and convergence is very difficult to achieve.

On the whole, the optimization scheme solved the synthesis problems of spherical four-bar path generators efficiently, as shown in the thesis The formulation has no strict limitation on the type of linkages or the nature of the synthesis problems to be solved For the other synthesis problems in the realm of spherical linkages, such as rigid-body guidance, we can define a linkage error for the specific case and find a corresponding normality condition as in eq (4.8) and then apply the optimization scheme Even for other types of linkages, the above idea can be still applied, whereas the computational procedure of Jacobian matrices has to be reorganized to meet the special case, which might be a major task The author found that the computation of the Jacobian matrix of the objective function is a painful process, while it is essential for the success of the whole optimization scheme It is desired that the said computation be simplified or replaced with an easier equivalence However, how to achieve this remains unsolved and is left for future research

Finally, a few words on the new criterion of mobility analysis for spherical four-bar linkages, as given in Chapter 2 Although only the spherical case is studied here, finding its equivalence in the planar case is straightforward, since we have a similar form for the input-output function for planar linkages. Actually, this has been studied and will be included in a forthcoming paper

# References

Akhras, R and Angeles, J., 1987, "Unconstrained Nonlinear Least-square Optimization of Planar Linkages for Rigid-body Guidance", *Technical Report, McGill Research Center for Intelligent Machines(McRCIM)*

Alizade, R I., Novruzbekov, G I and Sandor, G N., 1975, "Optimization of Four-Bar Function Generating Mechanisms Using Penalty Functions with Inequality and Equality Constraints", *Mechanism and Machine Theory*, Vol 10, pp 327-336

Anderson, K., 1987, "Inverse Kinematics of Robotic Manipulators in the Presence of Singularities and Redundancies", *Master Thesis*, Department of Mechanical Engineering, McGill University

Angeles, J., 1982, *Spatial Kinematic Chains, Analysis, Synthesis and Optimizations*, Spriger-Verlag, Berlin

Angeles, J., 1988, "Mechanics of Robotic Manipulators I", *Course Notes*

Angeles, J., Alivizatos, A and Akhras, A., 1987, "A Constrained Nonlinear Least-Square Method of Optimization of *RRRR* Planar Path Generators", *To appear in Mechanism and Machine Theory*

Angeles J., Anderson, K and Gosselin, C., 1987, "An Orthogonal-Decomposition Algorithm for Constrained Least-Square Optimization", *Proceedings of the 13th ASME Design and Automation Conference*, Boston, Ma Sept 1987, pp 215-220

Angeles, J and Bernier, A., 1987, "A General Method of Four-Bar Linkage Mobility Analysis", *ASME Journal of Mechanisms, Transmissions and Automation in Design*, Vol. 109, pp 197-203

Angeles, J and Callejas, M., 1984, "An Algebraic Formulation of Grashof's Mobility Criteria with Application to Linkage Optimization Using Gradient-Dependent Methods", *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, Vol 106, No. 3, pp. 327-332

Angeles, J and Rojas, A., 1987, "Manipulator Inverse Kinematics via Condition-Number Minimization and Continuation", *International Journal of Robotics and Automation* Vol. 2, No. 2, pp 61-69

Avila, J H, Jr, 1974, "The Feasibility of Continuation Methods for Nonlinear Equations", *SIAM Journal of Numerical Analysis*, Vol.11, No 1, pp 102-122

Betts, J T, 1980, "A Compact Algorithm for Computing the Stationary Point of a Quadratic Function Subject to Linear Constraints", *ACM Transactions on Mathematical Software*, Vol 6, No 3, pp 391-397

Brent, R P, 1973, *Algorithm for Minimization Without Derivatives*, Prentice-Hall, Englewood Cliffs, New Jersey

Chi-Yeh, H, 1966, "A General Method For the Optimum Design of Mechanisms", *Journal of Mechanisms*, Vol 1, pp 301-313

Chiang, C H 1984, "On the Classification of Spherical Four-bar Linkages", *Mechanism and Machine Theory*, Vol 19, No 3, pp 283-287

Chiang, C.H 1986, "Synthesis of Spherical Four-bar Path Generators", *Mechanism and Machine Theory*, Vol 21, No 2, pp 135-143

Dahlquist, G and Bjorck, Å., 1974, *Numerical Methods*, Prentice-Hall, Inc, Engelwood Cliffs, NJ, pp 442-444

Duditiza, F and Dittrich, G, 1969, "Die Bedingungen fur die Umlauffahigkeit spharischer viergliedriger kurbelgetriebe", *Industrie-Anzeiger*, Vol 91, pp 1687-1690

Dijksman, A E, 1966, "Jerk-Free Geneva Wheel Driving", *Journal of Mechanisms*, Vol 1, pp 235-283

Fiacco, A V and McCormick, G P., 1964, "Computational Algorithm for the Sequential Unconstrained Minimization Technique for Nonlinear Programming", *Management Science*, Vol. 10, No. 4, pp 601-617

Forsythe, G.E., Malcolm, M.A and Moler, C.B., 1977, *Computer Methods for Mathematical Computations*, Prentice-Hall Inc., Englewood Cliffs, New Jersey

Fox, R.L., 1971, *Optimization Methods for Engineering Design*, Addison Wesley

Golinski, J., 1970, "Optimal Synthesis Problems Solved by Means of Nonlinear Programming and Random Methods", *Journal of Mechanisms*, Vol 5, pp 287-309

Golub, G H and Van Loan, C F, 1983, *Matrix Computations*, The John Hopkins University Press, Baltimore

Gosselin, C and Angeles, J, 1988, "Mobility Analysis of Planar and Spherical Linkages At A Glance", *Journal of Computers in Mechanical Engineering*, July/August 1988, pp 56-60

Grashof, F, 1883, *Theorethische Maschinenlehre*, Berlin

Gupta, K C, 1986, "Rotatability Considerations for Spherical Four-Bar Linkage with Applications to Robot Wrist Design", *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, Vol 108, No 3, pp 387-391

Hartenberg, R S and Denavit, J, 1964, *Kinematic Synthesis of Linkages*, McGraw-Hill, New York

Iqbal, M, 1983, *An Introduction to Solar Radiation*, Academic Press, New York, 1983

Kraus, R, 1952, "Wertigkeitsbilanz und ihre Anwendung auf eine Geradfuhrung fur Meßgeräte", *Feinwerktechnik*, Vol 56 No 3 pp 57-63

Lawson, C L and Hanson, R J, 1974, *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, New Jersey

Lewis, D W and Falkenhagen, G L, 1968, "Displacement and Velocity Kinematic Synthesis", *ASME Journal of Engineering for Industry*, Series B, Vol 90, No 3, pp.527-530, Aug, 1968

Ma, O. and Angeles, J 1988, "Performance Evaluation of Path-Generating Planar, Spherical and Spatial Four-bar Linkages", *Mechanisms and Machine Theory*, Vol.23, No 4, pp 257-268

Pazouki, M E and Rees Jones, J, 1982, "The Kinematic Synthesis of A Linkage Driven Mechanism", *Mechanism and Machine Theory*, Vol 17, pp. 221-228

Richter, S.L. and DeCarlo, R. A. 1983. "Continuation Methods· Theory and Applications", *IEEE Transaction of Automatic Control*, Vol AC-28, No 6, 1983, pp. 660-665

Rao, A C., 1979. "Synthesis of 4-Bar Function-Generators Using Geometric Programming", *Mechanism and Machine Theory*, Vol 14, pp 141-149

Rosen, J B , 1960. "The Gradient Projection Method for the Nonlinear Programming—Part I", *Journal of Soc Indus Appl Math* Vol 8, pp 181-217

