第二次作业

尹朝阳 物理学系

1 题目 1

1.1 题目描述

Sketch the function $x^3 - 5x + 3 = 0$.

(i) Determine the two positive roots to 4 decimal places using the bisection method. Note: You first need to bracket each of the roots.

(ii) Take the two roots that you found in the previous question (accurate to 4 decimal places) and "polish them up" to 14 decimal places using the Newton-Raphson method.

(iii) Determine the two positive roots to 14 decimal places using the hybrid method.

1.2 程序描述

本程序需要用三种方法进行给定方程 $x^3 - 5x + 3 = 0$ 的 2 个正根的查找。

1. 二分法

首先找到两个根的大致范围。对于 $f(x) = x^3 - 5x + 3$，容易发现 $f(0) > 0$，$f(1) < 0$，$f(2) > 0$，则 2 个正根分别在区间 $(0, 1)$ 和 $(1, 2)$ 中，此时将 $0, 1, 2$ 设置为初值，采用二分法进行查找，即令 $x \leftarrow \frac{a+b}{2}$，考察 $f(x)$ 的正负性，从而确定根处于 x 的左半区间还是右半区间内，于是将搜索区间的长度减为原来的一半。通过递归式查找的方式，最终求得较为精确的解。

2. 牛顿法

首先将 $f(x)$ 在 x_0 处 Taylor 展开到 1 阶项：$f(x) = f(x_0) + f'(x_0) \cdot (x - x_0) + o(x^2)$，对于根 x，有 $f(x) = 0$，则可以得到迭代公式：

$$x \leftarrow x - \frac{f(x)}{f'(x)}$$

由于初始值 x_1，x_2 为二分法解得的粗略解，且函数在这两个初值附近为平滑的好函数，故不会出现 $f'(x) = 0$ 的情况。

3. 混合法

由于前两种方法无法处理所有的函数（例如：二分法无法找到 $x^2 = 0$ 的解，牛顿法使用时可能出现 $f'(x) = 0$ 的情况），故提出一种更为普适的方法：混合法。其结合了二分法和牛顿法两种方法，流程图如图 1 所示。

本程序源文件为/FindRoots/bisection_method.py，/FindRoots/Newton-Raphson_method.py 和/FindRoots/hybrid_method.py。

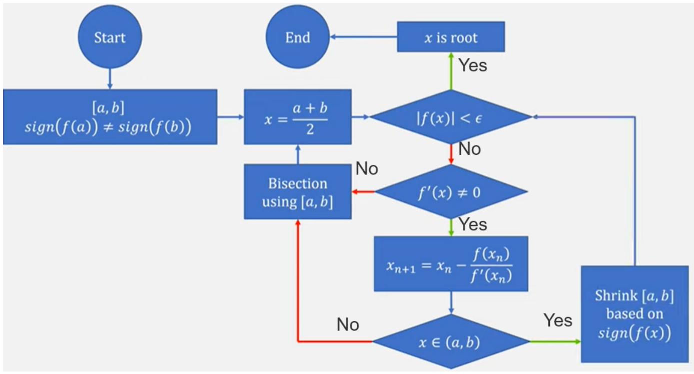

图 1: flowchart of hybrid method

1.3 伪代码

Algorithm 1.1 Find 2 positive roots of the equation by the **bisection** method

Output : 2 positive roots of the equation: $root1, root2$

function BISECTION(a, b)
    while $b - a \geq \delta$ **do**
        $x \leftarrow \frac{a+b}{2}$
        if $f(x) \cdot f(a) < 0$ **then**
            $b \leftarrow x$ $\triangleright$ $[a, b] \rightarrow [a, \frac{a+b}{2}]$
        else if $f(x) \cdot f(b) < 0$ **then**
            $a \leftarrow x$ $\triangleright$ $[a, b] \rightarrow [\frac{a+b}{2}, b]$
        end if
    end while
    return x
end function
$root1 \leftarrow$ BISECTION($0, 1$) $\triangleright$ root1 is bracketed by $0, 1$
$root2 \leftarrow$ BISECTION($1, 2$) $\triangleright$ root1 is bracketed by $1, 2$

