[2008][07-2]

Computer aided ship design

Part 3. Optimization Methods

October 2008

Prof. Kyu-Yeul Lee

Department of Naval Architecture and Ocean Engineering,
Seoul National University of College of Engineering

Advanced
Ship
Design
Automation
Laboratory

참고 문헌

- **이규열, 노명일, 차주환, 전산선박설계**, 3th Ed., 2003.9
- Arora, J. S., Introduction to Optimum Design, 2nd Ed., Elsevier, 2004
- Michael G. Parsons, Optimization Methods For Use In Computer-Aided Ship Design, The society of naval architects and marine engineers, 1975

최적 설계 문제의 예

비제약 최적화 문제

- **다음의 다섯 가지 비제약 최적화 방법**(unconstrained optimization method)**을 이용하여** 2**변수 함수의 최소점을 구하시오. 단, 시작점 x**$^{(0)}$ = (0, 0), convergence tolerance ε = 0.001**이며, x**$^{(3)}$**까지 구하시오.**

$$\textit{Minimize} \;\; f(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$

➡ **미지수 2개인 최적화 문제**

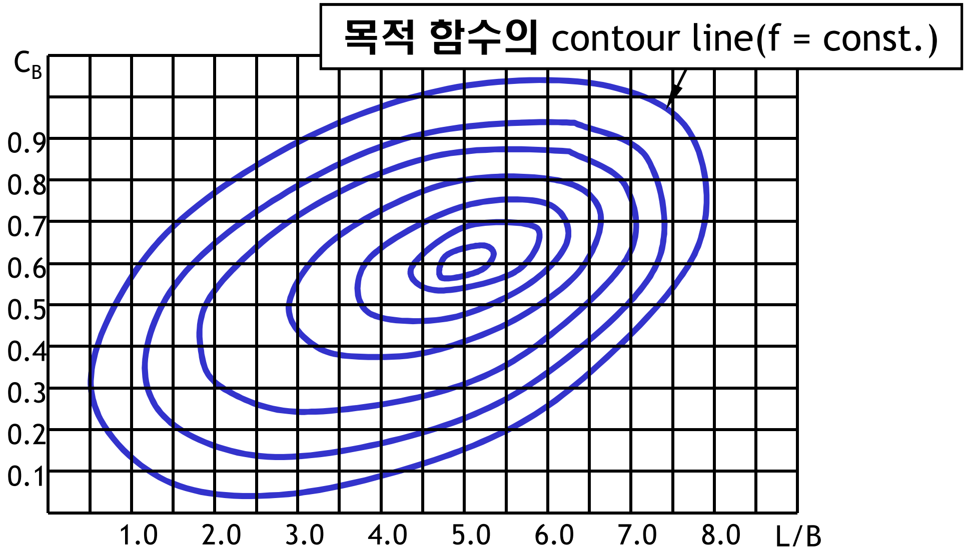
- Steepest descent **방법(최속 강하법, 최대 경사법)**
- Conjugate gradient **방법(공액 경사도 방법)**
- Newton **방법**
- Davidon-Fletcher-Powell(DFP) **방법**
- Broydon-Fletcher-Goldfarb-Shanno(BFGS) **방법**

비제약 최적화 문제 - 응용 문제

문제: **어느 선박의 상대적 건조비를** L/B**와** C_B**의 함수로 다음 그림과 같이 표시할 수 있다고 하면 건조비가 최소가 되는** L/B**와** C_B**의 값을** Hooke & Jeeves**의 탐사법과** Nelder & Mead**의** Simplex **방법을 이용하여 최적점을 구하고 그 과정을 그림에 각각 표시하시오.**

- ✓ Hooke & Jeeves**의 탐사법**
  - **출발점:** L/B = 1, C_B = 0.1
  - **출발점의** step size: $\Delta(L/B) = 0.5$, $\Delta(C_B) = 0.1$
- ✓ Nelder & Mead**의** Simplex **방법**
  - **출발 모서리점:** (L/B, CB) = (1, 0.1), (1.5, 0.1), (1.5, 0.2)
  - **중지 기준:** 0.01

미지수 2개인 최적화 문제 ⇦



선형 계획법 문제 – 응용 문제

- 선형 계획법을 이용한 최적 밸러스트 탱크 배치(1/2)

어느 선박의 밸러스트 탱크(ballast tank)는 다음과 같은 특성을 갖고 있다. 이 선박의 현재 트림(trim) 및 GM은 각각 2m, 0.1m인데, 밸러스트 탱크들을 부분적으로 또는 완전히 채워서 트림을 0.0mm로 GM을 0.5m 이상으로 유지하고자 한다. 어떤 탱크에 얼마의 양의 밸러스트 수(ballast water)를 채우면 되겠는가? 물론 이때 밸러스트 수의 양을 최소로 해야 한다.

탱크	탱크의 용량	1,000m^3 탱크 용량의 트림 변화량(m)	1,000m^3 탱크 용량의 GM 변화량(m)
1	500	0.0	0.1
2	1,000	-2.0	0.1
3	1,500	0.5	0.5
4	2,000	-1.0	0.2
5	2,500	-0.2	0.4

선형 계획법 문제 - 응용 문제

- 선형 계획법을 이용한 최적 밸러스트 탱크 배치(2/2)

어느 선박의 밸러스트 탱크(ballast tank)는 다음과 같은 특성을 갖고 있다. 이 선박의 현재 트림(trim) 및 GM은 각각 2m, 0.1m인데, 밸러스트 탱크들을 부분적으로 또는 완전히 채워서 트림을 0.0mm로 GM을 0.5m 이상으로 유지하고자 한다. 어떤 탱크에 얼마의 양의 밸러스트 수(ballast water)를 채우면 되겠는가? 물론 이때 밸러스트 수의 양을 최소로 해야 한다.

탱크	탱크의 용량	1,000m³ 탱크 용량의 트림 변화량(m)	1,000m³ 탱크 용량의 GM 변화량(m)
1	500	0.0	0.1
2	1,000	-2.0	0.1
3	1,500	0.5	0.5
4	2,000	-1.0	0.2
5	2,500	-0.2	0.4

i번째 탱크에 채운 밸러스트 수의 양을 1,000m^3 단위로 x_i로서 표현하면

Find x_1, x_2, x_3, x_4, x_5

Minimize $f(x_1, x_2, x_3, x_4, x_5) = x_1 + x_2 + x_3 + x_4 + x_5$

Subject to

$$\left.\begin{array}{l} 0 \le x_1 \le 0.5,\ 0 \le x_2 \le 1.0,\ 0 \le x_3 \le 1.5, \\ 0 \le x_4 \le 2.0,\ 0 \le x_5 \le 2.5 \end{array}\right\} \text{: 각 탱크의 용량 제한 조건}$$

$$0.1x_1 + 0.1x_2 + 0.5x_3 + 0.2x_4 + 0.4x_5 \ge 0.4 \text{ : GM 요구 조건}$$

$$2x_2 - 0.5x_3 + x_4 + 0.2x_5 = 2 \text{ : 트림 요구 조건}$$

➡ 미지수 5개, 등호 제약 조건 1개, 부등호 제약 조건 6개인 최적화 문제

등호 제약 조건이 있는 최적화 문제에서의 Lagrange 함수의 구성

Original Problem

Minimize $f(\mathbf{x}) = (x_1 - 1.5)^2 + (x_2 - 1.5)^2$

Subject to $h(\mathbf{x}) = x_1 + x_2 - 2 = 0$

Lagrange Function

Minimize $L(\mathbf{x}, v) = f(\mathbf{x}) + vh(\mathbf{x})$

$$= (x_1 - 1.5)^2 + (x_2 - 1.5)^2 + v(x_1 + x_2 - 2)$$

Necessary Condition: $\nabla L(\mathbf{x}^*, v^*) = 0$

$\nabla f(\mathbf{x}^*) + v^* \nabla h(\mathbf{x}^*) = 0$

$\therefore -\nabla f(\mathbf{x}^*) = v^* \nabla h(\mathbf{x}^*)$

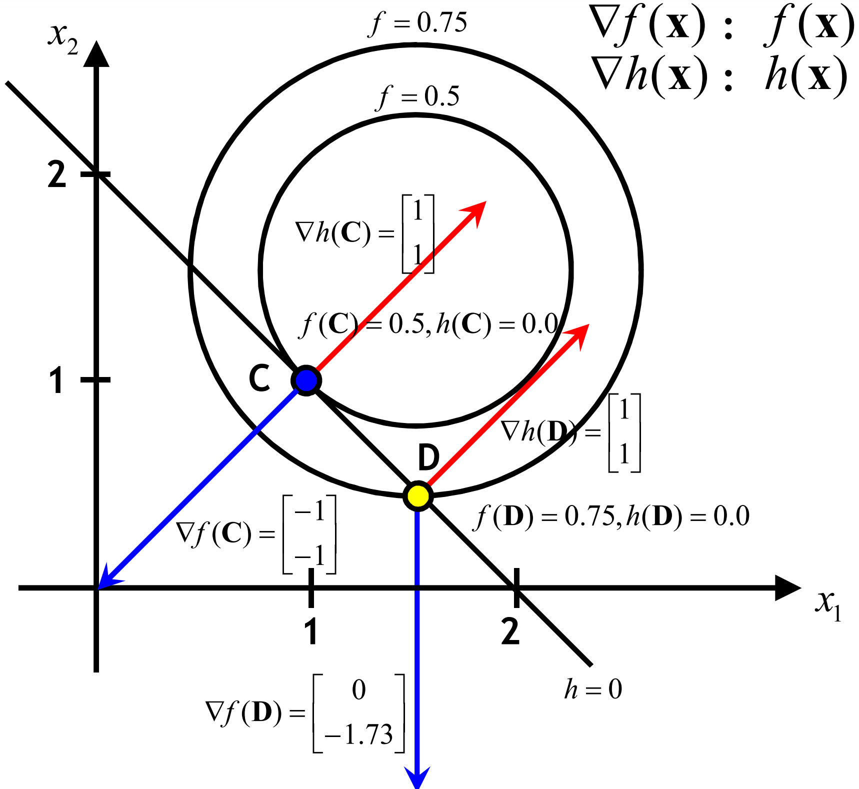
$\nabla f(\mathbf{x}) = \begin{bmatrix} 2(x_1 - 1.5) \\ 2(x_2 - 1.5) \end{bmatrix}, \quad \nabla h(\mathbf{x}) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$-2(x_1^* - 1.5) = v^*, \quad -2(x_2^* - 1.5) = v^*$

$x_1^* + x_2^* - 2 = 0$

$\Rightarrow x_1^* = x_2^* = 1, v^* = 1$ (점 C)

$\nabla f(\mathbf{x})$: $f(\mathbf{x})$ 의 증가 방향
$\nabla h(\mathbf{x})$: $h(\mathbf{x})$ 의 증가 방향



후보 최적점 C에서 $-\nabla f(\mathbf{x}^*) = v^* \nabla h(\mathbf{x}^*)$ 의 의미를 살펴 보면,

목적 함수 및 제약 함수의 Gradient 벡터는 동일 작용선 상에 있고, 서로 비례하며 이때 Lagrange multiplier v^*가 비례 상수임

$$\nabla f(\mathbf{C}) = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \quad \nabla h(\mathbf{C}) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad v^* = 1$$

그러나 점 D에서는 위 식을 만족하지 않으므로 후보 최적점이 아님

후보 최적성 필요 조건을 이용한 부등호 제약 최적화 문제의 해법

Original Problem

Minimize $f(\mathbf{x}) = (x_1 - 1.5)^2 + (x_2 - 1.5)^2$

Subject to $g(\mathbf{x}) = x_1 + x_2 - 2 \le 0$

➡ $g(\mathbf{x}) + s^2 = x_1 + x_2 - 2 + s^2 = 0$

부등호 제약 조건을 등호 제약 조건으로 변환하기 위해 도입한 완화 변수(slack variable)

Lagrange Function

Minimize $L(\mathbf{x}, u, s) = f(\mathbf{x}) + u[g(\mathbf{x}) + s^2]$

$$= (x_1 - 1.5)^2 + (x_2 - 1.5)^2$$

$$+ u(x_1 + x_2 - 2 + s^2)$$

Necessary Condition: $\nabla L(\mathbf{x}^*, u^*, s^*) = 0$

$$\frac{\partial L}{\partial x_1} = 2(x_1 - 1.5) + u = 0, \quad \frac{\partial L}{\partial x_2} = 2(x_2 - 1.5) + u = 0$$

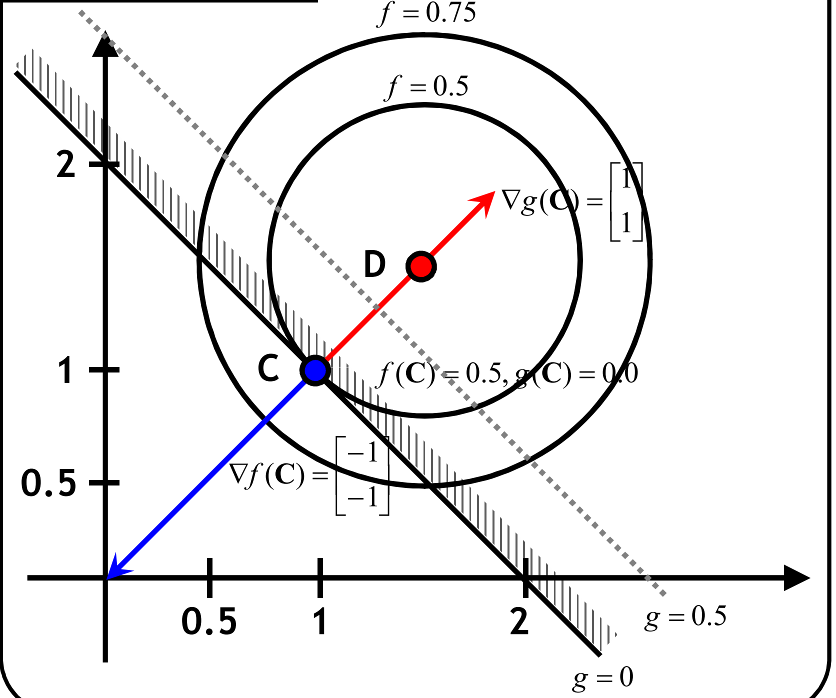
$$\frac{\partial L}{\partial u} = x_1 + x_2 - 2 + s^2 = 0, \quad \frac{\partial L}{\partial s} = 2us = 0 \quad \text{단, } u \ge 0$$

(1) s = 0일 때(부등호 제약 조건이 등호 제약 조건으로 변환)

$x_1^* = x_2^* = 1, u^* = 1$ ➡ 후보 최적점(점 C)

(2) u = 0일 때(부등호 제약 조건을 만족함, 즉 부등호 제약 조건이 없는 문제임)

$x_1^* = x_2^* = 1.5, u^* = 0, s^2 = -1$ (점 D: 제약 조건을 위배)



제약 최적화 문제
- Kuhn-Tucker 필요 조건을 이용한 국부적 후보 최적해 도출

①

$$Minimize\ \ f(\mathbf{x}) = x_1^{\,2} + x_2^{\,2} - 3x_1x_2$$

$$g(\mathbf{x}) = x_1^{\,2} + x_2^{\,2} - 6 \le 0$$

②

$$L(\mathbf{x}, u, s) = x_1^{\,2} + x_2^{\,2} - 3x_1x_2 + u(x_1^{\,2} + x_2^{\,2} - 6 + s^2)$$

③

$$\frac{\partial L}{\partial x_1} = 2x_1 - 3x_2 + 2ux_1 = 0$$

$$\frac{\partial L}{\partial x_2} = 2x_2 - 3x_1 + 2ux_2 = 0$$

$$\frac{\partial L}{\partial u} = x_1^{\,2} + x_2^{\,2} - 6 + s^2 = 0,\ s^2 \ge 0,\ u \ge 0$$

$$\frac{\partial L}{\partial s} = \boxed{2us = 0}$$ ➡ 두가지 경우가 나옴

CASE #1 : $u = 0$(제약 조건을 고려하지 않아도 되는 경우)

$$2x_1 - 3x_2 = 0$$
$$-3x_1 + 2x_2 = 0$$

➡ 점 A: $x_1^* = 0,\ x_2^* = 0,\ f(x_1^*, x_2^*) = 0$

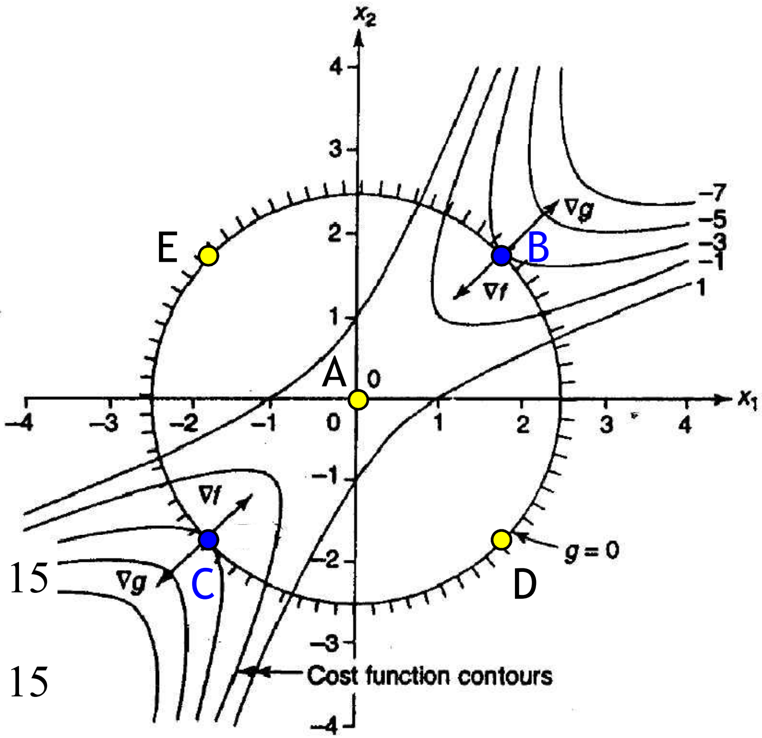
CASE #2 : $s = 0$ (제약 조건의 경계 상에 해가 있는 경우)

$x_1 = x_2 = \sqrt{3},\ u = \frac{1}{2}$ ➡ 점 B: $x_1^* = x_2^* = \sqrt{3}, f(x_1^*, x_2^*) = -3$

$x_1 = x_2 = -\sqrt{3},\ u = \frac{1}{2}$ ➡ 점 C: $x_1^* = x_2^* = -\sqrt{3}, f(x_1^*, x_2^*) = -3$

$x_1 = -x_2 = \sqrt{3},\ u = -\frac{5}{2}$ ➡ 점 D: $x_1^* = \sqrt{3}, x_2^* = -\sqrt{3}, f(x_1^*, x_2^*) = 15$

$x_1 = -x_2 = -\sqrt{3},\ u = -\frac{5}{2}$ ➡ 점 E: $x_1^* = -\sqrt{3}, x_2^* = \sqrt{3}, f(x_1^*, x_2^*) = 15$



K-T 필요 조건을 이용한 2차 계획 문제 최적해 구하기

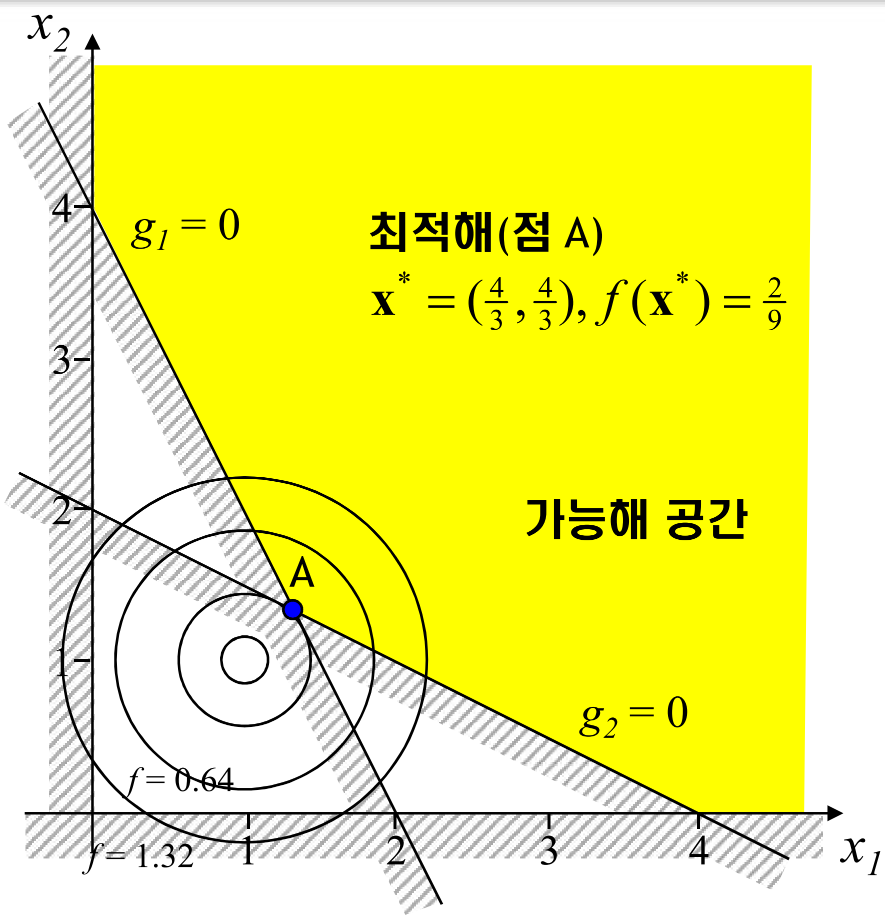
$$\textit{Minimize} \quad f(\mathbf{x}) = x_1^2 + x_2^2 - 2x_1 - 2x_2 + 2$$

$$\textit{Subject to} \quad g_1(\mathbf{x}) = -2x_1 - x_2 + 4 \le 0$$

$$g_2(\mathbf{x}) = -x_1 - 2x_2 + 4 \le 0$$

단, $x_1 \ge 0, x_2 \ge 0$

최적해는 $\mathbf{x}^* = (\frac{4}{3}, \frac{4}{3}), f(\mathbf{x}^*) = \frac{2}{9}$



Lagrange 함수

$$L(\mathbf{x}, \mathbf{u}, \mathbf{s}, \zeta) = x_1^2 + x_2^2 - 2x_1 - 2x_2 + 2$$
$$+ u_1(-2x_1 - x_2 + 4 + s_1^2)$$
$$+ u_2(-x_1 - 2x_2 + 4 + s_2^2)$$
$$- \zeta_1 x_1 - \zeta_2 x_2$$

CSD(Constrained Steepest Descent) 방법을 이용한 풀이 예

$$Minimize \quad f(\mathbf{x}) = x_1^2 + x_2^2 - 3x_1x_2$$

$$Subject\ to \quad g_1(\mathbf{x}) = \frac{1}{6}x_1^2 + \frac{1}{6}x_2^2 - 1.0 \le 0$$

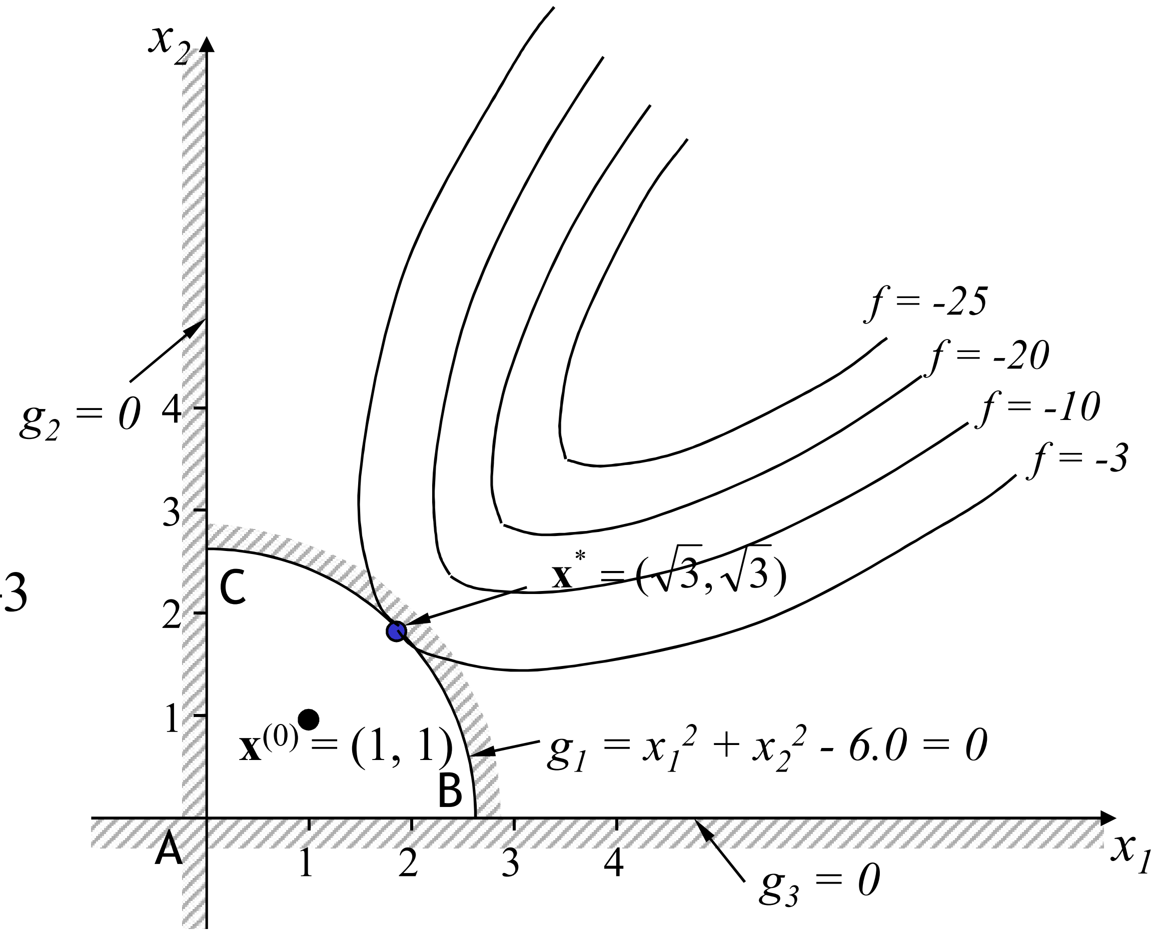
$$g_2(\mathbf{x}) = -x_1 \le 0$$

$$g_3(\mathbf{x}) = -x_1 \le 0$$

초기 시작점은 $\mathbf{x}^{(0)} = (1,1)$,

$R_0 = 10, \varepsilon_1 = \varepsilon_2 = 0.001$

이라 가정



최적해는 $\mathbf{x}^* = (\sqrt{3}, \sqrt{3}), f(\mathbf{x}^*) = -3$

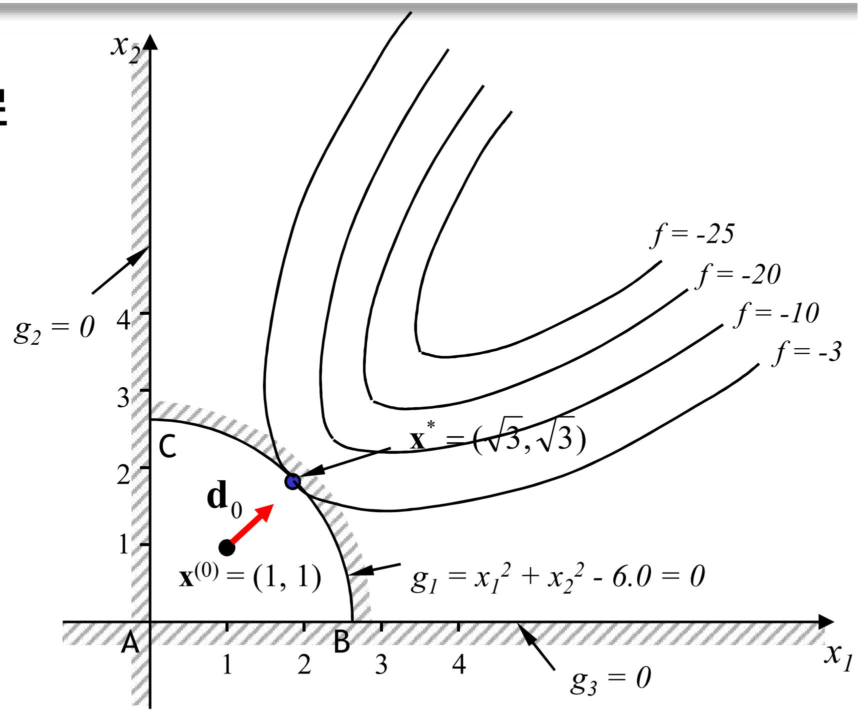
이때 Lagrange multiplier**는**

$$\mathbf{u}^* = (3,0,0)$$

CSD(Constrained Steepest Descent) 방법을 이용한 풀이 예

(vi) 단계 6: 황금 분할법을 사용하여 탐색 방향($\mathbf{d}^{(0)}$)으로 강하 함수를 최소화 하는 개선된 설계점 계산

※ 황금분할법 수행시 시작점에서의 함수값보다 δ만큼 더한 위치에서의 함수값이 더 크면 시작점의 함수값보다 작을 때 까지 δ를 줄여나감



탐색방향 : $\mathbf{d}_0 = (1,1) = (\delta_1, \delta_2)$

$$\mathbf{x}^{(0,0)} = (1,1)$$

$$\Phi\left(\mathbf{x}^{(0,0)}\right) = f\left(\mathbf{x}^{(0,0)}\right) + R \cdot V^{(0,0)} = -1 + 10 \times 0 = -1$$

$$\mathbf{x}^{(0,1)} = (1+\delta_1, 1+\delta_2) = (2,2)$$

$$\Phi\left(\mathbf{x}^{(0,1)}\right) = f\left(\mathbf{x}^{(0,1)}\right) + R \cdot V^{(0,1)} = -4 + 10 \times 0.333 = -0.667$$

$\Phi\left(\mathbf{x}^{(0,0)}\right) < \Phi\left(\mathbf{x}^{(0,1)}\right)$ 이므로, δ를 줄여 다시 계산

탐색방향 : $\mathbf{d}_0 = (0.5,0.5) = (\delta_1, \delta_2)$

$$\mathbf{x}^{(0,1)} = (1+\delta_1, 1+\delta_2) = (1.5,1.5)$$

$$\Phi\left(\mathbf{x}^{(0,1)}\right) = f\left(\mathbf{x}^{(0,1)}\right) + R \cdot V^{(0,1)} = -2.25 + 10 \times 0 = -2.25$$

$\Phi\left(\mathbf{x}^{(0,0)}\right) > \Phi\left(\mathbf{x}^{(0,1)}\right)$ 이므로, 다음 계산 수행

$$\mathbf{x}^{(0,2)} = (1+\delta_1 + 1.618\delta_1, 1+\delta_2 + 1.618\delta_2) = (2.309, 2.309)$$

$$\Phi\left(\mathbf{x}^{(0,2)}\right) = f\left(\mathbf{x}^{(0,2)}\right) + R \cdot V^{(0,2)} = -5.331 + 10 \times 0.777 = 2.440$$

$\Phi\left(\mathbf{x}^{(0,0)}\right) > \Phi\left(\mathbf{x}^{(0,1)}\right), \Phi\left(\mathbf{x}^{(0,1)}\right) < \Phi\left(\mathbf{x}^{(0,2)}\right)$이므로,

이 구간에 최소값 존재함

제약 최적화 문제 #1 (1/2)

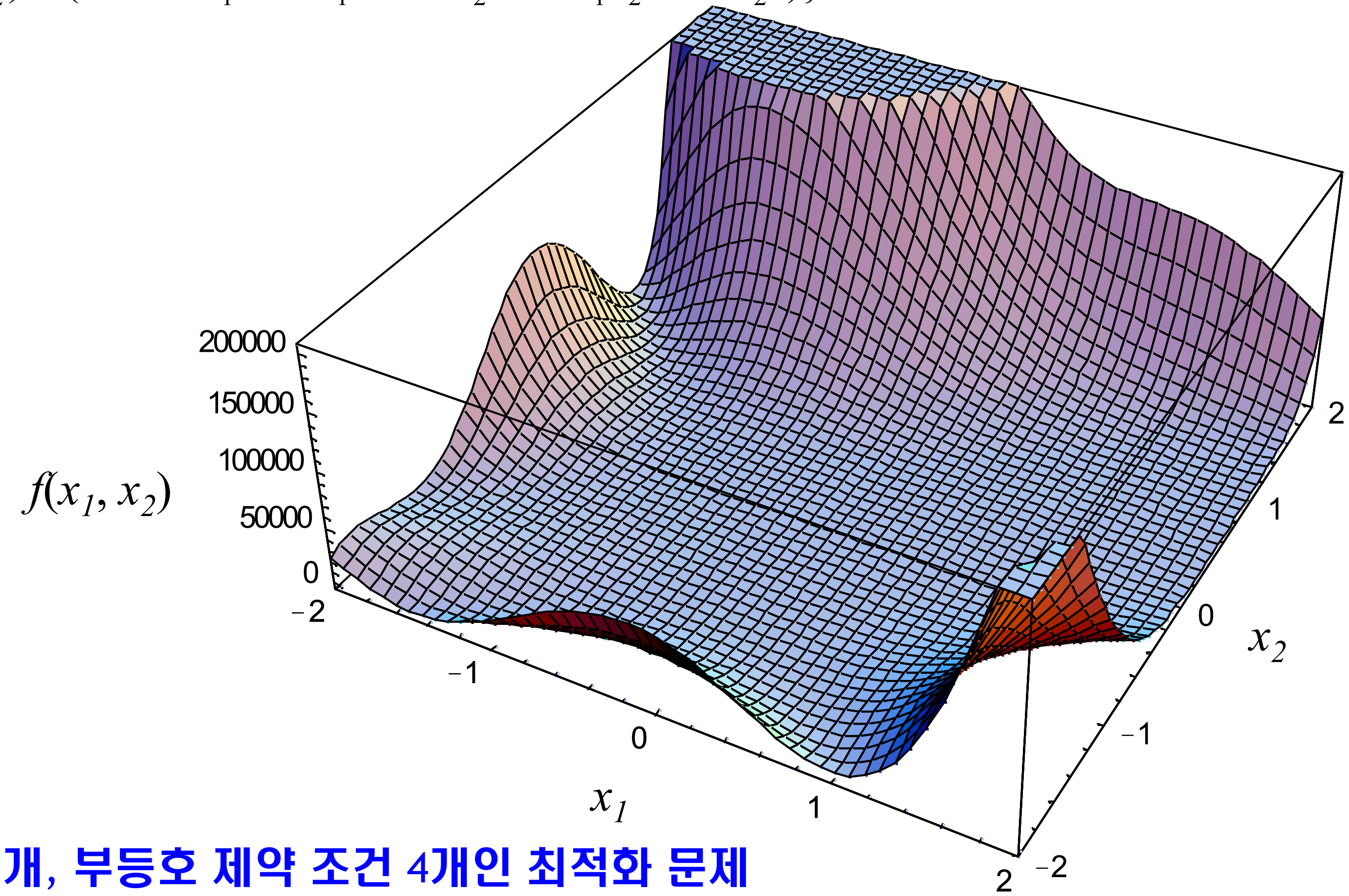
Goldstein-Price Function

Minimize

$$f(x_1,x_2)=\{1+(x_1+x_2+1)^2\cdot(19-14x_1+3x_1{}^2-14x_2+6x_1x_2+3x_2{}^2)\}$$
$$\cdot\{30+(2x_1-3x_2)^2\cdot(18-32x_1+12x_1{}^2+48x_2-36x_1x_2+27x_2{}^2)\}$$

Subject to

$$g_1(x_1,x_2)=-2-x_1\le 0$$
$$g_2(x_1,x_2)=-2-x_2\le 0$$
$$g_3(x_1,x_2)=x_1-2\le 0$$
$$g_4(x_1,x_2)=x_2-2\le 0$$

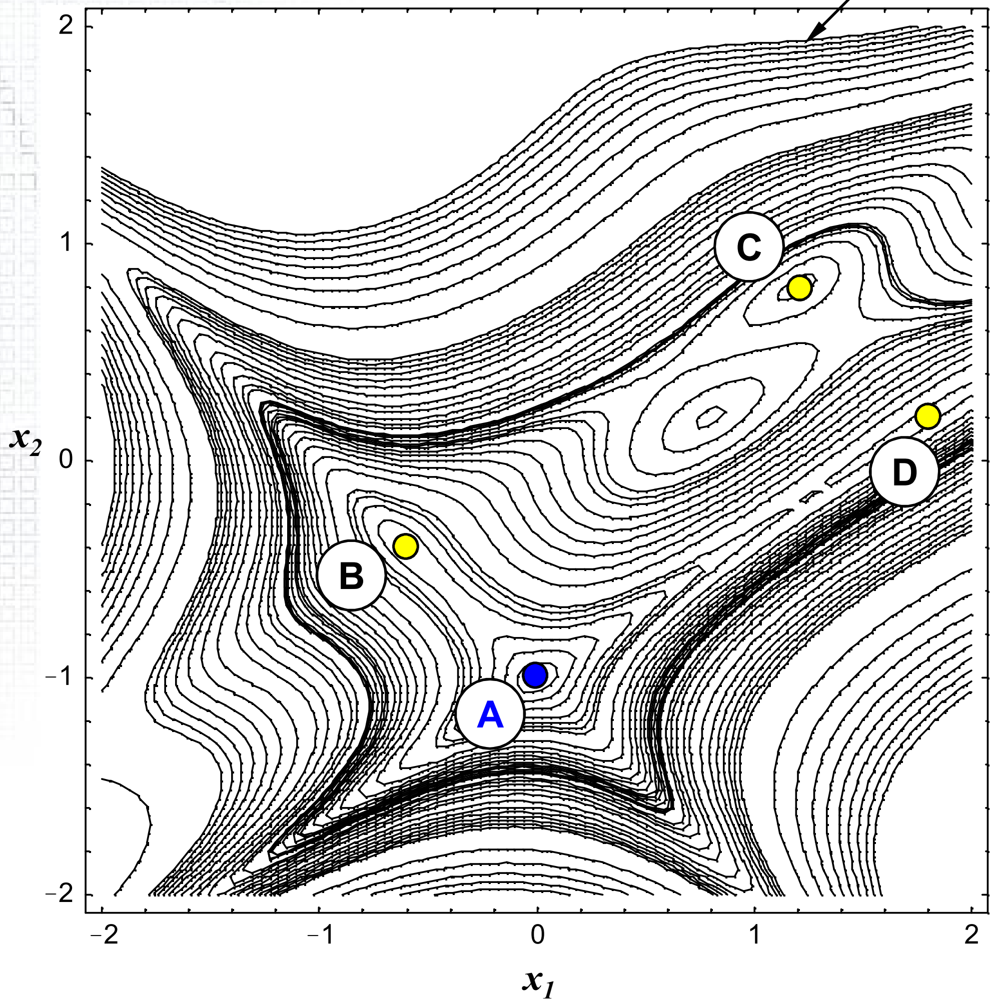


➡ **미지수 2개, 부등호 제약 조건 4개인 최적화 문제**

제약 최적화 문제 #1 (2/2)

Goldstein-Price Function

목적 함수의 contour line(f = const.)



