

특징형상 위치 결정을 위한 형상 구속조건의 이용

김 성 환*, 이 건 우**

Using Geometric Constraints for Feature Positioning

Sung Hwan Kim*, Kunwoo Lee**

ABSTRACT

This paper describes the development of new feature positioning method which embedded into the top-down assembly modeling system supporting conceptual design. In this work, the user provides the geometric constraints representing the position and size of features, then the system calculates their proper solution. The use of geometric constraints which are easy to understand intuitively enables the user to represent his design intents about geometric shapes, and enables the system to propagate the changes automatically when some editing occurs. To find the proper solution of given constraints, the Selective Solving Method in which the redundant or conflict equations are detected and discarded is devised. The validity of feature shapes satisfying the constraints can be maintained by this technique, and under or over constrained user-defined constraints can also be estimated. The problems such as getting the initial guess, controlling the multiple solutions, and dealing with objects of rotational symmetry are also resolved. Through this work, the feature based modeling system can support more general and convenient modeling method, and keeps the model being valid during modifying models.

Key Words: CAD System, Feature(특징형상), Geometric constraint(형상구속조건), Non-Manifold(비다양체)

1. 서 론

기계 분야에서의 CAD 시스템은 점, 선 등의 형상요소들에 관한 정보를 만들고 저장하는 와이어프레임(wire-frame) 모델러로부터 면 정보를 다룰 수 있는 서피스(surface) 모델러를 거쳐 부피를 가지는 물체를 완전하게 모델링 할 수 있는 솔리드(solid) 모델러로 발전하여

왔다. 이렇게 저장 정보의 단계가 높아지면서 처음에는 도면 작성 등의 비교적 단순한 작업 지원 도구에서 이제는 물성치 계산, 가공경로의 자동생성, 유한요소의 자동 생성 등의 고도화된 분야에서의 설계지원 도구로 그 활용 영역이 확대되었다. 하지만 이러한 솔리드 모델링 시스템의 저장 정보는 형상요소라는 낮은 차원으로만 이루어져 있어 형상에 관해 설계자가 의도했던 공학적 의미는 수용

* 대우공업전문대학 기계과
** 서울대학교 기계설계학과

하지 못하는 단점이 있다. 예를 들어 모델에 존재하는 관통구멍(through hole)은 모델 내의 꼭지점, 모서리 및 면들이 만들어낸 우연한 결과일 뿐 솔리드 모델 자체로는 이의 존재를 말해줄 수 없으며, 더욱이 여기에 다른 축이 결합될 수 있다던가 드릴링으로 가공될 수 있다거나 하는 정보는 담지 못한다. 이러한 공학적 의미의 유실은 설계에서 생산까지를 자동화하려는 CIM 환경에서 모델 표현 방법으로서의 장애요소이며, 이를 해결하려는 노력의 일환이 특징형상(feature)을 사용한 모델링 시스템이다.

특징형상은 Shah⁽¹⁾에 따르면 "설계, 해석 및 생산에 있어서 물체의 특성을 유추하는데 사용될 수 있는 구성요소"라 정의되며 따라서 적용분야에서 공학적 의미가 있는 모든 정보는 특징형상으로 규정될 수 있다. 한편 기존의 연구자^(1~4)들에 따르면 특징형상을 다루는 시스템의 기본 구사항으로는 특징형상의 사용자 정의기능, 특징형상 인스턴스의 생성, 삭제, 수정 및 이동을 통해 솔리드 모델을 만들어 줄 수 있는 기능, 특징형상 인스턴스를 손쉽게 위치시키고 그들간의 관계를 정의할 수 있는 적절한 사용환경, 특징형상의 적합성(validity)의 검증기능 등이 있다. 이중에서 특히 특징형상의 위치 결정 작업은 원활한 모델링 작업을 위하여 중요한 기능이며 이의 성공적인 해결이 사용환경의 개선에 관건이라는 지적⁽²⁾이 있으나 아직 이에 관한 연구는 활발하지 못한 실정이다.

한편 형상구속조건(geometric constraint)이란 형상 요소들 사이에 만족되어야 할 구속조건이다. 두 모서리가 정렬되어야 한다던가, 두 면이 정렬되어야 한다던가 하는 것 등이 그 예이며, 형상요소간의 치수(dimension) 조건 등도 이에 포함된다. 이러한 형상구속조건의 사용과 이를 만족시키는 형상요소의 위치, 방향, 치수, 구속 자유도 등을 찾는 문제는 GCSP(Geometric Constraint Satisfaction Problem)라 불리는데 CAD 시스템 내에서 이것이 사용되는 분야로는 구속조건 기반의 설계 시스템, 조립체 모델의 생성 및 표현, 자유도에 기초한 기구의 설계 등을 꼽을 수 있다.⁽⁵⁾

Fig. 1은 사각평판 형상의 위치를 지정하기 위해 사용자로부터 부여된 형상구속조건의 예를 보여주고 있다. 이러한 형상구속조건은 사용자 환경으로 사람이 다루기가 쉽고 직관적으로 이해하기 편리하다는 장점이 있을 뿐 아니라 그 자체가 형상의 위치 및 치수 결정에 대한 설계자의 의도를 나타내므로 설계의도를 저장하는 도구가 된다. 더불어 만일 일부 형상에 수정이 있을 때 이를 다른 형상에 전파시킬 때 이용할 수 있음으로 해서 형상에 관한 설

계의도를 항상 유지할 수 있게 해주는 유용한 도구가 될 수 있다.

