최적설계의 개념

이 태 희*
(한양대학교 기계공학부)

최적화(optimization)는 "최선"이란 의미의 라틴어 "OPTIMUS"로부터 유래되었다. 공학에서 최적설계는 가능하면 짧은 시간에 최고 성능을 갖는 제품을 값싸게 생산할 수 있는 설계로 정의한다. 2차 세계대전 이후 가볍고 성능이 우수한 항공기 구조물의 설계를 위하여 시작한 현대 최적설계 기법은 컴퓨터 계산능력의 발전과 함께 실용범위를 점차 확대하고 있다. 본 특집에서는 최적설계의 기본개념을 소개하기 위하여 최적설계의 정식화, 도해 최적화, 최적화의 수치적 기법, EXCEL을 이용한 최적설계 등을 간략하게 소개한다.

1. 최적설계 정식화

일반적으로 설계문제는 소비자의 요구사항으로부터 정의한다. 최적설계문제의 정식화는 설계요구를 수학적 표현으로 변환하는 것이다. 정식화의 구성요소는 설계자가 결정의 해야 할 모든 변수의 집합인 설계변수(design variables), 최적설계의 기준이 되는 가격함수(cost function) 또는 목적함수(objective function)와 설계가 꼭 만족하여야 하는 제한조건(constraint)이 있다. 정식화의 과정은 일반적으로 설계변수의 정의, 목적함수의 설정, 제한조건의 도출 등 세 단계를 거친다.

1.1. 설계변수

설계변수는 두께, 길이, 재료 물성치, 형상 등 설계자가 결정해야 할 모든 변수로 정의한다. 설계변수를 정의할 때는 가능하면 서로 독립적으로 정의하여야 한다. 설계변수의 개수는 적게 사용하는 것이 효과적이다.

*E-mail: thlee@hanyang.ac.kr

1.2. 목적함수

여러 가지 설계 중 최적의 설계를 결정하기 위한 함수가 목적함수이다. 일반적으로 설계는 같은 성능이라면 가격이 낮은 설계가 좋은 설계이므로 목적함수는 많은 경우 가격함수가 될 수 있다. 주로 사용하는 목적함수는 제품의 가격, 이익, 중량, 에너지소모량, 승차감, 최대응력, 고유진동수 등이다. 어떤 경우에는 2개 이상의 목적함수가 있을 수 있다. 예를 들면 구조물의 중량은 최소화하는 동시에 어떤 점의 응력이나 변위를 최소화하고 싶은 경우 등이다. 이렇게 목적함수가 여러 개 존재하는 경우를 다중목적함수라 한다.

1.3. 제한조건

설계가 꼭 만족해야 하는 모든 조건을 제한 조건이라고 한다. 예를 들면 최대응력은 허용응력보다 작아야 한다 등이 되며 가격 또는 성능 등도 제한조건이 될 수 있다. 제한조건은 일반적으로 부등식으로 표현되며 이런 제한조건을 부등식제한조건(inequality constraints)이라고 한다. 등식으로 표현되는 제한조건도 취급할 수 있다.

다음은 최적설계 정식화를 예를 들어 설명한다.

1.3.1. 머그컵 최적설계

그림 1과 같은 원통형 머그컵을 설계하고자 한다. 머그컵의 용량은 최소 900 cm^3 이어야 한다. 머그컵의 재료비를 최소화 하는 설계를 구하여보자. 머그잔의 직경은 5 cm 보다 크고 9 cm 보다는 작아야 하며, 높이는 10 cm 보다 크고 20 cm 보다 작아야 한다. 머그컵을 설계하기 위한 설계변수는 머그컵의 직경(d)과 높이(h)로 정할 수 있다. 머그컵의 재료비는 머그컵의 표면적에 비례하므로 가격

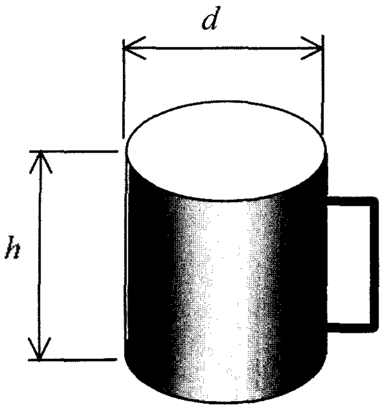

그림 1. 머그컵 설계

함수 또는 목적함수를 머그컵의 표면적으로 할 수 있다. 즉, 목적함수는 다음과 같이 정의한다.

$$f(d, h) = \frac{\pi}{4}d^2 + \pi dh$$

머그컵 용량 제한조건을 수식으로 표현하면

$$V = \frac{\pi}{4}d^2 h \geq 900$$

설계변수에 대한 제한조건을 수식으로 표현하면 다음과 같다.

$$5 \leq d \leq 9$$
$$10 \leq h \leq 20$$

1.3.2. 재활용 처리공장의 최적운영

어떤 가전제품 재활용 회사가 2개의 재활용 처리공장과 2개의 재활용 집하장을 가지고 있다. 재활용 집하장에서는 하루에 250 kg 씩 재활용품이 수집되며, 재활용품 1 kg을 1 km 운송하는데 50원이 든다. 표 1은 재활용 처리공장의 처리능력과 재활용품 집하장부터 처리장까지의 거리를 나타낸다. 이

표 1. 재활용품 집하장으로부터 처리공장까지의 거리

처리공장	거리(km)		1일 처리양
	집하장 1	집하장 2	
A	24.0	20.5	240 kg
B	17.2	18.0	200 kg

재활용 회사에서는 적어도 하루에 400 kg의 재활용품을 처리하여야 한다. 운송비를 가장 적게 들이기 위한 문제로 정의하여 보자.

