

공리적 설계 이론 향상을 위한 제언

최 덕 현* · 황 운 봉†

(2003 년 10 월 17 일 접수, 2004 년 5 월 24 일 심사완료)

A Suggestion and a Contribution for the Improvement of Axiomatic Design

Dukhyun Choi and Woonbong Hwang

Key Words : Variation of Design Parameter(설계 변수의 변화량), Flow Chart(흐름도), End of Decoupling(비연성화의 종료)

Abstract

For a long time, design methodologies strive to systematize the design process in order to make the practice more efficient and effective. One of such methodologies is Axiomatic Design. However, this design theory still has some problems and is not completely settled. In this paper, the new issue for the non-linear design in Axiomatic Design is suggested and the representation of system architecture by flow chart is corrected. In the case that the design equation varies as a function of the variation of DP (δDP), the design should be regarded as the nonlinear design because the design equation is not a constant. When system architectures are represented by a flow chart, all systems could not be independently represented. However, by adding the notation of the end of decoupling to the notations using in flow chart, those can be independently represented.

기호설명

- FR: 기능적 요구 사항
- DP: 설계 변수
- A: 설계 행렬
- I: 정보량
- P: 성공확률
- M: 모듈
- Ⓢ: 비연성화 관계
- Ⓣ: 비연성 관계
- Ⓤ: 연성 관계
- =>: 비연성화의 종료

1. 서론

이전까지의 제품 개발 방식은 직관 및 시행착오에 의한 경험이 주종을 이루었다. 이러한 제품

개발 방법은 제품 개발 과정뿐만 아니라 제품 개발 후에도 많은 시간 및 자원을 낭비하게 된다.^(1,2) 특히, 1980 년대에 발생한 인도의 화학공장 사고, 구소련의 체르노빌 원자력 발전소 사고, NASA 우주 왕복선의 O-링의 결함과 같은 공학 설계 분야의 많은 실패 사례들은 잘못된 설계의 치명적인 문제점에 대한 새로운 인식을 불러 넣었다.⁽³⁾ 더욱이, 정밀해지고 정교해지는 최근의 제품 개발 과정에서 체계적이고 논리적인 제품 설계의 필요성은 과거의 제품 설계 과정에서 보다 크게 부각되고 있다.⁽⁴⁾

공리적 설계(Axiomatic Design)는 1980 년대에 MIT 에서 개발된 설계 공학 이론으로 창의적이며, 체계적으로 적용할 수 있는 설계 이론이다. 이 이론은 설계의 초기 단계부터 공리 이론을 적용하여 복잡한 문제를 단순화 시키고, 설계의 성공 확률을 높이는 방향으로 설계를 진행한다. 또한, 이미 설계된 시스템에 대한 진단도 가능하다.⁽⁵⁾ 그러나, 공리적 설계론은 아직 보완 및 개선 해야 할 부분이 남아있다. 본 논문에서는 아직까지 공리적 설계에서 다루지 않은 문제를 제시하고, 공리적 설계에서의 시스템 구조 표현 기법을 수정하고자 한다.

* 회원, 포항공과대학교 대학원 기계공학과

† 책임저자, 회원, 포항공과대학교 기계공학과

E-mail : whwang@postech.ac.kr

TEL : (054)279-2174 FAX : (054)279-5899

2. 공리적 설계 이론

공리적 설계는 두 가지 설계 공리를 바탕으로 한다. 공리 1은 기능과 물리적 변수 사이의 관계를 설명하고 공리 2는 설계의 복잡성에 대해 다룬다. 이를 간단히 표현하면 다음과 같다.

공리 1 독립 공리

기능적 요구 사항의 독립성을 유지하라.

공리 2 정보 공리

설계의 정보량을 최소화 하라.

