

설계 공리를 이용한 TV 브라운관 유리 제품 설계 자동화 시스템 구축

Software Development for Glass Automatic Design Integrated System(GADIS)
using Design Axiom

도성희, 이정숙 (한양대학원 기계설계학과)

박경진 (한양대학교 기계공학과)

김창훈, 김재선, 최주호(삼성코닝 (주))

1. 서론

최근 들어 국내 제조업 분야의 대외 경쟁력을 향상시키기 위해 각 산업체에서는 여러 가지 자동화 시스템을 구축하려는 경향이 지배적이다. 자동화 시스템을 구축하는 방법 및 대상은 여러 가지가 있을 수 있으며 그 중에서 설계를 담당하는 자동화 시스템의 개발은 제품 생산을 위한 독자적인 기술을 확보하기 위해서 반드시 필요한 대상 중의 하나이다. 그러나 대개의 경우 설계 과정은 상당히 복잡하고 어려운 작업으로서 최종 제작 도면을 만들어 내기까지에는 몇 가지 단계를 거치게 된다. 이를 과정을 설계를 수행하는 기술자가 보다 쉽고 빠르게 처리할 수 있는 자동화 시스템을 구축한다면 제품 생산에 상당한 과급효과를 기대할 수 있을 것이다.

설계 과정에 대한 학문적이고 체계적인 연구 노력은 설계 과정의 일반화가 어렵고 다양하기 때문에 아직까지는 활발한 토의가 진행되고 있지는 않고 있다. 다만 설계 공리(design axiom)라는 이론화된 연구 주제가 제기되어 몇 가지 경우에 대해 적용을 시도하고 있다.⁽¹⁾ 설계에 대한 정의 및 이론에 대한 설명을 기능 요구(functional requirements, FRs)와 설계 변수(design parameters, DPs)로 분석하고 이들의 사상 과정(mapping process)을 통해 설계 결정 과정(decision making process)을 수행하는 과정으로 구성되어 있다. 설계 과정의 최종 목표는 주어진 조건을 만족하는 좋은 설계를 창출하는데 있는 만큼 이론적인 배경을 통해 적합한 해(solution)를 찾는 시스템의 구축은 필수적이다.

본 연구에서는 설계 공리를 이용하여 제품 설계의 전 과정을 분석하고 제품 설계자가 쉽게 설계를 수행할 수 있도록 설계 자동화용 소프트웨어 시스템을 구축하는데 목표를 둔다. 개발된 소프트웨어의 대상은 TV 브라운관 유리 제품(glass bulb)으로서 현재 수행되고 있는 제품 설계의 전 과정을 설계 공리의 관점으로 재 조명하고 효율적으로 개선하는 몇 가지 소프트웨어를 구축하였다. 전체 시스템은 메뉴 시스템을 적용한 사용자 인터페이스(graphic user interface, GUI)기법을 이용하여 사용자가 보다 편리하게 설계 과정을 수행할 수 있도록 하였다.

2. 설계 공리에 대한 고찰

2-1. 설계 공리

대부분의 설계 대상은 항상 기능적인 영역(functional domain)에서 언급되는 반면 물리적인 해 즉, 그것을 이루기

위한 방법은 물리적인 영역(physical domain)에서 이루어진다. 설계 대상이 되는 특별한 요구들(requirements)에 대한 개념은 기능 요구(FRs)라 하고 이들 기능 요구들을 만족하기 위한 물리적인 형체는 설계 변수(DPs)의 개념으로 특정 지울 수 있다. 설계 과정이란 기능적인 영역의 기능 요구와 물리적인 영역에서의 설계 변수 사이의 사상(mapping)을 통해 파생되는 과정을 의미하며 이를 두 영역을 관계 짓는 것이 바로 설계라고 할 수 있다.⁽²⁾ Fig. 1은 고객이 원하는 것이 실제 생산을 위한 변수로 사상되어지는 과정을 설명한다. 고객이 요구하는 사항을 기능적인 영역 내의 기능 요구 집합으로 사상시키고 물리적인 영역 내에서 기능 요구를 만족하는 설계 변수를 구한다. 이때 설계 변수는 기능 요구와 설계 변수의 같은 계층 구조 단계를 서로 오고가는 zig-zag 방식으로 구할 수 있다. 다음으로 생산 변수(process variables, PVs)로 이루어진 생산 영역(process domain)으로 사상된다. 이러한 사상 과정은 유일한(unique) 과정이 아니므로 기능 요구에 맞는 적절한 설계 변수를 찾아내는 것은 여러 개의 해가 나올 수 있고 이는 절대적으로 설계자의 손에 달려 있다고 할 수 있다.

