

# 공리적 설계

## Introduction to Axiomatic Design



박 경 진 / 한양대학교  
Gyeong-Jin Park / Hanyang University



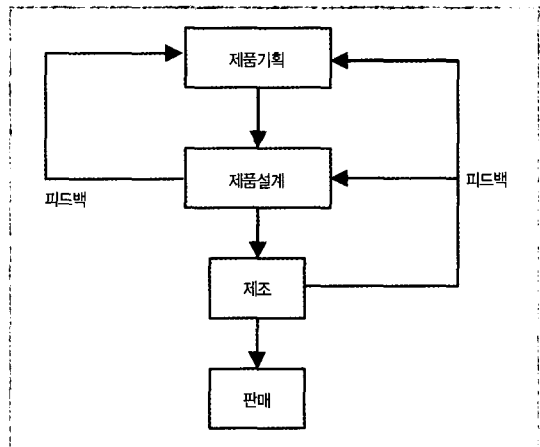
강 병 수 / 한양대학교  
Byung-Soo Kang / Hanyang University

### 1. 공학설계란 무엇인가?

설계란 무엇인가? 이에 대한 답은 사람마다 다를 수 있다. 일상생활에 대한 계획을 의미하기도 하고 예술적 아름다움과 생활의 편리함을 주는 도구 창출을 뜻하기도 한다. 여기서 사용하는 설계의 개념은 이런 포괄적인 의미가 아니라 공학적 제품을 생산할 때 세우는 일련의 계획을 뜻한다. 이를 달리 기계설계라고도 한다. 기계설계는 제품의 전반적 기능의 정의나 배치 동일 수도 있고, 자세한 치수의 결정이나 도면 제작을 의미할 수도 있다. 기계설계는 선사시대에 사냥을 위한 도구를 개발할 때부터 존재해 온 무엇보다도 오래된 공학적 행위이다. 그런데 공학적 행위 중 어느 단계를 설계로 보느냐의 문제에 대한 대답은 명확하지 않다. 이렇듯 설계 개념의 불명확성에도 불구하고 일반적으로 공학자들은 계획 단계를 설계로 간주하는 데에는 대체로 동의하고 있다.

공학적 작업에서 설계의 위치를 살펴보자. 제품의 기획단계에서 판매까지의 흐름도가 <그림 1>에 있다. <그림 1>에서 음영 처리된 부분이 기계설계 과정이다. 이 과정에서는 기획단계에서 정의한 문제를 공학적 수

단을 통해 물리적으로 제조가 가능하도록 변환하는 작업이 이루어진다. 즉, 제품의 기획단계에서 문자나 개념으로 정의된 사항을 물리적으로 구현할 수 있도록 구체화 하는 계획이 이루어진다. 설계에서의 주된 작업은 기획 단계에서 수립된 사항을 면밀히 검토하고 이의 공학적 타당성을 검증하여 최종 생산에 필요한 각종 부품 및 치수를 결정하는 일이다. 많은 경우 그 결과를 도면화하게 된다. 한편, 설계의 목적은 초기의 작업과 후반의 작업을 원활히 연결시키는데 있다.



<그림 1> 제품생산의 흐름도

설계과정에서 설계자는 필히 어떤 결정(Decision Making)을 하게 된다. 그런데 그 결정을 하는 연유나 논리적 과정은 어떤 것일까? 또 좋은 결정의 특성을 객관적으로 정리할 수 있을까? 결정을 내리는 과정에 대한 시각차에 따라 설계를 예술(Art)로 보는 쪽과 과학(Science)으로 보는 쪽이 존재한다. 어떤 시각이 옳은 지에 대해서는 논란이 계속되고 있다. 설계를 예술로 간주하는 쪽은 결정 과정에서 직관을 중요하게 여긴다. 설계를 과학으로 간주하는 쪽은 직관보다는 객관적인 논리를 강조하며 보편적인 설계 원리가 존재한다고 주장한다. 이와 같이 설계의 특성은 다양하게 나타나지만 좋은 설계에 대한 보편적인 원리를 찾아낸다면 일반적인 설계법이나 설계이론을 말할 수 있을 것이다. 실제로 최근에는 좋은 설계라고 알려진 것들 사이의 공통점을 찾아내어 이들을 정형화된 설계 이론으로 설명하려는 연구 경향이 대두하고 있다.