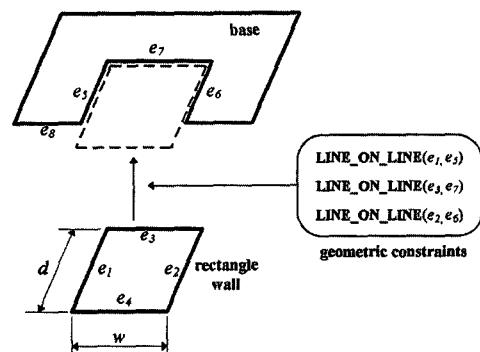


Fig. 1 Geometric constraints on the rectangle wall

본 연구에서는 특징형상 모델링 시스템이 일반적이고 확장된 특징형상 위치 지정 사용자 환경을 제공할 수 있게 하고 또 형상 설계의도가 명확히 저장되고 유지되는 특성을 가지게 하기 위하여, 형상구속조건을 도입한 특징형상 위치 결정 모듈을 개발하였다. 이 논문에서는 이러한 특징형상 위치 결정 모듈이 가져야 할 요구조건과 이 때 발생하는 GCSP의 풀이 방법에 대해 기술한다.

2. 기본개념

본 연구는 개념설계를 지원하는 특징형상 기반의 조립체 모델링 시스템을 개발⁽⁶⁾하려는 노력의 일환으로 수행되었다. 이 모델링 시스템에서의 모델링 과정을 개괄적으로 보면 다음과 같다.

[과정 1] 평판을 두께없는 면, 축을 직선으로 표현하는 등의 축약모델 만들기

[과정 2] 축약 특징형상에 부피를 부여하여 상세화 하기

[과정 3] 상세화된 특징형상들을 불리안 연산을 사용해 부품별로 합쳐진 모델 만들기

이 중 과정 1에서 사용자는 축약 모델에 해당하는 특징형상을 인스턴스하여 기존 형상에 대한 형상구속조건을 부여하여 적절히 위치시킨다. 따라서 이것이 특징형상 위치 결정 모듈이 사용되어야 할 첫 번째 경우이다. 한편 과정 2 또는 3까지 진행된 후 사용자는 특징형상의 크기나 위치를 수정할 수 있는데 시스템은 이 때 그 결과로 영향 받는 축약 특징형상의 수정된 크기 및 위치를 다시 계산하고 나머지 과정을 반복할 수 있게 하여야 하며, 이때가

형상구속조건의 해가 구해져야하는 두 번째 경우이다. 위의 두 경우를 각각 초기 배치 작업(initial positioning)과 재배치 작업(re-positioning)이라 부르기로 하자. Fig. 1은 초기 배치 작업의 하나의 예이며, Fig. 2는 기준 형상인 base의 흠 부분이 넓어졌을 때 사각평판(rectangle wall)의 수정된 형상을 구하는 재배치 작업의 예를 보여준다.

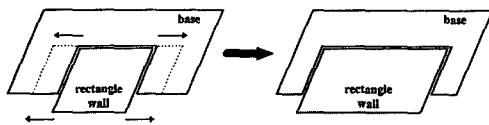


Fig. 2 Re-positioning

그런데 이들 예를 살펴보면 사용자는 Fig. 1에 표시된 3개의 구속조건으로 형상이 완전히 구속되기를 항상 기대할 것이지만, 시스템의 입장에서는 여러 가지 경우가 발생함을 알 수 있다. 우선 초기 배치 작업에서 사각평판의 치수 w 가 흠의 너비와 같을 때는 단순한 위치 이동만으로 모든 구속 조건을 만족시킬 수 있다. 반면에 그렇지 않을 때는 치수의 변화를 동반하며 이동해 가야 한다. 하지만 이 두 경우에 항상 d 의 치수는 변화되지 않아야 한다. 치수 w 와 d 가 다루어지는 방법에서 이렇게 차이가 있는 것은 사용자가 부여한 구속조건에 의해 치수가 영향을 받는가 그렇지 않은가에 따라 정해진다. 즉 치수 w 는 사용자가 부여한 구속조건을 만족시키려면 무시되기도 해야 하는 반면 치수 d 는 사용자 구속조건과 아무 관계가 없으므로 항상 유지되어져야 한다. 한편 재배치 작업의 경우는 항상 부분적인 치수 변화를 유발시키는데 최종 위치에 이미 가까이 있으므로 임의로 정해지는 비교적 큰 이동은 없다.

이와 같이 특징형상의 위치 결정문제는 단순히 강체(rigid body) 이동을 위한 변환행렬(transformation matrix)을 구하는 문제가 아니며 경우에 따라 형상변이가 수반되어야 한다. 그런데 형상변이가 일어날 때 그 특징형상이 가지는 본래의 특성까지 깨어져서는 안된다. 예를 들어 사용자가 부여한 구속조건들을 만족시키려다 보니까 사각평판의 네 꼭지점이 동일 평면 위에 놓이지 않게 되었다거나 꼭지각이 직각이 아니라던가 하는 결과가 나온다면 이는 특징형상을 선택할 때 사용자가 가졌던 모

델링 의도를 위배한 것이 되므로 이때는 사용자 구속조건이 잘못된 것으로 판단하여 시스템은 이를 알려줘야 한다.

이와 같은 문제를 해결하기 위하여 본 연구에서는 풀어야 할 변수를 특성점⁽⁶⁻⁸⁾ (characteristic point)으로 하고 각 형상 구속 조건식은 이에 대한 함수 식으로 정의한다. 특성점이란 물체의 형상을 대표하는 점으로 특성점의 위치를 정함으로써 물체의 형상을 완전히 정의할 수 있는 점들이다. Fig. 3은 본 연구에서 사용된 특징형상들과 각각의 특성점을 보여준다. 앞서 설명한 바와 같이 본 연구는 개념설계를 지원하는 특징형상 기반의 조립체 모델링 시스템을 목표하고 있으므로 선이나 면만으로 이루어진 비다양체⁽¹⁰⁻¹²⁾ 특징형상들을 포함하고 있으며, 자체 개발된 비다양체 모델링 시스템⁽¹³⁻¹⁵⁾을 바탕으로 프로그램되었다.