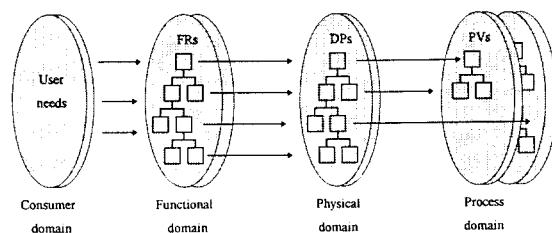


Fig. 1 Concept of domain, mapping and spaces

Fig. 1의 과정을 수행하는데 필요한 지배적인 이론은 설계 공리(design axiom)라는 형태로 제공되고 있으며 다음과 같이 정의된다.

Axiom 1 : The independence axiom

Maintain the independence of functional requirements(FRs).

Axiom 2 : The information axiom

Minimize the information content.

공리 1은 사상 과정에서 좋은 설계가 되기 위한 기준을 제공한다. 일반적으로 기능적인 영역과 물리적인 영역 사이의 사상 과정에서 설계 방정식은 식 (1)과 같이 나타낼 수 있다.

$$\{FRs\} = [A]\{DPs\} \quad (1)$$

여기서 $\{FRs\}$ 는 독립적인 FR_i 요소들로 생산물의 기능적인 영역을 표현하는 벡터(vector)이다. 또한 $\{DPs\}$ 는 $\{FRs\}$ 의 영향 하에서 생산물을 정의하는 설계 변수를 타나내는 벡터이다. $\{FRs\}$ 와 $\{DPs\}$ 사이에는 설계 행렬.design matrix) $[A]$ 의 꼽으로 표현된다. 설계 행렬의 인자 A_{ij} 는 식 (2)로 구할 수 있다.

$$A_{ij} = \frac{\partial FR_i}{\partial DP_j} \quad (2)$$

이 경우 설계 공리를 만족하기 위해서 설계 행렬 $[A]$ 는 대각 행렬(diagonal matrix)이나 삼각 행렬(triangular matrix)이어야 한다. 설계 행렬이 대각 행렬인 설계를 비연성 설계(uncoupled design), 설계 행렬이 삼각 행렬인 설계는 비연성화 설계(decoupled design)라고 부르며 이런 설계는 공리 1을 만족한다. 이외의 다른 설계 행렬을 갖는 설계는 연성 설계(coupled design)라 한다. 마찬가지로 물리적인 영역과 생산 영역 사이의 사상 과정은 식(3)과 같이 나타낼 수 있다.

$$\{DPs\} = [B]\{PVs\} \quad (3)$$

$[B]$ 행렬은 생산 설계 행렬(product design matrix)이고 $\{PVs\}$ 는 생산 영역의 생산 변수를 나타내는 벡터이다. 똑같이 $[B]$ 행렬의 인자들은 생산 변수(process variables, PVs)에 대한 설계 변수를 편 미분한 값이다.

공리 2에서 정의된 정보 내용(information content)은 설계의 생산물과 생산 과정에 따른 기능 요구와 설계 변수를 성공적으로 얻을 확률로 정의된다. 정보 I_i 는 식(4)로 정의된다.

$$I_i = \log_2 \frac{1}{p} \quad (4)$$

여기서 p 는 FR_i 를 얻을 확률이다. n 개의 기능 요구가 있을 때 최상의 설계 즉 공리 2를 만족하는 설계는 가장 적은 정보 내용을 가진 설계로 식 (5)와 같이 나타낼 수 있다.

$$I_{\min} = \min \left\{ \sum_{i=1}^n I_i \right\} \quad (5)$$

공리 1과 공리 2의 관계를 살펴보면 공리 1은 독립성을 요구하는 것으로서 설계의 단순한 형태의 설계를 요구한다. 공리 2도 최소한의 정보량을 요구하는 것으로 공리 1과 서로 관계된다. 따라서 공리 1을 만족하는 설계는 공리 2를 만족하리라고 생각할 수 있다. 그러나 실제 설계 시에는 공리 1과 공리 2 모두 만족하는 설계를 찾아내기란 쉽지 않다. 예를 들어 공리 1을 만족하는 비연성화된 설계보다 적은 정보를 갖는 연성화된 다른 설계를 선택할 수도 있다. 반대로 공리 2를 만족하는 설계 중에서 공리 1을 만족하는 비연성화된 설계를 선택하는 것이 불가능할 수도 있다. 대다수의 경우에는 공리 1을 만족하는 몇 가지 비연성화된 설계들이 제안된 이후에 공리 2를 고려하는 방향이 제시되고 있다.

