Automated Generation of Kempe Linkages for Algebraic Curves and Surfaces*

Xiao-Shan Gao and Chang-Cai Zhu
Institute of Systems Sciences, Academia Sinica
Beijing 100080, P.R. China
E-mail: (xgao,czhu)@mmrc.iss.ac.cn

Shang-Ching Chou and Jian-Xin Ge
Department of Computer Science, Wichita State University
Wichita KS 67260
E-mail: (chou,ge)@cs.twsu.edu

Abstract

The well-known Peaucellier linkage converts a circular motion into a linear motion. A more shocking result by A. Kempe shows that any plane algebraic curve can be delineated by a linkage. In this paper, we improve Kempe's result to give a precise algorithm for generating the Kempe linkage. We prove that for an algebraic plane curve of degree n, the corresponding Kempe linkage uses at most $O(n^4)$ links. We extend Kempe's result to the 3D case: any algebraic surface can be delineated by a 3D linkage. We implement a program which can construct the Kempe linkage and animate the drawing process for a given plane curve.

Keywords CAD, linkage synthesis, Peaucellier linkage, Kempe linkage, algebraic curves and surfaces, animation, complexity.

1 Introduction

Design of linkages to generate certain curves is one of the basic and important kinds of mechanism design, which may be traced back to the ancient age of human history. However, vigorous mathematical study of linkage design began with James Watt's invention of a linkage that can draw a straight line approximately. The well-known Peaucellier linkage solves this

*This work is supported in part by the NSF Grant CCR-9901062 and by Chinese NSF (NO. 69725002).

problem exactly. The study of linkages attracted many great mathematicians of that time such as Cayley, Kempe, Hart, Darboux, Clifford, Chebsheff, etc. In particular, Kempe proved that any plane algebraic curve can be delineated with a linkage [14].

There are basically two approaches to linkage design: *exact design* and *approximate design*. In the exact design approach, we design the exact linkage to delineate a given curve. In the approximate design approach, we use simpler linkages, such as the four-bar linkage, to generate certain curves approximately.

The design of the Kempe linkage is a typical exact design approach. The approximate design has been a focus in most of the research work [1, 8, 16, 18], and is used more frequently in real mechanism design [10]. The exact design is of more theoretical importance. It is used, e.g., in the complexity analysis of robotics [12]. The exact design also has certain applications, e.g., it is used in classrooms as assistant tools [19], in computer aided instruction [6, 13], and parametric CAD [5]. The exact design also has potential industrial applications, e.g., certain high precision machines may need linkage to generate exact curves.

In this paper, we improve Kempe's result to give a precise algorithm for generating the Kempe linkage for a given algebraic curve. We prove that for an algebraic plane curve of degree n, the corresponding Kempe linkage uses at most $O(n^4)$ linkages. We also implement a program which can construct the Kempe linkage and animate the drawing process for a given plane curve. In this program, methods of automated diagram construction [3, 7, 9, 11, 15, 17] are used. The program construct the linkage using ruler and compass. The numerical optimization method [7] is used to animate the drawing process of the plane algebraic curves. In [15], construction methods are also used to simulate the movement of linkages.

We extend Kempe's result to the 3D case: any algebraic surface can be delineated by a 3D linkage. The complexity of the 3D Kempe construction is $\mathrm{O}(n^6)$.

The rest of this paper is organized as follows. In Section 2, we introduce some basic linkages. In Section 3, the 2D Kempe construction is presented. In Section 4, the 3D Kempe construction is presented. In Section 4, we introduce the program to generate the Kempe linkage for plane curves.

2 Some Basic Linkages

In this section, we introduce several basic linkages which will be used in the Kempe linkage construction. Most of them are known. However, We make some modifications to make them more robust.

2.1 Straight Line Motion Linkage

A basic question in mechanism is to design a linkage converting a circular motion into a linear motion. The most famous straight line motion linkage is the Peaucellier linkage, shown in Figure 1(a). In the diagram, $b = |PD| = |PB|$, $a = |AB| = |BC| = |CD| = |DA|$, and $|OP| = |OA| = c$. Let AC and BD meet in point E. Thus

$$PA \times PC = (PE - AE) \times (PE + AE) = PE^2 - AE^2 = b^2 - DE^2 - AE^2 = b^2 - a^2.$$

Therefore, point C is the inversion point of point A with respect to a circle whose center is point O. When point A moves on a circle passing through point P, point C's trace is a line. Condition $|OP| = |OA| = c$ ensures that point A will move on such a circle.

Because of the limited length of links, the locus of point C is a line segment called *slot*, shown as XY. This linkage is often used to constrain one endpoint to a line. When we say that one point is in a slot, it means that we use a Peaucellier linkage to do so.

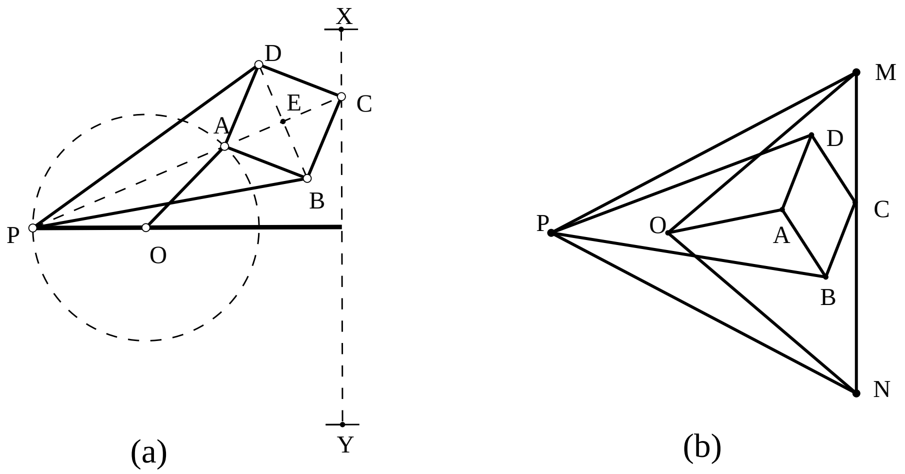

Figure 1: The Peaucellier linkage

Figure 1(b) is a modification of the Peaucellier linkage. Points P, O are attached to a link MN that is free to move in the plane. Thus, the slot itself can be moved freely. We say the slot is on a "platform".

