Problem 1

记 $x=[x_1,x_2,x_3,x_4]^\top=[H^+,OH^-,HCO_3^-,CO_3^{2-}]^\top$,利用如下定义的 f 可以将题目条件重写为 $f(x)=0$ 的形式，

$$f(x)=\begin{bmatrix} x_1x_2-K_W \\ x_1x_3-\frac{K_1K_Hp_{CO_2}}{10^6} \\ \frac{x_1x_4}{x_3}-K_2 \\ x_1-x_2-x_3-2x_4 \end{bmatrix}$$

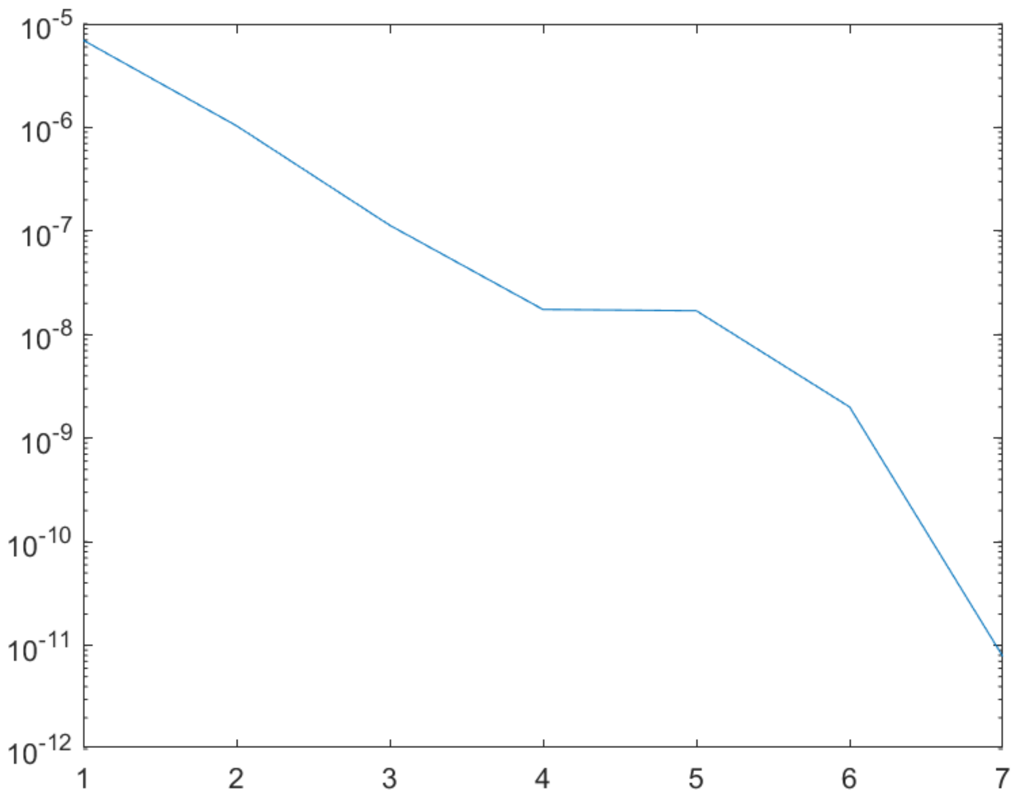
其Jacobi矩阵为，

$$J=\begin{bmatrix} x_2 & x_1 & 0 & 0 \\ x_3 & 0 & x_1 & 0 \\ \frac{x_4}{x_3} & 0 & -\frac{x_1x_4}{x_3^2} & \frac{x_1}{x_3} \\ 1 & -1 & -1 & -2 \end{bmatrix}$$

利用如下的Newton迭代法求解，

$$x_{k+1}=x_k-J^{-1}f(x_k)$$

选取初值为 $10^{-7}\times[1,1,1,1]^\top$, Newton法在7步内收敛，误差对数图如下，



采用 (Good) Broyden方法进行求解，其秩1扰动对应的公式如，

$$
\begin{aligned}
\Delta f_k &= f(x_k) - f(x_{k-1}) \\
\Delta x_k &= x_k - x_{k-1} \\
r_k &= \Delta f_k - J_k \Delta x_k \\
y_k &= \frac{\Delta x_k}{\|\Delta x_k\|^2} \\
J_{k+1} &= J_k + r_k y_k^\top
\end{aligned}
$$

如果采用直接求逆，

$$
x_{k+1} = x_k - J_k^{-1} f(x_k)
$$

利用Shermann–Morrison–Woodbury公式， 基础初始Jacobi矩阵的逆 $H_0 = J_0^{-1}$，每次更新扰动的矩阵 U_k, V_k

$$
\begin{aligned}
U_k &= [U_{k-1}, r_k] \\
V_k &= [V_{k-1}, y_k] \\
H_{k+1} &= H_0 - H_0 U_k (I_k + V_k^\top H_0 U_k)^{-1} V_k H_0
\end{aligned}
$$