설계변수를 다음과 같이 정의하자.

x_1 = 집하장 1에서 처리공장 A까지 수송하는 재활용품 무게
x_2 = 집하장 2에서 처리공장 A까지 수송하는 재활용품 무게
x_3 = 집하장 1에서 처리공장 B까지 수송하는 재활용품 무게
x_4 = 집하장 2에서 처리공장 B까지 수송하는 재활용품 무게

다음은 운송비를 설계변수의 함수로 계산하여 보자.

$$cost = 24(50)x_1 + 20.5(50)x_2 + 17.2(50)x_3 + 18(50)x_4$$

이 문제의 제한조건을 고려하여 보자. 재활용 처리공자의 처리량에 대한 제한조건은 다음과 같다.

처리공장 A: $x_1 + x_2 \leq 240$
처리공장 B: $x_3 + x_4 \leq 200$

집하장의 재활용품 수집량 제한조건은 다음과 같다.

집하장 1: $x_1 + x_3 \leq 250$
집하장 2: $x_2 + x_2 \leq 250$

하루에 재활용공장에서 처리하여야 하는 양에 대한 제한조건은 다음과 같다.

$$x_1 + x_2 + x_3 + x_4 \geq 400$$

또한 실제 문제에서는 모든 설계변수는 음수가 될 수 없으므로 다음과 같은 조건을 추가한다.

$$x_i \geq 0, i = 1, \ldots, 4$$

이 문제를 정리하면 다음과 같이 쓸 수 있다

Find x_1, x_2, x_3, x_4
To minimize $cost = 24(50)x_1 + 20.5(50)x_2 + 17.2(50)x_3 + 18(50)x_4$
Subject to
$x_1 + x_2 \leq 240$
$x_3 + x_4 \leq 200$

$$x_1 + x_3 \le 250$$
$$x_2 + x_2 \le 250$$
$$x_1 + x_2 + x_3 + x_4 \ge 400$$
$$x_i \ge 0,\ i\ =\ 1, \ldots, 4$$

일반적으로 최적화 문제를 정의하면 다음과 같이 수식으로 표현한다.

Find an n-design-variable vector $\mathbf{x} \in R^n$
to minimize a cost/objective function $f(\mathbf{x})$
subject to the m inequality constraints
$g_i(\mathbf{x}) \le 0;\ i = 1,2,\ldots,m$
and the p equality constraints
$h_j(\mathbf{x}) = 0;\ i = 1,2,\ldots,p$

2. 도해 최적화

머그컵 최적설계처럼 설계변수가 2개인 경우는 2차원 평면에 설계영역을 도해하여 최적점을 찾을 수 있다. 설계제한조건인 설계변수의 상하한 값과 용량을 그림으로 도식하고 여기에 등가격선을 그려 최적점을 찾을 수 있다. 그림 2는 MATLAB을 이용하여 제한조건과 목적함수를 그림으로 도식한 것이다. 우측 상단의 삼각형 영역이 모든 설계제한조건을 만족하는 영역이며 이 영역을 가용영역(feasible region)이라 부른다. 한편, 설계 제한조건을 만족하지 않는 영역을 비가용 영역이라 한다. 가용영역에서 목적함수의 값이 가장 작은 곳은 용량제한조건선과 지름 상한선이 만나는 점이 되며, 이 점이 최적설계가 된다.

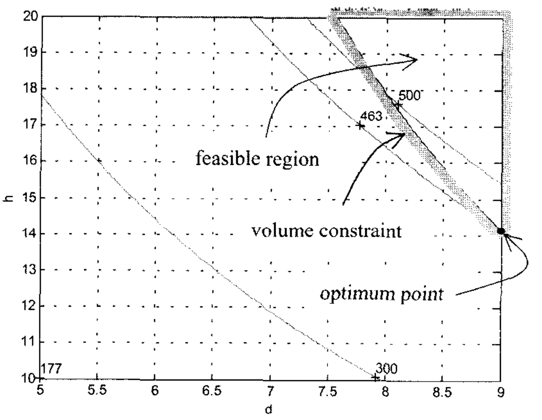

그림 2. 머그컵 최적설계문제의 제한조건과 목적함수

3. 최적설계의 해법

3.1. 해석적 최적화 방법

최적설계는 설계에 주워진 모든 제한조건을 만족하면서 가격이 가장 낮은 설계를 의미한다. 따라서 이런 최적설계문제를 정식화 하면 다음과 같이 수학적 문제로 변환할 수 있다.

Minimize $f(\mathbf{x});\ \mathbf{x} \in R^n$

Subject to

$$g_i(\mathbf{x}) \le 0;\ i\ =\ 1\ \text{ to } m \qquad (1)$$
$$h_j(\mathbf{x})\ =\ 0;\ j\ =\ 1\ \text{ to } p$$

즉, m개의 부등호제한조건 $g_i(\mathbf{x})$와 p개의 등호제한조건 $h_j(\mathbf{x})$을 만족하면서 목적함수 $f(\mathbf{x})$를 최소화 하는 n개의 실수인 설계변수 $\mathbf{x} \in R^n$ 를 구하는 문제가 된다. 이런 최적화문제는 다음의 카러쉬쿤터거 필요조건(Karush-Khun-Tucker necessary condition)을 만족하는 해를 구함으로써 해석적으로 풀 수 있다.