Rosen, J B , 1961. "The Gradient Projection Method for the Nonlinear Programming—Part II", *Journal of Soc Indus Appl Math* Vol 8, pp 414-432

Savage, M and Hall, A S , 1970. "Unique Descriptions of All Spherical Four-Bar Linkage", *ASME Journal of Engineering for Industry*, Vol 92, pp 559-563

Schittkowski, K , 1985. "An Unified Outline of Nonlinear Programming Algorithm", *ASME Journal of Mechanism, Transmission and Automation in Design*, Vol 107, pp 449-453

Shigley, E.J , 1959, *Kinematic Analysis of Mechanisms*, McGraw-Hill, New York

Soni, A h and Harrisberger, L , 1967, "The design of the Spherical Drag Link Mechanism", *ASME Journal of Engineering for Industry*, Vol 89, pp 177-181

Suh, C H and Radcliffe, C W 1978, *Kinematics and Mechanisms Design*, John Wiley and Sons Inc

Sutherland, G H and Karwa, N R , 1978, "Ten-Design-Parameter 4-Bar Synthesis With Tolerance Consideration", *Mechanisms and Machine Theory*, Vol 13, pp 311-327

Tomáš, J , 1968. "The Synthesis of Mechanisms as a Nonlinear Programming Problem", *Journal of Mechanism*, Vol 3, pp 119-130

Wasserstrom, E 1973, "Numerical Solutions By the Continuation Method", *SIAM Review* Vol 15, No 1, pp 89-119

Williams, R.L II and Reinholtz, C F , 1986, "Proof of Grashof's Law Using Polynomial Discriminants", *ASME Journal of Mechanisms, Transmission and Automation in Design*, Vol 108, No 4, pp. 562-564

Williams, R.L II and Reinholtz, C.F., 1987, "Mechanism Link Rotatability and Limit Position Analysis Using Polynomial Discriminants", *ASME Journal of Mechanisms, Transmission and Automation in Design* Vol 109, No 2, pp 178-182

Zangwill, W I , 1967, "Nonlinear Programming via Penalty Functions", *Management Science* , Vol 13, No 5, pp 344-358

Zhou, J and Mayne, R W , 1985, "Monotonicity Analysis and Recursive Quadratic Programming in Constrained Optimization", *Trans. ASME Journal of Mechanism, Transmission and Automation in Design*, Vol, 107, pp 495-499

Zoutendijk, G , 1960, *Methods of Feasible Directions*, Elsevier, Amsterdam

# Appendix A. Mobility Analysis with Linkage Parameters

Same as denoted in Section 2.1.2, we assume $s = (k_1 + k_2 + k_3 + k_4)/2$ in the following procedure The inequalities representing the mobility region for the input link are first handled

For eqs (2.12$a$) and (2.12$b$), we have the following:

($i$) If $k_3 - k_4 \geq 0$, then

$$k_1 + k_2 \quad k_3 - k_4$$

$$k_2 + k_1 \leq k_4 - k_3$$

from which we obtain

$$k_3 \geq s \leq k_4$$

($ii$) If $k_3 - k_4 \leq 0$, then

$$k_2 + k_1 \geq k_4 - k_3$$

$$k_2 + k_1 \leq k_3 - k_4$$

which yield

$$k_4 \leq s \leq k_3$$

Both ($i$) and ($ii$) hold if

$$\min(k_3,\ k_4) \leq s \leq \min(k_3,\ k_4)$$

From eqs.(2.12$c$ & $b$), we have:

($i$) If $k_3 + k_4 \geq 0$, then

$$k_2 - k_1 \leq k_3 + k_4$$

$$k_2 - k_1 \geq -k_3 - k_4$$

which leads to

$$s \geq k_2, \quad s \geq k_1$$

or simply

$$s \geq \max(k_1, k_2)$$

(*ii*) If $k_3 + k_4 \leq 0$, then

$$k_2 - k_1 \leq -k_3 - k_4$$

$$k_2 - k_1 \geq k_3 + k_4$$

from which we obtain

$$s \leq k_1, \; s \leq k_2$$

or

$$s \leq \min(k_1, k_2)$$

Now, the inequalities representing the mobility range of the output link are analyzed

From eqs (2.16*a* & *b*), we have

(*i*) If $k_2 + k_3 \geq 0$, then

$$k_1 - k_4 \leq k_2 + k_3$$

$$k_1 - k_4 \geq -k_2 - k_3$$

which yield

$$s \geq k_1, \quad s \geq k_4$$

or

$$s \geq \max(k_1, k_4)$$

(*ii*) If $k_2 + k_3 \leq 0$, then

$$k_1 - k_4 \leq -k_2 - k_3$$

$$k_1 - k_4 \geq k_2 + k_3$$

from which we obtain

$$s \quad k_1, \quad s \quad k_4$$

or

$$s \leq \min(k_1, k_4)$$

From eqs (2.16*a* & *b*), we have

(*i*) If $k_2 - k_3 \geq 0$, then

$$k_1 + k_4 \leq k_2 - k_3$$

$$k_1 + k_4 \geq -k_3 + k_2$$

which yield

$$k_3 \leq s \leq k_2$$

.

(*ii*) If $k_2 - k_3 \leq 0$, then

$$k_1 + k_4 \leq k_3 - k_2$$

$$k_1 + k_4 \geq k_2 - k_3$$

which yields

$$k_3 \leq s \leq k_2$$

If we combine the results of (*i*) and (*ii*), the following is obtained:

$$\min(k_2, k_3) \leq s \leq \max(k_2, k_3)$$

# Appendix B. Computation of the Rotation Matrices $\mathbf{Q}_1$ and $\mathbf{Q}_2$

Based on eqs. (3.4) and (3.5) and the geometric relationships of Figs. 2.1 and 3.1, we have

$$\mathbf{Q}_1 = [\mathbf{l}, \ \mathbf{m}, \ \mathbf{n}]$$

where

$$\mathbf{l} = \mathbf{a}, \qquad \mathbf{m} = \frac{1}{s_1}(\mathbf{d} - \mathbf{a}c_1), \qquad \mathbf{n} = \mathbf{l} \times \mathbf{m}$$

and

$$\mathbf{Q}_2 = [\mathbf{u}, \ \mathbf{v}, \ \mathbf{w}]$$

with

$$\mathbf{u} = \begin{bmatrix} c_2 \\ s_2 \cos\psi \\ s_2 \sin\psi \end{bmatrix},$$

$$\mathbf{v} = \frac{1}{s_3}\begin{bmatrix} c_1 c_4 - s_1 s_4 \cos\phi - c_2 c_3 \\ s_4 c_1 \cos\phi + c_4 s_1 - s_2 c_3 \cos\psi \\ s_4 \sin\phi - s_2 c_3 \sin\psi \end{bmatrix},$$

$$\mathbf{w} = \mathbf{u} \times \mathbf{v}$$

# Appendix C. Detailed Computation of the Jacobian Matrix $\mathbf{F}(\mathbf{x})$

In the following discussion, the components of vector **a** in the first coordinate frame are denoted $x_A$, $y_A$, $z_A$, with similar definitions for the components of vector **b**, **c** and **d**. Throughout, we will use the notation introduced in Chapter 4.