설계는 설계방법이나 주위의 환경에 따라 다르게 분류할 수 있다. 우선 과정의 선후에 따라 <그림 1>의 음영 처리된 부분을 개념설계와 상세설계로 나눈다. 일반적으로 개념설계를 마친 후 그 결과에 따라 상세설계를 실시한다. 개념설계는 초기 작업으로 기능의 정의, 주요 부품의 취사 선택, 각 부품의 위치나 서로 간의 관계 등을 결정한다. 영향이 큰 사항은 대부분 이 단계에서 완성된다. 반면 상세설계에서는 개념설계 단계에서 결정한 설계 결과를 바탕으로 각 부품의 치수, 모양 등을 결정한다. 이 때 제품의 성능을 판별하기 위한 공학적인 실험이나 해석이 수반된다.

공리적 설계는 앞에서 언급한 객관적이고 보편적인 원리를 지닌 정형화된 설계이론을 정립하기 위한 노력의 일환으로 개발되었다. 일반적으로 공리적 설계는 개념설계에서 사용한다. 상세설계에서 사용한 예도 있으나 현재까지의 응용사례를 살펴보면 대부분 개념설계에서 사용하였음을 알 수 있다. 여기서는 지면이 제

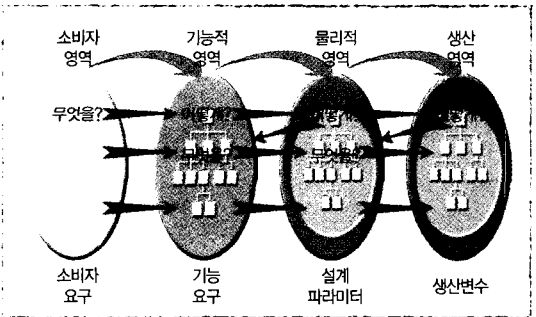
한되기 때문에 기본적인 이론에 관한 부분만을 설명하려 한다. 더 자세한 내용과 설계공리를 적용한 다양한 예제들은 참고문헌에 수록되어 있다.

## 2. 공리적 설계

공리적 설계는 매사추세츠 공과대학의 서남표 교수에 의해 창출되고 보급되어 온 설계이론이다. 그러나 좀더 정확히 말한다면 공리적 설계는 설계이론이라기 보다는 일반적인 공학적 과정에 적용할 수 있는 원리를 포함하는 설계프레임워크라 할 수 있다. 공리적 설계는 독립공리와 정보공리로 이루어져 있다. 공리적 설계에서 좋은 설계는 두 공리를 만족하는 설계이다. 두 공리를 만족하지 않는 설계는 좋은 설계가 아니다. 공리라는 용어는 기하학에서 사용되는 것과 동일한 의미로, 증명되지는 않지만 반례가 발견될 경우 거짓이 된다는 것을 의미한다. 그런데 아직까지는 공리적 설계의 공리들에 대한 반례가 제시되지 않았다. 오히려 공리적 설계로 설명이 가능한 좋은 설계의 예들이 증가하고 있다.

설계는 일반적으로 '무엇을 이룰 것인가?'와 '그것을 어떻게 얻을 것인가?'의 상호작용이라 정의할 수 있다. 설계자는 양쪽의 기능 및 자원을 적절히 조합하고 배치해서 원하는 것을 얻으려 한다. 설계를 수행할 때에 문제를 정의하고 설계를 완성하는 등의 순서에 따라 <그림 2>와 같은 영역들을 만들 수 있다. 소비자 영역에서는 소비자들이 원하는 요구가 정의되며 대개 이들 요구는 서술적으로 표현할 수 있다. 그러나 최근에는 소비자들의 요구가 없는데도 소비자가 필요로 할 것 같은 사항을 생산자가 예측하여 직접 정의하는 경우도 흔하다. 이렇게 정의한 것을 소비자 요구(Customer Attributes, CAs)라 한다. 소비자 영역에서 정의한 사항은 기능적 영역에서 기능요구

