

Revisiting Variable Radius Circles in Constructive Geometric Constraint Solving

Ching-Shoei Chiang
Department of Computer and Information Science
Soochow University
Taiwan, R.O.C.

Robert Joan-Arinyo
Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain

`chiang@cis.scu.edu.tw`, `robert@lsi.upc.es`

March 13, 2002

Abstract

Variable-radius circles are common constructs in planar constraint solving and are usually not handled fully by algebraic constraint solvers. We give a complete treatment of variable-radius circles when such a circle must be determined simultaneously with placing two groups of geometric entities. The problem arises for instance in solvers using triangle decomposition to reduce the complexity of the constraint problem.

This work offers a set of basic constructive methods that permits to determine variable radius circles simultaneously with placing two rigid geometric objects when geometric constraints are defined on both the circumference and center point of the constraint circle. The problem has been classified by the geometric entities in two groups, one is fixed and the other has translational and rotational movement, so that the variable radius circles satisfy the constraints on the geometric entities in these two groups. The number of solutions for each problem is also given.

Keywords: Geometric constraint solving, variable radius circles, constructive solvers, algebraic solvers, cyclographic maps.

1 Introduction

In constraint-based geometric design, the designer creates a rough sketch of an object made out of simple geometric elements. Then the intended exact shape is specified by annotating the sketch with constraints. A geometric constraint solver then checks whether the set of geometric constraints coherently defines the object and, if so, determines the position of the geometric elements.

Many techniques have been reported in the literature that provide powerful and efficient methods for solving systems of geometric constraints. For example, see [2] and references therein for an extensive analysis of work on constraint solving. Among them, our interest focuses on *constructive* techniques,

Constructive solvers have two major components: the analyzer and the constructor. The analyzer symbolically determines whether a geometric problem defined by constraints is solvable. If the problem is solvable, the output of the analyzer is a sequence of construction steps, known as the *construction plan*, that places each geometric element in such a way that constraints are satisfied. After assigning specific values to the parameters, the constructor interprets the construction plan and builds an object instance, provided that no numerical incompatibilities arise.

The complexity of geometric constraint solving is doubly exponential, a feat that derives from the ability to express polynomial algebraic equations by geometric constraint configurations. As a result, it is accepted that practical solvers are not complete, that is, they solve a subclass of geometric problems.

A practical useful class of problems are twodimensional constraint problems where the geometric elements are points, straight lines, and circles with fixed radii, and in which the constraints are like distance between two points, distance from a point to a line, angle between two lines, line-circle tangency and so on.

Various extensions to the geometric repertoire that constructive solvers can handle in two dimensions can be considered. Examples are circles with variable radius, conics and Bézier curves. Of them, variable radius circles are common constructs in two dimensional constraint solving and are usually not handled fully by constructive solvers. Probably they are the most useful extension as they permit auxiliary construction in addition, as explained

by Hoffmann and Vermeer, [9], Hoffmann and Joan-Arinyo, [8], and Joan-Arinyo and Soto, [10].

When the underlying solver is numerical and good initial guesses are available for the geometric elements, variable radius circles pose no particular problem. But the numerical approach to solving constraints has many drawbacks, including reliance on good starting values and the inability to explore solution variants, [2]. What is needed is a constructive solution, preferable one in which there is no need to solve high-degree polynomials.

Recently, Hoffmann and Chiang, [6, 7], reported on an extension of the basic constructs to deal with variable radius circles in constructive solvers. Here, the constraints on the variable radius circle are placed only on the circumference.

In this work we further extend the basic constructions to consider constraint problems in which variable radius circles occur with constraints defined on their center points.

The rest of the paper is organized as follows. In Section 2 we give a short overview on related work. Next in Section 3 we define a minimal set of tools a user interface should provide to define variable radius circles. Section 4 recalls the fundamental concepts of the cyclographic model geometry we will make use of. General algorithms for the constructions which solve the problem considered here are given in Section 5. In Section 6 we present solving strategies to keep to a minimum the complexity of the algorithms implementation. Finally we offer some conclusions in Section 7.

2 Prior Work

There is a paucity of published works reporting on variable radius circles in constructive geometric constraint solving. Ramanathan, [14], studied the Apollonius problem which consists in constructing a circle tangent to three given circles. The work addressed two problems: Devising a coordinate-independent enumeration method of the eight possible solutions and performing the computations efficiently. The technique was applied to constraint-based, variable radius fitting of fillets to two lines.

Joan-Arinyo and Soto described in [10] a hybrid technique that allows to solve constraint problems involving geometric elements with more than two degrees of freedom. In particular it is shown how the method solves variable

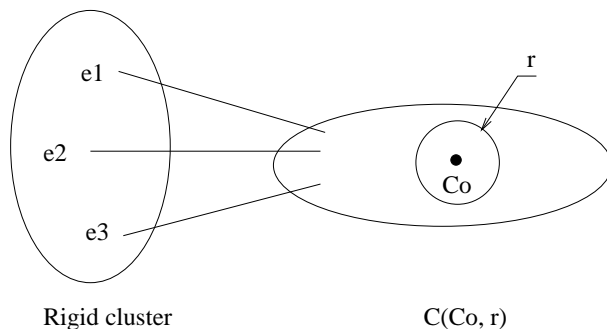


Figure 1: Variable radius circle $C(C_0, r)$ attached to one cluster.

radius circles attached to one geometric object which is determined up to position and orientation, from now on referred to as a *cluster* [3], through three constraints. See Figure 1.

Hoffmann and Chiang recently, [6, 7], reported on a more general approach to constructively solving constraint problems involving variable radius circles. The approach uses cyclographic maps, a special case of Laguerre geometry, [5], and handles the situation where the variable radius circle, $C(C_0, r)$, (see Figure 2) is attached to two clusters, S_1 and S_2 , which share a common geometric element, E . The total number of degrees of freedom that need to be canceled for S_1 , S_2 and the variable radius circle to define a cluster is four: three for the circle itself plus one due to the possible relative motion between S_1 and S_2 along E . These constraints are canceled by attaching the variable radius circle to each cluster through two constraints. Note that, otherwise the problem could be reduced to the previous case. A limitation of the method is that it only considers constraints placed on the circumference of the variable radius circle.

The DCM is a commercial solver, [1], which permits sequential constructions of variable radius circles when they are attached to one given cluster through three constraints. As far as we know, [11, 12], no details have been disclosed about how the variable radius circles are handled.

3 Definition of Variable Radius Circles

To define geometric problems involving circles with variable radius, the user interface should provide an appropriate set of tools. A minimal set of tools

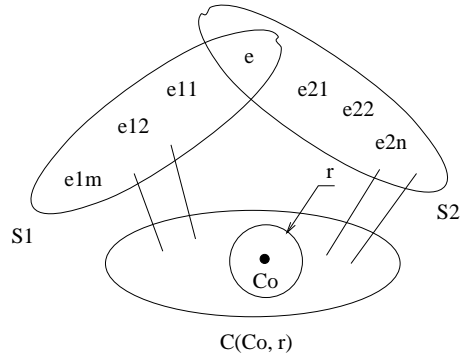


Figure 2: Variable radius circle $C(C_0, r)$ attached to two clusters.

would include an explicit command to trigger the variable radius circle definition along with operations to place geometric constraints on its circumference and on its center.

We assume that the geometric elements from which constraint problems are built are points, straight lines and circles. A sufficient set of constraints to define variable radius circles includes tangencies and distances.

3.1 Constraints Placed on the Circumference

We define the tangency constraints placed on the circumference as follows. See Figure 3. The circumference of the variable radius circle can be tangent to

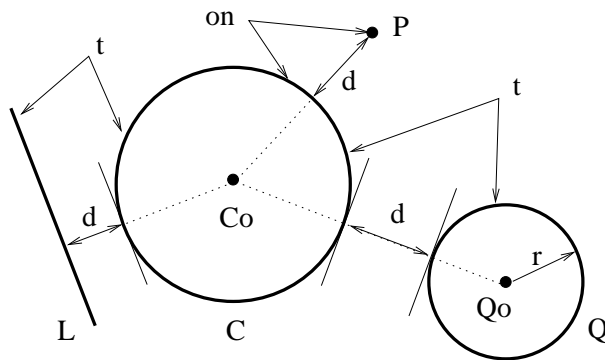


Figure 3: Constraints placed on the circumference of a variable radius circle.

- The circumference of a fixed radius circle.
- A straight line.
- A point. This is the usual *on* constraint.