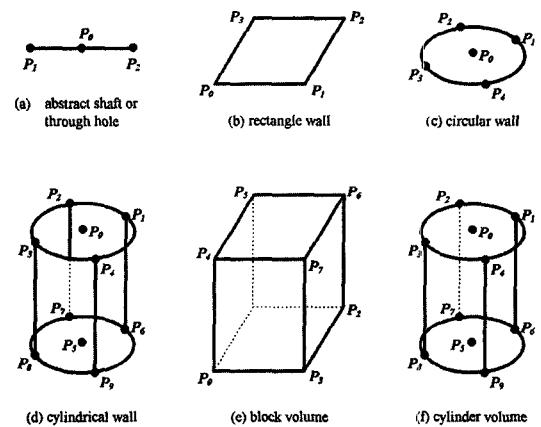


Fig. 3 Features and their characteristic points

형상 구속조건으로부터 특성점에 관한 조건식들이 얻어지면 비선형 연립방정식의 형태를 띠게 되는데 이는 수치적 방법을 통해 풀어지게 된다. 그런데 이를 위해서는 앞서 설명한 바와 같이 사용자 정의 형상구속조건을 위배하는 초기 치수조건은 무시되어야 하고, 특징형상의 본래 성질을 위배하는 사용자 정의 구속조건은 감지되어야 하는 등 방정식간의 충돌여부를 판별해 낼 수 있는 방법이 요구되며 이의 해결 방법으로 방정식의 우선 순위에 따른 선택적 풀이 방법이 고안되었다.

3. 관련연구

특징형상에 관한 기존 연구^(1,3,4)에서는 구비해야 할 특징형상의 종류가 유한하지 않음으로 해서 사용자에 의해 새로운 특징형상이 정의될 수 있어야 하고 추가 정의된 특징형상도 미리 정의된 특징형상들과 동일한 프로그램에 의해 동일한 방법으로 다루어져야 하므로 특징형상을 객체(object)로 하는 객체지향(object oriented) 시스템이 되어야 하였다. 이에 따라 특징형상 객체가 가져야 할 성질(property)로 위치결정 변수와 위치결정 방법을 포함시켰다. 예를 들어 Fig. 4 (a)의 axial through hole은 위치 결정을 위해 e 와 θ 를 입력받아야 한다는 사실이 이미 특징형상 내에 정의되어져 있다. 그러나 이러한 방법은 특징형상을 위치시키는 방법을 제한하는 결과를 낳는다. 더욱이 Fig. 4의 (b)와 같은 관통구멍은 단지 위치시키는 방법의 차이 때문에 다른 특징형상으로 정의되어야 하고 이렇게 세분된 특징형상의 정의는 그만큼 사용자에게 많은 부담을 줄 수 있다. 특징형상 위치 결정에 관한 다른 연구로는 Gossard⁽¹⁶⁾ 등이 제안한 RPO(Relative Position Operator), Wang⁽¹⁷⁾ 등이 제안한 자료구조의 Cnode 등이 있다. 그러나 이들은 특정 특징형상에만 적용 가능하거나 치수조건만을 염두에 두고 강체이동만 가능한 등 사용자에게 자유로운 모델링 환경을 제시하고 형상에 관한 설계의도를 수용하려는 노력과는 거리가 있다.

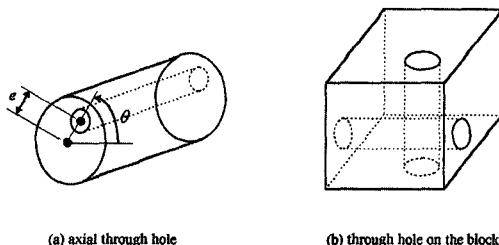


Fig. 4 Two types of through hole

한편 형상구속조건을 사용하는 한 분야는 주로 조립체 내에서 부품의 상호 위치관계를 기술하고 계산하는 문제인데, Rossignac⁽¹⁸⁾, Bruderlin⁽¹⁹⁾, Gossard와 Andrews⁽²⁰⁻²¹⁾, Rocheleau와 이건우⁽²²⁾, 김성환과 이건우⁽²³⁾, Srikanth와 Turner⁽²⁴⁾, Kramer^(5,25-26) 등의 연구가 있다. 이를 연구는 접근 방법에 따라 수치적(numeri-

cal) 방법과 기호적(symbolic) 방법으로 나누어진다. 그러나 이들의 공통점은 형상을 강체로 취급하여 물체 형상의 변화는 허용치 않고 변환행렬만을 구한다는 것으로 따라서 치수조건을 구속조건으로 포함시킬 수 없다.

형상구속조건을 사용하는 또 다른 한 분야는, 치수를 형상 구속조건으로 삼아 물체의 형상을 결정하는 변이형상(variational geometry) 분야이다. 이 분야의 연구는 Aldefeld⁽²⁷⁾, Sunde⁽²⁸⁻²⁹⁾ 등의 AI 접근방법이 있고, Light, Lin, Gossard⁽⁶⁻⁸⁾ 등의 수치 해석적 접근방법이 있다. 이 중 수치 해석적 방법은 다양한 구속조건과 여러 종류의 물체형상을 다룰 수 있지만 초기값 등에 따른 안정성과 계산시간 면에서 상대적으로 취약하다.

본 연구는 특성점의 위치를 수치 해석적으로 계산한다는 점에서 변이형상 기법의 수치 해석적 접근방법과 유사하지만, 변이형상기법은 초기위치로부터 비교적 크지 않은 형상변화를 구하는 것과는 달리 큰 변이를 다루어야 하고, 치수 구속조건의 사용여부를 자동 판단해야 하는 등의 문제들을 해결해야 한다는 것이 다르다.

