Numerical Heat Transfer
(数值传热学)
Chapter 11 Application Examples of the General Code for 2D Elliptical FF & HT Problems

Instructor：Fang, Wen-Zhen; Tao, Wen-Quan
Email: fangwenzhen@xjtu.edu.cn

Key Laboratory of Thermo-Fluid Science & Engineering
Xi'an Jiaotong University
2023-Nov-22

11.1 2D steady heat conduction without source term in Cartesian coordinate

11.2 Steady heat conduction in a hollow cylinder

11.3 Fully-developed heat transfer in a square duct

11.4 Fully developed heat transfer in annular space with straight fin at inner wall

Conduction

11.5 Fluid flow and heat transfer in a 2-D sudden expansion

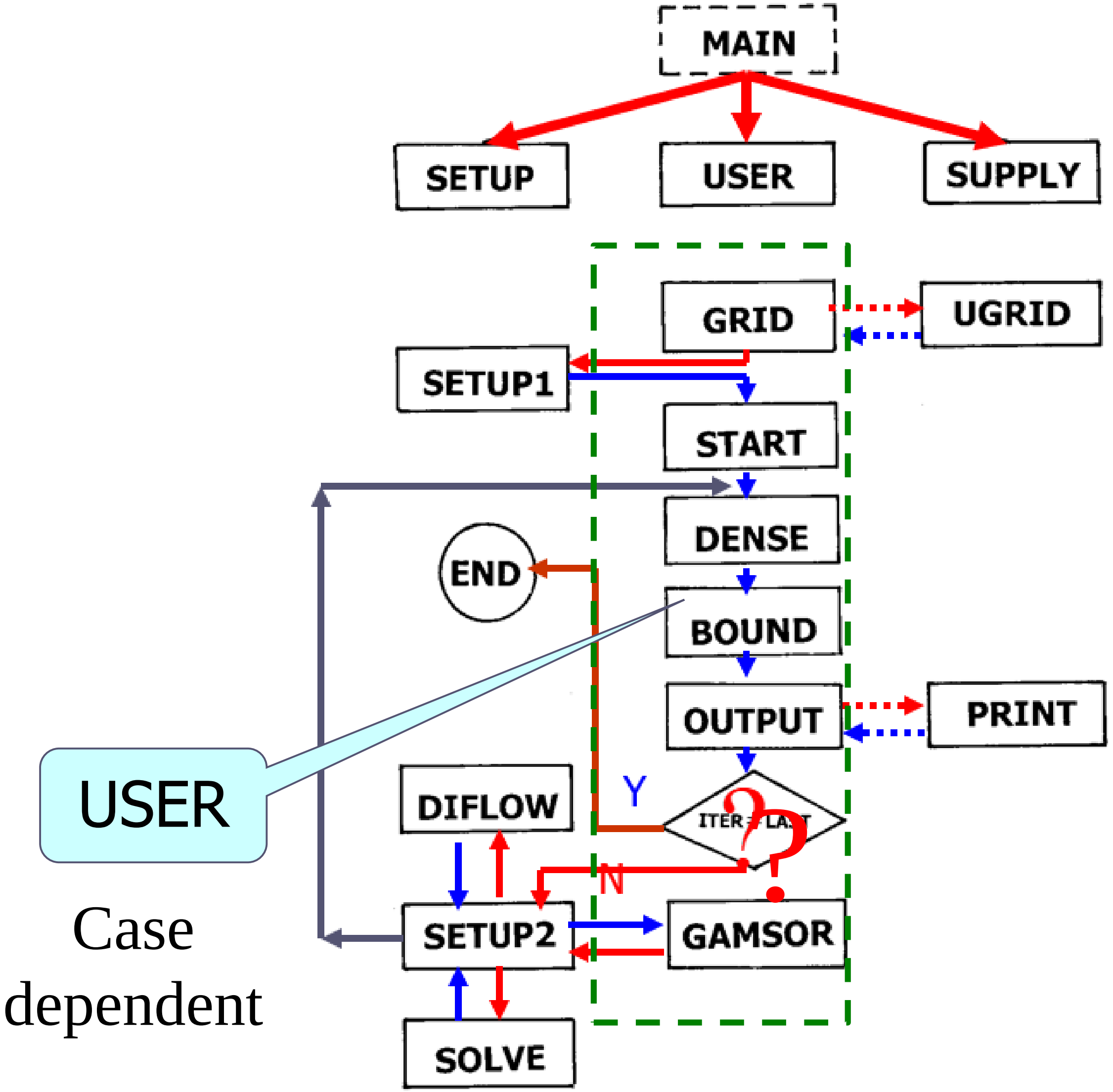
11.6 Complicated fully developed fluid flow and heat transfer in square duct

