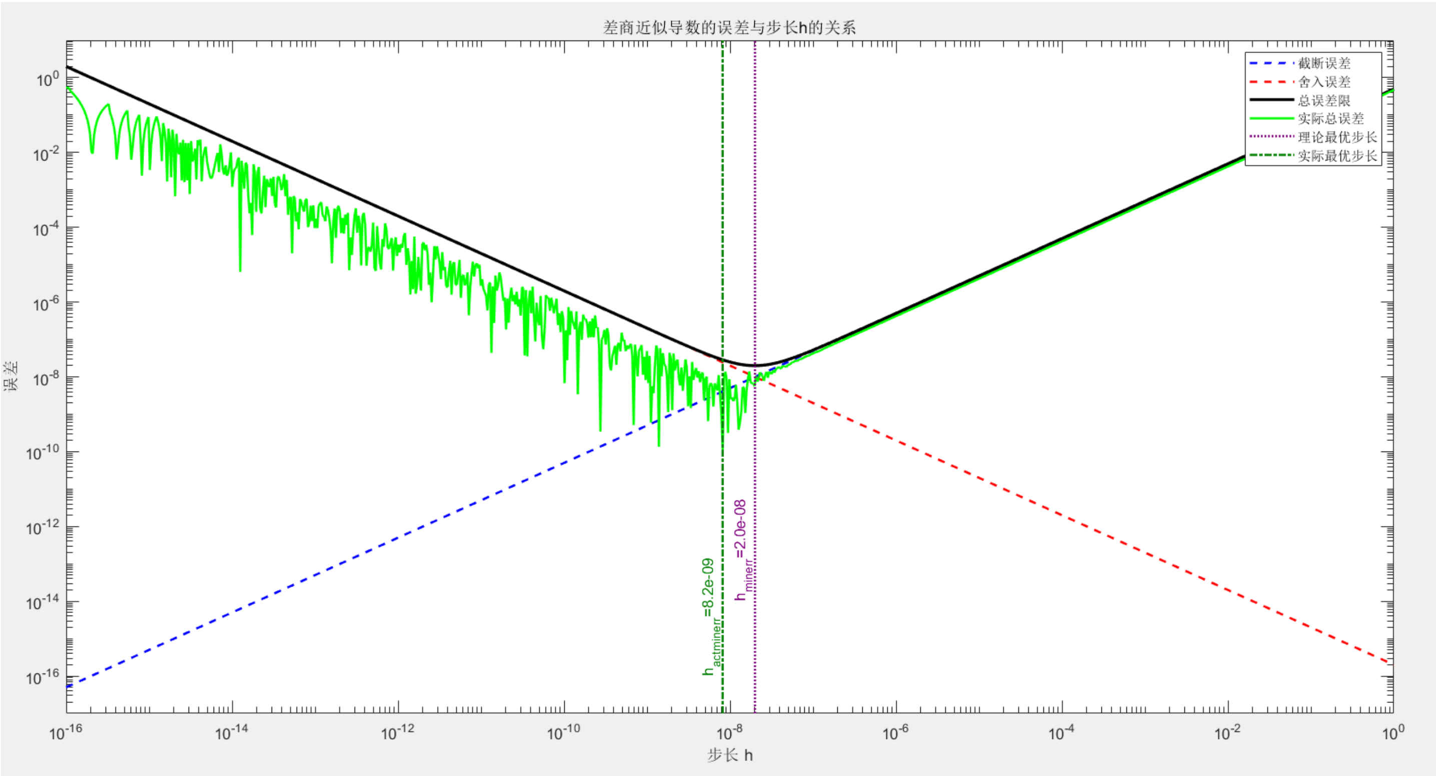
实验一

一、截断误差与舍入误差

第一章上机题1: 编程实现例 1. 4 , 绘出图 1-2 , 体会两种误差对结果的不同影响.