라그란지승수(u_i, v_j)를 도입하여 라그란지 함수를 정의하자.

$$L\ =\ f(x) + \sum_{j=1}^{p} v_j h_j + \sum_{i=1}^{m} u_i(g_i + s_i^2) \qquad (2a)$$

최적값의 필요조건은 다음과 같이 유도된다.

$$\nabla L\ =\ \nabla f(x) + \sum_{j=1}^{p} v_j \nabla h_j + \sum_{i=1}^{m} u_i \nabla g_i = 0 \qquad (2b)$$

$$h_j\ =\ 0,\ j\ =\ 1, \ldots, p \qquad (2c)$$

$$g_i + s_i^2\ =\ 0,\ i\ =\ i, \ldots, m \qquad (2d)$$

$$u_i s_i\ =\ 0,\ i\ =\ 1, \ldots, m \qquad (2f)$$

$$u_i \ge 0 \qquad (2g)$$

즉 총 (n+p+2m)개의 방정식을 유도하였다. 우리가 구해야 하는 변수는 설계변수 n개, 라그란지승

수 m+p개, 부등식을 등식으로 변환하기 위한 잉여변수 s_i가 m개 등 (n+p+2m)개이다. 따라서 식 (2b)~(2f)의 연립방정식을 풀면 최적값의 필요조건을 구할 수 있다. 이런 필요조건을 카라쉬쿤터커 필요조건이라 한다.

3.2. 수치적 최적화 방법

함수의 비선형성이나 제한조건이 많은 경우는 해석적 방법으로 해를 구하는 것은 매우 어렵다. 많은 공학설계문제에서는 해석적 방법이 적절하지 못하다. 따라서 이런 공학문제의 최적화를 위한 체계적인 수치해석법이 필요하다. 수치해석법에서는 초기설계를 설정하고 이 설계변수를 최적성조건을 만족할 때까지 반복적으로 구한다. 이 방법을 이용하면 초기설계 근방의 국소최적설계에 수렴하게 되며 이를 국소최적설계라 한다. 먼저 수치해석법의 일반적 개념에 대하여 소개하고, 단변수 최적화 수치해석법, 다변수 비제한최적화 수치해석법에 대하여 소개를 한 후, 제한최적화 수치해석법에 대하여 소개한다.

3.2.1. 수치해석법의 일반개념

3.2.1.1. 일반 알고리듬

최적화 문제의 일반 알고리듬은 다음의 반복과정으로 표현된다.

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k)};\ k = 0, 1, 2, \ldots \tag{3}$$

이 식에서 상첨자(k)는 반복수를 나타내며, $\Delta\mathbf{x}^{(k)}$는 (k)번째 설계단계에서의 설계변수 변화량을 나타내며 벡터성분인 탐색방향 $\mathbf{d}^{(k)}$와 양의 스칼라인 이동거리 $\alpha^{(k)}$로 분리하여 다음과 같이 표현한다.

$$\Delta\mathbf{x}^{(k)} = \alpha^{(k)}\mathbf{d}^{(k)} \tag{4}$$

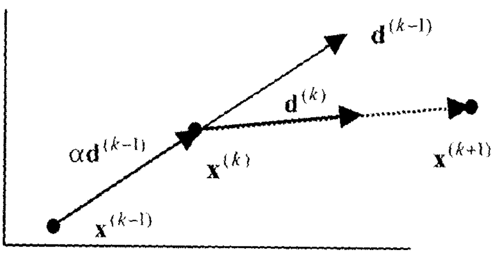

그림 3. 최적설계의 반복단계

즉,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)} \tag{5}$$

비제한최적설계문제나 제한최적설계문제에 모두 적용할 수 있는 일반 알고리듬은 다음과 같이 정리할 수 있다.

- 단계 1: 초기 설계변수 $\mathbf{x}^{(0)}$를 설정하며, 반복회수 $k=0$으로 한다.
- 단계 2: 탐색방향 $\mathbf{d}^{(k)}$를 계산한다.
- 단계 3: 알고리듬의 수렴성을 검토하여 수렴하면 종료하고 그렇지 않으면 계속한다.
- 단계 4: 양의 이동거리 $\alpha^{(k)}$를 계산한다.
- 단계 5: 새로운 설계변수를 다음과 같이 구한다: $\mathbf{x}^{(k+1)}=\mathbf{x}^{(k)}+\alpha^{(k)}\mathbf{d}^{(k)}$. $k=k+1$으로 두고 단계 2로 간다.

일반 알고리듬에서 알 수 있듯이 국소최적화 알고리듬은 탐색방향과 이동거리의 계산방법에 따라 여러 가지 알고리듬이 개발될 수 있다.

3.2.1.2. 탐색방향벡터의 특징

만약 k번째의 설계변수 $\mathbf{x}^{(k)}$가 아직 목적함수 $f(\mathbf{x})$의 최소점에 도달하지 않았다고 가정하면, 보다 작은 목적함수 값을 갖는 설계변수 $\mathbf{x}^{(k+1)}$이 존재하게 된다. 즉,

$$f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)}) \tag{6}$$

한편 $f(\mathbf{x}^{(k+1)})$를 식 (5)를 이용하여 $\mathbf{x}^{(k)}$에 대하여 선형테일러급수로 전개하면 다음과 같다.

$$f(\mathbf{x}^{(k+1)}) = f(\mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}) = f(\mathbf{x}^{(k)}) + \mathbf{c}^{(k)} \cdot \alpha^{(k)}\mathbf{d}^{(k)} \tag{7}$$

여기서 $\mathbf{c}^{(k)} = \nabla f(\mathbf{x}^{(k)})$는 $f(\mathbf{x})$의 경사도이며, $(\cdot)$는 두 벡터의 내적이다. 식 (7)을 식 (6)에 대입하면

$$\mathbf{c}^{(k)} \cdot \mathbf{d}^{(k)} < 0 \tag{8}$$

따라서 목적함수의 값이 감소하기 위하여 탐색방향벡터는 목적함수의 경사도와 90°와 270° 사이에 있어야 한다.