4. 구속조건식

앞서 언급한 바와 같이 특징형상의 크기와 위치를 정하는 구속조건식은 특성점들의 좌표 값에 대한 함수 형태인데, 본 연구에서는 이를 점, 직선, 원호, 평면 등의 기하요소간의 관계를 정의하는 기호로 표현한다. 예를 들면 구속 조건 POINT_ON_GPOINT (P, P_o)는 동점 $P(x, y, z)$ 가 주어진 점 $P_o(x_o, y_o, z_o)$ 와 같은 위치에 있어야 한다는 조건이다. 이 조건은 시스템에 다음과 같은 세 개의 조건식을 만들어 준다.

$$\begin{aligned}x - x_o &= 0 \\y - y_o &= 0 \\z - z_o &= 0\end{aligned}\quad (1)$$

또 POINT_DIST_GPLANE (P, P_o, n, D)은 동점 $P(x, y, z)$ 로부터 주어진 평면까지의 거리가 D 가 되어야 한다는 조건이며 평면 위의 한 점이 $P_o(x_o, y_o, z_o)$, 평면의 법선 벡터가 $n(a, b, c)$ 일 때 다음의 조건식을 만든다.

$$\sqrt{\{a(x - x_o) + b(y - y_o) + c(z - z_o)\}^2} - D = 0 \quad (2)$$

한편 LINE_HAS_GDIR (P_1, P_2, v) 과 같은 구속조건은 동점 $P_1(x_1, y_1, z_1)$ 과 $P_2(x_2, y_2, z_2)$ 로 이루어진 직선이 주어진 벡터 $v(a, b, c)$ 와 같은 방향을 가져야 한다는 조건으로 다음과 같은 세 개의 조건식을 만든다.

$$\begin{aligned} b(x_1 - x_2) - a(y_1 - y_2) &= 0 \\ c(y_1 - y_2) - b(z_1 - z_2) &= 0 \\ c(z_1 - z_2) - c(x_1 - x_2) &= 0 \end{aligned} \quad (3)$$

그런데 위의 식 (3)은 세 개중 사실은 두 개만이 서로 독립적이다. 따라서 일반적으로는 이를 중 임의로 선택된 두 식만으로도 원하는 구속이 모두 이루어지지만 모든 경우에 그렇지는 않다. 예를 들어 벡터 v 가 z 축과 평행하면 첫 번째 식은 구속조건식으로의 의미를 잃게 된다. 하지만 5장에서 설명할 방정식의 풀이방법은 이와 같이 적절한 조건식을 선택해야 하는 문제를 스스로 해결할 수 있으므로 일단은 세 식을 모두 조건식으로 채용해 둔다. 본 연구에서는 이와 같은 구속조건들이 약 20개정도 프로그램 되었다.

한편 위의 구속조건들로 위치가 정의된 특성점이 때로는 복수 개의 가능한 해를 가질 수 있다. 예를 들어 Fig. 5에서와 같이 주어진 직선 위에 있어야 하고 주어진 평면으로부터의 거리가 d 라는 구속조건을 가지고 있는 점 P 는 자유도가 완전히 구속되고 따라서 조건식은 수치적으로도 완전하다. 그러나 실제로 존재하는 해는 그림에 보는 바와 같이 두 개이며 이를 수치적으로 풀었을 때 어느 해를 얻게 될지는 초기치(initial guess)에 의해 전적으로 좌우된다. 한편 사용자는 이를 중 특정한 해를 원할 수 있으므로 시스템은 이를 제어할 수 있어야 한다.

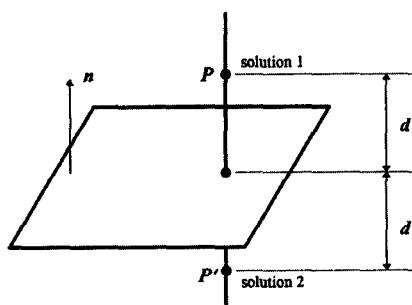


Fig. 5 Multiple solutions

본 연구에서는 사용자가 위와 같은 복수 해를 제어할 필요가 있을 때는 추가 조건식을 부여할 수 있게 하였다. 이러한 추가 조건식은 수치적으로 완벽한 식의 세트에 잉여식을 첨가시키는 것이며 이것은 5장에서 설명될 초기치 결정을 위한 전 처리 작업 중에 사용된다. 그런데 이러한 복수 해의 제어 문제는 기존의 변이형상(variational geometry) 기법에서는 발생하지 않았었다. 왜냐하면 기존의 문제는 초기형상으로부터의 비교적 적은 변이를 구하는 것이므로 현재 위치를 초기치로 사용하면 되었기 때문이다.

위와 같은 경우 해를 제어하기 위해 사용하는 추가 구속조건은 POINT_INSIDE_GPLANE과 POINT_OUTSIDE_GPLANE 중 하나가 된다. Fig. 6처럼 평면의 법선 벡터의 반대쪽(inside)에 점을 위치시키고자 할 때는 POINT_INSIDE_GPLANE (P, P_o, n)을 선택해야 하며 이는 (4) 식으로 구현될 수 있다.

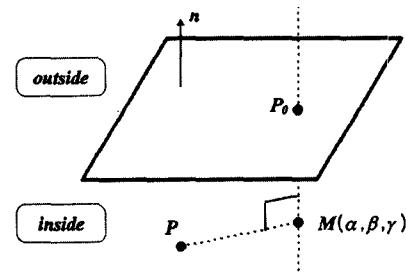


Fig. 6 POINT_INSIDE_GPLANE

$$\frac{\mathbf{M} - \mathbf{P}_o}{|\mathbf{M} - \mathbf{P}_o|} + \mathbf{n} = 0 \quad (4)$$

여기서 P_o 는 평면 위의 한 점, n 은 평면의 법선 방향 단위벡터이며 점 $M(\alpha, \beta, \gamma)$ 은 P_o 를 지나 n 의 방향을 가지는 직선 위로 P_o 로부터 내린 수선의 발이다. 이때 M 은 식 (5)와 같이 얻어지므로 구하는 조건식은 식 (6)과 같아 된다.