Distance constraints placed on the circumference of the variable radius circle, see Figure 3, are defined as follows

- Distance to a fixed radius circle Q : The minimum distance between the two circles measured along the straight line defined by their center points.
- Distance to a straight line L : The minimum distance between the circle and the line measured along the perpendicular to the given line through the circle center point.
- Distance to a point P : Distance between the circle and the point measured along the straight line defined by the point and the center of the circle.

3.2 Constraints Placed on the Center Point

Center points of variable radius circles have no privileges over other points. Therefore the set of constraints that apply to generic points apply also to center points of circles with variable radius. We assume that the constraints available at the user interface are, see Figure 4,

- The center point can be at a given distance from another geometric element.
- The center point can be on (tangent to) another geometric element.

To facilitate the user interaction, other constraints could be added to the repertoire so far presented.

4 The Cyclographic Model

Several geometric design problems can be solved in a surprisingly simple way if one uses Laguerre geometry. A specific case of this geometry, known

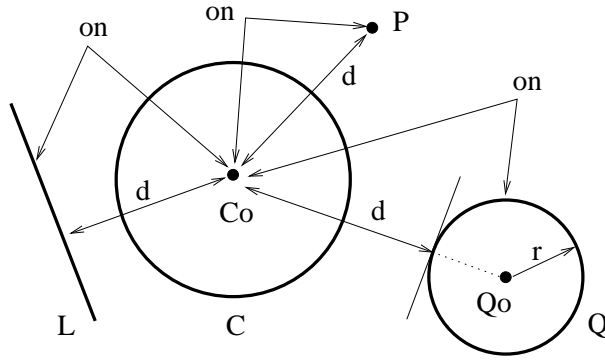


Figure 4: Constraints placed on the center of a variable radius circle.

as the *cyclographic model*, results particularly useful to solve the problem we have at hand, [14].

For the sake of completeness, we recall the fundamental concepts of the cyclographic model for the embedding of space \mathbb{R}^2 in \mathbb{R}^3 we will make use of. For a general and more in depth discussion on the cyclographic model and its applications to computer aided geometric design see Hoffmann [5], and Pottmann and Peternell [13].

The fundamental elements in \mathbb{R}^2 we consider are *rays* and *cycles*. A ray is an oriented straight line. A cycle is an oriented circle or a point (cycle with radius 0). The orientation is fixed by a unit normal vector field in the case of rays and by a signed radius in the case of the cycle.

The basic relation is that of oriented contact between cycles and rays. Refer to Figure 5. An oriented cycle and a ray are in oriented contact, if they are tangent and the unit normals coincide at the point of contact. For a point and a ray, oriented contact equals incidence.

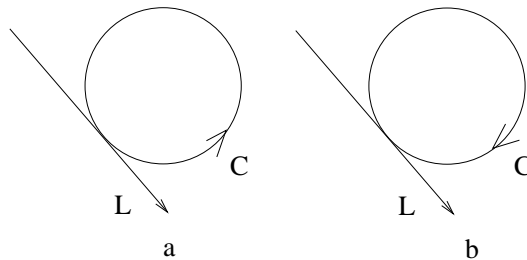


Figure 5: Contact ray-cycle. a) Oriented. b) No oriented.

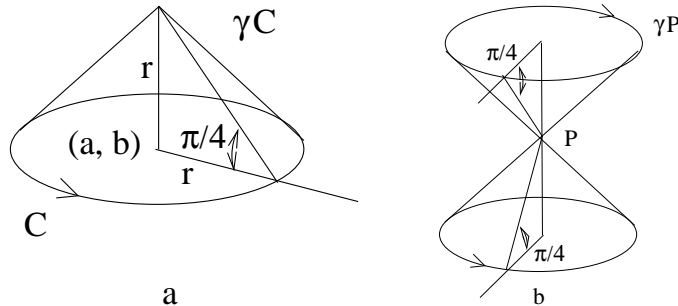


Figure 6: Cyclographic maps. a) Cycle. b) Point.

Let $C(a, b, r)$ denote a cycle with center point (a, b) and signed radius r . We assume that when $r > 0$, the cycle is oriented counterclockwise; if $r < 0$, the cycle is oriented clockwise. When $r = 0$, the cycle represents a point and is considered to have both orientations simultaneously.

With each cycle $C(a, b, r)$ there is an associated *cyclographic map*, denoted by γC , defined as the cone whose apex is the point (a, b, r) in \mathbb{R}^3 , whose axis is parallel to the Z axis and whose angle is equal to $\pi/4$. Figure 6 illustrates this concept.

Consider the line in \mathbb{R}^2 whose equation is $ax + by + c = 0$. Note that, depending on the orientation, a straight line can support two different rays. The orientation of a ray, given by its direction vector, is defined as the vector $[b, -a]$; that is, the result of rotating clockwise by 90° the vector $[a, b]$, which is normal to the line. We shall denote a ray by $L(a, b, c)$ or just L .

With each ray $L(a, b, c)$ there is an associated *cyclographic map*, denoted by γL , defined as the plane in \mathbb{R}^3 which intersects the XY plane at L and at an angle with $[b, -a]$ equal to $\pi/4$. See Figure 7.

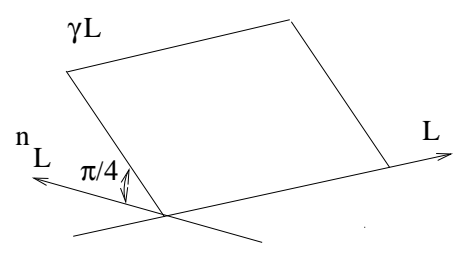


Figure 7: Cyclographic map of a ray.

The distance of a point to a ray is measured as a positive quantity if the point is to the left of the ray as seen in the ray's orientation. The angle between a pair of rays, $\angle(L_i, L_j)$ is measured from the direction of L_i clockwise to the direction of L_j .

5 Solving Variable Radius Circles

The set of constraints given in Section 3 refer to the tools available at the user interface to provide a friendly interaction. To facilitate the solving process, we first show how to transform the problem defined at the user interface into an equivalent problem, where distance constraints on the variable radius circle are expressed as tangencies.

Next, to handle the constraints defined on the center point of variable radius circles in a uniform and consistent way, we extend the cyclographic model with two new auxiliary maps.

Then we give general algorithms that compute variable radius circles that are attached to two rigid clusters through constraints placed on both, the circumference and the center point of the circle. Following [6, 7], we consider two different scenarios: 1) The geometric element E shared by clusters S_1 and S_2 , see Figure 2, is a straight line and relative motion is translational, and 2) E is a circle or point and the relative motion of clusters is a rotation.

5.1 Problem Transformation

First we consider the distance constraints placed on the circumference of the variable radius circle to be determined, $C(C_0, r)$. The circle-line distance constraint, $dis(C, L) = d$, see Figure 8, is equivalent to a tangency constraint between the circle C and a line L' which has been translated a distance d along its normal. Let $Q(Q_0, r)$ be a circle with center Q_0 and fixed radius r . The distance constraint $dis(C, Q(Q_0, r)) = d$ is transformed into the tangency constraint $t(C, Q(Q_0, r + d))$. Finally, the distance constraint between C and point P , $dis(C, P) = d$, is transformed into the equivalent tangency constraint $t(C, Q(P, d))$.

Distance constraints placed on the center point C_0 of $C(C_0, r)$ are transformed into tangency (on) constraints as follows. The center point-point distance, $dis(C_0, P) = d$ is transformed into $t(C_0, Q(P, d))$. The center point-line constraint $dis(C_0, L) = d$ is transformed into $t(C_0, L')$, where L'

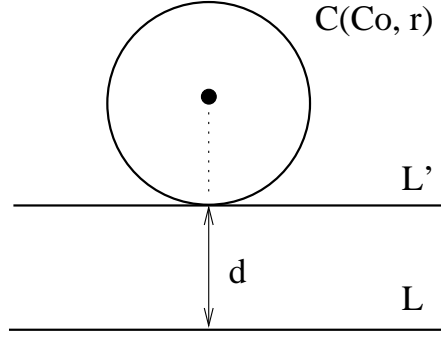


Figure 8: Transforming constraint $dis(C, L) = d$ into $t(C, L')$.

is the line resulting from translating L a distance d along its normal. The center point-circle distance constraint $dis(C_0, Q(Q_0, r)) = d$ is transformed into the tangency $t(C_0, Q(Q_0, r + d))$.

All the transformations are performed with respect to the cluster the geometric elements belong to.