공리 1은 설계 과정 중, 기능적 영역의 요구 사항으로부터 물리적 영역의 설계 변수로 진행되는 과정을 특정 설계 변수의 변동이 오직 관련된 기능적 요구 사항에만 영향을 주도록 정의해야 한다는 것을 의미한다. 일반적으로 기능적 요구 사항과 설계 변수의 수는 같게 되고, 이때 기능적 요구 사항과 설계 변수의 관계는 식 (1)처럼 표시할 수 있다.

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

식 (1)에서 FRs(Functional Requirements)는 기능적 요구 사항으로 수요자가 요구하는 제품의 기능과 관련된 요소이다. DPs(Design Parameters)는 설계 변수로 기능적 요구 사항을 수행하기 위한 물리적 요소가 된다. [A]는 설계 행렬로 이와 관련된 설계 행렬은 식 (2)의 세가지 형태가 있다.

$$A_1 = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ 0 & 0 & A_{33} \end{bmatrix} \quad (2.a)$$

$$A_2 = \begin{bmatrix} A_{11} & 0 & 0 \\ A_{21} & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \quad (2.b)$$

$$A_3 = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \quad (2.c)$$

식 (2.a)는 비연성(Uncoupled) 행렬이라 하고 독립 공리를 완전히 만족하는 설계가 된다. 식 (2.b)는 비연성화(Decoupled) 행렬이라 하고 독립 공리를 완전히 만족하지는 못하지만 설계 순서를 조절하면 독립 공리를 만족하는 설계가 된다. 식 (2.c)는 연성(Coupled) 행렬이라 하고 독립 공리를 만족하

지 못하는 설계가 된다. 따라서 공리적 설계로 제품 설계할 때 식 (2.c)와 같은 설계 행렬이 나오면 그 설계는 잘못된 설계이고 이를 식 (2.a) 또는 식 (2.b)와 같은 설계가 되도록 기능적 요구 사항과 설계 변수를 바꾸어야 한다.

공리 2는 독립 공리를 만족하는 설계 중에서 최소한의 정보량(Information Contents)을 갖는 설계가 가장 좋은 설계라는 것을 나타내고 있다. 여기에서 정보량은 성공 확률 및 복잡성(Complexity)과 관련이 있고 정보량은 식 (3)와 같이 표현된다.

$$I = \log_2(1/P) \quad (3)$$

식 (3)에서 I는 정보량이고 P는 성공 확률(Probability Density)이다. 성공 확률은 Fig. 1에 나타난 공통 영역(Common Range)을 시스템 영역(System Range)으로 나눈 값이다. 즉, 성공확률은

공통 영역이 클 때 커지는데 시스템 영역이 설계 영역에 포함될 때 공통 영역은 시스템 영역과 같아져서 성공 확률을 1이 되고 이때 정보량은 0이 되어 가장 좋은 설계가 된다.^(3,5)

따라서, 공리적 설계는 독립 공리를 만족하는 설계 중 정보 공리에 의해 최적화된 설계를 선택할 수 있게 해준다. 이 이론에 의하면 설계 초기부터 독립 공리가 만족되지 않는 설계는 최적화될 수가 없다. 즉, 기존의 설계를 최적화하는 노력을 하기 전에 공리적 설계 이론을 바탕으로 독립 공리가 만족되는지를 진단해 보고 이를 만족하지 않으면 최적화 될 수 없으므로 설계를 다시 해야 한다. 이런 방법으로 기존의 설계를 진단하고 새로운 모델을 설계할 때 초기 단계부터 최적화된 설계를 가능하게 해준다.

3. 이론 개선을 위한 제언

공리적 설계에서 기능적 요구 사항과 설계 변수가 하나인 설계의 경우, 이들의 관계는 식 (4)과 같이 되고 이에 대한 미분 형태는 식 (5)와 같이 된다.

$$FR_1 = A_{11}DP_1 \quad (4)$$

$$dFR_1 = \frac{\partial FR_1}{\partial DP_1} dDP_1 \quad (5)$$

여기에서 식 (4)를 미분하여 식 (5)와 비교하면 설계 행렬 A_{11} 에 대한 식 (6)이 된다.