2-2. 설계 공리의 소프트웨어에 대한 적용

소프트웨어의 설계는 각각의 설계 목적들을 적절한 설계 영역으로 나누고 연속적인 사상 과정을 통한 특정 백터들의 계층 구조를 만들어 내는 것으로 나뉘어질 수 있다. 일반적으로 기능적인 영역과 물리적인 영역은 기능 요구와 설계 변수의 계층적인 구조이며 각 단계별로 서로의 영역(domain)을 zig-zag 형태로 넘나들면서 사상을 해 나가게 된다. 소프트웨어 시스템의 설계에 있어서도 기능적인 영역의 기능 요구 계층 구조와 물리적인 영역의 설계 변수 계층 구조로 구성되어 진다. 이때 소프트웨어의 결과는 기능 요구의 집합이 되고 소프트웨어의 코드 즉 프로그램은 같은 계층 구조상에서 설계 변수를 기능 요구로 변환시키는 설계 행렬에 해당된다.

기능 요구와 설계 변수의 사상 과정은 물리적인 영역의 최고 단계의 기능 요구에서 시작된다. 이 단계에서의 설계 변수는 공리 1의 독립성을 위반하지 않는 범위 내에서 규정된 기능 요구와 부합하도록 선택되어진다. 만일 설계해가 선택된 설계 변수에 대해서 완전히 만족되지 않는다면 기능 요구는 하위 단계로 분해(decomposition)되어야 한다. 하위 단계에서 기능 요구를 정의할 때 이미 선택된 설계 변수는 기능 요구를 제한하게 되며 기능 요구의 설정은 명시된 설계 변수가 선택되었을 때 제한 조건(constraint)을 만족하는 범위 내에서 선택될 것이다. 이러한 방식으로 분해 과정은 수행된다.

3. 제품 설계 자동화 소프트웨어의 구성

TV나 컴퓨터에 사용되는 모니터의 출력장치로 사용되는 브라운관 유리는 별브(glass bulb)로 명명되며 그 재질이 유리로 구성되어 있기 때문에 제품을 설계하는 과정이 비교적 단순하다. 생산된 제품에는 새도우 마스크(shadow mask)와 들판 전자총과 같은 장치가 부착되고 캐비닛(cabinet)에 고정시키기 위한 밴드(band)가 장착되어야 비로소 브라운관이라 부르게 된다. 결국 별브 자체는 하나의 부품으로 취급되는 단위 요소 제품으로 전면 유리(panel)와 후면 유리(funnel) 두 가지로 구성된다.⁽³⁾

현재 제품 설계를 수행하는 방식은 데이터 베이스의 미비로 생산을 위한 정보 체계가 도면을 통해 이루어지고 있으며 각종 설계 과정을 분산된 여러 시스템에서 수행하는 등 비효율적으로 진행되고 있다. 이를 개선하는 소프트웨어를 제작함에 있어 설계 공리를 이용하여 개발되는 소프트웨어의 성능을 향상시키고 각종 설계 단계에서의 정보 전달 체계를 강화하고자 한다.

3-1. 계층 구조와 분해

(hierarchical structuring and decomposition)

별브를 설계하기 위한 기능 요구 및 설계 변수들을 고찰하면 다음과 같다. 각각의 경우는 계층화된 구조를 zig-zag 방식으로 전환해 가며 좋은 설계 방안을 도출해 내게 된다.

step 1 : FRs → DPs

먼저 최고 단계의 기능 요구 집합을 정리하면 다음과 같다.

- FR1 : 신제품에 대한 데이터 베이스를 생성한다.
- FR2 : 제품의 형상을 구축한다.
- FR3 : 제품의 성능을 검토한다.
- FR4 : 제품 도면을 생성한다.

이에 대한 설계 변수들을 선정하면

- DP1 : 신제품 데이터 (panel / funnel)
- DP2 : 제품의 3차원 형상 구조 (panel / funnel)
- DP3 : 하중 조건 (panel / funnel)
- DP4 : 설계 데이터 집합

과 같으며 이에 대하여 설계 행렬은 다음과 같이 비연성화된 설계 결과가 생성된다.

$$\begin{Bmatrix} FR1 \\ FR2 \\ FR3 \\ FR4 \end{Bmatrix} = \begin{Bmatrix} X & 0 & 0 & 0 \\ X & X & 0 & 0 \\ X & X & X & 0 \\ X & X & X & X \end{Bmatrix} \begin{Bmatrix} DP1 \\ DP2 \\ DP3 \\ DP4 \end{Bmatrix}$$

DP1과 DP2가 결합되면 FR3은 상용 구조 해석 소프트웨어를 이용하여 해결 가능하므로 FR3은 하나의 모듈(M3)로 구성할 수 있다. 나머지 요소들은 주어진 DP에 대해 세분화 과정이 필요하다.

step 2 : DPs → FRs

FR1에 대하여 하위 레벨의 기능 요구 집합을 선정하면

- FR11 : 신제품의 ID number를 부여한다.
- FR12 : 신제품의 데이터 집합을 구축한다.

