Proceedings of DETC'03
2003 ASME Design Engineering Technical Conferences
September 2–6, 2003, Chicago, Illinois, USA

DETC2003/DAC-48816

GEOMETRIC DESIGN OF CYLINDRIC PRS SERIAL CHAINS

Hai-Jun Su
suh@eng.uci.edu,
Robotics and Automation Laboratory
University of California Irvine

Charles W. Wampler
charles.w.wampler@gm.com,
Manufacturing Systems Research Lab
General Motors R& D Center

J. Michael McCarthy
jmmccart@uci.edu,
Robotics and Automation Laboratory
University of California Irvine

ABSTRACT

This paper considers the design of the cylindric PRS serial chain. This five degree-of-freedom robot can be designed to reach an arbitrary set of eight spatial positions. However, it is often convenient to choose some of the design parameters and specify a task with fewer positions. For this reason, we study the three through eight position synthesis problems and consider various choices of design parameters for each. A linear product decomposition is used to obtain bounds on the number of solutions to these design problems. For all cases of six or fewer positions, the bound is exact and we give a reduction of the problem to the solution of an eigenvalue problem. For seven and eight position tasks, the linear product decomposition is useful for generating a start system for solving the problems by continuation. The large number of solutions so obtained contraindicates an elimination approach for seven or eight position tasks, hence continuation is the preferred approach.

1 Introduction

This paper examines the geometric design of a five degree-of-freedom PRS serial chain constructed so that the prismatic, or sliding, joint (P) and the revolute, or hinged, joint (R) are parallel. In this configuration the center of the spherical wrist (S) moves on a right circular cylinder around the axis of the revolute joint. We call this a "cylindric" PRS chain. Our goal is to determine the dimensions and location of this cylinder so that an end-effector of the chain can reach an arbitrary set of task positions.

A cylindric PRS chain is kinematically equivalent to the CS chain, where C denotes a cylindric joint. Our choice of terminology emphasizes the fact that this chain is a special case of the general PRS chain in which the S joint lies on a skew cylinder that is traced by a circle that is tilted relative to the direction of travel along the prismatic joint.

2 Literature Review

The elementary principles of the geometric design of linkages can be found in the text by McCarthy (2000). An advanced approach by Tsai (1972) provides a foundation for the design of robotic systems; also see Tsai and Roth (1972). Krovi et al. (2001) study the design of coupled spatial RR chains, Liao and McCarthy (2001) focus on SS chains and their assembly into the single degree-of-freedom 5-SS platform, and Mavroidis et al. (2001) obtain design equations for spatial RR chains using robotic kinematics equations. Kihonge et al. (2002) provide virtual reality design environment for CC chains, and 4C linkages. Current research is focussed on generalizing these ideas to achieve task-based design for serial chains with two to five degrees of freedom.

This paper addresses the design of cylindric PRS chains, which, having the directions of the prismatic and revolute joints equal, may also be called a CS chain. The design equations for a cylindric PRS chain were studied originally by Chen and Roth (1967). They considered the design of systems for which the direction of the cylindric joint is specified by the designer and six task positions are specified. They concluded that this design problem had at most 26 solutions. A solution procedure for these design equations was presented by Nielsen and Roth (1995) using sparse matrix elimination techniques. This approach yields 175 sparse equations in 140 monomials that can be reduced to a 26×26 generalized eigenvalue problem.

In addition to the six-position problem considered in the literature, we also solve cases where from three up to the maximum of eight task positions are specified. For each of these, we consider various possibilities for specifying a subset of design parameters to exactly determine the design. (That is, the total of the number of task positions and the number of specified design parameters is eight.) This allows a designer the flexibility to trade away direct control of the design parameters to obtain more task positions or vice versa.

For all of these design problems, we show how to bound the number of solutions using a linear product decomposition. This counting method is equivalent to the "set structure" theory in Verschelde and Haegemans (1993) and is a special case of the general product decomposition theory presented in Morgan et al. (1995). For six or fewer task positions, this count is exact. Since the solution count is manageable for these cases (at most 26), we give elimination procedures for each. These procedures resemble the one in Nielsen and Roth (1995) and reduce to solving a generalized eigenvalue problem.

In order to analyze the seven and eight position design problems, we use numerical polynomial continuation, specifically the public-domain software PHC developed by Verschelde (1999). The first use of continuation to synthesize a spatial chain was the treatment of the seven-position SS chain in Wampler, et al. (1990). More recently, Lee and Mavroidis (2002) used the method for synthesizing RRR chains.

The synthesis problems solved here can be used to design a variety of mechanisms. These can range from a single PRS chain which is actuated as an open-chain, five-degree-of-freedom robot, to a one-degree-of-freedom spatial mechanism having five PRS legs in parallel. For example, three PRS chains are used as the legs of the Eclipse parallel machining center described in Ryu et al. (1998) and in Kim et al. (1999). In this case, the axes of the P and R joints are at right angles, and the skew cylinder flattens into a plane parallel to the axis of the P joint.

3 The cylindric PRS chain

A cylindric PRS chain consists of a prismatic joint and a revolute joint with parallel axes followed by a spherical wrist, see Fig. 1. Hence, the center of the sphere joint, denoted as $\mathbf{P}$, moves on a cylinder. Let $\mathbf{B}$ be a point fixed on the axis of the cylinder, $\mathbf{G}$ be a vector along the axis of the cylinder, and let $\mathbf{P}^i$ denote the position of the sphere center when the end-effector is in its i^{th} position. Then, since the radius, R, of the cylinder must remain constant from position to position, we have the constraint equation

$$\left((\mathbf{P}^i-\mathbf{B})\times\mathbf{G}\right)^2-\left((\mathbf{P}^1-\mathbf{B})\times\mathbf{G}\right)^2=0. \qquad (1)$$

In order to design a chain, we assume that we are given n goal positions defined by the transformations $[T_i], i=1,\ldots,n$. We then seek a point $\mathbf{p}$ in the moving body such that $\mathbf{P}^i=[T_i]\mathbf{p}$ satisfies the constraint equation 1 in each of the goal positions. This equation has 9 dimensional parameters: three each for $\mathbf{p}$, $\mathbf{B}$ and $\mathbf{G}$. However, there are actually only 7 independent parameters available for design purposes. In what follows, we introduce two additional linear constraints to make the parameterization unique.

