472 KSME Journal, Vol. 9, No. 4, pp. 472~482, 1995

A Numerical Treatment for Ill-Determined Systems in Mechanical Assemblies

Jay Jung Kim*

(Received March 30, 1995)

The locations of components in mechanical assemblies are determined by reconciling various constraints among the components that arise from physical, geometric and kinematic relationships, human factors, maintenance concerns, etc. Among them some constraints require that particular spatial relationships between components be maintained exactly, i.e., equality constraints. In general the equality constraints can be expressed as systems of equations. However the systems of equations deduced from the equality constraints are mostly ill-determined so that special numerical attentions are required. This paper proposes a numerical treatment for ill-determined systems in mechanical assemblies. It utilizes singular value decomposition and Newton-Raphson methods in corporation with minimum weighted deviation criteria. The treatment was implemented on an assembly modeler for automatic packaging task.

Key Words: CAD, Assembly, Packaging, Equality Constraint, Ill-Determined System, Singular Value Decomposition, Minimum Weighted Deviation, Newton- Raphson Method

1. Introduction

The process of locating components in an available space while satisfying spatial constraints among the components is called packaging. The task requires extensive spatial reasoning about geometric shapes. It is a generic design task common to many domains. It is also very time-consuming, especially as machines become more compact and complex.

In general, the packaging process can be divided into four stages as shown in Fig. 1. The process starts with geometric descriptions of the individual components, the design space in which they must fit, as well as a specification of packaging goals. The packaging goals may be thought of as objective functions to be optimized. An example of a packaging goal might be to minimize the volume of the design space.

Constraint identification is the second phase of packaging, in which constraints on the components are identified. One of the most basic constraints in assembly design is that no geometric interference be allowed among the components, i.e., no two solid objects may occupy the same space at the same time. Some constraints require that particular spatial relationships between components be maintained exactly, i.e., equality constraints. Other constraints require that a relationship be maintained only approximately, i.e., an inequality constraint. In the design of internal combustion engines, an example of the former is that the camshaft must be parallel to the crankshaft and an example of the latter is that the oil filter must be accessible to allow for easy maintenance.

Conceptual packaging is a stage in which various packaging alternatives are generated and approximately evaluated. The major task in this stage is to determine the approximate relative locations of the components in the design space.

Detail packaging is the final phase of packaging in which one or more packaging alternatives are selected from the conceptual packaging stage

* Department of Mechanical Design and Production Engineering, Hanyang University, 17 Haengdang-dong, Sungdong-ku, Seoul 133-791, Korea

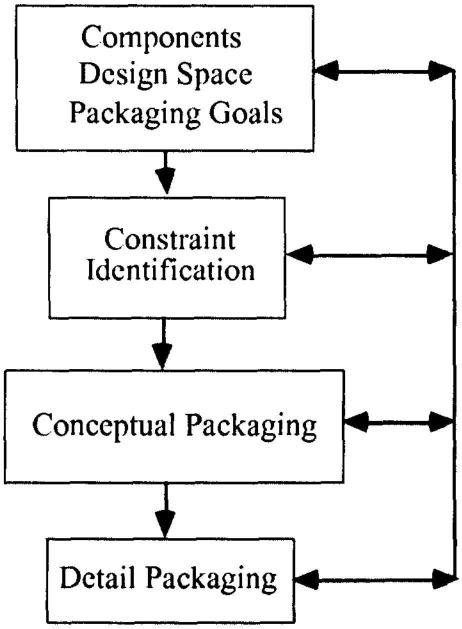


Fig. 1 Packaging Process

and refined through more detailed analysis and evaluation. All constraints, including geometric interference, are checked at the detail level and an optimal solution is sought.

As shown in Fig. 1, packaging is an iterative process involving continuous refinement of the problem. When a problem appears unsolvable, the designers may drop or relax some of the constraints, reshape the components or, if necessary, reshape the design space. Similarly, constraints may be added when many configurations look feasible. During the packaging process, designers generally refine the problem and meet the design goals by trading off the quality of the solution for a restrictiveness of the constraints. Also, depending on the assembly to be designed, the most difficulty may occur in the conceptual packaging stage, the detail stage or both. In the case of routine assembly designs, the conceptual packaging effort may constitute only a small portion of the whole packaging process.

As discussed, constraints in packaging might be classified into either inequality constraints or equalities. If an optmization technique is applied to solve packaging problems, while the inequality constraints are treated as penalty functions and included in the objective function , the equality constraints can be used to reduce the number of variables (Kim and Gossard, 1991). Moreover the equality constraints might be used to infer the initial locations of the components before optimizations start. However, since the number of independent equations generated from the equality constraints is less than that of location variables, i.e., underconstrained, infinite number of initial locations could be generated. Therefore, if design constraints result in an underconstrained system of equations, one solution should be chosen among many by applying proper criteria. On the other hand, even if the number of independent equations from equality constraints is as same as that of variables, the system might still include, in general, redundant equations. Therefore the redundant equations must be properly handled.