$$\begin{aligned} \alpha &= a\{a(x - x_o) + b(y - y_o) + c(z - z_o)\} + x_o \\ \beta &= b\{a(x - x_o) + b(y - y_o) + c(z - z_o)\} + y_o \\ \gamma &= c\{a(x - x_o) + b(y - y_o) + c(z - z_o)\} + z_o \end{aligned} \quad (5)$$

$$\begin{aligned} & \left\{ (\alpha - x_o) + a\sqrt{(\alpha - x_o)^2 + (\beta - y_o)^2 + (\gamma - z_o)^2} \right\}^2 + \\ & \left\{ (\beta - y_o) + b\sqrt{(\alpha - x_o)^2 + (\beta - y_o)^2 + (\gamma - z_o)^2} \right\}^2 + \quad (6) \\ & \left\{ (\gamma - z_o) + c\sqrt{(\alpha - x_o)^2 + (\beta - y_o)^2 + (\gamma - z_o)^2} \right\}^2 = 0 \end{aligned}$$

한편 이와 같이 자유도가 완전히 구속되었음에도 불구하고 생기는 복수 해를 제어하기 위해 마련된 추가 구속조건식은 두 가지 유형이 더 있다. 이 중 한 종류는 두 개의 특성점이 위치될 때, 그 방향이 주어진 벡터 방향과 일치한 형상과 반대의 형상으로 복수 해가 얻어지는 경우를 제어하기 위한 것으로, LINE_SAME_GDIR과 LINE_OPPOSITE_GDIR이다. 그리고 다른 한 종류는 세 개의 특성점이 한 평면에 놓이도록 위치될 때 세 점으로 만드는 두 벡터의 외적(cross product)의 방향이 평면의 법선 벡터와 같은 방향인 형상과 반대 방향인 형상으로 복수 해가 얻어지는 경우를 제어하기 위한 CROSS_SAME_GDIR과 CROSS_OPPOSITE_GDIR이다.

5. 구속조건식의 풀이방법

형상 구속조건으로부터 얻어진 비선형 연립방정식을 수치적으로 푸는데는 Newton-Raphson 반복법⁽³⁰⁾이 사용된다. 그러나 이를 사용하기 위해서는 두 가지 문제가 해결되어야 하는데, 첫 번째는 잉여(redundant)식과 충돌(conflict)식을 판별하여 제거하는 일이다. 형상 구속조건에 의해 만들어지는 구속조건식들은 사용자가 형상을 충분히 구속시키기 위해 중복되는 형상 구속조건을 입력하는 등, 대화식 시스템의 특성상 일반적으로 많은 잉여식을 포함하게 된다. 또 초기 치수 조건은 사용자에 의해 별도로 구속되지 않으면 반드시 사용되어야 하지만, 사용자가 정의한 구속조건을 위배하면 제거되어야 하는 충돌식이 된다. 이를 해결하는 방법이 잉여식 또는 충돌식을 검출하여 그 중 우선순위가 낮은 것을 제거하는 선택적 풀이방법이다. 선결되어야 할 두 번째 문제는 초기값(initial value)의 결정이다. 초기값의 결정은 선택적 풀이방법에서 수치 해법을 안정적으로 수렴하게 하는데도 필요하지만 복수 해를 제어하는 수단이 된다. 이제 조건식들의 종류와 그 우선 순위에 대해 설명하고 초기값 결정방법을 설명한 후 우선 순위에 따른 선택적 풀이방법에 대해 설명한다.

형상의 위치를 결정하기 위해 사용되어지는 구속 조건식들을 종류에 따라 우선 순위별로 정리하면 다음과 같다.

- 특징형상의 내재 구속조건 (Implicit constraint)
- 사용자 정의 구속조건 (User-defined constraint)
- 초기 치수 조건 (Default dimension constraint)
- 회전금지 구속조건 (Fix free rotation constraint)

특징형상의 내재 구속조건이란 그것이 어디에 위치되더라도 꼭 유지되어야 할 본연의 성질을 말한다. 이것들은 각 특징형상마다 자료구조 내에 정의되어 있으며 가장 높은 우선 순위를 갖는다. 예를 들면 사각 평판 특징형상에 대해서는 네 꼭지점이 한 평면 위에 놓여야 하고, 마주보는 변의 길이는 같아야 하며, 각 꼭지각은 직각이 되어야 한다는 것이다. 사용자 정의 구속조건은 사용자가 부여하는 구속조건으로 내재 구속조건보다 낮은 우선 순위를 갖는다. 초기 치수조건은 특징형상의 현재 크기를 표현하는 조건식으로 이것도 각 특징형상의 자료구조 내부에 정의되어 있다. 이 구속 조건은 사용자 정의 구속조건에 위배되면 버려지고 그렇지 않으면 사용되어야 하므로 다음 우선 순위를 갖는다.

마지막으로 회전 금지 구속조건은 회전 대칭인 특징형상의 경우에 사용된다. 예를 들어 Fig. 7의 원판은 축에 대해 회전대칭이므로 임의의 회전 각도를 가지도록 위치되어도 같은 형상을 의미하고, 따라서 사용자는 회전방향의 자유도는 구속하지 않으면서 임의의 각도로 위치되기를 기대하게 된다. 하지만 이렇게 얻어지는 구속조건식은 부족 구속 조건이 되고 풀이는 실패하게 된다. 이를 방지하기 위해 회전대칭을 갖는 특징형상은 대칭축에 대한 회전을 구속시키는 조건식을 자료구조 내에 가지고 있다. 그러나 만일 특정 각도로 위치되기를 원하여 사용자가 회

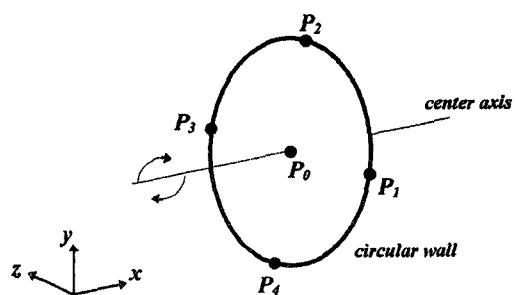


Fig. 7 Rotational symmetry

전 각도를 정하는 구속조건을 부여하면 회전 금지 구속조건은 무시되어야 하기 때문에 가장 낮은 우선 순위를 갖는다. 최종 위치가 Fig. 7과 같은 경우에 회전 금지 구속조건은 LINE_PERPENDICULAR_GDIR(P_o, P_p, z -axis)로 구현될 수 있다.