과 같고 마찬가지로 FR2, FR4에 대한 기능 요구 집합을 설정하면 다음과 같다.

- FR21 : 곡률을 검토한다.
(panel : 편평도 / funnel : 축별 profile)
- FR22 : 3차원 형상을 계산한다.
- FR23 : 제조 가능성성을 검토한다.

- FR41 : 제품의 형상을 표현한다.

- FR42 : 주변 악세사리(accessory)를 표현한다.

step 3 : FRs → DPs

두 번째 단계의 기능 요구들을 만족하는 설계 변수 및 설계 행렬을 구하면 다음과 같다. 우선 FR11, FR12에 대해서는

- DP11 : 신제품의 대표 코드
(ssc_code, eiaj_code, eiak_code)
 - DP12 : 신제품의 사양 데이터 집합
- $$\begin{Bmatrix} FR11 \\ FR12 \end{Bmatrix} = \begin{Bmatrix} X & 0 \\ X & X \end{Bmatrix} \begin{Bmatrix} DP11 \\ DP12 \end{Bmatrix}$$

과 같이 나타나며 신제품의 ID number는 설계 사양 데이터 집합이 만들어지기 전에 수행되므로 비연성화된 설계 결과를 도출한다. FR21, FR22, FR23에 대해서는

- DP21 : 제품의 내외면 곡률
- DP22 : 제품의 기하학적인 연관 관계식

DP23 : 금형 제작에 관련된 데이터 집합

$$\begin{Bmatrix} FR21 \\ FR22 \\ FR23 \end{Bmatrix} = \begin{Bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{Bmatrix} \begin{Bmatrix} DP21 \\ DP22 \\ DP23 \end{Bmatrix}$$

과 같이 나타나며 DP23에 대한 세분화 작업이 필요하므로 FR23에 대한 하위 레벨에 대한 정의가 다시 필요하다. FR41, FR42에 대해서는

DP41 : 제품 설계 데이터 집합

DP42 : 악세사리 데이터 집합

$$\begin{Bmatrix} FR41 \\ FR42 \end{Bmatrix} = \begin{Bmatrix} X & 0 \\ 0 & X \end{Bmatrix} \begin{Bmatrix} DP41 \\ DP42 \end{Bmatrix}$$

과 같이 나타나며 비연성 설계 결과를 도출한다.

step 4 : DPs → FRs

FR23에 대하여 하위 레벨의 기능 요구 집합을 설정하면

FR231 : 전면 유리의 유효 화면 거리(useful screen dimension)를 만족하는가 검토한다.

FR232 : 방출성(ejectability)이 용이해야 한다.

FR233 : 후면 유리의 주사선 편향 각도가 만족되는지 검토한다.

과 같다.

step 5 : FRs → DPs

세 번째 단계의 기능 요구들을 만족하는 설계 변수 및 설계 행렬을 구하면 다음과 같다.

DP231 : 내면 블렌딩 원(blending circle) 중심점의 중심 축으로부터의 거리

DP232 : 측면 벽(side wall) 각도

DP233 : 요크부 내면 곡률

$$\begin{Bmatrix} FR231 \\ FR232 \\ FR233 \end{Bmatrix} = \begin{Bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ 0 & 0 & X \end{Bmatrix} \begin{Bmatrix} DP231 \\ DP232 \\ DP233 \end{Bmatrix}$$

FR23을 구성하는 하부 레벨 설계는 비연성 설계 결과를 나타내며 결과적으로 FR2를 구성하는 하부 레벨 설계는 비연성화된 설계 결과를 얻을 수 있다.

Fig. 2에 설계용 소프트웨어를 위한 계층 구조와 분해 과정을 정리하여 설명하였다.

3-2. 모듈 결합 구조 (module-junction structure)

앞에서 제시한 기능 요구를 위한 모듈들은 프로그램으로 정형화되고 소비자에 의해 명시된 기능의 시스템으로 구성되어 질 수 있다. 이것은 소프트웨어 설계의 시스템 통합을 의미한다. Fig. 3과 같이 기능 요구와 설계 변수의 계층적 나무 구조(tree structure)와 설계 행렬의 형태로 나타난 해석의 결과는 하나의 도표로 나타내어 질 수 있고 이를 모듈 결합 구조 도표(module junction structure diagram)라고 한다.⁽²⁾

결합 구조 도표는 세 가지의 결합 형태가 있으며 다음과 같이 설명할 수 있다.