11.7 Impinging flow on a rotating disc

11.8 Turbulent flow and heat transfer in duct with a central jet

Convective

Review of Teaching Code



Teaching Code

Main Program | USER

Black box, no need to change

Varied to deal with a **specific** problem

USER: Includes modules GRID, START, DENSE, BOUND, OUTPUT, GAMSOR

Where to call six modules of USER in the main program?

```
C-----------------------------MAIN -------------------------------------------------
C*******************************************************************
    PROGRAM MAIN
    USE START_L
    IMPLICIT NONE
C*******************************************************************
    OPEN(8,FILE='RESULT.txt')
    CALL GRID       ! Grid generation (USER part)
    CALL SETUP1     ! Set up 1-D array (MAIN part)
    CALL START      ! Set up initial fields (USER part)
    DO WHILE (.NOT.LSTOP）
    CALL DENSE      ! Set up fluid density (USER part)
    CALL BOUND      ! Set up boundary condition (USER part)
    CALL OUTPUT     ! Print out (USER part)
    CALL SETUP2     ! Set coefficients and solve ABEqs. (MAIN part)
                              ! Call GAMSOR (USER part)
    ENDDO
    CALL OUTPUT
    CLOSE(8)
    STOP
    END
```

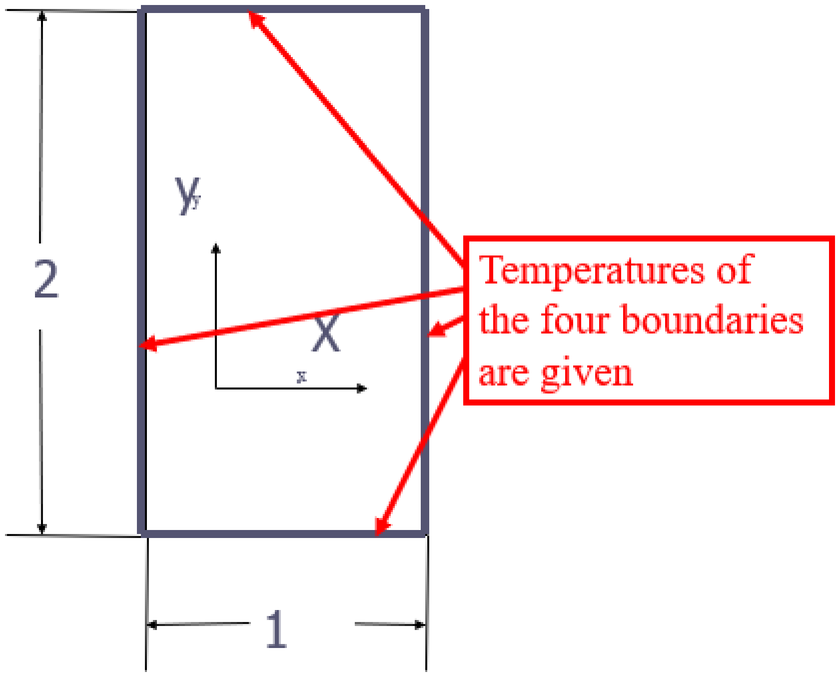
11-1 2D steady heat conduction without source term in Cartesian coordinate －Knowing USER structure

11-1-1 Physical problem and its math formulation

Known: Steady heat conduction of constant properties without source term shown in Fig. 1 has following temperature distribution on four boundaries：

$$T = x + y + xy$$

Find: Temp. distribution within the region.