## C.1 Partial Derivatives Pertaining to $\mathbf{Q}_1$

From the expression of $\mathbf{Q}_1$ in Appendix A, we have

$$\frac{\partial \mathbf{Q}_1}{\partial \mathbf{x}} = \left[\frac{\partial \mathbf{l}}{\partial \mathbf{x}}, \quad \frac{\partial \mathbf{m}}{\partial \mathbf{x}}, \quad \frac{\partial \mathbf{n}}{\partial \mathbf{x}}\right]$$

Hence, we obtain

$$\frac{\partial \mathbf{Q}_1}{\partial x_A} = \frac{1}{s_1}\begin{bmatrix} s_1 & -c_1 & 0 \\ 0 & 0 & -z_D \\ 0 & 0 & y_D \end{bmatrix}, \qquad \frac{\partial \mathbf{Q}_1}{\partial y_A} = \frac{1}{s_1}\begin{bmatrix} 0 & 0 & z_D \\ s_1 & -c_1 & 0 \\ 0 & 0 & -x_D \end{bmatrix}$$

$$\frac{\partial \mathbf{Q}_1}{\partial z_A} = \frac{1}{s_1}\begin{bmatrix} 0 & 0 & -y_D \\ 0 & 0 & x_D \\ s_1 & -c_1 & 0 \end{bmatrix}$$

$$\frac{\partial \mathbf{Q}_1}{\partial x_B} = \frac{\partial \mathbf{Q}_1}{\partial y_B} = \frac{\partial \mathbf{Q}_1}{\partial z_B} = \frac{\partial \mathbf{Q}_1}{\partial x_C} = \frac{\partial \mathbf{Q}_1}{\partial y_C} = \frac{\partial \mathbf{Q}_1}{\partial z_C} = \mathbf{0}_{3\cdot 3}$$

$$\frac{\partial \mathbf{Q}_1}{\partial x_D} = \frac{1}{s_1}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & z_A \\ 0 & 0 & -y_A \end{bmatrix}, \qquad \frac{\partial \mathbf{Q}_1}{\partial y_D} = \frac{1}{s_1}\begin{bmatrix} 0 & 0 & -z_A \\ 0 & 1 & 0 \\ 0 & 0 & x_A \end{bmatrix}$$

$$\frac{\partial \mathbf{Q}_1}{\partial z_D} = \frac{1}{s_1}\begin{bmatrix} 0 & 0 & y_A \\ 0 & 0 & -x_A \\ 0 & 1 & 0 \end{bmatrix}$$

where $\mathbf{0}_{3\times 3}$ is the $3 \cdot 3$ zero matrix.

As far as $s_i$ and $c_i$, for $i = 1, 2, 3, 4$, is concerned, $\mathbf{Q}_1$ depends only on $s_1$ and

$c_1$. Hence, we have

$$\frac{\mathbf{Q}_1}{\partial s_1} = [\frac{\partial \mathbf{l}}{\partial s_1}, \frac{\partial \mathbf{m}}{\partial s_1}, \frac{\partial \mathbf{n}}{\partial s_1}]$$

$$\frac{\partial \mathbf{Q}_1}{\partial c_1} = [\frac{\partial \mathbf{l}}{\partial c_1}, \frac{\partial \mathbf{m}}{\partial c_1}, \frac{\partial \mathbf{n}}{\partial c_1}]$$

where

$$\frac{\partial \mathbf{l}}{\partial s_1} = \mathbf{0}. \qquad \frac{\partial \mathbf{m}}{\partial s_1} = -\frac{c_1}{s_1^2}(\mathbf{d} - \mathbf{a}c_1)$$

$$\frac{\partial \mathbf{n}}{\partial s_1} = \mathbf{l} \times \frac{\partial \mathbf{m}}{\partial s_1}$$

$$\frac{\partial \mathbf{l}}{\partial c_1} = \mathbf{0}, \qquad \frac{\partial \mathbf{m}}{\partial c_1} = -\frac{1}{s_1}\mathbf{a}$$

$$\frac{\partial \mathbf{n}}{\partial c_1} = \mathbf{l} \times \frac{\partial \mathbf{m}}{\partial c_1}$$

and $\mathbf{0}$ is the 3-dimensional zero vector

## C.2 Partial Derivatives Pertaining to Matrix $\mathbf{Q}_2$

### C.2.1 Computation of $\partial \mathbf{Q}_2/\partial s_i$ and $\partial \mathbf{Q}_2/\partial c_i$

From the expression of $\mathbf{Q}_2$ in Appendix B, we have, for $i = 1, 2, 3, 4$.

$$\frac{\partial \mathbf{Q}_2}{\partial s_i} = [\frac{\partial \mathbf{u}}{\partial s_i}, \frac{\partial \mathbf{v}}{\partial s_i}, \frac{\partial \mathbf{w}}{\partial s_i}]$$

$$\frac{\partial \mathbf{Q}_2}{\partial c_i} = [\frac{\partial \mathbf{u}}{\partial c_i}, \frac{\partial \mathbf{v}}{\partial c_i}, \frac{\partial \mathbf{w}}{\partial c_i}]$$

where

$$\frac{\partial \mathbf{u}}{\partial s_1} = \frac{\partial \mathbf{u}}{\partial s_3} = \frac{\partial \mathbf{u}}{\partial s_4} = \mathbf{0}, \qquad \frac{\partial \mathbf{u}}{\partial s_2} = \begin{bmatrix} 0 \\ \cos\psi \\ \sin\psi \end{bmatrix}$$

$$\frac{\partial \mathbf{v}}{\partial s_1} = \frac{1}{s_3}\begin{bmatrix} -s_4\cos\phi \\ c_4 \\ 0 \end{bmatrix}, \qquad \frac{\partial \mathbf{v}}{\partial s_2} = \frac{1}{s_3}\begin{bmatrix} 0 \\ -c_3\cos\psi \\ -c_3\sin\psi \end{bmatrix}$$

$$\frac{\partial \mathbf{v}}{\partial s_3} = -\frac{c_3}{s_3^2}\begin{bmatrix} c_1c_4 - s_1s_4\cos\phi - c_2c_3 \\ s_4c_1\cos\phi + c_4s_1 - s_2c_3\cos\psi \\ s_4\sin\phi - s_2c_3\sin\psi \end{bmatrix}, \qquad \frac{\partial \mathbf{v}}{\partial s_4} = \frac{1}{s_3}\begin{bmatrix} -s_1\cos\phi \\ c_1\cos\phi \\ \sin\phi \end{bmatrix}$$

$$\frac{\partial \mathbf{w}}{\partial s_i} = \frac{\partial \mathbf{u}}{\partial s_i} \times \mathbf{v} + \mathbf{u} \times \frac{\partial \mathbf{v}}{\partial s_i}, \qquad \text{for } i = 1,\ 2,\ 3,\ 4.$$

Furthermore,

$$\frac{\partial \mathbf{u}}{\partial c_1} = \frac{\partial \mathbf{u}}{\partial c_3} = \frac{\partial \mathbf{u}}{\partial c_4} = \mathbf{0}, \qquad \frac{\partial \mathbf{u}}{\partial c_2} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\frac{\partial \mathbf{v}}{\partial c_1} = \frac{1}{s_3}\begin{bmatrix} c_4 \\ s_4\cos\phi \\ 0 \end{bmatrix}, \qquad \frac{\partial \mathbf{v}}{\partial c_2} = \frac{1}{s_3}\begin{bmatrix} -c_3 \\ 0 \\ 0 \end{bmatrix}$$

$$\frac{\partial \mathbf{v}}{\partial c_3} = \frac{1}{s_3}\begin{bmatrix} -c_2 \\ -s_2\cos\psi \\ -s_2\sin\psi \end{bmatrix}, \qquad \frac{\partial \mathbf{v}}{\partial c_4} = \frac{1}{s_4}\begin{bmatrix} c_1 \\ s_1 \\ 0 \end{bmatrix}$$

$$\frac{\partial \mathbf{w}}{\partial c_i} = \frac{\partial \mathbf{u}}{\partial c_i} \times \mathbf{v} + \mathbf{u} \times \frac{\partial \mathbf{v}}{\partial c_i}, \qquad \text{for } i = 1,\ 2,\ 3,\ 4$$

where $\mathbf{0}$ is the 3-dimensional zero vector.