2.2 Translator and Rotator

A translator that we can easily figure out is the linkage shown in Figure 2(a), which translates AB to ST. For this linkage, the distances between A, B and S, T are fixed. It is not convenient. A better translator is shown in Figure 2(b). For this translator, the distance between A and B does not exceed $2a$ and the distance between A and S does not exceed $2b$;

that is, the distance between A and B can change. The translator is often used to translate one link to another place. More detailed discussion about translators can be found in [12].

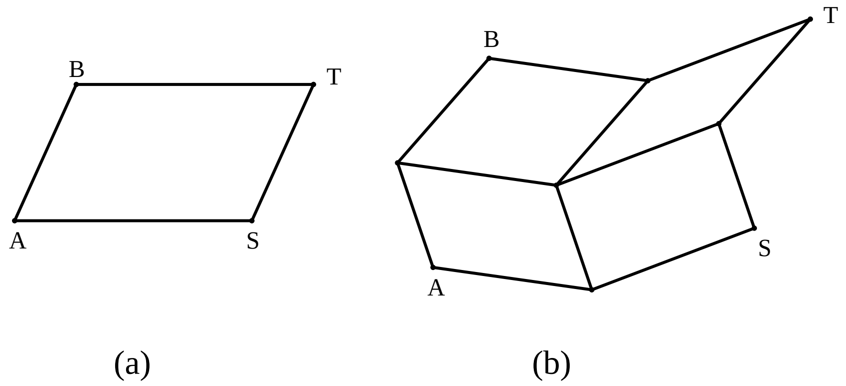

Figure 2: The translator

Another linkage is a rotator which rotates one link with some angle. Figure 3(a) is such a linkage. We have $|OA| = |OB|$, $|AC| = |BC|$, $|IR| = |IS|$, $|IU| = |IV|$, $|UR| = |VS|$. "Platforms" are used to constrain R, U in the slot along OA and to constrain S, V in the slot along OB. As a result, I will move along OC. Thus we have $|OR| = |OS|$, i.e., OS is rotated to OR by this rotator.

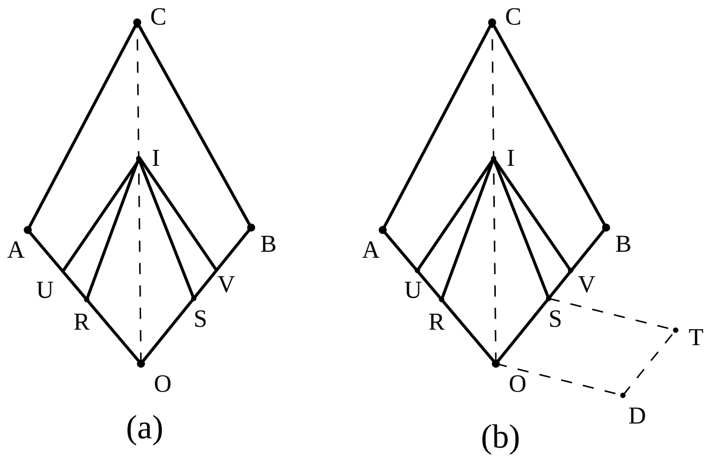

Figure 3: The distance copier

By combining a translator with a rotator, we obtain a distance copier (Figure 3(b)) where the length of DT always equals the length of OR.

2.3 Angle Multiplicator and Adder

Suppose that links OA, OB, and OC have the same length. If we attach a distance copier to AB and BC, then $\angle AOB = \angle BOC$. This may serve as an angle multiplicator. But, the angle multiplicator shown in Figure 4(a) is more simpler [14]. The links satisfy $\frac{|OC|}{|OB|} = \frac{|OB|}{|OA|}$. $ODAB$ and $OEBC$ are two isosceles trapezoids. Because $\triangle COE \sim \triangle BOD$ and $\angle COE = \angle BOD$, $\angle COB = \angle EOD$. Similarly $\triangle BOE \sim \triangle AOD$ implies $\angle BOA = \angle EOD$. From the above, we have $\angle COB = \angle BOA$. This is an angle multiplicator. With this construction, we can get an n times angle multiplicator.

Now we construct an angle adder, shown in Figure 4(b). There are two angles $\angle AOB$ and $\angle AOC$. We want to get an angle $\angle AOD = \angle AOB + \angle AOC$. First we make the lengths of four links OA, OB, OC, OD equal. Then we attach a "Distance Copier" to links AB and CD. Thus $\angle COD = \angle AOB$. Angle $\angle AOD$ is what we want.

