Parallel Computing: Homework #1

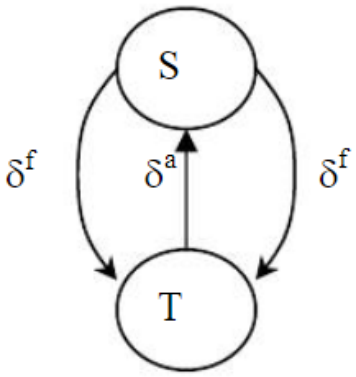
Professor QiLong Zheng

Xin Sun **Automation** **SA22010050**

ex-1

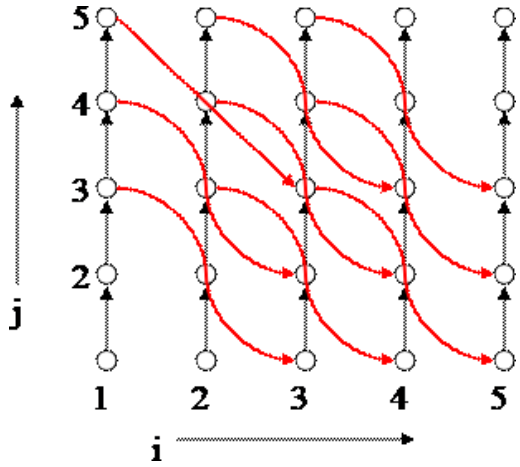
Problem 1

S: A(I) = C(I) + 2;
T: B(I) = A(I-1) - A(2*I - 5);
(1) S δ^f T ; $\{< S(i), T(j) > | i = j - 1; 1 <= j <= 100, 0 <= i <= 99\}$
(2) S δ^f T ; $\{< S(i), T(j) > | (i, j) = (1, 3), (3, 4), (5, 5)\}$
(3) T δ^f S ; $\{< S(i), T(j) > | j = 2i - 5; 6 <= i <= 52\}$



Problem 2

S: A(I+2,J) = B(2*I,J) - 5;
T: B(2*I,J-1) = A(I,J+2) + 4;
S δ^fT，距离向量 (2,-2)，方向向量 (1,-1);
S δ^aT，距离向量 (0,1)，方向向量 (0,1)



Problem 3

- (1)

  S: A(1:N) = B(1:N) + C(2:N+1);

  T: C(1:N) = A(1:N) * D(1:N);

- (2)

  不能向量化，S=T 且方向向量为 (1)

Problem 4

```
1 for I = 1 to 5 do
2 S: B(I) = B(I) / A(I,I);
3     for J = I+1 to 5 do
4 T:      B(J) = B(J) - A(I,J) * B(I);
5     end for
6 end for
```

依赖关系：

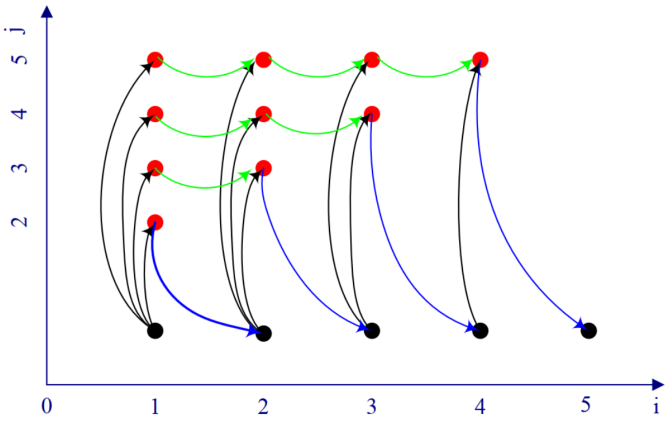
S δ^f T; T δ^f T; T δ^f S;

