注水问题

58121128 马浩轩

December 2022

1 实验目标

1.1 问题描述

注水问题: 考虑如下凸优化问题

$$
\begin{array}{rl}
\text{minimize} & -\sum_{i=1}^{n} \log\left(\alpha_i + x_i\right) \\
\text{subject to} & x \succeq 0\ , \quad \mathbf{1}^\top x = 1 \\
\text{condition} & x \in R^n, \quad \alpha_i \in R^n, \quad \alpha_i > 0
\end{array}
$$

1.2 实验要求

证明注水问题，并编程实现

2 实验过程

2.1 求解问题

其中 α_i>0。上述问题源自信息论，将功率分配给 n 个信道。变量 x_i 表示分配给第 i 个信道的发射功率，$\log\left(\alpha_i + x_i\right)$ 是信道的通信能力或者通信速率，因此上述问题即为将值为一的总功率分配给不同的信道，使得总的通信速率最大。

$$
\begin{gathered}
\text{对不等式约束 } x^* \succeq 0 \text{ 引入 Lagrange 乘子 } \lambda^* \in R^n \\
\text{对等式约束 } \mathbf{1}^\top x = 1 \text{ 引入一个乘子 } \nu^* \in R, \quad \text{得到如下 KKT 条件:}
\end{gathered}
$$

$$
\begin{gathered}
x^* \succeq 0 \\
\mathbf{1}^\top x^* = 1 \\
\lambda^* \succeq 0 \\
\lambda_i^* x_i^* = 0, \quad i = 1, \cdots, n \\
-1/(\alpha_i + x_i) - \lambda_i^* + \nu^* = 0, \quad i = 1, \cdots, n
\end{gathered}
$$

由于最后一个式子得 $\nu^* = 1/(\alpha_i + x_i) + \lambda_i^*$，代入前述公式得到如下转换：

$$
\begin{gathered}
x_i^*(\nu^* - 1/(\alpha_i + x_i)) = 0, \quad i = 1, \cdots, n \\
\nu^* \geq 1/(\alpha_i + x_i), \quad i = 1, \cdots, n
\end{gathered}
$$

因此，有如下两种情况：

(1) $\nu^* < 1/\alpha_i$:

当且仅当 $x_i^* > 0$ 时最后一个条件成立，因此 $\nu^* - 1/(\alpha_i + x_i) = 0$

$$
\text{解得：} \quad x_i^* = 1/\nu^* - \alpha^*
$$

(2) $\nu^* \geq 1/\alpha_i$:

当且仅当 $x_i^* = 0$ 时 KKT 条件成立

$$
\text{解得：} \quad x_i^* = 0
$$

所以有下式：

$$
x_i^* = \begin{cases} 1/\nu^* - \alpha^* & \nu^* < 1/\alpha_i \\ 0 & \nu^* \geq 1/\alpha_i \end{cases}
$$

化简得：$x_i^* = max\{\, 0, 1/\nu^* - \alpha_i\}$，代入条件 $\mathbf{1}^\top x^* = 1$ 得：

$$
\textstyle\sum_{i=1}^n max\{\, 0, 1/\nu^* - \alpha_i\} = 1
$$

方程左端是 $1/\nu^*$ 的分段线性函数，分割点为 α_i, 因此上述方程有唯一确定解。上述解题称为注水，这是因为我们可以将 α_i 看作第 i 片区域的的水平线，然后对整个区域注水，使其具有深度 $1/\nu$, 如图 1 所示，所需总水量为 $\sum_{i=1}^n max\{\, 0, 1/\nu^* - \alpha_i\}$。不断注水，直到水量为 1。第 i 片区域水位深度即为最优解 x_i^*。

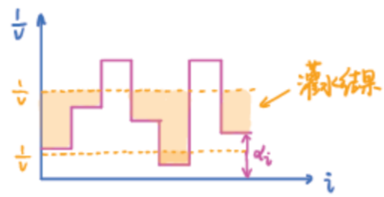


图 1: 注水问题

2.2 编程实现

详细代码见 58121128 马浩轩.cpp 文件。

在 print_alpha 函数中实现打印 α_i，如图 2：

```
void print_alpha(double* a) {
    for (auto i = 0; i < dimension; i++)
    {
        cout << "alpha[" << i << "] = " << fixed << setprecision(6) << a[i] << endl;
    }
}
```

图 2: print_alpha

在 print_x 函数中实现打印 x_i^*，如图 3:

```
void print_x(double* x) {
    for (auto i = 0; i < dimension; i++)
    {
        cout << "    x[" << i << "] = " << fixed << setprecision(6) << x[i] << endl;
    }
}
```

图 3: print_x

在 calculate_target 函数中，实现对通信速率计算，如图 4：

在 fill_water 函数中利用二分法，根据随机生成的 α_i，计算注水分量，如图 5：

```cpp
double calculate_target(double* alpha, double* x) {
    double target = 0;
    for (auto i = 0; i < dimension; i++)
    {
        target -= log(alpha[i] + x[i]);
    }
    return target;
}
```

