Assembling Objects Through Parts Correlation

Pamela J. Vermeer

May 29, 1997

Abstract

A solution is presented to the problem of assembling four solid objects by matching points and planes of pairs of the objects. This problem would be simple if the parts could be assembled by fixing one part and adding the others sequentially. In the situation we consider, however, after fixing one part, the remaining three parts must be positioned simultaneously. To do this requires solving an interlocked system of nonlinear equations with up to sixteen different solutions. Since we are interested in obtaining all possible positionings of the four objects which satisfy the given constraints, an analytic approach is taken to solve the resulting system of equations. In addition to the theoretical solution, results of an implementation of the solution are presented.

1 Introduction

When designing a complex mechanism or structure, it is often easiest to construct the object component by component. The simpler individual components can then be pieced together by correlating geometric elements of pairs of objects. In this paper, we present a solution to the problem of assembling four parts by matching points and planes in pairs of parts. This problem would be simple if the parts could be assembled by fixing one part and adding the others sequentially. In the situation we consider, however, after fixing one part, the remaining three parts must be positioned simultaneously. To do this requires solving an interlocked system of nonlinear equations with up to sixteen different solutions. Since we are interested in obtaining all possible positionings of the four objects which satisfy the given constraints, an analytic approach is taken to solve the resulting system of equations.

We first discuss the broader area of geometric constraint solving, within which this problem arises. Next we show how our problem arises in the context of geometric constraint solving and we establish the framework of the required matching, that is, how the four base objects are related to each other. This basic framework leads to a suite of problems which must be solved, each corresponding to a different type of matching between the four parts. We next develop a general solution method for finding all the possible positionings of the four components which satisfy a particular matching. We finish with a discussion of an implementation in progress of a program to allow interactive solving of these constraint problems. We include in an appendix demonstrations of constructions solved by the assembly library component of the program.

2 Background

The problem we study in this paper is an application of the general problem of three-dimensional geometric constraint solving. While two-dimensional constraint solving has been studied extensively, there remain many open questions in the arena of three-dimensional problems. The problem consists of a set of geometric entities, and a description of geometric constraints between the elements. The goal is to find all placements of the geometric entities which satisfy the given constraints.

A constraint solving problem is *well-constrained* if there are a finite number of solutions to the problem, while a problem with an infinite number of solutions is *underconstrained*. A problem is *overconstrained* if one constraint can be deleted yet the constraint system still has a finite number of solutions. An overconstrained problem may have a solution when the additional constraints are consistent with previous constraints, but often overconstrained problems have no solution.

There are two basic strategies for solving the constraint problem. An *instance solver* uses the explicit values of the given constraints to determine the possible geometric configurations which satisfy the constraints. A *generic* *solver* determines whether the given geometric elements can be placed independent of the particular values assigned to the constraints. That is, the constraints have a symbolic rather than numerical value. The geometric elements are placed only after a decision has been made about whether or not the problem is generically well-constrained.

There are a variety of ways to implement each of these two strategies. Numerical constraint solvers function by first translating the constraints into a system of algebraic equations. This system is then solved using an iterative technique such as the Newton-Raphson method. Examples of numerical solvers include Sketchpad [Sut63] and ThingLab [Bor81]. Symbolic constraint solvers also begin by setting up a system of algebraic equations. However, instead of applying numerical techniques to determine a solution, general symbolic computations are undertaken to find the solution to the system of equations. Methods such as Gröbner basis [Bos85] or Wu-Ritt [Wu86] techniques can be applied to find symbolic expressions for the solutions. For example, Kondo uses the symbolic method as the basis for adding and deleting constraints from a given system [Kon92]. The method of logical inference and term rewriting applies general logical reasoning techniques to the geometric problem of constraint solving. This approach has been taken by Aldefeld [Ald88] and Bruderlin [Bru90], among others. Kramer's degree of freedom analysis [Kra92] takes a kinematics view of the problem. Finally, the graph-based constraint method of Hoffmann *et. al.* [BFH+95, FH93] applies a two-step process. First, a graph of the constraints is formed and analyzed. If the graph is well-constrained, then the construction is carried out, using the graph to determine the order of constructions. A more extensive overview of the various approaches to constraint solving can be found in [Fud93, DH95]

3 Problem Framework

The problem considered in this paper arises from the graph-based approach to constraint solving. In the analysis phase of that method, geometric entities (or *geoms*) which can be placed relative to one another are formed into groups called *clusters*. Each geom begins with three degrees of freedom, so placing a new geom requires that it be constrained by three other previously fixed geoms. Thus to begin a cluster, there must be at least one set of