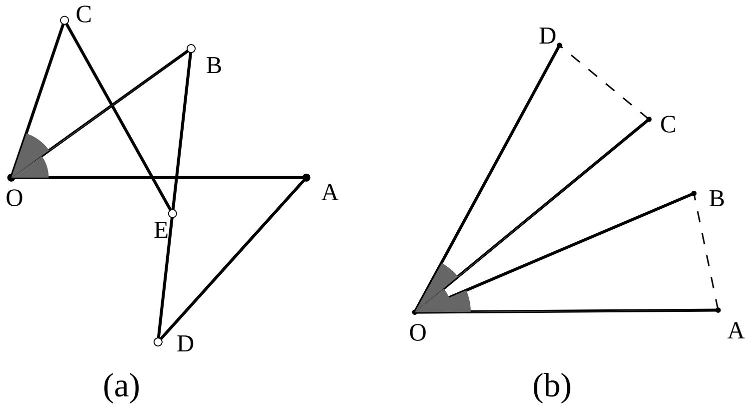

Figure 4: The angle multiplier and angle adder

3 Kempe Linkages for Plane Curves

In [14], Kempe proved that any algebraic plane curve can be generated with a linkage. Why do we emphasis algebraic curves? First, algebraic curves are the most basic and most important curves. Second, by Weierstrass' approximation theorem, for any continuous function, there is a series of polynomial functions which uniformly approximates it.

Now suppose that a curve

$$f(x,y) = \sum_{0 \le i+j \le n} a_{ij} x^i y^j = 0, a_{ij} \in R \tag{1}$$

of degree n is given. We construct a parallelogram (Figure 5(a)) such that OA and OB have lengths m and n and form angles θ and φ with the X-axis respectively. We will discuss how to determine the values of m and n later. P is a point on the curve. Its coordinates are then given below.

$$
\begin{aligned}
x &= m\cos\theta + n\cos\varphi; && (2)\\
y &= m\sin\theta + n\sin\varphi. && (3)
\end{aligned}
$$

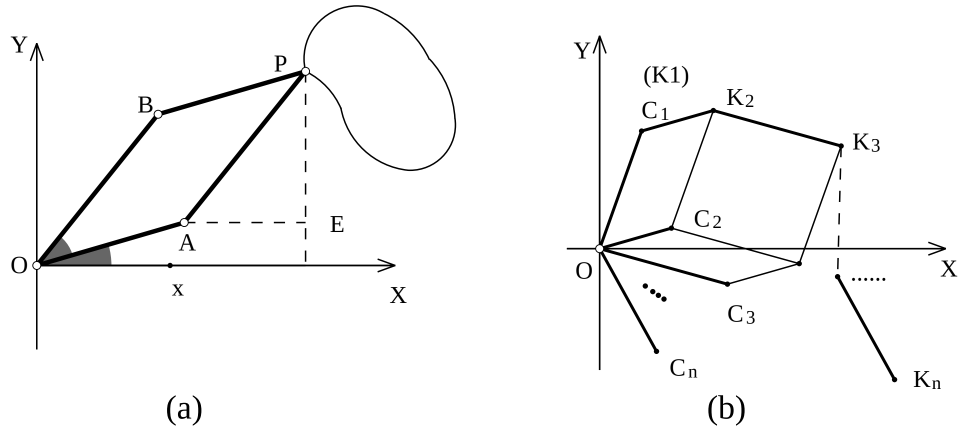

Figure 5: The Kempe linkage

Substituting equations (2) and (3) into (1) and using the following formulas,

$\sin\theta = \cos(\theta - \frac{\pi}{2})$,

$\cos^i\theta = \frac{1}{2^{i-1}}\cos i\theta + c_{i-2}\cos(i-2)\theta + \cdots$,

$\cos\theta\cdot\cos\varphi = \frac{1}{2}[\cos(\theta+\varphi) + \cos(\theta-\varphi)]$,

we shall have a sum of terms of the following form

$$
\begin{aligned}
f(x,y) &= \sum_{0\le i+j\le n} a_{ij}(l\cos\theta + m\cos\varphi)^i(l\sin\theta + m\sin\varphi)^j\\
&= \sum_{0\le i+j\le n} a_{ij}\Big(\sum_{s=0}^{i} c_i^s l^s m^{i-s}\cos^s\theta\cos^{i-s}\varphi\Big)\Big(\sum_{t=0}^{j} c_j^t l^t m^{j-t}\sin^t\theta\sin^{j-t}\varphi\Big)\\
&= \sum_{0\le i+j\le n} a_{ij}\sum_{s=0}^{i}\sum_{t=0}^{j} c_i^s c_j^t l^{s+t} m^{i+j-s-t}\cos^s\theta\cos^{i-s}\varphi\sin^t\theta\sin^{j-t}\varphi\\
&= \sum_{0<s+t\le n} (A_{s,t}\cdot\cos(s\theta\pm t\varphi) + B_{s,t}\cdot\cos(s\theta\pm t\varphi - \pi/2)) + C \qquad (4)
\end{aligned}
$$

where $A_{s,t}, B_{s,t}$ and C are constants. We may further simplify formula (4) as follows. If

$A_{s,t}B_{s,t} \neq 0$, then we have

$$A_{s,t} \cdot \cos(s\theta \pm t\varphi) + B_{s,t} \cdot \cos(s\theta \pm t\varphi - \pi/2) = \sqrt{A_{s,t}^2 + B_{s,t}^2} \cos(s\theta \pm t\varphi - \phi_{s,t})$$

where $\phi_{s,t}$ is an angle satisfies $\cos(\phi_{s,t}) = \frac{A_{s,t}}{\sqrt{A_{s,t}^2+B_{s,t}^2}}$, $\sin(\phi_{s,t}) = \frac{B_{s,t}}{\sqrt{A_{s,t}^2+B_{s,t}^2}}$. After this simplification, (4) becomes

$$f(x,y) = \sum_{0<s+t\leq n} D_{s,t} \cdot \cos(s\theta \pm t\varphi + \phi_{s,t}) + C \tag{5}$$

where $D_{s,t}$ and $\phi_{s,t}$ are constants.