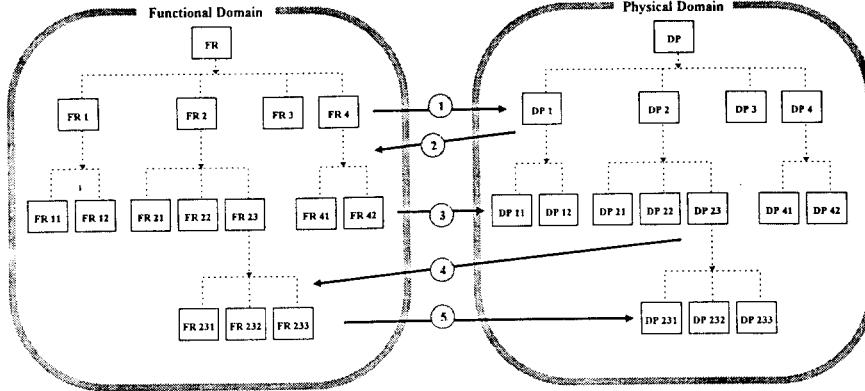


Fig. 2 The zig-zag decomposition of the FR and DP hierarchy structures

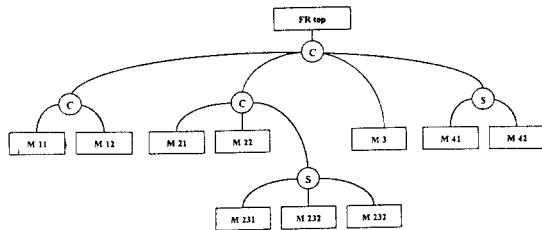


Fig. 3 Module junction structure diagram

합 결합(summation junction : ⑤) : 기능 요구가 비연성(uncouple) 되었을 경우로 부모(parent)의 기능 요구가 자식(child) 모듈의 모든 결과를 결합함으로서 만족되는 형태이다.

제어 결합(control junction : ©) : 기능 요구가 비연성화(decoupled) 되었을 경우로서 부모의 기능 요구가 오른쪽 모듈의 행위를 제어하기 위해 왼쪽 모듈의 결과를 사용하는 형태이다.

반복 결합(feedback junction : ⑥) : 기능 요구가 연성화(coupled) 되었을 경우로서 오른쪽 모듈의 결과를 왼쪽 모듈로 되돌려 보내는 형태로 정보의 진행 과정이나 프로그램의 이용에 있어서 수많은 반복(iteration)을 요구한다. 반복 결합을 포함하고 있는 부모들(parent-module)이 많을 때 그 프로그램은 곧 감당하기 어렵게 된다.

결합 특징을 사용함으로서 모듈 결합 구조도는 모듈들 간의 정보 흐름을 보여주는 네트워크 형태 도로 바꿔어지기가 쉽다. 이는 일반적인 구조화된 해석과 설계 방법에서의 중요한 그림적인 표현이었던 정보 흐름도와 같다. Fig. 4는 모듈 결합 구조도에서 유추된 정보 흐름도이다. Fig. 4의 그림 중 점선으로 표시된 선은 기능 요구와 설계 변수들 간의 분해 과정에 대한 반복 결합을 의미하는 것은 아니며 단지 설계의 흐름상 해석 결과가 좋지 않아 설계 변수 값의 변경을 주기 위한 반복(iteration)을 수행하기 위한 것이다.

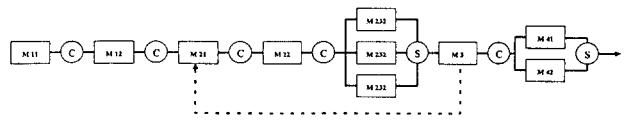


Fig. 4 Information flow diagram

3-3. 제품 설계 자동화 소프트웨어의 구성

제품 설계 자동화 시스템을 구축하기 위한 시스템의 구성은 앞에서 분석한 설계 결과의 방법을 충분히 지원할 수 있도록 Fig. 5와 같이 구현한다. 사용자 인터페이스를 통해 사용자는 원하는 기능을 수행할 수 있으며 구축된 데이터 베이스는 제품 생산의 전 과정에 참고 정보로 사용된다.

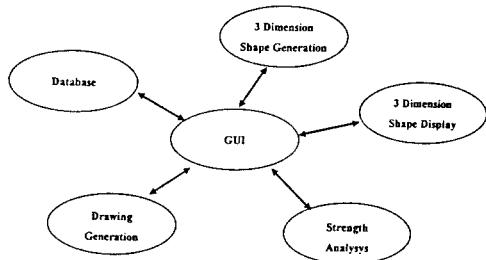


Fig. 5 Structure of the Glass Automatic Design Integrated System(GADIS)

사용자 인터페이스(GUI)

모듈 결합 구조도를 표시한 Fig. 3을 바탕으로 제품 설계를 위한 사용자 인터페이스를 구성하면, 기능별로 설계 변수 입력, 강도 해석, 도면 생성 등으로 구별하며 각각에 대해 전면 유리, 후면 유리를 설계하는 메뉴를 두어 설계 작업을 수행하도록 한다.