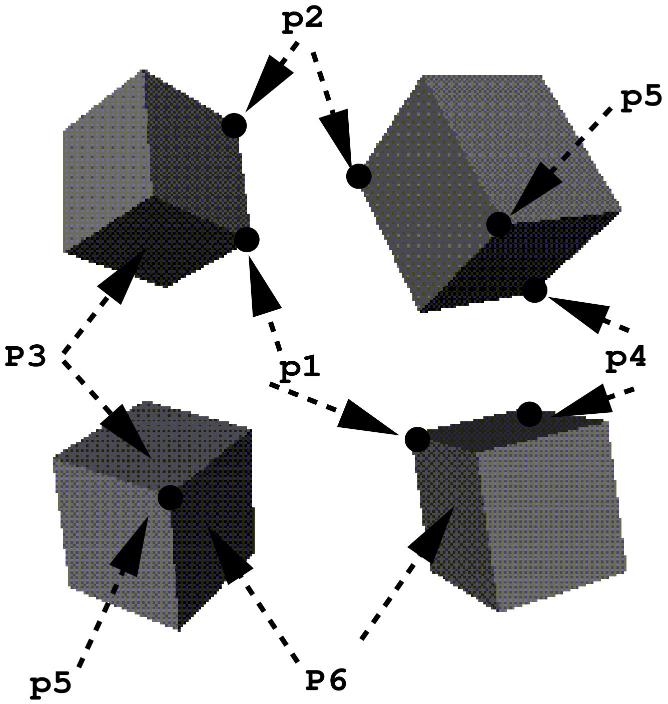

Figure 1: Four boxes to be assembled into a single unit

three pairwise constrained geoms. These three geoms can be positioned in a standard way depending on the types of geoms involved and the constraints between them. A new geom or another cluster can be added to a cluster if there are three constraints between the existing cluster and the new geom or cluster to be added. This process continues until no new clusters can be formed and no additional geoms can be added to any of the existing clusters.

Once the initial clusters have been formed, *cluster merging* takes place. Having multiple constraints may have allowed a geom to be part of more than one cluster. Cluster merging combines clusters which share a geometric entity in such a way that the result is rigid in $\Re^3$, up to translation and rotation as an entire body. Further details about cluster formation and merging in spatial constraint solving can be found in[DH95].

In the problem we are working on, we begin with four objects. In order to assemble these objects into a single object, each object must be matched

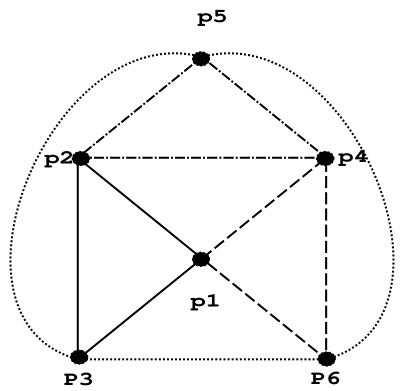

Figure 2: Cluster abstraction of cube assembly problem

with every other object. That is, three different geoms of the first object must be chosen such that one of them is the same as some geom on the second object, another is the same as some geom on the third object, and the third is the same as some geom on the fourth object. Then two more geoms on the second object must be chosen so that the first matches a geom on the third object, and the third selected geom matches a geom on the fourth object. Finally, a geom on the third object must be selected to match a geom on the fourth object. With this framework, the problem becomes one of merging four clusters, where each cluster shares exactly one geom with each other cluster. Each cluster corresponds to the geoms selected on one of the objects.

For example, consider Figure 1. Here each unit cube is one of the objects to be assembled into a larger unit. The cube in the upper left can be reduced to the cluster $\{p_1, p_2, P_3\}$ with constraints $d_{12} = 1$, $d_{13} = 0$, and $d_{23} = -1$ (assuming outward normals on the object surfaces). The cube in the upper right is simplified to the cluster $\{p_2, p_4, p_5\}$ with constraints $d_{24} = \sqrt{3}$, $d_{25} = \sqrt{2}$, and $d_{45} = \sqrt{2}$. The cube in the lower left is represented by the cluster $\{P_3, p_5, P_6\}$ with constraints $d_{35} = 0$, $\alpha_{36} = \pi/2$, and $d_{56} = 0$. Finally, the cube in the lower right can be thought of as the cluster $\{p_1, p_4, P_6\}$ with constraints $d_{14} = 1$, $d_{16} = 0$, and $d_{46} = -1$. Solving this cluster merging problem will give positions for the three geometric entities on each cube. This is enough information to be able to rigidly place the cubes together.

The abstraction of the problem to cluster merging can be visualized as in

Figure 2. Each of the four triangles in the figure with the same line style for edges represents one cluster. The lines indicate a known constraint between the geoms. The clusters are joined at exactly one geometric element; this indicates a matching between geoms in different clusters. Because of the octahedral nature of this abstract representation, we sometimes refer to the problem of matching objects in this way as *the octahedron problem*. In fact, if the geometries are all points and the constraints all positive distances between the points, the final configuration of the cluster geoms is an octahedron.

If one cluster is fixed in space, the other three clusters cannot be added sequentially because there are only two known constraints between the fixed cluster and any unfixed cluster. This means that to obtain a rigid configuration of the four clusters as a single object, the remaining three clusters must be placed simultaneously. In the rest of the paper, we present a solution to the problem of fixing the first cluster and then positioning the other three clusters with respect to that cluster.

Notation

Throughout the paper we use the following notation: g_i refers generically to either a point or a plane, while p_i is a point and P_i is a plane. Any constraint between two points or between a point and a plane is a signed distance d_{ij}, where the sign of the distance between a point and a plane is positive if the point is on the side of the plane towards which the normal points and is negative otherwise. Any constraint between two planes is an angle α_{ij}

4 A General Solution

An initial analysis of the problem in its abstract formulation as six geometric elements, each of which is either a point or a plane, comprising four clusters yields ten distinct cases which must be solved. Each line in Table 1 lists one configuration of four clusters.

The following steps outline the solution process for any of the subcases.

1. Fix the coordinate system using the first cluster. This is done by explicitly positioning the three elements in the cluster. We now consider only the three remaining clusters.