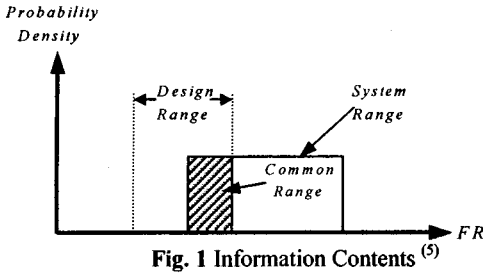


Fig. 1 Information Contents (5)

$$A_{11} = \frac{\partial FR_1}{\partial DP_1} - \frac{\partial A_{11}}{\partial DP_1} DP_1 \quad (6)$$

식 (6)에서 설계 행렬이 설계 변수의 함수일 경우 비선형 설계가 되고, 설계 행렬이 설계 변수의 함수가 아니면 식 (6)의 오른쪽 두 번째 항이 0이 되어 설계행렬은 상수가 되고 선형 설계가 된다.

한편, 현재까지 제안된 공리적 설계의 이론에서는 설계 행렬이 설계 행렬이 설계 변수의 변화량(δDP)의 함수가 되는 경우에 대한 분석은 제시되지 않았다. 이 경우, 설계 행렬이 상수가 아니므로 이를 비선형 설계로 분석하여야 하며, 이에 대한 구체적인 이론 정립이 필요하다. 다음은 설계 행렬이 설계 변수의 변화량의 함수가 되는 예제이다.(5)

스틸 샤프트(Steel Shaft)에 알루미늄 관(Al Tube)이 연결되어 있는 알루미늄 튜브를 -30°C 에서 70°C 온도 범위에서 단단하게 연결하기 위해 스틸 샤프트의 반경과 알루미늄 관의 두께를 구하는 경우, 기능적 요구 사항(FR₁)은 스틸 샤프트와 알루미늄 관 사이의 압축력((σ_{rr})_{r=r₀})이 되고, 설계 변수(DP₁)는 억지 끼워 맞춤(Interference Fit (vr))으로 생각할 수 있다. 한편, 온도 변화 및 가공 오차에 의해 불규칙 변화량(Random Variation (δ(vr)))이 발생하게 되고 이 경우 설계 방정식은 식 (7)과 같이 된다.

$$(\sigma_{rr})_{r=r_0} = f(r_0, t)vr \quad (7)$$

참고문헌 (1)에서는 설계 행렬이 설계 변수의 함수가 아니므로 위의 설계를 선형 설계로 간주하여 Fig. 2 와 같이 분석하였으나, 여기에서 δ(vr) = 0.002 + 0.0005r₀ 이므로 설계 행렬은 설계 변수의 변화량에 대한 함수가 되고 이 경우 설계 행렬은 상수가 아니므로 비선형 설계로 문제를 분석하여야 할 것이다.

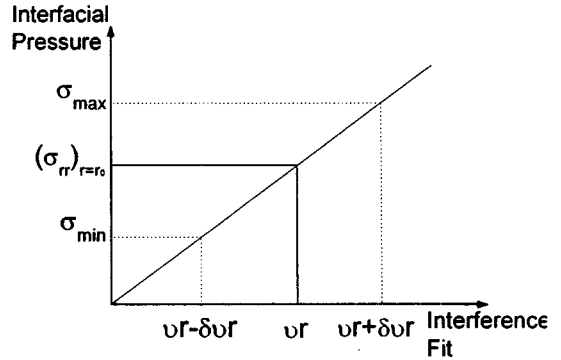


Fig. 2 Interfacial pressure vs. interference fit (5)