展开循环：

```
1 I = 1 : B(1) = B(1) / A(1,1);
2     J = 2, 5
3     J = 2 : B(2) = B(2) - A(1,2) * B(1)
4     J = 3 : B(3) = B(3) - A(1,2) * B(1)
5     J = 4 : B(4) = B(4) - A(1,2) * B(1)
6     J = 5 : B(5) = B(5) - A(1,2) * B(1)
7 I = 2 : B(2) = B(2) / A(2,2);
8     J = 3, 5
9     J = 3 : B(3) = B(3) - A(2,3) * B(2)
```

Problem 4 continued on next page...

```
    J = 4 : B(4) = B(4) − A(2,4) ∗ B(2)
    J = 5 : B(5) = B(5) − A(2,5) ∗ B(2)
I = 3 : B(3) = B(3) / A(3,3);
    J = 4, 5
    J = 4 : B(4) = B(4) − A(3,4) ∗ B(3)
    J = 5 : B(5) = B(5) − A(3,5) ∗ B(3)
I = 4 : B(4) = B(4) / A(4,4);
    J = 5, 5
    J = 5 : B(5) = B(5) − A(4,5) ∗ B(4)
I = 5 : B(5) = B(5) / A(5,5);
    J = 6, 5
    No Loop for J
```



ex-2

Problem 5

```
for  I = 1 to 100 do
    for  J = 1 to 100 do
S:              A(I,J) = B(I+4,J−2) − B(I−2,J+1) + B(I,J+3);
T:              B(I,J) = D(I,J−1) − C(I+2,J)
    endfor
endfor
```

Tδ^fS, 距离向量 (2,-1), 方向向量 (1,-1); 迭代对 $\{< S(i_1,j_1),T(i_2,j_2) > | i_1 = i_2 - 2, j_1 = j_2 + 1, 1 <= i_1 <= 98, 2 <= j_1 <= 100\}$
Sδ^aT, 距离向量 (4,-2), 方向向量 (1,-1); 迭代对 $\{< S(i_1,j_1),T(i_2,j_2) > | i_1 = i_2 - 4, j_1 = j_2 + 2, 1 <= i_1 <= 96, 3 <= j_1 <= 100\}$
Sδ^aT, 距离向量 (0,3), 方向向量 (0,1) 迭代对 $\{< S(i_1,j_1),T(i_2,j_2) > | i_1 = i_2, j_1 = j_2 - 3, 1 <= i_1 <= 100, 1 <= j_1 <= 97\}$

Problem 6

循环 2 存在依赖关系：Sδ^fS，距离向量为 (1，-1)，方向向量为 (1，-1)

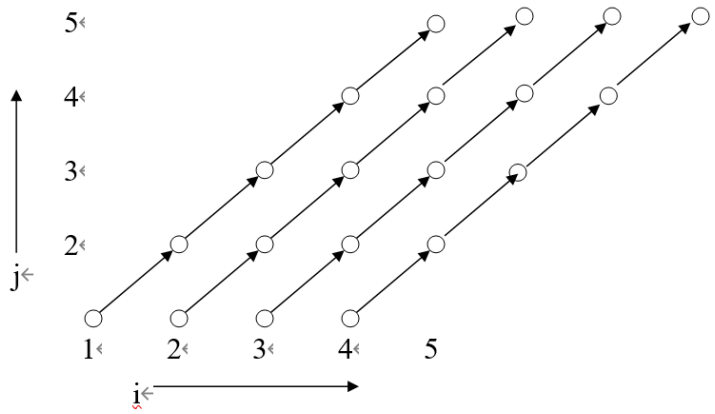
- (1) 循环 2 和循环 3 不等价，2 是流依赖，3 是反依赖
- (2) 循环 2 和循环 4 等价，不存在 (0，1) 的依赖关系，所以内层循环可以并行。
- (3) 循环 2 和循环 5 不等价，存在 (1，*) 的依赖关系，外层循环不能并行化。

Problem 7

1.

```
1  for I = 1 to 8 do
2      for J = max(I−3,1) to min(I,5) do
3  S:                A(I+1, J+1) = A(I,J) + B(I,J)
4      endfor
5  endfor
```