Remarks: In all examples, physical quantities are given by their numerical values without units. It is assumed that all units are homogeneous (单位和谐).

Solution: GGE $\cancel{\frac{\partial(\rho^* \Phi)}{\partial t}} + \cancel{div(\rho^* \vec{u}\Phi)} = div(\Gamma_\Phi grad\Phi) + \cancel{S_\phi^*}$

2D, steady state, conduction

constant property, no source term

Laplace equation: $\boxed{\operatorname{div}\left(\Gamma_\Phi \operatorname{grad} \Phi\right) = 0}$

Compared with the standard form, it is a diffusion problem with **Γ and source term** as follows:

$$\Gamma_\phi = \lambda = 1, S_\phi^* = 0$$

Boundary conditions: $\boldsymbol{T = x + y + xy}$ at four boundaries

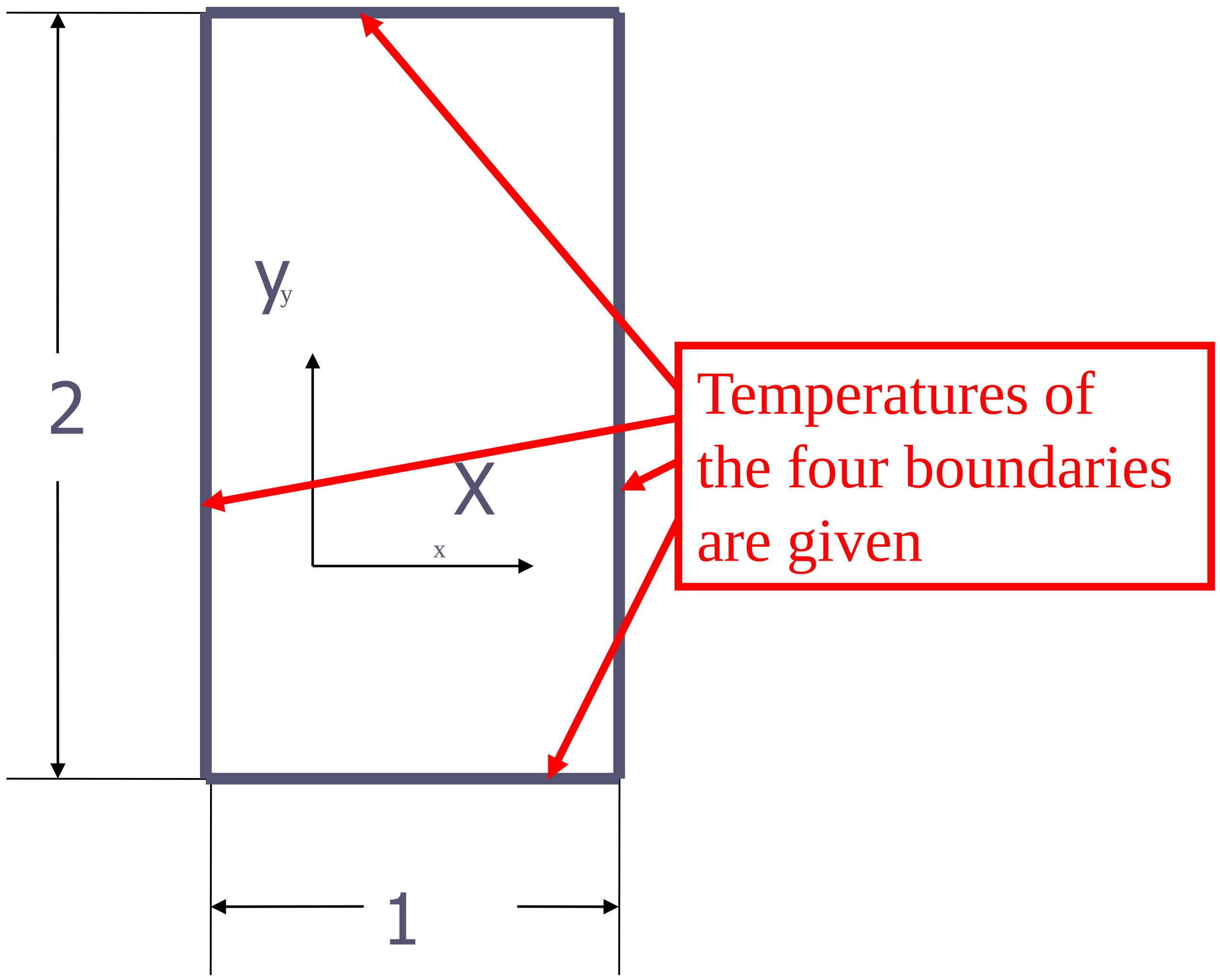


Fig.1 Computational domain

Math formulation:

GE: $div(\Gamma_{\Phi} grad\phi) = 0$

$\Gamma_{\phi} = \lambda = 1, S_{\phi}^{*} = 0$

BC: $\boldsymbol{T = x + y + xy}$

How to get temperature distribution?

Totally, we need < 60 statements in USER. Self-coding is not difficult !

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      MODULE USER_L
C*********************************************************************
       INTEGER*4 I,J