4. 이론 개선을 위한 수정

공리적 설계에서 시스템 설계 시 시스템 구조 (System Architecture)를 표현하는 방법에는 3 가지가 있다. 첫번째는 계층구조(Hierarchy)를 이용하는 방법이고 두 번째는 결합 모듈 선도(Junction Module Diagram)를 이용하는 방법 그리고 세 번째는 흐름도(Flow Chart)를 이용하는 방법이다. 계층구조를 이용하는 방법은 Fig. 3 에서 보듯이 각 변수를 더 낮은 계층으로 분해해서 표현하는 방법이고 결합 모듈 선도는 Fig. 4 에서 보듯이 각 모듈을 원형 기호로 연결해서 표현하는 방법이다. 여기에서 모듈이란 기능적 요구 사항과 설계 변수의 일대일 관계를 만족 시키는 설계 행렬 요소를 뜻하는 것으로 식 (8)과 같은 행렬에 대해 M₁ 과 M₂ 는 식 (9)와 같이 계산 된다.

$$\begin{Bmatrix} FR_1 \\ FR_2 \end{Bmatrix} = \begin{bmatrix} a & 0 \\ b & c \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \end{Bmatrix} \quad (8)$$

$$FR_1 = aDP_1 = M_1 \times DP_1 \quad (9.a)$$

$$FR_2 = bDP_1 + cDP_2 = M_2 \times DP_2$$

$$, \text{ where } M_1 = a \quad (9.b)$$

$$M_2 = b(DP_1 / DP_2) + c$$

흐름도를 이용하는 방법은 다음의 기호를 사용하여 Fig. 5 처럼 각 모듈사이의 관계와 순서를 표시해서 나타낸다. ㉓는 두 모듈이 비연성 되어있을 때 ㉔는 두 모듈이 비연성화 되어 있을 때 ㉕는 두 모듈이 연성 되어 있을 때 설계 행렬을 표시하는 기호이다. 시스템을 흐름도로 표시했을 때 시스템의 설계 순서를 알 수 있을 뿐만 아니라 모든 모듈이 한눈에 들어오기 때문에 시스템을

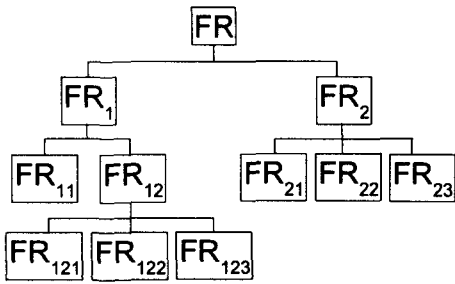


Fig. 3 Hierarchy⁽⁵⁾

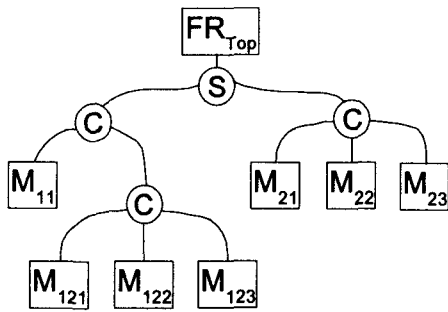


Fig. 4 Module-Junction Diagram⁽⁵⁾

이해하기가 쉬워진다. 즉, 위의 세 방법은 시스템의 복잡한 기능적 요구 사항과 설계 변수들을 체계적으로 나열함으로써 설계자가 범할 수 있는 오류를 줄여주며, 서로간의 관계를 명확하게 연결 지어주기 위해 필요한 구조도이다.

그러나 위의 기호만으로 모든 설계 행렬을 독립적으로 표시할 수 없다. 다음의 식 (10)와 같은 설계 행렬에 대한 흐름도는 Fig. 6 과 같다. 즉, Fig. 6의 흐름도가 식 (10.a)를 나타낼 수도 있고 식(10.b)를 나타낼 수도 있게 되어 기존에 사용되는 흐름도의 기호들만으로는 모든 설계 행렬을 독립적으로 표현 할 수 없다. 이는 참고문헌 (6)의 Table 1 과 Fig. 10에서도 볼 수 있다.