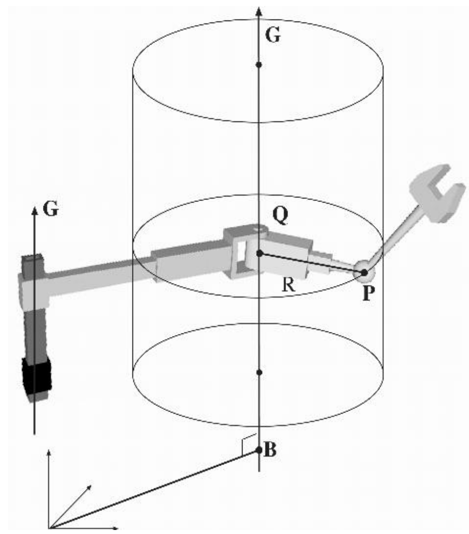

Figure 1. The cylindric PRS serial chain.

First, we note that any point along the axis of the cylinder can serve as the reference point $\mathbf{B}$. We determine a unique $\mathbf{B}$ by specifying an arbitrary plane $U:(\mathbf{n},d)$. In general, the axis of the cylinder must intersect this plane, and we select this point as $\mathbf{B}$ by including the linear equation

$$\mathbf{n}\cdot\mathbf{B}=d, \qquad (2)$$

where $\mathbf{n}$ is the unit normal to the plane and d the directed distance from the origin to the plane.

Next, we note that it is the direction of $\mathbf{G}$ that matters, not its magnitude. The vectors $\mathbf{G}$ and $\lambda\mathbf{G}$, for λ a nonzero scalar, define the same mechanism. It is not recommended that this ambiguity be resolved by requiring $\mathbf{G}$ to be a unit vector. This would result in a duplication of solutions, because $\pm\mathbf{G}$ both satisfy this condition. Instead, we introduce an arbitrary plane $V:(\mathbf{m},e)$, and note again that, in general, a line from the origin in the direction $\mathbf{G}$ must intersect this plane. Thus, we may require that $\mathbf{G}$ satisfy the linear equation

$$\mathbf{m}\cdot\mathbf{G}=e. \qquad (3)$$

Expanding Eq. 1 and cancelling $(\mathbf{B}\times\mathbf{G})^2$, we obtain the sys-

Table 1. CSn Design Problems

Name	n	g	p	b
CS3a	3	2	3	0
CS3b	3	2	2	1
CS3c	3	2	1	2
CS4a	4	2	2	0
CS4b	4	2	1	1
CS4c	4	2	0	2
CS5a	5	2	1	0
CS5b	5	2	0	1
CS6	6	2	0	0
CS7a	7	1	0	0
CS7b	7	0	1	0
CS7c	7	0	0	1
CS8	8	0	0	0

tem of design equations for n task positions,

$$\begin{aligned} \mathcal{P}_i: \quad & (\mathbf{P}^i \times \mathbf{G})^2 - (\mathbf{P}^1 \times \mathbf{G})^2 \\ & \quad + 2[(\mathbf{P}^1 - \mathbf{P}^i) \times \mathbf{G})] \cdot (\mathbf{B} \times \mathbf{G}) = 0, \quad i = 2, \ldots, n, \\ \mathcal{C}_1: \quad & \mathbf{n} \cdot \mathbf{B} = d \quad \text{and} \quad \mathcal{C}_2: \quad \mathbf{m} \cdot \mathbf{G} = e. \end{aligned} \tag{4}$$

Recall that $\mathbf{P}^i = [T_i]\mathbf{p}$. Each solution of this set of $n-1$ fourth degree polynomials $\mathcal{P}_i$, and two linear equations $\mathcal{C}_1$ and $\mathcal{C}_2$ defines a cylindric PRS serial chain that reaches the specified task positions.

Altogether, we have $n+1$ design equations and 9 unknowns, so we may prescribe a maximum of $n = 8$ positions. Alternatively, we can directly specify a subset of the dimensional parameters, and thereby reduce the number of task positions for the design problem. This allows the designer to trade control of the task against a direct influence on the mechanism's geometry. In particular, if the designer chooses to specify j dimensional parameters, or more generally, j extra linear constraints on the parameters, then the associated task would have $n = 8 - j$ positions.

To tabulate the possibilities, let $g \leq 2$, $p \leq 3$, and $b \leq 2$ be the number of components specified for $\mathbf{G}$, $\mathbf{p}$, and $\mathbf{B}$, respectively, with $n+g+p+b = 8$. The components can be specified in any reference frame; any such specification amounts to a linear equation in the reference frame we choose for computations. Following Chen and Roth (1967), we specify $\mathbf{G}$ when possible, which means we have $g = 2$ for all of $n \leq 6$. (This reduces the quartic constraint equations to quadratics.) Table 1 shows the combinations of specified parameters and tasks that are available.

4 Counting Solutions

For a system of n polynomial equations in n variables, a result known as Bezout's theorem states that the number of solutions over the complex numbers must be less than or equal to the product of the degrees d_i of the polynomials, that is $D = d_1 d_2 \ldots d_n$, which is called the total degree of the system.