Case	Geometric Elements	Clusters
1	6 points	(p_1, p_2, p_3) (p_1, p_4, p_5) (p_2, p_4, p_6) (p_3, p_5, p_6)
2	5 points, 1 plane	(p_1, p_2, P_3) (p_1, p_4, p_5) (p_2, p_4, p_6) (P_3, p_5, p_6)
3	4 points, 2 planes	(p_1, p_2, P_3) (p_1, P_4, p_5) (p_2, P_4, p_6) (P_3, p_5, p_6)
4	4 points, 2 planes	(p_1, p_2, P_3) (p_1, p_4, p_5) (p_2, p_4, P_6) (P_3, p_5, P_6)
5	3 points, 3 planes	(p_1, p_2, P_3) (p_1, P_4, p_5) (p_2, P_4, P_6) (P_3, p_5, P_6)
6	3 points, 3 planes	(p_1, p_2, P_3) (p_1, p_4, P_5) (p_2, p_4, P_6) (P_3, P_5, P_6)
7	2 points, 4 planes	(p_1, p_2, P_3) (p_1, P_4, P_5) (p_2, P_4, P_6) (P_3, P_5, P_6)
8	2 points, 4 planes	(p_1, P_2, P_3) (p_1, P_4, P_5) (P_2, P_4, p_6) (P_3, P_5, p_6)
9	1 point, 5 planes	(p_1, P_2, P_3) (p_1, P_4, P_5) (P_2, P_4, P_6) (P_3, P_5, P_6)
10	6 planes	(P_1, P_2, P_3) (P_1, P_4, P_5) (P_2, P_4, P_6) (P_3, P_5, P_6)

Table 1: The ten possible cluster combinations with points and planes

2. For each of the three remaining unfixed elements, establish a parametric representation of the point, or of the normal direction in the case of a plane. This representation is formulated from the interactions between the unfixed element and the fixed element in each cluster in which the unfixed element appears.

3. Use the remaining constraint in each cluster between two unfixed elements to establish three algebraic equations in the three unknown parameters of the representation in step 2.

4. Solve the system of equations.

5. Substitute the parameter values into the representation in step 2 to determine the position of the three remaining elements.

In the following sections, these steps are explained in greater detail, drawing on the specific examples of the cluster combination in Table 1. Although we include the first case in the discussion, we note that it is solved in its entirety in Nanua *et. al.* [NWM90]

4.1 Fixing the First Cluster

The first cluster is fixed by establishing a coordinate system based on the geometric elements of the first cluster and their relative positions. In every case but the last, element g_1 is a point. We fix this point as the origin.

In the first case, when g_3 is also a point, we position it on the xy-axis so that it is distance d_{13} from the origin. In all the other cases, when g_3 is a plane, we fix it as the xy-plane with normal in the positive z direction. This requires that we adjust any distance constraints between P_3 and the other points in the system by $-d_{13}$. When the final solutions are determined, P_3 must be moved by d_{13} along the z-axis to satisfy the original constraints of the system.

When g_2 is a point, it is positioned in the xz-plane so that it satisfies its constraints with the two previously positioned elements. In the two cases when g_2 is a plane and g_1 is a point, P_2 is positioned so that the intersection with P_3 is a line parallel to the y-axis.

In the last case, all six geometric elements are planes, and the only constraints are angles between the planes. This situation is underconstrained, as is the case of five planes and one point, but we can find necessary normal directions for the planes. We again position P_3 as the xy-plane. We require that P_2 intersect P_3 in the y-axis. We then position P_1 by initially placing it coincident to the xz-plane, and then rotating it as necessary to ensure the angle constraints are correct between the three planes. A complete coordinate system is not obtained by this method, however, since we do not have a distance from P_1 to the origin. We impose a coordinate system on the system by letting the intersection of the three planes be the origin.

4.2 Determining the Parametric Representation

The first step of the placement process completely determines the positions of the three geometric elements g_1, g_2, and g_3 of the first cluster. In this second step, we use the constraints between an unfixed element and two of the three fixed elements to establish a parametric representation of the unfixed element. For a point, the parameterization is the set of possible positions for the point as determined from the two fixed geometries. For a plane, the parameterization is the set of the possible directions of the normal vector. We begin with the three different ways in which the parameterization of a point can occur, and then cover the three ways in which a plane normal can be parameterized.

Unfixed Point Parameterizations

If the two fixed geometries are points and the unfixed geometry is also a point, then the unfixed point must lie on a circle which is the intersection of two spheres. Such a situation arises in case 2, for example, where p_4 is constrained to be distance d_{14} from p_1, and so must be on the sphere of radius d_{14} about p_1. At the same time, p_4 is distance d_{24} from p_2, implying that it must also lie on the sphere of radius d_{24} about p_2. A parametric form of this circle is

$$(A + B \sin u,\ C \cos u,\ D + E \sin u)$$

where u is the parameter and A, B, C, D, and E are simple functions of a subset of the constraints. We obtain this form by rotating the circle

$$(c_1, c_2 \cos u, c_3 \sin u)$$

about the y axis; by choosing the circle

$$(c_1, c_2 \sin u, c_3 \cos u)$$

as our base to rotate, we could obtain the trigonometrically symmetric form

$$(\tilde{A} + \tilde{B} \cos u,\ \tilde{C} \sin u,\ \tilde{D} + \tilde{E} \cos u)$$