For each term $D_{s,t} \cdot \cos(s\theta \pm t\varphi + \phi_{s,t})$ in (5) such that $D_{s,t} \neq 0$, we will construct a link $OE_{s,t}$ as follows.

1. Let $u, v \leq n$ be the largest integers such that $u\theta$ and $v\varphi$ occur in (5). We will use the angle multiplicator to construct links $OA_i, i = 2, \cdots, u$ and $OB_j, j = 2, \cdots, v$ such that $\angle XOA_i = i\angle XOA$ and $\angle XOB_j = j\angle XOB$. Note that we need only do this step once for the whole Kempe construction.
2. Use the angle adder, we may construct link OC such that $\angle XOC = \angle XOA_s + \angle XOB_t$.
3. Construct a triangle OCD such that $|OD| = |OC|$ and $|CD| = \sqrt{2(1-\cos\phi_{s,t})}|OC|$. Therefore, $\angle COD = \phi_{s,t}$.
4. Construct a link $OE_{s,t}$ along OD such that $|OE_{s,t}| = |D_{s,t}|$. If $D_{s,t} > 0$, points D and $E_{s,t}$ are on the same side of O. Otherwise, points D and $E_{s,t}$ are on different sides of O. It is clear that link $OE_{s,t}$ is of length $D_{s,t}$ and forms an angle $s\varphi \pm t\theta + \phi_{s,t}$ with the X-axis. Thus, the X-coordinate of point $E_{s,t}$ is $D_{s,t} \cdot \cos(s\theta \pm t\varphi + \phi_{s,t})$

Using the translators, we can construct a chain of links $OK_1, K_1K_2, \ldots, K_{n-1}K_n$ as shown in Figure 5(b), such that $K_1 = C_1$ and OC_i is translated to $K_{i-1}K_i$ with i parallelograms. Therefore, the X-coordinate of point K_n is:

$X = \sum_{0<s+t\leq n} D_{s,t} \cdot \cos(s\theta \pm t\varphi + \phi_{s,t}) = f(x,y) - C$ (by equation (5)).

If P moves along the given curve, its coordinates x, y satisfy: $f(x,y) = 0$. Accordingly, the locus of the end point K_n of the chain is

$$X + C = 0$$

a straight line parallel to the Y-axis. Conversely, if K_n moves along this line (with the help of a Peaucellier cell, for instance) point P will generate the curve $f(x,y) = 0$. We thus complete the construction.

Note that the movement of the linkage is in the reverse order of the above construction steps, i.e., the driving point for the linkage is point K_n, which will move on line $X + C = 0$.

We still need to determine the values of n and m. If the values of n and m are too small, the linkage may not reach the curve. For convenience, let us assume that $m = n$. Then $x = n(\cos\theta + \cos\varphi)$, $y = n(\sin\theta + \sin\varphi)$. To reach the curve $f(x, y) = 0$, the following equation system

$$f(x, y) = 0, x^2 + y^2 = (2n)^2 \tag{6}$$

must have solutions, and at the critical point this equation system should have solutions of multiple order. Eliminate y from (6) to get a polynomial $R(x, n)$. Let $D(n)$ be the discriminant of $R(x, n)$, considered as a polynomial of x. Let N be the minimal root of $D(n) = 0$ at which (6) has real solutions. So the n must satisfy $n > N$. A simpler way is to take a value of n larger than the maximal root of $D(n) = 0$.

To give the complexity analysis of the above procedure, first let us note that the number of items in the right side of (5) does not exceed n^2+3n. To construct $s\theta\pm t\varphi$, we need to construct $\theta, 2\theta, \cdots, n\theta, \varphi, 2\varphi, \cdots, n\varphi$. From the construction process of the angle multiplicator, it is easy to see that we need only to construct $n\theta$ and $n\varphi$. These angles need $4(n+1)$ links. The angle adder needs a fixed number of links, assuming it as N. Then it totally needs $(n^2 + 3n) \cdot 2N$ links. Finally the translators need $2 \cdot 1 + 2 \cdot 2 + \cdots + 2(n^2 + 3n - 1) = (n^2 + 3n)(n^2 + 3n - 2)$ links. Totally, the Kempe linkage needs $4(n + 1) + (n^2 + 3n) \cdot 2N + (n^2 + 3n)(n^2 + 3n - 2)$ links. So the complexity of Kempe construction is $O(n^4)$.

4 Kempe Linkages for Space Surfaces

In this section, we extend Kempe's result: for any algebraic surface defined by the polynomial equation $f(x, y, z) = 0$, we can construct a 3D linkage to delineate this surface.

4.1 Construction of Kempe Linkage

First, we will introduce several basic linkages. All the basic linkages introduced in Section 2 still work in the 3D case if we constrain the links to a plane by choosing prismatic joints.

We may use two Peaucellier linkages to form a linkage whose locus is a plane. Suppose that points P and O are fixed on a rigid body in the linkage in Figure 1(a), and this rigid body moves along a line perpendicular to PO. Then the locus of point C is a plane perpendicular to PO.

A 3D rotator linkage is needed to transform a link OA in the XY-plane to a link OB in the XZ-plane such that OB has the same length as that of OA and forms the same angle with the X-axis as that of OA. We first construct a link OB of length OA, and then construct two links BC and AC of the same length such that point C will be constrained to move on the X-axis. Point B is constrained to move on the XZ-plane by a 3D Peaucellier linkage. Then link OB satisfies all the requirements.

Now suppose that a surface $f(x, y, z) = 0$ is given. We construct three links OA, OB' and OC' in the XY plane such that OA, OB' and OC' have lengths a, b, and c and form angles α, β and γ with the X-axis respectively. Using the 3D rotator, we construct links OB and OC in the YZ and XZ planes such that OB and OC have lengths b and c and form angles β and γ with the Z-axis and X-axis.