따라서, 비연성화의 종료를 의미하는 Fig. 7의 새로운 기호를 추가하여 식 (10)에 대한 흐름도를 Fig. 8 과 같이 독립적으로 표현하였다. 즉, 식 (10.a)에서는 첫번째 기능적 요구 사항(FR₁)이 세번째 설계 변수(DP₃)에 영향을 받아 두 관계가 비연성화 관계지만 식 (10.b)에서는 두 모듈이 비연성 관계가 된다. 그런데, 단일 화살표의 연결은 앞 모듈과의 관계가 계속 유지되어 M₁ 과 M₃의 관계도 비연성화 관계가 된다.

그러나, 기존의 기호들만으로는 이를 구분시킬 수가 없다. 그러므로 비연성화의 종료를 나타내는

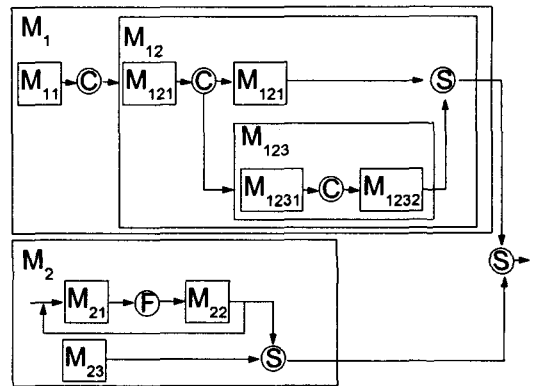


Fig. 5 Flow Chart about design equation indicated in Fig. 3 and Fig.4⁽⁵⁾

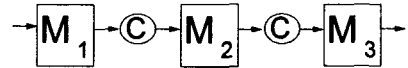


Fig. 6 Flow Chart of Eq. (10.a) and Eq. (10.b)

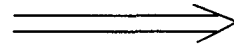


Fig. 7 Notation indicating the End of Decoupling

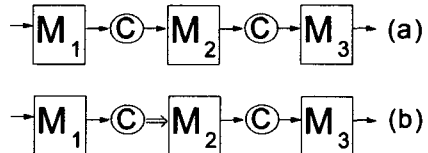


Fig. 8 Independent Flow Chart(a) Eq.(10.a) (b) Eq.(10.b)

이중 화살표를 사용하여 이 기호가 사용된 부분의 앞쪽 모듈은 다음 부분의 모듈과 비연성화 관계가 이어지지 않게 한다. Fig. 8은 비연성화의 종료를 나타내는 기호를 사용하여 Fig. 6과 같이

$$\begin{cases} FR_1 \\ FR_2 \\ FR_3 \end{cases} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{cases} DP_1 \\ DP_2 \\ DP_3 \end{cases} \quad (10.a)$$

$$\begin{cases} FR_1 \\ FR_2 \\ FR_3 \end{cases} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ 0 & X & X \end{bmatrix} \begin{cases} DP_1 \\ DP_2 \\ DP_3 \end{cases} \quad (10.b)$$

독립적이지 않은 흐름도가 나온 식 (10)의 설계 행렬을 독립적으로 표현한 흐름도이다. Fig. 8(a)의

흐름도에서는 단일 화살표와 © 기호로 모듈을 연결하여 M₃ 모듈과 M₁ 모듈의 비연성화 관계가 이어지게 된다. 그러나 Fig. 8(b)의 흐름도에서는 이중 화살표를 사용하여 M₁ 모듈이 M₂ 모듈에서 비연성화가 끝나게 되어 M₃ 모듈과 M₁ 모듈의 비연성 관계가 되므로 두 설계 행렬에 대한 독립적인 표현이 가능하게 되었다. 다양한 설계행렬에 대한 독립적인 흐름도를 부록에 첨부하였다.

5. 결론

본 논문에서는 공리적 설계 이론의 정립을 위해 고려되는 설계 문제를 제언하고 시스템 구조 표현 기법을 개선하였다. 이와 관련되어 다음의 결론을 내릴 수 있다.