A : Global Minimum

$x_1^* = 0.0,\ x_2^* = -1.0,\ f(x_1^*, x_2^*) = 3.0$

B : Local Minimum

$x_1^* = -0.6,\ x_2^* = -0.4,\ f(x_1^*, x_2^*) = 30.0$

C : Local Minimum

$x_1^* = 1.2,\ x_2^* = 0.8,\ f(x_1^*, x_2^*) = 840.0$

D : Local Minimum

$x_1^* = 1.8,\ x_2^* = 0.2,\ f(x_1^*, x_2^*) = 84.0$

제약 최적화 문제 #2 (1/2) - 응용 문제

- 어느 선박의 상대적 기관 마력 가격 f를 아래와 같이 B/T와 1/C_B의 함수로 나타낼 수 있다고 하자.

  상대적 기관 마력 가격:

$$f(B/T, C_B) = (B/T)^2 + 2(1/C_B)^2 - 4(B/T) - 2(B/T)\cdot(1/C_B) + 10$$

  그리고 B/T ≤ 3, $C_B \geq 0.6$이라는 제약 조건이 있을 때, Kuhn-Tucker 필요 조건을 이용하여 상대적 기관 마력 가격을 최소로 하는 B/T, C_B의 값(최적해)을 Kuhn-Tucker 필요 조건을 이용하여 구하시오.

$$\begin{aligned}
&\textit{Find} && x_1(=B/T), x_2(=1/C_B) \\
&\textit{Minimize} && f(x_1, x_2) = x_1^2 + 2x_2^2 - 4x_1 - 2x_1x_2 + 10 \\
&\textit{Subject to} && x_1 \leq 3 \\
& && x_2 \geq 0.6
\end{aligned}$$

➡ 미지수 2개, 부등호 제약 조건 2개인 최적화 문제

제약 최적화 문제 #2 (2/2) - 응용 문제

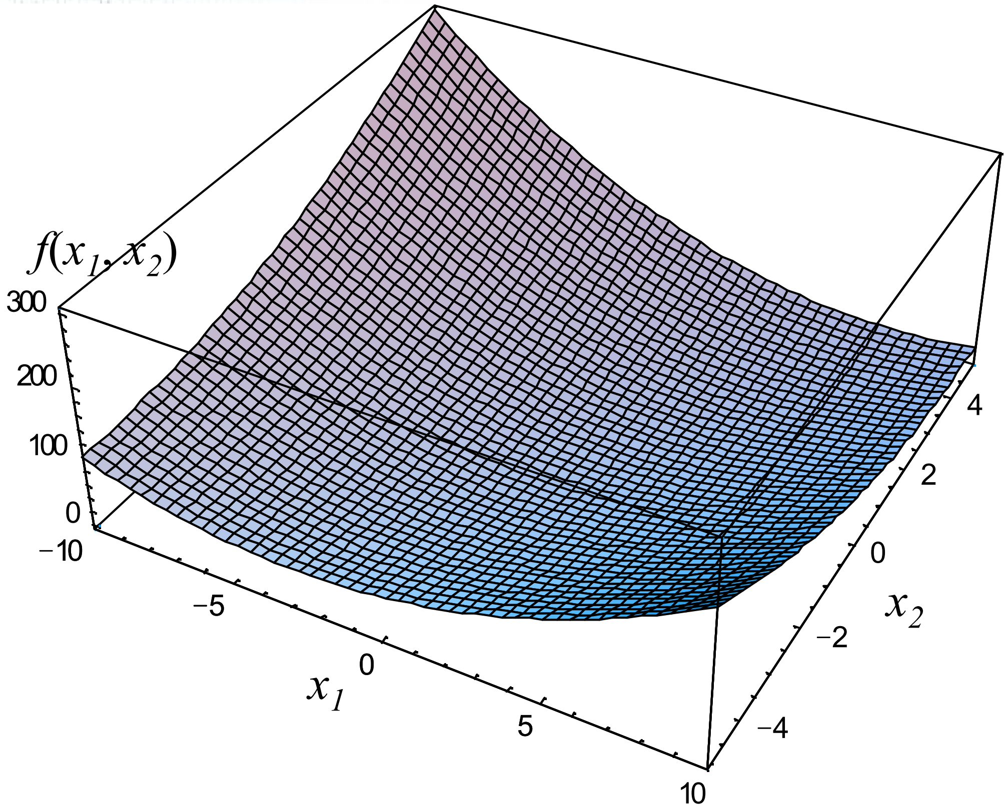
$$\textbf{Find} \quad x_1(=B/T), x_2(=1/C_B)$$

$$\textbf{Minimize} \quad f(x_1,x_2) = x_1^2 + 2x_2^2 - 4x_1 - 2x_1x_2 + 10$$

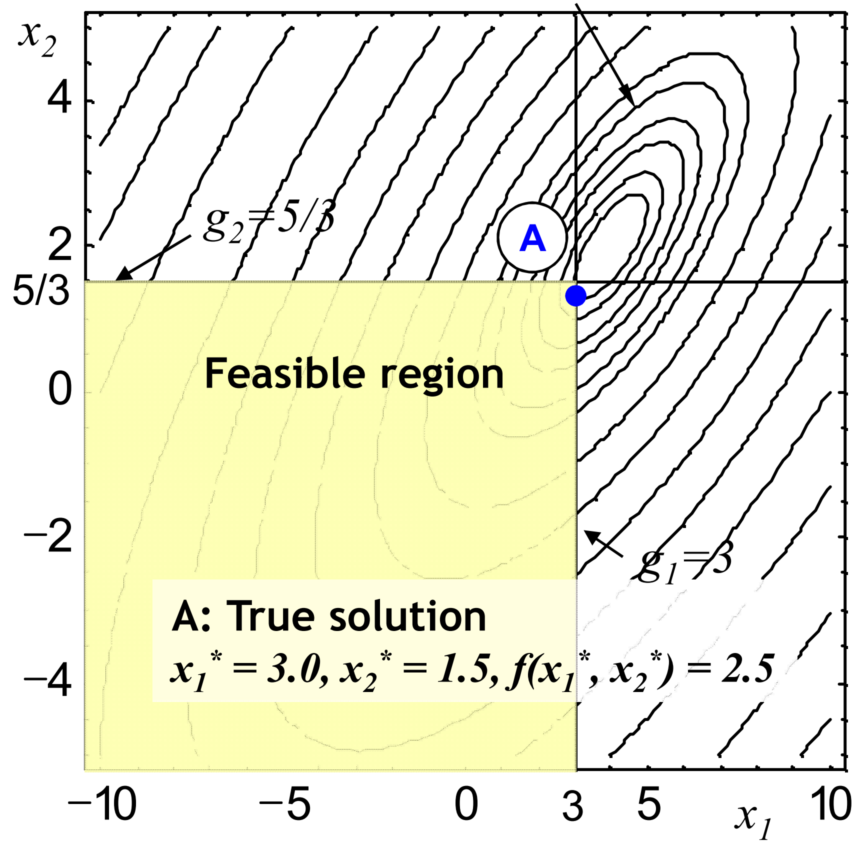
$$\textbf{Subject to} \quad g_1(x_1,x_2) = x_1 - 3 \le 0$$

➡ 미지수 2개, 부등호 제약 조건 2개인 최적화 문제

$$g_2(x_1,x_2) = x_2 - 5/3 \le 0$$



목적 함수의 contour line(f = const.)



프로펠러의 최적 주요 치수 결정 문제(간략화 된 문제)

Given $P, n, A_E / A_O, V$

Find $J, P_i / D_P$

Maximize $\eta_O = \dfrac{J}{2\pi} \cdot \dfrac{K_T}{K_Q}$ → K_T와 K_Q가 모두 J와 P_i/D_p의 함수이므로 목적 함수 역시 J와 P_i/D_p의 함수임

Subject to $\dfrac{P}{2\pi n} = \rho \cdot n^2 \cdot D_P{}^5 \cdot K_Q$

: 주기관이 전달한 토오크를 프로펠러가 흡수하는 조건

Where, $J = \dfrac{V(1-w)}{n \cdot D_P}$

$$K_T = f(J, P_i / D_P)$$

$$K_Q = f(J, P_i / D_P)$$

➡ 미지수 2개, 등호 제약 조건 1개인 최적화 문제

P: 프로펠러 전달 마력
n: 프로펠러 회전수
D_P: 프로펠러 직경
P_i: 프로펠러 피치
A_E/A_O: 프로펠러 날개 면적비
V: 선속
η_O: 프로펠러 효율

선박의 최적 주요 치수 결정 문제

구하는 값(설계 변수) L, B, D, C_B
길이 폭 깊이 방형 계수

주어진 값(선주 요구 조건) $DWT, CC_{req}, T_{\max}(=T), V$
재화 중량 요구 화물창 용적 최대 흘수 선속

부력(buoyancy)-중량(displacement) 평형 조건(등호 제약 조건)

$$L \cdot B \cdot T \cdot C_B \cdot \rho_{sw} \cdot C_\alpha = DWT_{given} + LWT(L, B, D, C_B)$$

$$= DWT_{given} + C_s \cdot L^{1.6}(B+D) + C_o \cdot L \cdot B + C_{ma} \cdot NMCR$$

$$= DWT_{given} + C_s \cdot L^{1.6}(B+D) + C_o \cdot L \cdot B$$

$$+ C_{power} \cdot (L \cdot B \cdot T \cdot C_B)^{2/3} \cdot V^3$$

요구되는 화물창 용적(cargo capacity) 조건(부등호 제약 조건)

$$CC_{req} \le C_{CH} \cdot L \cdot B \cdot D$$

최소 요구 건현 조건(부등호 제약 조건)

$$D \ge T + C_{FB} \cdot D$$

목적 함수(주요 치수 선정 기준)

$$Building\ Cost = C_{PS} \cdot C_s \cdot L^{1.6}(B+D) + C_{PO} \cdot C_o \cdot L \cdot B + C_{PM} \cdot C_{ma} \cdot NMCR$$

➡ **미지수 4개, 등호 제약 조건 1개, 부등호 제약 조건 2개인 최적화 문제**

선박의 최적 주요 치수 결정 문제의 수학적 정식화

Find L, B, D, C_B

Minimize $Building\ Cost = C_{PS} \cdot C_s \cdot L^{1.6}(B+D) + C_{PO} \cdot C_o \cdot L \cdot B + C_{PM} \cdot C_{ma} \cdot NMCR$

Subject to * 부력(buoyancy)-중량(displacement) 평형 조건

$$L \cdot B \cdot T \cdot C_B \cdot \rho_{sw} \cdot C_\alpha = DWT_{given} + LWT(L, B, D, C_B)$$
$$= DWT_{given} + C_s \cdot L^{1.6}(B+D) + C_o \cdot L \cdot B + C_{ma} \cdot NMCR$$
$$= DWT_{given} + C_s \cdot L^{1.6}(B+D) + C_o \cdot L \cdot B$$
$$+ C_{power} \cdot (L \cdot B \cdot T \cdot C_B)^{2/3} \cdot V^3$$

* 요구되는 화물창 용적 조건

$$CC_{req} \leq C_{CH} \cdot L \cdot B \cdot D$$

* 최소 요구 건현 조건

$$D \geq T + C_{FB} \cdot D$$

➡ **미지수 4개, 등호 제약 조건 1개, 부등호 제약 조건 2개인 최적화 문제**

Ch0. 최적 설계 소개

Advanced
Ship
Design
Automation
Laboratory

최적 설계에 앞서
- 부정 방정식과 그 해법

변수: x_1, x_2, x_3

식: $x_1 + x_2 + x_3 = 3$

✓ **변수의 개수: 3개**

✓ **식의 개수: 1개**

변수의 개수가 식의 개수보다 많으므로

위의 문제는 부정 방정식이다.

위의 부정 방정식 해법

2개의 변수를 가정한다.

변수의 개수(3) - 식의 개수(1)

예) $x_1 = 1, x_2 = 0$**로 가정**

$\rightarrow x_3 = 2$

1knot = 0.5144 m/s
= 1.852 km/h

속도: 15knot = 27.78 km/h

t초 후의 선박의 위치(d)는?

식: $d = v \cdot t + d_0$

변수: d, t (v: 속도, d_0: 초기 위치)

✓ **변수의 개수: 2개**

✓ **식의 개수: 1개**

시간 t를 가정하여 임의의 시간에 대한 선박의 위치 d를 알 수 있다.

최적 설계에 앞서
- 부정 방정식과 그 해법

연립 방정식

변수: x_1, x_2, x_3

식: $f_1(x_1, x_2, x_3)=0$

$f_2(x_1, x_2, x_3)=0$

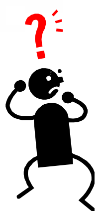
$f_3(x_1, x_2, x_3)=0$

식 f_1, f_2, f_3 이 모두 독립이라면

✓ 변수의 개수: 3개

✓ 식의 개수: 3개

식의 개수와 변수의 개수가 같으므로 풀 수 있는 문제



만일 $2\times f_3 = f_2$ 라면?

f_2는 f_3 로 표현이 가능하므로 독립이 아님

서로 독립인 식의 개수가 변수의 개수보다 적으므로 **부정 방정식**이 된다.

부정 방정식

변수: x_1, x_2, x_3

식: $f_1(x_1, x_2, x_3)=0$

$f_2(x_1, x_2, x_3)=0$

$f_3(x_1, x_2, x_3)=0$

식 f_1, f_2 이 서로 독립이라면

✓ 변수의 개수: 3개

✓ 식의 개수: 2개

식의 개수가 변수의 개수보다 적으므로 식을 추가하여 해를 구한다.

추가한 식	구한 해
$f_4^1 = 0$	(x_1^1, x_2^1, x_3^1)
$f_4^2 = 0$	(x_1^2, x_2^2, x_3^2)
⋮	

추가한 식에 따라 무수히 많은 해가 구해짐 → 부정 방정식

무수히 많은 해 중 어떤 것이 좋은지 판단할 기준이 필요하며, 판단 기준으로서 목적함수를 추가하면 **최적화 문제**가 된다.

최적 설계에 앞서
- 제약 최적화 문제와 그 해법 - Lagrange Multiplier를 이용

최적화 문제

Minimize $f(x_1, x_2, x_3)$ ···· ①

Subject to $h_1(x_1, x_2, x_3) = 0$ ···· ②

$h_2(x_1, x_2, x_3) = 0$ ···· ③

함수 f 값이 최소가 되기 위한 필요조건: 식 ①'

$$df = \frac{\partial f}{\partial x_1}dx_1 + \frac{\partial f}{\partial x_2}dx_2 + \frac{\partial f}{\partial x_3}dx_3 = 0 \cdots ①'$$

$$h_1(x_1, x_2, x_3) = 0 \cdots ②$$

$$h_2(x_1, x_2, x_3) = 0 \cdots ③$$

식 h_1, h_2로 인하여 dx_1, dx_2, dx_3는 서로 독립이 아니다.

dx_1, dx_2, dx_3 의 관계를 알기 위해 식②, ③를 변형

식 ②, ③ 으로부터 식 ②', ③'를 유도

$$df = \frac{\partial f}{\partial x_1}dx_1 + \frac{\partial f}{\partial x_2}dx_2 + \frac{\partial f}{\partial x_3}dx_3 = 0 \cdots ①'$$

$$dh_1 = \frac{\partial h_1}{\partial x_1}dx_1 + \frac{\partial h_1}{\partial x_2}dx_2 + \frac{\partial h_1}{\partial x_3}dx_3 = 0 \cdots ②'$$

$$dh_2 = \frac{\partial h_2}{\partial x_1}dx_1 + \frac{\partial h_2}{\partial x_2}dx_2 + \frac{\partial h_2}{\partial x_3}dx_3 = 0 \cdots ③'$$

식 ①'의 dx_1, dx_2를 소거하기 위하여 식 ②', ③'을 이용하여 다음 식을 유도

$$df + \lambda_1 dh_1 + \lambda_2 dh_2 = 0 \cdots$$ ①'에 새로운 변수 λ_1, λ_2를 도입하고 ②', ③'을 대입 → 식④,⑤,⑥을 유도

$$\underbrace{\left(\frac{\partial F}{\partial x_1} + \lambda_1\frac{\partial f_1}{\partial x_1} + \lambda_2\frac{\partial f_2}{\partial x_1}\right)dx_1}_{=0\ (dx_1\text{를 소거하기 위함, 식④})} + \underbrace{\left(\frac{\partial F}{\partial x_2} + \lambda_1\frac{\partial f_1}{\partial x_2} + \lambda_2\frac{\partial f_2}{\partial x_2}\right)dx_2}_{=0\ (dx_2\text{를 소거하기 위함, 식⑤})} + \underbrace{\left(\frac{\partial F}{\partial x_3} + \lambda_1\frac{\partial f_1}{\partial x_3} + \lambda_2\frac{\partial f_2}{\partial x_3}\right)dx_3}_{=0\ (dx_3\text{는 독립변수 이므로, 식⑥})} = 0$$

$$dh_1 = \frac{\partial h_1}{\partial x_1}dx_1 + \frac{\partial h_1}{\partial x_2}dx_2 + \frac{\partial h_1}{\partial x_3}dx_3 = 0 \cdots ②'$$

$$dh_2 = \frac{\partial h_2}{\partial x_1}dx_1 + \frac{\partial h_2}{\partial x_2}dx_2 + \frac{\partial h_2}{\partial x_3}dx_3 = 0 \cdots ③'$$

최적 설계에 앞서
- 제약 최적화 문제와 그 해법 - Lagrange Multiplier를 이용

최적화 문제

Minimize $f(x_1, x_2, x_3)$ ···· ①

Subject to $h_1(x_1, x_2, x_3) = 0$ ···· ②

$h_2(x_1, x_2, x_3) = 0$ ···· ③

$$\underbrace{\left(\frac{\partial F}{\partial x_1}+\lambda_1\frac{\partial f_1}{\partial x_1}+\lambda_2\frac{\partial f_2}{\partial x_1}\right)}dx_1+\underbrace{\left(\frac{\partial F}{\partial x_2}+\lambda_1\frac{\partial f_1}{\partial x_2}+\lambda_2\frac{\partial f_2}{\partial x_2}\right)}dx_2+\underbrace{\left(\frac{\partial F}{\partial x_3}+\lambda_1\frac{\partial f_1}{\partial x_3}+\lambda_2\frac{\partial f_2}{\partial x_3}\right)}dx_3=0$$

$=0$ (dx_1를 소거하기 위함, 식④) $=0$ (dx_2를 소거하기 위함, 식⑤) $=0$ (dx_3는 독립변수 이므로, 식⑥)

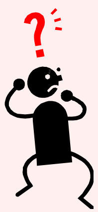
$$dh_1=\frac{\partial h_1}{\partial x_1}dx_1+\frac{\partial h_1}{\partial x_2}dx_2+\frac{\partial h_1}{\partial x_3}dx_3=0 \quad \cdots ②'$$

$$dh_2=\frac{\partial h_2}{\partial x_1}dx_1+\frac{\partial h_2}{\partial x_2}dx_2+\frac{\partial h_2}{\partial x_3}dx_3=0 \quad \cdots ③'$$

함수 f, h_1, h_2는 이미 주어진 함수이기 때문에 식 ④, ⑤, ⑥은 x_1, x_2, x_3의 (비)선형 대수 방정식 형태이다.

식 ④, ⑤, ⑥은 식 ①로 부터 유도 되었다.(변수 λ_1, λ_2 가 추가되어 식의 개수가 3개가 됨)

변수가 총 5개($x_1, x_2, x_3, \lambda_1, \lambda_2$)이므로 식 ②, ③, ④, ⑤, ⑥을 사용하여 문제를 풀 수 있다.



1. 식 ①', ②', ③'은 미분 방정식이 아닌가요?

→ 만일 다음과 같은 문제라면 미분 방정식이 맞습니다.

- Given: $\frac{\partial F}{\partial x_1}dx_1+\frac{\partial F}{\partial x_2}dx_2+\frac{\partial F}{\partial x_3}dx_3=0,\ \frac{\partial f_1}{\partial x_1}dx_1+\frac{\partial f_1}{\partial x_2}dx_2+\frac{\partial f_1}{\partial x_3}dx_3=0,\ \frac{\partial f_2}{\partial x_1}dx_1+\frac{\partial f_2}{\partial x_2}dx_2+\frac{\partial f_2}{\partial x_3}dx_3=0,$ - Find: 함수 F, f_1, f_2

그런데 함수 F, f_1, f_2(식①, ②, ③)는 이미 주어진 것이고, 우리가 구하려고 하는 것은 x_1, x_2, x_3이기 때문에 미분 방정식이 아닙니다.

함수 값이 최소가 되기 위한 필요 조건

1변수 최적화 문제의 경우

Given **목적 함수**: $f(x)$

Find **목적 함수를 최소화하는 점** x^*

목적 함수를 테일러 전개 (2차항까지 전개)

$$f(x^* + \Delta x) = f(x^*) + \frac{\partial f}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(\Delta x)^2$$

$$f(x^* + \Delta x) - f(x^*) = \frac{\partial f}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(\Delta x)^2$$

x^*에서 함수 값이 최소라고 하면

LHS $\quad f(x^* + \Delta x) - f(x^*) \ge 0$

RHS $\quad \Delta x$ **가 독립이므로**(부호를 알 수 없으므로)

$$\boxed{\boxed{\frac{\partial f}{\partial x} = 0,} \quad \frac{\partial^2 f}{\partial x^2} > 0}$$

필요 조건 **필요 충분 조건**

2변수 최적화 문제의 경우

Given **목적 함수**: $f(x_1, x_2)$

Find **목적 함수를 최소화하는 점** $\mathbf{x}^*$

목적 함수를 테일러 전개 (2차항까지 전개)

$$f(x_1^* + \Delta x_1, x_2^* + \Delta x_2) = f(x_1^*, x_2^*) + \frac{\partial f}{\partial x_1}\Delta x_1 + \frac{\partial f}{\partial x_2}\Delta x_2 + \frac{1}{2}\frac{\partial^2 f}{\partial x_1^2}(\Delta x_1)^2 + \frac{\partial^2 f}{\partial x_1 \partial x_2}\Delta x_1 \Delta x_2 + \frac{1}{2}\frac{\partial^2 f}{\partial x_2^2}(\Delta x_2)^2$$

$$f(x_1^* + \Delta x_1, x_2^* + \Delta x_2) = f(x_1^*, x_2^*) + \underbrace{\begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix}}_{\nabla f^T} \underbrace{\begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix}}_{\Delta \mathbf{x}} + \frac{1}{2}\underbrace{\begin{bmatrix} \Delta x_1 & \Delta x_2 \end{bmatrix}}_{\Delta \mathbf{x}^T} \underbrace{\begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix}}_{\Delta \mathbf{x}}$$

x^*에서 함수 값이 최소라고 하면

$\boxed{\boxed{\nabla f = 0,} \quad \mathbf{H}}$ (Hessian Matrix)**는 양정행렬**

필요 조건 **필요 충분 조건**

최적 설계에 앞서
- Design

Esthetic* Design

구하는 값(설계 변수)

- 팔 길이, 소재, 색상 등..

제약 조건

- 제약 조건이 있지만 수치화 하기 어려움
- Designer의 설계 감각으로 제약 조건을 만족시킴

목적 함수(주요 치수 선정 기준)

- 선호도, 가격 등..
- 설계 변수로서 수치화 하기 어려움

* 미적가치관, 미적인 감각

최적 설계에 앞서
- Design

최적화 문제

➡ 제약 조건을 만족하면서
목적함수를 최소화(최대화)하는
설계 변수를 결정 하는 것

구하는 값(설계 변수)

x_1, x_2, x_3, x_4

등호 제약 조건

$h(x_1, x_2, x_3, x_4)=0$

목적 함수

$f(x_1, x_2, x_3, x_4)$

- C_i, ρ_{sw}, DWT_{given}, $NMCR$: **주어진 계수**

Engineering Design

구하는 값(설계 변수) $L(=x_1), B(=x_2), D(=x_3), C_B(=x_4)$

길이 폭 깊이 방형 계수

부력(buoyancy)-중량(displacement) 평형 조건(등호 제약 조건)

$$L \cdot B \cdot T \cdot C_B \cdot \rho_{sw} \cdot C_\alpha = DWT_{given} + LWT(L, B, D, C_B)$$

$$x_1 \cdot x_2 \cdot x_4 \cdot C_1 = C_2 + h'(x_1, x_2, x_3, x_4)$$

$$x_1 \cdot x_2 \cdot x_4 \cdot C_1 - C_2 + h'(x_1, x_2, x_3, x_4) = h(x_1, x_2, x_3, x_4) = 0$$

목적 함수(주요 치수 선정 기준)

$$Building\ Cost = C_{PS} \cdot C_s \cdot L^{1.6}(B+D) + C_{PO} \cdot C_o \cdot L \cdot B + C_{PM} \cdot C_{ma} \cdot NMCR$$

$$f(x_1, x_2, x_3, x_4) = C_3 \cdot x_1^{1.6}(x_2 + x_3) + C_4 \cdot x_1 \cdot x_2 + C_5$$

제약조건의 특징

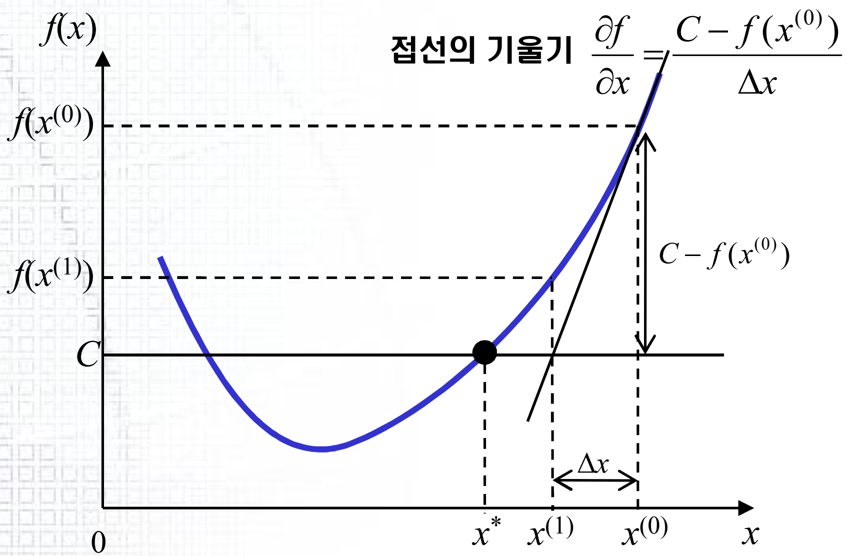
✓ 물리 법칙은 보통 등식으로 표현 됨 (선박 설계의 예: 부력-중량 평형 조건)

✓ 정치, 경제, 사회, 문화적으로 정의된 규약이나 조건은 부등호 제약 조건식으로 표현 된다. (선박 설계의 예: 최소 요구 건현 조건, 요구되는 화물창 용적 조건)

$f(x)$=C를 만족하는 점과 $f(x)$를 최소화하는 점을 찾는 방법 비교

Given $f(x)$

Find $f(x)$ =C 를 만족하는 점 x^*



$x^{(0)}$에서 테일러 전개(1차 항까지 고려)

$$f(x^{(0)}+\Delta x)=f(x^{(0)})+\frac{\partial f}{\partial x}\Delta x=C\quad\left(\Rightarrow\frac{\partial f}{\partial x}=\frac{C-f(x^{(0)})}{\Delta x}\right)$$

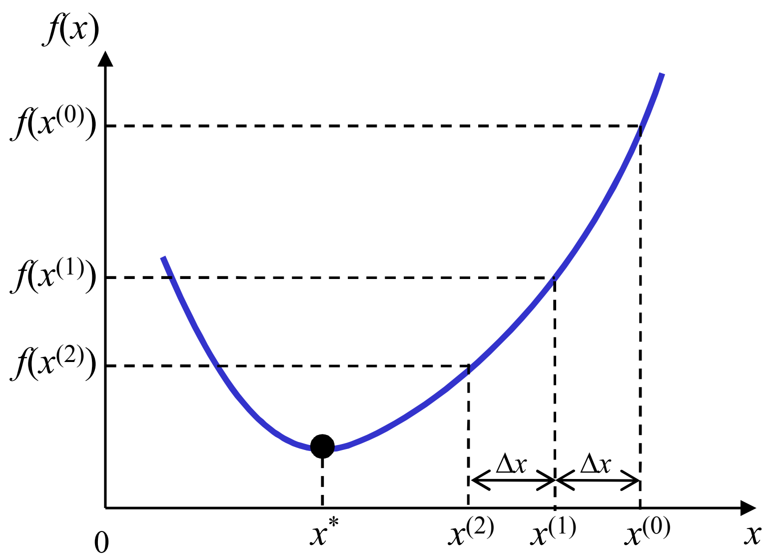
$f(x^{(0)}+\Delta x)=C$ 을 만족하는 Δx를 구한다.

Δx 만큼 이동한 점 $x^{(1)}\left(=x^{(0)}+\Delta x\right)$에서
위의 과정을 반복하여 x^*를 찾는다.