More refined estimates of the number are possible by considering the structure of these polynomials. To do this, one compiles a list of the monomials that appear in each polynomial of the system. The collection of n lists of monomials forms the "monomial structure" of the system. It is a fundamental result in algebraic geometry that polynomial systems that have the same monomial structure and differ only in the scalar coefficients of the monomials, form a family for which almost all members have the same number of solutions over the complex numbers. We call this the "root count" or the generalized "Bezout number" of the family. Exceptional members of the family cannot have more than this number of isolated solutions; that is, the Bezout number is a bound on the number of isolated solutions, or roots.

Each of our design problems has a particular monomial structure with its coefficients determined by the task positions and whatever design parameters we have specified. In what follows, we determine bounds on the number of roots for each case.

4.1 Linear Product Decomposition

Bernshtein (1975) shows that the root count for any monomial structure can be obtained from the mixed volume of the associated Newton polytopes, a result sometimes known as the "BKK bound." This mixed volume is combinatorial in nature, and can require computer calculation. However, a special case known as the linear product decomposition (LPD) bound is applicable to our cylindric PRS design problem and is convenient for hand calculation.

A monomial list has a product decomposition (Morgan et al. 1995) if each monomial in this list can be obtained as a product of one element from each of two or more "factor" lists of monomials. If the monomials in these factor lists have degree at most one, then the product decomposition is *linear*, also called a set structure by Verschelde and Haegemans (1993). The important result is that the root count for the original monomial list cannot be greater than that of the product decomposition.

The linear product decomposition allows the determination of the number of roots by tabulation of all ways of choosing one linear factor from each equation such that all the variables are determined and, equivalently, none is overdetermined. Each admissible set of linear factors defines a single root. It is not necessary that each equation have the same monomial structure, and any number of linear factors can be present.

As an example, consider the pair of quadratic equations both of which have the monomial structure $\langle 1, x, y, x^2, xy \rangle$,

$$\begin{aligned} c_{11} + c_{12}x + c_{13}y + c_{14}x^2 + c_{15}xy = 0, \\ c_{21} + c_{22}x + c_{23}y + c_{24}x^2 + c_{25}xy = 0, \end{aligned} \tag{5}$$

where c_{ij} are general coefficients. The monomial structure of these equations has the linear product decomposition $\langle 1,x,y,x^2,xy\rangle = \langle 1,x\rangle\langle 1,x,y\rangle$, therefore the pair of equations

$$\begin{aligned}(d_{11}+d_{12}x)(d_{13}+d_{14}x+d_{15}y) &= 0\\ (d_{21}+d_{22}x)(d_{23}+d_{24}x+d_{25}y) &= 0,\end{aligned} \tag{6}$$

with d_{ij} as general coefficients, has the same root count as (5).

While there are four ways to choose one linear factor from each of the two equations in (6), only three such choices give a root of the system. The simultaneous choice of both linear factors in x is not admissible, because in general these give distinct solutions for x and hence have no common solution. Thus, the system (6) has a Bezout number of three, and by the product decomposition theory, so does system (5).

If, instead of system (5), we had a system of two general quadratic polynomials in two variables, the monomial structure would be $\langle 1,x,y,x^2,xy,y^2\rangle = \langle 1,x,y\rangle^2$. This has four roots, in agreement with Bezout's theorem. The reduction from four to three roots for system (5) results from the fact that y^2 is missing in both equations.

4.2 Root Counts for CSn Designs

We now consider the monomial structure of our system of design equations (4), which has the design variables $\mathbf{G} = (g_1,g_2,g_3)$, $\mathbf{B} = (b_1,b_2,b_3)$ and $\mathbf{p} = (p_1,p_2,p_3)$. The polynomials $\mathcal{P}_i$ are linear combinations of monomials in the set generated by

$$\begin{aligned}&(\langle g_1,g_2,g_3\rangle\langle 1,p_1,p_2,p_3\rangle)^2\\ &\qquad \cup \langle g_1,g_2,g_3\rangle^2\langle 1,p_1,p_2,p_3\rangle\langle b_1,b_2,b_3\rangle.\end{aligned} \tag{7}$$

Rewriting the union, we have that the design equations (4) have the monomial structure,

$$\begin{aligned}&\mathcal{P}_i \in \langle g_1,g_2,g_3\rangle^2\langle 1,p_1,p_2,p_3\rangle\langle 1,p_1,p_2,p_3,b_1,b_2,b_3\rangle,\\ &\mathcal{C}_1 \in \langle b_1,b_2,b_3,1\rangle, \quad \text{and}\\ &\mathcal{C}_2 \in \langle g_1,g_2,g_3,1\rangle.\end{aligned} \tag{8}$$

4.2.1 $3 \le n \le 6$ Task positions

We now enumerate the root count for the tasks CSn, with $3 \le n \le 6$. For these cases, we assume $\mathbf{G}$ is specified, so design equations become $\{\mathcal{P}_i, i = 2,\ldots,n \le 6; \mathcal{C}_1\}$ with the linear product structure

$$\begin{aligned}&\mathcal{P}_i \in \langle 1,p_1,p_2,p_3\rangle\langle 1,p_1,p_2,p_3,b_1,b_2,b_3\rangle,\\ &\mathcal{C}_1 \in \langle b_1,b_2,b_3,1\rangle.\end{aligned} \tag{9}$$

If less than six task positions are specified then we can have p specified components of $\mathbf{p}$, and b specified components of $\mathbf{B}$, such that $n+p+b = 6$. Thus, if n and p are given, then b is determined.