(Functional Requirements, FRs)로 변환된다. 기능요구는 소비자 영역의 소비자 요구를 공학적 용어로 재정의한 것이다. 이는 앞에서 언급한 '무엇을 이룰 것인가?'에 대한 공학적 정의라 볼 수 있다. 기능요구는 물리적 영역에서 설계파라미터(Design Parameters, DPs)를 정의하거나 선정함으로써 만족시킨다. 대개 이 과정을 설계과정이라 부른다. 설계파라미터를 결정하면 위와 동일한 방법으로 설계파라미터로부터 제조를 위한 생산변수(Production Variables, PVs)를 정의하게 된다. 이와 같이 공리적 설계에서의 설계는 두 영역 사이의 관계를 파악하여 앞의 영역에서 정의한 사항에 따라 다음 영역에서의 사항을 정의한다. 이를 영역간 사상이라 한다. 좋은 설계란 이러한 사상을 효율적으로 정의한 것을 뜻하기도 하다.



〈그림 2〉 도메인, 사상, 설계공간간의 관계

설계공리는 모든 공학적 행위의 근간에 흐르는 공통적인 원리를 집약적으로 표현한 것으로 다음과 같이 정의된다.

**제 1공리 : 독립공리(The Independence Axiom)**  
 기능요구의 독립성을 유지하라. (Maintain the Independence of the Functional Requirements(FRs))

**제 2공리 : 정보공리(The Information Axiom)**  
 설계 대상의 정보량을 최소화하라. (Minimize the Information Content of the Design)

공리는 대개 기하학에서 정의해 왔는데, 기하학의 경우와 마찬가지로 여기에서도 공리로부터 증명이 가능한 많은 정리 및 추론을 도출할 수 있다. 현재까지의 각종 정리 및 추론이 참고문헌에 수록되어 있다.

### 3. 독립공리

독립공리는 〈그림 2〉에 있는 영역간의 사상 과정에서 오른쪽 영역에 정의되는 사항들이 왼쪽 영역의 사항들을 독립적으로 만족시키는 것이 좋은 설계라는 의미이다. 앞 절에서 정의한 FR-DP 혹은 DP-PV 간의 관계를 독립적으로 유지하라는 것이다. 복수의 FR이 정의 되었을 때, 각 DP가 각 FR을 독립적으로 만족시킬 수 있어야 한다. 이러한 FR-DP 간의 관계는 설계행렬로 표현할 수 있다. 복수의 FR과 DP는 벡터로 표시할 수 있다. 이들의 관계는 설계행렬 A를 사용하여 다음과 같이 표현할 수 있다.

$$FR = A \cdot DP$$

독립공리를 만족하는지의 여부는 행렬 A의 특징으로 결정된다. FR과 DP 벡터의 요소 수가 각각 3이라 하면 행렬 A는 다음과 같다.

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix}$$

설계행렬 A의 특성에 따른 관계식이 〈표 1〉에 있다. 〈표 1〉에서 설계행렬이 대각행렬이 되는 경우를 비연성설계(Uncoupled Design)라 한다. 이 경우 각 설계파라미터가 대응하는 기능요구를 독립적으로 만족할 수 있으므로 완벽하게 독립공리를 만족하게 된다. 다음으로 설계행렬이 삼각행렬인 경우가 있다. 이것을 비연성화설계(Decoupled Design)라 부른다. 비연성