We construct a parallelepiped $OABC$-$UVWP$ with point P on the surface. The coordinates of point P are then given below.

$$\begin{aligned}
x &= a\cos\alpha + c\cos\gamma; \\
y &= a\sin\alpha + b\sin\beta = a\cos(\pi/2-\alpha) + b\cos(\pi/2-\beta); \\
z &= b\cos\beta + c\sin\gamma = b\cos\beta + c\cos(\pi/2-\gamma).
\end{aligned}$$

Similar to Section 3, if substituting the above equations into $f(x, y, z) = 0$, we shall have a sum of terms of the following form

$$f(x, y, z) = \sum_{i=1}^{m} A_i \cdot \cos(a_i\alpha \pm b_i\beta \pm c_i\gamma \pm \epsilon_i) + C = 0 \tag{7}$$

where A_i, C, ϵ_i are constants, and a_i, b_i, c_i are positive integers.

Since OA, OB' and OC' are in the same plane, similar to the 2D case, we can construct a chain of links $OK_1, K_1K_2, \ldots, K_{n-1}K_n$ in the same plane such that the X-coordinate of K_n is

$$X = \sum_{i=1}^{m} A_i \cdot \cos(a_i\alpha \pm b_i\beta \pm c_i\gamma \pm \epsilon_i) = f(x, y, z) - C.$$

If P moves along the given surface, its coordinates x, y, z satisfy $f(x, y, z) = 0$. Accordingly, the locus of the end point K_n of the chain is

$$X + C = 0$$

which is a line in the XY plane. Conversely, if point K_n moves along this line, point P will move on $f(x, y, z) = 0$. For a fixed point K_n, there are only two constraints for the linkage. The linkage still has one degree of freedom. Since α, β, γ are symmetric, angle α is

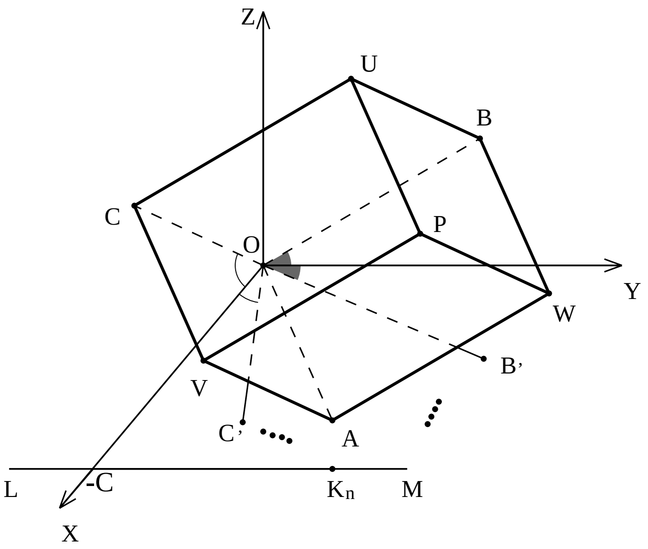

Figure 6: 3D Kempe linkage

not determined and hence link OA is movable. Therefore, to draw the surface with point P, we will move point K_n along the line $X + C = 0$ and for each position of K_n rotate link OA around point O.

4.2 Complexity Analysis of Kempe Linkage for Space Surfaces

Now we discuss the complexity of the 3D case, i.e., how many links are needed in the Kempe linkage. Similar to the 2D case, given an algebraic equation of degree n

$$\sum_{0 \le i+j+k \le n} a_{ijk} x^i y^j z^k = 0, a_{ijk} \in R,$$

let $x = a\cos\alpha + c\cos\gamma$, $y = a\sin\alpha + b\sin\beta$, $z = b\cos\beta + c\sin\gamma$. The above equation becomes

$$\begin{aligned}
f(x,y) &= \sum_{0 \le i+j+k \le n} a_{ijk}(a\cos\alpha + c\cos\gamma)^i (a\sin\alpha + b\sin\beta)^j (b\cos\beta + c\sin\gamma)^k + C \\
&= \sum_{0 \le i+j+k \le n} a_{ijk} (\sum_{r=0}^{i} c_i^r a^r c^{i-r} \cos^r\alpha \cos^{i-r}\gamma)(\sum_{s=0}^{j} c_j^s a^s b^{j-s} \sin^s\alpha \sin^{j-s}\beta) \\
&\quad (\sum_{t=0}^{k} c_k^t b^t c^{k-t} \cos^t\beta \sin^{k-t}\gamma) + C \\
&= \sum_{0 \le i+j+k \le n} a_{ijk} \sum_{r=0}^{i} \sum_{s=0}^{j} \sum_{t=0}^{k} c_i^r c_j^s c_k^t a^{r+s} b^{j-s+t} c^{i-r+k-t} \\
&\quad \cos^r\alpha \cos^{i-r}\gamma \sin^s\alpha \sin^{j-s}\beta \cos^t\beta \sin^{k-t}\gamma + C \\
&= \sum_{0 < r+s+t \le n} A_{rst} \cdot \cos(r\alpha \pm s\beta \pm t\gamma + \epsilon_{rst}) + C \qquad (8)
\end{aligned}$$

where A_{rst} and ϵ_{rst} are constants. The number of the non-constant terms in the right side of (8) does not exceed $\frac{4}{3}n(n+1)(2n+10) = m$. The angle multiplicator needs only $6(n+1)$ links; the angle adder needs a fixed number of links, say N. Then it totally needs $m \cdot 3N$ links. Finally the translators need $4 + 8 + \cdots + 4(m-1) = 2m(m-2)$ links. Therefore, the Kempe linkage needs $6(n+1) + m \cdot 2N + 2m(m-2)$ links. So the complexity of the 3D Kempe construction is $O(n^6)$.