C*********************************************************************
      END MODULE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      SUBROUTINE USER
C**********************************************************************
      USE START_L
      USE USER_L
      IMPLICIT NONE

      ENTRY GRID
      LAST=500
      LSOLVE(4)=.TRUE.
      LPRINT(4)=.TRUE.
      TITLE(4)=' TEMP '
      XL=1.
      YL=2.
      L1=27
      M1=27
      CALL UGRID
      RETURN
```

Module GRID

```
      ENTRY START
      DO 100 J=1,M1
      DO 101 I=1,L1
      T(I,J)=0.
      IF(I==1.OR.I==L1) T(I,J)=(X(I)+Y(J)+X(I)*Y(J))
      IF(J==1.OR.J==M1) T(I,J)=(X(I)+Y(J)+X(I)*Y(J))
101   ENDDO
100   ENDDO
      RETURN
```

Module START

```
      ENTRY DENSE
      RETURN
```

Module DENSE

```
*
      ENTRY BOUND
      RETURN
```

Module BOUND

```
*
      ENTRY OUTPUT
      IF(ITER==0) THEN
      PRINT 401
      WRITE(8,401)
401   FORMAT(1X,'  ITER',13X,'T(4,4)',14X,'T(5,3)')
      ELSE
      PRINT 403, ITER,T(4,4),T(5,3)
      WRITE(8,403) ITER,T(4,4),T(5,3)
403   FORMAT(1X,I5,2F20.6)
      ENDIF
      IF(ITER==LAST) CALL PRINT
      RETURN
```

Module OUTPUT