3.2.1.3. 이동거리

만약 탐색방향이 목적함수를 감소시키는 방향이

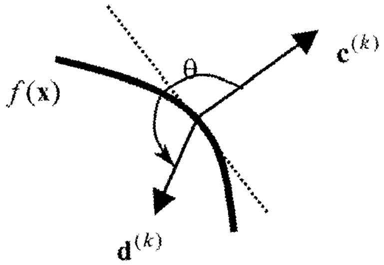

그림 4. 탐색방향과 경사도

라고 가정을 하면

$\mathbf{x}^{(k+1)}=\mathbf{x}^{(k)}+\alpha^{(k)}\mathbf{d}^{(k)}$

이 되며, 목적함수는 다음과 같이 표현된다.

$$f(\mathbf{x}^{(k+1)})=f(\mathbf{x}^{(k)}+\alpha\mathbf{d}^{(k)})=f(\alpha) \tag{9}$$

즉, 목적함수의 변수는 단일변수가 되어 다음과 같이 스칼라 최소화 문제로 변환된다.

$$\text{Minimize } f(\alpha) \tag{10}$$

여기서

$$f(0)=f(\mathbf{x}^{(k)}),\ f'(0)=\mathbf{c}^{(k)}\cdot\mathbf{d}^{(k)}<0 \tag{11}$$

따라서 함수 $f(\alpha)$는 그림 5와 같이 주어진 구간 내에서 유일한 최소값이 존재하는 단일극함수(unimodal function)의 성질을 갖고 있다. 이런 단일변수함수는 일반적으로 선탐색을 이용하여 해를 반복적으로 구한다.

3.2.1.4. 알고리듬의 수렴성

최적화의 수치방법의 성공은 제시한 알고리듬이 최적점에 수렴한다는 것을 보장하여야 한다. 이렇게 보장된 알고리듬은 설계초기값에 관계없이 국소최적점으로 수렴을 하게 된다. 따라서 좋은 알고리듬은 그 수렴성이 수학적으로 보장이 되어야 한다. 또 수렴성이 보장이 된 알고리듬의 효율성은 수렴속도로 판단한다. 수렴속도는 만족할 만한 최적해를 구할 때까지 반복회수와 함수재산회수로 측정한다.

3.2.2. 비제한 최적화 수치해석법

비제한 최적화 문제는 제한조건이 존재하지 않는 문제로 다음과 같이 정의된다.

$$\textit{Minimize } f(\mathbf{x});\ \mathbf{x}\in R^n \tag{12}$$

비제한 최적화 문제는 이동거리계산 부문제와 탐색방향계산 부문제로 나누어 계산한다.

3.2.2.1. 이동거리 계산

이동거리의 계산은 단변수 함수의 최적화문제이며 또 일정 구간에서 단일극함수의 성질을 갖고 있다. 이런 단일극함수의 수치해법은 주로 선탐색법을 사용하며 여기에서는 선탐색법 중 가장 간단한 등간격탐색법을 소개한다.

그림 6과 같이 구간 $0\le\alpha\le\bar{\alpha}$에서 함수 $f(\alpha)$는 단일극함수이므로 전역최소값을 갖는다. 이 구간을 최소점이 있는 구간에 대한 상한값과 하한값을 결정하여 불확정구간 $I=[\alpha_L,\ \alpha_U]$를 정의한다. 이 불확정구간을 반복적으로 줄여가는 방법이 선탐색이며 불확정구간을 등간격으로 분할하는 방법이 등간격탐색법이다. 먼저 작은 수 δ를 정하고 $f(\alpha)$를 다음과 같이 계산한다.

$$f(\delta)>f(2\delta)>\ldots>f((q-1)\delta)>f(q\delta) \tag{13}$$

이처럼 함수가 계속 감소하면 최소점을 통과하지 않는 것이다. 그러나 다음 단계에서

$$f(q\delta)<f((q+1)\delta) \tag{14}$$

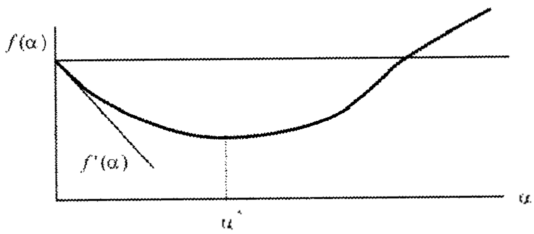

그림 5. 단일극함수의 개략도

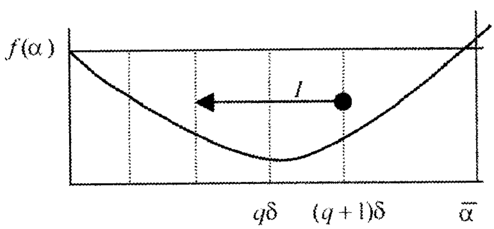

그림 6. 등간격탐색의 불확정구간

로 함수가 증가를 한다면 이제는 최소점을 통과한 것이다. 그러면 단일극함수의 최소점을 포함하는 불확정 구간은 다음과 같이 정의한다.

$$\alpha_L=(q-1)\delta,\ \alpha_U=(q+1)\delta$$
$$I=[\alpha_L,\ \alpha_U]=2\delta \tag{15}$$

따라서 이 불확정구간 내에 최소점이 존재한다. 그러면 탐색간격을 줄여서 이 불확정구간을 반복적으로 줄여가는 방법이 등간격탐색법이다(그림 6).

선탐색법으로는 등간격탐색법 외에도 황금분할탐색, 다항식보간법 등이 있다.