### C.2.2 Computation of $\partial \mathbf{Q}_2/\partial\psi$

We have

$$\frac{\partial \mathbf{Q}_2}{\partial \psi} = [\frac{\partial \mathbf{u}}{\partial \psi}, \ \frac{\partial \mathbf{v}}{\partial \psi}, \ \frac{\partial \mathbf{w}}{\partial \psi}]$$

where

$$\frac{\partial \mathbf{u}}{\partial \psi} = \begin{bmatrix} 0 \\ -s_2 \sin\psi \\ s_2 \cos\psi \end{bmatrix}, \qquad \frac{\partial \mathbf{v}}{\partial \psi} = \frac{1}{s_3} \begin{bmatrix} 0 \\ s_2 c_3 \sin\psi \\ -s_2 c_3 \cos\psi \end{bmatrix}$$

$$\frac{\partial \mathbf{w}}{\partial \psi} = \frac{\partial \mathbf{u}}{\partial \psi} \times \mathbf{v} + \mathbf{u} \times \frac{\partial \mathbf{v}}{\partial \psi}$$

### C.2.3 Computation of $\partial \mathbf{Q}_2 / \partial \phi$

We have

$$\frac{\partial \mathbf{Q}}{\partial \phi} = [\frac{\partial \mathbf{u}}{\partial \phi}, \frac{\partial \mathbf{v}}{\partial \phi}, \frac{\partial \mathbf{w}}{\partial \phi}]$$

where

$$\frac{\partial \mathbf{u}}{\partial \phi} = \mathbf{0}, \qquad \frac{\partial \mathbf{v}}{\partial \phi} = \frac{1}{s_3} \begin{bmatrix} s_1 s_4 \sin\phi \\ -s_4 c_1 \sin\phi \\ s_4 \cos\phi \end{bmatrix}$$

$$\frac{\partial \mathbf{w}}{\partial \phi} = \mathbf{u} \times \frac{\partial \mathbf{v}}{\partial \phi}$$

and $\mathbf{0}$ is the 3-dimensional zero vector.

## C.3 Partial Derivatives of $[\mathbf{p}]_3$

Form the geometric relationship of Fig. 2.1, we have

$$[\mathbf{p}]_3 = \begin{bmatrix} c_5 \\ s_5 c_6 \\ s_5 s_6 \end{bmatrix}$$

Hence,

$$\frac{\partial [\mathbf{p}]_3}{\partial c_5} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \qquad \frac{\partial [\mathbf{p}]_3}{\partial c_6} = \begin{bmatrix} 0 \\ s_5 \\ 0 \end{bmatrix}$$

$$\frac{\partial [\mathbf{p}]_3}{\partial s_5} = \begin{bmatrix} 0 \\ c_6 \\ s_6 \end{bmatrix}, \qquad \frac{\partial [\mathbf{p}]_3}{\partial s_6} = \begin{bmatrix} 0 \\ 0 \\ s_5 \end{bmatrix}$$

## C.4 Computation of $s_i$ and $c_i$ and their derivatives for $i = 1, 2, \ldots, 7$

From the geometry of Fig. 21, we have the following relationships:

$$c_1 = \mathbf{a} \cdot \mathbf{d}, \qquad s_1 = \|\mathbf{a} \cdot \mathbf{d}$$

$$c_2 = \mathbf{a} \cdot \mathbf{b}, \qquad s_2 = \|\mathbf{a} \cdot \mathbf{b}$$

$$c_3 = \mathbf{b} \cdot \mathbf{c}, \qquad s_3 = \|\mathbf{b} \cdot \mathbf{c}|$$

$$c_4 = \mathbf{c} \cdot \mathbf{d}, \qquad s_4 = \|\mathbf{c} \cdot \mathbf{d}|$$

$$c_5 = \mathbf{b} \cdot \mathbf{p}_0, \qquad s_5 = \|\mathbf{b} \times \mathbf{p}_0|$$

$$c_7 = \mathbf{p}_0 \cdot \mathbf{c}, \qquad s_7 = \|\mathbf{p}_0 \times \mathbf{c}|$$

$$c_6 = \frac{\mathbf{b} \times \mathbf{c}}{\|\mathbf{b} \times \mathbf{c}\|} \cdot \frac{\mathbf{b} \times \mathbf{p}_0}{\|\mathbf{b} \times \mathbf{p}_0\|}$$

$$s_6 = \frac{\mathbf{c} \times \mathbf{p}_0 \cdot \mathbf{b}}{\|\mathbf{b} \times \mathbf{c}\| \cdot \|\mathbf{b} \times \mathbf{p}_0\|}$$

Here, the position vector of the coupler point in its initial configuration, $\mathbf{p}_0$, is included and all sines have been assumed positive, given that the associated angles are supposed to lie between 0 and $\pi$. The numerator of the expression of $s_6$ is positive since the mixed product there is in counter-clockwise order. Moreover, for $i = 1, 2, 3, 4, 5, 7$, we can directly use the following relations

$$\frac{dc_i}{d\mathbf{x}} = [(\frac{dc_i}{d\mathbf{a}})^T, \ (\frac{dc_i}{d\mathbf{b}})^T, \ (\frac{dc_i}{d\mathbf{c}})^T, \ (\frac{dc_i}{d\mathbf{d}})^T]^T$$

$$\frac{ds_i}{d\mathbf{x}} = [(\frac{ds_i}{d\mathbf{a}})^T, \ (\frac{ds_i}{d\mathbf{b}})^T, \ (\frac{ds_i}{d\mathbf{c}})^T, \ (\frac{ds_i}{d\mathbf{d}})^T]^T$$

Hence, the following is obtained:

$$\frac{dc_1}{d\mathbf{x}} = [\mathbf{d}^T, \mathbf{0}^T, \mathbf{0}^T, \mathbf{a}^T]^T, \quad \frac{ds_1}{d\mathbf{x}} = -\frac{c_1}{s_1}[\mathbf{d}^T, \mathbf{0}^T, \mathbf{0}^T, \mathbf{a}^T]^T$$

$$\frac{dc_2}{d\mathbf{x}} = [\mathbf{b}^T, \mathbf{a}^T, \mathbf{0}^T, \mathbf{0}^T]^T, \quad \frac{ds_2}{d\mathbf{x}} = -\frac{c_2}{s_2}[\mathbf{b}^T, \mathbf{a}^T, \mathbf{0}^T, \mathbf{0}^T]^T$$

$$\frac{dc_3}{d\mathbf{x}} = [\mathbf{0}^T, \mathbf{c}^T, \mathbf{b}^T, \mathbf{0}^T]^T, \quad \frac{ds_3}{d\mathbf{x}} = -\frac{c_3}{s_3}[\mathbf{0}^T, \mathbf{c}^T, \mathbf{b}^T, \mathbf{0}]^T$$

$$\frac{dc_4}{d\mathbf{x}} = [\mathbf{0}^T, \mathbf{0}^T, \mathbf{d}^T, \mathbf{c}^T]^T, \quad \frac{ds_4}{d\mathbf{x}} = -\frac{c_4}{s_4}[\mathbf{0}^T, \mathbf{0}^T, \mathbf{d}^T, \mathbf{c}^T]^T$$