Not only in packaging but in the process of general design, it is quite common that some of the design constraints are given with a system of equations. Often, inherently or by designers' mistake, the system of equations is ill-determined; it is underconstrained or overconstrained and/or it has redundant or conflicting equations. It is natural that redundant (dependent) equations appear in the system of equations since design constraints are frequently given with redundant information. However, conflicting equations are mostly result of designers' mistakes and should be properly modified by designers.

In the case of an overconstrained, there is no solution unless the number of independent equations in the system is as same as that of the variables. Also it is clear that conflicting equations preclude any solutions and redundant equations, although not affecting the solution itself, may cause numerical problem such as singularity in computational environments.

The rest of the paper is organized as follow. The second chapter begins with the brief review of the related work and the third chapter introduces equality constraints. The fourth chapter addresses the minimum weighted deviation algorithm which utilizes a novel combination of singular value decomposition (SVD) and Newton-Raphson methods in order to treat ill-determined systems. The fifth chapter describes the implementation of the algorithm in the assembly modeler and the last chapter concludes the paper.

2. Related Work

Substantial work has been done to deal with geometric constraints. Light and Gossard (1983) used row and column operations on the Jacobian of a system of non-linear equations to detect overconstrained and underconstrained dimensioning scheme. Also, Serrano and Gossard (1987) have worked on constraint management to examine the system of equations. They used the graph theory to generate the causality information among variables. To detect any redundant or conflicting equations Gauss-elimination technique was used on the Jacobian.

Pabon (1985) worked on the problem of redesigning an existing design to meet new requirements. He has developed a method to single out one solution out of many by using Lagrange multipliers for underconstrained systems. To the solution, the method applies the criterion that the solution should be minimally deviated from initial conditions. However, the method is based on the assumption that there is no redundant or conflicting equation in the systems.

Approaches have been developed to determine the locations of components from spatial relationships. Lee and Rocheleau (1987) developed an assembly modeler computing the locations from the mating relationships (equality constraints). From the mating relationships, the system of equations expressed as the functions of location variables are generated and the Newton-Raphson method is applied to compute the transformation matrices describing the relative locations between components. In Lee and Rocheleau's case, even though there are more equations than variables, the number of variables is as same as that of independent equations so that an unique solution is generated. To handle redundant equations, they incorporated the least square technique into the Newton-Raphson method.

Mullineux (1987) has done preliminary work on computing the locations of components from spatial relationships that include both equality and inequality constraints. By transforming these equalities and inequalities into a penalty function, he formulated the problem as an unconstrained optimization problem. Also, Witkin et. al. (1987) attempted animation and construction of objects by formulating equality and inequality spatial constraints with objective functions, i.e., which they called energy functions, essentially similar to the penalty functions in unconstrained optimizations. A major drawback with these approaches is that some of the equality constraints may not be exactly satisfied. Also these approaches may make a system numerically very stiff.

Kim and Gossard (1991) formulated the packaging task as a constrained optimization problem in a solid modeling environment. The spatial relationships are represented as objective functions, equality or inequality constraints. Unlike Mullineux's approach, Kim and Gossard utilized the equality constraints to reduce the number of variables for optimization. In general since the optimization is numerically sensitive to initial conditions, it is better to provide "good" initial locations for packaging. It may be ideal if the initial locations can be computed from the equality constraints. However this computation is a very cumbersome task because the equality constraints are mostly ill-determined so that it requires special numerical attentions, which are the main discussion of this paper.

3. Representing Equality Constraints

3.1 Locations of assembly components

In the assembly modeler in this study, there are two types of components: reference components and movable components. The components whose locations are fixed are called reference components. For convenience, in the discussion to follow we will assume there is only one reference component which has an associated reference coordinate system fixed to it. The components whose locations are to be determined are called movable components. We assume there is an unique body coordinate system fixed to each of movable components.

Each movable component has six degrees of freedom: three for translation and three for rotation. The location of a component's body coordinate system with respect to the reference coordinate system can be defined by a location vector:

$$\boldsymbol{r}=(x,y,z,\phi,\theta,\psi) \qquad (1)$$

where position is specified by three translational components, x, y, z and orientation by three angles of rotation in sequence: ϕ(roll), θ(pitch), ψ(yaw) about the z, y and x axes of the body coordinate system respectively.

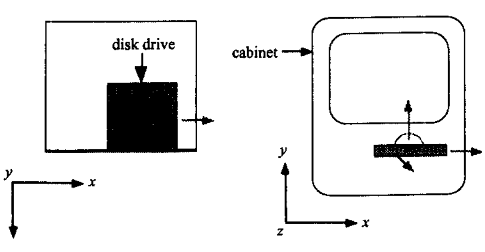
A 4×4 matrix describes the homogeneous coordinate transformation from each (movable) component's body coordinate system to the reference coordinate system:

$$\boldsymbol{T}(\boldsymbol{r})=\begin{bmatrix} c\phi c\theta & c\phi s\theta s\psi - s\phi c\psi & c\phi s\theta s\psi + s\phi c\psi & x \\ s\phi c\theta & s\phi s\theta s\psi + c\phi s\psi & s\phi s\theta c\psi - c\phi s\psi & y \\ -s\theta & c\theta s\psi & c\theta c\psi & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2)$$

where c and s denote the *cos* and *sin* functions, respectively.