5 Automated Generation and Simulation of Plane Kempe Linkages

We implement a program which can construct the 2D Kempe linkage and animate the drawing process for a given plane curve. The program uses our software *Geometry Expert* [4] for 2D drawings.

5.1 Automated Geometry Diagram Construction

Automated geometry diagram construction (AGDC), also called *geometric constraint solving*, is to generate diagrams automatically if the geometric constraints in the diagram are given [3]. For instance, to draw a triangle if the lengths of its three medians are given. Three main approaches have been developed for automated diagram construction: the synthetic approach [9, 17], the numerical approach[11, 7], and the symbolic approach [3].

Geometry Expert (GEX) [4] is a program for AGDC and automated geometry theorem proving and discovering. As an AGDC program, GEX can be used to build *dynamic visual models* which can be used to assist the design of mechanisms. The following functions of GEX are used to generate the Kempe linkage.

Geometric Construction. Basically speaking, GEX uses the ruler and compass construction to draw the linkage. In Sections 5.2 and 5.3, we will show how to use the following basic constructions to draw the Kempe linkage.

- Take free points.
- Take free points on a line or a circle.
- Take the intersection between lines and circles.
- Rotate a line segment.
- Ratio constructions.

Animation and Locus Generation. Through animation, the user may observe the generation process of curves or the figures of functions. In the case of the Kempe linkage, the plane curve is generated as the locus of a point during animation.

Equations Solving. To do animation, we need solutions of polynomial equation systems in real time. If the diagram can be described in a constructive way, we can find the explicit solutions of the corresponding equations and the computation is very fast. In the general case, we need to solve a system of simultaneous equations. In GEX, a method based on the optimization techniques for solving geometric constraint problems is used [7]. Experiments with this method show that it is fast and quite stable. Generally, a geometric constraint problem can be first translated into a system of equations:

$$\begin{gathered} f_1(x_1, \ldots, x_n) = 0 \\ f_2(x_1, \ldots, x_n) = 0 \\ \cdots \\ f_m(x_1, \ldots, x_n) = 0. \end{gathered}$$

Then the problem is how to solve this equation system. The optimization approach solves the equation system by converting it into finding X at which the sum of squares

$$\sigma(X) = \sum_{i=1}^{m} f_i(X)^2$$

is minimal, where $X = (x_1, x_2, \ldots, x_n)$ is the set of variables. It is obvious that the equation system has a *real* solution X^* if and only if $\min \sigma(X^*)$ is 0. The problem of solving a system of equations is thus converted into the problem of finding the minimum of a real multi-variate function. The problem now can be solved by various well-developed numerical optimization methods [2].

5.2 Kempe Linkage for Straight Lines

To generate straight lines, we do not need Kempe's general construction method. We will design a special method which does not need a general method of equation solving. Let the line be given by the following equation

$$f(x, y) = ax + by + c = 0.$$

Let $x = n\cos\theta + n\cos\varphi$, $y = n\sin\theta + n\sin\varphi$. Substituting them into $f(x, y) = 0$, we get a new equation

$$f(x, y) = n\sqrt{a^2 + b^2}(\cos(\theta - \phi) + \cos(\varphi - \phi)) + c = 0$$

where angle ϕ satisfies $\cos(\phi) = \frac{a}{\sqrt{a^2+b^2}}$, $\sin(\phi) = \frac{b}{\sqrt{a^2+b^2}}$.

To generate the line $f(x, y) = 0$, the sum of the two sides of the parallelogram $2n$ must be greater than the distance from the original point to the line, i.e., n must satisfy

$$n > \frac{|c|}{2\sqrt{a^2 + b^2}}.$$

The algorithm has two main steps.

Step 1. To use geometric constructions of GEX to draw a linkage (Figure 7).

1. Construct a line LM which is parallel with the Y-axis and has a distance of c to the Y-axis.
2. F is a free point on line LM.
3. Construct a parallelogram $OGFH$ such that $OG = GF = FH = HO = n\sqrt{a^2 + b^2}$.
4. Construction two lines OJ and OI such that $\angle GOJ = \phi$, $OJ = n$, $\angle HOI = \phi$, $OJ = n$.
5. Construct a parallelogram $PJOI$. When F moves on LM between $(-c, \sqrt{4n^2(a^2 + b^2)})$ and $(-c, -\sqrt{4n^2(a^2 + b^2)})$, the remaining points in the diagram will move according to the geometric constraints and point P will draw the line $f(x, y) = 0$.

In software GEX, the construction sequence for the above construction is as follows.

Take a point F on line LM.
Take the intersection G of two circles with centers O and F and radii $\sqrt{n^2(a^2 + b^2)}$.
Take a symmetric point H of G with line OF.
Rotate OH clockwise of an angle ϕ to obtain OI.
Rotate OG clockwise of an angle ϕ to obtain OJ.
Take the intersection of two parallel lines to obtain P.

Step 2. To use the animation and locus generation function of GEX to generate the line $f(x, y) = 0$. We will move point F from $(-c, \sqrt{4n^2(a^2 + b^2)})$ to $(-c, -\sqrt{4n^2(a^2 + b^2)})$ by a small step, say $\sqrt{4n^2(a^2 + b^2)}/50$. For each position of F, GEX will compute the positions of other points according to the above construction sequence, and the set of all positions of point P thus generated is the line. Note that to find the position for the point constructed in each of the above construction, we need only solve linear and quadratic equations. Therefore, a general equation solver is not needed.