S δ^f S, 距离向量 (1，1)，方向向量 (1，1)



2.

```
1  for I = 2 to 9 do
2  U:  if A(I) > 0 then
3  S:                A(I) = B(I−1) + 1
4      else
5  T:                B(I) = A(I) * 2
6      endif
7  endfor
```

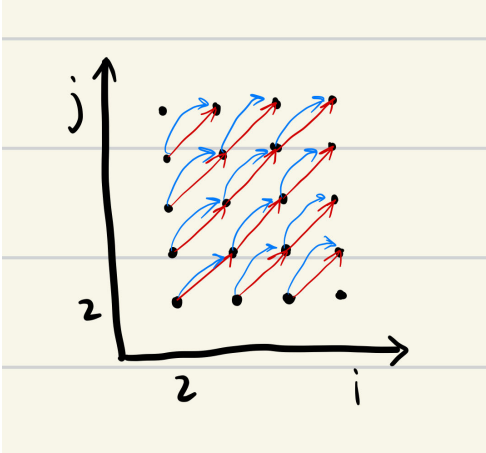
A[i] $<$ 0 且 A[i+1] $>$ 0, 存在 Tδ^fS。

A[i] $>$ 0: U δ^a S

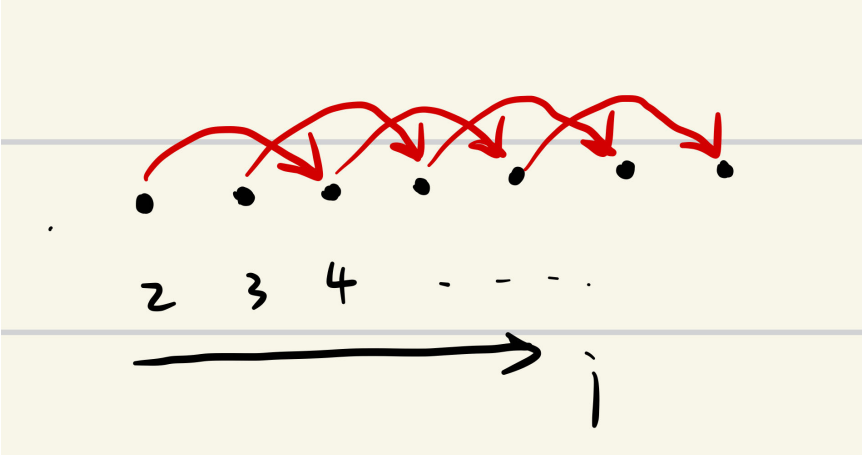
ex-3

Problem 8

1. S δ^f S, S δ^a S.



2. S δ^f S 方向向量 (2),

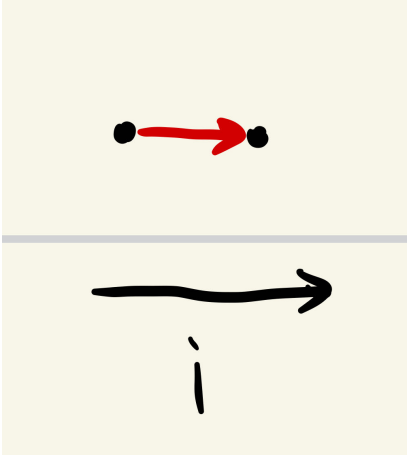


3.

```
1  for  i = 2 to 20 do // 循环 3
2       if  A[i] > 0 then
3  S:       B[i] = C[i−1] + 1
4       else
5  T:       C[i] = B[i] − 1
6       endif
7  endfor
```