3.2 Equality constraints

Equality constraints can be applied to specify the relationships that have the highest priority or that must be satisfied. For example, in the packaging of personal computers, internal floppy disk drives are usually located adjacent to the front side of the computer cabinet as shown in Fig. 2 (a), so that they can be conveniently accessed by the user inserting diskettes. In this case, we can assume that designers give higher priority to the position of the disk drive in the z-direction than to the position in the x- and y-directions, as shown in Fig. 2 (b). While the position in the z-direction is fixed by an equality constraint, the position of the disk drive in the x-and y-directions will be determined by its relationship to other components in the cabinet, i.e., by packaging goals and inequality constraints (Kim and Gossard, 1991). However it should be noted that in general the original location of the disk drive in the form of a CAD database is different from the one shown in Fig. 2 (a) satisfying the adjacent constraint. Therefore before the optimization starts, computation which positions the disk drive from an arbitrary location to the one in Fig. 2 (a) is required so that it can be served as initial conditions.



(a) Top view (b) Front view

Fig. 2 Locating a disk drive for a personnel computer. Dotted arrows represent DOF in the directions indicated.

The concept of the equality constraints can be applied to assembling a mechanism with components by specifying lower-pair mechanisms such as spherical pairs, plane pairs, cylindrical pairs, etc. Among these we like to discuss the constraints applied to planar faces and characteristic lines.

3.2.1 Constraints on faces

There are several basic spatial relationships among components that may be described by equality constraints on faces. Among the more important of these relate two planar faces: they are called the coplanar-plus and coplanar-minus constraints. These constraints allow two translations along the axes defining the plane and a rotation about an axis perpendicular to the plane.

Coplanar-plus: Two planar faces, f_1 and f_2, as shown in Fig. 3 (a), satisfy the coplanar-plus constraint if two points, p_1 and p_2 on the faces f_1 and f_2, respectively, are on the same plane, and the unit surface normal vectors of the faces f_1 and f_2 are in the same direction. The condition that the two points p_1 and p_2 be on the same plane can be formulated as:

$$\left\{ \boldsymbol{T}_1 \begin{bmatrix} n_{1x} \\ n_{1y} \\ n_{1z} \\ 0 \end{bmatrix} \right\}^T \cdot \left\{ \boldsymbol{T}_1 \begin{bmatrix} p_{1x} \\ p_{1y} \\ p_{1z} \\ 1 \end{bmatrix} - \boldsymbol{T}_2 \begin{bmatrix} p_{2x} \\ p_{2y} \\ p_{2z} \\ 1 \end{bmatrix} \right\} = 0 \tag{3}$$

where $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ are the transformation matrices locating the components with respect to the reference coordinate system,

$$\boldsymbol{n}_1 = [\, n_{1x} \; n_{1y} \; n_{1z} \,]^T \text{ and}$$
$$\boldsymbol{n}_2 = [\, n_{2x} \; n_{2y} \; n_{2z} \,]^T \tag{4}$$

are the unit normal vectors of faces f_1 and f_2 defined with respect to the body coordinate systems, and

$$\boldsymbol{p}_1 = [\, p_{1x} \; p_{1y} \; p_{1z} \,]^T \text{ and}$$
$$\boldsymbol{p}_2 = [\, p_{2x} \; p_{2y} \; p_{2z} \,]^T \tag{5}$$

are the points on the respective faces also defined with respect to the body coordinate systems.

The condition that the unit normal vectors of the faces f_1 and f_2 be in the same direction can be formulated as:

$$\boldsymbol{T}_1 \begin{bmatrix} n_{1x} \\ n_{1y} \\ n_{1z} \\ 0 \end{bmatrix} = \boldsymbol{T}_2 \begin{bmatrix} n_{2x} \\ n_{2y} \\ n_{2z} \\ 0 \end{bmatrix} \tag{6}$$

Eq. (3) yields one independent equation, while Eq. (6) yields two independent equations. In other words Eq. (3) constrains one translation and Eq. (6) constrains two rotations. Therefore, three equations can be derived from the coplanar-plus constraint, leaving three degrees of freedom for one component relative to the other as shown in Fig. 3 (a).