If one fixed geometry is a plane and the other is a point, and the unfixed geometry is a point, then the unfixed point lies on a circle which is the intersection of a plane and a point. This only occurs when the fixed plane is P_3, such as in case 5 where p_5 is constrained by p_1 and P_3. The distance constraint between p_5 and P_3 means that p_5 must lie on a plane parallel to P_3 and the distance constraint between p_5 and p_1 constrains p_5 to lie on a sphere. The intersection of these two surfaces is a circle of form

$$(A + B \cos u,\ C \sin u,\ D)$$

Finally, in the case that both fixed geometries are planes and the unfixed geometry is a point, the unfixed point must lie on the intersection line of two planes. Such a situation arises only in case 8, where p_6 is constrained by the planes P_2 and P_3. The distance constraints between p_6 and each plane imply that p_6 lies on one plane parallel to P_2 and on another parallel to P_3. Because of the initial positioning of P_2 and P_3, the intersection of the two parallel planes is a line parallel to the y-axis. Its parameterization thus has the form

$$(A,\, u,\, B)$$

Unfixed Plane Parameterizations

We now turn to the cases where the unfixed geometry is a plane. When both fixed geometries are points, the plane must be simultaneously tangent to two spheres. The envelope of such a set of planes is a cone, and the normal direction to the plane points from the center of one of the spheres to a point on the circle of tangency between the sphere and the cone. The normal and the distance between one of the points and the unfixed plane are then used to determine the distance from the plane to the origin, thus fixing the plane up to the normal constraint. For example, in case 3, P_4 is a fixed distance from points p_1 and p_2. Since p_1 is the origin, we use the sphere of radius d_{14} about the origin and find the circle of tangency between that sphere and the cone-envelope of the planes. The parametric form of this circle is

$$(A + B\sin u,\, C\cos u,\, D + E\sin u)$$

where as before, u is the parameter and A, B, C, D, and E are simple functions of a subset of the constraints. In this example the plane's distance comes directly from the d_{14} constraint.

When one fixed geometry is a point and the other is a plane, a parametric representation of the normal of the unfixed plane is found from the known angle between the two planes. The distance between the point and the unfixed plane is used to fix the plane's distance from the origin. When the fixed plane is P_3, the normal must lie on a circle whose parametric form is

$$(A\cos u,\, B\sin u,\, C)$$

In this case, A and B are simply the cosine of the angle between P_3 and the unfixed plane, while C is the sine of that angle. The only other way in which this scenario occurs is when the fixed plane is P_2. The parametric form of the circle must then be rotated about the y-axis by the angle between P_2 and P_3, with a resulting form of

$$(A + B \sin u,\ C \cos u,\ D + E \sin u)$$

The final case is when both fixed geometries are planes. This only occurs when five or six geometries are planes as in cases 9 and 10 in Table 1. As an example, in case 10, the parametric representation of the normals to P_5 and P_6 are found using the angle between P_3 and the respective plane from cluster 4, while P_4's normal representation is found from its angle with P_2 from cluster 3. Thus the parametric forms are the same as in the previous two cases. As noted before, these two cases are underdetermined, since not every distance from the origin for these planes will be computable from the given information.

In summary, all the possible interactions between two fixed geometries and one unfixed geometry lead to parametric representations of the unfixed geometry which is one of three forms:

$$(A + B \cos u,\ C \sin u,\ D) \tag{1}$$

$$(A + B \sin u,\ C \cos u,\ D + E \sin u) \tag{2}$$

$$(A,\ u,\ B) \tag{3}$$

In Table 2 the ten cases are listed as in Table 1, with the corresponding parametric forms for g_4, g_5, and g_6 in the right hand column. Note in case one that the form for point p_5 is similar to Form 1, except that the constant component is in the x-coordinate instead of the z-coordinate. Also in this case, the form for p_6 is the trigonometrically symmetric form of Form 2 mentioned above.

4.3 Forming the Equations

In the previous step, a parametric representation was determined for each unfixed point or plane normal. We now use these representations and the constraint between the two unfixed geometries in each of the three unfixed clusters to set up three equations in the three parameters. There are three different equations, depending on whether the two unfixed geometries are both points, both planes, or one point and one plane.

Case	Geometric Elements	Parametric Forms		
1	6 points	p_4: Form 2	p_5: Form 1*	p_6: Form 2*
2	5 points, 1 plane	p_4: Form 2	p_5: Form 1	p_6: Form 1
3	4 points, 2 planes	P_4: Form 2	p_5: Form 1	p_6: Form 1
4	4 points, 2 planes	p_4: Form 2	p_5: Form 1	P_6: Form 1
5	3 points, 3 planes	P_4: Form 2	p_5: Form 1	P_6: Form 1
6	3 points, 3 planes	p_4: Form 2	P_5: Form 1	P_6: Form 1
7	2 points, 4 planes	P_4: Form 2	P_5: Form 1	P_6: Form 1
8	2 points, 4 planes	P_4: Form 2	P_5: Form 1	p_6: Form 3
9	1 point, 5 planes	P_4: Form 2	P_5: Form 1	P_6: Form 1
10	6 planes	P_4: Form 2	P_5: Form 1	P_6: Form 1