$$\frac{dc_5}{d\mathbf{x}} = [\mathbf{0}^T, \mathbf{p}_0^T, \mathbf{0}^T, \mathbf{0}^T]^T, \quad \frac{ds_5}{d\mathbf{x}} = -\frac{c_5}{s_5}[\mathbf{0}, \mathbf{p}_0^T, \mathbf{0}^T, \mathbf{0}^T]^T$$

$$\frac{dc_7}{d\mathbf{x}} = [\mathbf{0}^T, \mathbf{0}^T, \mathbf{p}_0^T, \mathbf{0}^T]^T, \quad \frac{ds_7}{d\mathbf{x}} = -\frac{c_7}{s_7}[\mathbf{0}^T, \mathbf{0}^T, \mathbf{p}_0^T, \mathbf{0}^T]^T$$

where $\mathbf{0}$ stands for the 3-dimensional zero vector. For $c_6$ and $s_6$, we have the following:

$$\frac{dc_6}{d\mathbf{x}} = \left[\frac{\mathbf{b}\times\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|}\right]^T \frac{d}{d\mathbf{x}}\left[\frac{\mathbf{b}\times\mathbf{p}_0}{\|\mathbf{b}\times\mathbf{p}_0\|}\right] + \left[\frac{\mathbf{b}\times\mathbf{p}_0}{\|\mathbf{b}\times\mathbf{p}_0\|}\right]^T \frac{d}{d\mathbf{x}}\left[\frac{\mathbf{b}\times\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|}\right]$$

where

$$\frac{d}{d\mathbf{x}}\left[\frac{\mathbf{b}\times\mathbf{p}_0}{\|\mathbf{b}\times\mathbf{p}_0\|}\right] = \left[\mathbf{0}_{3\times 3}, \frac{d}{d\mathbf{b}}\left(\frac{\mathbf{b}\times\mathbf{p}_0}{\|\mathbf{b}\times\mathbf{p}_0\|}\right), \mathbf{0}_{3\times 3}, \mathbf{0}_{3\times 3}\right]$$

$$\frac{d}{d\mathbf{b}}\left(\frac{\mathbf{b}\times\mathbf{p}_0}{\|\mathbf{b}\times\mathbf{p}_0\|}\right) = \frac{1}{\|\mathbf{b}\times\mathbf{p}_0\|}\left[\mathbf{1}\times\mathbf{p}_0 + \frac{\mathbf{b}\cdot\mathbf{p}_0}{\|\mathbf{b}\times\mathbf{p}_0\|^2}(\mathbf{b}\times\mathbf{p}_0)\otimes\mathbf{b}\right]$$

$$\frac{d}{d\mathbf{x}}\left[\frac{\mathbf{b}\times\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|}\right] = \left[\mathbf{0}_{3\times 3}, \frac{d}{d\mathbf{b}}\left(\frac{\mathbf{b}\times\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|}\right), \frac{d}{d\mathbf{c}}\left(\frac{\mathbf{b}\times\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|}\right), \mathbf{0}_{3\times 3}\right]$$

$$\frac{d}{d\mathbf{b}}\left(\frac{\mathbf{b}\times\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|}\right) = \frac{1}{\|\mathbf{b}\times\mathbf{c}\|}\left[\mathbf{1}\times\mathbf{c} + \frac{\mathbf{b}\cdot\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|^2}(\mathbf{b}\times\mathbf{c})\otimes\mathbf{b}\right]$$

$$\frac{d}{d\mathbf{c}}\left(\frac{\mathbf{b}\times\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|}\right) = \frac{1}{\|\mathbf{b}\times\mathbf{c}\|}\left[\mathbf{1}\times\mathbf{b} + \frac{\mathbf{b}\cdot\mathbf{c}}{\|\mathbf{b}\times\mathbf{c}\|^2}(\mathbf{b}\times\mathbf{c})\otimes\mathbf{c}\right]$$

and

$$\frac{ds_6}{d\mathbf{x}} = \frac{1}{\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|}\left[\frac{d}{d\mathbf{x}}(\mathbf{c}\times\mathbf{p}_0\cdot\mathbf{b}) - \frac{\mathbf{c}\times\mathbf{p}_0\cdot\mathbf{b}}{\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|}\frac{d}{d\mathbf{x}}(\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|)\right]$$

with

$$\frac{d}{d\mathbf{x}}(\mathbf{c}\times\mathbf{p}_0\cdot\mathbf{b}) = [\mathbf{0}^T,\ (\mathbf{c}\times\mathbf{p}_0)^T,\ (\mathbf{p}_0\times\mathbf{b})^T,\ \mathbf{0}^T]^T$$

$$\frac{d}{d\mathbf{x}}(\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|) = \left[\mathbf{0}^T,\ \left(\frac{d}{d\mathbf{b}}(\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|)\right)^T,\ \left(\frac{d}{d\mathbf{c}}(\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|)\right)^T,\ \mathbf{0}^T\right]^T$$

$$\frac{d}{d\mathbf{b}}(\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|) = \frac{\|\mathbf{b}\times\mathbf{c}\|}{\|\mathbf{b}\times\mathbf{p}_0\|}(\mathbf{b}\cdot\mathbf{p}_0)\mathbf{p}_0 + \frac{\|\mathbf{b}\times\mathbf{p}_0\|}{\|\mathbf{b}\times\mathbf{c}\|}(\mathbf{b}\cdot\mathbf{c})\mathbf{b}$$

$$\frac{d}{d\mathbf{c}}(\|\mathbf{b}\times\mathbf{c}\|\|\mathbf{b}\times\mathbf{p}_0\|) = \frac{\|\mathbf{b}\times\mathbf{p}_0\|}{\|\mathbf{b}\times\mathbf{c}\|}(\mathbf{b}\cdot\mathbf{c})\mathbf{b}$$

and $\mathbf{0}$ is the 3-dimensional zero vector

## C.5 Partial Derivatives of the Input-Output Function $f(\psi, \phi, \{c_i\}_1^4, \{s_i\}_1^4)$

From eqs (2.1) and (2.2$a$-$d$), we obtain

$$\frac{\partial f}{\partial \psi} = -k_2\sin\psi - k_3\sin\psi\cos\phi + \cos\psi\sin\phi$$

$$\frac{\partial f}{\partial \phi} = -k_3\cos\psi\sin\phi + k_4\sin\phi + \sin\psi\cos\phi$$

$$\frac{\partial f}{\partial s_i} = \frac{\partial k_1}{\partial s_i} + \frac{\partial k_2}{\partial s_i}\cos\psi + \frac{\partial k_3}{\partial s_i}\cos\psi\cos\phi \quad \frac{\partial k_4}{\partial s_i}\cos\phi$$

$$\frac{\partial f}{\partial c_i} = \frac{\partial k_1}{\partial c_i} + \frac{\partial k_2}{\partial c_i}\cos\psi + \frac{\partial k_3}{\partial c_i}\cos\psi\cos\phi \quad \frac{\partial k_4}{\partial c_i}\cos\phi$$

for $i = 1, 2, 3, 4$, where

$$\frac{\partial k_1}{\partial s_1} = 0, \quad \frac{\partial k_1}{\partial s_2} = -\frac{c_1c_2^2c_4 + c_2c_3}{s_2^2s_4}, \quad \frac{\partial k_1}{\partial s_3} = 0, \quad \frac{\partial k_1}{\partial s_4} = \frac{c_1c_2c_4^2 + c_3c_4}{s_4^2s_2}$$