3.2.2.2. 탐색방향 계산

목적함수 $f(\mathbf{x})$의 최소값을 반복적으로 구할 때 설계변수는 다음과 같이 반복적으로 계산된다.

$$\mathbf{x}^{(k+1)}=\mathbf{x}^{(k)}+\alpha^{(k)}\mathbf{d}^{(k)} \tag{16}$$

이 때 함수 $f(\mathbf{x}^{(k+1)})$를 $\mathbf{x}^{(k)}$에 대해서 선형테일러급수로 전개하면 다음과 같다.

$$f(\mathbf{x}^{(k+1)}) \cong f(\mathbf{x}^{(k)})+\alpha^{(k)}\mathbf{c}^{(k)}\cdot\mathbf{d}^{(k)} \tag{17}$$

이 경우 비제한 최적화 문제는 다음과 같이 표현될 수 있다.

$$\text{Minimize } \bar{f}\ (\mathbf{d}^{(k+1)}) \tag{18}$$

여기서

$$\bar{f}\,(\mathbf{d}) \equiv f(\mathbf{x}^{(k)})+\alpha^{(k)}\mathbf{c}^{(k)}\cdot\mathbf{d}^{(k)} \tag{19}$$

식 (19)에서 $f(\mathbf{x}^{(k)})$와 $\alpha^{(k)}$가 이미 계산된 상수이므로 다음과 같이 탐색방향 $\mathbf{d}^{(k)}$를 구하는 문제로 다시 쓸 수 있다.

$$\text{Minimize } \mathbf{c}^{(k)}\cdot\mathbf{d}^{(k)} \tag{20}$$

이 문제는 선형문제이므로 유용영역이 유계영역(unbounded feasible region)이 아닐 수 있으므로 해가 존재하지 않을 수 있다. 따라서 다음과 같은 제한조건을 추가하면 유계 유용영역 내에서 최적화 문제를 구할 수 있다.

$$\|\mathbf{d}^{(k)}\|=1,\ \|\mathbf{d}\|=\sqrt{\mathbf{d}^T\mathbf{H}\mathbf{d}} \tag{21}$$

탐색방향 결정 문제의 카라쉬쿤터커 필요조건을 구하면 다음과 같다.

$$L = \mathbf{c}^{(k)}\cdot\mathbf{d}^{(k)}+\nu(\sqrt{\mathbf{d}^T\cdot\mathbf{H}\mathbf{d}}-1),\ \frac{\partial L}{\partial \mathbf{d}^{(k)}} = 0 \tag{22}$$

$$\mathbf{H}\mathbf{d}^{(k)} = -\frac{1}{\nu}\mathbf{c}^{(k)} \tag{23}$$

(1) 최대경사법(steepest descent algorithm)

식 (21)의 탐색방향의 이동한계를 정의하는 수식에서 행렬 $\mathbf{H}$를 단위행렬로 대치하면 즉, $\mathbf{H}=\mathbf{I}$, 탐색방향은 식 (23)로부터 다음과 같이 정의된다.

$$\mathbf{d}^{(k)}=-\mathbf{c}^{(k)} \tag{24}$$

즉, 탐색방향은 함수 $f(\mathbf{x})$의 경사도의 반대방향으로 결정하는 방법으로 최대경사법이라 한다. 이 방법은 1847년 코쉬(Cauchy)에 의해서 제안되었으며 전역수렴성이 보장되지만 수렴속도가 매우 낮은 단점도 있다.

(2) 뉴톤법(Newton's method)

식 (21)의 탐색방향의 이동한계를 정의하는 수식에서 행렬 $\mathbf{H}$를 함수 $f(\mathbf{x})$의 $\mathbf{x}^{(k)}$에 대한 2차미분 행렬인 헤시안행렬(Hessian Matrix)을 사용하면 탐색방향은 다음과 같이 계산된다.

$$\mathbf{d}^{(k)}=-\mathbf{H}^{-1}\mathbf{c}^{(k)} \tag{25}$$

이 방법은 2차미분을 사용하므로 수렴속도는 빠르나 목적함수의 헤시안을 구하여야 하며 또한 이 헤시안행렬의 역행렬을 구해야 하므로 계산시간이 많이 걸린다. 또한 헤시안행렬이 양정행렬(positive definite matrix)이 아닌 경우는 전역수렴성이 보장되지 않는다. 이런 방법을 뉴톤법이라 한다.

3.2.2.3. 기타 알고리듬

목적함수의 미분값을 사용하지 않고 최적점을 반복적으로 구하는 패턴탐색법(Hooke and Jeeves, 1961)이 있으며, 1차 미분값과 수렴이력을 고려하여 탐색방향을 찾는 공액경사법(Fletcher and Reeves, 1964) 등이 있다. 뉴톤법과 최대경사법을 복합하여 방향탐색을 하는 마콰드법(Marquardt, 1963)이 있다. 2차 미분값을 사용하면 수렴속도는 개선되나 헤시안행렬의 계산이 많은 시간이 요구되므로 헤시안행렬을 근사적으로 구하는 유사뉴톤법이 제안되

었으며 DFP(Davidon, 1959; Fletcher and Powell, 1963)법과 BFGS(Broyden-Fletcher-Goldfarb-Shanno)법이 있다.

3.2.3. 제한최적화 수치해석법

제한최적설계의 문제는 부등식제한조건과 등식제한조건을 포함하는 최적화 문제로 다음과 같이 정의된다.