Algorithm 1.2 Find 2 positive roots of the equation by **Newton-Raphson** method

Output : 2 positive roots of the equation: $root1, root2$

function NEWTON_RAPHSON(x)
    $x_0 \leftarrow x$ $\triangleright$ store the previous x
    $x \leftarrow x - \frac{f(x)}{f'(x)}$
    while $|x - x_0| \geq \epsilon$ **do**
        $x_0 \leftarrow x, \quad x \leftarrow x - \frac{f(x)}{f'(x)}$
    end while
    return x
end function
$root1 \leftarrow$ NEWTON_RAPHSON($x1$)
$root2 \leftarrow$ NEWTON_RAPHSON($x2$) $\triangleright$ x_1, x_2 are inherited from Algorithm 1.1

Algorithm 1.3 Find 2 positive roots of the equation by **hybrid** method

Output : 2 positive roots of the equation: $root1, root2$

function HYBRID(a, b)
  $x \leftarrow \frac{a+b}{2}$
  while $|f(x)| \geq \epsilon$ **do**
    if $f^{'}(x) \neq 0$ **then**
      $x_0 \leftarrow x$
      $x \leftarrow x - \frac{f(x)}{f^{'}(x)}$
      if $|x_0 - x| < \epsilon$ **then**
        return x
      end if
      if $x \in [a, b]$ **then**
        if $f(x) \cdot f(a) < 0$ **then**
          $b \leftarrow x$ $\triangleright [a, b] \rightarrow [a, \frac{a+b}{2}]$
        else if $f(x) \cdot f(b) < 0$ **then**
          $a \leftarrow x$ $\triangleright [a, b] \rightarrow [\frac{a+b}{2}, b]$
        end if
      else
        BISECTION(a, b)
      end if
    else
      BISECTION(a, b)
    end if
  end while
  return x
end function