상세 과정 설명 ▷

Given 목적 함수: $f(x)$

Find 목적 함수를 최소화하는 점 x^*



$x^{(0)}$에서 테일러 전개(1차 항까지 고려)

$$f(x^{(0)}+\Delta x)=f(x^{(0)})+\frac{\partial f}{\partial x}\Delta x$$

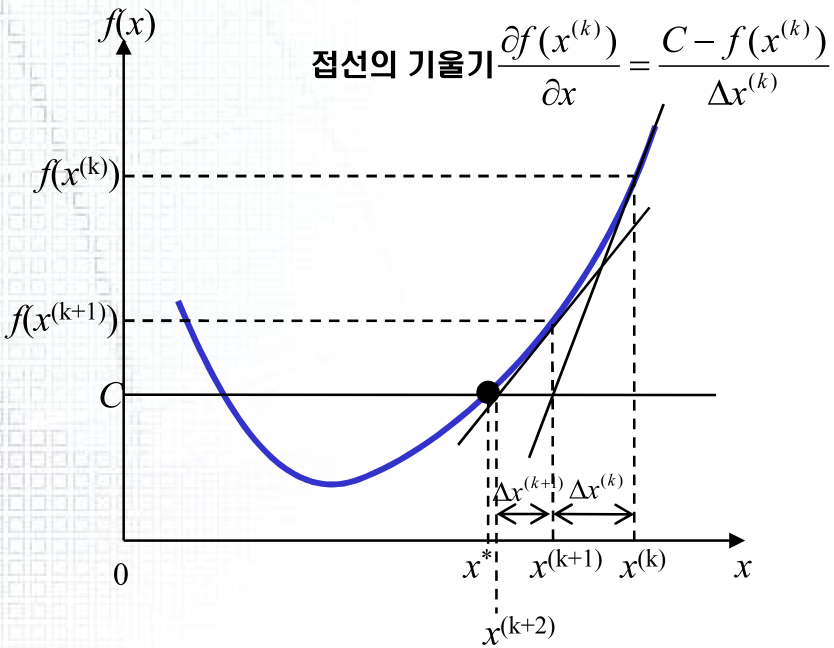
$f(x^{(0)}+\Delta x)$가 최소 값이라고 가정 했으므로
$f(x^{(0)}+\Delta x)<f(x^{(0)})$를 만족 하는 Δx 를 구한다.
Δx 만큼 이동한 점 $x^{(1)}\left(=x^{(0)}+\Delta x\right)$에서
위의 과정을 반복하여 x^*를 찾는다.

상세 과정 설명 ▷

$f(x)$=C를 만족하는 점을 찾는 방법
- Newton-Rapson 방법

Given $f(x)$

Find $f(x)$ =C 를 만족하는 점 x^*



현재의 설계점 $x^{(k)}$에서 테일러 전개 한다.

$$f(x^{(k)} + \Delta x^{(k)}) = f(x^{(k)}) + \frac{\partial f(x^{(k)})}{\partial x} \Delta x^{(k)} = C$$

$x^{(k)} + \Delta x^{(k)}$에서 함수값이 C가 되었다고 가정하고 $\Delta x^{(k)}$ 를 구한다.

↓

테일러 전개식을 이용하여 $\Delta x^{(k)}$ 를 계산한다.

$$\Delta x^{(k)} = \left(C - f(x^{(k)})\right) / \left(\frac{\partial f(x^{(k)})}{\partial x}\right)$$

Given

↓

다음 설계점을 정의한다.

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)}$$

↓

$|x^{(k+1)} - x^{(k)}| < \varepsilon$ 를 만족 하는가?

YES → $x^* = x^{(k+1)}$로 두고 탐색 종료

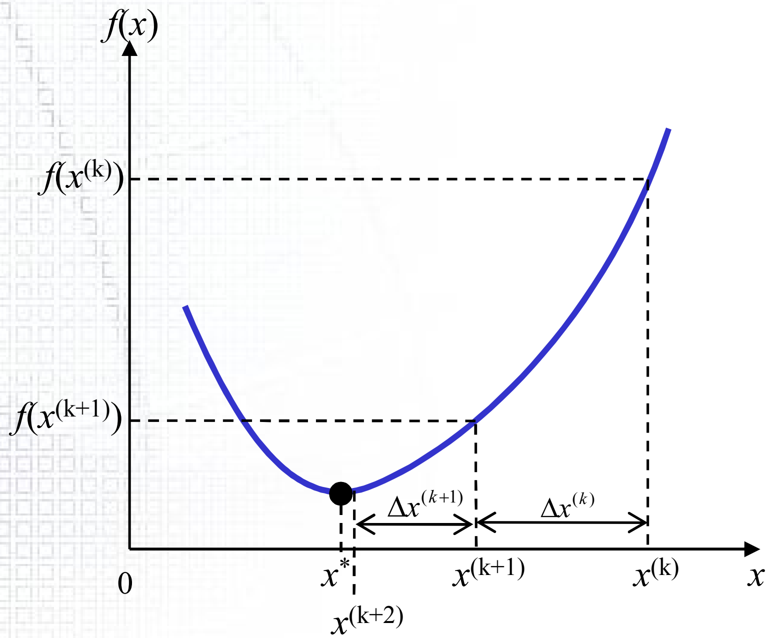
NO ↓

$$k = k + 1$$

$f(x)$를 최소화하는 점을 찾는 방법
- Newton's Method

Given $f(x)$

Find $f(x)$를 최소화하는 점 x^*



현재의 설계점 $x^{(k)}$에서 테일러 전개 한다.

$$f(x^{(k)}+\Delta x^{(k)}) = f(x^{(k)}) + \frac{\partial f(x^{(k)})}{\partial x}\Delta x^{(k)} + \frac{1}{2}\frac{\partial^2 f(x^{(k)})}{\partial x^2}\left(\Delta x^{(k)}\right)^2$$

$$f(\Delta x^{(k)}) - f(x^{(k)}) = \frac{\partial f(x^{(k)})}{\partial x}\Delta x^{(k)} + \frac{1}{2}\frac{\partial^2 f(x^{(k)})}{\partial x^2}\left(\Delta x^{(k)}\right)^2$$

여기서 $x^{(k)}$는 상수이므로, $\Delta x^{(k)}$를 변수로 취급하여

$$\bar{f}(\Delta x^{(k)}) = \frac{\partial f(x^{(k)})}{\partial x}\Delta x^{(k)} + \frac{1}{2}\frac{\partial^2 f(x^{(k)})}{\partial x^2}\left(\Delta x^{(k)}\right)^2$$

↓

테일러 전개 식을 $\Delta x^{(k)}$로 미분 한다.

$$\frac{d\bar{f}(\Delta x^{(k)})}{d\Delta x^{(k)}} = \frac{\partial f(x^{(k)})}{\partial x} + \frac{\partial^2 f(x^{(k)})}{\partial x^2}\Delta x^{(k)} = 0 \rightarrow$$

함수 f가 최소 값을 가질 필요 조건

↓

$\Delta x^{(k)}$ 를 계산한다.

$$\Delta x^{(k)} = \left(-\frac{\partial f(x^{(k)})}{\partial x}\right) \Big/ \left(\frac{\partial^2 f(x^{(k)})}{\partial x^2}\right)$$

↓

다음 설계점을 정의한다.

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)}$$

↓

$|x^{(k+1)} - x^{(k)}| < \varepsilon$ 를 만족 하는가?

YES → **$x^* = x^{(k+1)}$로 두고 탐색 종료**

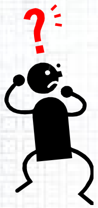
NO ↓

$$k = k+1$$

[참고] 최적화 기법을 사용한 연립 방정식의 풀이

다음과 같은 연립 방정식이 있다.

	변수	: x_1, x_2
Given	**식**	: $f_1(x_1, x_2)=0$
		$f_2(x_1, x_2)=0$
Find	**두 식을 만족하는** x_1^*, x_2^*	



최적화 기법을 이용하여 위의 문제를 풀 수 있을까?

식 $f_1(x_1, x_2)=0$**와** $f_1(x_1, x_2)=0$**는 점**(x_1^*, x_2^*) **에서만 만족한다.**

즉 점(x_1^*, x_2^*)**가 아닌 곳에서는** $f_1(x_1, x_2)\neq 0$, $f_1(x_1, x_2)\neq 0$**이다.**

따라서 다음과 같은 목적함수를 최소화 하는 점 (x_1^*, x_2^*)**가 연립 방정식의 해가 된다.**

$$G=\left(f_1\ (x_1,x_2)\right)^2+\left(f_2\ (x_1,x_2)\right)^2$$

$\left(f_1\ (x_1,x_2)\right)^2\geq 0, \left(f_2\ (x_1,x_2)\right)^2\geq 0$ **이기 때문에 목적함수** G**를 최소화 하는 곳은** $f_1\ (x_1,x_2)=0, f_2\ (x_1,x_2)=0$ **인 점이다.**

함수 G**가 최소 값을 가질 필요 조건**

$$\frac{\partial G}{\partial f_1}=2f_1=0,\ \frac{\partial G}{\partial f_2}=2f_2=0$$

[참고] 최적화 기법을 사용한 연립 방정식의 풀이
예) 프로펠러 주요치수 결정 문제

프로펠러 주요치수 결정 문제의 변수

D_P [m] : 프로펠러 직경, v [m/s] : 배의 속력, P [kW] : 디젤엔진이 프로펠러에 전달하는 마력, n [1/s] : 프로펠러 회전수

A_E/A_O : 프로펠러 전개 면적비, P_i [m] : 프로펠러 피치, $R_T(v)$ [kN] : 선박의 속력에 따른 저항, z : 프로펠러 날개수

프로펠러 주요치수 결정 문제의 제약 조건 및 목적 함수

- 조건식1 : 디젤엔진이 전달한 Torque를 프로펠러가 흡수하는 조건

$$\frac{P}{2\pi n} = \rho \cdot n^2 \cdot D_P{}^5 \cdot K_Q$$

- 조건식2 : 배가 어떤 속력에서 필요로 하는 추력을 프로펠러가 내야 하는 조건

$$\frac{R_T}{1-t} = \rho \cdot n^2 \cdot D_P{}^4 \cdot K_T$$

1.

프로펠러 주요치수 결정 문제 - Case #1	
Given	$P[kW]$, $n[1/s]$, $R_T(v)[kN]$, z
Find	$D_p[m]$, $v[m/s]$, $P_i[m]$, A_E/A_o

목적 함수 : Find Maximum η_0

$$\eta_o = \frac{J}{2\pi} \cdot \frac{K_T}{K_Q}$$

Case #1에서 프로펠러 주요치수를 결정

2.

프로펠러 주요치수 결정 문제 - Case #2	
Given	$D_p[m]$, $P_i[m]$, $R_T(v)[kN]$, z, A_E/A_o
Find	$v[m/s]$, $P[kW]$, $n[1/s]$

v를 가정하면 **미지수가 2개, 조건식이 2개인 연립 방정식**이 된다.

조건식 1, 2를 다음과 같이 수정한다.

$$\frac{P}{2\pi n} - \rho \cdot n^2 \cdot D_P{}^5 \cdot K_Q = 0 \Rightarrow f_1(P,n) = 0$$

$$\frac{R_T}{1-t} - \rho \cdot n^2 \cdot D_P{}^4 \cdot K_T = 0 \Rightarrow f_2(P,n) = 0$$

다음의 목적함수를 최소화 하는 P, n을 구하면 $f_1(P,n)=0,\ f_2(P,n)=0$를 만족하는 P, n을 구할 수 있다.

$$G = \left(f_1\ (P,n)\right)^2 + \left(\ f_2\ (P,n)\right)^2$$

함수 G가 최소 값을 가질 필요 조건

$$\partial G/\partial f_1 = 2f_1 = 0,\ \partial G/\partial f_2 = 2f_2 = 0$$

[참고] 함수 값이 최소가 되기 위한 필요 조건

1변수 최적화 문제의 경우

Given 목적 함수: $f(x)$

Find 목적 함수를 최소화하는 점 x^*

목적 함수를 테일러 전개

$$f(x^*+\Delta x)=f(x^*)+\frac{\partial f}{\partial x}\Delta x+\frac{1}{2}\frac{\partial^2 f}{\partial x^2}(\Delta x)^2+\cdots$$

$$f(x^*+\Delta x)-f(x^*)=\frac{\partial f}{\partial x}\Delta x+\frac{1}{2}\frac{\partial^2 f}{\partial x^2}(\Delta x)^2+\cdots$$

1. 최적화

$$f(x^*+\Delta x)-f(x^*)=\frac{\partial f}{\partial x}\Delta x+\cancel{\frac{1}{2}\frac{\partial^2 f}{\partial x^2}(\Delta x)^2+\cdots}$$

3차 이상의 항은 무시

x^*에서 함수 값이 최소라 하면

LHS $\quad f(x^*+\Delta x)-f(x^*)\geq 0$

RHS $\quad \Delta x$가 독립이므로(부호를 알 수 없으므로)

$$\frac{\partial f}{\partial x}=0 \qquad \frac{\partial^2 f}{\partial x^2}>0$$

필요 조건 충분 조건

2. Variational Method

$$\underbrace{f(x^*+\Delta x)-f(x^*)}_{df}=\frac{\partial f}{\partial x}\Delta x+\cancel{\frac{1}{2}\frac{\partial^2 f}{\partial x^2}(\Delta x)^2+\cdots}$$

2차 이상의 항은 무시

x^*에서 함수 값이 최소라 하면 전미분 df는 0이다.
(예: 산 정상에서는 아주 조금 옆으로 이동해도 높이 변화가 없다.[1])

LHS $\quad df=\frac{\partial f}{\partial x}\Delta x=0$

RHS $\quad \Delta x$가 독립이므로(부호를 알 수 없으므로)

$$\frac{\partial f}{\partial x}=0$$

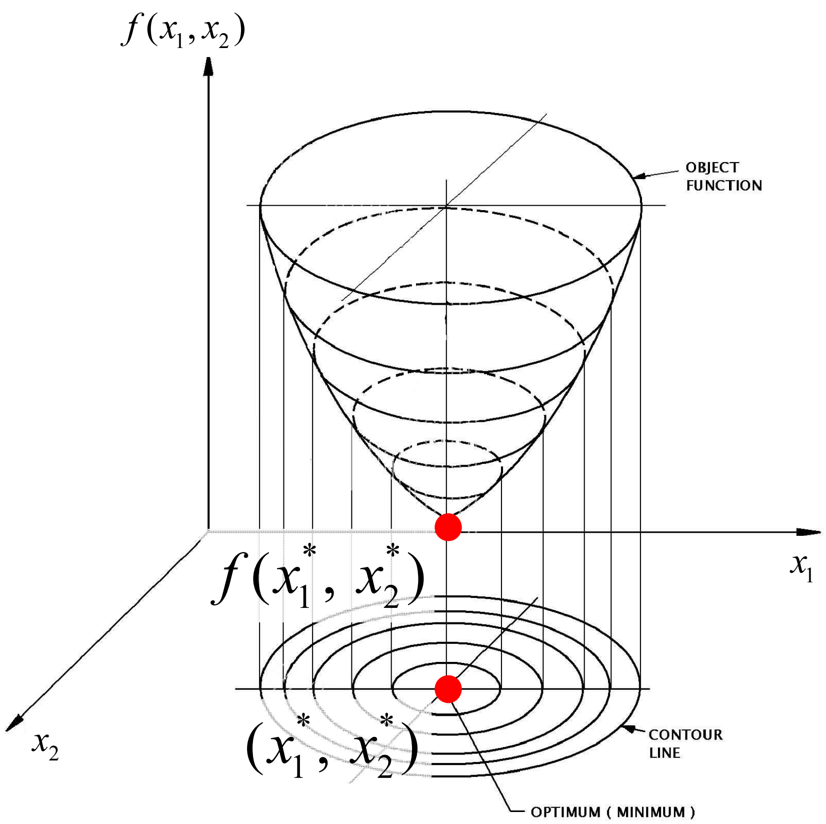
필요 조건

1) On the peak of a mountain all points of the infinitesimal neighbourhook must gave the same height(to first order), which means that *the rate of change of the height must be zero in shatever direction we proceed.*
"LANCZOS, The variational priciples of mechanics, University of toronto press, 1970, p.36"

[참고] 함수의 상점(Stationary Point)

Given: minimize $f(x_1, x_2)$

Find: **상점** $(x_1{*}, x_2{*})$



- **어떤 점** $(x_1{*}, x_2{*})$**에서 미소변위**(dx_1, dx_2) **만큼 이동한 점에서의 함수값의 변화량** df**는 다음과 같다.**

$$df = \frac{\partial f}{\partial x_1} dx_1 + \frac{\partial f}{\partial x_2} dx_2$$

모든 방향으로의 함수값의 변화량 df**가 0인 점을 상점(Stationary Point)이라고 하며, 최소점, 최대점, 안장점**(Saddle Point)**은 상점에 속한다.**

참고: 일반적인 공학적 최적화 문제에서는 목적 함수의 최적값 (Optimum Value)**보다는 최적점**(Optimum Point)**이 더 중요하다.**
[예시] 건조비를 최소화 하는 선박의 주요치수(L, B, D, C_B)**가 건조비 자체보다 중요한 개념이다.**

Ch1. 최적 설계 개요

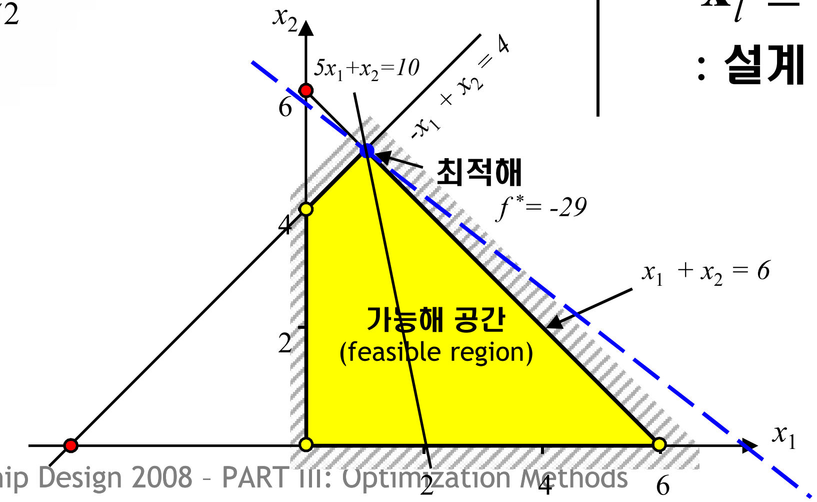
1.1 최적 설계 문제의 일반적 정식화

1.2 최적 설계 문제의 구성 요소

1.3 최적화 기법의 분류

1.1 최적 설계 문제의 일반적 정식화

Minimize $f = -4x_1 - 5x_2$	*Minimize* $f(\mathbf{x})$	Objective Function
Subject to	*Subject to*	Constraints
$-x_1 + x_2 \le 4 \Rightarrow -x_1 + x_2 - 4 \le 0$ $x_1 + x_2 \le 6 \Rightarrow x_1 + x_2 - 6 \le 0$	$g_j(\mathbf{x}) \le 0, j = 1, \cdots, m$: 부등호 제약 조건(Inequality Constraint)	
$5x_1 + x_2 = 10 \Rightarrow 5x_1 + x_2 - 10 = 0$	$h_k(\mathbf{x}) = 0, k = 1, \cdots, p$: 등호 제약 조건(Equality Constraint)	
$0 \le x_1, x_2$	$\mathbf{x}_l \le \mathbf{x} \le \mathbf{x}_u$: 설계 변수 벡터에 대한 상·하한값 제약 조건	



@SDAL Advanced Ship Design Automation Lab. http://asdal.snu.ac.kr

1.2 최적 설계 문제의 구성 요소(1)

▪ 설계 변수(Design Variable)

- 설계하고자 하는 치수, 위치 등을 나타내는 변수로서 자유 변수(Free Variable) 또는 독립 변수(Independent Variable)라고 함
- 종속 변수(Dependent Variable)
  - 설계 변수의 결정 후 종속적으로 결정되어지는 변수

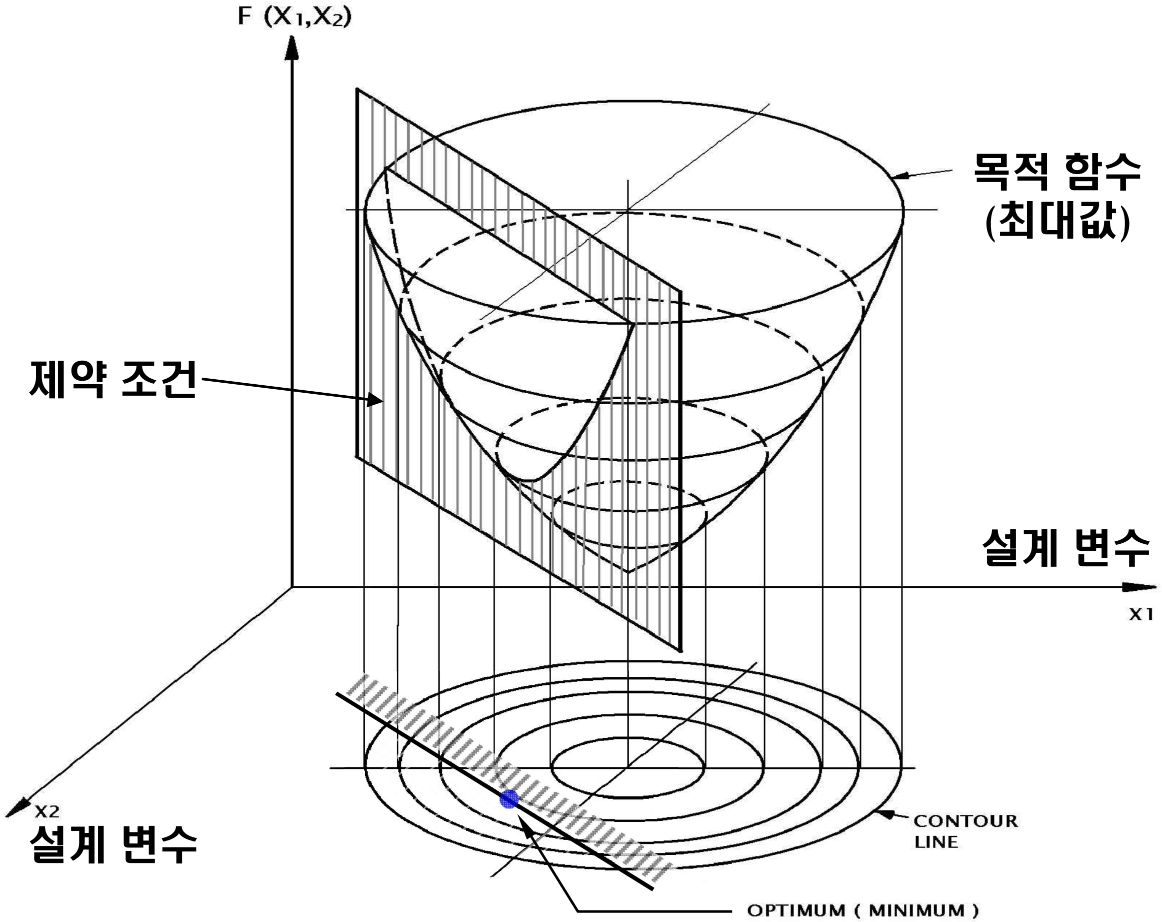
▪ 제약 조건(Constraint)

- 설계에서 기능상 요구되는 조건 또는 크기의 제한 등을 정의
- 부등호 제약 조건(Inequality Constraint), 등호 제약 조건(Equality Constraint)

1.2 최적 설계 문제의 구성 요소(2)

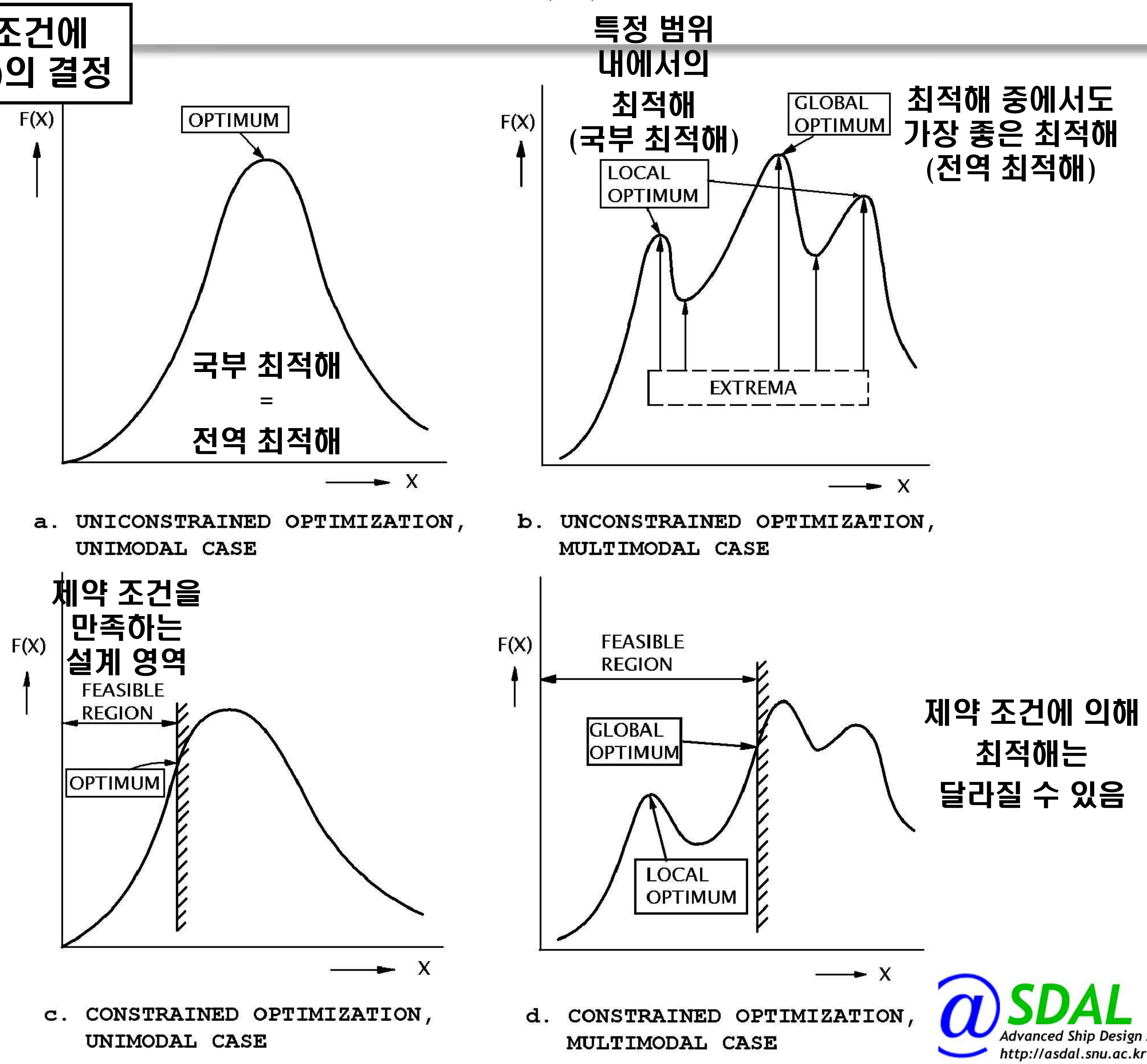
■ 목적 함수(Objective Function)

- 최적(Optimum)을 나타내는 기준으로 비용, 무게 등과 같은 값을 비교하여 어느 설계 대안(Design Alternative)이 보다 나은지를 나타낼 수 있는 함수

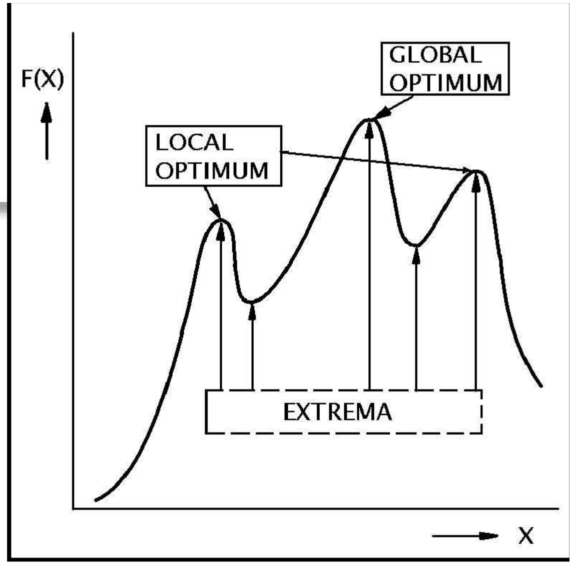


1.2 최적 설계 문제의 구성 요소(3)

목적 함수와 제약 조건에
따른 최적해(최대값)의 결정



1.3 최적화 기법의 분류(1)



- **전역 최적화 기법(Global Optimization Methods)**
  - 장점
    - 다수의 국부 최적해(Local Optima)를 가진 대규모 문제에 적합
  - 단점
    - 최적해를 얻기 위해 많은 Iteration이 필요(긴 계산 시간 요구)
  - Genetic Algorithms(GA), Simulated Annealing, etc.

- **국부 최적화 기법(Local Optimization Methods)**
  - 장점
    - 최적해를 얻기 위해 상대적으로 적은 Iteration이 필요(짧은 계산 시간 요구)
  - 단점
    - 초기해(Starting Point)에 가까운 국부 최적해를 도출(거의 국부 최적해만 찾음)
  - Method of Feasible Directions(MFD), Sequential Quadratic Programming(SQP), Multi-Start Optimization Method, etc.

1.3 최적화 문제의 분류(2)

- 제약 조건의 유무
  - 비제약 최적화 문제(Unconstrained Optimization Problem)
    - 제약 조건이 없는 최적화 문제
    - 직접 탐사법(Hooke & Jeeves Method, Nelder & Mead's Simplex Method), Gradient 방법(Steepest Descent Method, etc.), etc.
  - 제약 최적화 문제(Constrained Optimization Problem)
    - 제약 조건이 있는 최적화 문제
    - Penalty Function Method, Sequential Linear Programming, Constrained Steepest Descent Method, Method of Feasible Directions(MFD), Sequential Quadratic Programming(SQP), etc.
- 목적 함수의 수
  - 단일 최적화 문제(Single-Objective Optimization Problem)
  - 다중 최적화 문제(Multi-Objective Optimization Problem)
    - Weighting Method, Constraint Method

최적화 문제의 분류와 해법(2009.01.15 수정)

<table>
<tr><td rowspan="2"></td><td colspan="2">비제약 최적화 문제</td><td colspan="3">제약 최적화 문제</td></tr>
<tr><td>선형</td><td>비선형</td><td>선형</td><td colspan="2">비선형</td></tr>
<tr><td>목적 함수 (예시)</td><td>minimize $f(\mathbf{x})$
$f(\mathbf{x}) = x_1 + 2x_2$</td><td>minimize $f(\mathbf{x})$
$f(\mathbf{x}) = x_1^2 + x_2^2 - 3x_1x_2$</td><td>minimize $f(\mathbf{x})$
$f(\mathbf{x}) = x_1 + 2x_2$</td><td>minimize $f(\mathbf{x})$
$f(\mathbf{x}) = x_1^2 + x_2^2 - 3x_1x_2$</td><td>minimize $f(\mathbf{x})$
$f(\mathbf{x}) = x_1^2 + x_2^2 - 3x_1x_2$</td></tr>
<tr><td>제약 조건 (예시)</td><td>없음</td><td>없음</td><td>$h(\mathbf{x}) = x_1 + 5x_2 = 0$
$g(\mathbf{x}) = -x_1 \le 0$</td><td>$h(\mathbf{x}) = x_1 + 5x_2 = 0$
$g(\mathbf{x}) = -x_1 \le 0$</td><td>$g_1(\mathbf{x}) = \frac{1}{6}x_1^2 + \frac{1}{6}x_2^2 - 1.0 \le 0$
$g_2(\mathbf{x}) = -x_1 \le 0$</td></tr>
<tr><td rowspan="3">국부 최적화 방법</td><td colspan="2" rowspan="3">① 직접 탐사법
- Hooke&Jeeves
- Nelder&Mead

② Gradient 방법
- Steepest Descent방법[4]
- 공액 경사도 방법[4](Conjugate Gradient 방법)
- Newton 방법[5]
- Davidon-Fletcher-Powell(DFP) 방법[5]
- Broyden-Fletcher-Goldfarb-Shanno(BFGS) 방법[5]</td><td>Penalty Function으로 해를 구할 수 있으나 일반적으로 선형 계획법을 사용</td><td colspan="2">- Penalty Function[1]을 구성한 후 비제약 최적화 문제로 변환한 후 해를 구함</td></tr>
<tr><td>- Simplex 방법 (선형 계획법, Linear Programming)</td><td rowspan="2">- 2차 계획 법 (Quadratic Programming)</td><td>- 근사화 방법 - SLP
선형 계획 문제[2]로 근사화 후 개선된 탐색점을 찾고, 그 점에서 다시 선형 계획 문제를 푸는 방법</td></tr>
<tr><td></td><td>- 근사화 방법 - SQP
2차 계획 문제[3]로 근사화 후 개선된 탐색점을 찾고, 그 점에서 다시 2차 계획 문제를 푸는 방법</td></tr>
<tr><td>전역 최적화 방법</td><td colspan="5">Genetic Algorithms(GA), Simulated Annealing, etc.</td></tr>
</table>

1) Penalty Function
제약 조건의 위배량을 원래 목적 함수에 더한 수정된 목적 함수

2) 선형 계획 문제
(Linear Programming Problem)
목적함수: 1차 형식
제약조건: 1차 형식

3) 2차 계획 문제
(Quadratic Programming Problem)
목적함수: 2차 형식
제약조건: 1차 형식

4) Gradient 방법 중
함수의 1차 미분만을 고려하는 방법

5) Gradient 방법 중
함수의 2차 미분까지 고려하는 방법

Ch2. 비제약 최적화 기법

2.1 Gradient 방법
- Steepest Descent 방법
- 공액 경사도 방법(Conjugate Gradient 방법)
- Newton의 방법
- Davidon-Fletcher-Powell(DFP) 방법
- Broyden-Fletcher-Goldfarb-Shanno(BFGS) 방법

2.2 황금 분할에 의한 1차원 탐색 방법

2.3 직접 탐사법(Direct Search Method)
- Hooke & Jeeves의 직접 탐사법
- Nelder & Mead의 Simplex 방법