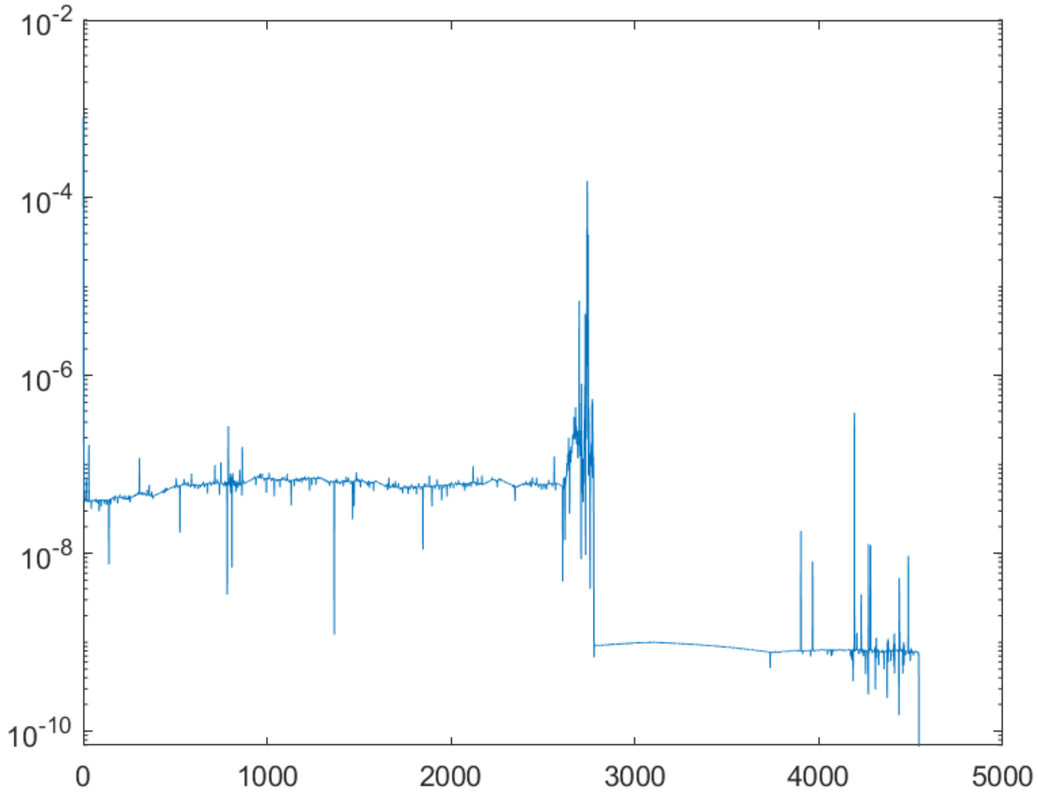
此时更新公式为，

$$
x_{k+1} = x_k - H_k f(x_k)
$$

但是对于该问题，由于 $n = 4$, 采用秩 k 修正是更贵的方法，考虑采用秩 1 修正，

$$
\begin{aligned}
H_{k+1} &= H_k - H_k r_k (1 + y_k^\top H_k r_k)^{-1} y_k^\top H_k \\
&= H_k - \frac{H_k(\Delta f_k - J_k \Delta x_k)\Delta x_k^\top H_k}{\Delta x_k^\top \Delta x_k + \Delta x_k^\top H_k(\Delta f_k - J_k \Delta x_k)} \\
&= H_k - \frac{(H_k \Delta f_k - \Delta x_k)\Delta x_k^\top H_k}{\Delta x_k^\top H_k \Delta f_k}
\end{aligned}
$$

但此时Broyden方法的收敛比起Newton方法要慢的多，但Broyden方法作为一阶算法，在复杂度上优于Newton方法，其对数误差图如下所示，可以看到其算法也并不是特别稳定，当然可能也与该问题本身较为病态有关，