Each root of (9) is associated with an admissible set of linear terms that define the variables. For the $n-1$ equations $\mathcal{P}_i$, we can select at most $3-p$ of the factors $\langle 1,p_1,p_2,p_3\rangle$, because more that this would overdetermine the variables in $\mathbf{p}$. The second factor must be chosen in the remaining $\mathcal{P}_i$. We have no option in the selection of the factor from the last equation. These facts yield a formula for the linear decomposition bound for n task positions and p components specified, given by

$$\mathrm{LPD}(n,p) = \sum_{i=0}^{3-p}\binom{n-1}{i}, \quad n \le 6. \tag{10}$$

Table 2 lists the values given by this formula for the various design problems. The entry $D = 2^{n-1}$ is the total degree of the system—this becomes one for CS3a because $\mathbf{p}$ is completely specified. For problem CS6, the LPD bound of 26 is equal to the root count given previously by Chen and Roth (1967).

4.2.2 7 and 8 Task positions

For $n = 7$ and $n = 8$ task positions, we must include the direction $\mathbf{G}$ in the variable list which means the constraint equations are of fourth degree. The linear product structure (8) allows us to compute a bound on the number of roots to these two design problems. Notice that for CS7 we can specify only one component of $\mathbf{G}$, $\mathbf{p}$, or $\mathbf{B}$, and that for CS8 none can be specified. We consider the case of CS8 first.

We must choose the first factor $\langle g_1,g_2,g_3\rangle$ from two of the seven $\mathcal{P}_i$, which combine with $\mathcal{C}_2$ to define a root $\mathbf{G}$. Because this factor is squared, the number of choices is increased by a factor of $2^2 = 4$. Next, we can choose the second factor $\langle 1,p_1,p_2,p_3\rangle$ in up to three of the five remaining constraint equations to define $\mathbf{p}$. The third factor must be chosen for the remaining equations. This yields

$$\mathrm{LPD}(n,g,p) = \mathrm{LPD}(8,0,0) = 2^2\binom{7}{2}\sum_{i=0}^{3}\binom{5}{i} = 2184, \tag{11}$$

which is much reduced from the total degree of $4^7 = 16384$.

This formula can be generalized to include the case CS7 by noting that $n+g+p+b = 8$, where g is the number of components specified in $\mathbf{G}$, and p and b are the number of components specified in $\mathbf{p}$ and $\mathbf{B}$, as described above. Using this notation, the we must choose the first factor in $2-g$ of the $n-1$ equations $\mathcal{P}_i$, and multiply the root count by 2^{2-g} to take into account the multiplicity of this factor. The second factor may be chosen in at

Table 2. Linear-product bounds for $n \leq 6$.

Name	CS3a	CS3b	CS3c	CS4a	CS4b	CS4c	CS5a	CS5b	CS6
n	3	3	3	4	4	4	5	5	6
p	3	2	1	2	1	0	1	0	0
D	1	4	4	8	8	8	16	16	32
LPD	1	3	4	4	7	8	11	15	26

most $3-p$ of the remaining $(n-1)-(2-g)$ equations $\mathcal{P}_i$, and the third factor is chosen in the rest. Thus, we have the formula for the root count for cases CS7 and CS8 as

$$\mathrm{LPD}(n,g,p) = 2^{2-g}\binom{n-1}{2-g}\sum_{i=0}^{3-p}\binom{n+g-3}{i}, \quad n=7,8. \quad (12)$$

The values of this formula for the various designs is shown in Table 3.

Table 3. Linear-product bounds for $n = 7,8$.

Name	CS7a	CS7b	CS7c	CS8
n	7	7	7	8
g	1	0	0	0
p	0	1	0	0
Deg	4096	4096	4096	16384
LPD	312	660	900	2184

5 Solution by Elimination

For $3 \leq n \leq 6$ task positions, the LPD bound is low enough to suggest that an elimination procedure may be convenient. In what follows, we describe the mathematical framework for our elimination procedure, and then apply it to these design problems.

5.1 Eigenvalue Elimination Procedure

In the context of solving multi-loop spherical mechanisms, Wampler (2002) shows an approach to formulating the solution of a system of polynomial equations as a generalized eigenvalue problem. We apply this technique to our design equations. The procedure has the following basic steps:

1. Consider a set of m polynomials $\mathcal{P}_i$ in m variables x_i, $i = 1,\dots,m$. Choose k of these variables and form the list of all monomials up to degree μ, of which there are M= $\binom{k+\mu}{\mu}$. Multiply the given equations by this list to define the set of Mm polynomials Q_j. Now each of these polynomials can be written as a linear combination of the monomials in a list $\mathbf{y}$ of N monomials, such that $Q_j = \Sigma a_{jl} y_l$. This set of polynomials can be written in the form $A\mathbf{y} = 0$, where A is an $Mm \times N$ constant matrix.
2. Gaussian elimination of A generates a row reduced set of $r = \mathrm{rank}(A)$ independent equations $B\mathbf{y} = 0$, where B is an $r \times N$ constant matrix.
3. Select one of the variables x_k to be the eigenvariable λ and generate identities of the form $y_i - \lambda y_j = 0$, where y_i and y_j are monomials in the list $\mathbf{y}$. For example, if $\lambda = x_1$, the monomials x_1x_2 and $x_1^2x_2$ satisfy the identity $(1)x_1^2x_2 - (\lambda)x_1x_2 = 0$.
4. Append $N-r$ of these identities $[\lambda C - D]\mathbf{y} = 0$, where both C and D have entries that are simply 1 or 0, in order to define the $N \times N$ matrix equation

$$[E(\lambda)]\mathbf{y} = \begin{bmatrix} B \\ \lambda C - D \end{bmatrix}\mathbf{y} = 0. \quad (13)$$

5. If the matrix $E(\lambda)$ has full rank for arbitrary values of λ, then the $N-r$ values of λ determined by $\det[E(\lambda)] = 0$ are its generalized eigenvalues. The solutions for x_1 to the original set of polynomials $\mathcal{P}_i$ must be among these eigenvalues, and the associated eigenvector defines the values of the remaining variables x_i. The $N \times N$ problem can be reduced to an $(N-r) \times (N-r)$ eigenvalue problem, see Appendix.