Minimize $f(\mathbf{x})$; $\mathbf{x} \in R^n$

Subject to

$$g_i(\mathbf{x}) \le 0; \quad i = 1 \ to \ m \qquad (26)$$
$$h_j(\mathbf{x}) = 0; \ j = 1 \ to \ p$$

제한최적설계 문제의 수치해법도 비제한최적설계의 경우와 같이 설계변수를 체계적으로 변경하여 반복적으로 해를 구할 수 있다. 즉,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k)}; \ k = 0, 1, 2, \ldots \qquad (27)$$

따라서 제한최적설계 문제도 탐색방향결정과 이동거리결정 부문제를 포함하게 된다. 이 두 부문제의 해는 비제한최적설계 문제의 경우와는 다르게 목적함수와 제한조건의 값 뿐만 아니라 현재 설계점의 경사도에 따라 달라진다. 현재 설계점에서 부등호제한조건은 활성화, 입시론(ε)활성화, 위배 또는 만족 상태가 된다. 등호제한조건은 만족 또는 위배상태가 된다. 제한조건의 상태를 수식으로 표현하면 다음과 같다.

활성화제한조건(active constraint):
$g_i(x) = 0$; $h_j(x) = 0$, $j = 1 : p$

만족제한조건(inactive constraint):
$g_i(x) < 0$

위배제한조건(violated constraint):
$g_i(x) > 0$; $h_j(x) \ne 0$, $j = 1 : p$

입시론활성화제한조건(ε-active constraint):
$g_i(x) < 0$ but $g_i(x) + \varepsilon \ge 0$

제한최적화 문제의 경우도 현재의 설계점에서 근사화하여 최적화를 수행하며 일반적으로 선형근사화를 하거나 2차 근사화를 하여 최적화를 수행한다. 따라서 근사화 정도에 따라 순차선형계획법(sequential linear programming)과 2차 계획법(quadratic programming)이 있다.

3.2.3.1. 순차선형계획법

설계변수는 탐색방향과 이동거리로 분할하여 표현한다.

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)} \qquad (28)$$

목적함수와 제한조건 를 $\mathbf{x}^{(k)}$에 대해서 선형테일러급수로 전개하면 다음과 같다.

$$f(\mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k)}) \cong f(\mathbf{x}^{(k)}) + \mathbf{c}^T\Delta\mathbf{x}^{(k)} \qquad (29)$$

$$g_i(\mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k)}) \cong g_i(\mathbf{x}^{(k)}) + \nabla g_i^T\Delta\mathbf{x}^{(k)} \le 0; \quad i = 1:m \qquad (30)$$

$$h_j(\mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k)}) \cong h_j(\mathbf{x}^{(k)}) + \nabla h_j^T\Delta\mathbf{x}^{(k)} = 0; \quad j = 1:p \qquad (31)$$

이제 선형근사 최적화 부문제를 정리하면 다음과 같이 탐색방향 결정 부문제를 정의할 수 있다.

$$\text{Minimize } \bar{f}(\mathbf{d}^{(k)}) = \mathbf{c}^T\mathbf{d}^{(k)} \qquad (32)$$

Subject to

$$\bar{g}_i(\mathbf{d}^{(k)}) = g_i(\mathbf{x}^{(k)}) + \nabla g_i^T\mathbf{d}^{(k)} \le 0; \ i = 1:m \qquad (33)$$

$$\bar{h}_j(\mathbf{d}^{(k)}) = h_j(\mathbf{x}^{(k)}) + \nabla h_j^T\mathbf{d}^{(k)} = 0; \ j = 1:p \qquad (34)$$

탐색방향 결정 부문제는 선형문제이므로 유용영역이 유계영역이 아닐 수 있으므로 해가 존재하지 않을 수 있다. 따라서 그림 7과 같은 이동한계를 추

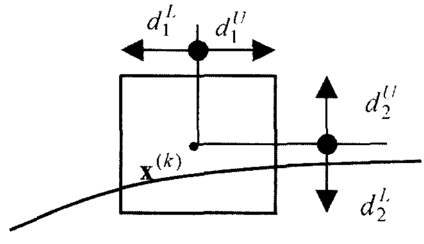

그림 7. 순차선형계획법에서 설계변수 이동한계

가하면 유계 유용영역 내에서 최적화 문제를 정의할 수 있다.

$$d_i^L \le d_i \le d_i^U \tag{35}$$

이 방법을 순차선형계획법이라 부른다. 이 순차선형계획법은 수렴속도가 매우 늦으며, 수렴의 강건성이 보장되지 않는다. 또한 수렴속도는 이동한계의 크기에 좌우된다.

3.2.3.2. 이차계획법

이차계획법에서는 선형근사화를 2차 근사화까지 확장하여 수렴속도를 증가시키는데 목적이 있다. 2차근사화의 대상은 이동한계를 2차 함수로 표현하는 이동거리 제한조건과 목적함수를 2차 근사화하는 2차 부문제가 있다.

(1) 이차이동거리 제한조건

순차계획법의 성능은 이동한계의 크기에 의존한다. 또한 매 반복 회마다 이동한계를 조정해야 한다. 이런 문제점을 극복하기 위하여 이동거리한계를 다음과 같은 놈(norm)으로부터 정의하여 식 (21) 대신 사용한다.

$$0.5\mathbf{d}^T\mathbf{d} \le \xi^2 \tag{36}$$

이는 반경이 $\sqrt{2}\,\xi$ 인 초구(hypersphere)가 된다. 이 문제에서 목적함수 및 제한조건의 근사화는 선형근사화 하였으나 이동거리한계가 비선형이므로 탐색방향 결정 부문제는 비선형문제가 된다.