5.2 Auxiliary Maps

Hoffmann and Chiang's solution to the construction of variable radius circles, [6, 7], relies on the following fact. Let C be a variable radius circle. Let E be either a ray or a fixed radius circle related to the circumference of C by a tangency constraint. Then, if C and E are placed with respect to each other in such a way that the tangency constraint holds as an oriented contact, their γ -cyclographic maps, γC and γE , are tangent regardless the relative location of C and E or which the radius of C is.

γ -cyclographic maps capture properties derived from the cycle-cycle or ray-cycle oriented tangency relationships, but they do not capture directly the relationships between the center of a cycle and other geometric elements. Since a variable radius cycle and its center are related through an unknown distance, constraints placed between the center point of a cycle and other geometric elements cannot be translated into γ -cyclographic maps. Note, for example, that the distance between the center point of C and a ray or a fixed radius circle depends on their relative position. Therefore, the abovementioned fact no longer holds. To handle in a uniform way constraints placed on the center point, on the circumference or on both, we extend the cyclographic model with two new auxiliary maps.

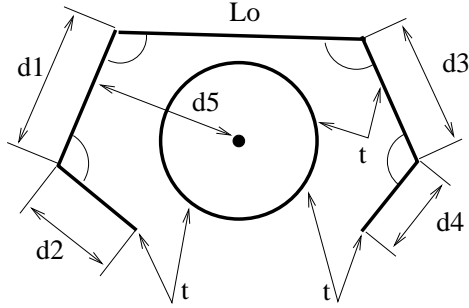


Figure 9: Translational Exemple.

Let C' denote a fixed radius cycle related to the variable radius circle C through a tangency constraint. We define the τ -map of C' , denoted by $\tau C'$, as the cylinder in \mathbb{R}^3 , whose intersection with the XY plane is C' and whose axis is parallel to the Z axis. We will also call to $\tau C'$ a τ -cylinder.

Let $L(a, b, c)$ be a ray related to the variable radius circle C through a tangency constraint. We define the τ -map of L , denoted by τL , as the plane in \mathbb{R}^3 which intersects the XY plane at L and at an angle with $[b, -a]$ equal to $\pi/2$. We will also call to τL a τ -plane.

5.3 Algorithm for the Translational Merge Problem

To illustrate the problem we solve, consider the exemple in Figure 9 where d_i denotes a distance constraint, t denotes a tangency and an arc between two straight lines denotes an angle constraint. The set of constraints defines two clusters which can be determined up to position and orientation in the plane. See Figure 10. When these two clusters are combined through the common straight line L_0 , there is still one degree of freedom: a translation along L_0 . Since the variable radius circle has three degrees of freedom, four constraints are needed to combine the clusters and circle into a rigid object. In the exemple the circle is attached to each cluster through two tangency constraints.

The rational for the solution is the following. We consider two clusters, $S_1 = \{L_0, L_1, L_2\}$ and $S_2 = \{L_0, L_3, L_4\}$ sharing the line L_0 and located in the XY plane. If S_1 is fixed at a given position in the plane, cluster S_2 can be translated along L_0 and positioned such that rays L_2, L_3 and L_4 are tangent, with the correct orientation, to a common circle C whose center is

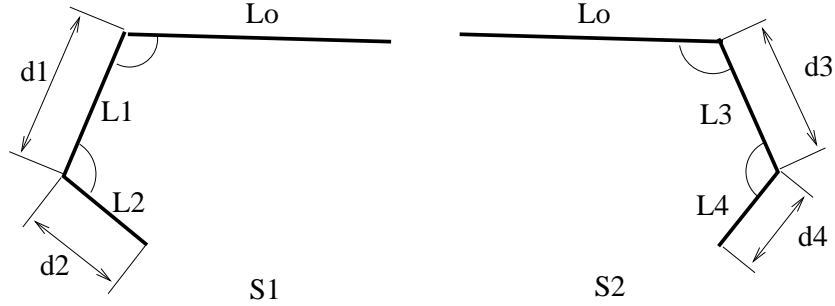


Figure 10: Translational Exemple. Clusters with a translational degree of freedom along L_0 .

on a plane which is perpendicular to the XY plane and that intersects it in a line parallel to the ray L_1 .

Note that, when the variable radius circle C is properly placed, its γ -cyclographic map, γC , is tangent to the γ -cyclographic maps of those geometric elements related to C by constraints on its circumference. Moreover, the apex of γC lies on the τ -cyclographic maps of those geometric elements related to C by constraints on its center point. Therefore, the apex of γC is the intersection point of the γ - and τ -cyclographic maps.

To express our algorithms we shall make use of the following concepts. Geometric elements in the clusters are denoted generically by E . $T(E, d)$ denotes the operation that translates E along the X axis by a distance d and $E(d)$ denotes the translated element. Similarly, $R(E, \theta)$ denotes the operation that rotates E an angle θ around the center of the circle shared by the two clusters and $E(\theta)$ denotes the rotated element.

The Boolean function $circum(E, C)$ returns true whenever there is a constraint between E and the circumference of the variable radius circle C . We will call these constraints *circumference constraints*. The Boolean function $center(E, C)$ returns true whenever there is a constraint between E and the center of the variable radius circle C . We will call these constraints *center constraints*.

Functions $\gamma(E)$ and $\tau(E)$ return, respectively, the γ -cyclographic and the τ -cyclographic maps of the geometric element E .

The general algorithm for clusters with a translational degree of freedom is as follows.

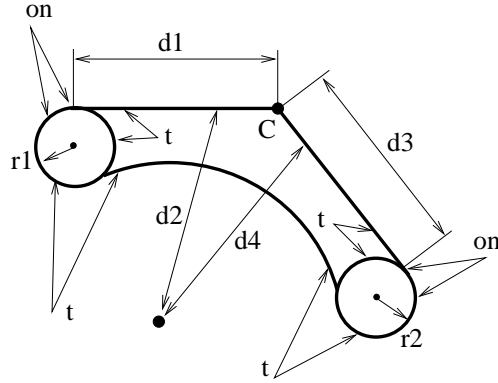


Figure 11: Rotational Example.

Algorithm *TranslationalMerge*

1. Select $S_f, S_t \in \{S_1, S_2\}$ such that $S_f \neq S_t$ with S_f the cluster considered fixed and S_t the cluster to be translated.
2. Place the coordinate system in such a way that line L_0 , the geometric element common to S_1 and S_2 , defines the X axis.
3. For $i \in [1..4]$ do
 - 3.1 If $E_i \in S_t$ then $E_i(d) = T(E_i, d)$ endif
 - 3.2 If $circum(E_i, C)$ then
 - 3.3 $M_i := \gamma(E_i(d))$
 - 3.4 else if $center(E_i, C)$ then
 - 3.4 $M_i := \tau(E_i(d))$
 - endif
- endif
4. Compute the displacement d to be applied to S_t so that $M_i, 1 \leq i \leq 4$, intersect at a common point, $P(x, y, z)$.
5. If $P(x, y, z)$ is a finite point then
 - The variable radius circle seeked has (x, y) as center point and $r = z$ as radius.
- endif

5.4 Algorithm for the Rotational Merge Problem

Consider now the example in Figure 11 where constraints are denoted as before. The set of constraints defines two clusters, S_1 and S_2 which can be determined up to position and orientation in the plane. See Figure 12.

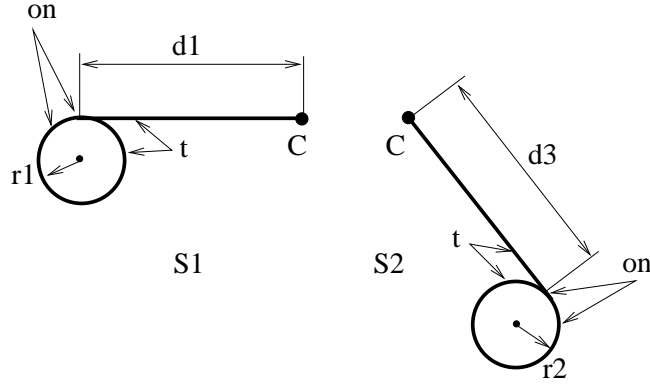


Figure 12: Rotational Exemple. Clusters with a rotational degree of freedom with respect to C .

Now clusters S_1 and S_2 share as a common element the point C . Recall that points are circles with radius zero. When these two clusters are combined through the common point C , there is still one degree of freedom: a rotation about C . In the example, the four constraints needed to combine the clusters and the variable radius circle into a rigid object are two tangencies placed on the circle circumference and two distances placed on the center point.

If S_1 is fixed, S_2 can move relatively to S_2 by a rotation about the point (in general, a rotation about the center of the shared circle). In these conditions, the rational given for the translational clusters applies here if translation $T(E, d)$ is replaced with rotation of an angle θ with respect to the center of the shared circle, $R(E, \theta)$.