선택적 풀이방법의 수치 반복계산을 안정시키고 복수 해를 제어하기 위한 초기치는 원하는 해에 가까운 것일수록 바람직하다. 현재 시스템이 갖고 있는 구속조건은 위에서 설명한 4가지 종류와 4장에서 설명한 추가 구속조건인데 이 중에는 임여식 및 충돌식이 섞여 있을 것이며 식의

수가 변수의 수보다 훨씬 많다. 그런데 임여식과 충돌식이 어떤 것인지 아직 알지 못하며 충돌식이 있다면 모든 구속조건을 만족하는 해가 존재하지 않는다. 이러한 조건에서 근사 해를 얻기 위해서는 모든 조건식에 대해 최소자승(least square)법을 동반한 Newton-Raphson 반복법을 사용한다. 최소자승 반복법은 일반적으로 식의 수가 변수의 수보다 많아 과다 구속되었을 때 각 식의 오차를 최소로 하는 해를 구해주는데, 이런 전체 조건식의 근사 해는 최종 해에 가까운 좋은 초기조건이 된다.

한편 최소자승법으로 전체 구속조건을 풀어 근사 해를 구하는 전 처리 과정에서 모든 조건을 만족시키는 정확한 해가 얻어질 수도 있다. 이는 충돌식이 존재하지 않는 경우이며 사용자가 특정 형상의 크기를 구속하는 조건을 부여하지 않았을 때 발생할 수 있다. 이 경우에는 뒤의 선택적 풀이 방법을 수행할 필요 없이 풀이 과정이 종료된다.

전 처리에 의해 비교적 좋은 초기치를 얻게 되면 선택적 풀이방법을 수행하게 된다. 선택적 풀이 방법의 수행 순서는 다음과 같다. 우선 주어진 구속조건식들을 우선 순위에 따라 나열한 뒤 변수의 개수만큼 방정식을 취해 연립방정식을 구성한 후 풀이를 시도한다. 이때 얻는 결과는 세 가지로 나뉘어지는데, 첫째는 서로 임여가 되는 식 군(郡)이 얻어지는 경우이고, 둘째는 서로 충돌되는 식 군(郡)이 얻어지는 경우이며, 마지막은 해가 구해지는 경우이다. 정확한 해가 구해지면 풀이를 종료하게 되지만 만일 임여식 군이 얻어지게 되면 임여식 중 하나를 버리고 대신 사용되지 않은 식 중 앞선 우선 순위의 식을 취해 다시 연립방정식을 구성한다. 그리고 충돌식이 얻어진 경우라면 서로 충돌되는 식들 중 가장 우선 순위가 낮은 것을 버리고 역시 다음 식을 택해 연립방정식을 구성한다. 새로운 연립방정식이 구성되면 이를 다시 푸는 과정이 반복된다. 이 과정을 진행하는 동안 충돌된 식들 중 가장 낮

은 우선 순위의 식이 사용자 구속조건에 의해 부여된 것 일 경우는 사용자가 만족될 수 없는 구속조건을 부여한 경우이다. 이때 시스템은 사용자에게 이를 알려주고 새로운 입력을 요구한다. 그리고 마지막 조건식까지 모두 사용하였음에도 불구하고 여전히 임여 또는 충돌식이 감지되는 경우는 사용자가 부족 구속조건을 부여한 경우이다. 이 때도 시스템은 이를 사용자에게 알려준다.

한편 비선형 연립방정식을 푸는 과정에서 임여식 또는 충돌식을 감지되는 것은 매 반복과정 중 식 (7)과 같은 선형 연립 방정식을 풀 때이다.

$$\left[\frac{\partial f_i}{\partial x_j} \right]_k \{ \Delta x_j \}_k = \{ -f_i \}_k \quad (k \text{는 반복횟수}) \quad (7)$$

$$[J]_k \{ \Delta x_j \}_k = \{ R \}_k$$

만일 비선형 연립방정식 내에 임여 또는 충돌식이 존재한다면 위 식의 자코비안 행렬은 해가 대입되었을 때 특이(singular)해진다. 따라서 초기조건으로부터 반복법에 의해 해에 가까워지면 이를 검출할 수 있다. 즉 가우스 소거법으로 선형 연립방정식을 풀 때 어느 한 열(row)이 모

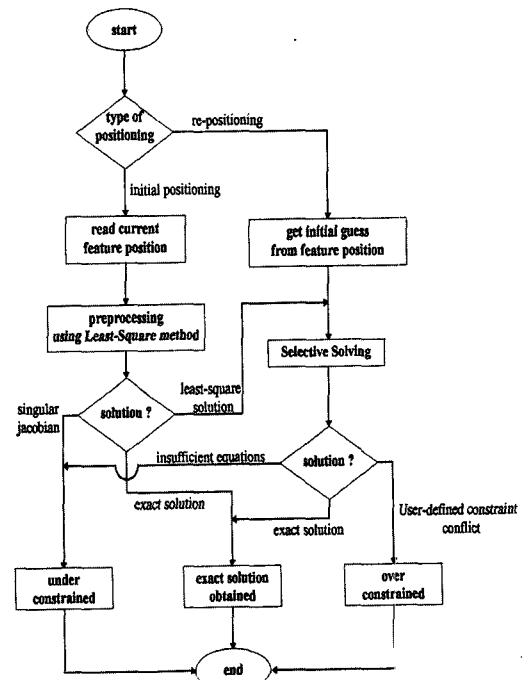


Fig. 8 Flow chart of equation solving

두 0이 되면 이 식과 이미 피봇(pivot)된 식들이 서로 일치 또는 충돌이다. 잉여식인지 충돌식인지의 판단은 그 열의 모든 요소가 0이 되었을 때 레지듀얼 f_i 도 0이 되었는지 아닌 지로 판단된다. 여기서 설명한 선택적 풀이 방법을 순서대로 나타내면 Fig. 8과 같다.