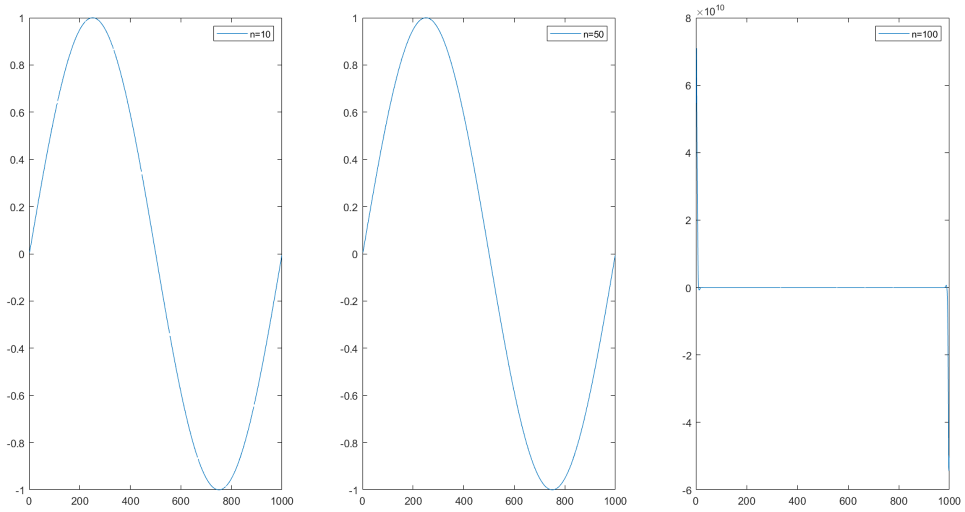
Problem 2

利用Lagrange插值多项式,

$$f(x)=\sum_{k=0}^{n} y_k \frac{\prod_{j\neq k}(x-x_j)}{\prod_{j\neq k}(x_k-x_j)}$$

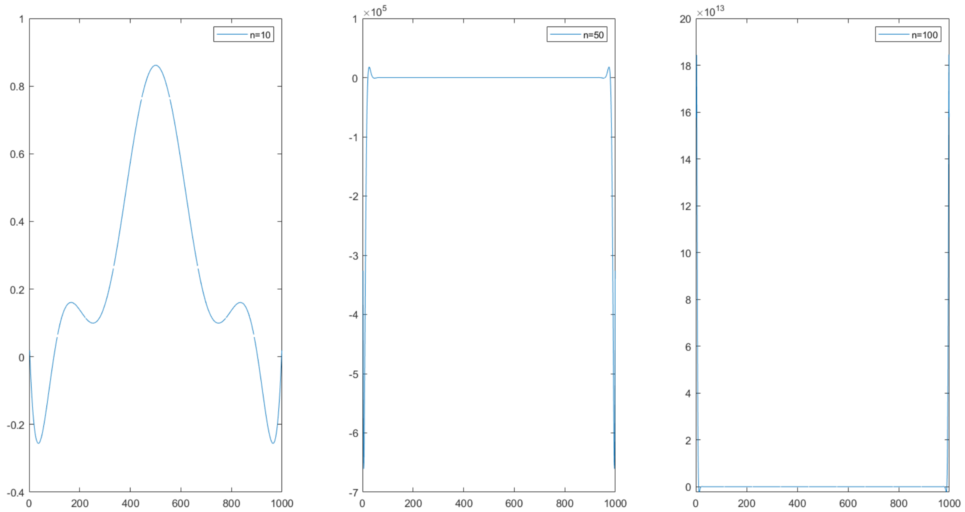
调用函数 `interpolation_p2(n,f,a,b)` 对函数 f 在区间 $[a,b]$ 内进行插值，取 $n=[10,50,100]$ ，

对于函数 $f(x)=\sin(x),x\in[0,2\pi]$ ，其误差如下图，



可以看到当 $n=100$ 的时候产生了龙格现象，或者此时由于多项式次数过高反而出现了过拟合，下面对于龙格函数探究该现象，

对于函数 $f(x)=\frac{1}{1+25x^2}, x\in[0,1]$ ，可以发现对于龙格函数，插值点越多反而导致插值误差越大，



而如果采用MATLAB库函数进行插值，由于其采用的是样条插值，可以发现并不会出现上述现象。

Problem 3

利用Newton插值公式，递推地定义插值多项式

$$p_{1,2,\dots,n+1}(x)=p_{1,2,\dots,n}(x)+\alpha_n\prod_{j=1}^{n}(x-x_j)$$

其中 α_n 为 n 阶差分，

$$\alpha_n=f[x_1,x_2,\dots,x_n]$$

利用 n 阶差分的定义可以知道其满足 $\alpha_n n! \in \mathbb{Z}$,

下面利用归纳法证明 $f(k) \in \mathbb{Z}$, 首先当 $n=1$ 时显然成立，下面假设对 $p_{1,2,..n}(x)$ 成立该条件，则对于 $p_{1,2,...n+1}(x)$,

由于当 $i \leq i \leq i+n$ 的时候显然成立该结果，若 $k > i+n$, 如下可以归纳下去,

$$\frac{\prod_{j=1}^{n-1}(k-x_j)}{n!} = \frac{(k-i)!}{n!(k-i-n-1)!} = (k-i-n)\binom{k-i}{n-1} \in \mathbb{Z}$$

若 $k < i$,

$$|\frac{\prod_{j=1}^{n-1}(k-x_j)}{n!}| = \frac{(i+n-k)!}{n!(i-k-1)!} = (i-k)\binom{i+n-k}{n} \in \mathbb{Z}$$

因此归纳可以进行下去，从而证明了 $f(k) \in \mathbb{Z}, \forall k \in \mathbb{Z}$.

但满足该条件的函数未必具有整数系数，例如对于函数 $f(x) = \frac{1}{2}x^2 + \frac{1}{2}x$, 满足

$$f(0) = 0, f(1) = 1, f(2) = 3$$