由图可见:

1. 随着步长增大，截断误差增大，这是因为截断误差为 $\frac{h}{2}f^{''}(\xi)$，即 $O(h)$，也可以理解为步长越大，差分公式对于导数的近似就越不精确
2. 随着步长增大，舍入误差减小，这是因为舍入误差为 $\frac{(\delta_2-\delta_1)}{h} \le \frac{2h}{\epsilon}$
3. 根据教材：对于一个给定的问题，有时只有截断误差和舍入误差中的某一个占主导地位。一般地，能在有限步之内求解的纯代数问题中，舍入误差往往占主导地位(如高斯消去法)，而对于涉及积分、求导、非线性这类理论上是无限逼近过程的问题，截断误差往往占主导地位。

二、无穷级数求和

第一章上机题3: 编程观察无穷级数

$$\sum_{n=1}^{\infty}\frac{1}{n}$$

的求和计算.

(1) 采用 IEEE 单精度浮点数, 观察当 n 为何值时, 求和结果不再变化, 将它与理论分析的结论进行比较 (注: 在 MATLAB 中可用 single 命令将变量转成单精度浮点数).

(2) 用 IEEE 双精度浮点数再次计算 (1) 中得到的前 n 项, 评估 IEEE 单精度浮点数计算结果的误差.

(3) 采用 IEEE 双精度浮点数, 请估计当 n 为何值时上述无穷级数求和结果不再变化, 这在当前做实验的计算机上大概需要多长的计算时间? 注意：编程时用简单的for循环，并假设程序顺序运行，也没有任何编译或运行时优化技术。

1. 单精度浮点数

理论估计

- IEEE 单精度浮点数尾数有 23 位，加上隐藏的一位 1 共可以表示最多 24 位二进制有效数字
- 从二进制转换为十进制，大约为 $N_{10}=\dfrac{N_2}{log_2(10)}\approx 7.225$，因此 IEEE 最多可以表示 7 位十进制有效数字
- 结果如下：

```
(1) 单精度求和停止变化的n值: 2097151
    此时的结果为: 15.4037
    单精度求和停止变化的n估计值: ~2000000
```

2. 双精度与单精度的误差

单精度、双精度各算一遍求误差即可，结果如下：

```
(2) 双精度结果为: 15.1333
    绝对误差: 0.27038
    相对误差: 0.017866
```

3. 双精度浮点数极限估计

根据定理 1.6 “(1) 若 $\dfrac{|x_2|}{|x_1|}\le\dfrac{1}{2}\epsilon_{mach}$，则 x_1+x_2 的计算结果一定等于 x_1，即 x_2 对 x_1+x_2 的结果毫无影响。”

于是我们得到**停止条件**：当 $\dfrac{1}{n}\le\dfrac{\epsilon_{mach}\cdot S_n}{2}$ 时，求和结果不再变化。

- 双精度浮点数的机器精度（Matlab 标准）约为 $\epsilon\approx 2.224\times 10^{-16}$
- 根据调和级数的性质，对于有限的 n，调和级数的部分和可以近似为

$$S_n = \sum_{n=1}^{\infty} \frac{1}{n} \approx \ln(n) + \gamma$$

其中 γ 是欧拉-马歇罗尼常数（约为 0.5772）。

代入得到：

$$\frac{1}{n} < \epsilon \cdot (\ln(n) + \gamma)$$

$$\text{即 } n > \frac{1}{\epsilon \cdot \ln(n) + \gamma}$$

由于 $\ln(n)$ 增长缓慢，于是我才用迭代法估计 n 的值。给定一个初值，然后每次放大 2 倍直至超过阈值。

- 结果如下：

```
(3) 双精度求和停止变化的n估计值: ~400000000000000
    预计计算时间: ~0.014377 年
```

最终结果

```
>> lab1_2
(1) 单精度求和停止变化的n值: 2097151
    此时的结果为: 15.4037
    单精度求和停止变化的n估计值: ~2000000
(2) 双精度结果为: 15.1333
    绝对误差: 0.27038
    相对误差: 0.017866
(3) 双精度求和停止变化的n估计值: ~400000000000000
    预计计算时间: ~0.014377 年
```