데이터 베이스(Database)

설계 작업을 수행하는 단계에 있어서 정의되는 각종 설계 변수를 관리하기 위해서는 데이터 베이스의 구축이 가장 필

요한 작업이다. 제품 설계를 위해 필요한 정보는 크게 형상 정보와 도면 관련 정보 그리고 관리 정보로 분류할 수 있다. 형상 정보는 제품의 3차원 형상에 관한 정보로 각 부위별 치수와 각종 포인트 정보(point data), 다항식 정보(polynomial data), 다중 반경 정보(multi r data)등으로 구성된다. 이들중 일부는 도면을 제작하기 위한 2차원 정보로 변환되어 도면의 형상을 구현하는데 필요하다. 도면 관련 정보는 형상 정보 외에 각종 측정 위치 좌표, 핀(pin)과 같이 제품에 장착되는 부품에 관한 데이터 등의 정보로 구성된다. 관리 정보는 제품 번호, 설계 진행 코드 등이 있다.

데이터 베이스로 구축되는 정보들은 끊임없이 입력, 수정, 삭제 등이 작업이 이루어지므로 이들의 관리에는 상당히 주의를 기울여야 한다. 아울러 정보들의 속성을 파악하여 적절한 데이터 군으로 분류하고 이를 군들의 연관 관계를 정립하는 과정은 전체 시스템의 성능에 상당한 영향을 미치기 때문에 FR과 DP의 배치 관계처럼 entity - relation 관계(E-R Diagram)를 정의하여 전체 데이터 베이스를 구현하는 방법을 많이 사용한다. Fig. 6에 제품 설계를 위한 E-R 도표를 설명한다.

3차원 형상 생성

유리 별브 제품은 3차원 형상이 중요하기 때문에 주어지는 설계 정보를 바탕으로 3차원 형상을 구현하는 소프트웨어가 필요하다. 3차원 형상을 구현하기 위해서는 몇 가지 기하학적인 기법을 이용하여 계산을 통해 형상 정보를 추출해 낼 수 있다. 기초 설계 정보로부터 3차원 형상이 구축될 수 있는지 여부를 판단한 뒤 형상 구축이 불가능하면 설계 정보를 변경해 가며 형상 정보를 구성해 가면 된다.

해석

유리 별브 제품의 해석 대상이 되는 작업은 진공 해석과 밴드 해석의 두 가지가 있으며 구현된 3차원 형상 정보를 바탕으로 유한 요소용 메쉬(mesh)를 생성하는 전 처리(post process) 과정을 거치면 제품의 강도 등을 검증하는 전 단계가 완료된다. 전 처리(post process) 과정이 끝나면 강도 해석용 소프트웨어를 이용하여 설계된 유리 별브의 강도 해석을 실시한다. 해석 결과는 후처리(pre processor)를 통하여 그래픽적으로 사용자에게 보여질 수 있다.

사용자는 해석 결과를 바탕으로 설계 변수를 수치 값을 조정하는 피드백(feedback) 과정을 거치는 과정이 반복된다. Fig. 4의 정보 흐름도 중 점선에 해당하는 과정이 반복(iteration)의 과정을 표시하는 것으로 기능 요구와 설계 변수를 재조정하는 것이 아니라 설계 변수의 값만을 조정하여 전체 시스템을 운영하게 된다.

3차원 형상 출력

유리 별브의 설계에 있어서 계산된 제품의 3차원 형상이 정확히 구성되었는지 사용자가 확인해 볼 필요가 있으며 이를 위해 3차원 형상의 출력이 필요하다. 아울러 강도 해석에 필요한 제품의 매쉬 생성 결과 등을 강도 해석 이전에 출력해 주는 전 처리(post process)로서의 기능도 필요하다.

도면 출력

모든 설계 과정이 끝나면 설계된 정보를 바탕으로 도면을 출력하는 작업이 필요하며 이 과정은 CAD 소프트웨어를 이용하여 구현할 수 있다. 보통의 경우 도면화하는 작업은 CAD 소프트웨어에 능숙한 사람이 도면을 제작하게 되지만 설계 데이터가 제공된다면 도면화하는 작업 또한 자동으로 처리할 수 있다. 대부분의 CAD 소프트웨어에는 프로그래밍이 가능하도록 연결 도구(interface tool)를 제공하여 이를 연결 도구를 사용하여 도면을 제작하는 전용 프로그램을 작성할 수 있다.

4. 소프트웨어의 개발

제품 설계 자동화를 위한 소프트웨어는 몇 가지 성격이 다른 부분들을 통합해야 하기 때문에 제작되는 실행 화일의 크기가 커지는 단점이 있다. 이러한 단점을 보완하고 3장에서 제시한 설계의 분석 결과를 적용하여 프로그램을 모듈화하고 공용 라이브러리(shared library) 형태로 만들어 서로 연결하는 방식을 추진한다.