Table 2: The ten possible cluster combinations with points and planes

The first equation comes from the distance between two points. If the two points are $p_i : (x_i, y_i, z_i)$ and $p_j : (x_j, y_j, z_j)$, then the equation is

$$(p_i - p_j) \cdot (p_i - p_j) = {d_{ij}}^2 \tag{4}$$

The second equation comes from the angle between two planes. If the plane unit normals are $n_i : (x_i, y_i, z_i)$ and $n_j : (x_j, y_j, z_j)$, then the equation is

$$n_i \cdot n_j = \alpha_{ij} \tag{5}$$

The final equation comes from the distance between a point and a plane. If the point is $p_i : (x_i, y_i, z_i)$ and the plane has normal $n_j : (x_j, y_j, z_j)$ and distance to the origin l_j, then the equation is

$$n_j \cdot p_i = d_{ij} + l_j \tag{6}$$

To set up the actual equations to solve, we choose u as the parameter for the unfixed geometry g_4, v for g_5, and w for g_6. We then compute the coefficients of each parametric representation, and substitute the form into the appropriate equation above. For cases two through seven, the parametric representations always consist of two instances of Form 1 and one of Form 2, so it is not surprising that carrying out this process and making the standard trigonometric substitutions $\cos u = \frac{1-q_4^2}{1+q_4^2}$ $\sin u = \frac{2\,q_4}{1+q_4^2}$, $\cos v = \frac{1-q_5^2}{1+q_5^2}$ $\sin v = \frac{2\,q_5}{1+q_5^2}$, $\cos w = \frac{1-q_6^2}{1+q_6^2}$, and $\sin w = \frac{2\,q_6}{1+q_6^2}$, we obtain in each of these cases three equations with identical structure:

$$(A_1\,q_5^2 + A_2\,q_5 + A_3)\,q_4^2 + (A_4\,q_5^2 + A_6)\,q_4 + (A_7\,q_5^2 + A_8\,q_5 + A_9) = 0 \quad (7)$$
$$(B_1\,q_6^2 + B_2\,q_6 + B_3)\,q_4^2 + (B_4\,q_6^2 + B_6)\,q_4 + (B_7\,q_6^2 + B_8\,q_6 + B_9) = 0 \quad (8)$$
$$(C_1\,q_5^2 + C_3)\,q_6^2 + C_5\,q_5\,q_6 + (C_7\,q_5^2 + C_9) = 0 \quad (9)$$

where the A_i, B_i, and C_i are functions of the given constraints between the geometries. These same three equations arise using the modified parametric forms in case 1, as noted in Table 2.

The special case seven containing parametric Form 3 produces equations with a simpler form:

$$(A_1\,q_5^2 + A_3)\,q_4^2 + A_5\,q_5\,q_4 + (A_7\,q_5^2 + A_9) = 0 \quad (10)$$
$$B_3\,q_4^2 + B_5\,q_6\,q_4 + B_9 = 0 \quad (11)$$
$$C_5\,q_5\,q_6 + C_7\,q_5^2 + C_9 = 0 \quad (12)$$

Finally, the two underconstrained cases, nine and ten, yield three equations which all have the same form as Equation 10.

4.4 Solving the Equations

Because we desire all possible solutions to the assembly problem, the system of equations is solved symbolically as far as possible. When we perform elimination on Equations 7, 8, and 9, we obtain a polynomial of degree 16 in the remaining variable. Once the polynomial is determined via elimination, we must rely on numerical methods to find solutions. For each solution of the polynomial, there is at most one solution for the remaining two variables which satisfy all three equations. Thus there are at most 16 different positions in real space which satisfy the given constraints.

Practically speaking, solving the equations is the most difficult part of solving the problem, as sometimes the original choice of elimination order yields the degenerate equation, $0 = 0$. When this occurs, a different elimination order must be chosen to successfully find solutions.

In the final step, having obtained values for each of the parameters u, v, and w, we substitute into the original parametric forms to obtain the positions of the points and the directions of the normals. In addition, except in the final two cases, we are able to determine the distance of each plane from the origin, since we either have the distance directly as a constraint or we know the distance from some fixed point to the plane and can compute the distance from the origin using that information. For example, in case seven, we have the distance from p_2 to P_6 is d_{26}. Knowing the normal $N_6 : (x_6, y_6, z_6)$ of P_6 we can compute

$$d_{16} = p_2 \cdot N_6 - d26$$

5 Implementation

We have implemented a library which solves all ten cases of this problem, making distance assumptions in the last two cases to make them well-constrained. This library uses Maple to carry out the necessary numerical solving processes. Currently we have tied this library in with a simple program which reads a data file of distances and angles between geometric entities and which textually reports the positions of the points and/or planes which satisfy the given constraints. In the appendix to this report, we demonstrate results from this program.

We have also implemented a graphical user interface front-end to the assembler library using Open Inventor. This interface displays four objects and allows the user to interactively select a matching between points and planes of the four objects which correspond to well-constrained matching problem described in the paper. Matched geometries are indicated in the interface by similar coloring. When a valid matching is selected, the program extracts the positions of the selected points and planes, and computes the distances and angles to input to the library. An example of the interface is shown in Figure 3.

We are still completing the graphical back-end. This will use the new positions of the geometries generated by the solver to compute transformations which take the geometries from their original positions to their new positions. The corresponding transformation will be applied to each object and the assembled object will be displayed. The user will be able to browse through all the solutions returned by the assembler library.