Advanced
Ship
Design
Automation
Laboratory

2.1 Gradient 방법

Advanced
Ship
Design
Automation
Laboratory

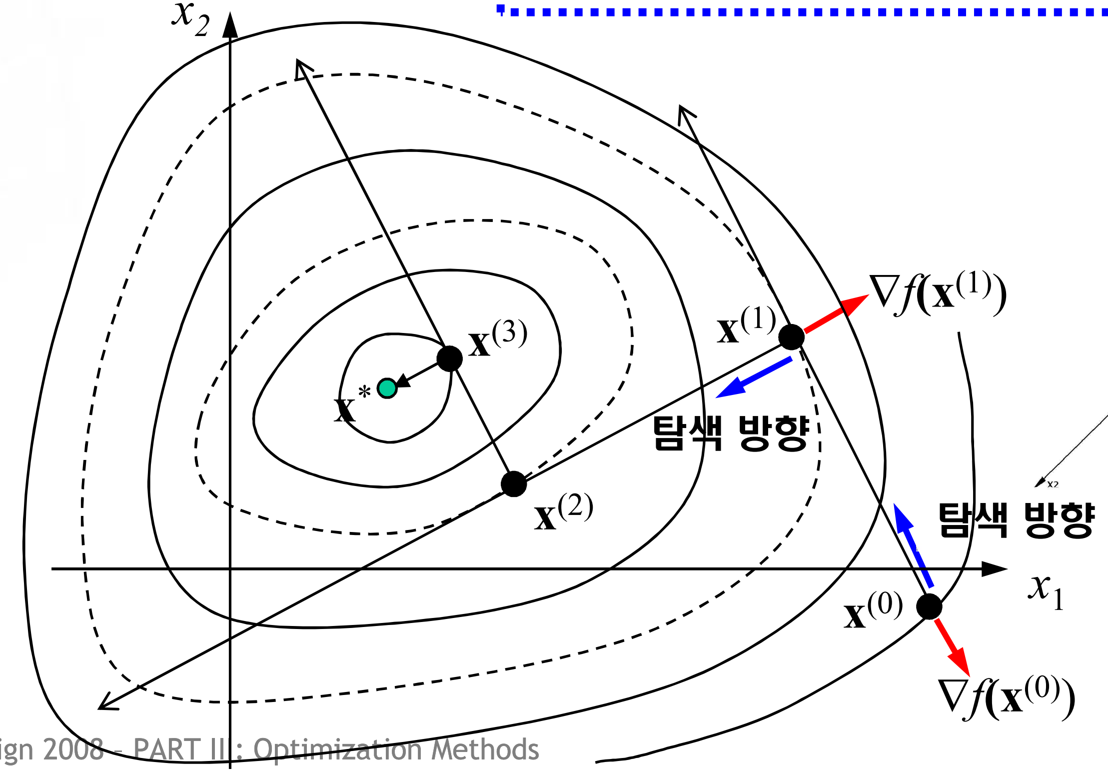
2.1 Gradient 방법
- Steepest Descent 방법(최속 강하법)

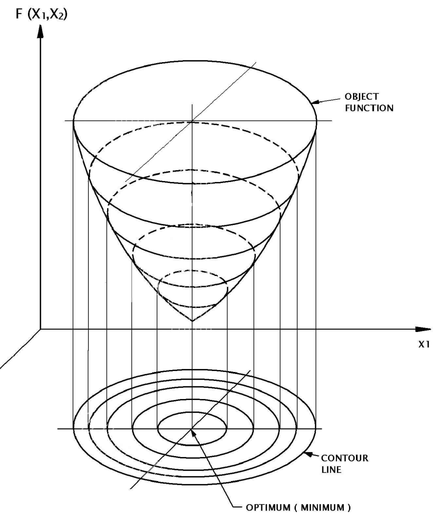
- **탐색 방향**(Search Direction)**을 목적 함수의** Gradient Vector**의 반대 방향으로 가정하고 순차적으로 최적해를 찾는 방법**
  - ➡ Gradient Vector($\nabla \boldsymbol{f}(\mathbf{x})$): **함수 값이 최대로 증가하는 방향**

탐색 방향 : 함수 값이 가장 많이 감소하는 방향

$$\mathbf{d} = -\mathbf{c} \equiv -\nabla f(\mathbf{x})$$

목적 함수를 최소화 하는 문제일 경우



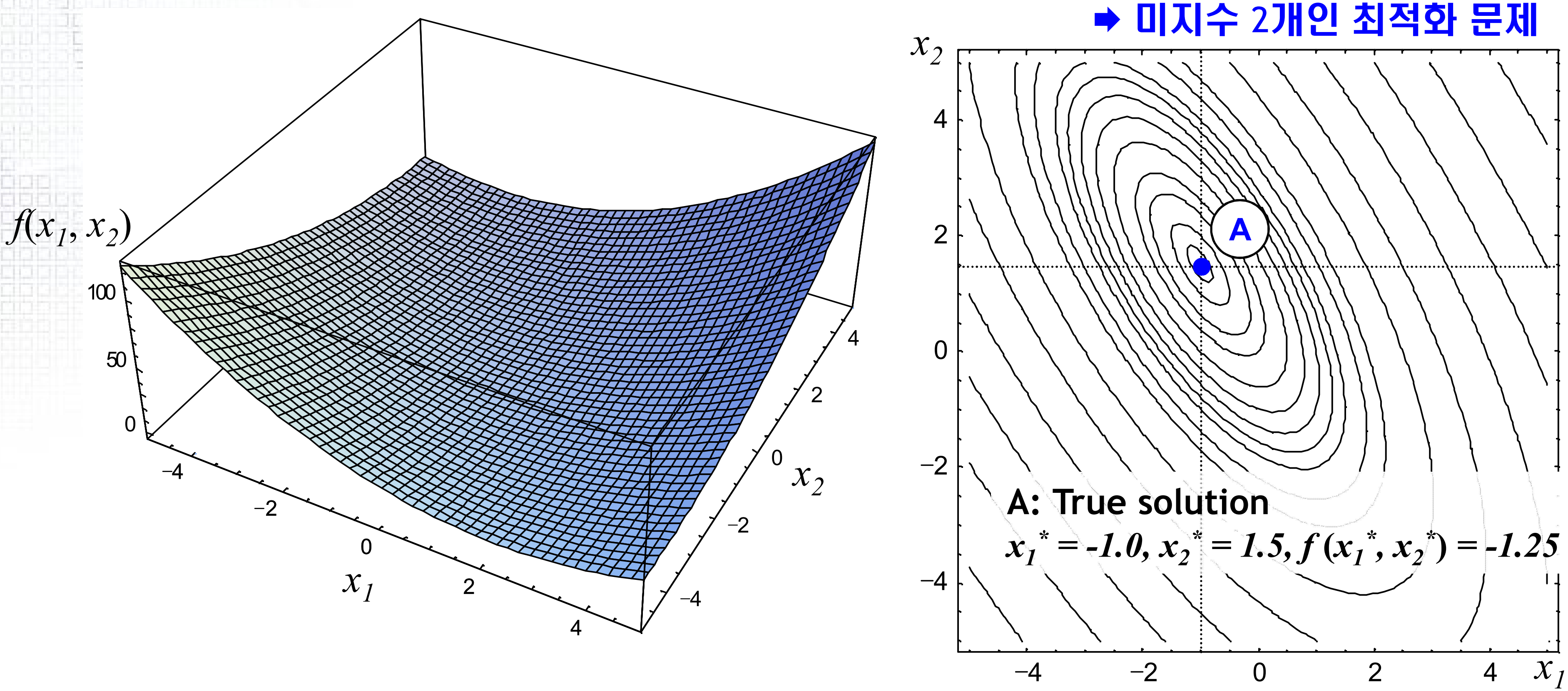


비제약 최적화 문제

- Steepest descent **방법(최속 강하법, 최대 경사법)을 이용하여** 2**변수 함수의 최소점을 구하시오. 단, 시작점 x**$^{(0)}$ = (0, 0), convergence tolerance ε = 0.001**이며, x**$^{(3)}$**까지 구하시오.**

$$\textit{Minimize} \quad f(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$

➡ **미지수 2개인 최적화 문제**



비제약 최적화 문제
- Steepest descent 방법을 이용한 해법(1)

$$Minimize \;\; f(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$

시작점 $\mathbf{x}^{(0)} = (0, 0)$

$$\nabla f(\mathbf{x}) = \nabla f(x_1, x_2) = \begin{pmatrix} 1 + 4x_1 + 2x_2 \\ -1 + 2x_1 + 2x_2 \end{pmatrix}$$

- **단계 1 -** $\mathbf{x}^{(1)}$ 구하기

$$\nabla f(\mathbf{x}^{(0)}) = \nabla f\begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 + 4x_1 + 2x_2 \\ -1 + 2x_1 + 2x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} - \alpha^{(0)} \nabla f(\mathbf{x}^{(0)})$$

$$= \begin{pmatrix} 0 \\ 0 \end{pmatrix} - \alpha \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} -\alpha \\ \alpha \end{pmatrix}$$

편의상 $\alpha^{(0)}$을 α로 대체함

$\mathbf{x}^{(1)} = (-\alpha, \alpha)$를 목적함수에 대입

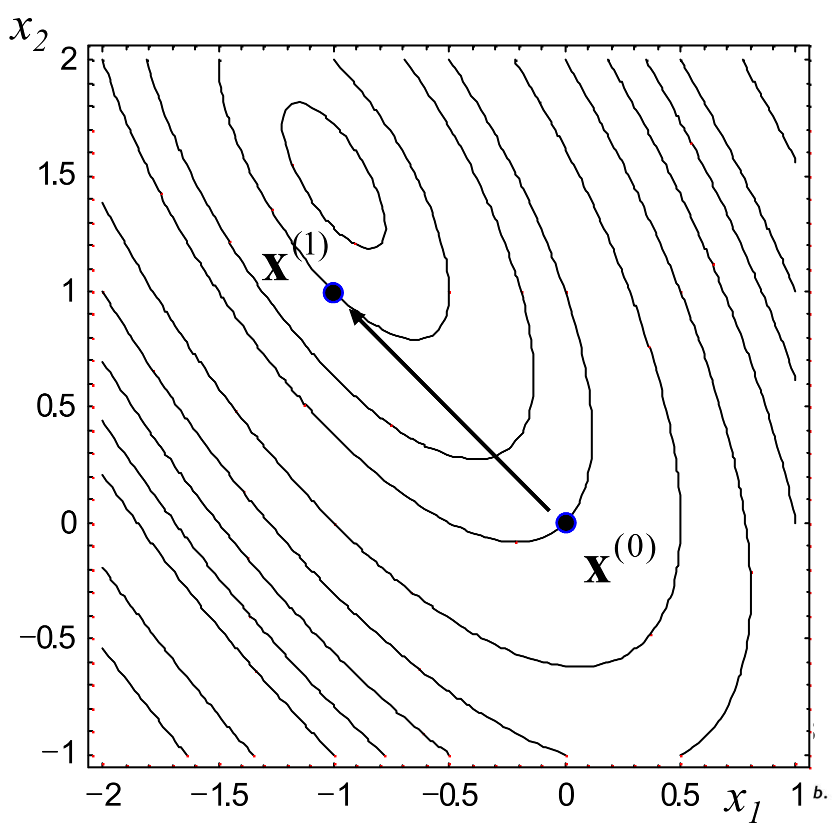
$$f(\mathbf{x}^{(1)}) = -\alpha - \alpha + 2\alpha^2 - 2\alpha^2 + \alpha^2$$
$$= \alpha^2 - 2\alpha$$

함수 $f(\mathbf{x}^{(1)})$가 최소값을 가질 조건

$$\boxed{\frac{df(\mathbf{x}^{(1)})}{d\alpha}} = 2\alpha - 2 = 0 \text{으로부터 } \alpha = 1.0$$

$$\therefore \mathbf{x}^{(1)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

함수 f를 어떻게 α 로 미분 할 수 있는가?



비제약 최적화 문제
- [참고] x의 함수 f를 x가 아닌 다른 변수로 미분

$$Minimize\ \boxed{f(x_1,x_2)} = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$ 시작점 $\mathbf{x}^{(0)} = (0, 0)$

■ **단계 1 - $\mathbf{x}^{(1)}$구하기**

$$\nabla f(\mathbf{x}^{(0)}) = \nabla f\begin{pmatrix}0\\0\end{pmatrix} = \begin{pmatrix}1+4x_1+2x_2\\-1+2x_1+2x_2\end{pmatrix} = \begin{pmatrix}1\\-1\end{pmatrix}$$

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} - \alpha^{(0)}\nabla f(\mathbf{x}^{(0)})$$

$$= \begin{pmatrix}0\\0\end{pmatrix} - \alpha\begin{pmatrix}1\\-1\end{pmatrix} = \begin{pmatrix}-\alpha\\\alpha\end{pmatrix}$$ 편의상 $\alpha^{(0)}$을 α로 대체함

$\boxed{\mathbf{x}^{(1)} = (-\alpha,\alpha)}$를 목적함수에 대입

$$f(\mathbf{x}^{(1)}) = -\alpha - \alpha + 2\alpha^2 - 2\alpha^2 + \alpha^2$$

$$= \alpha^2 - 2\alpha$$

함수 $f(\mathbf{x}^{(1)})$가 최소값을 가질 조건

$\boxed{\dfrac{df(\mathbf{x}^{(1)})}{d\alpha}} = 2\alpha - 2 = 0$으로부터 $\alpha = 1.0$ $\quad\therefore \mathbf{x}^{(1)} = \begin{pmatrix}-1\\1\end{pmatrix}$

함수 f를 어떻게 α로 미분 할 수 있는가?

$f(x_1,x_2) = f(\mathbf{x})$: **함수 f는 x에 대한 함수**

$\mathbf{x}^{(1)} = (-\alpha,\alpha)$: **$\mathbf{x}^{(1)}$는 α에 대한 함수**

→ $\mathbf{x}^{(1)}$를 함수 f에 대입 하였으므로
함수 f는 α에 대한 함수이고,
α로 미분 가능하다.

이와 유사하게 다음도 생각해 볼 수 있다.

함수 $f(\mathbf{x}^* + \Delta\mathbf{x})$가 최소 값을 가질 조건

$f(\mathbf{x}^* + \Delta\mathbf{x})$ **의 테일러 전개(2차항까지 고려)**

$$f(\mathbf{x}^* + \Delta\mathbf{x}) = f(\mathbf{x}^*) + \mathbf{c}^T\Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^T\mathbf{H}(\mathbf{x}^*)\,\Delta\mathbf{x}$$

$$f(\mathbf{x}^* + \Delta\mathbf{x}) - f(\mathbf{x}^*) = \mathbf{c}^T\Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^T\mathbf{H}(\mathbf{x}^*)\,\Delta\mathbf{x}$$

여기서 x*는 상수이므로, $\Delta\mathbf{x}$를 변수로 취급하여

$$f(\Delta\mathbf{x}) = \mathbf{c}^T\Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^T\mathbf{H}(\mathbf{x}^*)\,\Delta\mathbf{x}$$

함수 f가 최소 값을 가질 조건

$$\frac{df(\Delta\mathbf{x})}{d\Delta\mathbf{x}} = \mathbf{c} + \mathbf{H}(\mathbf{x}^*)\,\Delta\mathbf{x} = 0$$

$$\Rightarrow \mathbf{H}(\mathbf{x}^*)\,\Delta\mathbf{x} = -\mathbf{c}$$

$$\Rightarrow \boxed{\Delta\mathbf{x} = -\mathbf{H}(\mathbf{x}^*)^{-1}\mathbf{c}}$$

비제약 최적화 문제
- Steepest descent 방법을 이용한 해법(2)

$$Minimize\ f(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$

- **단계 2** - $\mathbf{x}^{(2)}$구하기

$$\nabla f(\mathbf{x}^{(1)}) = \nabla f\begin{pmatrix}-1\\1\end{pmatrix} = \begin{pmatrix}1+4x_1+2x_2\\-1+2x_1+2x_2\end{pmatrix} = \begin{pmatrix}-1\\-1\end{pmatrix}$$

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} - \alpha^{(1)}\nabla f(\mathbf{x}^{(1)})$$

$$= \begin{pmatrix}-1\\1\end{pmatrix} - \alpha\begin{pmatrix}-1\\-1\end{pmatrix} = \begin{pmatrix}-1+\alpha\\1+\alpha\end{pmatrix}$$ 편의상 $\alpha^{(1)}$을 α로 대체함

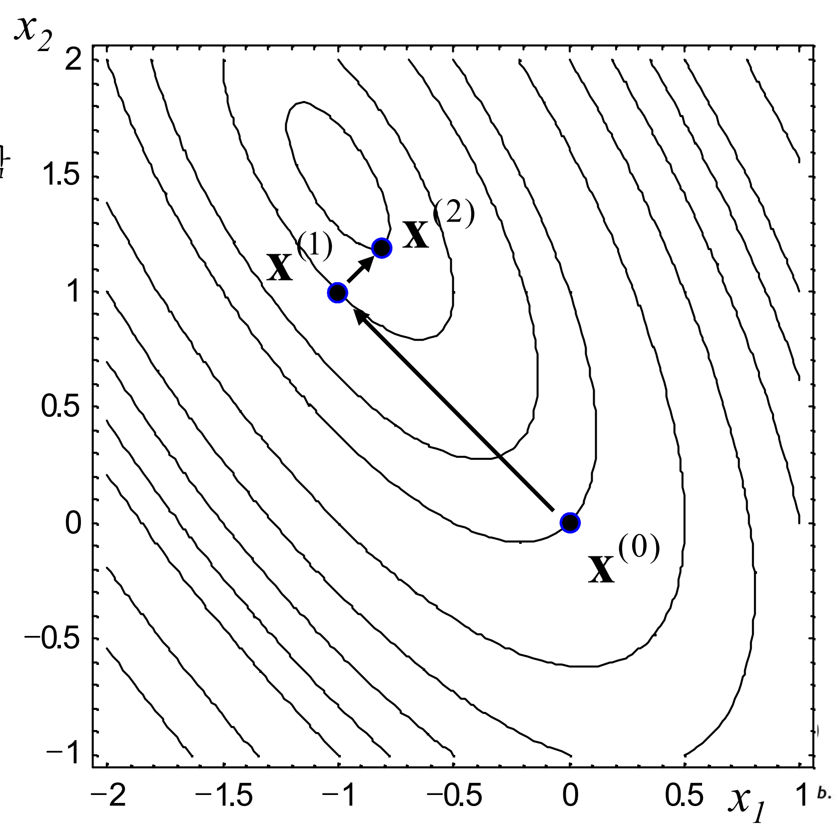
$\mathbf{x}^{(2)} = (-1+\alpha, 1+\alpha)$를 목적함수에 대입

$$f(\mathbf{x}^{(2)}) = 5\alpha^2 - 2\alpha - 1$$

함수 $f(\mathbf{x}^{(2)})$가 최소값을 가질 조건

$\dfrac{df(\mathbf{x}^{(2)})}{d\alpha} = 10\alpha - 2 = 0$으로부터 $\alpha = 0.2$

$$\therefore \mathbf{x}^{(2)} = \begin{pmatrix}-0.8\\1.2\end{pmatrix}$$



비제약 최적화 문제
- Steepest descent 방법을 이용한 해법(3)

- **단계 3 -** $\mathbf{x}^{(3)}$구하기

$$\nabla f(\mathbf{x}^{(1)}) = \nabla f\begin{pmatrix} -0.8 \\ 1.2 \end{pmatrix} = \begin{pmatrix} 1+4x_1+2x_2 \\ -1+2x_1+2x_2 \end{pmatrix} = \begin{pmatrix} 0.2 \\ -0.2 \end{pmatrix}$$

$$\mathbf{x}^{(3)} = \mathbf{x}^{(2)} - \alpha^{(2)}\nabla f(\mathbf{x}^{(2)})$$

$$= \begin{pmatrix} -0.8 \\ 1.2 \end{pmatrix} - \alpha\begin{pmatrix} 0.2 \\ -0.2 \end{pmatrix} = \begin{pmatrix} -0.8-0.2\alpha \\ 1.2+0.2\alpha \end{pmatrix}$$ 편의상 $\alpha^{(2)}$을 α로 대체함

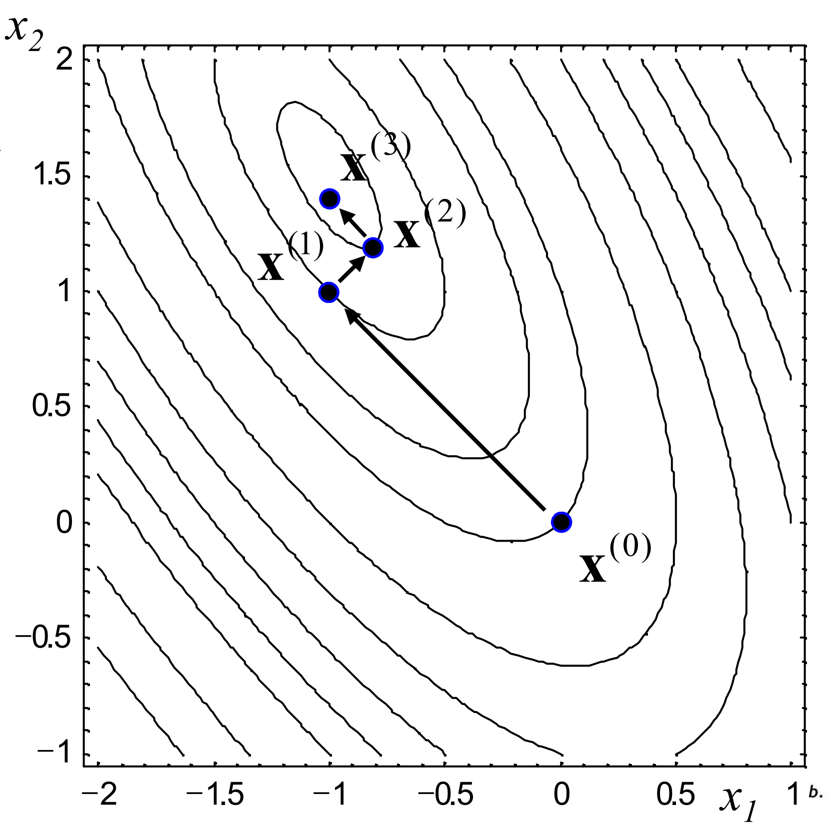
$\mathbf{x}^{(3)} = (-0.8-0.2\alpha, 1.2+0.2\alpha)$를 목적함수에 대입

$f(\mathbf{x}^{(3)}) = 0.04\alpha^2 - 0.08\alpha - 1.2$

함수 $f(\mathbf{x}^{(3)})$가 최소값을 가질 조건

$\dfrac{df(\mathbf{x}^{(3)})}{d\alpha} = 0.08\alpha - 0.08 = 0$으로부터 $\alpha = 1.0$

$$\therefore \mathbf{x}^{(3)} = \begin{pmatrix} -1 \\ 1.4 \end{pmatrix}$$

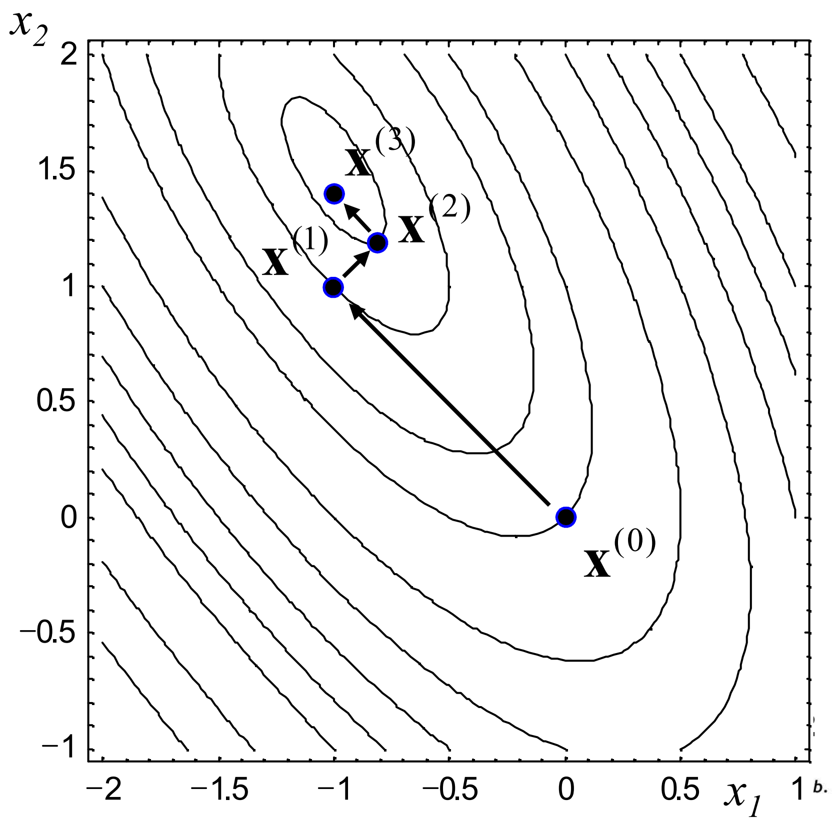


비제약 최적화 문제
- Steepest descent 방법을 이용한 해법(4)

- **단계 4 -** 최적해 구하기

이와 같은 과정을 반복하여 $\left|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right| \le \varepsilon$일 경우

중지하며 그때의 $\mathbf{x}^{(k+1)}$이 최적해가 된다.



Gradient 방법
- 공액 경사도 방법(Conjugate Gradient 방법)(1)

- 탐색 방향을 수정하여 최속 강하법보다 수렴율을 향상시킨 방법. 현재의 최속 강하 방향에 직전에 사용된 탐색 방향을 척도화 시켜 더한 것을 탐색 방향으로 사용함
  - 단계 1 : 초기 설계점 $\mathbf{x}^{(0)}$를 추정한다. 반복 횟수 번호를 $k=0$으로 둔다. 또한 수렴 매개 변수 ε 을 선정하고 최적 탐색 방향을 계산한다.

$$\mathbf{d}^{(0)} = -\mathbf{c}^{(0)} \equiv -\nabla f(\mathbf{x}^{(0)})$$

    그리고, $\left\|\mathbf{c}^{(0)}\right\| < \varepsilon$ 식을 만족하면 반복 과정을 마치고 만족하지 않으면 단계 4로 간다(공액 경사법과 최속 강하법의 단계 1은 동일함).
  - 단계 2 : 목적 함수의 경사도를 계산한다.

$$\mathbf{c}^{(k)} = \nabla f(\mathbf{x}^{(k)})$$

    여기서, 만일 $\left\|\mathbf{c}^{(k)}\right\| < \varepsilon$ 이면 멈춘다. 그렇지 않으면 계속한다.

Gradient 방법
- 공액 경사도 방법(Conjugate Gradient 방법)(2)

- 단계 3 : 다음과 같이 새로운 탐색 방향을 계산한다.

$$\boxed{\mathbf{d}^{(k)} = -\mathbf{c}^{(k)} + \beta_k \mathbf{d}^{(k-1)}} \rightarrow \text{전 단계의 탐색 방향}$$

$$\beta_k = \left(\left\| \mathbf{c}^{(k)} \right\| / \left\| \mathbf{c}^{(k-1)} \right\| \right)^2$$

전 단계의 탐색 방향을 고려하여 현재의 탐색 방향을 설정한다는 의미

- 단계 4 : $f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)})$ 를 최소화 하는 $\alpha = \alpha_k$ 를 계산한다.
- 단계 5 : 현재의 설계점을 다음과 같이 변경한다.

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

$k = k+1$ 로 두고 단계 2로 간다.

비제약 최적화 문제
- Conjugate gradient 방법을 이용한 해법(1)

$$\textit{Minimize}\;\; f(x_1,x_2)=x_1-x_2+2x_1^2+2x_1x_2+x_2^2$$

$$\nabla f(\mathbf{x})=\nabla f(x_1,x_2)=\begin{pmatrix}1+4x_1+2x_2\\-1+2x_1+2x_2\end{pmatrix}$$

- **단계 1 -** $\mathbf{x}^{(1)}$구하기

$$\nabla f(\mathbf{x}^{(0)})=\nabla f\begin{pmatrix}0\\0\end{pmatrix}=\begin{pmatrix}1+4x_1+2x_2\\-1+2x_1+2x_2\end{pmatrix}=\begin{pmatrix}1\\-1\end{pmatrix}$$

$$\mathbf{x}^{(1)}=\mathbf{x}^{(0)}-\alpha^{(0)}\nabla f(\mathbf{x}^{(0)})$$

$$=\begin{pmatrix}0\\0\end{pmatrix}-\alpha\begin{pmatrix}1\\-1\end{pmatrix}=\begin{pmatrix}-\alpha\\\alpha\end{pmatrix}$$

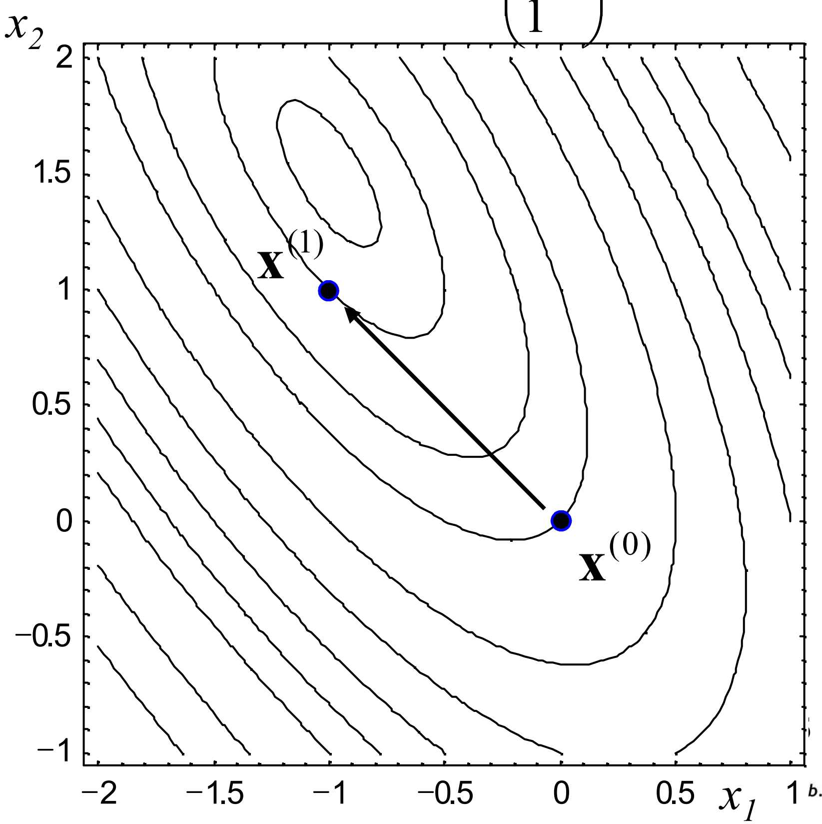
편의상 $\alpha^{(0)}$을 α로 대체함

$$f(\mathbf{x}^{(1)})=\alpha^2-2\alpha$$

함수 $f(\mathbf{x}^{(1)})$가 최소값을 가질 조건

$$\frac{df(\mathbf{x}^{(1)})}{d\alpha}=0\text{으로부터 }\alpha=1.0$$

$$\therefore \mathbf{x}^{(1)}=\begin{pmatrix}-1\\1\end{pmatrix}$$



비제약 최적화 문제
- Conjugate gradient 방법을 이용한 해법(2)

- **단계 2 -** $\mathbf{x}^{(2)}$구하기

$\mathbf{x}^{(1)}$에서 $\nabla f(\mathbf{x}^{(1)})$를 구하고

"*Conjugate direction*"을 구한다.

$$\nabla f(\mathbf{x}^{(1)}) = \nabla f\begin{pmatrix} -1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1+4x_1+2x_2 \\ -1+2x_1+2x_2 \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} - \alpha^{(1)}[-\nabla f(\mathbf{x}^{(1)}) + \frac{\left|\nabla f(\mathbf{x}^{(1)})\right|^2}{\left|\nabla f(\mathbf{x}^{(0)})\right|^2}\cdot \mathbf{d}^{(0)}]$$

로 가정하고 $f(\mathbf{x}^{(2)})$이 최소가 되는 $\alpha^{(1)}$를 구한다.

여기서, $\mathbf{d}^{(0)} = -\nabla f(\mathbf{x}^{(0)})$이다.

비제약 최적화 문제
- Conjugate gradient 방법을 이용한 해법(3)

$$\mathbf{x}^{(2)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix} + \alpha \left\{ -\begin{pmatrix} -1 \\ -1 \end{pmatrix} + \frac{2}{2}(-1)\begin{pmatrix} 1 \\ -1 \end{pmatrix} \right\}$$

$$= \begin{pmatrix} -1 \\ 1+2\alpha \end{pmatrix}$$

편의상 $\alpha^{(1)}$을 α로 대체함

$$f(\mathbf{x}^{(2)}) = 4\alpha^2 - 2\alpha - 1$$

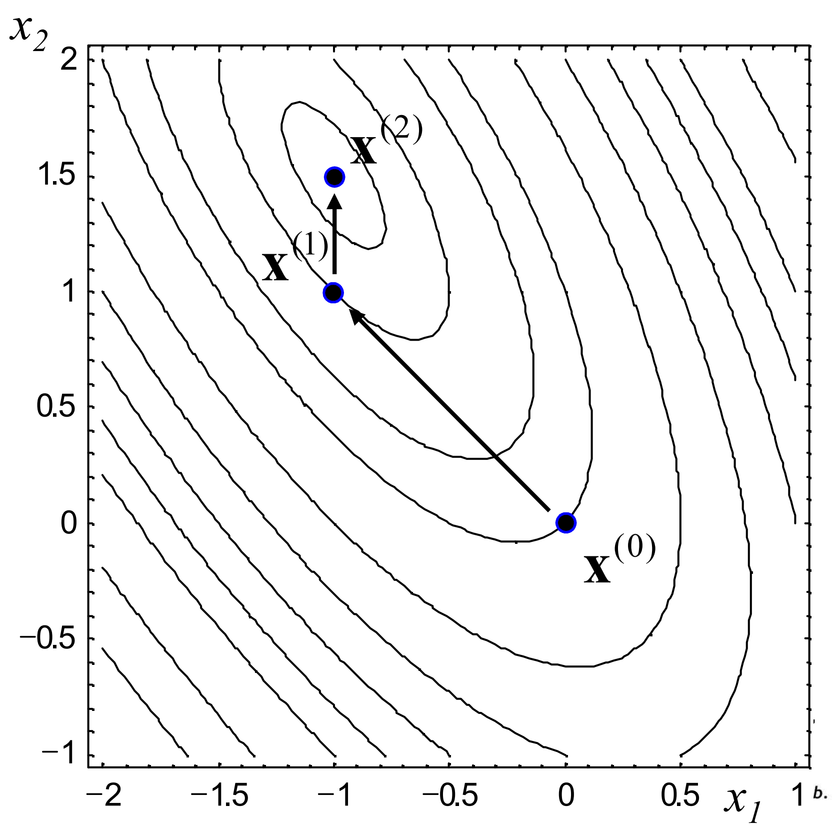
함수 $f(\mathbf{x}^{(2)})$가 최소값을 가질 조건

$\dfrac{df(\mathbf{x}^{(2)})}{d\alpha} = 0$으로부터 $\alpha = 0.25$

$$\therefore \mathbf{x}^{(2)} = \begin{pmatrix} -1 \\ 1.5 \end{pmatrix}$$

$\left\| \mathbf{c}^{(2)} \right\| = 0 \le \varepsilon$이므로

$\mathbf{x}^{(2)}$가 최적점이 된다.