유리 별브의 설계를 위해서는 각종 곡률 데이터 및 데이터 베이스를 운영하기 때문에 처리하고자 하는 데이터의 양을 정확히 추정하기가 어렵다. 따라서 이를 정보들을 저장할 메모리 공간을 확보하기가 쉽지 않기 때문에 동적 메모리 할

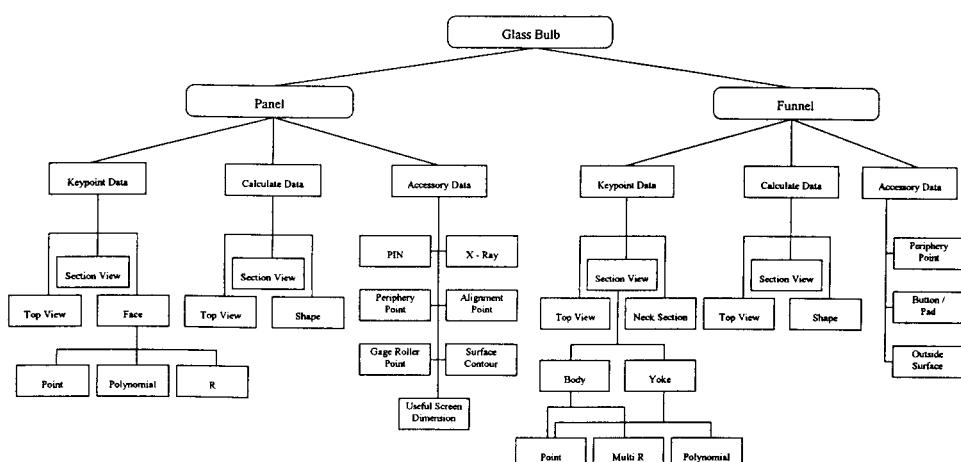


Fig. 6 Entity - relation diagram for the software

당 기법을 사용하여 그때그때 필요한 메모리를 확보하여 처리하는 방법을 데이터 처리를 위한 기본 방법으로 사용한다.

사용자 인터페이스(GUI) - X-Window/MOTIF

사용자 인터페이스의 도구로는 UNIX 시스템의 표준 그래픽 도구인 X-window(MOTIF)를 이용하여 소프트웨어를 개발한다.⁽⁴⁾ 사용자 인터페이스를 통해 각종 메뉴 시스템이라든지 그래프 등의 표현 양식을 제공하게 된다.

데이터 베이스 - ORACLE

데이터 베이스와의 접속을 위한 방법으로는 데이터 베이스를 운영하는 소프트웨어를 통해 사용자가 직접 데이터 베이스 운영 명령을 입력하여 사용하는 방법과 데이터 베이스 운영 명령을 프로그램화하여 사용하는 방법의 두 가지가 있다. 데이터 베이스 운영 명령으로는 입력, 수정, 삭제 그리고 조회 등에 관련된 명령들로서 이들 명령을 통해 데이터 베이스를 구축할 수 있다. 이들 운영 명령을 프로그램화하여 사용하는 경우에는 데이터 베이스 소프트웨어가 제공하는 컴파일러(pre compiler)를 이용하여 실행 파일을 제작한 뒤 실행하면 동일한 효과를 얻을 수 있다.⁽⁵⁾

전체 제품 설계 시스템의 완벽한 지원을 위해 데이터 베이스 접속 프로그램은 두 가지 방식으로 사용하도록 구성한다. 첫 번째로 사용자 인터페이스를 통해 정상적으로 데이터 베이스를 접속하는 방법과 두 번째로 비상시의 경우(예를 들어 데이터 베이스의 복사(backup) 및 이전, 재구축 등)에 사용자 인터페이스와는 상관없이 데이터 베이스를 접속하도록 하는 방식으로서 이들의 접속시 수행되는 기본 명령들은 공용 라이브러리(library)로 구축하며 이들 기본 명령들은 입력, 수정, 삭제, 조회 등으로 구성된다.

3차원 형상 및 메쉬 생성 - BULB-3D

전면 유리와 후면 유리의 3차원 형상 및 메쉬 생성을 위한 각종 기하학 계산을 위해 자체 개발된 소프트웨어인 BULB-3D를 실행시키면 제품의 3차원 형상 정보를 얻을 수 있다. BULB-3D는 몇 가지 기하학과 수치 해법을 이용하여 3차원 형상 및 강도 해석을 위한 메쉬를 생성해 주는 기능을 담당하고 있다.