This process yields the roots of the original set of polynomials, however, it may also generate extraneous roots. In particular, the monomial list $\mathbf{y}$ may contain the monomial 1 and the solution of the eigenvalue problem can yield an eigenvector that has a zero for that entry; this is a solution "at infinity."

While this procedure is general, there are three related aspects that must be adapted to a given set of polynomials: (i) the selection of monomials in Step 1, (ii) the choice of the eigenvariable, and (iii) the selection of the identities in Step 4. The rank of $E(\lambda)$ tests whether a particular choice of monomials and identities is satisfactory. However, once this is done, the formulation is valid for any general member of the family of polynomials systems related to $\mathcal{P}_i, i = 1,\dots,m$.

5.2 Application to CSn, $3 \le n \le 6$

The monomial structure of the CSn problems is given in (9). Recall that p and b denote the number of components, respectively, of $\mathbf{p}$ and $\mathbf{B}$ that are specified in the design problem. We assume that these linear equations and the linear equation $\mathcal{C}_1$ in (9) are used to eliminate p components of $\mathbf{p}$ and $b+1$ components of $\mathbf{B}$ before commencing the elimination procedure.

Elimination procedures for all the CSn problems, $3 \le n \le 6$, can be found using as multipliers monomials in the set $\langle 1, p_1, \ldots, p_{3-p}\rangle^\mu$ for some value of μ. The size of this set is M=$\binom{\mu+3-p}{3-p}$. Multiplying the $m = n-1$ equations in a CSn problem by this list yields Mm=$(n-1)\binom{\mu+3-p}{3-p}$ equations $A\mathbf{y} = 0$. This expanded set contains monomials up to degree $\mu+2$ in the components of $\mathbf{p}$ together with monomials up to degree $\mu+1$ in $\mathbf{p}$ multiplied by each of the components of $\mathbf{B}$, which yields the monomial count

$$N = \binom{\mu+5-p}{3-p} + (2-b)\binom{\mu+4-p}{3-p}. \tag{14}$$

If b_1 is used as the eigenvariable, then the identities can be obtained from the monomials in $p_1, \ldots, p_{3-p}$ up to degree $\mu+1$, or $L = \binom{\mu+4-p}{3-p}$.

We can discover elimination procedures by starting with $\mu = 0$ and working up until the rank of the $(Mm+L)\times N$ matrix of all equations and identities is N for a random choice of λ. Table 4 shows elimination procedures that were found in this way. The column labelled "Multipliers" lists the monomials used to multiply the original design equations to obtain the Mm equations $A\mathbf{y} = 0$. The eigenvariable in the column "λ" is used to formulate the identities. The column labelled "Identities" lists the monomials that play the role of y_j in the identities of the form $y_i - \lambda y_j = 0$. The last column gives the size of the final generalized eigenvalue problem, $N-r$, which in each case is equal to the LPD root count listed in Table 2.

In carrying out this procedure, it is convenient to generate all L of the identities associated with a specific eigenvariable and degree μ, and then simply check whether the system has full column rank. Once this is satisfied, one can use linear algebra to find a subset of $N-r$ identities that maintain full rank.

The Case of CS4b. To illustrate the approach, consider the case CS4b, which consists of $m = 3$ equations with one component each of $\mathbf{p}$ and $\mathbf{B}$ specified, that is $p = b = 1$. This problem has the monomial structure

$$\begin{aligned}\mathcal{P}_i &\in \langle 1, p_1, p_2\rangle\langle 1, p_1, p_2, b_1\rangle \\ &= \langle 1, p_1, p_2, b_1, p_1^2, p_1p_2, p_2^2, p_1b_1, p_2b_1\rangle,\ i = 1,2,3. \end{aligned} \tag{15}$$

Notice that the variable b_2 has been eliminated using the linear constraint $\mathcal{C}_1$ in (9). We choose b_1 as the eigenvariable, and we use the monomials $\langle 1, p_1, p_2\rangle^\mu$ as multipliers. The results as we begin with $\mu = 0$ and increment μ are summarized in the following table.

μ	N	Mm	r	$N-r$	L	rank(E)
0	9	3	3	6	3	–
1	16	9	9	**7**	6	–
2	25	18	18	**7**	10	$25 = N \leftarrow$ done.

For $\mu = 0$ and $\mu = 1$, matrix E has fewer rows than columns (i.e., $L+r < N$) so $\mu = 2$ is the first opportunity for a valid elimination. It checks, and we have an eigenvalue problem of size $N-r = 7$, which is the LPD root count.

The Case CS6. The same process can be applied to CS6, which has $m = 5$ design equations and no linear constraints on $\mathbf{p}$ or $\mathbf{B}$ (i.e., $p = b = 0$). Applying the above formulas, we have the following table:

μ	N	Mm	r	$N-r$	L	rank(E)
0	18	5	5	13	4	–
1	40	20	20	20	10	–
2	75	50	50	25	20	–
3	126	100	100	**26**	35	125
4	196	175	170	**26**	56	$196 = N \leftarrow$ done.

For $\mu \le 2$, there are not enough identities to achieve full rank, because $L < N-r$. On the other hand, while there seem to be plenty of identities for $\mu = 3$, the rank test fails. Thus, $\mu = 4$ is the first successful elimination procedure in the sequence.