Gradient 방법
- Newton의 방법

- 헷세 행렬을 사용하여 탐색 방법을 개선한 방법
  - 단계 1 : 시작점 $\mathbf{x}^{(0)}$을 추정한다. 반복 회수 번호를 $k=0$으로 둔다. 그리고 종료 기준으로 허용치 ε을 선정한다.
  - 단계 2 : $i=1$ 에서 n 까지 $c_i^{(k)} = \partial f(\mathbf{x}^{(k)}) / \partial x_i$를 계산한다.
  - 단계 3 : 헷세 행렬을 계산한다.

$$H(\mathbf{x}^{(k)}) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j}\right], \qquad i=1,\cdots,n;\ j=1,\cdots,n$$

  - 단계 4 : 탐색 방향을 계산한다.

$$\boxed{\mathbf{d}^{(k)} = \Delta\mathbf{x}^{(k)} = -\mathbf{H}^{-1}\mathbf{c}^{(k)}}$$

$f(\mathbf{x}^* + \Delta\mathbf{x}) = f(\mathbf{x}^*) + \mathbf{c}^T \Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^T \mathbf{H}(\mathbf{x}^*)\,\Delta\mathbf{x}$ 일 때 최소값을 가질 조건

$df(\Delta\mathbf{x}) / d\Delta\mathbf{x} = \mathbf{c} + \mathbf{H}(\mathbf{x}^*)\Delta\mathbf{x} = 0$

$\Rightarrow \mathbf{H}(\mathbf{x}^*)\,\Delta\mathbf{x} = -\mathbf{c} \Rightarrow\ \Delta\mathbf{x} = -\mathbf{H}(\mathbf{x}^*)^{-1}\mathbf{c}$

  - 단계 5 : 설계점을 $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$ 로 수정한다. 이때 α_k는 $f(\mathbf{x}^{(k)} + \alpha\mathbf{d}^{(k)})$를 최소화하도록 계산된다. α를 계산하기 위해서는 어떠한 1차원 탐색법도 사용할 수 있다. 초기 이동 거리 추정으로 $\alpha = 1$로 가정한다.
  - 단계 6 : $k = k+1$로 두고 단계 2를 수행한다.

[참고] 2변수 함수의 테일러 전개(Taylor Series Expansion)(1)

2변수 함수 $f(x_1, x_2)$에 대한 점 (x_1^*, x_2^*)에서의 테일러 전개식

$$f(x_1,x_2)=f(x_1^*,x_2^*)+\frac{\partial f}{\partial x_1}(x_1-x_1^*)+\frac{\partial f}{\partial x_2}(x_2-x_2^*)$$

$$+\frac{1}{2}\left(\frac{\partial^2 f}{\partial x_1^2}(x_1-x_1^*)^2+2\frac{\partial^2 f}{\partial x_1\partial x_2}(x_1-x_1^*)(x_2-x_2^*)+\frac{\partial^2 f}{\partial x_2^2}(x_2-x_2^*)^2\right)+R$$

⬇ 각 항을 다시 표현하면

$$\frac{\partial f}{\partial x_1}(x_1-x_1^*)+\frac{\partial f}{\partial x_2}(x_2-x_2^*)=\begin{bmatrix}\frac{\partial f}{\partial x_1}\\ \frac{\partial f}{\partial x_2}\end{bmatrix}^T\begin{bmatrix}x_1-x_1^*\\ x_2-x_2^*\end{bmatrix}=\underline{\nabla f(\mathbf{x}^*)^T(\mathbf{x}-\mathbf{x}^*)}$$

$$\frac{1}{2}\left(\frac{\partial^2 f}{\partial x_1^2}(x_1-x_1^*)^2+2\frac{\partial^2 f}{\partial x_1\partial x_2}(x_1-x_1^*)(x_2-x_2^*)+\frac{\partial^2 f}{\partial x_2^2}(x_2-x_2^*)^2\right)=\frac{1}{2}\begin{bmatrix}\frac{\partial^2 f}{\partial x_1^2}(x_1-x_1^*)+\frac{\partial^2 f}{\partial x_2\partial x_1}(x_2-x_2^*) & \frac{\partial^2 f}{\partial x_1\partial x_2}(x_1-x_1^*)+\frac{\partial^2 f}{\partial x_2^2}(x_2-x_2^*)\end{bmatrix}\begin{bmatrix}x_1-x_1^*\\ x_2-x_2^*\end{bmatrix}$$

$$=\frac{1}{2}\begin{bmatrix}x_1-x_1^* & x_2-x_2^*\end{bmatrix}\begin{bmatrix}\frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1\partial x_2}\\ \frac{\partial^2 f}{\partial x_2\partial x_1} & \frac{\partial^2 f}{\partial x_2^2}\end{bmatrix}\begin{bmatrix}x_1-x_1^*\\ x_2-x_2^*\end{bmatrix}$$

$$=\underline{\frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T\mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*)}$$

$$f(\mathbf{x})=f(\mathbf{x}^*)+\underline{\nabla f(\mathbf{x}^*)^T(\mathbf{x}-\mathbf{x}^*)}+\underline{\frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T\mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*)}+R$$

2x2 Matrix의 원소

$$\left(\mathbf{x}=(x_1,x_2)^T,\ \mathbf{x}^*=(x_1^*,x_2^*)^T,\ \boxed{\mathbf{H}\in M_{2\times 2}}\right)$$

[참고] 2변수 함수의 테일러 전개)(2)

2변수 함수 $f(x_1, x_2)$ 에 대한 점 (x_1^*, x_2^*) 에서의 테일러 전개식(3차 이상의 고차항은 무시할 경우)

$$f(x_1,x_2) = f(x_1^*,x_2^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}(x_1 - x_1^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}(x_2 - x_2^*)$$

$$+\frac{1}{2}\left(\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1-x_1^*)^2 + 2\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_1-x_1^*)(x_2-x_2^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2-x_2^*)^2\right) \quad \cdots\cdots ①$$

$\mathbf{c} = \nabla f(\mathbf{x}^*)$, $\mathbf{d} = (\mathbf{x} - \mathbf{x}^*)$라 가정

$$\Rightarrow f(\mathbf{x}) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T(\mathbf{x}-\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T \mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) = f(\mathbf{x}^*) + \mathbf{c}^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\mathbf{H}(\mathbf{x}^*)\mathbf{d} \quad \cdots\cdots ②$$

식 ②를 x로 미분하면 어떤 형태의 Matrix로 표현 될까?

식①을 전개하면,

$$f(x_1,x_2) = f(x_1^*,x_2^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}x_1 - \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}x_1^* + \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}x_2 - \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}x_2^*$$

$$+\frac{1}{2}\left(\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1^2 - 2x_1x_1^* + x_1^{*2}) + 2\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}(x_1x_2 - x_1^*x_2 + x_1^*x_2^* - x_1x_2^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2^2 - 2x_2x_2^* + x_2^{*2})\right) \cdots\cdots ③$$

식③을 전개하면,

$$f(x_1,x_2) = \boxed{f(x_1^*,x_2^*)} + \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}x_1 - \boxed{\frac{\partial f(x_1^*,x_2^*)}{\partial x_1}x_1^*} + \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}x_2 - \boxed{\frac{\partial f(x_1^*,x_2^*)}{\partial x_2}x_2^*}$$

$$+\frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1^2 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1x_1^* + \boxed{\frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1^{*2}}$$

$$+\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1x_2 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1^*x_2 + \boxed{\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1^*x_2^*} - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1x_2^* \quad \cdots\cdots ④$$

$$+\frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2^2 - \frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2x_2^* + \boxed{\frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2^{*2}}$$

x^*는 상수 이므로
파란색 네모 안의 수는
모두 상수이다.

[참고] 2변수 함수의 테일러 전개)(3)

2변수 함수 $f(x_1, x_2)$ 에 대한 점 (x_1^*, x_2^*) 에서의 테일러 전개식(3차 이상의 고차항은 무시할 경우)

$$f(x_1,x_2) = f(x_1^*,x_2^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}(x_1 - x_1^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}(x_2 - x_2^*)$$

$$+\frac{1}{2}\left(\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1-x_1^*)^2 + 2\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_1-x_1^*)(x_2-x_2^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2-x_2^*)^2\right) \quad \cdots\cdots ①$$

$\mathbf{c} = \nabla f(\mathbf{x}^*),\ \mathbf{d} = (\mathbf{x}-\mathbf{x}^*)$라 가정

$$\Rightarrow f(\mathbf{x}) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T(\mathbf{x}-\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T \mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) = f(\mathbf{x}^*) + \mathbf{c}^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\mathbf{H}(\mathbf{x}^*)\,\mathbf{d} \quad \cdots\cdots ②$$

식 ②를 x로 미분하면 어떤 형태의 Matrix로 표현 될까?

$$f(x_1,x_2) = \boxed{f(x_1^*,x_2^*)} + \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}x_1 - \boxed{\frac{\partial f(x_1^*,x_2^*)}{\partial x_1}x_1^*} + \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}x_2 - \boxed{\frac{\partial f(x_1^*,x_2^*)}{\partial x_2}x_2^*} + \frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1^2 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1 x_1^* + \boxed{\frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1^{*2}}$$

$$+\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1x_2 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1^*x_2 + \boxed{\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1^*x_2^*} - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1x_2^* + \frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2^2 - \frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2x_2^* + \boxed{\frac{1}{2}\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2^{*2}} \quad \cdots ④$$

식④를 x_1과 x_2로 각각 미분(3차 이상의 고차항은 무시할 경우)

x*는 상수 이므로
파란색 네모 안의 수는
모두 상수이다.

$$\frac{\partial f(x_1,x_2)}{\partial x_1} = \frac{\partial f(x_1^*,x_2^*)}{\partial x_1} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}x_1^* + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_2 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_2^*$$

$$= \frac{\partial f(x_1^*,x_2^*)}{\partial x_1} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1-x_1^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}(x_2-x_2^*)$$

$$\frac{\partial f(x_1,x_2)}{\partial x_2} = \frac{\partial f(x_1^*,x_2^*)}{\partial x_2} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}x_2^* + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1 - \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}x_1^*$$

$$= \frac{\partial f(x_1^*,x_2^*)}{\partial x_2} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2-x_2^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1\partial x_2}(x_1-x_1^*)$$

[참고] 2변수 함수의 테일러 전개)(3)

2변수 함수 $f(x_1, x_2)$ 에 대한 점 (x_1^*, x_2^*) 에서의 테일러 전개식(3차 이상의 고차항은 무시할 경우)

$$f(x_1,x_2) = f(x_1^*,x_2^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}(x_1 - x_1^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}(x_2 - x_2^*)$$

$$+\frac{1}{2}\left(\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1 - x_1^*)^2 + 2\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_1 - x_1^*)(x_2 - x_2^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2 - x_2^*)^2\right) \quad \cdots\cdots ①$$

$\mathbf{c} = \nabla f(\mathbf{x}^*)$, $\mathbf{d} = (\mathbf{x} - \mathbf{x}^*)$라 가정

$$\Rightarrow f(\mathbf{x}) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) + \frac{1}{2}\left(\mathbf{x} - \mathbf{x}^*\right)^T \mathbf{H}(\mathbf{x}^*)\left(\mathbf{x} - \mathbf{x}^*\right) = f(\mathbf{x}^*) + \mathbf{c}^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\mathbf{H}(\mathbf{x}^*)\,\mathbf{d}$$

식 ②를 x로 미분하면 어떤 형태의 Matrix로 표현 될까?

$$\frac{\partial f(x_1,x_2)}{\partial x_1} = \frac{\partial f(x_1^*,x_2^*)}{\partial x_1} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1 - x_1^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_2 - x_2^*), \quad \frac{\partial f(x_1,x_2)}{\partial x_2} = \frac{\partial f(x_1^*,x_2^*)}{\partial x_2} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_1 - x_1^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2 - x_2^*)$$

$$\Rightarrow \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f(x_1,x_2)}{\partial x_1} \\ \frac{\partial f(x_1,x_2)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} \frac{\partial f(x_1^*,x_2^*)}{\partial x_1} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1 - x_1^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_2 - x_2^*) \\ \frac{\partial f(x_1^*,x_2^*)}{\partial x_2} + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_1 - x_1^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2 - x_2^*) \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial f(x_1^*,x_2^*)}{\partial x_1} \\ \frac{\partial f(x_1^*,x_2^*)}{\partial x_2} \end{bmatrix} + \begin{bmatrix} \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1 - x_1^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_2 - x_2^*) \\ \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_1 - x_1^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2 - x_2^*) \end{bmatrix} = \begin{bmatrix} \frac{\partial f(x_1^*,x_2^*)}{\partial x_1} \\ \frac{\partial f(x_1^*,x_2^*)}{\partial x_2} \end{bmatrix} + \begin{bmatrix} \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2} & \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2} & \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2} \end{bmatrix} \begin{bmatrix} (x_1 - x_1^*) \\ (x_2 - x_2^*) \end{bmatrix}$$

$$= \nabla f(\mathbf{x}^*) + \mathbf{H}(\mathbf{x}^*)\left(\mathbf{x} - \mathbf{x}^*\right) = \mathbf{c} + \mathbf{H}(\mathbf{x}^*)\,\mathbf{d}$$

$\mathbf{c} = \nabla f(\mathbf{x}^*)$, $\mathbf{d} = (\mathbf{x} - \mathbf{x}^*)$ 라 가정

[참고] 2변수 함수의 테일러 전개)(4)

2변수 함수 $f(x_1, x_2)$ 에 대한 점 (x_1^*, x_2^*) 에서의 테일러 전개식(3차 이상의 고차항은 무시할 경우)

$$f(x_1,x_2) = f(x_1^*,x_2^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_1}(x_1 - x_1^*) + \frac{\partial f(x_1^*,x_2^*)}{\partial x_2}(x_2 - x_2^*)$$

$$+\frac{1}{2}\left(\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1^2}(x_1-x_1^*)^2 + 2\frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_1 \partial x_2}(x_1-x_1^*)(x_2-x_2^*) + \frac{\partial^2 f(x_1^*,x_2^*)}{\partial x_2^2}(x_2-x_2^*)^2\right) \quad \cdots\cdots ①$$

$\mathbf{c} = \nabla f(\mathbf{x}^*),\ \mathbf{d} = (\mathbf{x}-\mathbf{x}^*)$라 가정

$$\Rightarrow f(\mathbf{x}) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T(\mathbf{x}-\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T\mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) = f(\mathbf{x}^*) + \mathbf{c}^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\mathbf{H}(\mathbf{x}^*)\,\mathbf{d} \quad \cdots\cdots ②$$

식 ②를 d로 미분하면 어떤 형태의 Matrix로 표현 될까?

식 ①을 d_1과 d_2로 각각 미분(3차 이상의 고차항은 무시할 경우)

Chain rule에 의해

$$\frac{\partial f(x_1,x_2)}{\partial d_1} = \frac{\partial f(x_1,x_2)}{\partial x_1}\frac{\partial x_1}{\partial d_1} = \frac{\partial f(x_1,x_2)}{\partial x_1} \qquad \frac{\partial f(x_1,x_2)}{\partial d_2} = \frac{\partial f(x_1,x_2)}{\partial x_2}\frac{\partial x_2}{\partial d_2} = \frac{\partial f(x_1,x_2)}{\partial x_2}$$

$\because d_1 = x_1 - x_1^*$ 양면을 d_1으로 미분 $1 = \frac{\partial x_1}{\partial d_1}$ $\qquad$ $\because d_2 = x_2 - x_2^*$ 양면을 d_2로 미분 $1 = \frac{\partial x_2}{\partial d_2}$

식 ②를 d로 미분한 결과는 x로 미분한 결과와 같다.

$$\frac{\partial f(\mathbf{d})}{\partial \mathbf{d}} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \nabla f(\mathbf{x}^*) + \mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) = \mathbf{c} + \mathbf{H}(\mathbf{x}^*)\,\mathbf{d}$$

$\mathbf{c} = \nabla f(\mathbf{x}^*),\ \mathbf{d} = (\mathbf{x}-\mathbf{x}^*)$ 라 가정

비제약 최적화 문제
- Newton 방법을 이용한 해법(1)

$$Minimize \ \ f(x_1,x_2)=x_1-x_2+2x_1^2+2x_1x_2+x_2^2$$

$$\nabla f(\mathbf{x})=\nabla f(x_1,x_2)=\begin{pmatrix} f_{x_1} \\ f_{x_2} \end{pmatrix}=\begin{pmatrix} 1+4x_1+2x_2 \\ -1+2x_1+2x_2 \end{pmatrix}$$

$$H(\mathbf{x})=\begin{pmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1 \partial x_2} \\ \dfrac{\partial^2 f}{\partial x_1 \partial x_2} & \dfrac{\partial^2 f}{\partial x_2^2} \end{pmatrix}=\begin{pmatrix} f_{x_1x_1} & f_{x_1x_2} \\ f_{x_1x_2} & f_{x_2x_2} \end{pmatrix}=\begin{pmatrix} 4 & 2 \\ 2 & 2 \end{pmatrix}$$

- **단계 1 -** $\mathbf{x}^{(1)}$구하기

이 방법은 출발점 $\mathbf{x}^{(k)}$에서 $\Delta\mathbf{x}^{(k)}$만큼 증가한점$\mathbf{x}^{(k+1)}=\mathbf{x}^{(k)}+\Delta\mathbf{x}^{(k)}$에서 함수 f가 최소값을갖는다고 가정한다.

여기서, $\Delta\mathbf{x}^{(0)}=\mathbf{x}^{(1)}-\mathbf{x}^{(0)}$

즉, $\Delta x_1^{(0)}=x_1^{(1)}-x_1^{(0)}$, $\Delta x_2^{(1)}=x_2^{(2)}-x_2^{(1)}$이다.

비제약 최적화 문제
- Newton 방법을 이용한 해법(2)

$$\begin{pmatrix} f_{x_1x_1} & f_{x_1x_2} \\ f_{x_1x_2} & f_{x_2x_2} \end{pmatrix}\begin{pmatrix} \Delta x_1^{(0)} \\ \Delta x_2^{(0)} \end{pmatrix} = -\begin{pmatrix} f_{x_1} \\ f_{x_2} \end{pmatrix}$$ 이고

$$f_{x_1} = 1 + 4x_1 + 2x_2$$

$$f_{x_2} = -1 + 2x_1 + 2x_2$$

$$f_{x_1x_1} = 4$$

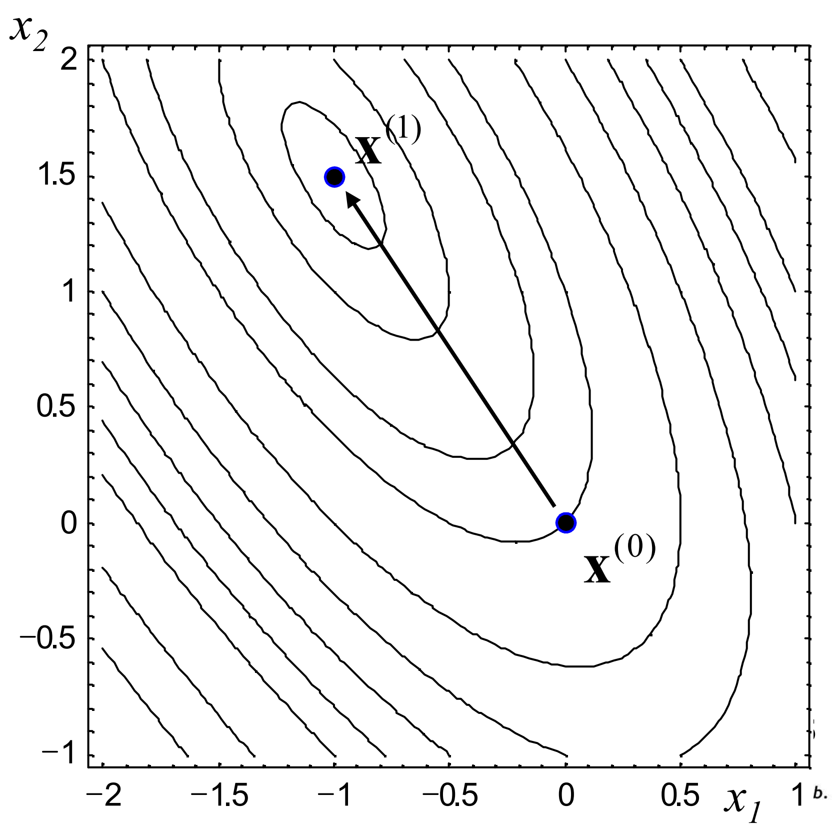
$$f_{x_2x_2} = 2$$

$$f_{x_1x_2} = f_{x_2x_1} = 2$$ 이므로

$$\begin{pmatrix} 4 & 2 \\ 2 & 2 \end{pmatrix}\begin{pmatrix} \Delta x_1^{(0)} \\ \Delta x_2^{(0)} \end{pmatrix} = -\begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

에서 $\Delta x_1^{(0)} = -1,\ \Delta x_2^{(0)} = 1.5$ 이다.

$$\therefore \mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \Delta\mathbf{x}^{(0)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 1.5 \end{pmatrix} = \begin{pmatrix} -1 \\ 1.5 \end{pmatrix}$$



비제약 최적화 문제
- Newton 방법을 이용한 해법(3)

- **단계 2 - $\mathbf{x}^{(2)}$구하기**

단계 1과 같은 과정을 거치면

$$\begin{pmatrix} f_{x_1x_1} & f_{x_1x_2} \\ f_{x_1x_2} & f_{x_2x_2} \end{pmatrix}\begin{pmatrix} \Delta x_1^{(1)} \\ \Delta x_2^{(1)} \end{pmatrix} = -\begin{pmatrix} f_{x_1} \\ f_{x_2} \end{pmatrix}$$

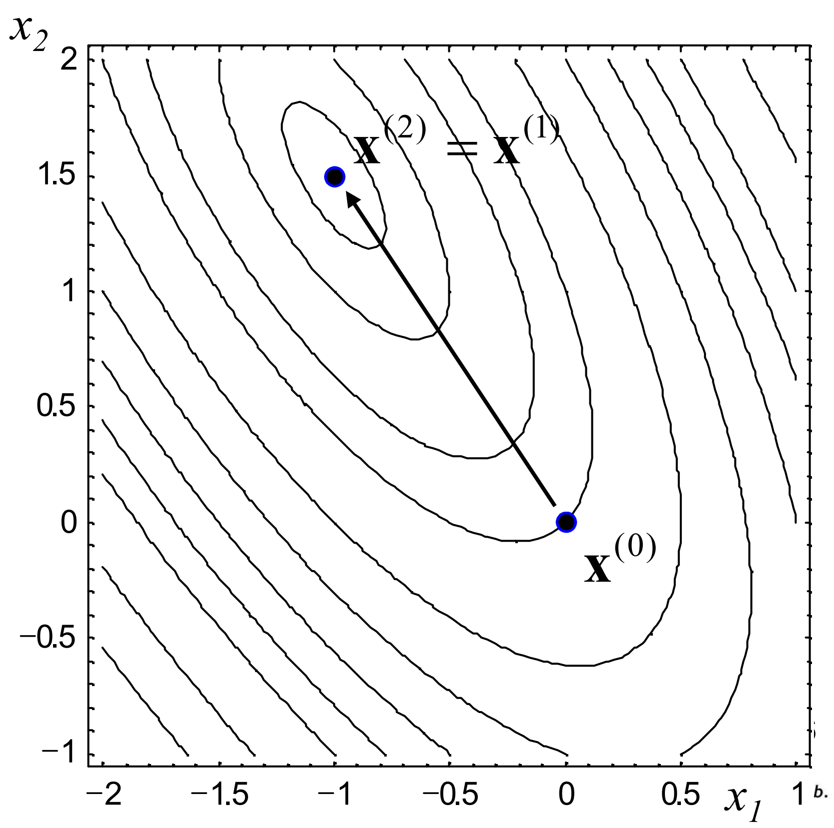
$$\begin{pmatrix} 4 & 2 \\ 2 & 2 \end{pmatrix}\begin{pmatrix} \Delta x_1^{(1)} \\ \Delta x_2^{(1)} \end{pmatrix} = -\begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

에서 $\Delta x_1^{(1)} = 0,\ \Delta x_2^{(1)} = 0$이다.

$$\therefore \mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \Delta\mathbf{x}^{(1)} = \begin{pmatrix} -1 \\ 1.5 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 1.5 \end{pmatrix}$$

따라서 $\left|\mathbf{x}^{(2)} - \mathbf{x}^{(1)}\right| \le \varepsilon = 0.001$이므로

$\mathbf{x}^{(2)}$이 최적해가 된다.



Gradient 방법
- Davidon-Fletcher-Powell(DFP) 방법(1/2)

- 1차 미분만을 이용하여 $f(\mathbf{x})$의 역 헷세 행렬을 근사적으로 계산하는 방법
  - 단계 1 : 시작점 $\mathbf{x}^{(0)}$를 추정한다.
    목적 함수의 역 헷세 행렬을 근사적으로 추정하기 위해 대칭의 양정 행렬 $\mathbf{A}^{(0)}$을 선정한다. 처음 시작할 때는 $\mathbf{A}^{(0)} = \mathbf{I}$ 를 선택할 수 있다. 수렴 매개 변수 ε을 정하고, $k = 0$ 이라 두고 Gradient 벡터를 다음과 같이 계산한다.
$$\mathbf{d}^{(0)} = \mathbf{c}^{(0)} \equiv \nabla f(\mathbf{x}^{(0)})$$
  - 단계 2 : Gradient 벡터의 Norm $\left\|\mathbf{c}^{(k)}\right\|$를 계산한다.
    만일 $\left\|\mathbf{c}^{(k)}\right\| < \varepsilon$ 이면 반복 과정을 멈추고 그렇지 않으면 다음 단계로 계속 진행한다. 이 방법의 첫 단계는 최속 강하법과 같다. 즉, 첫 단계에서는 단계 4로 간다.
  - 단계 3 : 탐색 방향을 다음과 같이 계산한다.
$$\boxed{\mathbf{d}^{(k)} = -\mathbf{A}^{(k)}\mathbf{c}^{(k)}}$$
    즉, Newton 방법의 $\mathbf{H}^{-1}$을 근사적으로 $\mathbf{A}$로 대체한다.

Gradient 방법
- Davidon-Fletcher-Powell(DFP) 방법(2/2)

- 단계 4 : $f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)})$를 최소화하는 이동 거리 $\alpha_k = \alpha$를 계산한다.
- 단계 5 : 설계점을 $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$로 수정한다.
- 단계 6 : 목적 함수의 근사적 역 헷세 행렬 $\mathbf{A}^{(k)}$를 다음과 같이 보정한다.

$$\mathbf{A}^{(k+1)} = \mathbf{A}^{(k)} + \mathbf{B}^{(k)} + \mathbf{C}^{(k)} \quad ; \; n \times n \text{ 행렬}$$

여기서, 보정 행렬 $\mathbf{B}^{(k)}$, $\mathbf{C}^{(k)}$는 다음과 같이 계산한다.

$$\mathbf{B}^{(k)} = \frac{\mathbf{s}^{(k)}(\mathbf{s}^{(k)})^T}{(\mathbf{s}^{(k)} \cdot \mathbf{y}^{(k)})} \quad ; \; n \times n \text{ 행렬} \qquad \mathbf{C}^{(k)} = \frac{-\mathbf{z}^{(k)}(\mathbf{z}^{(k)})^T}{(\mathbf{y}^{(k)} \cdot \mathbf{z}^{(k)})} \quad ; \; n \times n \text{ 행렬}$$

$$\mathbf{s}^{(k)} = \alpha_k \mathbf{d}^{(k)} \quad ; \; n \times 1 \text{ 벡터}$$

$$\mathbf{y}^{(k)} = \mathbf{c}^{(k+1)} - \mathbf{c}^{(k)} \quad ; \; n \times 1 \text{ 벡터}$$

$$\mathbf{c}^{(k+1)} = \nabla f(\mathbf{x}^{(k+1)}) \quad ; \; n \times 1 \text{ 벡터}$$

$$\mathbf{z}^{(k)} = \mathbf{A}^{(k)} \mathbf{y}^{(k)} \quad ; \; [n \times n \text{ 행렬}][n \times 1 \text{ 벡터}] = n \times 1 \text{ 행렬}$$

- 단계 7 : $k = k + 1$로 두고 단계 2를 수행한다.

비제약 최적화 문제
- Davidon-Fletcher-Powell(DFP) 방법을 이용한 해법(1)

$$f(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$

$$\nabla f(\mathbf{x}) = \nabla f(x_1, x_2) = \begin{pmatrix} 1 + 4x_1 + 2x_2 \\ -1 + 2x_1 + 2x_2 \end{pmatrix}$$

- **단계 1 - $\mathbf{x}^{(1)}$구하기**

$$\mathbf{x}^{(0)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \ \mathbf{A}^{(0)} = \mathbf{I}$$

$$\mathbf{c}^{(0)} = \begin{pmatrix} 1 + 4x_1 + 2x_2 \\ -1 + 2x_1 + 2x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

$$\left\| \mathbf{c}^{(0)} \right\| = \sqrt{1^2 + (-1)^2} > \varepsilon = 0.001$$

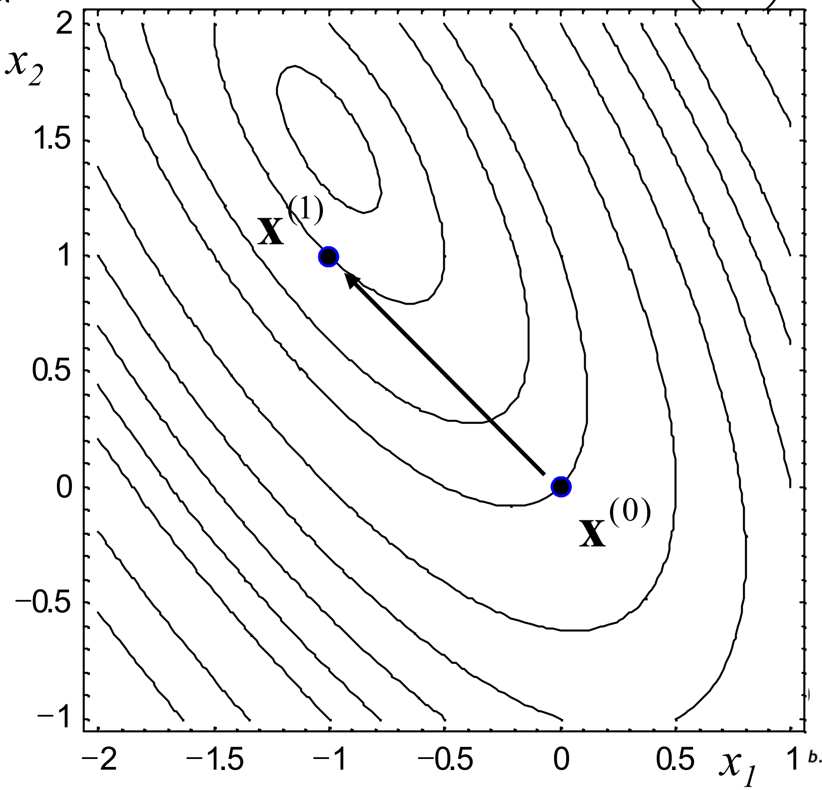
$$\mathbf{d}^{(0)} = -\mathbf{A}^{(0)}\mathbf{c}^{(0)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

$$\mathbf{x}^{(1)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \alpha \begin{pmatrix} -1 \\ 1 \end{pmatrix} = \begin{pmatrix} -\alpha \\ \alpha \end{pmatrix}$$

$$f(\mathbf{x}^{(1)}) = \alpha^2 - 2\alpha$$

함수 $f(\mathbf{x}^{(1)})$가 최소값을 가질 조건

$\dfrac{df(\mathbf{x}^{(1)})}{d\alpha} = 0$으로부터 $\alpha = 1 \quad \therefore \mathbf{x}^{(1)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$



비제약 최적화 문제
- Davidon-Fletcher-Powell(DFP) 방법을 이용한 해법(2)

$$\mathbf{s}^{(0)} = \alpha \mathbf{d}^{(0)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

$$\mathbf{c}^{(1)} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

$$\mathbf{y}^{(0)} = \mathbf{c}^{(1)} - \mathbf{c}^{(0)} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

$$\mathbf{z}^{(0)} = \mathbf{A}^{(0)}\mathbf{y}^{(0)} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

$$\mathbf{s}^{(0)} \cdot \mathbf{y}^{(0)} = 2$$

$$\mathbf{y}^{(0)} \cdot \mathbf{z}^{(0)} = 4$$

$$\mathbf{s}^{(0)}\mathbf{s}^{(0)^T} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

$$\mathbf{B}^{(0)} = \frac{\mathbf{s}^{(0)}\mathbf{s}^{(0)^T}}{\mathbf{s}^{(0)} \cdot \mathbf{y}^{(0)}} = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}$$

$$\mathbf{z}^{(0)}\mathbf{z}^{(0)^T} = \begin{pmatrix} 4 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{C}^{(0)} = \frac{-\mathbf{z}^{(0)}\mathbf{z}^{(0)^T}}{\mathbf{y}^{(0)} \cdot \mathbf{z}^{(0)}} = \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{A}^{(0)} = \mathbf{A}^{(0)} + \mathbf{B}^{(0)} + \mathbf{C}^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 1.5 \end{pmatrix}$$

비제약 최적화 문제
- Davidon-Fletcher-Powell(DFP) 방법을 이용한 해법(3)

- **단계 2 - $\mathbf{x}^{(2)}$구하기**

$$\|\mathbf{c}^{(1)}\| = \sqrt{2} > \varepsilon$$

$$\mathbf{d}^{(1)} = -\mathbf{A}^{(1)}\mathbf{c}^{(1)} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\mathbf{x}^{(2)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix} + \alpha \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 1+\alpha \end{pmatrix}$$

$$f(\mathbf{x}^{(2)}) = \alpha^2 - \alpha - 1$$

함수 $f(\mathbf{x}^{(2)})$가 최소값을 가질 조건

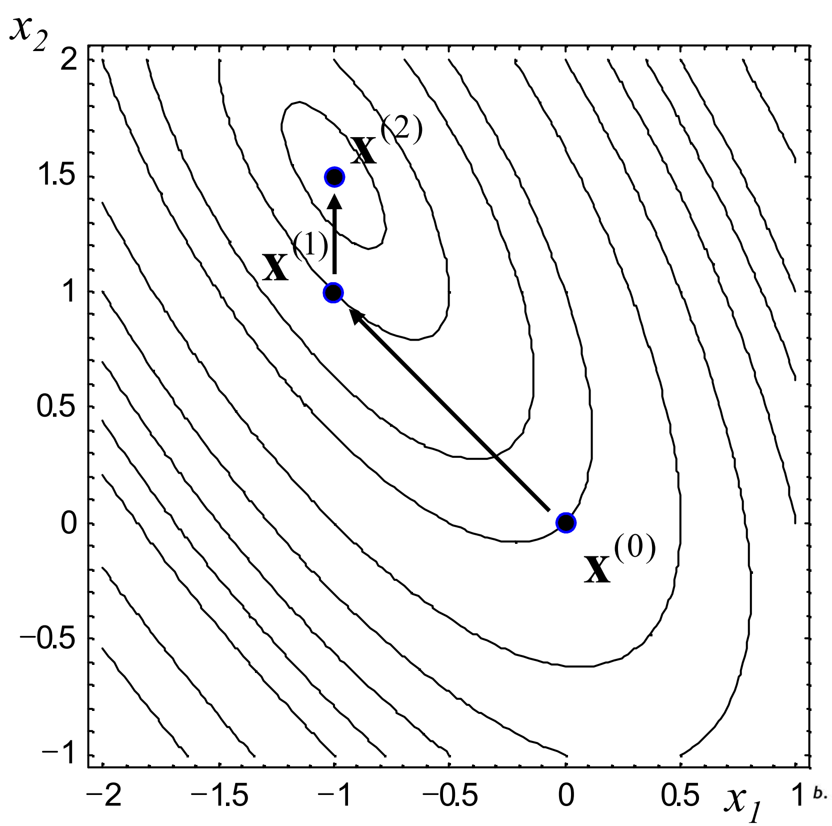
$\dfrac{df(\mathbf{x}^{(2)})}{d\alpha} = 0$으로부터 $\alpha = 0.5$

$$\therefore \mathbf{x}^{(2)} = \begin{pmatrix} -1 \\ 1.5 \end{pmatrix}$$

$$\mathbf{s}^{(1)} = \alpha\mathbf{d}^{(1)} = \begin{pmatrix} 0 \\ 0.5 \end{pmatrix} \qquad \mathbf{c}^{(2)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$\|\mathbf{c}^{(2)}\| = 0 \le \varepsilon$이므로

$\mathbf{x}^{(2)}$가 최적점이 된다.