A[i-1]<0 并且 A[i]>0, 此时，T δ^f S

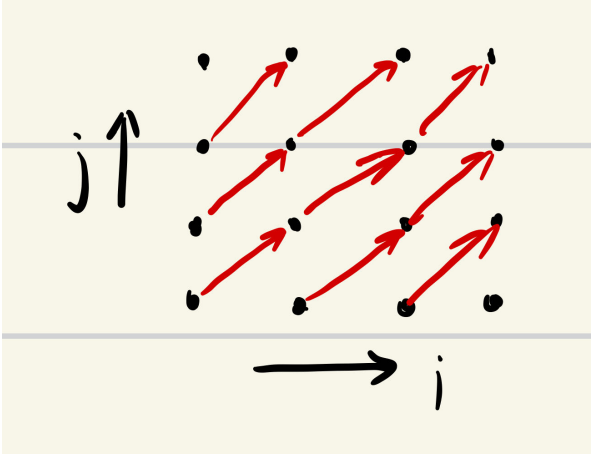
此时，迭代依赖图为



Problem 9

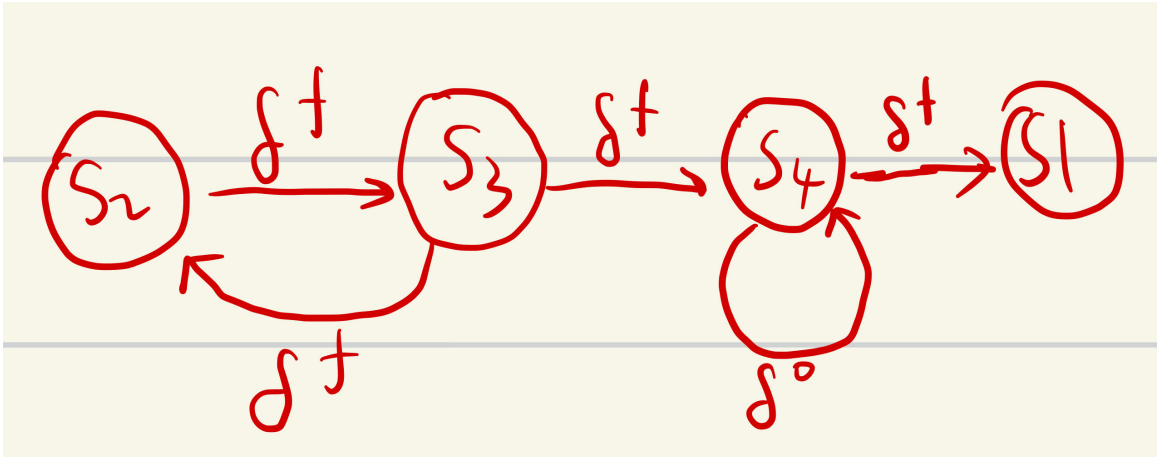
1.

迭代依赖图：

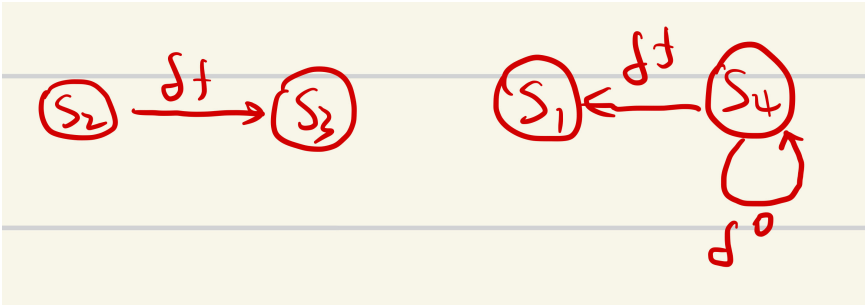


不能逆转外层循环，逆转后方向向量变为 (-1，1)，存在负值，所以不能逆转。能交换内外循环次序，方向向量是 (1，1)，与置换矩阵相乘之后大于 0.