〈표 1〉 설계행렬의 특성에 따른 FR-DP 관계

	설계 방정식	설계결과
비연성설계	$\begin{cases} FR_1 \\ FR_2 \\ FR_3 \end{cases} = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ 0 & 0 & A_{33} \end{bmatrix} \begin{cases} DP_1 \\ DP_2 \\ DP_3 \end{cases}$	$\begin{aligned} FR_1 &= A_{11} * DP_1 \\ FR_2 &= A_{22} * DP_2 \\ FR_3 &= A_{33} * DP_3 \end{aligned}$
비연성화설계	$\begin{cases} FR_1 \\ FR_2 \\ FR_3 \end{cases} = \begin{bmatrix} A_{11} & 0 & 0 \\ A_{21} & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{cases} DP_1 \\ DP_2 \\ DP_3 \end{cases}$	$\begin{aligned} FR_1 &= A_{11} * DP_1 \\ FR_2 &= A_{21} * DP_1 + A_{22} * DP_2 \\ FR_3 &= A_{31} * DP_1 + A_{32} * DP_2 \\ &\quad + A_{33} * DP_3 \end{aligned}$
연성설계	$\begin{cases} FR_1 \\ FR_2 \\ FR_3 \end{cases} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{cases} DP_1 \\ DP_2 \\ DP_3 \end{cases}$	$\begin{aligned} FR_1 &= A_{11} * DP_1 + A_{12} * DP_2 \\ &\quad + A_{13} * DP_3 \\ FR_2 &= A_{21} * DP_1 + A_{22} * DP_2 \\ &\quad + A_{23} * DP_3 \\ FR_3 &= A_{31} * DP_1 + A_{32} * DP_2 \\ &\quad + A_{33} * DP_3 \end{aligned}$

화 설계는 설계파라미터를 결정하는 순서에 따라서 기능요구를 독립적으로 만족시킬 수 있다. 〈표 1〉의 두 번째 경우를 보면 우선  $DP_1$ 을 결정하여  $FR_1$ 을 만족시킨다. 이어서  $DP_2, DP_3$ 을 결정하여  $FR_2, FR_3$ 을 순서대로 만족시키게 된다. 그러나 〈표 1〉의 세번째의 경우처럼 설계행렬의 비대각 항이 영이 아닐 때에는 어떤 순서로도 FR과 DP가 연성의 관계를 가진다. 이를 연성설계(Coupled Design)라 한다. 비연성설계나 비연성화설계는 독립공리를 만족하는 것으로 간주하며 연성설계는 독립공리를 위배하는 것으로 간주한다. 위에서 기능요구인 FR과 설계파라미터인 DP를 표기할 때, 각 요소들을 구분하기 위해 아래 첨자를 사용했다. 그러나 표현방법에 있어서 아래 첨자를 사용하지 않기도 한다. 따라서  $FR_i$ 와  $FR_j$ 는 동일한 것이다. 한편, 설계행렬 A는 일반적인 수학적 연산 가운데 곱셈만 가능하고 좌표변환 등의 연산은 불가능하다.

어떤 설계를 공리적 설계 방법으로 분석했을 때 비연성설계나 비연성화설계는 좋은 설계이며 연성설계는 그렇지 않다는 것을 의미한다. 따라서 연성설계

는 설계파라미터의 재조정을 통해 비연성설계나 비연성화설계로 변환하여야 한다.

여기서 주목할 사항은 제한조건(Constraints, Cs)이다. 제한조건은 설계 시 꼭 만족시켜야 하는 항목이다. 예컨대 비용이나 설계사양 등은 어떤 환경에서도 만족되어야 하는 명제이다. 제한조건은 설계의 독립성에 관계없이 정의할 수 있으며, 다수의 설계파라미터가 연성되어 있을 수도 있다. 〈그림 2〉에서 보는 바와 같이 DP나 PV영역에서 제한조건을 정의할 수 있다

#### 4. 정보공리

공리적 설계에서는 독립공리가 우선 만족하여야 한다. 그런데 독립공리를 만족하는 설계를 두 개 이상 발견할 수 있다. 독립공리를 만족하는 복수의 설계들 중에서 가장 우수한 설계를 선택하려면 정량적으로 정보량이 최소인 것을 선정한다. 그렇다면 정보량은 어떻게 정의할까? 상황에 따라 달라질 수 있지만 정보량은 일반적으로 복잡성(Complexity)을 의미한다. 정보공

리에서는 정보량을 계산하기 위해 지수를 정의한다. 각 설계의 특성에 따라 정보량의 지수는 다르게 정의할 수 있다. 현재까지 일반적인 설계에 적용 가능한 지수로는 성공의 확률(Probability of Success)이 알려져 있다. 설계한 제품을 실제로 사용할 때 특정  $DP_i$ 가 해당  $FR_i$ 를 만족하는 성공의 확률을  $p$ 라 하자. 이 때 정보량  $I_i$ 를 다음과 같이 정의한다.

$$I_i = \log_2 \frac{1}{p}$$

여기서 확률의 역수를 취한 이유는 성공의 확률이 높을수록 정보량을 적게 만들기 위해서이다. 그리고 로그함수를 취한 것은 정보량의 가산성(Additivity)을 증진시키기 위함이며, 로그함수의 밑수가 2인 이유는 정보량을 비트(Bit) 단위로 표시하기 위함이다.