The reason that $\mu = 3$ fails to result in a successful elimination can be found in the structure of the design equations which can be written in the form

$$\mathcal{P}_i = \alpha_i + \beta_i b_2, \quad i = 1, \ldots, 5. \tag{16}$$

Note that α_i is of the form $\langle 1, p_1, p_2, p_3\rangle\langle 1, p_1, p_2, p_3, b_1\rangle$ and β_i is of the form $\langle 1, p_1, p_2, p_3\rangle$.

Now, for any three $\ell, m, n \in \{1,2,3,4,5\}$, we must have the identity

$$D_{\ell mn} = \det\begin{pmatrix}\alpha_\ell & \beta_\ell & \mathcal{P}_\ell \\ \alpha_m & \beta_m & \mathcal{P}_m \\ \alpha_n & \beta_n & \mathcal{P}_n\end{pmatrix} = 0, \tag{17}$$

because the last column is a linear combination of the first two.

All terms of this determinant take the form $\langle 1, p_1, p_2, p_3\rangle^3\mathcal{P}_i$ or $b_1\langle 1, p_1, p_2, p_3\rangle^2\mathcal{P}_i$, which means that each $D_{\ell mn}$ is a linear

Table 4. Elimination formulations for Tasks $3 \le n \le 6$

Case	Multipliers	Mm	λ	Identities	N	r	$N-r$
CS3a	$\langle 1 \rangle$	2	b_1	$\langle 1 \rangle$	3	2	1
CS3b	$\langle 1 \rangle$	2	p_1	$\langle 1, p_1, b_1 \rangle$	5	2	3
CS3c	$\langle 1, p_1, p_2 \rangle$	6	p_1	$\langle 1, p_1, p_2, p_1^2 \rangle$	10	6	4
CS4a	$\langle 1 \rangle$	3	p_1	$\langle 1, p_1, b_1, b_2 \rangle$	7	3	4
CS4b	$\langle 1, p_1, p_2 \rangle^2$	18	b_1	$\langle 1, p_1, p_1^2, p_1^3, p_2, p_1 p_2, p_1^2 p_2 \rangle$	25	18	7
CS4c	$\langle 1, p_1, p_2, p_3 \rangle^2$	30	p_1	$\langle 1, p_1, p_2 \rangle^2 \cup \langle p_1^3, p_1^2 p_2 \rangle$	35	27	8
CS5a	$\langle 1, p_1, p_2 \rangle^2$	24	b_1	$\langle 1, p_1, p_2 \rangle^3 \cup \langle p_1^4 \rangle$	35	24	11
CS5b	$\langle 1, p_1, p_2, p_3 \rangle^3$	80	b_1	$\langle 1, p_1, p_2 \rangle^4$	91	76	15
CS6	$\langle 1, p_1, p_2, p_3 \rangle^4$	175	b_1	$\langle 1, p_1, p_2 \rangle^5 \cup \langle 1, p_1 \rangle^4 \langle p_3 \rangle$	196	170	26

combination of the 100 equations and 35 identities. There are $\binom{5}{3} = 10$ such relations, all independent, so the 135×126 matrix for the expanded system has row-rank of only $135 - 10 = 125$: the system is not full rank.

The case $\mu = 4$ is interesting because $r = 170 < Mm = 175$. The reason for this can again be seen in the structure of the design equations, now written in the form

$$\mathcal{P}_i = \alpha_i + \beta_i b_1 + \gamma_i b_2, \tag{18}$$

where α_i is of the form $\langle 1, p_1, p_2, p_3 \rangle^2$ and β_i, γ_i are both of the form $\langle 1, p_1, p_2, p_3 \rangle$. We now form the five identities

$$D_i = \det[\alpha_j \ \beta_j \ \gamma_j \ \mathcal{P}_j, j \neq i = 1, \ldots, 5] = 0, \quad i = 1, \ldots, 5. \tag{19}$$

The terms in these determinants are all of the form $\langle 1, p_1, p_2, p_3 \rangle^4 \mathcal{P}_i$, which are linear combinations of the expanded set of equations. Thus, the rank of the expanded set of equations is $175 - 5 = 170$. Nielsen and Roth (1995) provide a similar analysis for this design problem.

6 Solution by Continuation

For polynomial systems with a large number of roots, elimination is not attractive, but we may find all solutions using polynomial continuation. For cases with $n = 7, 8$ task positions, the LPD bounds listed in Table 3 are large, so we attack these with continuation. We do not know at the outset whether the LPD bounds are sharp. By solving a generic example of each case, we can determine the exact root count for each problem. If it were to happen that the count is small, one could then be encouraged to look for an elimination method.

As it turns out, the LPD bounds are not sharp, but the number of roots is still too large to make an elimination approach desirable. Using PHC (Verschelde 1999), we computed the roots for random test cases. For each task position, we used a random number generator to obtain 7 numbers. Three are used as the position vector and the other 4 are normalized to a unit quaternion representing spatial orientation. With probability one, such a set of tasks will be generic; that is, the number of solutions to the synthesis problem defined by the tasks will be the generic root count. PHC includes an option to use a “random linear” start system, which will give exactly the number of continuation paths as the LPD bound. Some paths diverge to infinity, leaving a reduced number of finite roots. The root counts and execution times (1.5GHz Pentium 4) for the runs are summarized in Table 5. It should be noted that the solutions for these generic test problems can be used as start points in a parameter continuation (Morgan and Sommese, 1989). The number of continuation paths will then be reduced to actual root counts, hence the approximate running time will reduce by a factor of PHC/LPD, where these mean the root counts shown in Table 5.

Table 5. Root counts from PHC for $n = 7, 8$.

Name	CS7a	CS7b	CS7c	CS8
LPD	312	660	900	2184
PHC	186	216	588	804
time	0h15m	0h50m	1h23m	4h57m

7 Java Implementation and Numerical Examples for CS6

The generalized eigenvalue solution for the case CS6 has been implemented using pure Java language. The code has been

Table 6. A set of six design positions that has 26 real solutions.