$root1 \leftarrow$ HYBRID$(0, 1)$
$root2 \leftarrow$ HYBRID$(1, 2)$

1.4 测试用例

3 个程序的输出如下：

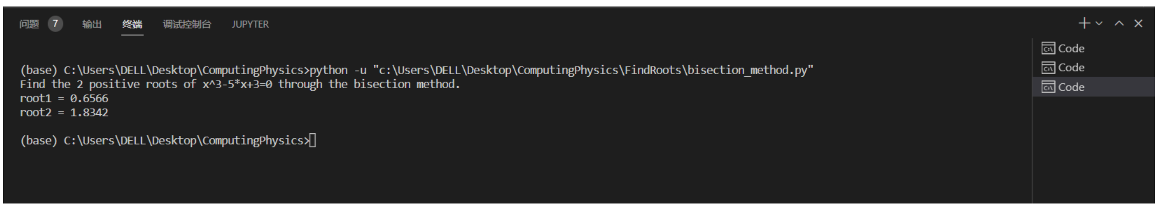

图 2: bisection_method.py

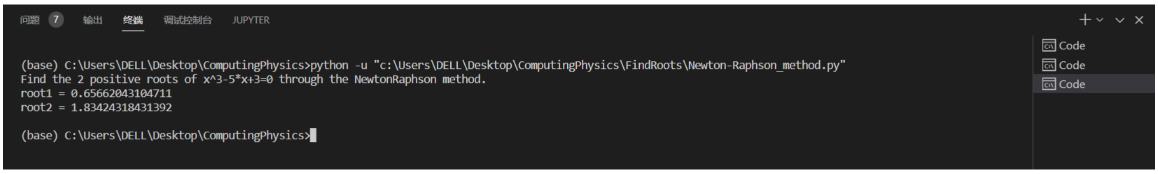

图 3: Newton-Raphson_method.py

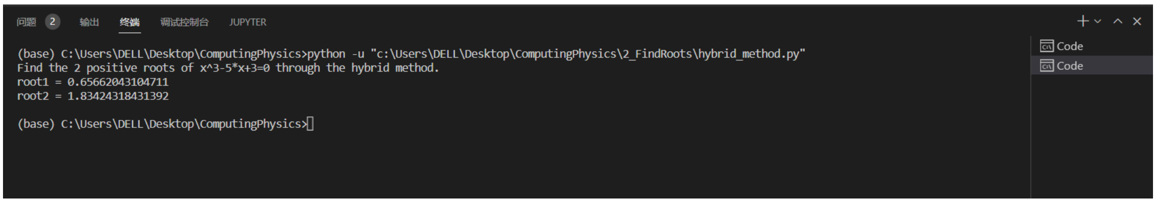

图 4: hybrid_method.py

2 题目 2

2.1 题目描述

Search for the minimum of the function $g(x,y) = \sin(x+y) + \cos(x+2*y)$ in the whole space.

2.2 程序描述

本程序选择最速下降法搜索二元函数的极小值（最小值）。

最速下降法的迭代公式为

$$x \leftarrow x - a * \frac{\partial}{\partial x} g(x,y)$$

$$y \leftarrow y - a * \frac{\partial}{\partial y} g(x,y)$$

其中 a 为参数，其决定了搜索的迭代次数和最后结果的精度。本程序选择 $a = 0.05$。

对于最后结果的精度，本程序通过考察迭代后的 x, y 与前一次的 x, y 值的差值，来计算并控制精度。

本程序的源文件为 /SearchMin/Search_Min.py。

本程序采用 utf-8 格式编码。

2.3 伪代码

Algorithm 2 Search for the minimum of the function

Input : x_{ini}, y_{ini}

Output : z as the minimum of $g(x, y)$

```
function STEEPEST_DESCENT(x, y)
    while not |x_0 − x| < ϵ do
        x_0 ← x                                  ▷ store the previous x
        y_0 ← y                                  ▷ store the previous y
        x ← x − a ∗ ∂/∂x g(x, y)                 ▷ a as a parameter
        y ← y − a ∗ ∂/∂y g(x, y)
    end while
    return x, y
end function
x_ofMin, y_ofMin ← STEEPEST_DESCENT(x_ini, y_ini)
z = g(x_ofMin, y_ofMin)
```

2.4 测试用例

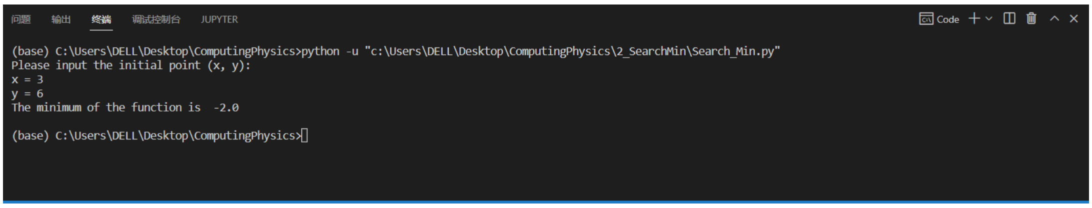

图 5: Search_Min.py

3 题目 3

3.1 题目描述

Temperature dependence of magnetization

Determine $M(T)$ the magnetization as a function of temperature T for simple magnetic materials.

3.2 程序描述

方法 1：直接求解

对于所求的 $M(T)$ 函数，因为有 $m(T) = \frac{M(T)}{N\mu}$，$t = \frac{T}{T_c}$ 和 $T_c = \frac{N\mu^2\lambda}{k_B}$，所以求 $M(T)$ 可以转化为求 $m(t)$ 函数。

m 和 t 有关系式

$$m(t) = \tanh(\frac{m(t)}{t}). \tag{1}$$

显然有 $-1 < m < 1$。且 $m = 0$ 显然为方程的一个解，对 t > 0 均成立。

因为当 $m \neq 0$ 时，有

$$t = \frac{m(t)}{\arctan(m(t))} = f(m(t)). \tag{2}$$

由 (2) 式可知，满足 $m(t) \neq 0$ 的 t 要求 $t < 1$。则对于 $t \geq 1$, 有 $m = 0$，对于 $t < 1$ 的情况，数学解有 $m = 0$，但是若考虑其物理意义，显然解为 $m(t) = f^{-1}(t) \neq 0$ 。