If C is the variable radius circle, the general algorithm for clusters with a rotational degree of freedom is as follows.

Algorithm *RotationalMerge*

1. Select $S_f, S_r \in \{S_1, S_2\}$ such that $S_f \neq S_r$ with S_f the cluster considered fixed and S_r the cluster to be rotated.
2. Place the coordinate system in such a way that the origin is the center point of the fixed radius circle shared by S_1 and S_2 .
3. For $i \in [1..4]$ do
 - 3.1 If $E_i \in S_r$ then $E_i(\theta) = R(E_i, \theta)$ endif
 - 3.2 If $circum(E_i, C)$ then
 - 3.3 $M_i := \gamma(E_i(\theta))$

```

    else if  $center(E_i, C)$  then
3.4       $M_i := \tau E_i((\theta))$ 
    endif
  endfor
4.  Compute the rotation  $\theta$  to be applied to  $S_r$  so that
     $M_i, 1 \leq i \leq 4$ , intersect at a common point,  $P(x, y, z)$ .
5.  If  $P(x, y, z)$  is a finite point then
    The variable radius circle seeked has  $(x, y)$  as center point
    and  $r = z$  as radius.
  endif

```

6 Implementation Strategies

Algorithms given in Section 5 are conceptual and an straightforward implementation would result in the need to solve polynomials with degree higher than strictly required. Next we present solving strategies to keep to a minimum the degree of the polynomials involved.

6.1 Notation

If E is either a ray, L , or a cycle, C , we will denote the translational and the rotational merge problems by $E_0(E_1, E_2, E_3, E_4)$, where $S_1 = \{E_0, E_1, E_2\}$ is the cluster considered fixed, $S_2 = \{E_0, E_3, E_4\}$ is the cluster to be moved and E_0 is the geometric element shared by S_1 and S_2 . If $E_0 = L$, the problem is a translational merge while if $E_0 = C$, the problem is a rotational merge.

All of the constraints for the merge problem considered by Hoffmann and Chiang in [6, 7] are tangencies on the circumference of the variable radius circles, that is circumference constraints. According to the type of geometric elements involved, Hoffmann and Chiang in [6, 7] classify the translational merge problem into six classes. Similarly, rotational merge problems are classified into six classes. These classes are listed in the first column of Table 1. Translational and rotational cases are individualized by replacing $E()$ with $L()$ and $C()$, respectively.

Here we consider also constraints defined on the center of the variable radius circles, that is, center constraints. Since a point has two degrees of freedom, in wellconstrained problems, there are at most two constraints on the center of the variable radius circle to be determined. In what follows,

| Problem | 2 constraints | 1 constraint |
|-----------------------|--|---|
| E1: $E(LL,LL)(1,4)$ | E11: $E(LL,L'L')(1,2)$ E12: $E(LL',LL')(1,2)$ | E13: $E(LL,LL')(1,2)$ |
| E2: $E(CL,LL)(2,8)$ | E21: $E(CL,L'L')(2,4)$ E22: $E(CL',LL')(2,4)$ E23: $E(C'L,LL')(2,4)$ E24: $E(C'L',LL)(2,4)$ | E25: $E(CL,LL')(2,4)$ E26: $E(CL',LL)(2,4)$ E27: $E(C'L,LL)(2,4)$ |
| E3: $E(CL,CL)(4,16)$ | E31: $E(CL,C'L')(4,8)$ E32: $E(CL',CL')(4,4)$ E33: $E(C'L,CL')(4,16)$ E34: $E(C'L',C'L)(4,4)$ | E35: $E(CL,CL')(4,4)$ E36: $E(CL,C'L)(4,16)$ |
| E4: $E(CC,LL)(4,16)$ | E41: $E(CC,L'L')(2,4)$ E42: $E(CC',LL')(4,16)$ E43: $E(C'C',LL)(2,4)$ | E44: $E(CC,LL')(2,4)$ E45: $E(CC',LL)(4,16)$ |
| E5: $E(CC,CL)(8,32)$ | E51: $E(CC,C'L')(4,8)$ E52: $E(CC',CL')(8,16)$ E53: $E(CC',C'L)(8,16)$ E54: $E(C'C',CL)(4,8)$ | E55: $E(CC,CL')(4,4)$ E56: $E(CC,C'L)(4,16)$ E57: $E(CC',CL)(8,16)$ |
| E6: $E(CC,CC)(16,64)$ | E61: $E(CC,C'C')(4,8)$ E62: $E(CC',CC')(16,64)$ | E63: $E(CC,CC')(8,16)$ |

Table 1: General merge problem classification.

we will denote by L' and C' the geometric elements in a merge problem on which center constraints have been defined. For example $C(LL',C'L)$ will denote that one line in the fixed cluster and the circle in the moving cluster support center constraints with respect to the variable radius circle.

A general classification for the translational and rotational merge problem is shown in Table 1. The first column indicates the problems with circumference constraints only, the second and third columns show the different problems with two and one center constraints respectively. The pair (m, n) after each problem indicates the maximum number of solutions, m for translational and n for rotational problems. We will explain later on how these figures have been derived.

6.2 Degree of the Equations

To simplify the problem, we assume that in the translational case the straight line shared by the clusters is coincident with the X-axis and that in the rotational problem the cycle shared by the clusters is centered at the origin.

Using the notation already introduced in Section 5 and with simple geometry, we have the following theorem

Theorem 6.1 The equations for γL , τL , $\gamma L(d)$, and $\tau L(d)$ are degree one. The equations for γC , τC , $\gamma C(d)$, $\tau C(d)$, $\gamma L(\theta)$, and $\tau L(\theta)$ are degree two. And, the equations for $\gamma C(\theta)$ and $\tau C(\theta)$ are degree 4 equations.

Notice that the equations for γL , τL , γC and τC have 3 variables x, y, z , and the equations for $\gamma L(d)$, $\tau L(d)$, $\gamma C(d)$, $\tau C(d)$, $\gamma L(\theta)$, $\tau L(\theta)$, $\gamma C(\theta)$ and $\tau C(\theta)$ add one more variable, d .

As mentioned in Section 5, the variable radius circle can be found by intersecting four surfaces, each surface being a γ -map or a τ -map, depending on whether the constraint on the variable radius cycle is a circumference constraint or a center constraint. The system of equations can be easily generated by generating the equations of the maps associated to each geometric element involved. For example, the solution to the problem $L(CL', C'L)$ can be figured out by finding the intersection

$$\gamma C_1 \cap \tau L'_2 \cap \tau C'_3(d) \cap \gamma L_4(d)$$

Since equations $\tau L'_2$, and $\gamma L_4(d)$ are degree one and γC_1 and $\tau C'_3(d)$ are degree two equations, from Bezout's theorem, [4], we know there are 4 solutions for this system. By using Bezout's theorem, every subproblem Eij of problem Ei , see Table 1, has the same number of solutions. For example, every subproblem of $L1$, including $L11$, $L12$ and $L13$, has only one solution, and every subproblem of $C1$, including $C11$, $C12$ and $C13$, has four solutions. The number of solutions for each problem is summarized in Table 1. In the following section, we will derive strategies for each merge problem subclass to reduce these numbers to compute a variable radius circle.

Notice that the number of solutions given is based on our approach by using the γ cyclographic maps and τ maps. If we do not use these maps, the number of solutions must be multiplied by 8, the number of essentially distinct orientations of lines and cycles. For example, the number of solutions to the problems $L(LL, LL)$ and $C(LL, LL)$ would be 8 and 32, respectively.

6.3 Algorithms for the Translational Merge Problem

The subproblems given in Table 1 are individualized as translational merge problems by replacing Eij with Tij . To derive strategies to reduce the degree of the equations to be solved, we group the translational merge problems into three different classes as follows:

1. Problems with two center constraints defined in the same cluster. This includes T11, T21, T24, T31, T41, T43, T51, T54 and T61.
2. Problems that can be transformed into the intersection of three planes and one surface. This includes T12, T13, T22, T23, T25, T26, T27, T32, T34, T35, T44, T55, and part of the problems in the previous class.
3. Problems that can be transformed into the intersection of one γC , one τC , and two planes. This includes T33, T36, T42, T45, T52, T53, T56, T57, T62 and T63.

In the following three subsections we present, from easy to hard, the solution of the problems listed above.