Example 4.1. We will construct a linkage generating the line defined by $f(x, y) = x - y - 2 = 0$.

Let $x = \cos\theta + cos\varphi$, $y = \sin\theta + \sin\varphi$. Substituting them into $f(x, y) = 0$, we get a new equation

$$f(x, y) = cos(\theta + \pi/4) + cos(\varphi + \pi/4) - \sqrt{2} = 0.$$

Following the above construction procedure, we can construct a linkage, as shown in Figure 7. ABC is a right isosceles triangle whose right angle edge AB is of length 1. LM is a line parallel to the Y-axis and has a distance of $\sqrt{2}|BC|$ with the Y-axis. F is a free point on line LM. $OGFH$ is a parallelogram such that $|OG| = |FG| = |BC| = \sqrt{2}$. OGJ and OHI are isosceles triangles such that $\angle GOJ = \angle HOI = \pi/4$. $PJOI$ is a parallelogram. When point F moves on line LM, point P delineates the line $f(x, y) = 0$.

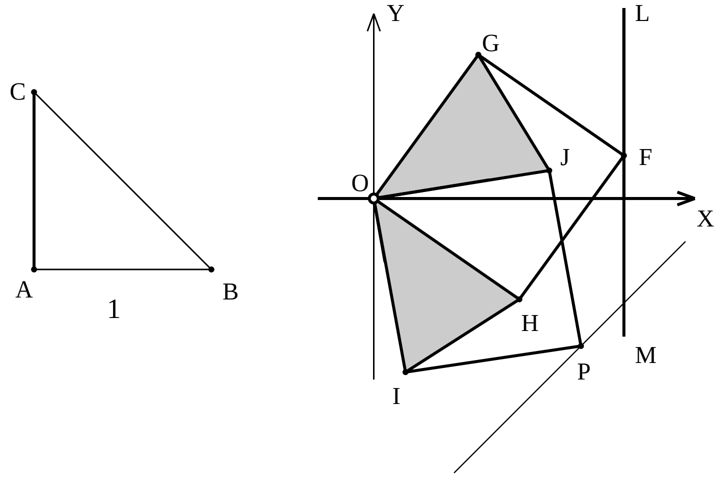

Figure 7: A Kempe linkage for a straight line

5.3 Kempe Linkage for Curves of High Degrees

We will use a simplified version of the method in Section 3 to generate a non-linear curve $f(x, y) = 0$. The idea is that since the construction here is just a simulation, we will use the simplest method to construct the linkage. The diagram thus obtained, although generates the same curve, is no linkage anymore. To show the movement of the linkage, we may just "attach" the necessary links to the diagram to form the Kempe linkage. These links are used just for display not for computation.

The construction has the following steps.

Step 1. Assuming $m = n$, generate equation (5). Determine a value n from equations (6).

Step 2. Generate the linkage.

1. Take points O and X such that O is the origin point and OX is the X-axis.

2. Construct a rhombus $OAPB$ with side n. Let θ and φ be the angles from OA, OB to the X-axis respectively.

3. To construct the linkage, we will not follow the construction procedure described in Section 3.

   Since the construction here is just a simulation, we will use the simplest method to construct the linkage.

   For instance, to construction a link OC_i such that C_i has X-coordinate $A_i \cos(a_i\theta + b_i\varphi + \phi_i)$, we need only the following constructions:

   - Rotate OA to OA' such that OA' forms angle $a_i\theta$ with the X-axis.
   - Rotate OB to OB' such that OA' forms angle $b_i\varphi$ with the X-axis.
   - Rotate OB' to OK such that $\angle KOB' = \angle A'OX + \phi_i$.
   - Take a point C_i on OK such that $|OC_i| = A_i$.

   In GEX, all the above constructions can be done automatically. For each construction, GEX will generate an equation or two equations of the involving points. The set of all these equations is the *defining equation set* for the Kempe linkage.

4. At the final step, we will construct points $K_1, \ldots, K_n$ and a line $l : x + C = 0$ as mentioned in Section 3. The construction of K_i can also be simplified. We may first construct the level-one parallelograms $OC_1C_{11}C_2$, $OC_3C_{12}C_4$, $\cdots$, and then construct the level-two parallelograms $OC_{11}C_{21}C_{12}$, $OC_{13}C_{22}C_{14}$, $\cdots$. Repeating the above process, only $log_2(n)$ level of parallelograms are needed to construct. Totally, we need to construct $O(n^2)$ points. This is much simpler than the $O(n^4)$ points in the real Kempe construction.

Step 3. Attach necessary links to form the Kempe linkage. Basically, we will construct the links missed in Step 2. We may start from the points drawn in Step 2 and use ruler-compass constructions to draw these links.

Step 4. Animation and locus generation. We will move point K_n on line $X + C = 0$ by a small step, and for each position of K_n using the optimization method in GEX to solve the defining equation system for the Kempe linkage to find positions for all points introduced in Step 2. The positions for the points introduced in Step 3 can be computed efficiently with the corresponding construction sequences. Finally, if we remember the positions for point P, these points will form the desired plane curve.

Example 4.2. We will construct a linkage to generate $f(x, y) = x^2 - y = 0$.