Gradient 방법
- Broyden-Fletcher-Goldfarb-Shanno(BFGS) 방법(1/2)

- DFP 방법은 매 반복 단계에서 역 헷세 행렬을 근사적으로 계산하는데 비해, BFGS 방법은 헷세 행렬을 수정하는 방법
  - 단계 1 : 시작점 $\mathbf{x}^{(0)}$을 추정한다.
    목적 함수의 헷세 행렬을 근사적으로 추정하기 위하여 대칭의 양정 행렬 $\mathbf{H}^{(0)}$를 선정한다. 처음 시작할 때는 $\mathbf{H}^{(0)}=\mathbf{I}$를 선택할 수 있다.
    수렴 매개 변수 ε을 정하고, $k=0$이라 두고 Gradient 벡터를 계산한다.

$$\mathbf{c}^{(0)}=\nabla f(\mathbf{x}^{(0)})$$

  - 단계 2 : Gradient 벡터의 Norm $\left\|\mathbf{c}^{(k)}\right\|$를 계산한다.
    $\left\|\mathbf{c}^{(k)}\right\|<\varepsilon$이면 반복 과정을 멈추고 아니면 다음 단계를 수행한다.
  - 단계 3 : 탐색 방향 $\mathbf{d}^{(k)}$를 구하기 위해 다음의 선형 방정식을 푼다.

$$\boxed{\mathbf{H}^{(k)}\mathbf{d}^{(k)}=-\mathbf{c}^{(k)}}$$

수학적으로는 Newton 방법의 $\mathbf{d}^{(k)}=-\mathbf{H}^{-1(k)}\mathbf{c}^{(k)}$ 와 같으나 실제로는 $\mathbf{H}$를 근사적으로 보정한다.

Gradient 방법
- Broyden-Fletcher-Goldfarb-Shanno(BFGS) 방법(2/2)

- 단계 4 : $f(\mathbf{x}^{(k)}+\alpha\mathbf{d}^{(k)})$를 최소화 하는 이동 거리 $\alpha_k=\alpha$ 를 계산한다.
- 단계 5 : 설계점을 $\mathbf{x}^{(k+1)}=\mathbf{x}^{(k)}+\alpha_k\mathbf{d}^{(k)}$ 로 수정한다.
- 단계 6 : 목적 함수의 근사적 헷세 행렬 $\mathbf{H}^{(k)}$를 다음과 같이 보정한다.

$$\mathbf{H}^{(k+1)}=\mathbf{H}^{(k)}+\mathbf{D}^{(k)}+\mathbf{E}^{(k)} \quad ; \; n\times n \text{ 행렬}$$

여기서, 보정 행렬 $D^{(k)}$ 와 $E^{(k)}$는 다음과 같이 계산한다.

$$\mathbf{D}^{(k)}=\frac{\mathbf{y}^{(k)}(\mathbf{y}^{(k)})^T}{(\mathbf{y}^{(k)}\cdot\mathbf{s}^{(k)})} \quad \text{; 설계점의 변화}$$

$$\mathbf{E}^{(k)}=\frac{\mathbf{c}^{(k)}(\mathbf{c}^{(k)})^T}{(\mathbf{c}^{(k)}\cdot\mathbf{d}^{(k)})} \quad \text{; 경사도의 변화}$$

$$\mathbf{s}^{(k)}=\alpha_k\mathbf{d}^{(k)}$$

$$\mathbf{y}^{(k)}=\mathbf{c}^{(k+1)}-\mathbf{c}^{(k)}$$

$$\mathbf{c}^{(k+1)}=\nabla f(\mathbf{x}^{(k+1)})$$

- 단계 7 : $k=k+1$ 로 두고 단계 2를 수행한다.

비제약 최적화 문제
- Broydon-Fletcher-Goldfarb-Shanno(BFGS) 방법을 이용한 해법(1)

$$f(x_1,x_2)=x_1-x_2+2x_1^2+2x_1x_2+x_2^2$$

$$\nabla f(\mathbf{x})=\nabla f(x_1,x_2)=\begin{pmatrix}1+4x_1+2x_2\\-1+2x_1+2x_2\end{pmatrix}$$

- **단계 1 - $\mathbf{x}^{(1)}$구하기**

$$\mathbf{x}^{(0)}=\begin{pmatrix}0\\0\end{pmatrix},\ \mathbf{H}^{(0)}=\mathbf{I}$$

$$\mathbf{c}^{(0)}=\begin{pmatrix}1+4x_1+2x_2\\-1+2x_1+2x_2\end{pmatrix}=\begin{pmatrix}1\\-1\end{pmatrix}$$

$$\left\|\mathbf{c}^{(0)}\right\|=\sqrt{1^2+(-1)^2}>\varepsilon=0.001$$

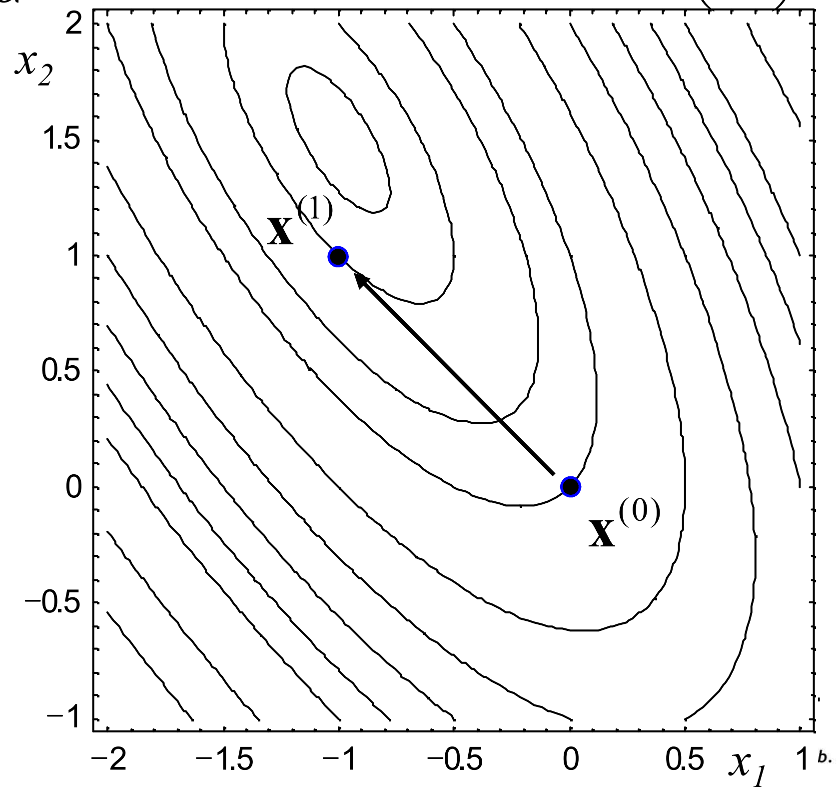
$$\mathbf{d}^{(0)}=-\mathbf{H}^{-1(0)}\mathbf{c}^{(0)}=\begin{pmatrix}-1\\1\end{pmatrix}$$

$$\mathbf{x}^{(1)}=\begin{pmatrix}0\\0\end{pmatrix}+\alpha\begin{pmatrix}-1\\1\end{pmatrix}=\begin{pmatrix}-\alpha\\\alpha\end{pmatrix}$$

$$f(\mathbf{x}^{(1)})=\alpha^2-2\alpha$$

함수 $f(\mathbf{x}^{(1)})$가 최소값을 가질 조건

$$\frac{df(\mathbf{x}^{(1)})}{d\alpha}=0\text{으로부터 } \alpha=1 \quad \therefore \mathbf{x}^{(1)}=\begin{pmatrix}-1\\1\end{pmatrix}$$



비제약 최적화 문제

- Broydon-Fletcher-Goldfarb-Shanno(BFGS) 방법을 이용한 해법(2)

$$\mathbf{s}^{(0)} = \alpha\mathbf{d}^{(0)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

$$\mathbf{c}^{(1)} = \begin{pmatrix} -1 \\ -1 \end{pmatrix} \qquad \nabla f(\mathbf{x}^{(1)}) = \mathbf{c}^{(1)}$$

$$\mathbf{y}^{(0)} = \mathbf{c}^{(1)} - \mathbf{c}^{(0)} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

$$\mathbf{y}^{(0)} \cdot \mathbf{s}^{(0)} = (-2,0)\cdot(-1,1) = 2$$

$$\mathbf{c}^{(0)} \cdot \mathbf{d}^{(0)} = (1,-1)\cdot(-1,1) = -2$$

$$\mathbf{y}^{(0)}\mathbf{y}^{(0)^T} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}\begin{pmatrix} -2 & 0 \end{pmatrix} = \begin{pmatrix} 4 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{D}^{(0)} = \frac{\mathbf{y}^{(0)}\mathbf{y}^{(0)^T}}{\mathbf{y}^{(0)} \cdot \mathbf{s}^{(0)}} = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{c}^{(0)}\mathbf{c}^{(0)^T} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}\begin{pmatrix} 1 & -1 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

$$\mathbf{E}^{(0)} = \frac{\mathbf{c}^{(0)}\mathbf{c}^{(0)^T}}{\mathbf{c}^{(0)} \cdot \mathbf{d}^{(0)}} = \begin{pmatrix} -0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix}$$

$$\mathbf{H}^{(1)} = \mathbf{H}^{(0)} + \mathbf{D}^{(0)} + \mathbf{E}^{(0)} = \begin{pmatrix} 2.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}$$

비제약 최적화 문제
- Broydon-Fletcher-Goldfarb-Shanno(BFGS) 방법을 이용한 해법(3)

- **단계 2 - $\mathbf{x}^{(2)}$구하기**

$$\left\|\mathbf{c}^{(1)}\right\| = \sqrt{2} > \varepsilon$$

$\mathbf{H}^{(1)}\mathbf{d}^{(1)} = -\mathbf{c}^{(1)}$에서 $\mathbf{d}^{(1)} = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$

$$\mathbf{x}^{(2)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix} + \alpha \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} -1 \\ 1+2\alpha \end{pmatrix}$$

$$f(\mathbf{x}^{(2)}) = 4\alpha^2 - 2\alpha - 1$$

함수 $f(\mathbf{x}^{(2)})$가 최소값을 가질 조건

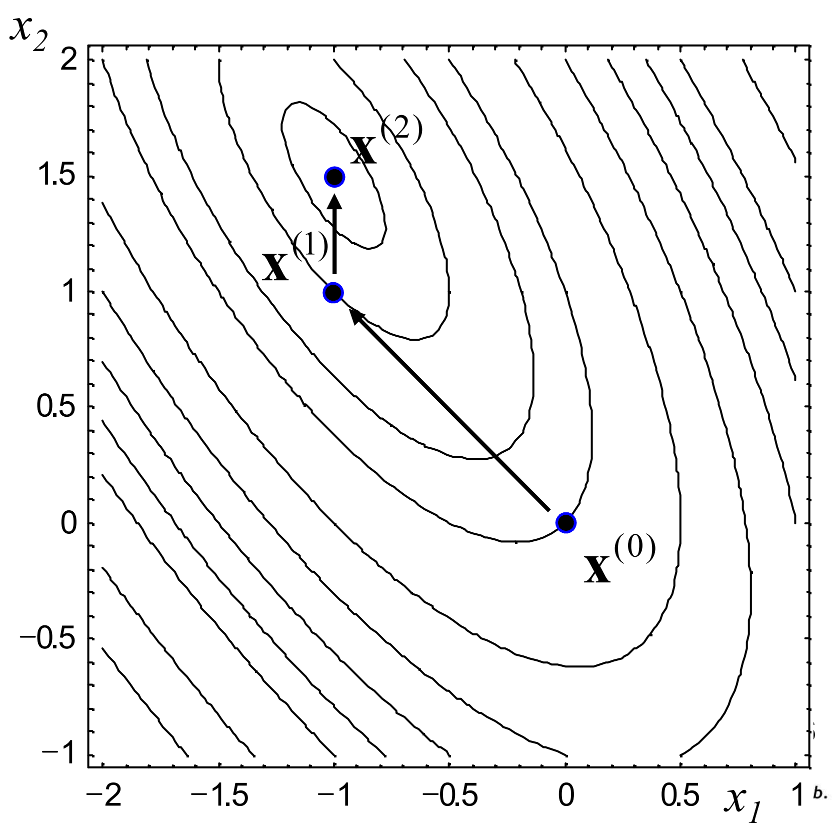
$\dfrac{df(\mathbf{x}^{(2)})}{d\alpha} = 0$으로부터 $\alpha = 0.25$

$$\therefore \mathbf{x}^{(2)} = \begin{pmatrix} -1 \\ 1.5 \end{pmatrix}$$

$$\mathbf{s}^{(1)} = \alpha \mathbf{d}^{(1)} = \begin{pmatrix} 0 \\ 0.5 \end{pmatrix} \qquad \mathbf{c}^{(2)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$\left\|\mathbf{c}^{(2)}\right\| = 0 \le \varepsilon$이므로

$\mathbf{x}^{(2)}$가 최적점이 된다.

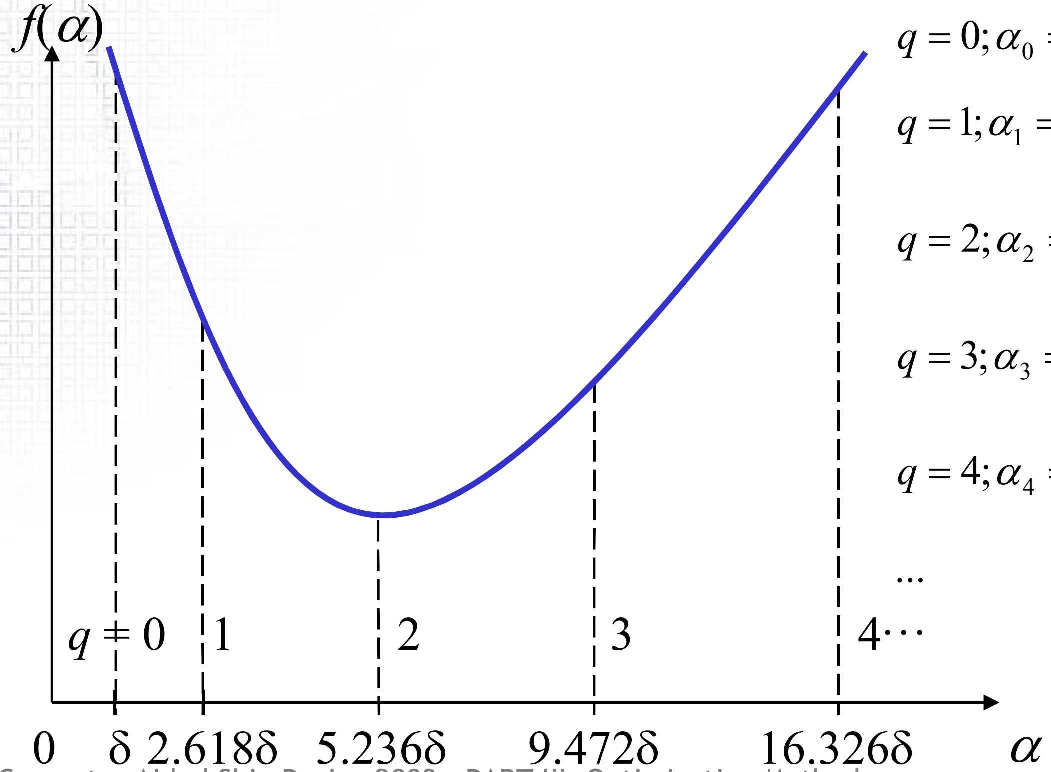


2.2 황금 분할에 의한 1차원 탐색 방법("황금 분할법")

Advanced
Ship
Design
Automation
Laboratory

2.2 황금 분할에 의한 1차원 탐색 방법
- 최소값이 위치하는 구간 탐색 (1/2)

- 최소값이 위치하는 구간 탐색
  - 황금 분할법은 최소값이 위치하는 초기 구간을 알고 시작해야 한다.
  - $\alpha = 0$과 $\alpha = \delta$에서의 함수값 $f(0)$과 $f(\delta)$를 계산하여 $f(\delta)$이 $f(0)$보다 작으면 이동량을 1.618δ의 증가치를 택한다.
    즉, 증분이 이전 증가의 1.618배이다. (Fibonacci sequence)



$$q = 0; \alpha_0 = \delta$$

$$q = 1; \alpha_1 = \delta + 1.618\delta = 2.618\delta = \sum_{j=0}^{1} \delta(1.618)^j$$

$$q = 2; \alpha_2 = 2.618\delta + 1.618(1.618\delta) = 5.236\delta = \sum_{j=0}^{2} \delta(1.618)^j$$

$$q = 3; \alpha_3 = 5.236\delta + (1.618)^3\delta = 9.472\delta = \sum_{j=0}^{3} \delta(1.618)^j$$

$$q = 4; \alpha_4 = 9.472\delta + (1.618)^4\delta = 16.326\delta = \sum_{j=0}^{4} \delta(1.618)^j$$

$$\cdots$$

$$\boxed{\therefore \alpha_q = \sum_{j=0}^{q} \delta(1.618)^j, \ q = 0, 1, 2, \ldots}$$

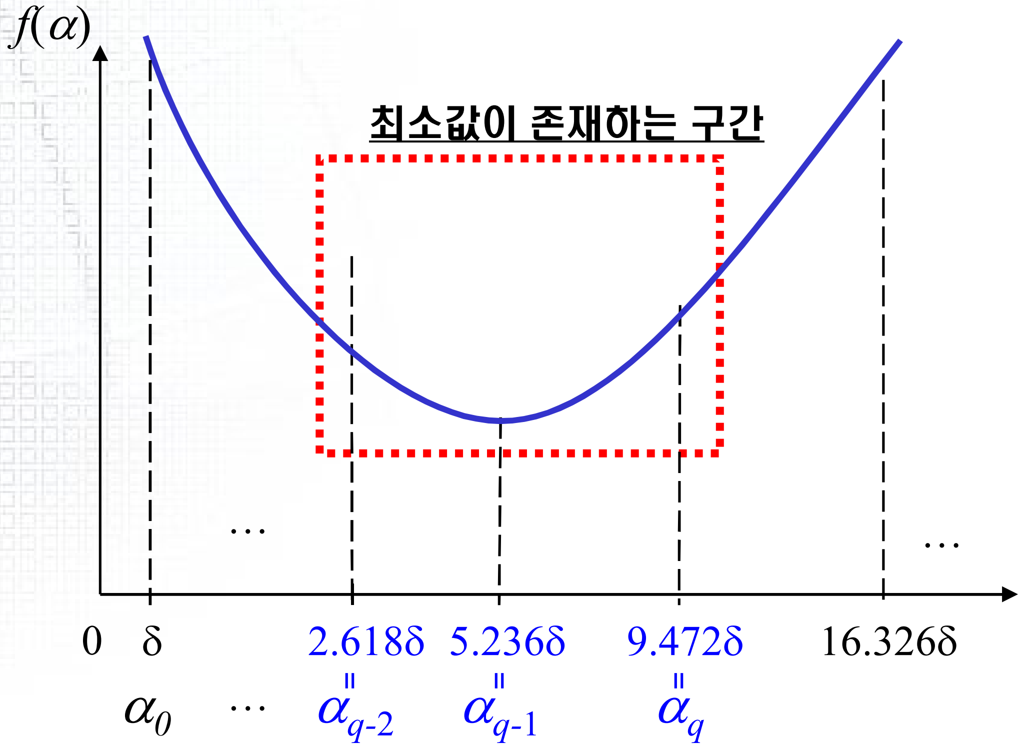
2.2 황금 분할에 의한 1차원 탐색 방법

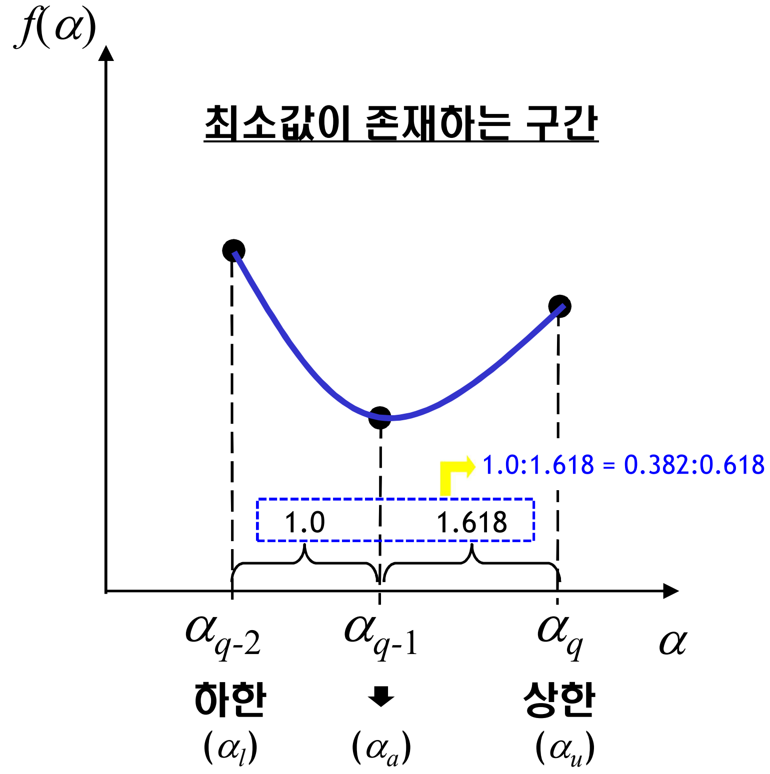
- 최소값이 위치하는 구간 탐색 (1/2)

- α_{q-1} 에서의 함수값이 α_{q-2} 와 α_q 에서의 값보다 작다면,

$$f(\alpha_{q-1}) < f(\alpha_{q-2}), \quad f(\alpha_{q-1}) < f(\alpha_q)$$

최소점은 두 구간, 즉 α_q 와 α_{q-2} 사이에 있다.





- 최소점이 존재하는 구간의 상한과 하한은

$$\underset{\text{상한}}{\alpha_u} \equiv \alpha_q = \sum_{j=0}^{q} \delta(1.618)^j, \quad \underset{\text{하한}}{\alpha_l} \equiv \alpha_{q-2} = \sum_{j=0}^{q-2} \delta(1.618)^j, \quad \alpha_a \equiv \alpha_{q-1} = \sum_{j=0}^{q-1} \delta(1.618)^j$$

2.2 황금 분할에 의한 1차원 탐색 방법

- Fibonacci sequence

Fibonacci sequence

정의: $F_0 = 0; \quad F_1 = 1; \qquad F_n = F_{n-1} + F_{n-2}, \quad n = 2,\ 3,\ \cdots$

➔ 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, …

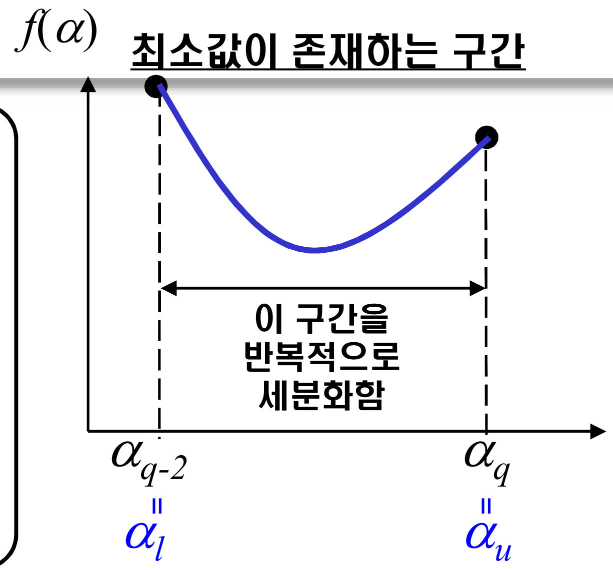
일반항: $F_n = \dfrac{\varphi^n - (1-\varphi)^n}{\sqrt{5}}, \quad \varphi = \dfrac{1+\sqrt{5}}{2} \approx 1.6180339887\cdots$

특성: $\lim\limits_{n\to\infty} \dfrac{F_n}{F_{n-1}} = \varphi, \quad 1-\varphi = -\dfrac{1}{\varphi}$

$$\left(\because \frac{1-\varphi}{\varphi} < 1 \right)$$

$$\lim_{n\to\infty} \frac{F_n}{F_{n-1}} = \lim_{n\to\infty} \frac{\varphi^n - (1-\varphi)^n}{\varphi^{n-1} - (1-\varphi)^{n-1}} = \lim_{n\to\infty} \frac{\dfrac{\varphi^n - (1-\varphi)^n}{\varphi^{n-1}}}{\dfrac{\varphi^{n-1} - (1-\varphi)^{n-1}}{\varphi^{n-1}}} = \lim_{n\to\infty} \frac{\varphi - (1-\varphi)\cancel{\left(\dfrac{1-\varphi}{\varphi}\right)^{n-1}}}{1 - \cancel{\left(\dfrac{1-\varphi}{\varphi}\right)^{n-1}}} = \varphi$$

2.2 황금 분할에 의한 1차원 탐색 방법
- 최소값이 위치하는 구간의 세분화



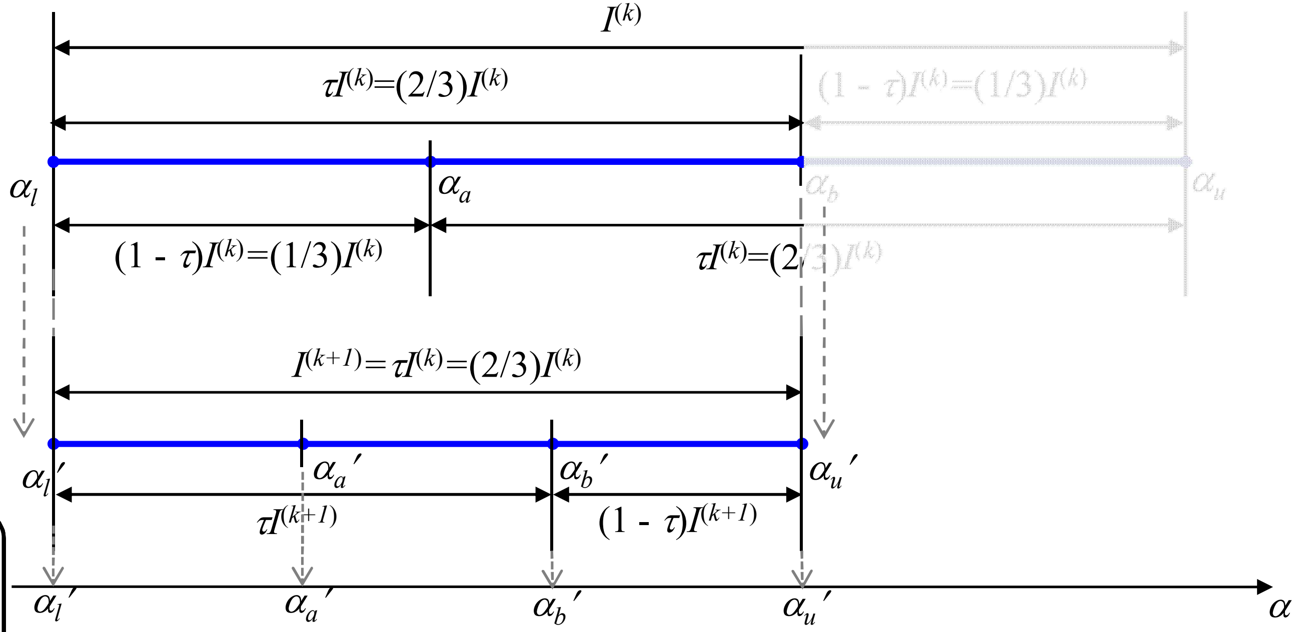
- **최소값이 위치하는 구간의 세분화**
  - 최소값이 위치하는 구간 $I^{(k)}$를 τ : 1-τ 로 내분하는 점을 양끝으로부터 각각 구한다.
  - 구한 점들에서의 함수값을 비교하여 구간을 세분화한다.
    - 아래 그림과 같이 양끝 점에서 대칭으로 위치한 두 점으로부터 같은 간격 $\tau I^{(k)}$만큼 떨어진 점 α_a 와 α_b 를 잡는다.
    - 그 점에서의 함수값을 계산하여 큰 쪽의 구간을 버리고, 남은 구간을 새로운 구간으로 정한다.

<τ = 2/3일 경우의 예>

(a)

$f(\alpha_a) < f(\alpha_b)$라고 가정하면
최소값이 존재하는 구간은
다음과 같이 변경됨

(b)

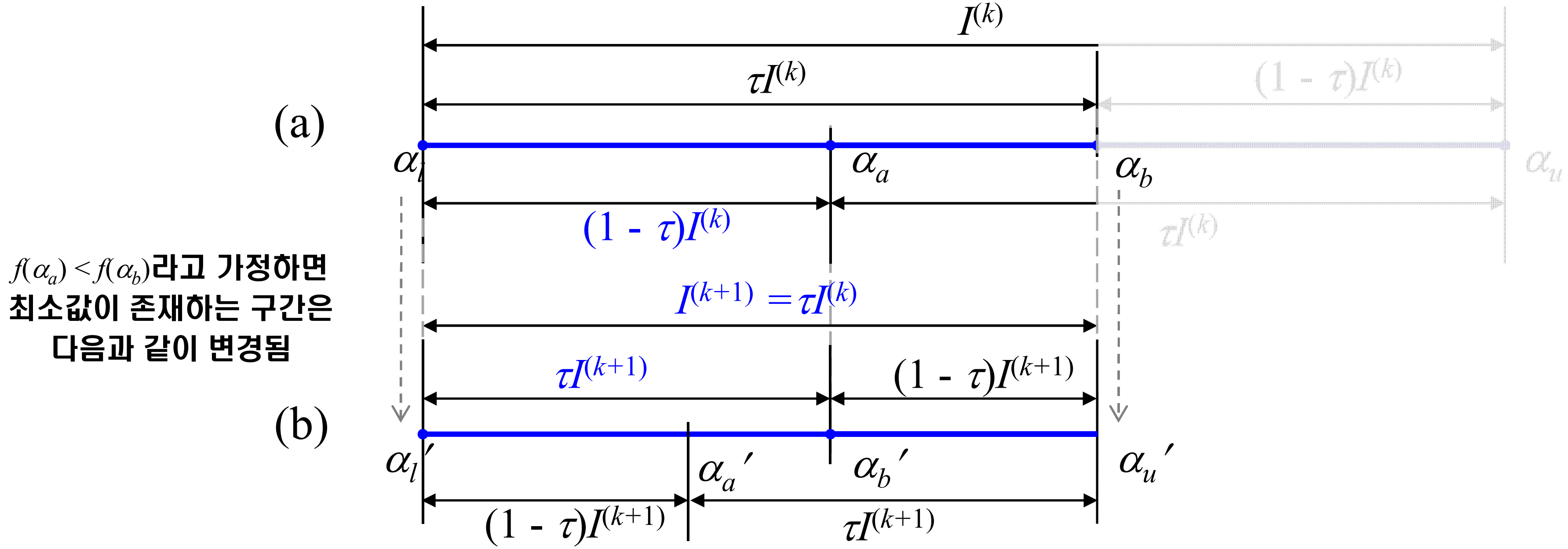


최소값이 존재하는 구간이 변경될 때마다
$f(\alpha_a'), f(\alpha_b')$에서의 함수값을 새로 구해야 함
<문제 제기>
이전 구간에서의 함수값을 활용할 수 없을까?

2.2 황금 분할에 의한 1차원 탐색 방법

- 최소값이 위치하는 구간의 세분화

- **최소값이 위치하는 구간의 세분화**
  - 최소값이 위치하는 구간 $I^{(k)}$를 τ : 1-τ 로 내분하는 점을 양끝으로부터 각각 구한다.