6.3.1 Two center constraints in the same cluster

This problem can be solved by forcing the cluster with two center constraints to be the moving cluster, so the problem becomes $L(E_1E_2, E'_3E'_4)$. Notice that if the moving cluster has two center constraints the cluster fixed has no center constraints. From the fact that the cone-cone intersection can be transformed into cone-plane intersection, [6, 7], the intersection of the geometric elements in the first cluster can always be represented by the intersection of the map of the first element E_1 with a fixed plane. The algorithm to solve the problem is:

Algorithm $L(E_1E_2, E'_3E'_4)$

1. Find the point $(x, y) = \tau E'_3 \cap \tau E'_4$.
2. Substitute the point $(x + d, y)$ into the equation of the plane generated in the first cluster and find $z(d)$ which is a degree one equation of variable d .
3. Substitute the point $(x + d, y, z(d))$ into the equation of E_1 to yield one equation in d .

4. Solve the equation to find the value of d .
5. The variable radius circle seeked has $(x + d, y)$ as center point and $z(d)$ as radius.

The following three theorems give the tools needed to figure out the intersection of $\tau E'_3$ and $\tau E'_4$. The first theorem is trivial and applies when both E'_3 and E'_4 are straight lines.

Theorem 6.2 Two different straight lines $L_i = [a_i, b_i, e_i]$ and $L_2 = [a_j, b_j, e_j]$ which are not parallel, intersect at point

$$(-e_i b_j + e_j b_i, -a_i e_j + a_j e_i, a_i b_j - a_j b_i)$$

The second theorem reduces the line-circle intersection to two line-line intersections.

Theorem 6.3 Let $L = [a, b, e]$ be a straight line and $C = (x, y, r)$ a circle. Intersecting L and C is equivalent to intersecting L and $L' = [a', b', e']$ where

$$\begin{aligned} a' &= a(2t) + b(1 - t^2) \\ b' &= -a(1 - t^2) + b(2t) \\ e' &= -M(1 - t^2) - N(2t) \end{aligned}$$

and

$$\begin{aligned} t &= \pm \sqrt{-(ax + by + e - r)/(ax + by + e + r)} \\ M &= bx - ay \\ N &= ax + by \end{aligned}$$

Furthermore, the intersection points are $(A_x + C_x t, A_y + C_y t, 1)$ where

$$\begin{aligned} A_x &= bM - ae, & C_x &= b\sqrt{r^2 - (N + e)^2} \\ A_y &= -aM - be, & C_y &= -a\sqrt{r^2 - (N + e)^2} \end{aligned}$$

Notice that t is real if and only if $-r \leq ax + by + d \leq r$ and that each different value for t represents a different line.

Notice further that when intersecting a circle C with a line $L(d) = [a, b, e - ad]$, the above theorem applies just replacing e by $e - ad$ or replacing

x by $x - d$. In this case, the relation between t and d becomes $ad(1 - t^2) = (e - r)(1 - t^2) + N(1 + t^2)$.

The third theorem transforms the intersection of two circles into a line-circle intersection.

Theorem 6.4 Let $C_i = (x_i, y_i, r_i)$ and $C_j = (x_j, y_j, r_j)$ be two circles. Intersecting C_i and C_j is equivalent to intersect either circle C_i or C_j with the line $L = [a, b, e]$ where

$$\begin{aligned} a &= x_j - x_i \\ b &= y_j - y_i \\ e &= \frac{1}{2}(x_i^2 + y_i^2 - r_i^2 - x_j^2 - y_j^2 + r_j^2) \end{aligned}$$

The number of solutions in each case can be easily calculated. Two lines intersect at most in one point and a line and a circle or two circles intersect at most in two points. If the cluster fixed has two cycles, we can transform the intersection of their cyclographic maps into the intersection of one cone with a plane, which has two equations of degree one. This increases the number of solutions by a factor of 2. Therefore, there is one solution for T11, two solutions for T21, T24, T41, T43, and four solutions for T31, T51, T54 and T61.

6.3.2 Intersection of three planes and one surface

In [6, 7], cone-cone intersections are transformed into cone-plane intersections. Let $\Pi(C_1, C_i)$ denote the plane that contains the intersection curves of the cones γC_1 and γC_i or τC_1 and τC_i . an consider, for example, the subproblem $E(CC, CC)$. First three planes, namely $\Pi(C_1, C_2)$, $\Pi(C_1, C_3)$ and $\Pi(C_1, C_4)$ are intersected. Then this intersection is substituted in the map γC_1 . We apply this approach here to find the intersection of four cones, with the degree of freedom that the third and fourth cones move along the X-axis or rotate about the origin. We use the similar approach to solve the problem in this criterion. Notice that the intersection of τC_1 and τC_2 can be convert into the intersection of τC_1 with a plane τL .

Table 2 summarizes the problems we consider here. The third, fourth and fifth columns give the planes, generated as γ and τ maps, whose common point defines the distance d that the moving cluster must be translated along the X-axis. This common point is then substituted in the equation given in

| Problem | Equation | Π_1 | Π_2 | Π_3 | Degree |
|-----------------|---------------|---------------|--------------------|-----------------|--------|
| T12: L(LL',LL') | γL_1 | $\tau L'_2$ | $\gamma L_3(d)$ | $\tau L_4(d)$ | 1 |
| T13: L(LL,LL') | γL_1 | γL_2 | $\gamma L_3(d)$ | $\tau L_4(d)$ | 1 |
| T22: L(CL',LL') | γC_1 | τL_2 | $\gamma L_3(d)$ | $\tau L_4(d)$ | 2 |
| T23: L(C'L,LL') | $\gamma C'_1$ | γL_2 | $\gamma L_3(d)$ | $\tau L_4(d)$ | 2 |
| T25: L(CL,LL') | γC_1 | γL_2 | $\gamma L_3(d)$ | $\tau L_4(d)$ | 2 |
| T26: L(CL',LL) | γC_1 | τL_2 | $\gamma L_3(d)$ | $\gamma L_4(d)$ | 2 |
| T27: L(C'L,LL) | $\tau C'_1$ | γL_2 | $\gamma L_3(d)$ | $\gamma L_4(d)$ | 2 |
| T32: L(CL',CL') | γC_1 | τL_2 | $P(C_1, C_3(d))$ | $\tau L_4(d)$ | 4 |
| T34: L(C'L,C'L) | $\tau C'_1$ | γL_2 | $P(C'_1, C'_3(d))$ | $\gamma L_4(d)$ | 4 |
| T35: L(CL,CL') | γC_1 | γL_2 | $P(C_1, C_3(d))$ | $\tau L_4(d)$ | 4 |
| T44: L(CC,LL') | γC_1 | $P(C_1, C_2)$ | $\gamma L_3(d)$ | $\tau L_4(d)$ | 2 |
| T55: L(CC,CL') | γC_1 | $P(C_1, C_2)$ | $P(C_1, C_3(d))$ | $\tau L_4(d)$ | 4 |

Table 2: Classification of the three planes and one surface intersection problem.

the second column, yielding an equation on the variable d whose degree is shown in the last column. The specific algorithm to solve the problem can be written as follows.

Algorithm $L(E_1E_2, E_3E_4)$

1. Find the point $(x(d), y(d), z(d), w(d)) = \Pi_1 \cap \Pi_2 \cap \Pi_3$.
2. If $\text{circum}(E_1, C)$ then
 - $M := \gamma E_1$
 - else if $\text{center}(E_1, C)$ then
 - $M := \tau E'_1$
 - endif
3. Generate one equation with one variable d by replacing $(x(d), y(d), z(d), w(d))$ in the equation of M .
4. Solve the system of equations to find variable d .
5. The variable radius circle seeked has $(x(d)/w(d), y(d)/w(d))$ as center point and $z(d)/w(d)$ as radius.

