

Python 代码 (带 C 加速):

```
import numpy as np
import copy
from numba import jit

@jit(nopython=True)
def sor(A, b, w, x, xT, limit):
    # # sor 迭代
    n = A.shape[1]
    iter1 = 0
    while (np.max(np.abs(x - xT)) >= limit):
        for i in range(n):
            sum = 0
            for j in range(n):
                if j != i:
                    sum = sum + A[i][j] * x[j]
            x[i] = (1-w) * x[i] + w * (b[i] - sum)/A[i][i]
        iter1 = iter1 + 1
    return x, iter1

def dis(x):
    return np.max(abs(x))

if __name__ == '__main__':
    A = np.array([[4, -1, 0], [-1, 4, -1], [0, -1, 4]])
    b = np.array([1, 4, -3])
    x0 = np.array([0, 0, 0])
    xT = np.array([1/2, 1, -1/2])
    x = copy.deepcopy(x0).astype(np.float64)
    limit = 5e-6
    w = 1.03
    print("A:\n", A, '\n')
    print('b:\n', b, '\n')
    result, iter = sor(A, b, w, x, xT, limit)
    print('result:\n', result, '\n')
    print('iter:\n', iter, '\n')
```

(一) $w=1$

A:

$\begin{bmatrix} 4 & -1 & 0 \end{bmatrix}$

$\begin{bmatrix} -1 & 4 & -1 \end{bmatrix}$

$\begin{bmatrix} 0 & -1 & 4 \end{bmatrix}$

b:

$\begin{bmatrix} 1 & 4 & -3 \end{bmatrix}$

result:

$\begin{bmatrix} 0.50000381 & 1.00000191 & -0.49999952 \end{bmatrix}$

iter:

6

(二) $w=1.03$

A:

$\begin{bmatrix} 4 & -1 & 0 \end{bmatrix}$

$\begin{bmatrix} -1 & 4 & -1 \end{bmatrix}$

$\begin{bmatrix} 0 & -1 & 4 \end{bmatrix}$

b:

$\begin{bmatrix} 1 & 4 & -3 \end{bmatrix}$

result:

$\begin{bmatrix} 0.50000447 & 1.00000161 & -0.49999974 \end{bmatrix}$

iter:

5

(三) $w=1.1$

A:

$\begin{bmatrix} 4 & -1 & 0 \end{bmatrix}$

$\begin{bmatrix} -1 & 4 & -1 \end{bmatrix}$

$\begin{bmatrix} 0 & -1 & 4 \end{bmatrix}$

b:

$\begin{bmatrix} 1 & 4 & -3 \end{bmatrix}$

result:

$\begin{bmatrix} 0.50000363 & 0.99999854 & -0.50000004 \end{bmatrix}$

iter:

6