1. $f(\alpha_a)$를 다음 구간 $I^{(k+1)}$에서 사용하려 함
2. α_a가 $I^{(k+1)}$ 구간의 α_a' 혹은 α_b'와 일치 하도록 τ를 결정

3-1. α_a가 α_a'와 일치한다고 가정

$$\alpha_a = \alpha_a'$$
$$(1-\tau)I^{(k)} = (1-\tau)I^{(k+1)}$$
$$(1-\tau)I^{(k)} = (1-\tau)\tau I^{(k)}$$
$$I^{(k)} = \tau I^{(k)}$$

τ =1이므로 우리가 원하는 값이 아님

3-2. α_a가 α_b'와 일치한다고 가정

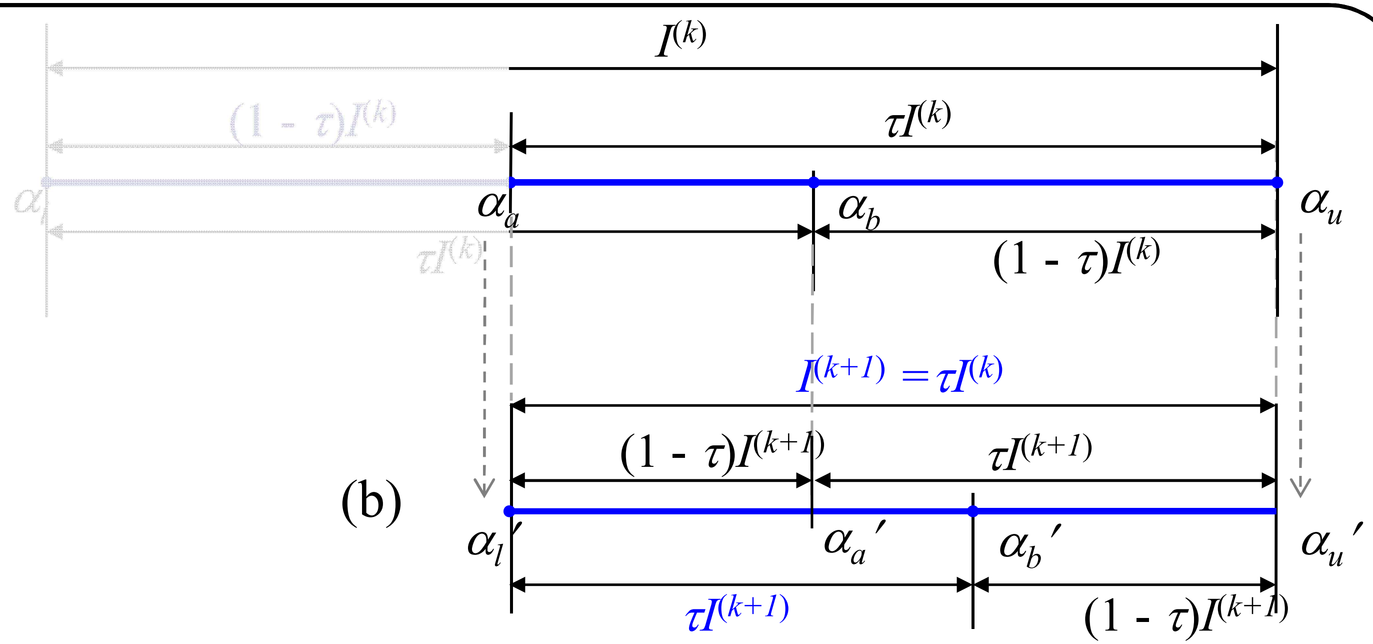
$$\alpha_a = \alpha_b'$$
$$(1-\tau)I^{(k)} = \tau I^{(k+1)}$$
$$(1-\tau)I^{(k)} = \tau \cdot \tau \cdot I^{(k)}$$
$$\tau \cdot \tau I^{(k)} - (1-\tau)I^{(k)} = 0$$
$$\tau^2 + \tau - 1 = 0$$

$\Rightarrow \tau = 0.618, -1.618$ $\Rightarrow$ 0.618

2.2 황금 분할에 의한 1차원 탐색 방법
- 최소값이 위치하는 구간의 세분화

- **최소값이 위치하는 구간의 세분화**
  - 최소값이 위치하는 구간 $I^{(k)}$를 τ : 1-τ 로 내분하는 점을 양끝으로부터 각각 구한다.

(a)



$f(\alpha_a) > f(\alpha_b)$라고 가정하면
최소값이 존재하는 구간은
다음과 같이 변경됨

1. $f(\alpha_b)$를 다음 구간 $I^{(k+1)}$에서 사용하려 함
2. α_b 가 $I^{(k+1)}$ 구간의 α_a' 혹은 α_b'와 일치 하도록 τ를 결정

3-1. α_b 가 α_b'와 일치한다고 가정

$$\alpha_b = \alpha_b'$$
$$(1-\tau)I^{(k)} = (1-\tau)I^{(k+1)}$$
$$(1-\tau)I^{(k)} = (1-\tau)\tau I^{(k)}$$
$$I^{(k)} = \tau I^{(k)}$$

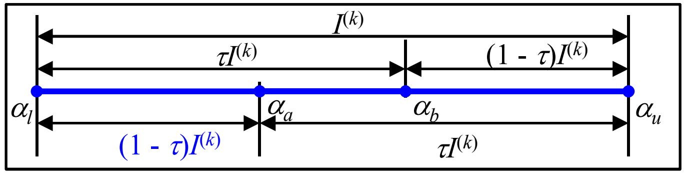
τ =1이므로 우리가 원하는 값이 아님

3-2. α_b 가 α_a'와 일치한다고 가정

$$\alpha_b = \alpha_a'$$
$$(1-\tau)I^{(k)} = \tau I^{(k+1)}$$
$$(1-\tau)I^{(k)} = \tau \cdot \tau \cdot I^{(k)}$$
$$\tau \cdot \tau I^{(k)} - (1-\tau)I^{(k)} = 0$$
$$\tau^2 + \tau - 1 = 0$$

⇨ $\tau = 0.618, -1.618$ ⇨ 0.618

2.2 황금 분할에 의한 1차원 탐색 방법
- 황금분할법 알고리즘 요약 (1/2)



- 단계 1 : 최소점이 존재하는 구간 탐색
  α 에서 미소의 이동량 δ를 선정하고 $f(\alpha_{q-1}) < f(\alpha_{q-2}), f(\alpha_{q-1}) < f(\alpha_q)$ 을 만족하는 q를 정하고, 다음의 식으로 상한과 하한을 결정한다.

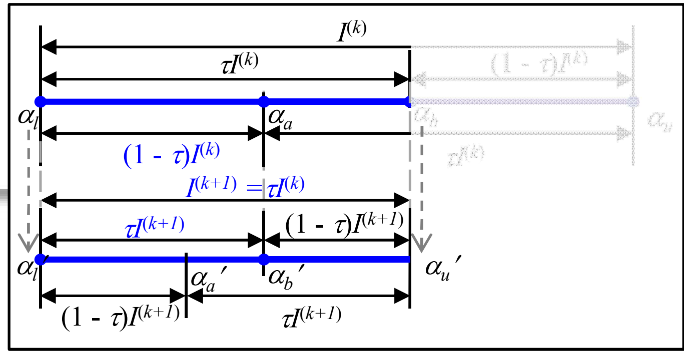
$$\alpha_u \equiv \alpha_q = \sum_{j=0}^{q} \delta(1.618)^j, \alpha_l \equiv \alpha_{q-2} = \sum_{j=0}^{q-2} \delta(1.618)^j$$

  여기서, 간격 $I = \alpha_u - \alpha_l$ 을 정한다.

- 단계 2 : $f(\alpha_a)$ 와 $f(\alpha_b)$를 계산한다. 이 때, $\alpha_a = \alpha_l + 0.382I$ 이고 $\alpha_b = \alpha_l + 0.618I$이다.

- 단계 3 : $f(\alpha_a)$ 와 $f(\alpha_b)$를 비교하여 그 결과에 따라 단계 4, 5, 6으로 간다.

2.2 황금 분할에 의한 1차원 탐색 방법
- 황금분할법 알고리즘 요약 (2/2)



- 단계 4 : $f(\alpha_a) < f(\alpha_b)$ 이면, 최적점 α^* 는 α_l과 α_b 사이에 있다.
세분화된 구간의 새로운 한계는 $\alpha_l' = \alpha_l$ 이고 $\alpha_u' = \alpha_b$ 이다.
또한 $\alpha_b' = \alpha_a$ 이다. $\alpha_a' = \alpha_l' + 0.382(\alpha_u' - \alpha_l')$ 에서 $f(\alpha_a')$ 를 계산하고 단계 7로 간다.

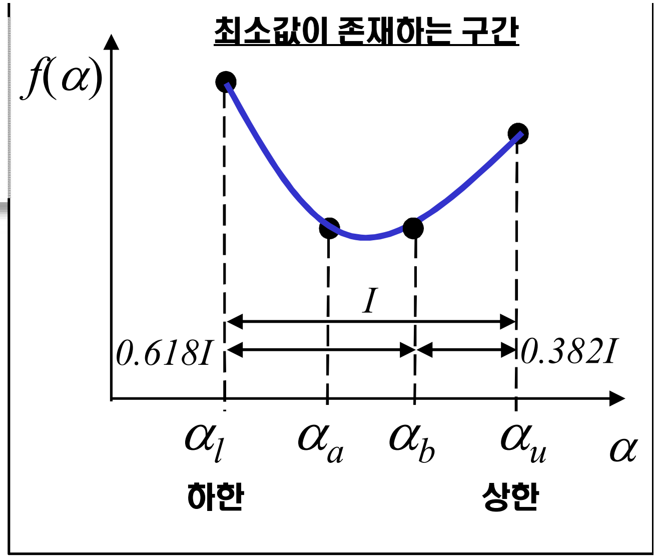
- 단계 5 : $f(\alpha_a) > f(\alpha_b)$ 이면, 최적점 α^* 는 α_a과 α_u 사이에 있다.
세분화된 구간의 새로운 한계는 $\alpha_l' = \alpha_a$ 이고 $\alpha_u' = \alpha_u$ 이다.
또한 $\alpha_a' = \alpha_b$ 이다. $\alpha_b' = \alpha_l' + 0.618(\alpha_u' - \alpha_l')$ 에서 $f(\alpha_b')$ 를 계산하고 단계 7로 간다.

- 단계 6 : $f(\alpha_a) = f(\alpha_b)$ 이면, $\alpha_l = \alpha_a, \alpha_u = \alpha_b$ 라 두고 단계 7로 간다.

- 단계 7 : 만일 새로운 세분화된 구간 $(I' = \alpha_u' - \alpha_l')$ 이 충분히 작아서 수렴 기준을 만족하면(즉, $I' < \varepsilon$), $\alpha^* = (\alpha_u' - \alpha_l')/2$ 라 두고 최적화 과정을 마친다. 그렇지 않으면, α_l', α_a', α_b', α_u' 의 Prime(') 부호를 삭제하고 단계 3으로 돌아간다.

@SDAL Advanced Ship Design Automation Lab. http://asdal.snu.ac.kr

황금 분할법 Programming Guide (2)



3단계 : $f(\alpha_a)$ 과 $f(\alpha_b)$ 를 비교

① $f(\alpha_a) < f(\alpha_b)$

최적점 α^* 는 α_l과 α_b 사이에 있다.

세분화된 구간의 새로운 한계는 $\alpha_b' = \alpha_a$이고 $\alpha_l' = \alpha_l$ 이다. 또한 $\alpha_u' = \alpha_b$ 이다.

$\alpha_a' = \alpha_l' + 0.382(\alpha_u' - \alpha_l')$ 에서 $f(\alpha_a')$ 를 계산하고 단계 4로 간다.

② $f(\alpha_a) > f(\alpha_b)$

최적점 α^* 는 α_a과 α_u 사이에 있다.

세분화된 구간의 새로운 한계는 $\alpha_l' = \alpha_a$이고 $\alpha_a' = \alpha_b$이다. 또한 $\alpha_u' = \alpha_u$ 이다.

$\alpha_b' = \alpha_l' + 0.618(\alpha_u' - \alpha_l')$ 에서 $f(\alpha_b')$ 를 계산하고 단계 4로 간다.

③ $f(\alpha_a) = f(\alpha_b)$

$\alpha_l = \alpha_a, \alpha_u = \alpha_b$ 라 두고 단계 4로 간다.

4단계 : 구간 $(I' = \alpha_u' - \alpha_l')$ 의 길이가 Tolerance보다 작으면 종료한다. 아닐 경우 2단계로 돌아간다.

황금 분할법 Programming Guide (3)

```
// [Input]
//    x[0]: 최소값이 존재하는 영역의 하한
//    x[2]: 최소값이 존재하는 영역의 상한
//    x[1]: x[0] < x[1] < x[2]인 동시에 f(x[0]) > f(x[1]) and f(x[2]) > f(x[1])인 점
// [Output]
//    xmin: f를 최소로 하는 점과 이 점에서의 목적 함수값

double GoldenSectionSearch(double *x, double (*f)(double), double *xmin)
{
    double TOLERANCE = 1.0e-6;
    double f1, f2, a0, a1, a2, a3;
    a0 = x[0];                    a3 = x[2];

    if (fabs(x[2] – x[1]) > fabs(x[1] - x[0])) {
           a1 = x[1]; a2 = x[1] + (1.0 - 0.618) * (x[2] - x[1]);  }
    else { a2 = x[1]; a1 = x[1] - (1.0 - 0.618) * (x[1] - x[0]);  }

    f1 = (*f)(a1);    f2 = (*f)(a2);

    while (fabs(a3 - a0) > TOLERANCE ) {
           if (f2 < f1) {        a0 = a1;   a1 = a2;   a2 = 0.618 * a1 + (1.0 - 0.618) * a3;
                                 f1 = f2;   f2 = (*f)(a2);                                 }
           else {      a3 = a2;  a2 = a1;   a1 = 0.618 * a2 + (1.0 - 0.618) * a0;
                       f2 = f1;  f1 = (*f)(a1);                                            }
    }

    if (f1 < f2) {    *xmin = a1;                      return f1; }
    else {            *xmin = a2;                      return f2; }
}
```

황금 분할법 Programming Guide (4)

```
#include <stdio.h>
#include <math.h>

double f1(double x);
double f2(double x);
double f3(double x);
void FindSection(double x_start, double x_delta, double (*ObjFunc)(double),
double *x);
double GoldenSectionSearch(double *x, double (*f)(double), double *xmin);

int main()
{
  double init_x = 0;
  double delta = 1;
  double *x = new double [3];
  double *xmin = 0;
  double f_min;

  //f1
  FindSection(init_x,delta,f1,x);
  f_min = GoldenSectionSearch(x,f1,xmin);

  //f2
  FindSection(init_x,delta,f2,x);
  f_min = GoldenSectionSearch(x,f2,xmin);

  //f3
  FindSection(init_x,delta,f3,x);
  f_min = GoldenSectionSearch(x,f3,xmin);

  return 0;
}
```

```
//f(x)=x^2
double f1(double x)
{
      return x*x;
}

//f(x)=sin x
double f2(double x)
{
     return sin(x);
}

//f(x)=x^3-x^2+x-1
double f3(double x)
{
       return pow(x,3) - pow(x,2) + x - 1;
}
```

황금 분할법의 확장
- 목적 함수가 다변수 함수일 때 탐색 방향이 주어질 경우

	목적 함수 식	탐색 방향	시작 위치
1변수 함수	double (*f)(double x) 하나의 변수만 입력	1변수 함수이므로 탐색 방향은 이미 주어져 있음	double x_start
다변수 함수	double (*f)(double *x, int n) 여러 개의 변수를 입력 받고, 변수의 개수도 입력	탐색 방향을 Vector 형태로 주어야 함	double *x_start

```
double f(double x);
void FindSection(double x_start, double x_delta, double (*f)(double), double *x);
double GoldenSectionSearch(double *section, double (*f)(double), double *xmin);
```

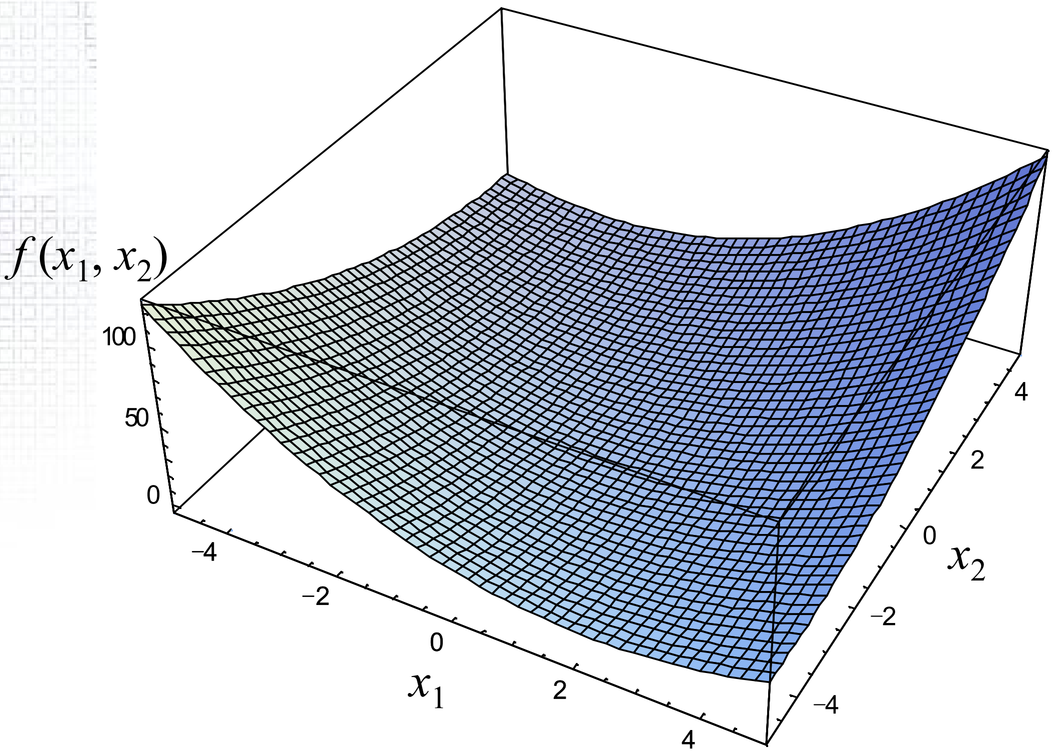
⇩

```
double f(double *x, int n);
void FindSection(double *x_start, double *x_delta, double (*f)(double*, int), double **x);
double GoldenSectionSearch(double **section, double (*f)(double*, int), double *xmin);
```

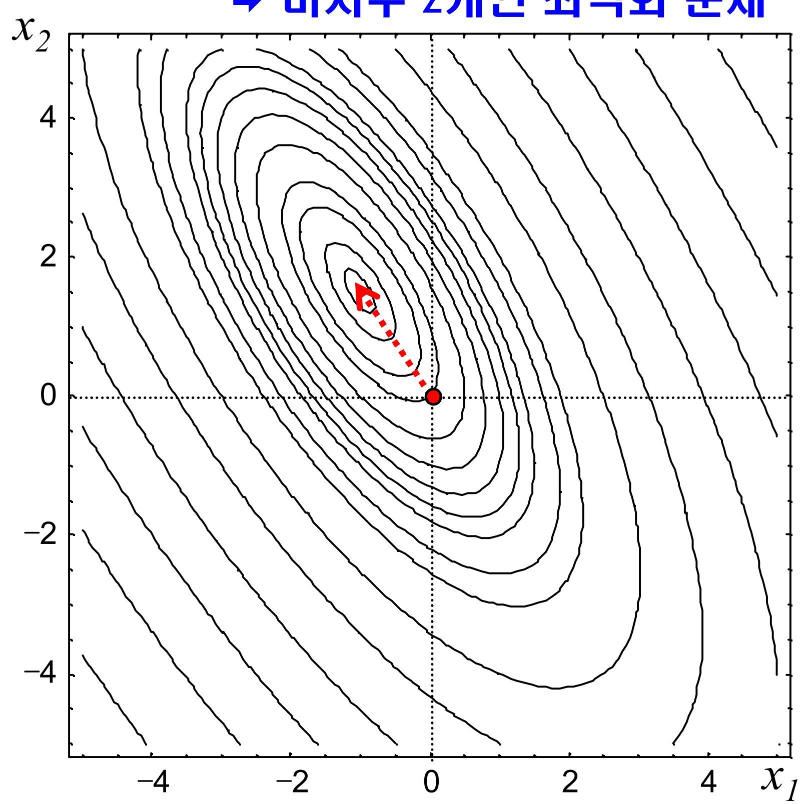
비제약 최적화 문제 예시

- **황금분할법을 이용하여 다음 2변수 함수의 최소점을 구하시오. 단, 시작점** $\mathbf{x}^{(0)}$ = (0, 0), convergence tolerance $\varepsilon = 10^{-6}$, **탐색 방향은** (1.0, -1.5)**이다.**

$$\textit{Minimize}\;\; f(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$



➡ 미지수 2개인 최적화 문제



참 고 자 료

공학수학 Review

- Directional Derivative & Gradient Vector

Advanced
Ship
Design
Automation
Laboratory

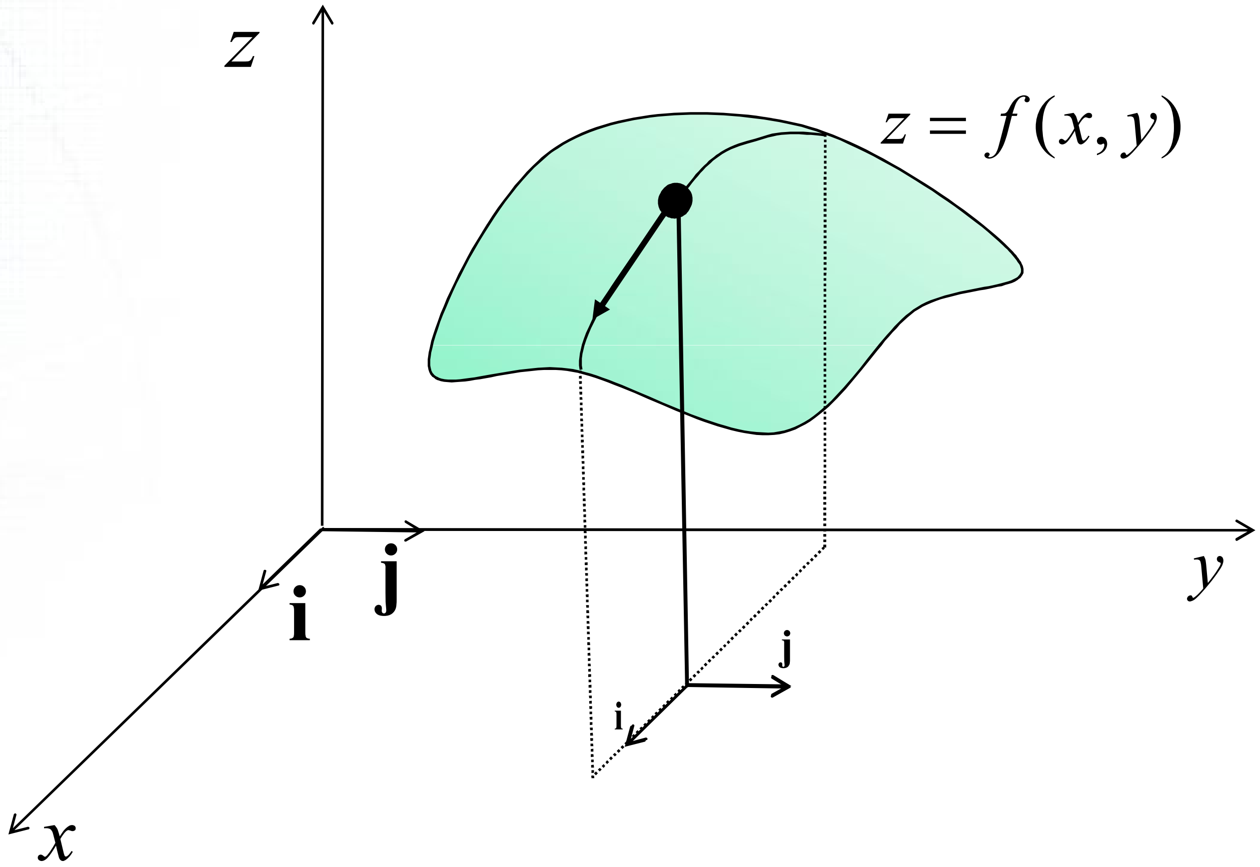
공학수학 Review

- Directional Derivative & Gradient Vector

Advanced
Ship
Design
Automation
Laboratory

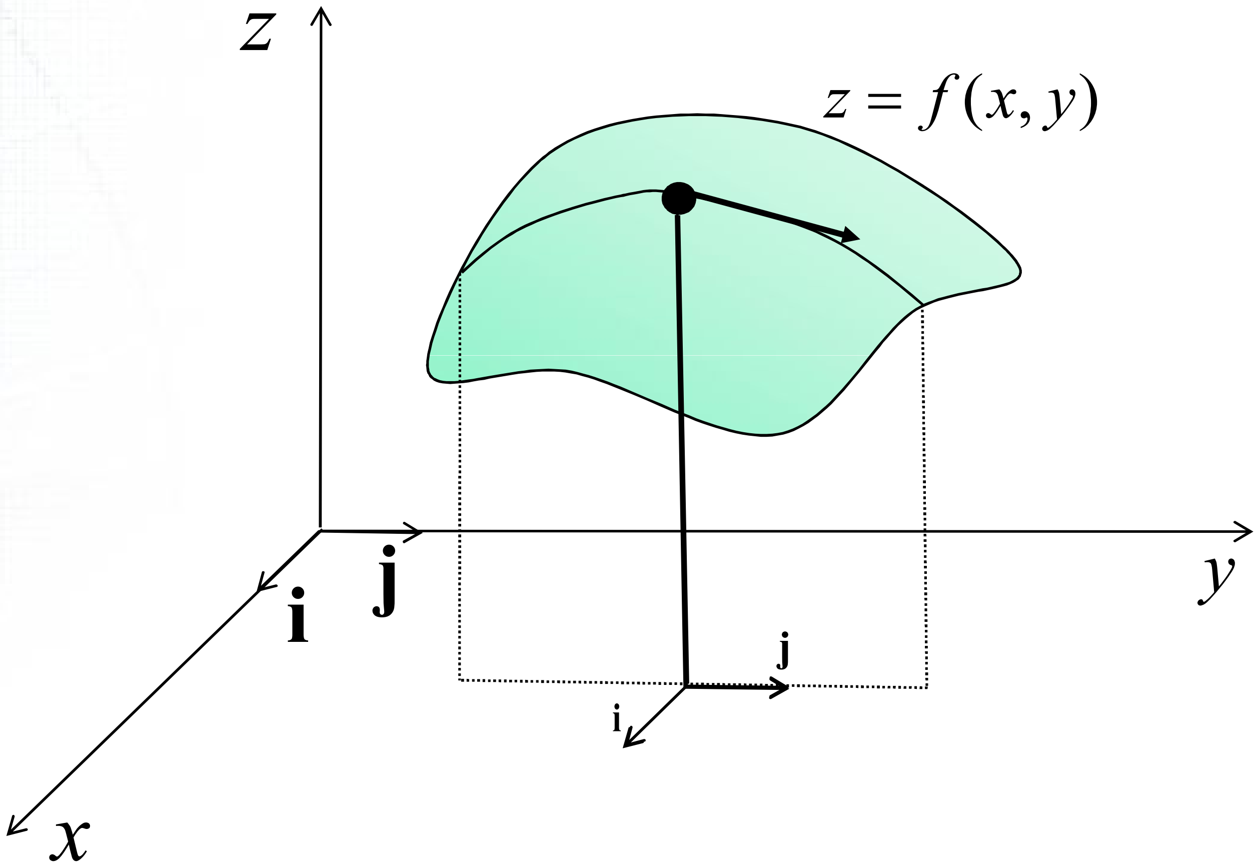
Directional Derivative

$\dfrac{\partial f}{\partial x}$:Rate of change of f in the $\mathbf{i}$ -direction



Directional Derivative

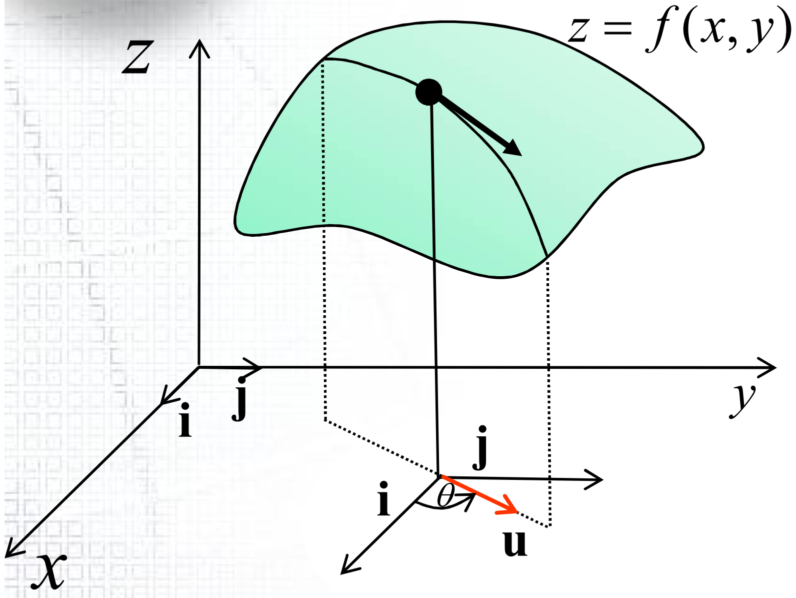
$\frac{\partial f}{\partial y}$:Rate of change of f in the $\mathbf{j}$ -direction



Directional Derivative

$$\nabla f(x,y) = \frac{\partial f}{\partial x}\mathbf{i} + \frac{\partial f}{\partial y}\mathbf{j}$$

The rate of change of f in the direction given by the vector $\mathbf{u}$: $D_{\mathbf{u}} f$



The rate of change of f in the direction of $\mathbf{x}$: $\frac{\partial f}{\partial x}$

The component of $\frac{\partial f}{\partial x}$ in the direction of $\mathbf{u}$: $\frac{\partial f}{\partial x}\cos\theta$

The rate of change of f in the direction of $\mathbf{y}$: $\frac{\partial f}{\partial y}$

The component of $\frac{\partial f}{\partial y}$ in the direction of $\mathbf{u}$: $\frac{\partial f}{\partial y}\cos(\frac{\pi}{2}-\theta)$

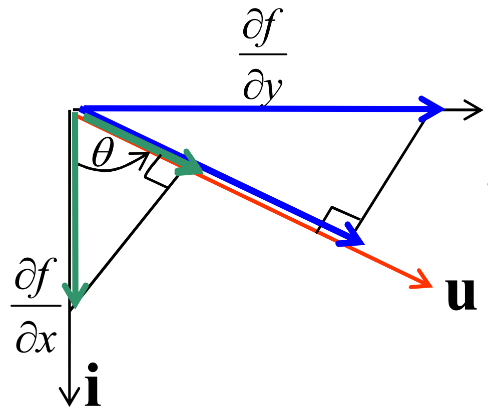
$$= \frac{\partial f}{\partial y}\sin\theta$$

The rate of change of f in the direction given by the vector $\mathbf{u}$: $D_{\mathbf{u}} f$

= +

$$= \frac{\partial f}{\partial x}\cos\theta + \frac{\partial f}{\partial y}\sin\theta$$

$$= \left(\frac{\partial f}{\partial x}\mathbf{i} + \frac{\partial f}{\partial y}\mathbf{j}\right)\bullet(\cos\theta\,\mathbf{i} + \sin\theta\,\mathbf{j}) = \nabla f \cdot \mathbf{u}$$



Directional Derivative

- Example 2
  Gradient at a Point

If $F(x,y,z)=xy^2+3x^2-z^3$, find $\nabla F(x,y,z)$ at $(2,-1,4)$.

Solution)

$$\nabla F(x,y,z) = \frac{\partial}{\partial x}(xy^2+3x^2-z^3)\mathbf{i} + \frac{\partial}{dy}(xy^2+3x^2-z^3)\mathbf{j} + \frac{\partial}{dz}(xy^2+3x^2-z^3)\mathbf{k}$$

$$= (y^2+6x)\mathbf{i} + 2xy\mathbf{j} - 3z^2\mathbf{k}$$

$$\nabla F(2,-1,4) = 13\mathbf{i} - 4\mathbf{j} - 48\mathbf{k}$$

Directional Derivative

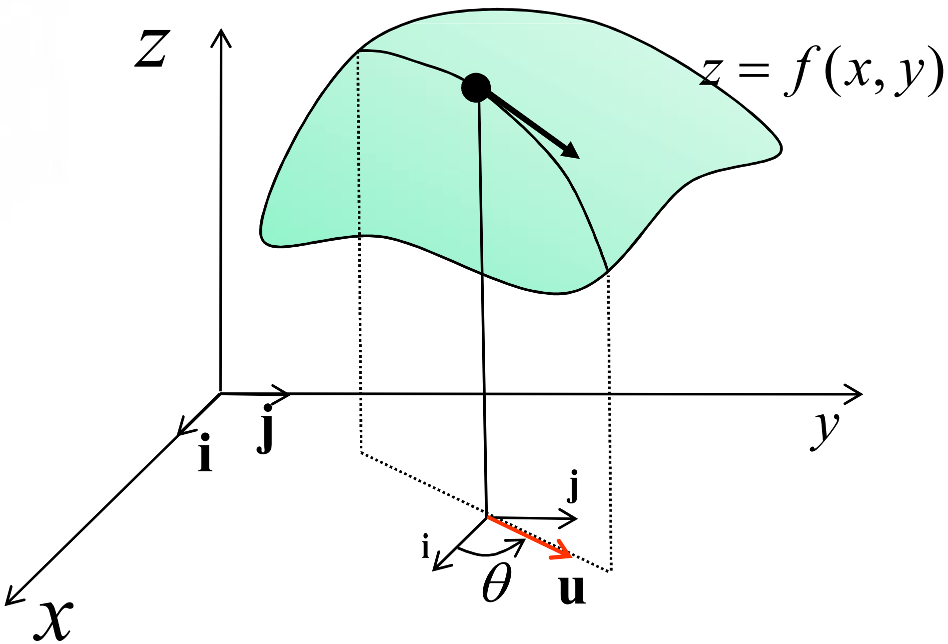
$$\nabla f(x,y) = \frac{\partial f}{\partial x}\mathbf{i} + \frac{\partial f}{\partial y}\mathbf{j}$$

Theorem 9.6 **Computing a Directional Derivative**

If $z = f(x,y)$ **is differentiable function of x and y and** $\mathbf{u} = \cos\theta\,\mathbf{i} + \sin\theta\,\mathbf{j}$
then,

$$D_{\mathbf{u}} f(x,y) = \nabla f(x,y)\cdot\mathbf{u}$$

$$D_{\mathbf{u}} f(x,y) = [f_x(x,y)\mathbf{i} + f_y(x,y)\mathbf{j}]\cdot(\cos\theta\,\mathbf{i} + \sin\theta\,\mathbf{j})$$
$$= \nabla f(x,y)\cdot\mathbf{u}$$



Directional Derivative

- **Example 3**
  Directional Derivative

Find the directional derivative of $f(x,y)=2x^2y^3+6xy$ at $(1,1)$ in the direction of a unit vector whose angle with the positive x-axis is $\pi/6$.