2. 如果 N<=100, 迭代依赖图：



如果 N>100, 迭代依赖图:



N<=100, 最内层 S3 可向量化或者并行化:

```
1 for  i  = 1 to 100 do // 循环 2 N 是常量
2     X[i] = Y[i] + 10; // 语句 S1
3     for  j  = 1 to 100 do
4         B[j] = A[j, N]; // 语句 S2
5         A[j+1, 1:100] = B[j] + C[j, 1:100] //S3 或者并行化
6         Y[i+j] = A[j+1, N]; // 语句 S4
7     endfor // loop-j
8 endfor // loop-i
```

N>100, 里面两层循环均可向量化。

```
1 for  i  = 1 to 100 do // 循环 2 N 是常量
2     X[i] = Y[i] + 10; // 语句 S1
3     B[1:100] = A[1:100, N]; // 语句 S2
4     A[2:101, 1:100] = B[1:100] + C[1:100, 1:100] //S3 B自动broadcast
5     Y[i+1: i+100] = A[2:101, N]; // 语句 S4
6 endfor // loop-i
```

Problem 10

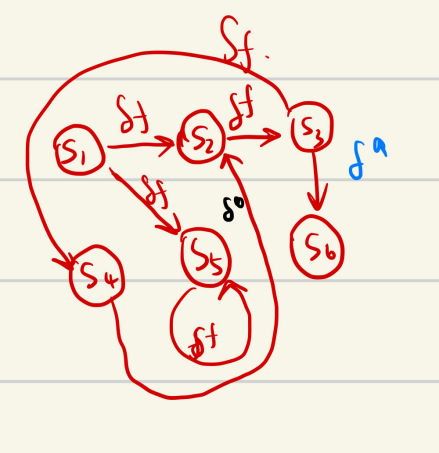
1. (a) S(i=31, j=18) δ^f S(i=32, j=19)

   (b) S(i=11, j=11)δ^fS(i=18, j=7)

   (c) 不能向量化，例如 S(i=36，j=20)δ^f S(i=36，j=22), 方向向量为 (0，1)，所以不能并行化。

2.