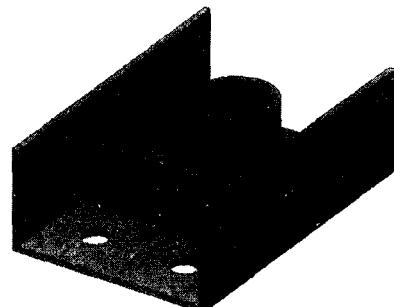
6. 적용예

본 연구는 개념설계를 지원하는 특징형상 모델러에 적용되었다. 특징형상 모델링 시스템에서는 필요 특징형상이 모델링 대상에 따라 다양하기 때문에 사용자가 새로운 특징형상을 자유로이 정의할 수 있도록 특징형상 정의 언어와 이를 읽어들일 수 있는 번역기의 개발이 바람직하나 이는 매우 방대한 작업이므로 추후 연구과제로 남기고 여기서는 C++ 언어의 베이스 클래스 및 파생 클래스로 정의하고 다뤄지는 방법들도 그 안의 멤버 함수로 정의하여 객체지향 프로그래밍을 지향하였다.

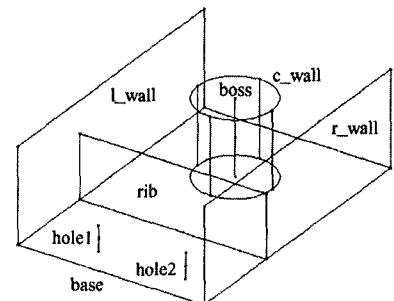
Fig. 9의 (a)는 모델링하고자 하는 물체의 모습이며, 이를 위한 축약모델이 (b)에 보여지고 있다. 이는 8개의 특징형상 인스턴스로 구성되고 있다. 일단은 공간에 배치된 모습일뿐 각각 서로 독립된 형상모델이다. 이들 축약특징형상들을 위치시키기 위해 특징형상 위치 결정 모듈이 사용되었는데, 이 중 l_wall의 경우는 4개의 모서리점의 좌표를 구속하기 위해 7개의 특징형상 내재 구속조건, 4개의 사용자 구속조건 그리고 2개의 치수 구속조건 등 13개의 구속조건이 부여되었으며 이로부터 발생된 22개 식의 비선형 연립방정식은 최소자승(least square)법에서 4번의 반복계산 후 수렴하였다. 이때 초기치는 원점에 인스턴스된 형상을 사용하였다. 나머지 특징형상들도 비슷한 수준의 계산을 거친 후 원하는 해가 얻어졌다. Fig. 9의 (c)는 초기 배치될 때의 구속조건에서 위치 변수 및 치수 변수만을 변화시킨 후 다시 계산한 재배치의 결과를 보여주고 있다. 이 경우에는 최소자승법을 거치지 않고 초기 배치된 상태를 초기치로 하여 선택적 풀이 방법이 수행되는데 l_wall의 경우 22개의 식 중 8개의 잉여식과 1개의 충돌식을 검출하여 제거한 후 해가 구해졌다.

7. 결 론

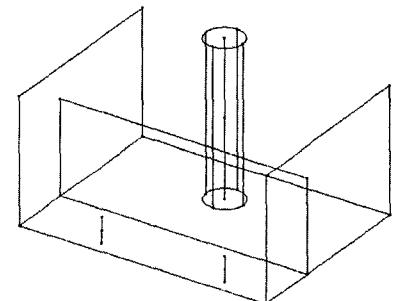
본 논문은 개념설계 단계를 지원하는 하향설계 방식의 조립체 모델링 시스템 개발의 일환으로 수행된 특징형상의 위치계산 모듈의 개발에 관해 기술하였다. 본 연구에



(a) model to be created



(b) features after initial positioning



(c) features after re-positioning

Fig. 9 Case study

서 특징형상의 위치 구속을 위하여 사용자는 보다 직관적으로 이해하기 좋은 형상구속조건을 사용하여 시스템은 이를 만족하는 특징형상의 위치를 계산하여 준다. 이때 설계자는 형상 구속조건을 통해 자신의 설계의도를 표현 할 수 있으며 시스템은 어떠한 수정이 일어났을 때 이를 항상 일관되게 유지하기 위하여 수정의 전파를 자동적으로 수행한다.

수정의 전파가 자동적으로 일어나기 위해서는 특징형상의 형상변이를 수반하는 재 위치 작업이 필요하며, 본 연구에서는 이를 선택적 풀이방법을 통해 구현하였다. 식의 우선 순위에 따른 선택적 풀이방법은 형상변이가 필요한 경우와 아닌 경우를 자동으로 감지하여 설계자가 의도한 특징형상의 크기와 위치를 결정해 줌으로써 모델의 정당성을 유지시켜 줄 뿐만 아니라 사용자의 부족 구속조건 및 과다 구속조건을 판별하여 준다. 한편 초기치 문제, 복수 해의 제어 및 회전대칭인 물체의 처리와 같은 기술적인 문제들이 해결되었다.

본 연구를 수행함으로써 사용자는 특징형상의 위치결정을 위해 시스템이 정해준 일방적인 방법 외의 자신의 의도를 표현할 수 있게 되었고 계속적인 수정작업 중에 설계의도가 항상 만족되어지는 시스템을 구축할 수 있게 되었다. 또한 사용자가 입력하는 구속조건들은 시스템 안에서 그 오류 및 타당성이 검증될 수 있으므로 설계자는 보다 효율적인 설계작업을 수행할 수 있다.