A major advantage to our method over numerical solving of the equation system is seen by browsing through all the returned solutions: A numerical

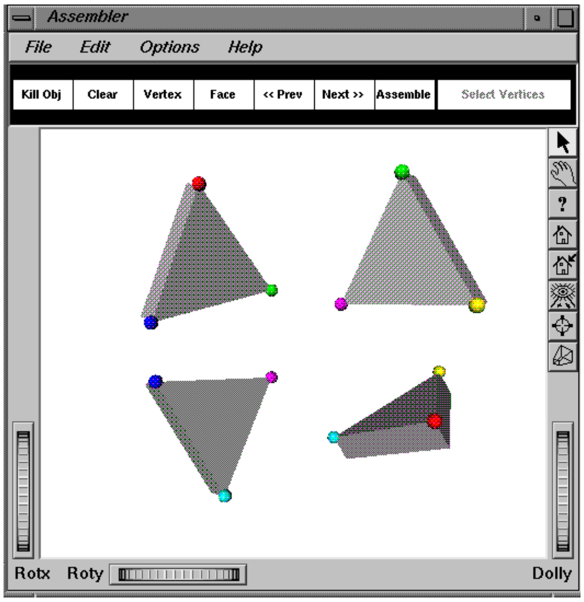

Figure 3: Four triangular wedges to be assembled into an octahedron

solution returns only one solution, which may be a solution in which the four objects intersect each other. Since our method returns all solutions, if there is a solution in which the objects do not intersect, it will be found and displayed.

6 Acknowledgements

Thanks to John Hills and Stephen Williard who developed and programmed the graphical user interface to the assembler library, and to Jean-Guy Speton who assisted in programming the assembler library. This work was funded in part by NSF grant number CCR-9505745, by the Washington and Lee University Glenn Grant program, and by the Washington and Lee University R. E. Lee Research Scholars program.

References

[Ald88] B. Aldefeld. Variation of geometries based on a geometric-reasoning method. *CAD*, 20(3):117–126, April 1988.

[BFH+95] W. Bouma, I. Fudos, C. Hoffmann, J. Cai, and R. Paige. A geometric constraint solver. *CAD*, 27, 1995.

[Bor81] A. H. Borning. The programming language aspects of ThingLab, a constraint oriented simulation laboratory. *ACM TOPLAS*, 3(4):353–387, 1981.

[Bos85] N.K. Bose, editor. *Multidimensional Systems Theory*, pages 184–232. D. Reidel Publishing Co., 1985.

[Bru90] B. Bruderlin. Symbolic computer geometry for computer aided geometric design. In *Advances in Design and Manufacturing Systems*. NSF, 1990.

[DH95] D.-Z. Du and F. Hwang, editors. *Computing in Euclidean Geometry*, pages 266–298. World Scientific Publishing Co Pte Ltd, 2 edition, 1995.

[FH93] I. Fudos and C. Hoffmann. Correctness of a geometric constraint solver. Technical Report CSD-TR-93-076, Purdue University, 1993.

[Fud93] I. Fudos. Editable representations for 2d geometric design. Master's thesis, Purdue University, 1993.

[HV95] Christoph M. Hoffmann and Pamela J. Vermeer. A spatial constraint problem. In *Computational Kinematics Workshop*, Nice, France, Sept. 1995.

[Kon92] K. Kondo. Algebraic method for manipulation of dimensional relationships in geometric models. *CAD*, 24(3):141–147, March 1992.

[Kra92] Glenn A. Kramer. *Solving Geometric Constraint Systems: A case study in kinematics*. MIT Press, 1992.

[NWM90] P. Nanua, K. J. Waldron, and V. Murthy. Direct kinematic solution of a stewart platform. *IEEE Transactions on Robotics and Automation*, 6(4):438–444, Aug. 1990.

[Sut63] I. E. Sutherland. Sketchpad: A man-machine graphical communication system. In *Proceedings–Spring Joint Computer Conference*, pages 329–346, 1963.

[Wu86] W.-T. Wu. Principles of mechanical theorem proving. *J. of Automated Reasoning*, 2:221–252, 1986.

Appendix

In this appendix we provide demonstrations of the output of our assembly library on data comprised of independent points and planes, rather than points and planes which are part of complete three-dimensional objects. We shows results from running an example of each of the well-constrained cases, *i.e.*, cases one through eight.

In each case, we begin with a set of known points and planes. We make sure that geometries g_1, g_2, and g_3 establish a coordinate system in $\Re^3$ as discussed in Section 4.1. We then select other arbitrary geometries to play the roles of g_4, g_5, and g_6. From these geometries, we compute the set of distances and angles between geometries within each cluster. This is the input required by the assembly library. With this input data, the program generates solutions; in each case, one of the solutions determined by the program is the same as the original data. Solutions always appear in pairs, as reflections about the xz-plane.

We show this data in table form. Each table lists the original geometries with which we started and the solutions which our program generated. We show only one of the reflective pairs. To obtain the other solution, simply negate the y-coordinate of each of g_4, g_5, and g_6. Also, because the values of g_1, g_2, and g_3 will always be exactly the desired values, we only list that part of each solution in the first solution. Below each solution set, we give the input data computed from the original geometries.

Case 1: Six Points

Datum	Original	Solution 1
p_1	$(0,0,0)$	$(0,0,0)$
p_2	$(\frac{1}{2},0,\frac{\sqrt{3}}{2})$	$(0.5,0,0.866025)$
p_3	$(1,0,0)$	$(1,0,0)$
p_4	$(0,\frac{\sqrt{2}}{\sqrt{3}},\frac{1}{\sqrt{3}})$	$(0,0.816497,0.57735)$
p_5	$(\frac{1}{2},\frac{\sqrt{2}}{\sqrt{3}},-\frac{1}{2\sqrt{3}})$	$(0.5,0.816497,-0.28867)$
p_6	$(1,\frac{\sqrt{2}}{\sqrt{3}},\frac{1}{\sqrt{3}})$	$(1,0.816497,0.57735)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12}:1$	$d_{14}:1$	$d_{24}:1$	$d_{35}:1$
$d_{13}:1$	$d_{15}:1$	$d_{26}:1$	$d_{36}:1$
$d_{23}:1$	$d_{45}:1$	$d_{46}:1$	$d_{56}:1$