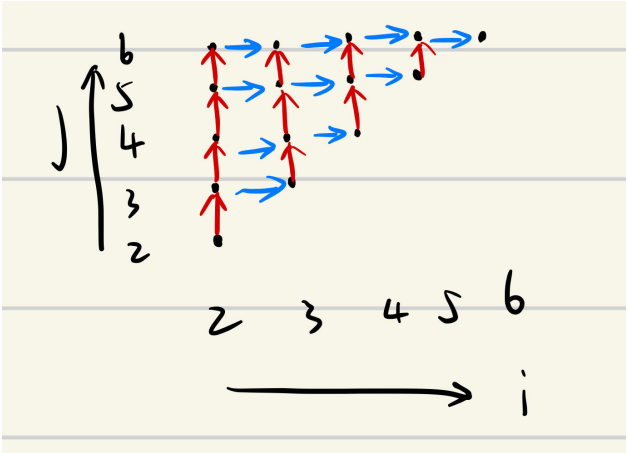
   语句依赖图



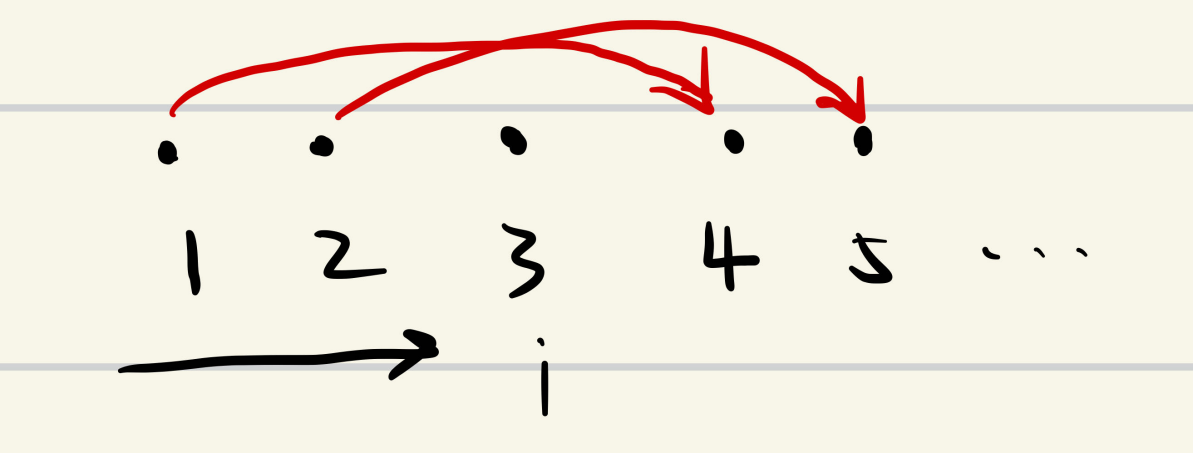
Problem 11

1. S δ^f S , 依赖向量为 (0，1) S δ^f S, 依赖向量为 (1，0)

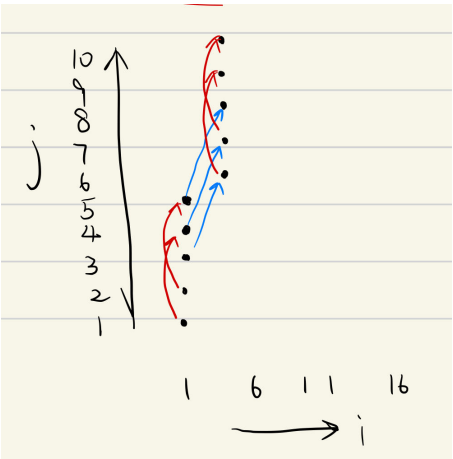
   迭代依赖图



2. Sδ^fS , 依赖距离向量为 (3)，依赖方向向量为 (1)



3. S δ^f S 迭代依赖图



Problem 12

1.

```
1 for i = 1 to 100 do //循环 1
2 S: A[i] = A[i] + B[i−1];
3 T: B[i] = C[i−1] ∗ 2 ;
4 U: C[i] = 1 / B[i] ;
5 V: D[i] = C[i] ∗ C[i] ;
6 endfor
```

U δ^f T(c), T δ^f U(B), T δ^f S(B), U δ^f V(C),

```
1 L1:
2 for i = 1 to 100 do
3 T: B[i] = C[i−1] ∗ 2 ;
4 U: C[i] = 1 / B[i] ; //顺序执行
5 endfor
```

```
L2:
doall  i  = 1 to 100 do //循环 1
S:   A[i] = A[i] + B[i−1];
V:   D[i] = C[i] * C[i] ;
enddoall //并行化
```

2.

```
for  i  = 1 to 999 do // 循环 2
S:   A[i] = B[i] + C[i];
T:   D[i] = ( A[i] + A[ 999−i+1 ] ) / 2 ;
endfor
```

S δ^f T , T δ^a S;

```
L1:
doall  i  = 1 to 499  //
     A[i] = B[i] + C[i];
     D[i] = A[i] + A[999−i+1]
enddoall //并行化
L2:
doall  i  = 500 to 999
     A[i] = B[i] + C[i];
     D[i] = A[i] + A[999−i+1] //并行化
enddoall
```

3.

```
for  i  = 1 to 100 do // 循环 3
     for  j  = 1 to 100 do
S:        A[3*i+2*j, 2*j] = C[i,j] * 2 ;
T:        D[i,j] = A[i−j+6, i+j] ;
     endfor
endfor
```

没有依赖关系，可以并行执行。

```
 doall  i  = 1 to 100  // 循环 3
     doall  j  = 1 to 100
S:        A[3*i+2*j, 2*j] = C[i,j] * 2 ;
T:        D[i,j] = A[i−j+6, i+j] ;
     enddoall
enddoall
```