6.3.3 Intersection of one γ -cylinder one τ -cylinder and two planes

The Table 3 summarizes the problems we consider in this section. These problems are transformed into the intersection of two planes with a cone,

| Problem | γC | $\tau C'$ | Π_1 | Π_2 | Deg 1 | Deg 2 | Degree |
|--------------------|--------------|----------------|--------------------|--------------------|---------|---------|--------|
| T33: $L(CL', C'L)$ | γC_1 | $\tau C'_3(d)$ | τL_2 | $\gamma L_4(d)$ | (1,1,0) | (0,0,0) | 4 |
| T36: $L(CL, C'L)$ | γC_1 | $\tau C'_3(d)$ | γL_2 | $\gamma L_4(d)$ | (1,1,0) | (0,0,0) | 4 |
| T42: $L(CC', LL')$ | γC_1 | $\tau C'_2$ | $\gamma L_3(d)$ | $\tau L_4(d)$ | (1,0,0) | (0,0,0) | 4 |
| T45: $L(CC', LL)$ | γC_1 | $\tau C'_2$ | $\gamma L_3(d)$ | $\gamma L_4(d)$ | (1,0,0) | (0,0,0) | 4 |
| T52: $L(CC', CL')$ | γC_1 | $\tau C'_2$ | $P(C_1, C_3(d))$ | $\tau(L_4(d))$ | (2,2,0) | (0,0,1) | 8 |
| T53: $L(CC', C'L)$ | γC_1 | $\tau C'_2$ | $P(C'_2, C'_3(d))$ | $\gamma L_4(d)$ | (2,2,0) | (0,1,1) | 8 |
| T56: $L(CC, C'L)$ | γC_1 | $\tau C'_3(d)$ | $P(C_1, C_2)$ | $\gamma L_4(d)$ | (1,1,0) | (0,0,0) | 4 |
| T57: $L(CC', CL)$ | γC_1 | $\tau C'_2$ | $P(C_1, C_3(d))$ | $\tau L_4(d)$ | (2,2,0) | (0,1,1) | 8 |
| T62: $L(CC', CC')$ | γC_1 | $\tau C'_2$ | $P(C_1, C_3(d))$ | $P(C'_2, C'_4(d))$ | (2,3,0) | (0,1,1) | 18(16) |
| T63: $L(CC', CC)$ | γC_1 | $\tau C'_2$ | $P(C_1, C_3(d))$ | $P(C_3, C_4)(d)$ | (2,2,0) | (0,1,1) | 8 |

Table 3: Classification of the γC , $\tau C'$ and two planes intersection problem.

γC , and a cylinder, $\tau C'$. The planes Π_1 and Π_2 , are given in the fourth and fifth columns, the second column lists the γ map and the third column the τ map.

Note that when the moving cluster is translated along the X-axis, the line common to planes Π_1 and Π_2 defines the distance d . Assume that

$$\Pi_1 = (a_1(d), b_1(d), c_1(d), w_1(d))$$

and

$$\Pi_2 = (a_2(d), b_2(d), c_2(d), w_2(d))$$

are two planes in Table 3, the parametric form of the line where Π_1 and Π_2 intersect is

$$L = (a_1(d), b_1(d), c_1(d), w_1(d)) + s (a_2(d), b_2(d), c_2(d), w_2(d))$$

The degrees on the variable d of each component in $(a_1(d), b_1(d), c_1(d))$ and $(a_2(d), b_2(d), c_2(d))$ are shown in the sixth and seventh column in Table 3. The algorithm for finding the solution to this class of problems is the following

Algorithm $L(CE_2, E_3E_4)$

1. According to the subproblem at hand and following Table 3, generate γC , $\tau C'$, Π_1 and Π_2
2. Find the parametric form of the line $L = \Pi_1 \cap \Pi_2$.
3. Generate two equations in d and s by substituting the explicit form of L into the implicit forms of γC and $\tau C'$.

4. Figure out d by solving the system of two equations derived in step 3.
5. Point $(a1(d), b1(d)) + s(a2(d), b2(d))$ is the center of the variable radius circle and $c1(d) + s * c2(d)$ is the radius.

Applying Bezout's theorem to the equations derived in step 3 in the Algorithm $L(CE_2, E_3E_4)$, we find the maximum number of solutions to the problem, which is given in the eighth column of Table 3. Notice that compared to the system of equations generated in Section 6.2, this approach yields 2 more solutions for the problem $L(CC', CC')$.

6.4 Algorithms for the Rotational Merge Problem

The classification for rotational clusters is the same as in the translational case. All that is needed is to replace Eij with Rij in Table 1 to indicate that the problem has the same constraint pattern but the moving cluster is now rotated. We also separate these problems into three different classes. They are:

1. Problems with two center constraints defined in the same cluster. This includes R11, R21, R24, R31, R41, R43, R51, R54 and R61.
2. Problems which can be transformed into the intersection of three planes and one surface. This includes R12, R13, R22, R23, R25, R26, R27, R32, R34, R35, R44, R55, and part of the problems in the previous class.
3. The remainder of the problems, including R33, R36, R42, R45, R52, R53, R56, R57, R62, R63.

6.4.1 Two center constraints in the same cluster

If the cluster with two center constraints is the one that will be rotated, the problems in this class are formally solved in the same way as $L(E_1E_2, E'_3E'_4)$. The intersection of the geometric elements in the cluster fixed can always be represented by the intersection of the map of the first element E_1 with a fixed plane. If θ is the rotation angle and $t = \tan(\theta/2)$ the algorithm to solve the problem is

Algorithm $C(E_1E_2, E'_3E'_4)$

1. Find the point $(x, y) = \tau E'_3 \cap \tau E'_4$.
2. Rotate the point (x, y) by an angle θ yielding $(x(t), y(t))$.
3. Replace $(x(t), y(t))$ into the equation of the plane generated in the first cluster and find $z(t)$ which is a degree one equation in t .
4. Substitute $(x(t), y(t), z(t))$ into the equation of E_1 to yield one equation in t .
5. Solve the equation to find the value of t .
6. The seeked circle is centered on $(x(t), y(t))$ and $z(t)$ is the radius.

6.4.2 Intersection of three planes and one surface

Problems in this class, $C(E_1E_2, E_3E_4)$, are solved as problems in the corresponding translational class $L(E_1E_2, E_3E_4)$ replacing the translation with the rotation α If $t = \tan(\theta/2)$, the algorithm is

Algorithm $C(E_1E_2, E_3E_4)$

1. Find the point $(x(t), y(t), z(t), w(t)) = \Pi_1 \cap \Pi_2 \cap \Pi_3$.
2. If $\text{circum}(E_1, C)$ then

$$M := \gamma E_1$$
 else if $\text{center}(E_1, C)$ then

$$M := \tau E'_1$$
 endif
3. Generate one equation with one variable t by replacing $(x(t), y(t), z(t), w(t))$ in the equation of M .
4. Solve the system of equations to find variable t .
5. The variable radius circle seeked has $(x(t)/w(t), y(t)/w(t))$ as center point and $z(t)/w(t)$ as radius.

To justify the degree reduction, we need the following result.

Theorem 6.5 Consider three planes with constant coefficients except for t :

$$\begin{aligned} \Pi_1 &= [a_2, b_2, c_2, -d_2] \\ \Pi_2 &= [a_3(1-t^2) - b_3(2t) + e_3(1+t^2), a_3(2t) + b_3(1-t^2) + f_3(1+t^2), \\ &\quad c_3(1+t^2), -d_3(1+t^2)] \\ \Pi_3 &= [a_4(1-t^2) - b_4(2t) + e_4(1+t^2), a_4(2t) + b_4(1-t^2) + f_4(1+t^2), \\ &\quad c_4(1+t^2), -d_4(1+t^2)] \end{aligned}$$

Then Π_1, Π_2 and Π_3 intersect in a point $(x(t), y(t), z(t), w(t))$ whose components are expressions of degree 2 in t .

Now it is easy to show that problems in this class have two solutions when the first component E_1 is a ray, such as in problems C12 and C13. Otherwise, they have 4 solutions, such as in problems C22, C23, C25, C26, C27, C32, C34, C35, C44 and C55. The number of solutions is given in the Table 1.

6.4.3 Remainder Problems

The remainder subproblems of rotational merge problems are C33, C36, C42, C45, C52, C53, C56, C57, C62 and C63.

Unfortunately, the strategy presented in Section 6.3 does not work here because the degree of the parametric line generated at an intermediate step does not allow to reduce the final degree of γC and $\tau C'$.

We need to derive some results before presenting our algorithm. Let C be a cycle centered at (x_1, y_1) with (signed) radius z_1 and let L and L' be rays with equations $a_2x + b_2y + d_2 = 0$, and $a_2^2 + b_2^2 = 1$ respectively. Let $\delta = (a_2, b_2, d_2) \cdot (x_1, y_1, 1) = (a_2x_1 + b_2y_1 + d_2)$.