$$\frac{\partial k_2}{\partial s_1} = \frac{c_4}{s_4}, \qquad \frac{\partial k_2}{\partial s_2} = \frac{\partial k_2}{\partial s_3} = 0, \qquad \frac{\partial k_2}{\partial s_4} = -\frac{s_1 c_4^2}{s_4^2}$$

$$\frac{\partial k_3}{\partial s_1} = \frac{\partial k_3}{\partial s_2} = \frac{\partial k_3}{\partial s_3} = \frac{\partial k_3}{\partial s_4} = 0$$

$$\frac{\partial k_4}{\partial s_1} = \frac{c_2}{s_2}, \qquad \frac{\partial k_4}{\partial s_2} = -\frac{s_1 c_2^2}{s_2^2}, \qquad \frac{\partial k_4}{\partial s_3} = \frac{\partial k_4}{\partial s_4} = 0$$

$$\frac{\partial k_1}{\partial c_1} = \frac{c_2 c_4}{s_2 s_4}, \qquad \frac{\partial k_1}{\partial c_2} = \frac{c_1 c_4}{s_2 s_4}, \qquad \frac{\partial k_1}{\partial c_3} = -\frac{1}{s_2 s_4}, \qquad \frac{\partial k_1}{\partial c_4} = \frac{c_1 c_2}{s_2 s_4}$$

$$\frac{\partial k_2}{\partial c_1} = \frac{\partial k_2}{\partial c_2} = \frac{\partial k_2}{\partial c_3} = 0, \qquad \frac{\partial k_2}{\partial c_4} = \frac{s_1}{s_4}$$

$$\frac{\partial k_3}{\partial c_1} = 1, \qquad \frac{\partial k_3}{\partial c_2} = \frac{\partial k_3}{\partial c_3} = \frac{\partial k_3}{\partial c_4} = 0$$

$$\frac{\partial k_4}{\partial c_1} = 0, \qquad \frac{\partial k_4}{\partial c_2} = \frac{s_1}{s_2}, \qquad \frac{\partial k_4}{\partial c_3} = \frac{\partial k_4}{\partial c_4} = 0$$

## C.6 Partial Derivatives of the Function $h(\psi,\ \phi,\ \{c_i\}_1^6, \{s_i\}_1^6,\ \mathbf{x})$ of the Configuration Normality Condition

From eq.(4.8), we can write the following.

$$\frac{\partial h_k}{\partial \psi_k} = -\mathbf{q}_k \cdot \left[\mathbf{Q}_1 \frac{\partial}{\partial \psi}\left(\frac{d\mathbf{Q}_2}{d\psi}\right)[\mathbf{p}]_3\right]_{\psi=\psi_k}$$

$$\frac{\partial h_k}{\partial \phi_k} = -\mathbf{q}_k \cdot \left[\mathbf{Q}_1 \frac{\partial}{\partial \phi}\left(\frac{d\mathbf{Q}_2}{d\psi}\right)[\mathbf{p}]_3\right]_{\psi=\psi_k}$$

$$\frac{\partial h_k}{\partial \mathbf{x}} = -\mathbf{q}_k \cdot \left[\frac{\partial \mathbf{Q}_1}{\partial \mathbf{x}} \frac{d\mathbf{Q}_2}{d\psi}[\mathbf{p}]_3\right]_{\psi=\psi_k}$$

$$\frac{\partial h_k}{\partial s_i} = -\mathbf{q}_k \cdot \left[\frac{\partial \mathbf{Q_1}}{\partial s_i}\frac{d\mathbf{Q_2}}{d\psi}[\mathbf{p}]_3 + \mathbf{Q_1}\frac{\partial}{\partial s_i}\left(\frac{d\mathbf{Q_2}}{d\psi}\right)[\mathbf{p}]_3 + \mathbf{Q_1}\frac{d\mathbf{Q_2}}{d\psi}\frac{\partial [\mathbf{p}]_3}{\partial s_i}\right]_{\psi=\psi_k}$$

$$\frac{\partial h_k}{\partial c_i} = -\mathbf{q}_k \cdot \left[\frac{\partial \mathbf{Q_1}}{\partial c_i}\frac{d\mathbf{Q_2}}{d\psi}[\mathbf{p}]_3 + \mathbf{Q_1}\frac{\partial}{\partial c_i}\left(\frac{d\mathbf{Q_2}}{d\psi}\right)[\mathbf{p}]_3 + \mathbf{Q_1}\frac{d\mathbf{Q_2}}{d\psi}\frac{\partial [\mathbf{p}]_3}{\partial c_i}\right]_{\psi=\psi_k}$$

for $i = 1, 2, \ldots, 6$

The mixed derivatives of $\mathbf{Q_2}$ in the above equations are computed in the following sections

## C.7 Mixed Derivatives and Partial Derivatives of $\mathbf{Q_2}$

### C.7.1 Computation of $d\mathbf{Q_2}/d\psi$

From the expression of $\mathbf{Q_2}$ in Appendix A, we have

$$\frac{d\mathbf{Q_2}}{d\psi} = \left[\frac{d\mathbf{u}}{d\psi}, \frac{d\mathbf{v}}{d\psi}, \frac{d\mathbf{w}}{d\psi}\right]$$

where

$$\frac{d\mathbf{u}}{d\psi} = \begin{bmatrix} 0 \\ -s_2 \sin\psi \\ s_2 \cos\psi \end{bmatrix}$$

$$\frac{d\mathbf{v}}{d\psi} = \frac{1}{s_3}\begin{bmatrix} s_1 s_4 \sin\phi\phi' \\ -c_1 s_4 \sin\phi\phi' + s_2 c_3 \sin\psi \\ s_4 \cos\phi\phi' - s_2 c_3 \cos\psi \end{bmatrix}$$

$$\frac{d\mathbf{w}}{d\psi} = \frac{d\mathbf{u}}{d\psi} \times \mathbf{v} + \mathbf{u} \times \frac{d\mathbf{v}}{d\psi}$$

where $\phi'$ denotes $d\phi/d\psi$ and is computed in section C.8.

### C.7.2 Computation of $\partial(d\mathbf{Q}_2/d\psi)/\partial\psi$

We have

$$\frac{\partial}{\partial \psi}\left(\frac{d\mathbf{Q}_2}{d\psi}\right) = \left[\frac{\partial}{\partial \psi}\left(\frac{d\mathbf{u}}{d\psi}\right), \frac{\partial}{\partial \psi}\left(\frac{d\mathbf{v}}{d\psi}\right), \frac{\partial}{\partial \psi}\left(\frac{d\mathbf{w}}{d\psi}\right)\right]$$

where

$$\frac{\partial}{\partial \psi}\left(\frac{d\mathbf{u}}{d\psi}\right) = \begin{bmatrix} 0 \\ s_2 \cos\psi \\ s_2 \sin\psi \end{bmatrix}$$

$$\frac{\partial}{\partial \psi}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 s_4 \sin\phi(\partial\phi'/\partial\psi) \\ c_1 s_4 \sin\phi(\partial\phi'/\partial\psi) + s_2 c_3 \cos\psi \\ s_4 \cos\phi(\partial\phi'/\partial\psi) + s_2 c_3 \sin\psi \end{bmatrix}$$

$$\frac{\partial}{\partial \psi}\left(\frac{d\mathbf{w}}{d\psi}\right) = \frac{\partial}{\partial \psi}\left(\frac{d\mathbf{u}}{d\psi}\right) \times \mathbf{v} + \frac{d\mathbf{u}}{d\psi} \times \frac{\partial \mathbf{v}}{\partial \psi} + \frac{\partial \mathbf{u}}{\partial \psi} \times \frac{d\mathbf{v}}{d\psi} + \mathbf{u} \times \frac{\partial}{\partial \psi}\left(\frac{d\mathbf{v}}{d\psi}\right)$$