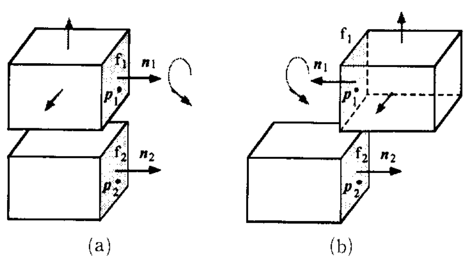
Coplanar-minus: The coplanar-minus constraint is similar to the coplanar-plus except that the direction of the normal vector is opposite:

$$\boldsymbol{T}_1 \begin{bmatrix} n_{1x} \\ n_{1y} \\ n_{1z} \\ 0 \end{bmatrix} = -\boldsymbol{T}_2 \begin{bmatrix} n_{2x} \\ n_{2y} \\ n_{2z} \\ 0 \end{bmatrix} \tag{7}$$

Three independent equations can also be derived from Eqs. (3) and (7). Similar to coplanar-plus, Eq. (3) constrains one translation and Eq. (7) constrains two rotations, leaving three degrees of freedom for one component if the other is fixed, as shown in Fig. 3 (b). Coplanar-minus constraints can be applied when two components need to be in contact. Note that the contact between the two faces cannot be ensured with only one coplanar-minus constraint because Eqs. (3) and (7) were derived for an infinite plane. For convenience, whenever a coplanar-plus or coplanar-minus constraint is specified between two faces, we will refer to them as mating faces.

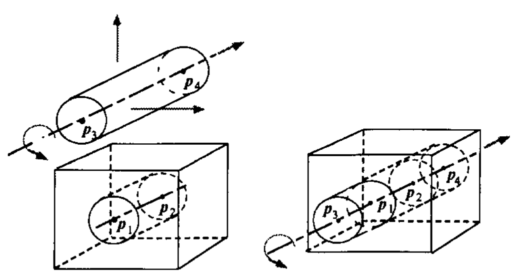
3.2.2 Constraints on characteristic lines

There are several basic spatial relationships among components that may be described by equality constraints on characteristic lines.



(a) Coplanar-plus: 3DOF
(b) Coplanar-minus: 3DOF

Fig. 3 Constraints on faces. Dotted arrows represent DOF in the directions indicated. For sim plicity the body coordinate systems are not displayed.



(a) Parallel: 4DOF (b) Coaxial: 2DOF

Fig. 4 Constraints on centerlines. Dotted arrows represent DOF in the directions indicated. For simplicity the body coordinate systems are not displayed

Among the more important of these are the parallel and coaxial constraints which relate two centerlines.

Parallel: The parallel constraint is met by two components if, as shown in Fig. 4 (a), a centerline of one component is parallel with a centerline of the other, that is, the cross product of the two vectors representing the two centerlines is equal to the zero vector:

$$\boldsymbol{T}_1\begin{bmatrix} p_{2x}-p_{1x} \\ p_{2y}-p_{1y} \\ p_{2z}-p_{1z} \\ 0 \end{bmatrix} \times \boldsymbol{T}_2\begin{bmatrix} p_{4x}-p_{3x} \\ p_{4y}-p_{3y} \\ p_{4z}-p_{3z} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{8}$$

where $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ are the transformation matrices locating the components with respect to the reference coordinate system. $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ define the points on the centerline of the bore relative to the body coordinate system and are given by:

$$\boldsymbol{p}_1=[\, p_{1x}\ p_{1y}\ p_{1z}\,]^T \text{ and}$$
$$\boldsymbol{p}_2=[\, p_{2x}\ p_{2y}\ p_{2z}\,]^T \tag{9}$$

Similarly, $\boldsymbol{p}_3$ and $\boldsymbol{p}_4$ define the points on the centerline of the cylinder relative to the body coordinate system and are given by:

$$\boldsymbol{p}_3=[\, p_{3x}\ p_{3y}\ p_{3z}\,]^T \text{ and}$$
$$\boldsymbol{p}_4=[\, p_{4x}\ p_{4y}\ p_{4z}\,]^T \tag{10}$$

Two independent equations can be derived from Eq. (8), leaving four DOF for one component. The parallel constraint is useful when orthogonality for the directions of components is desirable. In general, many components in a mechanical assembly are located such that the directions of centerlines are orthogonal. Two main reasons for this are to meet structural requirements of the mechanism of a machine and to make the machine easy to assemble and disassemble.

Coaxial: The coaxial constraint is a special case of the parallel constraint. The parallel constraint expressed by Eq. (8) still holds, and, in addition, the distance between the centerlines is zero, as shown in Fig. 4 (b). This is similar to the right circular cylindrical pair that allows a rotation about the cylinder axis and a translation along the axis. This constraint can be rephrased as if one of the two points p_3 and p_4 in Fig. 4 (b) lies on the centerline of the other or vice versa, while Eq. (8) still holds. In this case, the distance between the two centerlines becomes zero. The zero distance condition can be expressed as:

$$\boldsymbol{T}_1\begin{bmatrix} p_{2x}-p_{1x} \\ p_{2y}-p_{1y} \\ p_{2z}-p_{1z} \\ 0 \end{bmatrix} \times \boldsymbol{T}_2\begin{bmatrix} p_{3x}-p_{1x} \\ p_{3y}-p_{1y} \\ p_{3z}-p_{1z} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

or

$$\boldsymbol{T}_1\begin{bmatrix} p_{2x}-p_{1x} \\ p_{2y}-p_{1y} \\ p_{2z}-p_{1z} \\ 0 \end{bmatrix} \times \boldsymbol{T}_2\begin{bmatrix} p_{4x}-p_{1x} \\ p_{4y}-p_{1y} \\ p_{4z}-p_{1z} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{11}$$