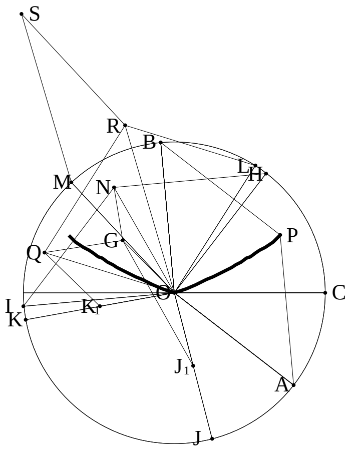

Figure 8: A Kempe linkage for a parabola

Let $x = n\cos\theta + n\cos\varphi$, $y = n\sin\theta + n\sin\varphi$. Substitute them into $f(x, y) = 0$, we have

$$f = n(\cos(\theta+\pi/2)+\cos(\varphi+\pi/2)+\frac{n}{2}\cos(2\theta)+\frac{n}{2}\cos(2\varphi)+n\cos(\theta+\varphi)+n\cos(\theta-\varphi))+n^2 = 0.$$

Therefore, the straight line l is

$$X + n^2 = 0.$$

Since this curve passes the origin point, we may assume $n = 1$. Here is how to construct this linkage.

1. Take two points O and C such that $n = |OC|$ and OC is the X-axis.

2. Take two free points A and B on the circle with center O and radius n and construct a parallelogram $OAPB$.

3. Rotate OA to OH such that $\angle HOA = \pi/2$. Rotate OB to OI such that $\angle IOB = \pi/2$. Points H and I will give terms $n\cos(\theta + \pi/2)$ and $n\cos(\varphi + \pi/2)$.

4. Rotate OA to OJ such that $\angle JOA = \angle AOC$. Take a point J_1 on OJ such that $OJ_1 = OC^2/2$. Point J_1 will give term $\frac{n^2}{2}\cos(2\theta)$.

5. Rotate OB to OK such that $\angle KOB = \angle BOC$. Take a point K_1 on OK such that $OK_1 = OC^2/2$. Point K_1 will give term $\frac{n^2}{2}\cos(2\varphi)$.

6. Rotate OB to OL and OM such that $\angle LOB = \angle AOC$ and $\angle MOB = \angle COA$. Take points L_1 and M_1 on OL and OM such that $OL_1 = OM_1 = OC^2$. Point L_1 and M_1 will give terms $n^2\cos(\theta + \varphi)$ and $n^2\cos(\theta - \varphi)$.

7. Construct five parallelograms to get the final point S. If S moves on line $X + n^2 = 0$. Point P will generate curve $y = x^2$, as shown in Figure 8.

Our experiments show that the method of equation solving based on optimization [7] is quite fast. For this example, the equation system consists of 36 quadratic equations. The average CPU time needed to compute one new position for point P is 0.3 second on a Pentium II-300 PC.

References

[1] A. K. Dhingra and N. K. Mani, Computer-aided Mechanism Design: A Symbolic-computing Approach, *Computer Aided Design*, **25**(5), 300-310, 1993.

[2] K. H. Elster (ed), *Modern Mathematical Methods of Optimization*, Akademie Verlag, 1993.

[3] X. S. Gao, Automated Geometry Diagram Construction and Engineering Geometry, in *Automated Deduction in Geometry*, pp. 226-252, Springer, 1999.

[4] X. S. Gao, J. Z. Zhang and S. C. Chou, *Geometry Expert*, Nine Chapter Pub., 1998 (in Chinese), Taiwan.

[5] X. S. Gao and C. C. Zhu, Geometric Constraint Solving with Linkages, *MM-Research Preprints*, No. 18, pp. 48–58, 1999.

[6] X. S. Gao, C. C. Zhu and Y. Huang, Building Dynamic Mathematical Models with Geometry Expert, II. Linkages, *Proc. of the Third Asian Symposium on Computer Mathematics*, (ed. Z.B. Li), pp.15-22, LanZhou Univ. Press, 1998.

[7] J. X. Ge, S. C. Chou and X. S. Gao, Geometric Constraint Satisfaction Using Optimization Methods, *Computer-Aided Design*, **31**(14), 867-879, 1999.

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

[9] C. M. Hoffmann and P. J. Vermeer, Geometric Constraint Solving in R^2 and R^3, in *Computing in Euclidean Geometry*, D.Z.Du and F.Huang(eds), Word Scientific, 1995, pp. 266-298.

[10] J. Hrones and G. L. Nelson, *Analysis of the Four-bar Linkage*, Wiley, New York, 1951.

[11] A. Heydon and G. Nelson, *The Juno-2 Constraint-Based Drawing Editor*, SRC Research Report 131a, 1994.

[12] J. Hopcroft, D. Joseph and A. S. Whitesides, Movement Problems for 2-Dimensional Linkages, *SIAM J. Comp.*, **13**(3), 610–629.

[13] M. Isoda, Developing the Curriculum for Curves Using History and Technology, *Proc. of the Third Asian Technology Conference in Mathematics*, (ed. W.C. Yang), pp. 82-89, Springer, 1998.

[14] A. B. Kempe, On a General Method of Describing Plane Curves of the nth Degree by Linkwork, *Proc. of L.M.S.*, 213–216, 1876.

[15] G. A. Kramer, *Solving Geometric Constraints Systems: A Case Study in Kinematics*, MIT Press, 1992.

[16] S. Krishnamurty and D. A. Turcic, Optimal Synthesis of Mechanisms Using Nonlinear Goal Programming Techniques, *Mech. Mach. Theory*, **27**(5), 599-612, 1992.

[17] J. Owen, *Algebraic Solution for Geometry from Dimensional Constraints*, in ACM Symp., Found of Solid Modeling, ACM Press, Austin Tx, 1991, pp. 397-407.

[18] K. Watanabe, Application of Natural Equations to the Synthesis of Curve Generating Mechanisms, *Mech. Mach. Theory*, **27**(3), 261-273, 1992.

[19] R. C. Yates, *Geometrical Tools, A Mathematical Sketch and Model Book*, Educational Pub. Inc., Saint Louis, 1949.