(1) 공리적 설계에서 설계 행렬이 설계 변수의 변화량에 대한 함수일 경우, 이에 대한 분석은 비선형 설계가 되어야 하며 이러한 경우에 대한 분석 기법이 정립되어야 함이 고려되었다.

(2) 흐름도로 시스템 구조를 표현할 때 비연성화의 종료를 의미하는 기호를 추가하여 모든 시스템의 구조의 표현을 독립적으로 나타낼 수 있게 하였다.

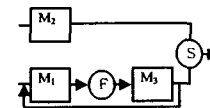
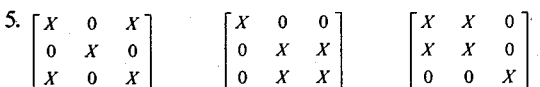
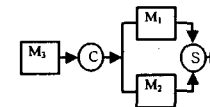
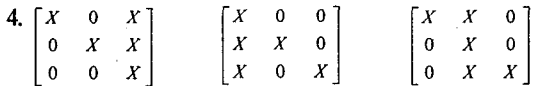
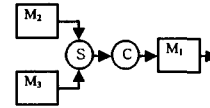
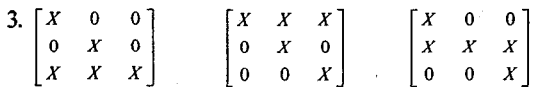
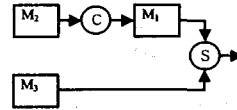
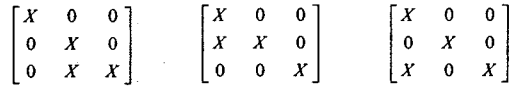
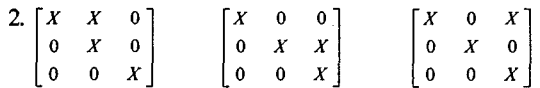
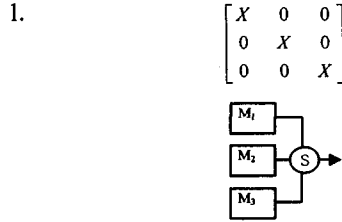
참고문헌

(1) Jeong, D. J., Cha, S. W. and Yoon, J. D., 2001, "Design for Microcellular Plastics using Axiomatic Approach," *Journal of KSPE*, Vol. 18, pp. 195~200.
 (2) Rikard, S. and Lars, L., 1999, "Computer Aided Assembly Robustness Evaluation," *J. Eng. Design*, Vol. 10, No. 2, 165~181.
 (3) Suh, N.P., 1990, *The Principles of Design*, Oxford.
 (4) Rinderle, J. R. and Suh, N. P., 1982, "Measures of Functional Coupling in Design," *J. Eng. Industry*, Vol. 104, 383~388.
 (5) Suh, N.P., 2001, *Axiomatic Design; Advances and Applications*, Oxford.
 (6) Park, G. J., Do, S. H. and Suh, N. P., 1999, "Design and Extension of Software Systems Using the Axiomatic Design Framework," *Transaction of KSME A*, Vol. 23, pp. 1536~1549.

부 록

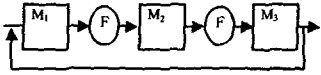
기능적 요구조건과 설계 변수가 3x3 인 설계행렬에 대한 흐름도를 정리하였다. 총 64 종류의 설계 행렬이 존재하고, 이 중에서 재배치를 통해 같

아지는 행렬들을 하나의 그룹으로 하여, 독립적인 흐름도를 갖는 설계 행렬을 15 그룹으로 나누어 정리하였다.



15.

$$\begin{bmatrix} X & X & X \\ X & X & X \\ X & X & X \end{bmatrix}$$



앞에 정리한 15 개의 그룹 중에서 그룹 6 과 그룹 7, 그룹 8 과 그룹 9 그리고 그룹 11 과 그룹 12 는 비연성화의 종료를 의미하는 기호를 이용하여 구별이 되지 않던 흐름도가 독립적으로 구별이 되는 것을 알 수 있다.