Eq. (11) yields two independent equations, just as two equations have been derived from Eq. (8) for the parallel constraint. Therefore, a coaxial constraint yields four independent equations and leaves two DOF for one component. The coaxial constraint can be applied to a cylindrical fit between a shaft and a bore. Note that Eqs. (8) and (11) do not ensure that two cylindrical faces will be in contact or that the shaft will fit into the bore without interference.

3.3 Ill-Determined systems generated from the equality constraints

In this section, we like to discuss the three classes of ill-determined systems generated from equality constraints. Consider the slot and key example in Fig. 5. Suppose the location of the slot is fixed so that six DOF of the key are the only variables. First, assume one coplanar-minus constraint between $f_{1,1}$ and $f_{2,1}$. This yields three independent equations in six variables as shown in Eqs. (3) and (7). Therefore three of the six

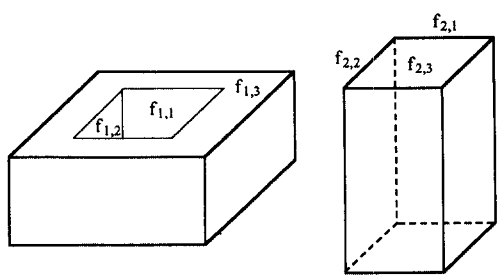


Fig. 5 Slot and key

variables are free. This is consistent because physically we can see that three DOF are left: two translations and one rotation. Clearly the system is underconstrained and there exists an infinite number of solutions.

Then let's add one more coplanar-minus constraint on $f_{1,2}$ and $f_{2,2}$. Since each coplanar-minus constraint yields three equations, now we have a total of six equations in six variables. However this is inconsistent because the key can still slide in the slot, implying that one of the six equations is redundant. Here again there is an infinite number of possible solutions.

Finally, one more constraint is added, a coplanar-plus constraint on $f_{1,3}$ and $f_{2,3}$, thus removing any DOF for the key with respect to the slot. Since the coplanar-plus constraint also introduces three equations, we have a total of nine equations for six variables, of which three are redundant. Therefore, in this case, there will be an unique solution. From the example, we can conclude that even though equality constraints are specified consistently, there may still be redundancy in the equations.

It can be generalized that if there are no conflicting constraints and no overconstrained ones, then inferring locations from equality constraints can be divided into three classes:

1) no redundant with an infinite number of solutions
2) redundant with an infinite number of solutions
3) redundant with an unique solution

In general, most packaging problems fall into Classes 1 and 2. An algorithm exists for an underdetermined system that selects, from among an infinite number of solutions, the solution that has the minimum weighted deviation from initial conditions (Pabon, 1985; Assada and Slotine, 1985). However, this algorithm requires the assumption that there are no redundant equations. Therefore, the algorithm can be applied only to Class 1. Class 3 is a special case in packaging because no DOF remain. Extensive work has been done on this type of problems by Rocheleau and Lee (1987).

In the following chapter an algorithm is proposed to treat all the three classes in the above. To the other classes of problems not introduced in the above, the algorithm can also be applied such as pointing out conflicting equations, if any, among the system of equations. In addition it can generate a solution from overconstrained systems in the least-square sense. Details on these are omitted for the sake of brevity and will be the subject of a separate paper.

4. Minimum Weighted Deviations

First, we introduce a case in which a system of equations is linear and then extend it for a nonlinear case (Note that equations generated form equality constraints are nonlinear). Our derivation extensively uses the concept of SVD (Strang, 1988; Press et. al., 1986).

Suppose that the following system of equations has been derived from equality constraints

$$\boldsymbol{f}(\boldsymbol{r})=0 \tag{12}$$

where $\boldsymbol{f}$ is an m-dimensional function vector, and $\boldsymbol{r}$ is an n-dimensional variable vector, representing location variables of movable components. Since $n>m$, i.e., $\boldsymbol{f}$ is an underconstrained system, it has an infinite number of solutions so that we may choose one of them by imposing some criteria. For example, as shown in Fig. 6 (a), if one coplanar-minus constraint is specified between the disk drive and the front side of the computer cabinet, then the system of equations generated from the coplanar-minus constraint is underconstrained and all locations of the disk drive shown with the dotted lines in Fig. 6 (b) can