(2) 2차 계획 부문제

2차 계획 부문제는 2차 근사 목적함수와 선형근사 제한함수로 구성된다. 또한 이런 2차 계획문제(quadratic problem)의 수치해법은 매우 잘 개발이 되어 있으므로 이를 이용하면 풀 수 있다. 2차계획 부문제는 항상 해가 존재하며, 문제가 엄밀하게 볼록(convex)이면 유일해가 존재하는 장점이 있다. 2차 계획 부문제를 풀기위한 상업용 프로그램도 다수 나와 있다. 여기에서는 2차 부문제의 수치적 해법보다는 2차 부문제의 구성에 중점을 두고 소개를 하고자 한다.

2차 부문제의 정식화는 다음과 같이 탐색방향 **d**를 구하는 문제로 정의할 수 있다.

$$\text{Minimize } \bar{f}\,(\mathbf{d}^{(k)}) = \mathbf{c}^T\mathbf{d}^{(k)} + 0.5\mathbf{d}^T\mathbf{d} \tag{37}$$

Subject to

$$\bar{g}_i\,(\mathbf{d}^{(k)}) = g_i(\mathbf{x}^{(k)} + \nabla g_i^T\mathbf{d}^{(k)} \le 0\,;\ i = 1:m \tag{38}$$

$$\bar{h}_j\,(\mathbf{d}^{(k)}) = h_j(\mathbf{x}^{(k)}) + \nabla h_j^T\mathbf{d}^{(k)} = 0;\ j = 1:p \tag{39}$$

2차 계획 부문제는 카러쉬쿤터거 필요조건을 이용하면 선형문제로 변환이 가능하며 이를 선형계획법을 이용하여 풀 수 있다.

3.2.3.3. 제한유사뉴톤법

2차 계획법에서는 탐색방향 결정 부문제에서 제한조건함수의 선형근사화만을 이용하였다. 이런 부문제를 기반으로 한 알고리듬은 수렴속도가 늦다. 이를 해결하기 위하여 목적함수 및 제한조건을 2차 근사화 하여 문제를 푸는 것은 원래 비선형 문제를 푸는 것만큼 어려운 일이다. 따라서 헤시안행렬을 직접 구하지 않고 근사적으로 구하는 방법을 이용한 유사뉴톤방법(constrained quasi-Newton method)이 개발되었다. 유사뉴톤법은 최근에 개발이 되었으며 가장 효율적이고 신뢰성 있는 방법으로 알려져 있다.

3.2.3.4. 변환법

변환법은 제한최적설계문제를 벌칙함수(penalty function)를 이용하여 복합함수로 구성하여 비제한 최적설계문제로 변화하고 이의 해를 구하는 방법이다. 벌칙함수는 다음과 같이 정의한다.

$$\Phi(\mathbf{x}, \mathbf{r}) = f(\mathbf{x}) + P(\mathbf{h}, \mathbf{g}, \mathbf{r}) \tag{40}$$

여기서 함수 P를 벌칙함수라 하는데 이의 선택에 따라서 벌칙함수법과 장애함수법이 있다. 벌칙함수법(penalty function method)은 제한조건의 위배가 있으면 큰 벌칙을 목적함수에 더하도록 하여 해를 유용영역 경계로 유도하는 방법이다. 여러 가지 벌칙함수를 정의할 수 있으며 잘 알려진 벌칙함수는 다음과 같다.

$$P = r\left[\sum_i [h_i(x)]^2 + \sum_j \langle g_j(\mathbf{x})\rangle^2\right] \tag{41}$$

여기서

$$\langle\alpha\rangle = \max(0, \alpha) = \begin{cases} \alpha & if\ \alpha \geq 0 \\ 0 & if\ \alpha < 0 \end{cases} \tag{42}$$

만약 등호제한조건이 만족하지 않고, 부등호제한조건이 위배상태라면 벌칙함수 P는 양의 값을 갖는다. 이 방법의 출발은 유용영역이나 불용영역에서 할 수 있다.

장애함수법(barrier function method)은 부등호제한조건에서만 사용할 수 있다. 잘 알려진 장애함수는 로그함수와 역함수가 있다.

$$P = -r\ \ln[-g(\mathbf{x})] \tag{43}$$

$$P = -\frac{1}{r}\left[\frac{1}{g(\mathbf{x})}\right] \tag{44}$$

여기서 부등호제한조건이 활성화 되면 장애함수 P는 무한대가 된다. 따라서 반복과정은 유용영역 점에서 출발하여야 하며 반복과정 중 장벽을 넘어갈 수 없으므로 불용영역으로 갈 수가 없다.

3.2.4.5. 기타 알고리듬

유용방향법(*feasible direction method*)은 제한최적화 문제를 푸는 방법 중 가장 오래된 방법이다. 이 방법의 기본원리는 한 유용설계에서 보다 개선된 유용설계로 이동한 방법이다. 따라서 유용설계 $\mathbf{x}^{(k)}$가 주어지면 탐색방향 $\mathbf{d}^{(k)}$를 결정하여 매우 작은 이동거리에 대하여 새로운 설계는 유용하고 목적함수는 감소하여야 한다는 조건을 만족하도록 한다. 일단 탐색방향이 결정되면 선탐색을 수행하여 이동거리를 결정한다.

경사투영법(*gradient projection method*)은 임의의 설계점에서 출발한다. 만약 이 점이 유용영역 내부에 존재하면 제한경계에 도달할 때까지 목적함수의 최대경사방향을 이용한다. 만약 출발점이 불용영역에 있으면 제한집합에 도달할 때까지 제한보정단계를 적용한다. 만약 점이 경계상에 있으면 제한면에 접선인 방향이 탐색방향이 된다. 이 방향은 목적함수의 최대강하를 접선면에 투영함으로써 계산된다. 이 경우 제한면에 접선인 방향이므로 새로운 점은 불용영역에 존재한다. 따라서 유용영역에 도달하기 위하여 여러 수정단계가 요구된다.