Pos	Long.(°).	Lat.(°)	Roll(°)	x	y	z
1	0.00	0.00	0.00	0.0000	0.0000	0.0000
2	23.20	64.47	-81.95	-0.3627	-0.1324	0.3325
3	-50.36	18.13	-25.06	-1.3539	-0.1925	1.4398
4	152.65	8.87	-30.05	0.6485	-0.1308	1.0832
5	76.54	40.79	171.09	-0.6574	-1.6225	1.4924
6	6.45	5.30	-8.07	0.1769	1.2474	0.8503

integrated into our synthesis software SYNTHETICA (Su et.al. 2002) that allows the designer to specify the spatial task and then view and evaluate the resulting serial chains. The software is available online at *http://synthetica.eng.uci.edu/∼mccarthy/*. Testing a large number of sample problems shows that the average running time is 40ms on a 1.5GHz Pentium 4 system.

We also wrote a Java program that generates five random task matrices (the other is fixed as identity matrix) with position limited in the box $|p_1| < 2.0, |p_2| < 2.0, 0.0 < p_3 < 2.0$ and orientation totally random. After solving about two million such task sets (took 20 hours), we found 11 examples that have 26 real solutions.

One of the problems that has all real solutions is as follows. The six task positions are listed in the Table 6. The chosen vector $\mathbf{G} = (0.7831, -0.0723, -0.6176)$, and the random plane for defining $\mathbf{B}$ is $\mathbf{n} = (0.0879, -0.3730, 1), d = -0.5144$. The 26 real solutions computed by the eigenvalue method are listed in the Table 7.

Table 7. The 26 Real Solutions

Sol.	b_1	b_2	b_3	p_1	p_2	p_3
1	0.8156	0.8727	-0.2605	-1.3815	-0.2636	1.9431
2	2.0037	-0.9764	-1.0549	0.0621	-2.0535	1.1678
3	2.0542	-1.3361	-1.1935	13.1695	6.6146	6.1159
4	2.4830	-0.7165	-1.0000	-0.0979	-0.7653	-0.2029
5	2.4838	-3.3402	-1.9789	12.7054	-8.2452	4.2742
6	2.5155	-5.4135	-2.7551	3.7900	-3.4716	-6.5247
7	2.6960	1.4774	-0.2003	-11.5314	7.4024	-2.9912
8	2.7625	-1.3573	-1.2636	1.2585	-1.1873	-1.5185
9	2.9783	-1.4512	-1.3177	0.1603	-1.3765	-0.4616
10	3.0749	-2.3461	-1.6600	1.9543	-1.8996	-2.5701
11	3.1532	2.4199	0.1111	-1.0486	0.8682	4.8628
12	-3.6638	-3.0465	-1.3288	2.0468	1.7818	-9.3171
13	3.7042	-1.2844	-1.3193	1.7046	-1.0000	-1.2753
14	3.9600	-0.6776	-1.1154	0.9110	-1.8948	2.4928
15	4.5768	-2.4599	-1.8345	2.805	-1.7922	-2.4145
16	4.7375	-0.5066	-1.1200	0.5371	-0.1493	0.4314
17	4.8660	-1.1053	-1.3546	1.9477	-9.5564	-10.1024
18	5.0184	-1.8024	-1.6281	5.9946	2.9515	-3.9823
19	5.9291	-1.6356	-1.6459	1.0035	-1.0127	0.1820
20	6.4672	-4.6470	-2.8167	5.3990	-2.6060	-4.4818
21	6.5754	-6.6294	-3.5657	8.5861	-1.5185	-0.5701
22	-12.5378	18.2489	7.3959	15.2253	4.6261	12.795
23	14.0722	2.1352	-0.9552	-3.4604	7.2038	2.2306
24	18.7135	2.2086	-1.3359	4.9241	-5.9532	-5.8661
25	-21.0900	-7.1920	-1.3430	8.7098	-15.9104	-2.7448
26	-84.5932	112.1800	48.7732	73.1091	27.374	64.3705

8 Conclusions

This paper examines the design of a cylindric PRS serial chain to reach a given set of spatial positions. A maximum of eight such task positions can be prescribed; we formulate thirteen different design problems with the number of task positions ranging from three to eight. All of these are analyzed using a linear product decomposition technique in order to determine a tighter bound on the number of solutions than the total degree. Solutions based on reduction to generalized eigenvalue problems are provided for a variety of three, four, five, and six position design problems. Polynomial homotopy continuation was used to numerically determine the solutions for the seven and eight position problems. In particular, the eight position synthesis problem has a total degree of 32768 with a linear product decomposition bound of 2184, while numerical experimentation yielded 804 solutions.

A Java implementation of the generalized eigenvalue solution for the six task position problem has been integrated into our computer aided design tool SYNTHETICA to allow the designer to specify the spatial task and then view and evaluate the resulting serial chains.

Acknowledgments

The authors gratefully acknowledge the support of the National Science Foundation and the assistance of Alba Perez.

REFERENCES

Bernshtein, D.N., 1975, “The Number of Roots of a System of Equations,” *Functional Anal. Appl.* 9(3):183–185.

Chen, P. and Roth, B., 1967, “Design Equations for Finitely and Infinitesimally Separated Position Synthesis of Binary Link and Combined Link Chains,” *ASME J. Engineering for Industry* 91:209-219.

Kihonge, J. N., Vance, J. M., and Larochelle, P. M., 2002, “Spatial Mechanism Design in Virtual Reality with Networking,” ASME J. Mechanical Design 124(3):435-440.