한편 본 연구는 현재 개념의 정당성을 확인하고 있는 단계이며 궁극적으로는 비교적 크고 복잡한 실제의 조립체 모델을 설계할 수 있는 모델링 시스템을 개발하는 것이다. 이것이 가능하기 위해서는 사용자가 필요한 특징형상을 정의할 수 있는 특징형상 정의 언어와 이에 따른 해석기(interpreter)의 개발이 필요하며 이를 통해 다양한 특징형상을 정의하고 구비하는 일이 필요하다. 또 특징형상 위치 결정 모듈이 실제 사용자의 입장에서 편리하게 사용될 수 있고 형상에 관한 설계의도를 표현할 수 있는 도구가 되기 위해서는 형상구속조건의 종류가 보다 다양하게 준비되어야 할 것이다.

참 고 문 헌

1. Shah, J. J., "Conceptual Development of Form Features and Feature Modelers," Research in Engineering Design, pp93-108, 1991.
2. Pratt, M. J., "Synthesis of an optimal approach to form feature modelling," Proceeding of the ASME International Computers in Engineering Conference, pp263-274, 1988.
3. Shah, J. J. and Rogers, M. T., "Expert form feature modelling shell," Computer Aided Design, Vol. 20, No. 9, 1988.
4. Shah, J. J., 1991, "Assessment of features technology," Computer Aided Design, Vol. 23, No. 5.
5. Kramer, G. A., 1991, "Using Degrees of Freedom Analysis To Solve Geometric Constraint Systems," Proc. Symposium on Solid Modeling Foundations and CAD/CAM Applications, Austin, Texas.
6. Light, R. and Gossard, D., 1982, "Modification of Geometric Models through variational geometry," Computer Aided Design, Vol 14, No 4.
7. Lin, V.C. 1981, "Three-dimensional variational geometry in Computer-aided design," Master Thesis, MIT.
8. Lin, V.C., Gossard, D.C. and Light, R.A.. 1981, "Variational Geometry in Computer-Aided Design," ACM Computer Graphics, Vol 15, No 3.
9. 김성환, 이건우, "하향식 설계방식을 지원하는 새로운 개념의 CAD 시스템," 대한기계학회 논문집, 제 19권, 제 7호, 1995.
10. Weiler, K. J., 1986, "Topological Structures for Geometric Modeling," PhD Thesis, RPI.
11. Choi, Y., 1989, "Vertex-Based Boundary Representation of Non-Manifold Geometric Models," PhD Thesis, Carnegie Mellon University.
12. Masuda, H., Shimada, K. Numao, M. and Kawabe, S., 1990, "A Mathematical Theory and Applications of Non-Manifold Geometric Modeling," Geometric Modeling for Engineering Applications, F.L. Krause and H. Jansen Editors), North-Holland.
13. 한영현, 이건우, 이상현, 김성환, 김영진, 배석훈, 안재홍, 이경진, "비다양체 모델을 지원하는 CAD 시스템 커널 개발을 위한 기반 구축," 산업공학, 제 8권, 제 3호, 1995년 9월.
14. 이상현, 이건우, "비다양체 형상 모델링을 위한 간결한 경계 표현 및 확장된 오일러 작업자," 한국 CAD/CAM 학회 논문집, 제 1권, 제 1호, 1996년 3월.
15. 김성환, 이건우, 김영진, "비다양체 모델을 수용하는

- CAD 시스템 커널을 위한 불리안 조작 의 개발," 한국 CAD/CAM 학회 논문집, 제 1권, 제 1호, 1996년 3월.
- 16. Gossard, D. C., Zuffante, R. P. and Sakurai, H., "Representing Dimensions, Tolerances, and Features in MCAE Systems," IEEE Computer Graphics & Applications, March, 1988.
 - 17. Wang, N. and Ozsoy, T. M., "A scheme to represent features, dimensions, and tolerances in geometric modeling," Journal of Manufacturing Systems, Vol. 10, No. 3, 1991.
 - 18. Rossignac, J. R., "Constraints in Constructive Solid Geometry," Proc. Workshop on Interactive 3D Graphics, Durham, N. C., Oct. 1986.
 - 19. Bruderlin, B., "Constructing Three-Dimensional Geometric Objects Defined by Constraints," Proc. Workshop on Interactive 3D Graphics, Durham, N. C., Oct. 1986.
 - 20. Lee, K. and Gossard, D.C., "A Hierarchical Data Structure for Representing Assemblies: Part 1," Computer Aided Design, Vol 17, No 1, 1985.
 - 21. Lee, K. and Andrews, G., "Inference of the Positions of Components in an Assembly: Part 2," Computer Aided Design, Vol 17, No 1, 1985.
 - 22. Rocheleau, D.N. and Lee, K., "System for Interactive Assembly Modelling," Computer Aided Design, Vol 19, No 2, 1987.
 - 23. Kim, S.H. and Lee, K., "An Assembly Modeling System for Dynamic and Kinematic Analysis," Computer Aided Design, Vol 21, No 1, 1989.
 - 24. Srikanth, S. and Turner, J. U., "Toward a Unified Representation of Mechanical Assemblies," Engineering with Computers, Vol. 6, pp. 103-112, 1990.
 - 25. Anantha, R., Kramer, G. A. and Crawford, R. H., "An architecture to represent over-, under-, and fully constrained assemblies," Proc. of ASME Conference, pp. 233-244, 1992.
 - 26. Kramer, G. A., "A geometric constraint engine," Artificial Intelligence, pp. 327-360, 1992.
 - 27. Aldefeld, B., "Variation of geometries based on a geometric-reasoning method," Computer Aided Design, Vol. 20, No. 3, 1988.
 - 28. Sunde, G., "Specification of Shape by Dimensions and other Geometric constraints," IFIP WG. 5.2 on Geometric Modeling, Rensselaerville, NY, May, 1986.
 - 29. Sunde, G., "A CAD System with Declarative Specification of Shape," EuroGraphics Workshop on Intelligent CAD Systems, April, 1987.
 - 30. Burden, R. L. and Faires, J. D., Numerical Analysis, Fourth Edition, PWS-KENT, 1988.