

# 2012 年度 大問 4

hari64boli64 (hari64boli64@gmail.com)

2023 年 5 月 22 日

## 1 問題

双対問題を用いた最適化

## 2 解答

(1),(2),(3)

略

(4)

(3) より、

$$\begin{aligned} & \text{Minimize} \quad s \\ & \text{subject to} \quad 3x_1^2 - 4x_1x_2 - 2x_2^2 + s \geq 0 \quad (\forall \mathbf{x} \in U) \end{aligned}$$

を解けばよい。

(2) より、

$$\begin{aligned} 3x_1^2 - 4x_1x_2 - 2x_2^2 + s \geq t_1(-2x_1^2 - 2x_1x_2 - x_2^2 + 4) \\ + t_2(x_1^2 - 1) \quad (\forall \mathbf{x} \in \mathbb{R}^n) \end{aligned}$$

を満たす  $t_1, t_2$  が存在する範囲内で、 $s$  を最小化すればよい。

(1) より、

$$\begin{pmatrix} 3 + 2t_1 - t_2 & -2 + t_1 & \\ -2 + t_1 & -2 + t_1 & \\ & & s - 4t_1 + t_2 \end{pmatrix} \succeq 0$$

の下で、 $s$  を最小化すればよい。

則ち、

$$\begin{pmatrix} 3 + 2t_1 - t_2 & -2 + t_1 \\ -2 + t_1 & -2 + t_1 \end{pmatrix} \succeq 0$$

の下で、 $4t_1 - t_2$  を最小化すればよい。

これは、固有値を求めた上で考えると、解の配置問題である。

尤も、直接解くのはかなり厳しい。

しかし、適当に考えると、 $t_1 = 2, t_2 = 7$  の時、固有値は共に 0 で、目的関数の値は 1 になる。

そして、 $x_1 = 1, x_2 = -1 + \sqrt{3}$  を代入すると、確かに元の最適化問題において、制約を満たした上で、この目的関数も 1 になる。

以上より、 $s = 1$  が最適解である。

なお、後半 3 行は計算機パワーを用いたチートであり、試験の上では、(3),(2),(1) を用いた変形さえ出来ていれば、十分ではないかと思われる。

### 3 おまけ

ソースコード 1 vis

```

1 import matplotlib.pyplot as plt
2 import random
3 import numpy as np
4 from tqdm import tqdm
5
6
7 def g_0(x1, x2):
8     return -3 * x1 * x1 + 4 * x1 * x2 + 2 * x2 * x2
9
10
11 def g_1(x1, x2):
12     return -2 * x1 * x1 - 2 * x1 * x2 - x2 * x2 + 4
13
14
15 def g_2(x1, x2):
16     return x1 * x1 - 1
17

```

```

18
19 def solve():
20     ans = -1e9
21     best_x1 = 0
22     best_x2 = 0
23     for _ in tqdm(range(int(1e5))):
24         x1 = random.random() * 10 - 5
25         x2 = random.random() * 10 - 5
26         if g_1(x1, x2) >= 0 and g_2(x1, x2) >= 0:
27             ans = max(ans, g_0(x1, x2))
28             if ans == g_0(x1, x2):
29                 best_x1 = x1
30                 best_x2 = x2
31     print(f"ans: {ans} x1: {best_x1} x2: {best_x2}")
32
33     x1 = np.linspace(-5, 5, 1000)
34     x2 = np.linspace(-5, 5, 1000)
35     X1, X2 = np.meshgrid(x1, x2)
36
37     Z = g_0(X1, X2)
38     Z[g_1(X1, X2) < 0] -= 100
39     Z[g_2(X1, X2) < 0] -= 100
40     plt.imshow(Z, extent=[-5, 5, -5, 5], origin="lower")
41     plt.title("visualization of problem")
42     plt.savefig("4_vis.png")
43     plt.close()
44
45     Z[g_1(X1, X2) < 0] = np.nan
46     Z[g_2(X1, X2) < 0] = np.nan
47     plt.imshow(Z, extent=[-5, 5, -5, 5], origin="lower")
48     plt.colorbar()
49     plt.title("maximize g_0 s.t. g_1 >= 0, g_2 >= 0")
50     plt.savefig("4_actual.png")
51     plt.close()
52
53
54 def solve2():
55     ans = +1e9
56     best_t1 = 0
57     best_t2 = 0
58     for _ in tqdm(range(int(1e5))):
59         t1 = random.random() * 20 - 10
60         t2 = random.random() * 20 - 10
61         A = np.array([[3 + 2 * t1 - t2, -2 + t1], [-2 + t1, -2 +
62             t1]])
63         if np.all(np.linalg.eigvals(A) >= 0):
64             ans = min(ans, 4 * t1 - t2)

```

```

64         if ans == 4 * t1 - t2:
65             best_t1 = t1
66             best_t2 = t2
67         print(f"ans: {ans} t1: {best_t1} t2: {best_t2}")
68
69
70 def main():
71     solve()
72     # solve2()
73
74
75 if __name__ == "__main__":
76     main()

```

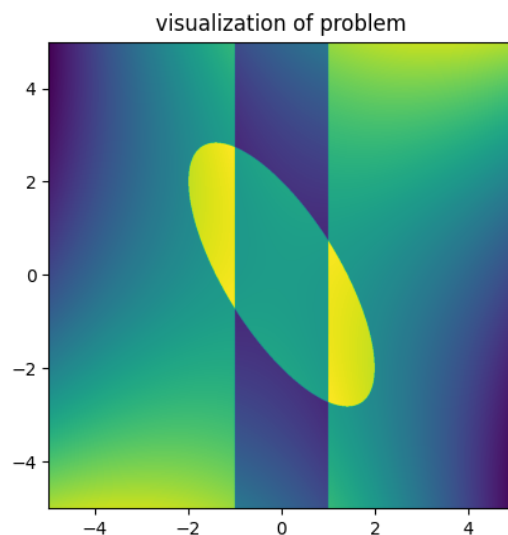


図1 イメージ図

1 個目の条件式は、とある楕円形内部の点であることを表す。  
 2 個目の条件式は、 $x_1 \leq -1$ 、または、 $x_1 \geq 1$ を表す。  
 目的関数は右上と左下の値が高い、双曲線のような形をしている。  
 そのことに対応した図であることが読みとれると思われる。

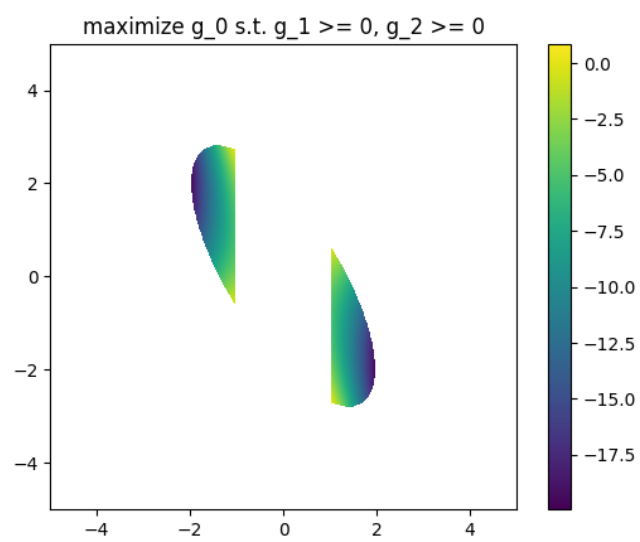


図 2 総当たりによる実際の図