Case 2: Five Points and One Plane

Datum	**Original**	**Solution 1**
p_1	$(0,0,0)$	$(0,0,0)$
p_2	$(3,0,4)$	$(3,0,4)$
P_3	$(0,0,1,0)$	$(0,0,1,0)$
p_4	$(3,4,0)$	$(3,4,-6.62516e-07)$
p_5	$(2,2,2)$	$(2,2,2)$
p_6	$(6,1,1)$	$(6,0.999998,1)$

Datum	Solution 2
p_4	$(-2.53245,-1.17034,4.14934)$
p_5	$(-2.67541,0.917702,2)$
p_6	$(0.411151,-1.816,1)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12}:5$	$d_{14}:5$	$d_{24}:4\sqrt{2}$	$d_{35}:2$
$d_{13}:0$	$d_{15}:2\sqrt{3}$	$d_{26}:\sqrt{19}$	$d_{36}:1$
$d_{23}:4$	$d_{45}:3$	$d_{46}:\sqrt{19}$	$d_{56}:3\sqrt{2}$

Case 3: Four Points and Two Planes, Planes in Separate Clusters

Datum	**Original**	**Solution 1**
p_1	$(0,0,0)$	$(0,0,0)$
p_2	$(-4,0,2)$	$(-4,0,2)$
P_3	$(0,0,1,0)$	$(0,0,1,0)$
P_4	$(\frac{3\sqrt{2}}{10},\frac{2\sqrt{2}}{5},\frac{-\sqrt{2}}{2},-4)$	$(0.424264,0.565685,-0.707107,-4)$
p_5	$(2,2,-2)$	$(2,-2,2)$
p_6	$(2,-2,2)$	$(2,2,2)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12}:\sqrt{20}$	$d_{14}:-4$	$d_{24}:\frac{\sqrt{2}}{5}-4$	$d_{35}:-2$
$d_{13}:0$	$d_{15}:\sqrt{12}$	$d_{26}:\sqrt{8}$	$d_{36}:2$
$d_{23}:2$	$d_{45}:\frac{12\sqrt{2}}{5}-4)$	$d_{46}:\frac{-6\sqrt{2}}{5}-4$	$d_{56}:\sqrt{32}$

Case 4: Four Points and Two Planes, Planes in Same Cluster

Datum	**Original**	**Solution 1**
p_1	$(0, 0, 0)$	$(0, 0, 0)$
p_2	$(3, 0, 4)$	$(3, 0, 4)$
P_3	$(0, 0, 1, 0)$	$(0, 0, 1, 0)$
p_4	$(6, 1, 1)$	$(6, 1, 1)$
p_5	$(2, 2, 2)$	$(2, 2, 2)$
P_6	$(\frac{-1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}, 4)$	$(-0.707107, -4.75513e - 10, 0.707107, -4)$

Datum	Solution 2
p_4	$(5.8504, -1.59242, 1.1122)$
p_5	$(2.63468, 1.02882, 2)$
P_6	$(-0.484121, 0.51539, 0.707107, -4.66896)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12} : 5$	$d_{14} : \sqrt{38}$	$d_{24} : \sqrt{19}$	$d_{35} : 2$
$d_{13} : 0$	$d_{15} : 2\sqrt{3}$	$d_{26} : \frac{1}{\sqrt{2}} - 4$	$\alpha_{36} : \frac{\pi}{4}$
$d_{23} : 4$	$d_{45} : 3\sqrt{2}$	$d_{46} : \frac{-5\sqrt{2}}{2} - 4$	$d_{56} : -4$

Case 5: Three Points and Three Planes, Planes Not All in Same Cluster

Datum	**Original**	**Solution 1**
p_1	$(0, 0, 0)$	$(0, 0, 0)$
p_2	$(20, 0, 5)$	$(20, 0, 5)$
P_3	$(0, 0, 1, 0)$	$(0, 0, 1, 0)$
P_4	$(\frac{-\sqrt{14}}{7}, \frac{3}{\sqrt{14}}, \frac{-1}{\sqrt{14}}, -3)$	$(0.534522, -0.801784, 0.267261, -3)$
p_5	$(-20, -18, -9)$	$(-20 - 18 - 9)$
P_6	$(\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, -5)$	$(0.57735, -0.57735, 0.57735, -5)$

Datum	Solution 2
P_4	$(0.411017, 0.501511, 0.761283, -3)$
p_5	$(-10.3728, 24.8275, -9)$
P_6	$(0.72507, 0.375419, 0.57735, -7.9544)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12} : \sqrt{425}$	$d_{14} : -3$	$d_{24} : \frac{45}{\sqrt{14}} - 3$	$d_{35} : -9$
$d_{13} : 0$	$d_{15} : \sqrt{805}$	$d_{26} : \frac{25}{\sqrt{3}} - 5$	$\alpha_{36} : \arccos(\frac{1}{\sqrt{3}}) + \frac{\pi}{2}$
$d_{23} : 5$	$d_{45} : \frac{5}{\sqrt{14}} - 3$	$\alpha_{46} : \pi - \arccos(\frac{\sqrt{42}}{7})$	$d_{56} : \frac{-11}{\sqrt{3}} - 5$