where $\partial\phi'/\partial\psi$ is computed in the Section C.8

### C.7.3 Computation of $\partial(d\mathbf{Q}_2/d\psi)/\partial\phi$

We have

$$\frac{\partial}{\partial \phi}\left(\frac{d\mathbf{Q}_2}{d\psi}\right) = \left[\frac{\partial}{\partial \phi}\left(\frac{d\mathbf{u}}{d\psi}\right), \frac{\partial}{\partial \phi}\left(\frac{d\mathbf{v}}{d\psi}\right), \frac{\partial}{\partial \phi}\left(\frac{d\mathbf{w}}{d\psi}\right)\right]$$

where

$$\frac{\partial}{\partial \phi}\left(\frac{d\mathbf{u}}{d\psi}\right) = \mathbf{0}$$

$$\frac{\partial}{\partial \phi}\left(\frac{d\mathbf{v}}{d\phi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 s_4\left[\cos\phi\phi' + \sin\phi(\partial\phi'/\partial\phi)\right] \\ -c_1 s_4\left[\cos\phi\phi' + \sin\phi(\partial\phi'/\partial\phi)\right] \\ s_4\left[-\sin\phi\phi' + \cos\phi(\partial\phi'/\partial\phi)\right] \end{bmatrix}$$

$$\frac{\partial}{\partial \phi}\left(\frac{d\mathbf{w}}{d\psi}\right) = \frac{d\mathbf{u}}{d\psi} \times \frac{\partial \mathbf{v}}{\partial \phi} + \mathbf{u} \times \frac{\partial}{\partial \phi}\left(\frac{d\mathbf{u}}{d\psi}\right)$$

and $\mathbf{0}$ is the 3-dimensional zero vector. The computation of $\partial\phi'/\partial\phi$ is given in the Section C.8

### C.7.4 Computation of $\partial(d\mathbf{Q}_2/d\psi)/\partial s_i$ and $\partial(d\mathbf{Q}_2/d\psi)/\partial c_i$

We have

$$\frac{\partial}{\partial s_i}\left(\frac{d\mathbf{Q}_2}{d\psi}\right) = \left[\frac{\partial}{\partial s_i}\left(\frac{d\mathbf{u}}{d\psi}\right), \quad \frac{\partial}{\partial s_i}\left(\frac{d\mathbf{v}}{d\psi}\right), \quad \frac{\partial}{\partial s_i}\left(\frac{d\mathbf{w}}{d\psi}\right)\right]$$

$$\frac{\partial}{\partial c_i}\left(\frac{d\mathbf{Q}_2}{d\psi}\right) = \left[\frac{\partial}{\partial c_i}\left(\frac{d\mathbf{u}}{d\psi}\right), \quad \frac{\partial}{\partial c_i}\left(\frac{d\mathbf{v}}{d\psi}\right), \quad \frac{\partial}{\partial c_i}\left(\frac{d\mathbf{w}}{d\psi}\right)\right]$$

for $i = 1, 2, 3, 4$, where

$$\frac{\partial}{\partial s_1}\left(\frac{d\mathbf{u}}{d\psi}\right) = \frac{\partial}{\partial s_3}\left(\frac{d\mathbf{u}}{d\psi}\right) = \frac{\partial}{\partial s_3}\left(\frac{d\mathbf{u}}{d\psi}\right) = \mathbf{0},$$

$$\frac{\partial}{\partial s_2}\left(\frac{d\mathbf{u}}{d\psi}\right) = \begin{bmatrix} 0 \\ \sin\psi \\ \cos\psi \end{bmatrix}$$

$$\frac{\partial}{\partial s_1}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_4 \sin\phi\left[\phi' + s_1(\partial\phi'/\partial s_3)\right] \\ -c_1 s_4 \sin\phi(\partial\phi'/\partial s_1) \\ s_4 \cos\phi(\partial\phi'/\partial s_1) \end{bmatrix}$$

$$\frac{\partial}{\partial s_2}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 s_4 \sin\phi(\partial\phi'/\partial s_2) \\ c_2 s_4 \sin\phi(\partial\phi'/\partial s_2) \\ s_4 \cos\phi(\partial\phi'/\partial s_2) \quad c_3 \cos\psi \end{bmatrix}$$

$$\frac{\partial}{\partial s_3}\left(\frac{d\mathbf{v}}{d\psi}\right) = -\frac{1}{s_3^2}\begin{bmatrix} s_1 s_4 \sin\phi\left[c_3 \quad s_3(\partial\phi'/\partial s_3)\right] \\ c_1 s_4 \sin\phi\left[c_3 - s_3(\partial\phi'/\partial s_3)\right] + c_2 c_3 s_4 \sin\psi \\ s_4 \cos\phi\left[c_3 \quad s_3(\partial\phi'/\partial s_3)\right] - s_2 c_3^2 \cos\psi \end{bmatrix}$$

$$\frac{\partial}{\partial s_4}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 \sin\phi\left[\phi' + s_4(\partial\phi'/\partial s_4)\right] \\ -c_1 \sin\phi\left[\phi' + s_4(\partial\phi'/\partial s_4)\right] + c_2 \sin\psi \\ \cos\phi\left[\phi' + s_4(\partial\phi'/\partial s_4)\right] \end{bmatrix}$$

$$\frac{\partial}{\partial c_1}\left(\frac{d\mathbf{u}}{d\psi}\right) = \frac{\partial}{\partial c_1}\left(\frac{d\mathbf{u}}{d\psi}\right) = \frac{\partial}{\partial c_2}\left(\frac{d\mathbf{u}}{d\psi}\right) = \frac{\partial}{\partial c_3}\left(\frac{d\mathbf{u}}{d\psi}\right) = \frac{\partial}{\partial c_4}\left(\frac{d\mathbf{u}}{d\psi}\right) = \mathbf{0}$$

$$\frac{\partial}{\partial c_1}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 s_4 \sin\phi(\partial\phi'/\partial c_1) \\ -s_4 \sin\phi\left[\phi' + c_1(\partial\phi'/\partial c_1)\right] \\ s_4 \cos\phi(\partial\phi'/\partial c_1) \end{bmatrix}$$

$$\frac{\partial}{\partial c_2}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 s_4 \sin\phi(\partial\phi'/\partial c_2) \\ c_1 s_4 \sin\phi(\partial\phi'/\partial c_2) \\ s_4 \cos\phi(\partial\phi'/\partial c_2) \end{bmatrix}$$

$$\frac{\partial}{\partial c_3}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 s_4 \sin\phi(\partial\phi'/\partial c_3) \\ c_1 s_4 \sin\phi(\partial\phi'/\partial c_3) + s_2 \sin\phi \\ s_4 \cos\phi(\partial\phi'/\partial c_3) \quad s_2 \cos\psi \end{bmatrix}$$

$$\frac{\partial}{\partial c_4}\left(\frac{d\mathbf{v}}{d\psi}\right) = \frac{1}{s_3}\begin{bmatrix} s_1 s_4 \sin\phi(\partial\phi'/\partial c_4) \\ c_1 s_4 \sin\phi(\partial\phi'/\partial c_4) \\ s_4 \cos\phi(\partial\phi'/\partial c_4)) \end{bmatrix}$$

and the computation of $\partial\phi'/\partial s_i$ and $\partial\phi'/\partial c_i$, for $i = 1,2,3,4$, is given in the Section C.8.