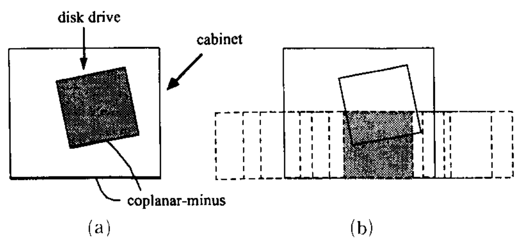


Fig. 6 Many locations of the disk driver from one coplanar-minus constraint.
Dotted squares illustrate many locations that satisfy the constraint.

be a solution. To choose one solution, it may be desirable to apply the criterion that the solution be minimally deviated from the initial location of the disk drive. With this criterion, the solution is the location of the disk drive shown with solid lines in Fig. 6 (b). This criterion is ideal for a computer-based packaging tool because, for example, if the designer sets a component's initial location near a mating surface, then the algorithm automatically places the component at a minimally deviated location where the equality constraint is satisfied (Later for detail packaging the minimally deviated location could be used as the initial conditions and further optimizd along the horizontal direction while maintaining equality constraints (Kim and Gossard, 1991)).

The criterion is modified and formulated in the more general form: find the solution that minimizes the weighted deviation from the initial locations of the movable components, $\boldsymbol{r}^0$, i.e., find the solution that minimizes

$$|\boldsymbol{W} \cdot (\boldsymbol{r}-\boldsymbol{r}^0)| \tag{13}$$

where $\boldsymbol{W}$ is a diagonal matrix of weighting factors. It is very important to introduce the weighting factors into packaging in order to increase the convergence of a pseudo Newton-Raphson method. Since the location vector $\boldsymbol{r}$ includes translation and rotation variables, the elements of the vector have two different units; length (meter) and angle (radian). To improve the convergence, we need to rescale one unit with respect to the other unit such that an optimal ratio between two weighting factors for translation and rotation variables is

$$\frac{w_r}{w_t} \approx \frac{2\pi}{\text{order of design space's size}} \tag{14}$$

where w_r and w_t are weighting factors for rotation and translation variables respectively.

In linear cases, the function vector $\boldsymbol{f}$ of Eq. (12) can be written as

$$\boldsymbol{A} \cdot \boldsymbol{r} - \boldsymbol{b} = 0 \text{ or } \boldsymbol{A} \cdot \boldsymbol{r} = \boldsymbol{b} \tag{15}$$

where $\boldsymbol{A}$ is an $m \times n$ constant matrix, and $\boldsymbol{b}$ is an m-dimensional constant vector. Eq. (15) can be rewritten by subtracting $\boldsymbol{A} \cdot \boldsymbol{r}^0$ from both terms and yields

$$\boldsymbol{A} \cdot (\boldsymbol{r}-\boldsymbol{r}^0) = \boldsymbol{b} - \boldsymbol{A} \cdot \boldsymbol{r}^0 \tag{16}$$

Now Eq. (16) can be modified by inserting $\boldsymbol{W}^{-1} \cdot \boldsymbol{W}$ between $\boldsymbol{A}$ and $(\boldsymbol{r}\text{-}\boldsymbol{r}^0)$ since $\boldsymbol{W}^{-1} \cdot \boldsymbol{W} = \boldsymbol{I}$ identity matrix and results in

$$\boldsymbol{A} \cdot \boldsymbol{W}^{-1} \cdot \boldsymbol{W} \cdot (\boldsymbol{r}-\boldsymbol{r}^0) = \boldsymbol{b} - \boldsymbol{A} \cdot \boldsymbol{r}^0 \tag{17}$$

which can be expressed in the more compact form

$$\boldsymbol{A}' \cdot \boldsymbol{r}' = \boldsymbol{b}' \tag{18}$$

where

$$\boldsymbol{A}' = \boldsymbol{A} \cdot \boldsymbol{W}^{-1},\ \boldsymbol{r}' = \boldsymbol{W} \cdot (\boldsymbol{r}-\boldsymbol{r}^0) \text{ and}$$
$$\boldsymbol{b}' = \boldsymbol{b} - \boldsymbol{A} \cdot \boldsymbol{r}^0 \tag{19}$$

The theory of SVD says that any $m \times n$ matrix whose number of rows m is greater than or equal to its number columns n, can be factored into (Press et. al., 1986):

$$\boldsymbol{A}' = \boldsymbol{U} \cdot \boldsymbol{D} \cdot \boldsymbol{V}^T \tag{20}$$

where $\boldsymbol{U}$ is an $m \times n$ column-orthogonal matrix, $\boldsymbol{D}$ is an $n \times n$ diagonal matrix with positive or zero elements and V is an $n \times n$ orthogonal matrix. Since the Press et. al.'s SVD algorithm (1986) hired for this study requires $m \geq n$, before computing Eq. (20), if $m < n$, then augmentation must be done on $\boldsymbol{A}'$ until it is filled up to be square with rows of zeros underneath its m non zero rows. Similary augmentation is required on $\boldsymbol{b}'$ with zeros. It should be noted that this augmentation process is essentially interpreted as that$(n-m)$ redundant equations are placed into the original system of Eq. (18). These redundant equations will be numerically taken care of during the computation of the pseudoinverse of $\boldsymbol{A}'$ (Press et. al., 1986; Strang, 1988). For simplicity, during the rest of the discussion, let's assume that Eq. (19) already represents the augmented expression. Suppose the SVD of $\boldsymbol{A}'$ can be expressed as Eq. (20) then the pseudoinverse of $\boldsymbol{A}'$ is