예를 들어 다음과 같은 비연성설계가 있다 하자.

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ 0 & 0 & A_{33} \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix}$$

$DP_1, DP_2, DP_3$ 가 각각  $FR_1, FR_2, FR_3$ 를 만족할 확률을  $p_1, p_2, p_3$ 라할 때 총정보량  $I_{total}$ 은 다음과 같다.

$$I_{total} = \sum_{i=1}^3 I_i = \sum_{i=1}^3 \log_2 \left( \frac{1}{p_i} \right)$$

여기서 주의할 점은 정보량은 정의된 기능요구에 관해서만 정의할 수 있다는 것이다. 이 외에도 확률밀도함수에 의해 정보량을 계산하는 방법도 있다.

### 5. 설계공리의 응용

대부분의 경우에 독립공리가 만족되면 정보량은 감소한다. 따라서 정보공리가 독립공리에 의존하므로 정보공리는 필요 없는 것으로 생각할 수 있다. 그러나 그렇지 않은 경우가 종종 생긴다. 예를 들어 주어진 기능

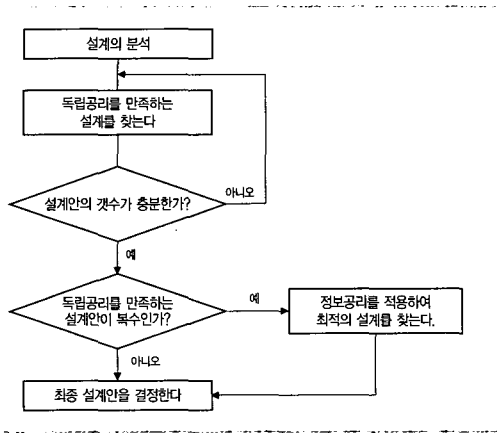
요구를 만족하는 비연성설계와 연성설계를 발견했다고 하자. 연성설계의 정보량이 비연성설계의 정보량보다 적을 수 있다. 그러면 연성설계가 더 좋은 설계라고 할 수 있는가? 대답은 '그렇지 않다'이다. 이런 경우는 발견한 연성설계보다 정보량이 더 적은 비연성설계나 비연성화설계가 존재함을 암시하는 것이다. 따라서 설계자는 더 좋은 설계를 찾기 위해 노력해야 한다. 다른 예로 비연성설계나 비연성화설계를 복수로 발견할 수도 있다. 독립공리에 대한 만족도가 동일하다고 가정하면, 정보량이 최소가 되는 설계를 최종적으로 선택해야 한다. 설계공리를 적용하는 과정이 <그림 3>에 있다.

공리적 설계방법을 실제 설계에서 어떤 방식으로 응용할 수 있는가에 대해 살펴본다. 대체로 다음과 같은 분야에 적용할 수 있다고 본다.

- (1) 창조적 설계
- (2) 기존 설계의 분석
- (3) 개선 설계

새로운 기능요구가 정의되고 이러한 기능요구를 만족하는 제품이 존재하지 않는다고 가정하자. 이 경우 새로운 제품을 창출하려 할 것이다. 창조적 두뇌를 가진 자에 의해 새로운 제품에 대한 아이디어가 창출되는 것이 일반적이다. 그러나 이렇게 창출된 설계안을 구체화하는 방법으로 공리적 설계법을 사용할 수 있다. 즉 아이디어를 공학적으로 분석하고 이를 구현할 수 있는 부품의 선정, 배치 등에 대한 결정과정을 공리를 이용해서 수행할 수 있다. 그러나 현실적으로 기계설계에 있어서 새로운 창조적 설계를 도출하는 것은 대단히 어렵고 드물다. 따라서 기존의 설계를 크게 변화시키고 개선할 수 있다면 창조적 설계로 간주하는 경향이 있다.