Theorem 6.6 The intersection points for γC and γL are given by

$$\begin{cases} x(u) &= x_1 + z_1 \frac{1-u^2}{1+u^2} - \frac{\delta}{w(u)}(1-u^2) \\ y(u) &= y_1 + z_1 \frac{2u}{1+u^2} - \frac{\delta}{w(u)}(2u) \\ z(u) &= 2z_1 - \frac{\delta}{w(u)}(1+u^2) \end{cases}$$

Notice that when $z_1 = 0$, the locus of the intersection points in the above theorem can be written in homogeneous form as:

$$\begin{cases} x(u) &= x_1w(u) - \delta(1-u^2) \\ y(u) &= y_1w(u) - \delta(2u) \\ z(u) &= \delta(1+u^2) \\ w(u) &= (a_2, b_2, -1) \cdot (1-u^2, 2u, 1+u^2) \end{cases}$$

Theorem 6.7 The intersection points for γC and $\tau L'$ are

$$\begin{cases} x(u) &= x_1w(u) - \delta(1-u^2) \\ y(u) &= y_1w(u) - \delta(2u) \\ z(u) &= z_1w(u) - \delta(1+u^2) \\ w(u) &= (a_2, b_2, 0) \cdot (1-u^2, 2u, 1+u^2) \end{cases}$$

Let C' be a cycle centered at (x_1, y_1) with (signed) radius z_1 such that supports a center constraint.

Theorem 6.8 The intersection points of $\tau C'$ and γL are given by

$$\begin{cases} x(u) &= x_1(1 + u^2) + z_1(1 - u^2) \\ y(u) &= y_1(1 + u^2) - z_1(2u) \\ z(u) &= \delta(1 + u^2) + z_1((a_2, b_2) \cdot (1 - u^2, 2u)) \\ w(u) &= 1 + u^2 \end{cases}$$

Notice that the intersection points in the above three theorems when $z_1 = 0$ are all degree 2 equations for $(x(u), y(u), z(u), w(u))$

Consider the problem $E(C_1E_2, E_3E_4)$. It is advantageous to "lift" the plane in which we solve the problem in the Z -direction by a distance equal to the (signed) radius of the first cycle C_1 . This has the effect of reducing C_1 to a point simplifying the cone equation γC_1 . The solution can then be dropped back down, to the original problem plane, by shifting the line, re-inflating the cycle, and increasing or diminishing the variable radius cycle. The details are routine. Notice that in this approach, the above three theorems for cases $\gamma C \cap \gamma L$, $\gamma C \cap \tau L'$ and $\tau C' \cap \tau L'$ can be all represented in homogenous coordinates with degree 2 in its components.

Many problems have the patterns $\gamma C \cap \gamma L$, $\gamma C \cap \tau L'$ and $\tau C' \cap \tau L'$ in the same cluster, including $C33, C36, C42, C45, C52, C53, C56, C57$ and $C63$. We assume they are the fixed cluster while the two elements in the second cluster are E_3 and E_4 . Without loss of generality, we assume that if the first element is a cycle which has a circumference constraint with the variable radius cycle, its cyclographic map γC_1 has the apex located in the XY plane. That is, the apex of γC_1 has 0 as its z -component value. With these assumptions, the algorithm for these problems is:

Algorithm $C33, C36, C42, C45, C52, C53, C56, C57, C63$

1. Find the intersection point $(x(u), y(u), z(u), w(u))$ for the geometric elements in the cluster fixed.
2. Find the equation for the geometric element in the moving cluster after its rotation. Intersection is given by two degree 4 equations, $f(x, y, z, t) = 0$ and $g(x, y, z, t) = 0$.
3. Substitute point $(x(u), y(u), z(u), w(u))$ into f and g to find two equations with two variables t and u .
4. Solve these two equations

$$F(u, t) = f(x(u)/w(u), y(u)/w(u), z(u)/w(u), t) = 0$$

$G(u, t) = g(x(u)/w(u), y(u)/w(u), z(u)/w(u), t) = 0$
to figure out u and t .

5. Point $(x(u)/w(u), y(u)/w(u))$ is the center of the circle and the radius is $(z(u)/w(u))$.

We need to prove that functions F and G in the above algorithm are degree 4. We use the subproblem $C33 = C(CL', C'L)$ as an example, the others are routine. Applying Theorem 6.7, we can find the parametric form for the intersection of the elements in the fixed cluster. Then we substitute this parametric form into the implicit form of F and G , which are $\tau C'_3(t)$ and $\gamma L_4(t)$ respectively, to get the following two degree 4 equations with two variables:

$$\begin{aligned} F(u, t) = & (x_1^2 + y_1^2 - z_1^2 + x_3^2 + y_3^2 - z_3^2)w(u)(1 + t^2) \\ & - 2\delta(1 + t^2)((x_1, y_2, -z_1) \cdot (1 - u^2, 2u, 1 + u^2)) \\ & - 2(x_3(1 - t^2) - y_3(2t))(x_1w(u) - \delta(1 - u^2)) \\ & - 2z_3(1 + t^2)(z_1w(u) - \delta(1 + u^2)) = 0 \end{aligned}$$

$$\begin{aligned} G(u, t) = & (a_4(1 - t^2) + b_4(2t))(x_1w(u) - \delta(1 - u^2)) \\ & + (a_4(2t) + b_4(1 - t^2))(y_1w(u) - \delta(2u)) \\ & + (d_4w(u) - z_1w(u) + \delta(1 + u^2))(1 + t^2) = 0 \end{aligned}$$

where $\delta = (a_2, b_2, d_2) \cdot (x_1, y_1, 1)$ and $w(u) = (a_2, b_2, 0) \cdot (1 - u^2, 2u, 1 + u^2)$. Therefore the number of solutions in this class of problem is 16.

A similar analysis applies to the subproblems $C36, C42, C45, C52, C53, C56, C57$ and $C63$.

We did not reduce the number of solutions for the case $C62 = C(CC', CC')$.

7 Conclusion

After adding the center constraints into the variable radius circle clusters, we classify the problem into three classes:

1. Two center constraints are on the same cluster.
2. The problem can be modify into the intersection of three planes with either a γ -map or a τ -map.

3. Neither one of above.

The solution strategy for solving variable radius circle clusters has different pattern for the problems in each class.

For problems in the first class, we transform the system of equations for the fixed cluster into the equation of a cone or cylinder plus the equation of a fixed plane. If n is the number of solutions for the intersection of two geometric entities supporting a center constraint, the maximum number of solutions for translational merge problem is $2n$ if the fixed cluster has a cycle, or n if the fixed cluster are all planes. As for the rotational merge problem, the number of solution are $4n$ if the fixed cluster has a cycle, or $2n$ if the fixed cluster only has planes.

Problems in the second class are transformed into the intersection of three planes with either a plane, a cylinder or cone. The number of solutions in this class is reduced by the fact that the intersection point of one fixed plane with two rotated planes with the same degree is a degree 2 equation.

In the third class and with the aim of reducing as much as possible the maximum number of possible solutions to the problems in the third class, we apply different approaches to the translational and rotational merge cases. In the translational merge case, we transform the problem into the intersection of two planes from the fixed cluster and a cone and a cylinder from the moving cluster. The number of solutions depends on the degree of a parametric line generates in an intermediate step of the computations. In the rotational merge problem no matter which are the maps of the geometric entities in the moving cluster (plane, cone or cylinder), we always get two degree 4 equations with two variables. Therefore the maximum number of solution for the problems in this class is 16.

Except for the subproblem $C62 = C(CC', CC')$ for which we do not reduce the number of solutions, the degree reduction achieved by the specific algorithms with respect to the conceptual algorithms is significative.

Acknowledgements

C.-S. Chiang has been partially supported by NSC in Taiwan under Grant 39201F. R. Joan-Arinyo has been partially supported by CICYT under the project TIC2001-2099-C03-01.

References

- [1] D-Cubed. *The Dimensional Constraint Manager*. Cambridge, England, 1994. Version 2.7.
- [2] C. Durand. *Symbolic and Numerical Techniques for Constraint Solving*. PhD thesis, Computer Science, Purdue University, December 1998.
- [3] I. Fudos and C.M. Hoffmann. A graph-constructive approach to solving systems of geometric constraints. *ACM Transactions on Graphics*, 16(2):179–216, April 1997.
- [4] R. Hartshorne. *Algebraic Geometry*. Graduate Texts in Mathematics. Springer-Verlag, 1977.
- [5] C. M. Hoffmann. Computer vision, descriptive geometry, and classical mechanics. In B. Falcidieno and I. Herman, editors, *Proc. Eurographics Workshop on Computer Graphics and Mathematics*, Eurographics Series. Springer Verlag, 1991.
- [6] C.M. Hoffmann and C.-H. Chiang. Variable-radius circles of cluster merging in geometric constraints. Part I: Translational clusters. *Computer Aided Design*, 2002. In press.
- [7] C.M. Hoffmann and C.-H. Chiang. Variable-radius circles of cluster merging in geometric constraints. Part II: Rotational clusters. *Computer Aided Design*, 2002. In press.
- [8] C.M. Hoffmann and R. Joan-Arinyo. Symbolic constraints in constructive geometric constraint solving. *Journal of Symbolic Computation*, 23:287–300, 1997.
- [9] C.M. Hoffmann and P.J. Vermeer. A spatial constraint problem. In J.P. Merlet and B. Ravani, editors, *Computational Kinematics'95*, pages 83–92. Kluwer Academic Publ., 1995.
- [10] R. Joan-Arinyo and A. Soto-Riera. Combining constructive and equational geometric constraint solving techniques. *ACM Transactions on Graphics*, 18(1):35–55, January 1999.
- [11] J.C. Owen. Algebraic solution for geometry from dimensional constraints. In R. Rossignac and J. Turner, editors, *Symposium on Solid Modeling Foundations and CAD/CAM Applications*, pages 397–407, Austin, TX, June 5-7 1991. ACM Press.