Kim, J., Park, F.C., and Lee, J. M., 1999, "A New Parallel Mechanism Machine Tool Capable of Five-Face Machining," *Annals of the CIRP*, 48(1):337-340.

Krovi, V., Ananthasuresh, G. K., and Kumar, V., 2001, "Kinematic Synthesis of Spatial RR Dyads for Path Following with Applications to Coupled Chain Mechanisms," *J. Mechanical Design*, 123(3):359-366.

Lee, E., and Mavroidis, D., 2002, "Solving the Geometric Design Problem of Spatial 3R Robot Manipulators Using Polynomial Homotopy Continuation," *ASME J. Mechanical Design*, 124(4):652-661.

Liao, Q. and McCarthy, J. M., 2001, "On the Seven Position Synthesis of a 5-SS Platform Linkage," *ASME J. Mechanical Design*, 123(1):74-79.

Mavroidis, C., Lee, E., and Alam, M., 2001, "A New Polynomial Solution to the Geometric Design Problem of Spatial RR Robot Manipulators Using the Denavit-Hartenberg Parameters," *ASME J. Mechanical Design*, 123(1):58-67.

McCarthy, J. M., 2000, *Geometric Design of Linkages*, Springer-Verlag, New York, 2000.

Morgan, A. P, and Sommese, A. J, 1989, "Coefficient Parameter Polynomial Continuation," *Appl. Mat. and Comput.*, 29:123–160.

Morgan, A. P, Sommese, A. J, and Wampler, C. W., 1995, "A Product-Decomposition Bound for Bezout Numbers," *SIAM J. of Numerical Analysis*, 32(4):1308-1325.

Nielsen, J. and Roth, B., "Elimination Methods For Spatial Synthesis "*Computational Kinematics* (eds. J.-P. Merlet and B. Ravani), Kluwer Academic Press, Netherlands, 1995.

Ryu, S.J, Kim, J.W., Hwang, J. C., Park, C., Cho, H. S., Lee, K., Lee, Y., Cornel, U., Park, F. C., and Kim, J., 1998, "ECLIPSE : An Overactuated Parallel Mechanism for Rapid Machining," *1998 ASME International Mechanical Engineering Congress and Exposition*, 8:681-689.

Su, H.-J., Collins, C., McCarthy, J. M., 2002,"An Extensible Java Applet for Spatial Linkage Synthesis," DETC2002/MECH-34371, *ASME Design Engineering Technical Conference, Montreal, Canada, Sept.29-Oct.02, 2002*.

Tsai, L.-W., 1972, "Design of Open Loop Chains for Rigid Body Guidance," Ph.D. Thesis, Department of Mechanical Engineering, Stanford University.

Tsai, L.-W., and Roth, B., 1972, "Design of Dyads with Helical, Cylindrical, Spherical, Revolute and Prismatic Joints," *Mechanism and Machine Theory*, 7:591-598.

Verschelde, J, and Haegemans, A., 1993, "The GBQ-Algorithm for Constructing Start Systems of Homotopies for Polynomial Systems," *SIAM J. Numerical Analysis*, 30(2):583-594.

Verschelde, J., 1999, "Algorithm 795: PHCpack: A General-Purpose Solver for Polynomial Systems by Homotopy Continuation," *ACM Trans Mathematical Software* 25(2):251-276.

Wampler, C., Morgan, A., and Sommese, A., 1990, "Numerical Continuation Methods for Solving Polynomial Systems Arising in Kinematics," *ASME J. Mechanical Design*, 112(1):59-68.

Wampler, C., 2002, "Displacement Analysis of Spherical Mechanisms Having Three or Fewer Loops," Paper DETC2002/MECH-34326, *ASME Design Engineering Technical Conference, Montreal, Canada, Sept.29-Oct.02, 2002*.

A Reduction of the Eigenvalue Problem to Size $N-r$

We wish to reduce the sparse $N\times N$ generalized eigenvalue problem of Eq.13 to size $N-r$, being the number of rows (and columns) in which the eigenvariable x appears. First, by re-ordering the monomials in $\mathbf{y}$ and the identity equations, we can always re-write the problem in block matrix form as

$$\begin{pmatrix} \hat{A}_1 & \hat{A}_2 & \hat{A}_3 & \hat{A}_4 \\ I_1x+C_1 & C_2 & 0 & 0 \\ 0 & I_2x & -I_2 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{pmatrix} = 0, \qquad (20)$$

where I_1, I_2 are identity matrices. In some cases, the last block-wise column is not present, but if it is, it must be full column rank. Using sparse Gaussian elimination, $\hat{A}_4$ can be reduced to upper triangular form yielding

$$\begin{pmatrix} \tilde{A}_{11} & \tilde{A}_{12} & \tilde{A}_{13} & U \\ \tilde{A}_{21} & \tilde{A}_{22} & \tilde{A}_{23} & 0 \\ I_1x+C_1 & C_2 & 0 & 0 \\ 0 & I_2x & -I_2 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{pmatrix} = 0, \qquad (21)$$

for some upper triangular matrix U. Pre-multiplying by the $(N-r)\times N$ matrix

$$\begin{pmatrix} 0 & 0 & I_1 & 0 \\ 0 & I_2 & 0 & \tilde{A}_{23} \end{pmatrix},$$

gives the equation

$$\begin{pmatrix} I_1x+C_1 & C_2 \\ \tilde{A}_{21} & \tilde{A}_{22}+\tilde{A}_{23}x \end{pmatrix} \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{pmatrix} = 0, \qquad (22)$$

where the trailing blocks have been dropped, since they are zero. The only computation involved is the triangularization of $\hat{A}_4$, which can be done efficiently by sparse routines. Eq.22 is the square generalized eigenvalue problem we seek.