Case 6: Three Points and Three Planes, Planes in Same Cluster

Datum	**Original**	**Solution 1**
p_1	$(0, 0, 0)$	$(0, 0, 0)$
p_2	$(15, 0, 20)$	$(15, 0, 20)$
P_3	$(0, 0, 1, 0)$	$(0, 0, 1, 0)$
p_4	$(20, -18, -9)$	$(20, -18, -9)$
P_5	$(\frac{\sqrt{38}}{19}, \frac{3}{\sqrt{38}}, \frac{-5}{\sqrt{38}}, -12)$	$(0.324443, 0.486664, -0.811107, -12)$
P_6	$(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, 8)$	$(0.57735, 0.57735, 0.57735, 8)$

Datum	Solution 2
p_4	$(11.0803, 26.0171, -2.31019)$
P_5	$(0.571964, -0.122321, -0.811107, -12)$
P_6	$(0.749448, -0.324028, 0.57735, 5.41853)$

Datum	Solution 3
p_4	$(25.2109, -1.67083, -12.9082)$
P_5	$(-0.178915, 0.556861, -0.811107, -12)$
P_6	$(-0.397362, 0.713281, 0.57735, 22.6207)$

Datum	Solution 4
p_4	$(23.6551, -10.3719, -11.7413)$
P_5	$(0.0648685, 0.581289, -0.811107, -12)$
P_6	$(0.247936, 0.777942, 0.57735, 12.9412)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12} : 25$	$d_{14} : \sqrt{805}$	$d_{24} : \sqrt{1190}$	$\alpha_{35} : \frac{3\pi}{2} - \arccos(\frac{5}{\sqrt{38}})$
$d_{13} : 0$	$d_{15} : -12$	$d_{26} : \frac{35}{\sqrt{3}} + 8$	$\alpha_{36} : \arccos(\frac{1}{\sqrt{3}}) + \frac{\pi}{2}$
$d_{23} : 20$	$d_{45} : \frac{31}{\sqrt{38}} - 12$	$d_{46} : \frac{-7}{\sqrt{3}} + 8$	$\alpha_{56} : \frac{\pi}{2}$

Case 7: Two Points and Four Planes, Three Planes in Same Cluster

Datum	**Original**	**Solution 1**
p_1	$(0, 0, 0)$	$(0, 0, 0)$
p_2	$(15, 0, 20)$	$(15, 0, 20)$
P_3	$(0, 0, 1, 0)$	$(0, 0, 1, 0)$
P_4	$(\frac{3}{\sqrt{29}}, \frac{-2}{\sqrt{29}}, \frac{4}{\sqrt{29}}, -18)$	$(0.557062, -0.371391, 0.742799, -18)$
P_5	$(\frac{\sqrt{38}}{19}, \frac{3}{\sqrt{38}}, \frac{-5}{\sqrt{38}}, -12)$	$(0.32447, 0.486646, -0.811107, -12)$
P_6	$(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, 8)$	$(0.577383, 0.577318, 0.57735, 7.99951)$

Datum	Solution 2
P_4	$(0.409453, 0.322297, 0.853506, -18)$
P_5	$(0.481134, -0.332588, -0.811107, -12)$
P_6	$(0.567551, -0.586986, 0.57735, 8.14699)$

Datum	Solution 3
P_4	$(0.557052, 0.371391, 0.742807, -18)$
P_5	$(-0.324431, 0.486672, -0.811107, -12)$
P_6	$(-0.310862, 0.755004, 0.57735, 21.3232)$

Datum	Solution 4
P_4	$(0.409402, 0.322261, 0.853544, -18)$
p_5	$(-0.2102, 0.545822, -0.811107, -12)$
P_6	$(-0.437164, 0.689605, 0.57735, 23.2177)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12} : 25$	$d_{14} : -18$	$d_{24} : \frac{125}{\sqrt{29}} - 18$	$\alpha_{35} : \frac{3\pi}{2} - \arccos(\frac{5}{\sqrt{38}})$
$d_{13} : 0$	$d_{15} : -12$	$d_{26} : \frac{35}{\sqrt{3}} + 8$	$\alpha_{36} : \arccos(\frac{1}{\sqrt{3}}) + \frac{\pi}{2}$
$d_{23} : 20$	$\alpha_{45} : \pi - \arccos(\frac{10\sqrt{1102}}{551})$	$\alpha_{46} : \arccos(\frac{5}{\sqrt{87}})$	$\alpha_{56} : \frac{\pi}{2}$

Case 8: Two Points and Four Planes, Two Planes in Each Cluster

Datum	**Original**	**Solution 1**
p_1	$(0,0,0)$	$(0,0,0)$
P_2	$(\frac{-1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}, 0)$	$(-0.707107, 0, 0.707107, 0)$
P_3	$(0,0,1,0)$	$(0,0,1,0)$
P_4	$(-1,0,0,-6)$	$(-1, 1.37985e-09, -9.10207e-10, -6)$
P_5	$(0, \frac{-1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, -12)$	$(-1.41421e-09, -0.707107, 0.707107, -12)$
p_6	$(15,12,20)$	$(15,12,20)$

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$d_{12} : 0$	$d_{14} : -6$	$\alpha_{24} : \frac{3\pi}{4}$	$\alpha_{35} : \frac{3\pi}{4}$
$d_{13} : 0$	$d_{15} : -12$	$d_{26} : \frac{5}{\sqrt{2}}$	$d_{36} : 20$
$\alpha_{23} : \frac{\pi}{4}$	$\alpha_{45} : \frac{\pi}{2}$	$d_{46} : -21$	$d_{56} : 4\sqrt{2} - 12$