则最终的解为

$$m(t) = \begin{cases} f^{-1}(t) & 0 < t < 1, \\ 0 & t \geq 1 \end{cases} \tag{3}$$

其中 f 满足 $f(x) = \frac{x}{\arctan(x)}$。

方法 2：求根法求解

对于 $t \geq 1$ 的情形, 已经知道了 $m(t) = 0$。对于 $0 < t < 1$ 的情形, 有 $m(t) = \tanh(\frac{m(t)}{t})$, 即

$$f(m) = \tanh(\frac{m(t)}{t}) - m(t).$$

此时即需要求 $f(m)$ 的根，其中 t 为 $0 < t < 1$ 的参数。对不同的 t，求出相应的方程的根 m，最后与 $t \geq 1$ 的情况一起绘制 $m(t)$ 图像即可。

本程序的代码源文件为/TdependenceofM/Solve_and_Plot_version1.py 和/TdependenceofM/Solve_and_Plot_version2.py，/TdependenceofM/FindrootThroughHybrid.py 为 v2.0 版本所用到的求根法的模块。方法 1、2 分别对应 v1.0 和 v2.0 版本。导入的第三方库有 matplotlib 和 numpy。

3.3 伪代码

由于方法 1 比较简单，这里给出方法 2 的伪代码。

Algorithm 3 Determine $M(T)$ the magnetization as a function of temperature T

Output : $m(t)$

for every t from 0 to 1 **do**
    $root_1(t) = \text{HYBRID}(t, -1, 0)$ ▷ t is received as a parameter
    $root_2(t) = \text{HYBRID}(t, 0, 1)$
    two m for one t: $m_1 \leftarrow root_1, m_2 \leftarrow root_2$
end for
for every t from 1 to ∞ **do**
    $m \leftarrow 0$
end for
Plot $m(t)$

3.4 测试用例

方法 1、方法 2 绘制的函数图像相同，这里给出方法 2 绘制的函数图像，如图 6 所示。

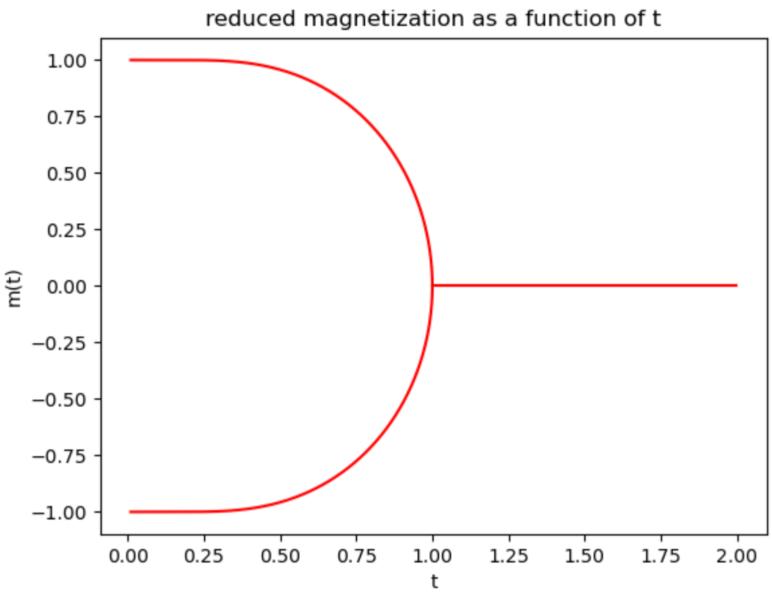

图 6: Function $m(t)$