$$\boldsymbol{A}'^{+} = \boldsymbol{V} \cdot \boldsymbol{D}^{+} \cdot \boldsymbol{U}^T \tag{21}$$

where $\boldsymbol{D}^{+}$ is the pseudo inverse of augmented matrix $\boldsymbol{D}$. One of powerful features of SVD is that $\boldsymbol{r}'^{+}$ whose magnitude is the minimum length can be simply obtained with the pseudoinverse $\boldsymbol{A}'^{+}$ in Eq. (21) and given by

$$\boldsymbol{r}'^{+} = \boldsymbol{A}'^{+} \cdot \boldsymbol{b}' = \boldsymbol{V} \cdot \boldsymbol{D}^{+} \cdot \boldsymbol{U}^T \cdot \boldsymbol{b}' \tag{22}$$

Substituting the primed variables with expression (19) gives r^+, with which $|W(r^+-r^0)|$ is minimum

$$r^+=r^0+W^{-1}\cdot V\cdot D^+\cdot U^T \cdot(b-A\cdot r^0) \tag{23}$$

From the derivation for the linear case it can be seen that SVD eliminates the problem of underconstrained systems as well as redundant equations. At this point it should be pointed out that Eq. (23) also provides the minimum length solution for overconstrained systems (Press et. al., 1986; Strang, 1988). In this case the augmentation for A' is not needed.

Second, with the concept of SVD, the nonlinear case can be solved iteratively using a pseudo Newton-Raphson method. The function vector is first linearized at the current locations of the iteration no. k, and a solution to the linearized problem is found. In this case, f is linearized as

$$f(r_k^+ + \Delta r^+) = f(r_k^+) + J_k\cdot\Delta r^+ \tag{24}$$

where r_k^+ is the current locations of the movable component at iteration no. k and Δr^+ is the increment vector defined as

$$\Delta r^+ = r_{k+1}^+ - r_k^+ \tag{25}$$

and J_k is the $m\times n$ Jacobian matrix evaluated at r_k^+, i.e.,

$$j_{ki,j}=\left(\frac{\partial f_i(r)}{\partial r_j}\right)_{r=r_{k+}} \quad i=1,m\ ;\ j=1,n \tag{26}$$

Unlike the linear case, the objective for the nonlinear case is to find the increment vector Δr^+ for which

$$f(r_k^+ + \Delta r^+)=0 \text{ or } J_k\cdot\Delta r^+ = -f(r_k^+) \tag{27}$$

and minimizes

$$|W\cdot(r_{k+1}^+ - r^0)| \tag{28}$$

Now the equation has the same form as linear cases and therefore, using SVD, we obtain the new current location, r_{k+1}^+, which minimizes $|W\cdot(r_{k+1}^+ - r^0)|$

$$r_{k+1}^+ = r^0 + W^{-1}\cdot V\cdot D^+\cdot U^T \cdot[-f(r_k^+)+J_k(r_k^+ - r^0)] \tag{29}$$

where, similar to the linear case, V, D^+ and U^T are computed from J_k with SVD. The vector function f is then linearized at the new current locations and the method is repeated. The iterative process ends when the norm of the increment vector Δr^+ is less than the specified tolerance.

Similar to the most of iteration approaches based on the basic Newton-Raphson method, the initial guess of r^0 in Eg. (28) influences not only the convergence rate of the algorithm but also the convergence success to the global minimum.

5. Results

The algorithm introduced in the paper was implemented as a prototype system on IRIS-4D 70/GT. The system allows the users to interactively specify equality constraints on solid models and returns the minimally deviated locations.

Figure 7 illustrates one of Class 1 examples as discussed in section 3.3. In this case, there is one coplanar-minus constraint for each bore and base components. Therefore there are three independent equations for six variables, an underconstrained system. The bore is located on the base by minimally deviating from the initial location, as shown in the lower right of Fig. 7.