일반축약경사법(*generalized reduced gradient method*)은 축약경사법을 확장하여 비선형부등제한조건을 다룰 수 있도록 한 방법이다(Abadie, 1970). 이 방법에서는 탐색방향이 알려지면 미세 이동에 대하여 현재 위배제한조건이 그대로 위배되게 둔다. 만약 위배제한조건이 제한함수의 비선형성으로 만족하지 않으면 제한조건의 경계로 환원하기 위하여 뉴톤-랩슨법을 사용한다. 따라서 이 방법은 경사투영법과 유사하다.

지금까지 소개한 국소최적설계의 알고리듬을 정리하면 다음과 같다.

변수	구분		방법
단변수	선탐색법		등간격탐색법 Fibonacci 법 황금분할법 다항식보간법
다변수	비제한 최적설계	직접탐색법	패턴 탐색법 (Hook-Jeeves) Rosenbrock
		경사법	최대경사법 (Steepest descent method) Newton법 공액경사법 (Fletcher-Reeves) DFP법 BFGS법
	제한최적 설계	직접법	순차선형계획법 2차계획법 제한유사뉴톤법 유용방향법 (Wolfe, 1963) 경사투영법 (Rosen, 1961) 축약경사법 (Reduced Gradient Method)
		변환법	벌칙함수법
			장애함수법

4. EXCEL을 이용한 머그컵 최적설계

머그컵 최적설계 문제를 EXCEL을 이용하여 최적설계의 해를 구하여 보자. 먼저 EXCEL의 셀에 설계변수를 설정하고 그림 8과 같이 목적함수를 설계변수(셀)의 함수로 표현한다. 제한조건도 같은 방법으로 설계변수의 함수로 정의한다. EXCEL의 서식메뉴의 "해 찾기"를 선택하면 그림 9와 같은 창이 나타난다.

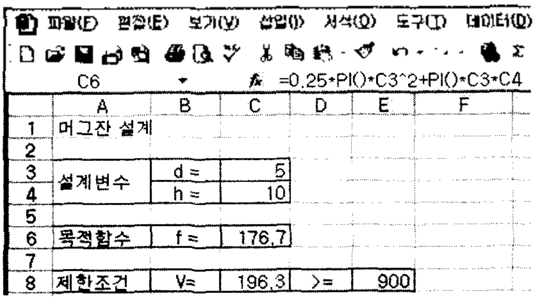

그림 8. 머그컵 설계를 위한 목적함수 정의

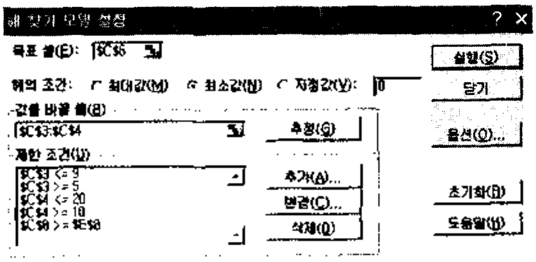

그림 9. 해 찾기 창

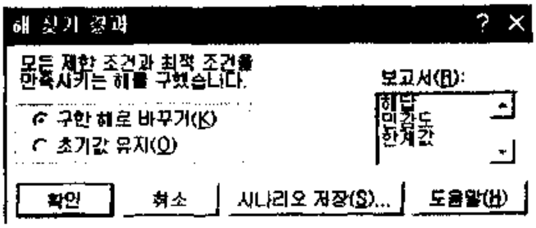

그림 10. 해 찾기 결과 창

해 찾기 창에서 목표셀은 목적함수가 정의된 셀을 선택하며, 목적함수의 성질에 따라서 최대값, 최소값 또는 지정값을 설정한다. 값을 바꿀 셀은 설계변수를 선택한다. 다음은 제한조건을 추가메뉴를 이용하여 입력한다. 목적함수, 설계변수, 제한조건의 설정이 완성되면 실행 단추를 클릭하면 그림 11과

	A	B	C	D	E
1	머그잔 설계				
2					
3	설계변수	d =	9		
4		h =	14.15		
5					
6	목적함수	f =	463.6		
7					
8	제한조건	V=	900	>=	900

그림 11. 머그컵 최적설계의 결과

같은 해 찾기 결과 창이 나타나며, 최적화 수행 결과가 나타난다. 만족하는 해를 구하지 못한 경우는 문제의 정의를 다시 검토하거나 초기값을 변경하여 재 수행을 하여야 한다.

머그컵 설계의 경우 지름이 9.0 cm, 높이가 14.15 cm 인 설계를 구할 수 있었으며 이 때 컵의 용량은 900 cm^3가 되어 설계제한조건을 모두 만족한다.

최적설계란 설계의 방법을 체계적으로 정리하여 수학적 문제로 변환하고 이를 최적화기법을 적용하여 논리적으로 좋은 설계에 접근하는 방법이다. 실제로 당면하는 문제들을 설계하기 위해서는 다양한 최적설계이론이 필요하다. 이번 호에서는 최적설계에 대한 기본 개념을 정리하고 주변에 쉽게 접할 수 있는 개인용 컴퓨터와 프로그램을 사용하여 최적설계를 실행할 수 있음을 설명하였다. 그러나, 실제 설계문제에서 최적설계의 정식화와 이의 해를 구하고 해의 물리적 의미를 부여하는 것은 많은 경험과 지식이 요청되므로 전문가에게 의뢰하는 것이 바람직하다. 여러분의 설계에 최적설계 기법을 도입하여 체계적이고 최적화 된 설계를 도출하는데 작은 보탬이 되길 기대한다.