이미 생산된 제품들을 평가할 때는 흔히 소비자들의 여론조사를 한다. 그러나 설계자의 관점에서 어떤



〈그림 3〉 공리적 설계 적용의 흐름도

제품이 더 우수한가를 평가할 때에 공리적 설계방법을 유용하게 도입할 수 있다. 즉 제품의 목적을 기능요구로 정의하고 현재의 제품을 공리적 설계방법으로 분석하여 평가할 수 있다. 이 때에는 어떤 제품이 독립공리를 더 만족하는가를 평가하게 된다. 복수의 제품이 동일한 정도로 독립공리를 만족한다면 최소의 정보량을 가지는 가장 좋은 제품을 선정할 수 있다.

또한 공리적 설계는 현재 제품의 성능을 개선하는데에 사용할 수 있다. 현재의 제품성능이 만족스럽지 않을 때나 좀더 좋은 성능을 가지도록 제품 개선을 꾀할 때에 우선 독립공리의 관점에서 기존 제품을 분석

한다. 기능요구와 설계파라미터를 정의하고 그들의 관계 정의를 통해 독립공리의 만족여부를 점검해 본다. 독립공리를 만족하지 않을 때는 독립공리를 만족할 수 있도록 개선된 설계를 도출한다. 독립공리를 만족하면 독립공리가 위배되지 않는 범위에서 정보량을 최소화 하는 설계파라미터를 도출한다.

앞에서 언급했듯이 공리적 설계는 개념설계에서 매우 유용하게 사용할 수 있다. 그러나 공리적 설계는 설계의 추진 방향을 제공할 뿐 각 설계의 구체적인 해를 제시하지는 않는다. 이러한 최종 설계는 다른 설계법과 마찬가지로 설계자의 몫이다. 최근에 사용되는 설계법으로 다루지법, TRIZ, 최적설계방법 등이 있다. 공리적 설계가 이들과 크게 다른 것은 복수의 기능요구를 조직적으로 처리할 수 있다는 점이다. 이러한 방법들은 개념설계나 상세설계과정에서 사용되고 있으나 모두 하나의 기능요구를 해결할 수 있다는 점에서 커다란 시스템설계에 적용하기에는 한계가 있다. 따라서 공리적 설계는 대형 시스템설계로의 적용이 용이하다. 아직은 개발 초기단계로서 많은 사례연구가 발표될 것으로 기대한다. 여기서는 공리적 설계의 사례연구보다는 기본이론의 설명에 주안점을 두었다. 각종 사례들은 참고문헌에 많이 있으므로 참고하기 바란다.

(박경진 교수 : gipark@hanyang.ac.kr)

참고문헌



1. 박경진, 도성희, 서남표, “공리적 설계 방법을 이용한 소프트웨어 시스템의 설계 및 확장 - TV Glass Bulb 설계 소프트웨어를 중심으로-,” 대한기계학회논문집(A) Vol. 23, No. 9, pp. 1536-1549, 1999
2. 차성운, 박경진 역, 공리적 설계 I, 동명사, 2002
3. Altshuller G, And Suddenly the Inventor Appeared, Technical Innovation Center, Massachusetts, 1996
4. Arora JS, Introduction to Optimum Design, 2nd ed., Elsevier, NY, 2004
5. NSF, Axiomatic Design Workshop for Professors, MIT, Massachusetts, 1998
6. Do SH, Park GJ, “Application of Design Axioms for Glass Bulb Design and Software Development for Design Automation,” Journal of Mechanical Design, Transaction of the ASME Vol. 123, No. 3, pp. 322-329, 2001
7. Park GJ, Analytic Methods for Design Practice, Spring-Verlag, 2005 (to be published)
8. Suh NP, The Principles of Design, Oxford University Press, New York, USA, 1990
9. Suh NP, Axiomatic Design : Advances and Applications, Oxford University Press, New York, USA, 2001
10. Taguchi G, Systems of Experimental Design, Kraus International Publications, New York, USA, 1987