## C.8 Mixed Derivatives and Partial Derivatives of the Output Angle $\phi$

Resorting to the input-output function of eqs.(2.1) and (2.2*a-d*), the following relations are obtained:

$$\phi' \equiv \frac{d\phi}{d\psi} = \frac{C_1}{C_2}$$

$$\frac{\partial\phi'}{\partial\psi} \equiv \frac{\partial}{\partial\psi}\left(\frac{d\phi}{d\psi}\right) = \frac{1}{C_2}\left(C_4 - C_3\frac{d\phi}{d\psi}\right)$$

$$\frac{\partial\phi'}{\partial\phi} \equiv \frac{\partial}{\partial\phi}\left(\frac{d\phi}{d\psi}\right) = \frac{1}{C_2}\left(C_5\frac{d\phi}{d\phi} - C_6\right)$$

$$\frac{\partial\phi'}{\partial s_i} \equiv \frac{\partial}{\partial s_i}\left(\frac{d\phi}{d\psi}\right) = \frac{1}{C_2}\left(C_{7i} - C_{8i}\frac{d\phi}{d\psi}\right)$$

$$\frac{\partial\phi'}{\partial c_i} \equiv \frac{\partial}{\partial c_i}\left(\frac{d\phi}{d\psi}\right) = \frac{1}{C_2}\left(C_{9i} - C_{10i}\frac{d\phi}{d\psi}\right)$$

for $i = 1,2.3,4..$ where

$$C_1 = k_2 \sin\psi + k_3 \sin\psi \cos\phi - \cos\psi \sin\phi$$

$$C_2 = k_4 \sin\phi - k_3 \cos\psi \sin\phi + \sin\psi \cos\phi$$

$$C_3 = k_3 \sin\psi \sin\phi + \cos\psi \cos\phi$$

$$C_4 = k_2 \cos\psi + k_3 \cos\psi \cos\phi + \sin\psi \sin\phi$$

$$C_5 = k_3 \cos\psi \cos\phi \quad k_4 \cos\phi + \sin\psi \sin\phi$$

$$C_6 = k_3 \sin\psi \sin\phi + \cos\psi \cos\phi$$

$$C_{7_i} = \frac{\partial k_2}{\partial s_i} \sin\psi + \frac{\partial k_3}{\partial s_i} \sin\psi \cos\phi$$

$$C_{8_i} = -\frac{\partial k_3}{\partial s_i} \cos\psi \sin\phi + \frac{\partial k_4}{\partial s_i} \sin\phi$$

$$C_{9_i} = \frac{\partial k_2}{\partial c_i} \sin\psi + \frac{\partial k_3}{\partial c_i} \sin\psi \cos\phi$$

$$C_{10_i} = -\frac{\partial k_3}{\partial c_i} \cos\psi \sin\phi + \frac{\partial k_4}{\partial c_i} \sin\phi$$

for $i = 1.2.3.4$

# Appendix D. Jacobian Matrix G(x) of the Constraint Function

From eq (4 14), we can readily obtain the following:

$$\mathbf{G}(\mathbf{x}) = 2 \begin{bmatrix} \mathbf{a}^T & \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T \\ \mathbf{0}^T & \mathbf{b}^T & \mathbf{0}^T & \mathbf{0}^T \\ \mathbf{0}^T & \mathbf{0}^T & \mathbf{c}^T & \mathbf{0}^T \\ \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T & \mathbf{d}^T \end{bmatrix}$$

where $\mathbf{0}$ is the 3-dimensional zero vector.

# Appendix E. A Brief Description of the SPHER1 Package

SPHER1 is a Fortran 77 code for implementing the optimization scheme presented in the thesis It was used for solving several design problems, as shown in Chapter 6. The program listing is omitted from the thesis since it contains more than 5000 lines and it is available in the McRCIM network Given in this appendix is a brief description of SPHER1, which serves to outline the package and provide the reader with basic knowledge for its use Attention will focus on the main program and those subroutines which the user will deal with directly in solving a particular problem

SPHER1 can be divided into three parts $(i)$ Orthogonal-Decomposition Algorithm, $(ii)$ Function and Jacobian Matrix Evaluation, and $(iii)$ Continuation Each part performs some special tasks The main program resides in the first part, which is the beginning of the optimization procedure An outline of the main program, along with some important subroutines in each part, are given below

### 1. Orthogonal-Decomposition Algorithm (ODA)

This part of the package is for implementing the ODA in our synthesis problem, which involves an iteration scheme for computing the correction $\Delta\mathbf{x}$ in the outer optimization layer A damping loop is included in each iteration to perform necessary reduction on the computed step size $\Delta\mathbf{x}$ The main program for the optimization scheme resides in this part In order to start the optimization procedure, the following data should be input and stored in the proper array, as indicated in the instructions in the program comments

- An initial guess of the design vector $\mathbf{x}$,
- The given set $\{Q_k\}_0^m$, with $Q_0$ specified as the given point,
- The number of points in the given set, $m$;

- Some tolerances, including the one imposed on the first-order normality condition (Angeles, Anderson and Gosselin, 1987), and the one imposed on the constraint function;

- Bounds on the number of iterations $i$) the maximum number of iteration allowed in the outer optimization layer, $ii$) the maximum number of iterations allowed within each damping loop,

- Weighting matrix **W**,

- Damping coefficient, $d$.

and then, the following subroutines have to be called:

DECOMP performs the Cholesky decomposition (Dahlquist and Björck, 1974) of **W** to obtain **V** matrix for later use ($\mathbf{W} = \mathbf{V}^T\mathbf{V}$).

NGCLS4 implements the ODA.

Additionally, the routine for the continuation scheme can be called in the main program for performing the overall optimization scheme in one single run

## 2. Function and Jacobian Matrix Evaluation

This part of the package serves to compute both the objective function, $\mathbf{f}(\mathbf{x})$, and the constraint function, $\mathbf{g}(\mathbf{x})$, as well as their Jacobian matrices, $\mathbf{F}(\mathbf{x})$ and $\mathbf{G}(\mathbf{x})$, respectively, to be used by the ODA Three subroutines are written for this purpose, which are called by subroutine NGCLS4 in the first part of the package They are:

FUNFDFDX evaluates both the objective function, $\mathbf{f}(\mathbf{x})$, and its Jacobian matrix, $\mathbf{F}(\mathbf{x})$,

FUNG evaluates the constraint function, $\mathbf{g}(\mathbf{x})$;

DGDX evaluates the Jacobian matrix of the constraint function, $\mathbf{G}(\mathbf{x})$.

There are various subroutines in FUNFDFDX. Those, under the main subroutine JACBSUB, implement the detailed computations given in Appendices C and D, under the main subroutine EVLPSI are the subroutines which solve the equations of the configuration normality condition for the set $\{\psi_k\}_1^m$.

### 3. Continuation

The continuation scheme introduced in Chapter 5 is implemented in this part of the package. It can be used in two ways either as a separate program to provide data for the main program, or as a subroutine to run together with the main program. In both ways, the following data should be input

- number of desired continuation steps, $l$;
- initial guess of the linkage together with the coupler point;
- The given set $\{Q_k\}_0^m$

and they should be stored in the proper array, as indicated in the comments of the program. The output consists of the sets of dividing points corresponding to different continuation steps. The detailed computations are implemented in the subroutine EQLDVD

Additionally, a group of routines, originally from the computer package KINVERS (Anderson, 1987), are also available for evaluating the condition number of the Jacobian matrices via Householder Reflections The main subroutine for this purpose is HHCOND and is called in NGCLS4 in the first part of the package.