Solution)

$$\nabla f(x,y) = \frac{\partial f}{\partial x}\mathbf{i} + \frac{\partial f}{\partial y}\mathbf{j}$$

$$= (4xy^3 + 6y)\mathbf{i} + (6x^2y^2 + 6x)\mathbf{j}$$

$$\nabla f(1,1) = 10\mathbf{i} + 12\mathbf{j}$$

$$\mathbf{u} = \cos\theta\,\mathbf{i} + \sin\theta\,\mathbf{j} = \cos\frac{\pi}{6}\mathbf{i} + \sin\frac{\pi}{6}\mathbf{j}$$

$$D_{\mathbf{u}} f(1,1) = \nabla f(1,1)\cdot\mathbf{u}$$

$$= (10\mathbf{i} + 12\mathbf{j})\cdot\left(\frac{\sqrt{3}}{2}\mathbf{i} + \frac{1}{2}\mathbf{j}\right)$$

$$= 5\sqrt{3} + 6$$

Directional Derivative

■ Example 4 Directional Derivative

Consider the plane that is perpendicular to the *xy*-plane and passes through the points $P(2,1)$ and $Q(3,2)$. What is the slope of the tangent line to the curve on intersection of this plane with the surface $f(x,y)=4x^2+y^2$ at $(2,1,17)$ in the direction of Q?

Solution)

$$f(x,y)=4x^2+y^2$$

$$\nabla f(x,y)=8x\mathbf{i}+2y\mathbf{j}$$

$$\nabla f(2,1)=16\mathbf{i}+2\mathbf{j}$$

$$\overrightarrow{PQ}=\mathbf{u}=\frac{1}{\sqrt{2}}\mathbf{i}+\frac{1}{\sqrt{2}}\mathbf{j}$$

$$\begin{aligned} D_{\mathbf{u}}f(2,1)&=\nabla f(2,1)\bullet\mathbf{u}\\ &=(16\mathbf{i}+2\mathbf{j})\bullet\left(\frac{1}{\sqrt{2}}\mathbf{i}+\frac{1}{\sqrt{2}}\mathbf{j}\right)\\ &=9\sqrt{2}\end{aligned}$$

Directional Derivative

- **Example 5**
  Directional Derivative

Find the directional derivative of $F(x,y,z)=xy^2-4x^2y+z^2$ at $(1,-1,2)$ in the direction of $6\mathbf{i}+2\mathbf{j}+3\mathbf{k}$.

Solution)

$$f(x,y,z)=xy^2-4x^2y+z^2$$

$$\nabla f(x,y,z)=\frac{\partial}{\partial x}(xy^2-4x^2y+z)\mathbf{i}+\frac{\partial}{\partial y}(xy^2-4x^2y+z)\mathbf{j}+\frac{\partial}{\partial z}(xy^2-4x^2y+z)\mathbf{k}$$

$$=(y^2-8xy)\mathbf{i}+(2xy-4x^2)\mathbf{j}+2z\mathbf{k}$$

$$\nabla f(1,-1,2)=9\mathbf{i}-6\mathbf{j}+4\mathbf{k}$$

$$\|6\mathbf{i}+2\mathbf{j}+3\mathbf{k}\|=7$$

$$\mathbf{u}=\frac{1}{7}\cdot(6\mathbf{i}+2\mathbf{j}+3\mathbf{k})=\frac{6}{7}\mathbf{i}+\frac{2}{7}\mathbf{j}+\frac{3}{7}\mathbf{k}$$

$$D_{\mathbf{u}}F(1,-1,2)=(9\mathbf{i}-6\mathbf{j}+4\mathbf{k})\bullet\left(\frac{6}{7}\mathbf{i}+\frac{2}{7}\mathbf{j}+\frac{3}{7}\mathbf{k}\right)$$

$$=\frac{54}{7}$$

Directional Derivative

∇f Points in the direction of maximum increase of f at P

The rate of change of f in the direction given by the vector $\mathbf{u}$:

$$D_{\mathbf{u}} f(x,y) = [f_x(x,y)\mathbf{i} + f_y(x,y)\mathbf{j}] \cdot (\cos\theta\,\mathbf{i} + \sin\theta\,\mathbf{j}) = \nabla f(x,y) \cdot \mathbf{u}$$

$$D_{\mathbf{u}} f = |\nabla f||\mathbf{u}|\cos\phi = |\nabla f|\cos\phi, \quad \phi\text{:angle between } \nabla f \text{ and } \mathbf{u}$$

$$-1 \le \cos\phi \le 1$$

The maximum value of $D_{\mathbf{u}} f \implies D_{\mathbf{u}} f = |\nabla f|$ **, When** $\cos\phi = 1,\ \phi = 0$

$\Downarrow$

$\mathbf{u}$ **has the same direction of** ∇f

$\Rightarrow$ ∇f **is the direction of maximum increase of f at P**

$-\nabla f$ **is the direction of maximum decrease of f at P**

Directional Derivative

- **Example 6**
  Max/Min of Directional Derivative

In Example 5 the maximum value of the directional derivative at F at $(1,-1,2)$ is

$$\|\nabla F(1,-1,2)\| = \sqrt{133}.$$

The minimum value of $D_{\mathbf{u}}F(1,-1,2)$ is then

$$-\sqrt{133}.$$

Directional Derivative

- **Example 7**
  Direction of Steepest Ascent

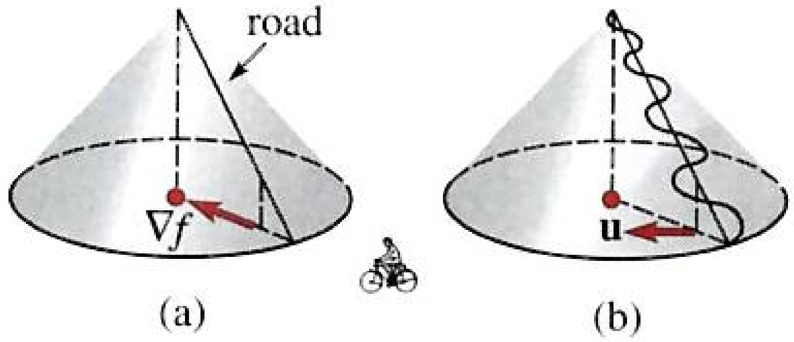
Each year in Los Angeles there is a bicycle race up to the top of a hill by a road known to be the steepest in the city. To understand why a bicyclist with a modicum of sanity will zigzag up the road, let us suppose the graph of

$$f(x,y) = 4 - \frac{2}{3}\sqrt{x^2+y^2},\ 0 \le z \le 4,$$

shown in Figure (a) is a mathematical model of the hill. The gradient of f is

$$\nabla f(x,y) = \frac{2}{3}\left[\frac{-x}{\sqrt{x^2+y^2}}\mathbf{i} + \frac{-y}{\sqrt{x^2+y^2}}\mathbf{j}\right] = \frac{2/3}{\sqrt{x^2+y^2}}\mathbf{r}$$

where $\mathbf{r}=-x\mathbf{i}-y\mathbf{j}$ is a vector pointing to the center of the circular base.



Thus the steepest ascent up the hill is a straight road whose projection in the xy-plane is a radius of the circular base. Since $D_{\mathbf{u}}f=\mathrm{comp}_{\mathbf{u}}\nabla f$, a bicyclist will zigzag, or seek a direction $\mathbf{u}$ other than ∇f, in order to reduce this component.

Directional Derivative

- Example 8
  Direction to Cool Off Fastest

The temperature in a rectangular box is approximated by

$$T(x,y,z) = xyz(1-x)(2-y)(3-z)$$
$$0 \le x \le 1,\ 0 \le y \le 2,\ 0 \le z \le 3.$$

If a mosquito is located at (½,1,1), in which direction should it fly to cool off as rapidly as possible?

Solution)

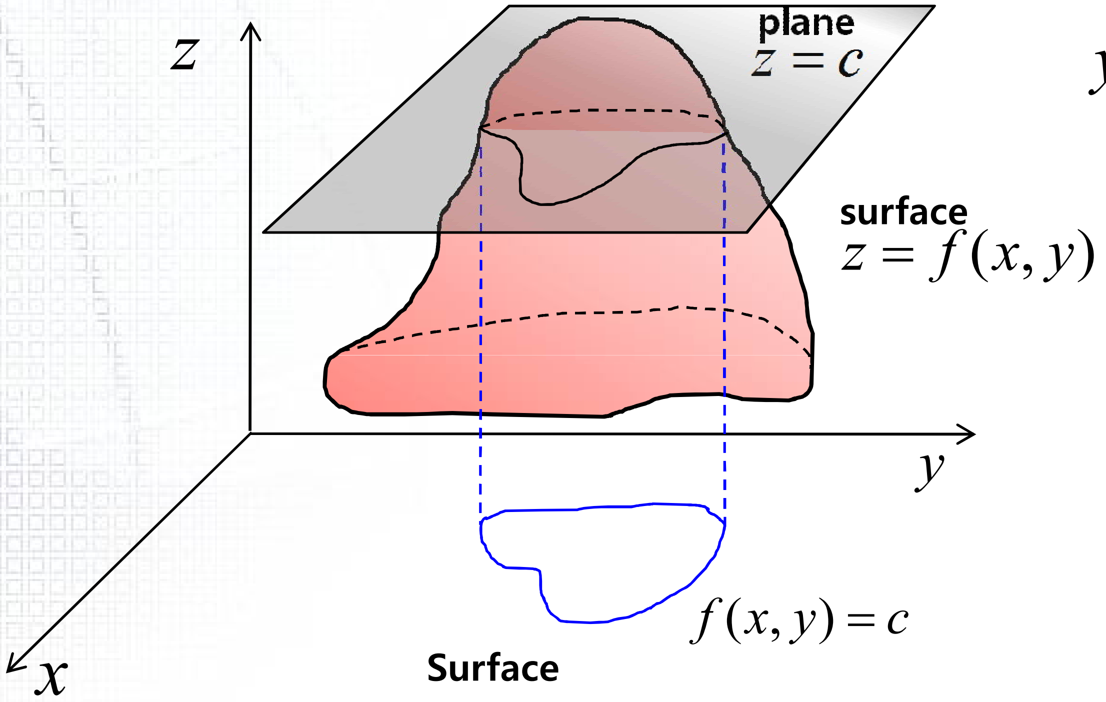
$$\nabla T(x,y,z) = \frac{\partial T(x,y,z)}{\partial x}\mathbf{i} + \frac{\partial T(x,y,z)}{\partial y}\mathbf{j} + \frac{\partial T(x,y,z)}{\partial x}\mathbf{k}$$
$$= yz(2-y)(3-z)(1-2x)\mathbf{i}$$
$$+ xz(1-x)(3-z)(2-2y)\mathbf{j}$$
$$+ xy(1-x)(2-y)(3-2z)\mathbf{k}$$

$$\nabla T\left(\frac{1}{2},1,1\right) = \frac{1}{4}\mathbf{k}$$

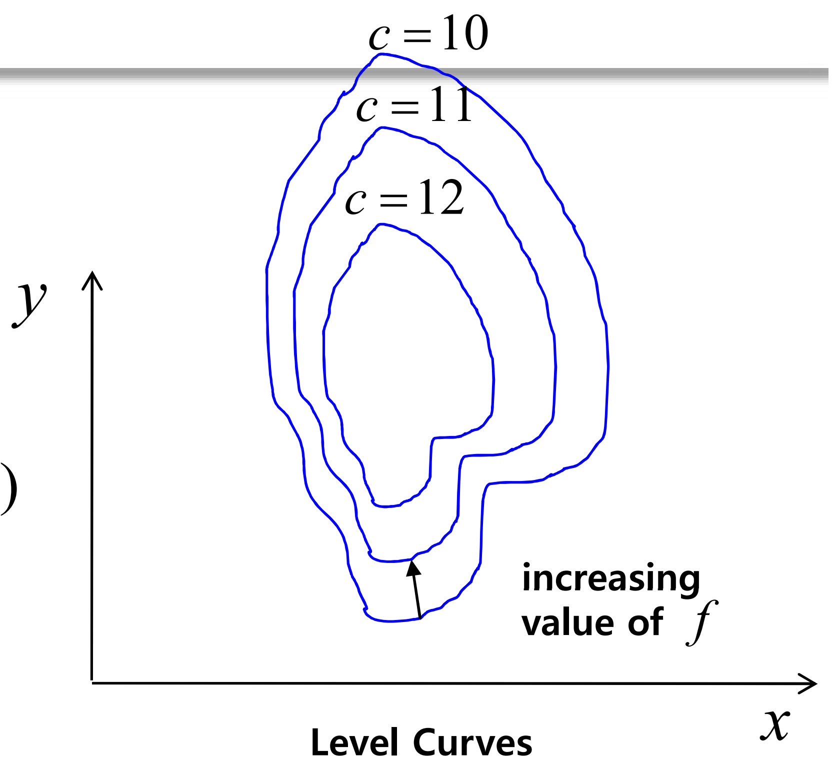
To cool off most rapidly, the mosquito should fly in the direction of $-1/4k$;
that is, it should dive for the floor of the box, where the temperature is $T(x,y,0)=0$.

Directional Derivatives

Level Curves



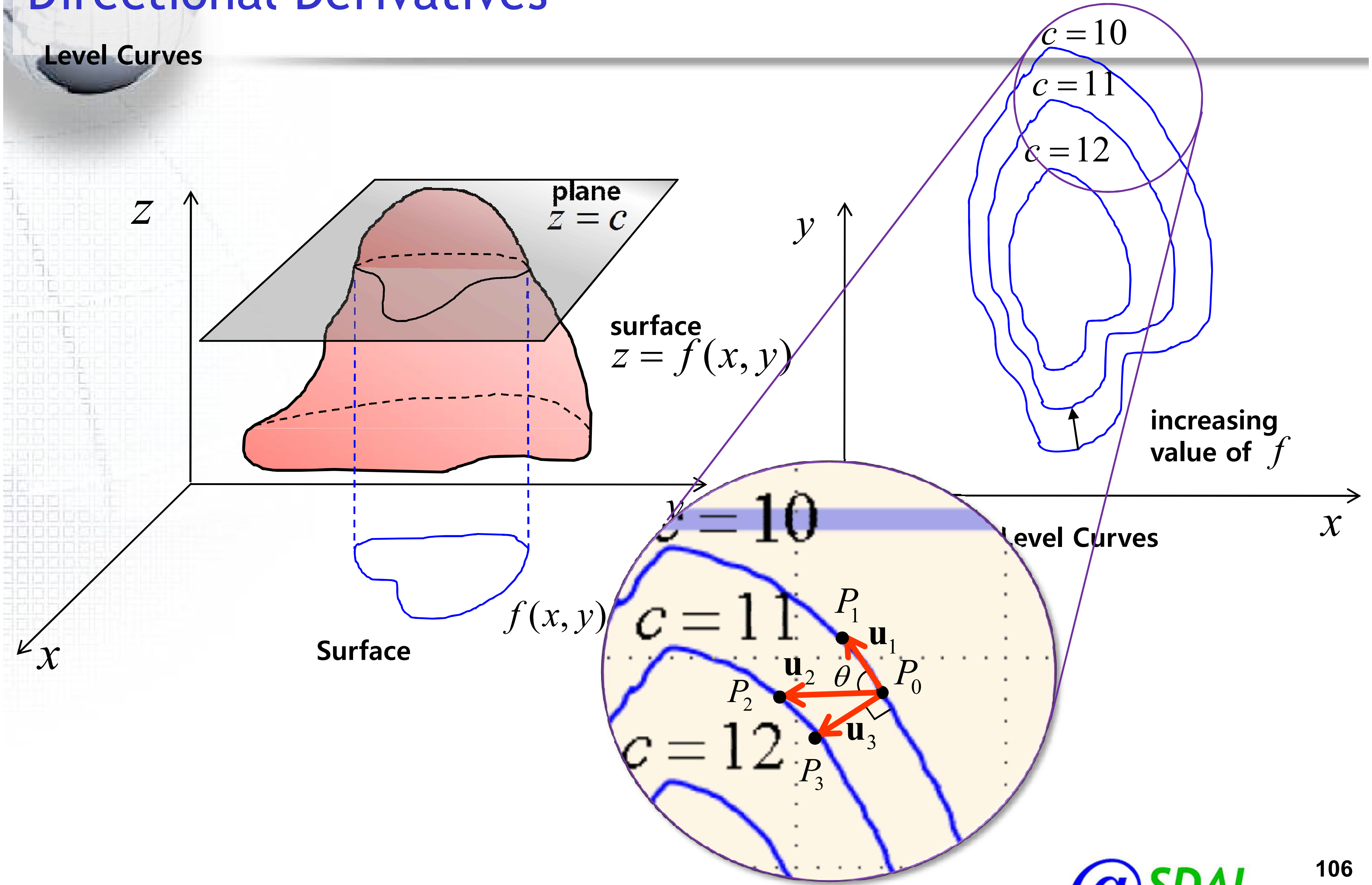
Surface



Level Curves

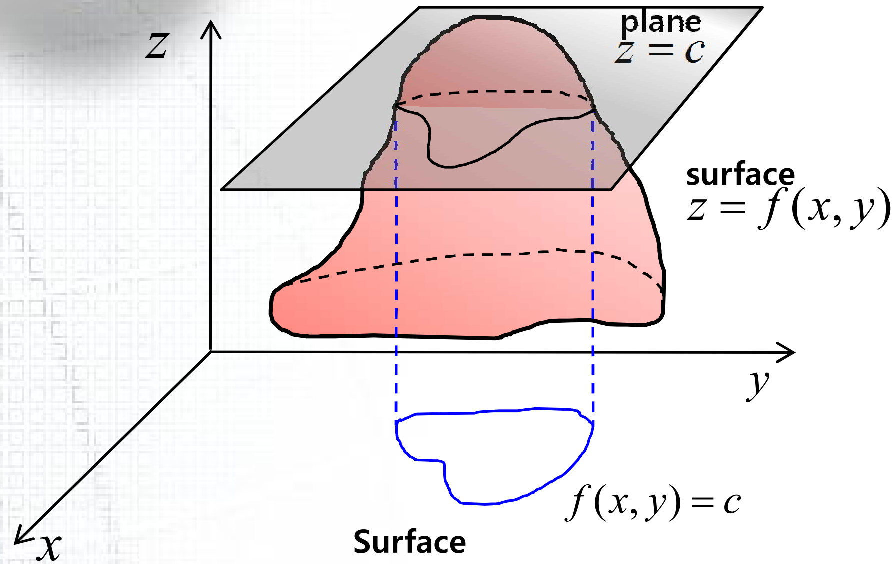
Directional Derivatives

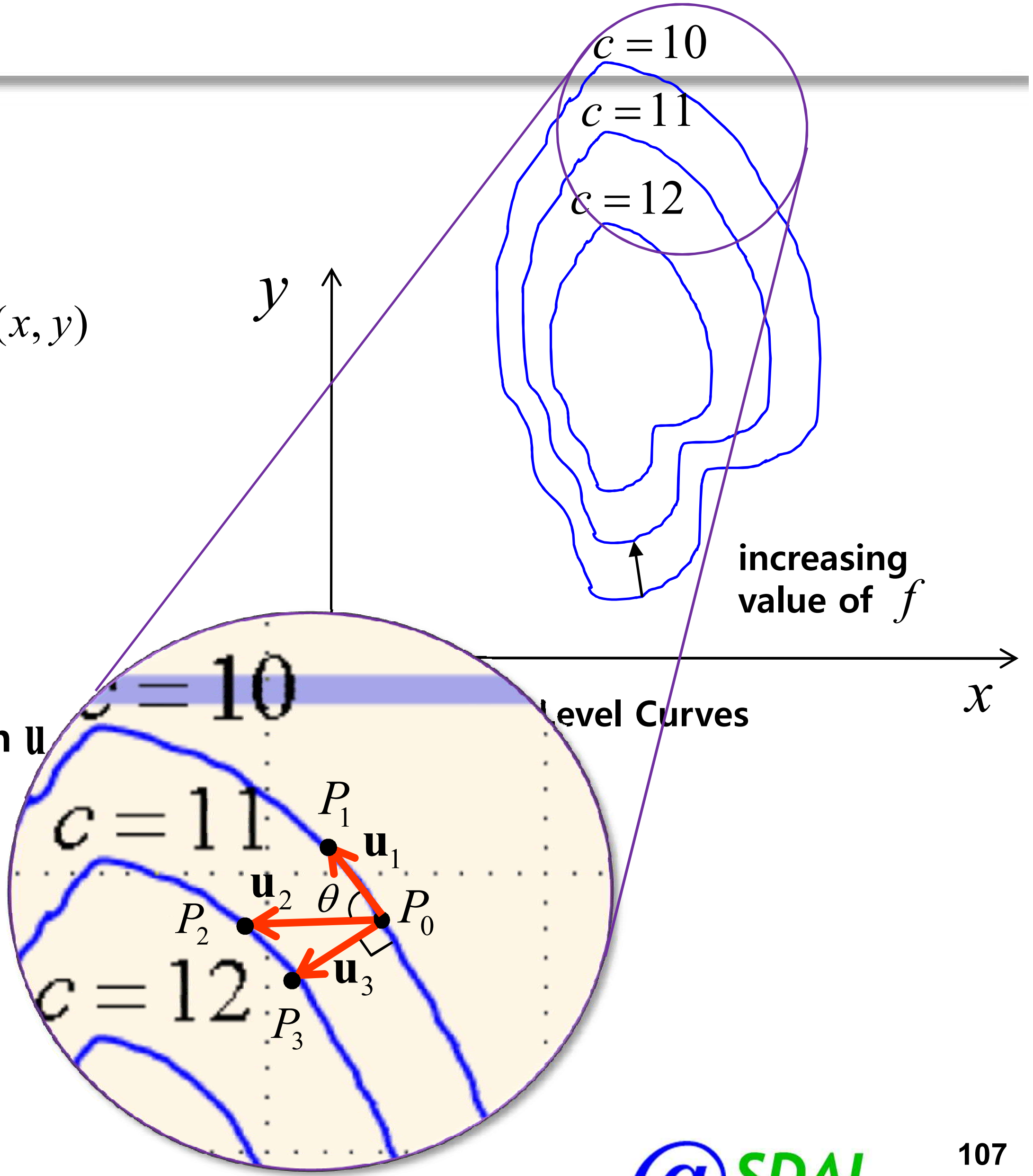
Level Curves



Directional Derivatives

Level Curves





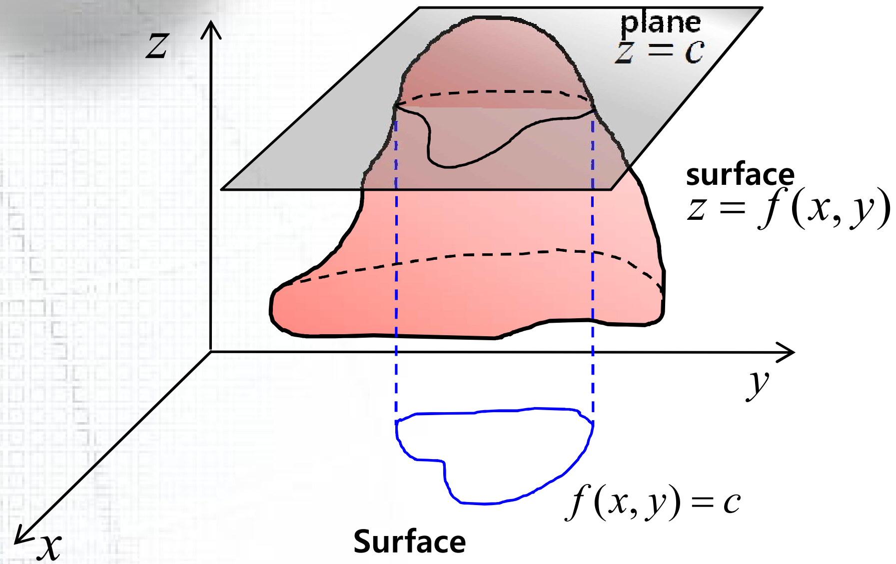
The rate of change of f **in the direction u given by the vector :** $D_{\mathbf{u}} f(x, y)$

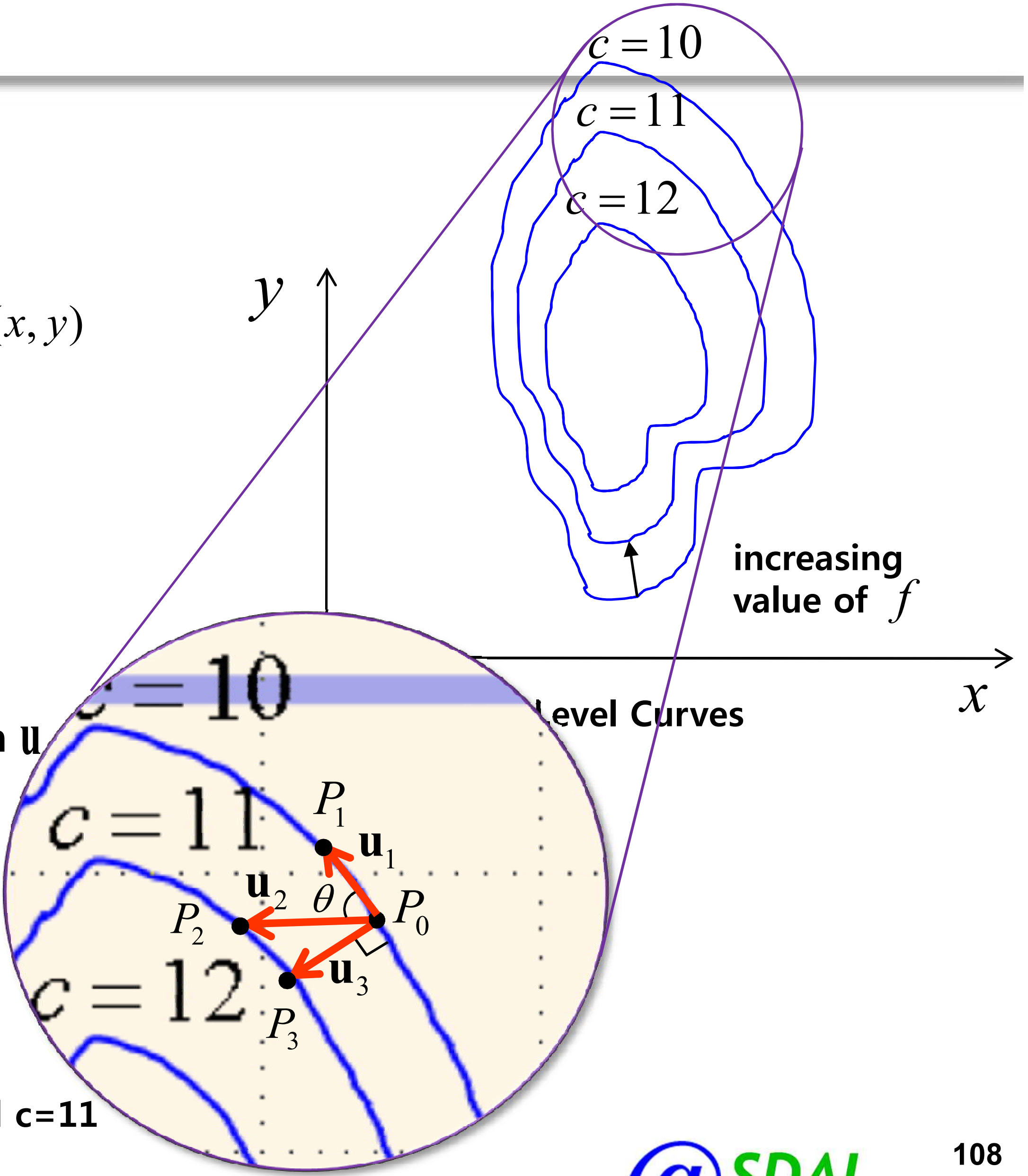
$$D_{\mathbf{u}_1} f(x, y) = 0$$

$$\because \frac{f(P_1) - f(P_0)}{\overline{P_1P_0}} = \frac{10 - 10}{\overline{P_1P_0}} = 0$$

Directional Derivatives

Level Curves





The rate of change of f in the direction u given by the vector : $D_{\mathbf{u}} f(x,y)$

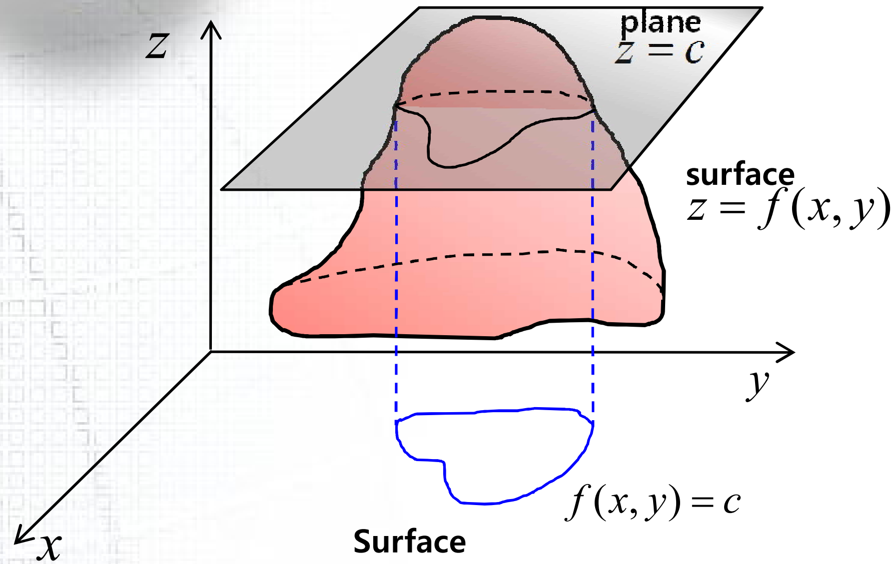
$$D_{\mathbf{u}_2} f(x,y) \text{ Vs. } D_{\mathbf{u}_3} f(x,y)$$

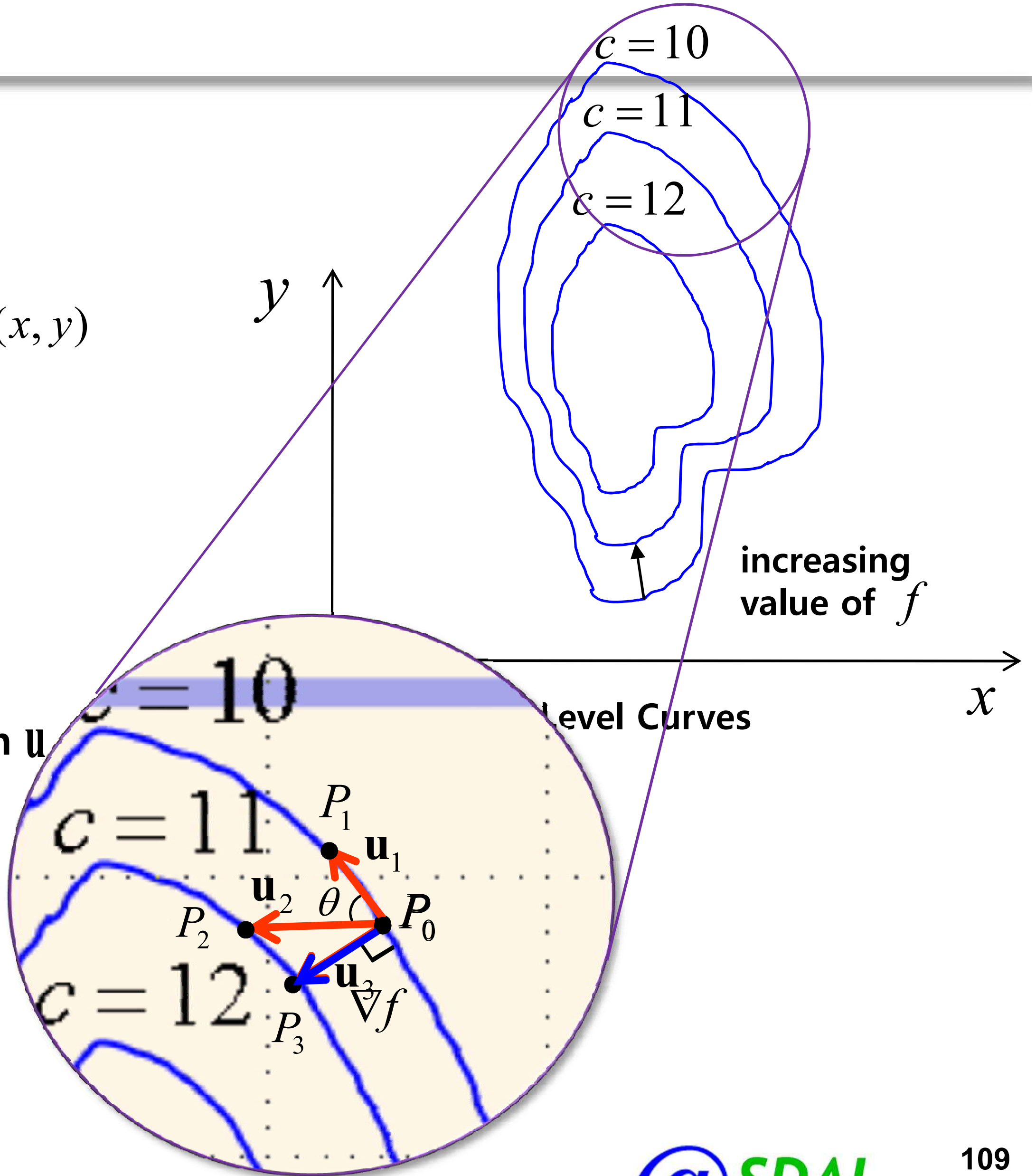
$$\frac{f(P_2) - f(P_0)}{\overline{P_2P_0}} < \frac{f(P_3) - f(P_0)}{\overline{P_3P_0}}$$

$\because$ $\overline{P_3P_0}$ **is the shortest path between c=10 and c=11**

Directional Derivatives

Level Curves





The rate of change of f in the direction $\mathbf{u}$ given by the vector : $D_{\mathbf{u}} f(x, y)$

∇f is the direction of maximum increase of f at P_0