Figure 8 shows Class 2 as discussed in section 3.3. There is one coaxial constraint specified to the centerlines of the bore and shaft components. In addition, one coplanar-minus constraints is added between the faces of the shaft and the bore. Therefore total seven equations can be generated:

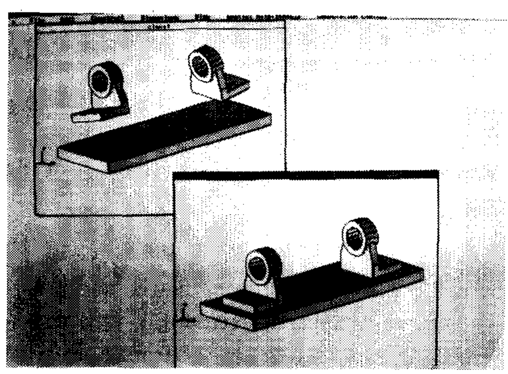


Fig. 7 Locating the bores. An example of Class 1

four from the coaxial and three from coplanar-minus. However still one DOF is left because the shaft can slide through the holes. In other words, two out of seven equations are redundant so that still an underconstrained system. The result is shown in the lower right of Fig. 8.

Finally Fig. 9 shows one of the Class 3 examples. Two coplanar-minus and one coaxial constraints are given to each bore and bolt components. In this case, total ten equations are generated from the constraints and four of them are redundant. This is a complete system in a sense that the unique solution may exist. Therefore the concept of minimum weighted deviation is not necessary but the pseudo Newton Raphson method can be still applied. The result is displayed in the lower right of Fig. 9. The computation time used to solve each of the three examples ranged from one to five CPU seconds on IRIS-4 D 70/GT.

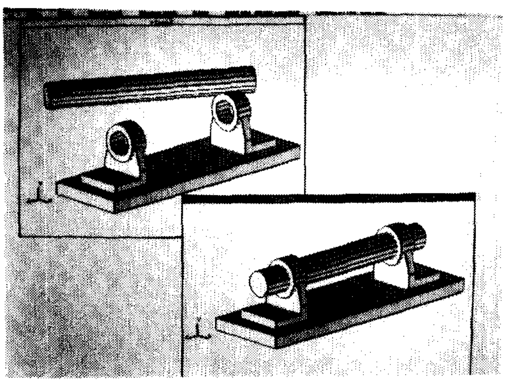


Fig. 8 Locating the shaft. An example of Class 2

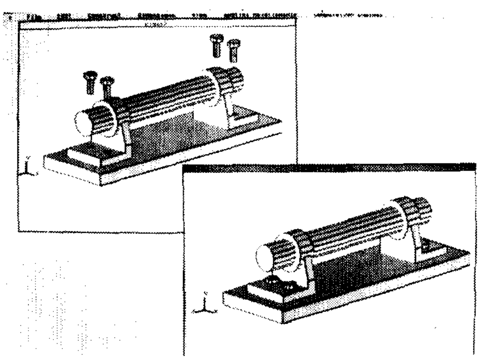


Fig. 9 Locating the bolts. An example of Class 3

6. Conclusions

Several spatial relationships in packaging were introduced and formulated as equality constraints, which resulted in ill-determined systems. To treat these ill-determined systems of equations a pseudo Newton-Raphson method was developed by utilizing the concept of singular value decomposition. To select one solution from the system, the concept of minimally weighted deviation was used. The solution, which is the location of each component, may be the final location in an assembly or can be served as an initial condition for further optimization in packaging.

References

Asada, H. and Slotine, J., 1985, *Robot Analysis and Control*, John Wiley and Sons.

Gossard, D., Zuffante, R. and Sakurai, H., 1988, "Representing Dimensions, Tolerances, and Features in MCAE Systems," *IEEE Computer Graphics & Applications*, Vol. 8, No. 2, pp. 51~59.

Kim, J. and Gossard, D., 1991, "Reasoning on the Location of Components for Assembly Packaging," *Transactions of the ASME Journal of Mechanical Design*, Vol. 113, December, pp. 402~407.

Light, R. and Gossard, D., 1983, "Variational Geometry: A New Method for Modifying Part Geometry for Finite Element Analysis," *Computers and Structure*, Vol. 17, No. 5~6, pp. 903~909.

Mullineux, G., 1987, "Optimization Scheme for Assembling Components," *Computer-Aided Design*, Vol. 19, No. 1, pp. 35~40.

Pabon, J., 1985, "Basic Steps Towards Computer Aided Scaling of Assemblies," M.S. Thesis, M.I.T., May.

Paul, R. P., 1982, *Robot Manipulators*, MIT Press.

Press, W., et. al., 1986, Numerical Recipes in C, *Cambridge University Press, Cambridge, England.*

Rocheleau, D. and Lee, K., 1987, "System for

Interactive Assembly Modelling,"*Computer-Aided Design*, Vol. 19, No. 2, pp. 65~72.

Serrano, D. and Gossard, D., 1987, "Constraint Management in Conceptual Design," *2nd Int. Conf. of the Application of AI in Engineering*, Boston, MA, August.

Strang, G., 1988, *Linear Algebra and its Applications*, Harcourt Brace Jovanovich, Inc.

Witkin, A., Kurt, F., and Alan, B., 1987, "Energy Constraints on Parameterized Models," *Computer Graphics*, Vol. 21, No. 4 July, pp. 225~232 (Proc. SIGGRAPH 1987).