图 4: calculate_target

```cpp
double fill_water(double* alpha, double total_water, double precision) {
    double x[dimension] = { 0 };//初始化x数组
    double upperbound = 1;//设置上界
    double lowerbound = 0;//设置下界
    double deviation = 1;//初始化总水量与注水量的差值
    double water_volume = 1.0;//初始化注水量
    while (Precision < abs(deviation)) {
        water_volume = (upperbound + lowerbound) / 2;//二分法逼近真实的注水量
        double temptotalwater = 0;//初始化已经注入的水量
        for (int i = 0; i < dimension; i++) {
            if (water_volume - alpha[i] > 0) {
                temptotalwater += (water_volume - alpha[i]);
            }
        }
        //即对注水量求和
        deviation = temptotalwater - total_water;
        if (deviation > 0) {
            upperbound = water_volume;
        }
        //真实的注水量多了，要减小上界
        else {
            lowerbound = water_volume;
        }
        //真实的注水量少了，要增加下界
    }
    //当误差大于精度时，进行逼近估算
    for (int i = 0; i < dimension; i++) {
        if (water_volume - alpha[i] > 0) {
            x[i] = water_volume - alpha[i];
        }
        else {
            x[i] = static_cast<double>(0);
        }
    }
    //计算x数组
    print_x(x);
    cout << "water_valume = " << water_volume << endl;
    return calculate_target(alpha, x);
}
```

图 5: fill_water

3 实验结果

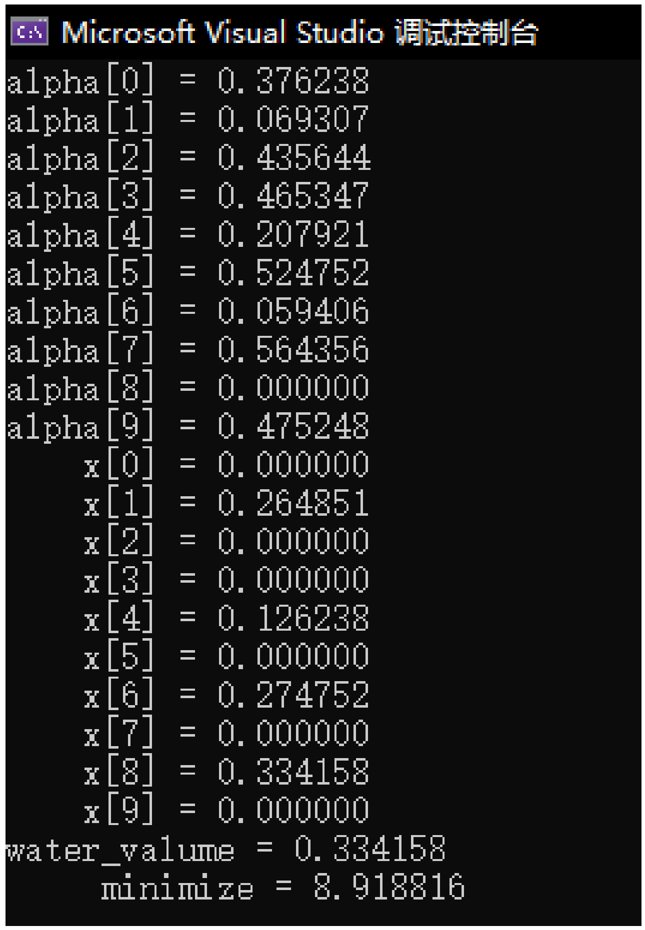
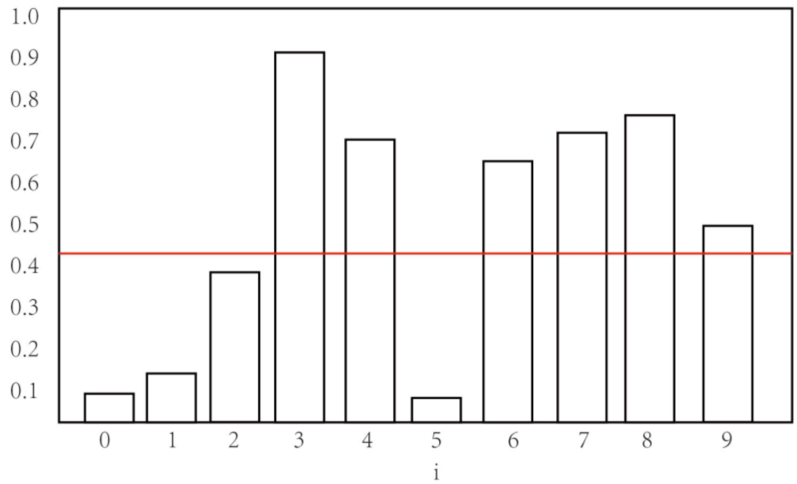


图 6: 输出结果

绘图如下：



如图所示，在精度为小数点后六位，维度为 10 的条件下，我们分析并完成了注水问题，求得了 alpha 数组与 x 数组，找到了最优值，红线所标为注水高度。实验结果正确，格式输出与样例相同，注水问题分析完成。

图 7: 绘图