3차원 형상 출력 - Starbase

3차원 형상을 표현하는 기법은 보통 세 가지가 있는데, wireframe 형상, 표면 형상(surface modeling) 그리고 고체 형상(solid modeling)이 그것이다. 이들 3차원 형상을 프로그램을 이용하여 표현하기 위한 각종 3차원 지원 그래픽 라이브러리(graphic library, GL)는 몇 가지가 있으며, 대부분 축소/확대(zooming) 기능, 이동(translation) 기능, 회전(rotation) 기능 그리고 렌더링(rendering) 기능 등을 제공한다. 이를 기능 등을 이용하면 제품의 3차원 형상에 대한 검증을 시작적으로 할 수 있다. 참고로 3차원 형상을 출력하기 위한 3차원 그래픽 라이브러리는 스타베이스(starbase)를 사용한다.⁽⁶⁾

강도 해석 - ANSYS

강도 해석을 위해서는 상용 해석 소프트웨어인 ANSYS를 이용하여 제품의 전공 강도 및 밴드 강도 성능을 검증하게 된다.⁽⁷⁾ BULB-3D를 이용하여 메쉬가 정의되었으면 유리의 물성치, 각종 하중 조건, 구속 조건 등의 데이터를 바탕으로 제품의 강도를 평가한다. 원하는 강도 결과를 얻을 때까지 3차원 형상 생성 모듈과 연계해서 반복적으로 수행한다.

도면 출도 - Unigraphics

도면 출도를 위해서는 유니그래픽스(unigraphics)라는 CAD 소프트웨어를 이용하여 설계된 유리 벌브의 도면을 자동으로 출도한다. 유니그래픽스가 제공하는 프로그래밍 연결 도구로는 그립(GRIP)과 사용자 함수(user function)가 있으며, 사용자 함수를 이용하여 C 프로그램만으로 선, 원, 곡선 등을 자동으로 그리도록 할 수 있다.⁽⁸⁾ 도면에 나타내는 내용으로는 제품의 형상뿐만 아니라 형상의 치수, 각종 측정 위치, 편장착 위치, 정렬 위치 그리고 각종 표등이 삽입되며 사용자의 선택에 따라 도면에 나타내는 항목을 조정할 수 있다. 각종 설계 변수에 따라 도면을 자동으로 출도하기 때문에 도면 관리가 편리하고 일관성 있게 도면 작성할 수 있다.

5. 결론

최근 들어 공학적인 설계(engineering design)에 대한 몇 가지 이론들이 제시되고 있고 설계 과정 자체를 개선하려는 일련의 노력들이 진행되고 있는 시점에서 설계 과정에 대한 몇 가지 분석을 통해 현재 수행 중인 과정보다 효과적이고 빠른 설계 결과를 도출해 내는 방법을 연구하였다. 설계 공리의 소프트웨어에 적용 가능성은 제시하였고 제안된 방법을 TV용 유리 벌브을 생산하는 업체에서 수행하는 설계 방법에 적용하여 만족할 만한 결과를 얻을 수 있었다. 본 연구를 통하여 자동화 시스템을 구축하려는 추세에 부응하는 일반적인 설계 방안을 도출하는 하나의 초석을 마련하였다 생각한다. 개발된 시스템을 통해 신제품의 개발 낭비 단축, 설계 자동화 기반 구축, 디자인 소량 생산 및 독자 타입 개발 대기 기본 시스템의 구축, 제품 개발 업무의 간소화 그리고 전 생산 공정의 생산 정보 단일화 등의 효과를 얻을 수 있었다. 설계 결과로 생성되는 설계 정보는 앞으로 제조 자동화 시스템이 구축되면 직접 제공될 수 있으리라 기대된다.

본 연구를 통해 설계 공리에 대한 각 분야의 연구 노력이 좀더 활성화될 수 있는 장을 마련할 수 있기를 희망한다.

참고문헌

1. N.P. Suh, "The Principles of Design", Oxford University Press, 1990.
2. S.J. Kim, N.P. Suh and S.G. Kim, "Design of Software Systems Based on Axiomatic Design", Robotics & Computer-Integrated Manufacturing, Vol. 8, No. 4, pp. 243-255, 1991.
3. 박경진 외 4인, "제품 설계자 동화 시스템 구축", 삼성코닝 최종 보고서, 1995.
4. Dan Heller, Paula M. Ferguson, "Motif Programming Manual", O'Reilly & Associates, Inc, 1994.
5. "ORACLE 3GL Programmers Guide", Oracle Corporation, 1990.
6. "Starbase Graphics Techniques", Hewlett-Packard Company, 1991
7. "ANSYS User's Guide", Swanson Analysis Systems, Inc., 1993.
8. "User Function Programming Manual", Electronic Data Systems Corporation, 1993