- [12] J.C. Owen. Constraints on simple geometry in two and three dimensions. In *3rd Conference on Geometric Design*. SIAM, November 1993.
- [13] H. Pottmann and M. Peternell. Applications of Laguerre geometry in CAGD. *Computer Aided Geometric Design*, 15(2):165–188, February 1998.
- [14] K. Ramanathan. Variable radius circle computations in geometric constraint solving. Master’s thesis, Department of Computer Science. Purdue University, 1996.

8 Appendix

In this section we collect some results concerning properties used in previous sections.

Theorem 8.1 Let $L = [a, b, c]$ be a straight line. For every point (x, y) on the line the relation $c = -(a, b)(x, y)$ holds.

Proposition 8.2 The line or ray from point $P_1 = (x_1, y_1)$ to point $P_2 = (x_2, y_2)$ is

$$L = [y_1 - y_2, x_2 - x_1, x_1y_2 - x_2y_1].$$

Proposition 8.3 The line or ray $L = [a, b, c]$ when translated along the X-axis by the distance d , becomes the line or ray $L' = [a, b, c - ad]$. Moreover, when rotated counter-clockwise about the origin by the angle θ , it becomes

$$L'' = [a \cos(\theta) - b \sin(\theta), a \sin(\theta) + b \cos(\theta), c]$$

Proposition 8.4 The plane $\Pi = [a, b, c, w]$ when translated along the X-axis by d , becomes the plane

$$\Pi' = [a, b, c, w - ad].$$

Moreover, when rotated counter-clockwise about the Z-axis by the angle θ , it becomes the plane

$$\Pi'' = [a \cos(\theta) - b \sin(\theta), a \sin(\theta) + b \cos(\theta), c, w]$$

Proposition 8.5 The cycle $C = (x, y, r, 1)$ translated along the X-axis by d , becomes $C' = (x + d, y, r, 1)$. When rotated counter-clockwise about the Z-axis by θ , it becomes

$$C' = (x \cos(\theta) - y \sin(\theta), x \sin(\theta) + y \cos(\theta), r, 1)$$

Proposition 8.6 The plane $\Pi = [a, b, c, w]$ spanned by three points $P_k = (x_k, y_k, z_k, -w_k)$, $k = 1, 2, 3$ is

$$a = \begin{vmatrix} w_1 & y_1 & z_1 \\ w_2 & y_2 & z_2 \\ w_3 & y_3 & z_3 \end{vmatrix} \quad b = \begin{vmatrix} x_1 & w_1 & z_1 \\ x_2 & w_2 & z_2 \\ x_3 & w_3 & z_3 \end{vmatrix}$$

$$c = \begin{vmatrix} x_1 & y_1 & w_1 \\ x_2 & y_2 & w_2 \\ x_3 & y_3 & w_3 \end{vmatrix} \quad w = \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}$$

Proposition 8.7 The intersection point of three planes $\Pi_k = [a_k, b_k, c_k, -w_k]$, $k = 1, 2, 3$ is the point $P = (x, y, z, w)$ where

$$x = \begin{vmatrix} d_1 & b_1 & c_1 \\ d_2 & b_2 & c_2 \\ d_3 & b_3 & c_3 \end{vmatrix} \quad y = \begin{vmatrix} a_1 & d_1 & c_1 \\ a_2 & d_2 & c_2 \\ a_3 & d_3 & c_3 \end{vmatrix}$$

$$z = \begin{vmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \\ a_3 & b_3 & d_3 \end{vmatrix} \quad w = \begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}$$

Proposition 8.8 Let $\Pi_1 = [a_1, b_1, c_1, w_1]$ and $\Pi_2 = [a_2, b_2, c_2, w_2]$ be two planes. Define $q_{mn} = m_1n_2 - m_2n_1$, where $m, n \in \{a, b, c, d\}$. If these two planes intersect, the intersection line has direction $[q_{bc}, q_{ca}, q_{ab}]$. If $q_{bc} \neq 0$, this line passes through the point $(0, -q_{dc}, q_{db}, q_{bc})$. If $q_{ca} \neq 0$, this line passes through the point $(q_{dc}, 0, -q_{da}, q_{ca})$. If $q_{ab} \neq 0$, this line passes through the point $(-q_{db}, q_{da}, 0, q_{ab})$.

Proposition 8.9 A line or ray that has distance d to a point $(x_0, y_0, 1)$ has the coordinates $[a, b, d\sqrt{a^2 + b^2} - (ax_0 + by_0)]$.

Proposition 8.10 The ray that intersects the X-axis at an angle θ , measuring the angle clockwise from the X-axis to the ray, is $[-\sin(\theta), -\cos(\theta), d]$. Furthermore, for every point $(x, y, 1)$ on the ray, $d = x \sin(\theta) + y \cos(\theta)$.

Theorem 8.11 The intersection curve of two normal cones whose apices are $P_0 = (x_0, y_0, z_0, 1)$ and $P_1 = (x_1, y_1, z_1, 1)$ lies in the plane with normal $[x_1 - x_0, y_1 - y_0, z_0 - z_1]_E$ and passing through the point $(P_0 + P_1)/2$. That is, the intersection of these two cones is equal to the intersection of either one of the cones with the plane:

$$\begin{aligned} C((x_0, y_0, z_0, 1)) \cap C((x_1, y_1, z_1, 1)) &= C((x_0, y_0, z_0, 1)) \cap \Pi \\ &= C((x_1, y_1, z_1, 1)) \cap \Pi \end{aligned}$$

where $\Pi = [x_1 - x_0, y_1 - y_0, z_0 - z_1, (x_0^2 + y_0^2 - z_0^2 - x_1^2 - y_1^2 + z_1^2)/2]$.

Theorem 8.12 The cyclographic map for the ray $[a, b, d]$ rotated about the origin by θ has the form

$$[a(1 - t^2) - b(2t), a(2t) + b(1 - t^2), c(1 + t^2), d(1 + t^2)]$$

where $t = \tan(\theta/2)$, $-\pi < \theta < \pi$, and $c = -\sqrt{a^2 + b^2}$. When $\theta = \pi$, the cyclographic map becomes $[-a, -b, c, d]$.

Theorem 8.13 Let $L = [a, b, d]$ be a ray. The maps $\gamma L(\theta)$ and $\tau L(\theta)$ have the form

$$[a(1 - t^2) - b(2t) + e(1 + t^2), a(2t) + b(1 - t^2) + f(1 + t^2), c(1 + t^2), d(1 + t^2)]$$

where $t = \tan(\theta/2)$, $-\pi < \theta < \pi$, $e = f = 0$, and $c = -\sqrt{a^2 + b^2}$, $c = 0$ for the $\gamma L(\theta)$ and $\tau L(\theta)$ respectively. When $\theta = \pi$, the $\gamma L(\theta)$ and $\tau L(\theta)$ becomes $[-a, -b, c, d]$.

Theorem 8.14 Let $C'_1 = (x_1, y_1, z_1)$ and $C'_3 = (x_3, y_3, z_3)$. The intersection plane of $\tau C'_1$ and $\tau C'_3(\theta)$ has the form

$$[a(1 - t^2) - b(2t) + e(1 + t^2), a(2t) + b(1 - t^2) + f(1 + t^2), c(1 + t^2), d(1 + t^2)]$$

where $t = \tan(\theta/2)$, $-\pi < \theta < \pi$, $a = x_3, b = y_3, e = -x_1, f = -y_1, c = 0$, and $d = (x_1^2 + y_1^2 - z_1^2 - x_3^2 - y_3^2 + z_3^2)/2$. When $\theta = \pi$, the cyclographic map becomes $[-a + e, -b + f, 0, d]$.