```
*
      ENTRY GAMSOR
      IF(ITER==0) THEN
      DO 500 J=1,M1
      DO 501 I=1,L1
      GAM(I,J)=1.
501   ENDDO
500   ENDDO
      ELSE
      ENDIF
      RETURN
      END
```

Module GAMSOR

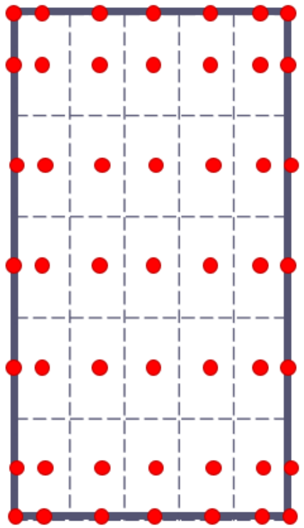
11-1-2 Program reading

Define new variables

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      MODULE USER_L
C*****************************************
      INTEGER*4 I,J
C*****************************************
      END MODULE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE USER
C*****************************************
      USE START_L
      USE USER_L
      IMPLICIT NONE
C*****************************************
C-----------------------------------PROBLEM ONE-----------------------------------
C                    Example of USER structure
C*****************************************************************
      ENTRY GRID
        LAST=10                   ! Numbers of iteration
        LSOLVE(4)=.TRUE.          ! 4th variable for solving temperature equation
        LPRINT(4)=.TRUE.          ! Print out the temperature filed
        TITLE(4)='  .TEMP.  '     ! Title for output temperature field.
```

```
MODULE START_L
 PARAMETER(NI=100,NJ=200,NIJ=NI,NFMAX=10,NFX4=NFMAX+4)
***********************************************************************
CHARACTER*8 TITLE(NFX4)
LOGICAL LSOLVE(NFX4),LPRINT(NFX4),LBLK(NFX4),LSTOP
REAL*8,DIMENSION(NI,NJ,NFX4)::F
REAL*8,DIMENSION(NI,NJ,6)::COF,COFU,COFV,COFP
REAL*8,DIMENSION(NI,NJ)::P,RHO,GAM,CP,CON,AIP,AIM,AJP,AJM,AP
REAL*8,DIMENSION(NI,NJ)::U,V,PC,T,DU,DV,UHAT,VHAT
REAL*8,DIMENSION(NI)::X,XU,XDIF,XCV,XCVS,XCVI,XCVIP
REAL*8,DIMENSION(NJ)::Y,YV,YDIF,YCV,YCVS,YCVR,YCVRS,ARX,ARXJ,
1    ARXJP,R,RMN,SX,SXMN
REAL*8,DIMENSION(NI)::FV,FVP,FX,FXM
REAL*8,DIMENSION(NJ)::FY,FYM
REAL*8,DIMENSION(NIJ)::PT,QT
REAL*8 RELAX(NFX4),TIME,DT,XL,YL,RHOCON,CPCON
INTEGER*4 NF,NP,NRHO,NGAM,NCP,L1,L2,L3,M1,M2,M3,
1    IST,JST,ITER,LAST,MODE,NTIMES(NFX4),IPREF,JPREF
REAL*8 SMAX,SSUM
REAL*8 FLOW,DIFF,ACOF
```

Module GRID



```
        XL=1.   ! Computation domain
        YL=2.                   ! MODE=1 is a default (Cartesian)
        L1=7    ! Grid number
        M1=7
        CALL UGRID   ! Generate interface position of CV
        RETURN
```

Module START

```
        ENTRY START
        DO 100 J=1,M1
        DO 101 I=1,L1
        T(I,J)=0.            ! Initial temperature values.
        IF(I==1.OR.I== L1)  T(I,J)=(X(I)+Y(J)+X(I)*Y(J))  ! Unchanged B.C.
        IF(J==1.OR.J== M1) T(I,J)=(X(I)+Y(J)+X(I)*Y(J))  are given here
  101 ENDDO
  100 ENDDO
        RETURN
*
```

$T = x + y + xy$

Module DENSE

```
        ENTRY DENSE   ! Empty, but keep it
        RETURN
*
```

Module BOUND

```
        ENTRY BOUND   ! Empty, B.C. has been set up in START
        RETURN
```

```
PROGRAM MAIN
USE START_L
IMPLICIT NONE
*****************************
OPEN(8,FILE='RESULT.TXT')
CALL GRID
CALL SETUP1
CALL START
DO WHILE(.NOT.LSTOP)
CALL DENSE
CALL BOUND
CALL OUTPUT
CALL SETUP2
ENDDO
```

```
PROGRAM MAIN
USE START_L
IMPLICIT NONE
*****************************
OPEN(8,FILE='RESULT.TXT')
```

Module OUTPUT

```
      ENTRY OUTPUT
      IF(ITER == 0) THEN          ! The title needs output once only
      PRINT 401         ! Output to screen
      WRITE(8,401)      ! Output through file
401 FORMAT(1X,'  ITER',13X,'T(4,4)',14X,'T(5,3)')   ! ITER        T(4,4)          T(5,3)
      ELSE
      PRINT 403,    ITER,  T(4,4), T(5,3)      ! Print out two temps. in each
      WRITE(8,403) ITER,  T(4,4), T(5,3)       iteration for observation
403 FORMAT(1X,I5,2F20.6)
      ENDIF
      IF(ITER == LAST) CALL PRINT          ! Output 2D field after
      RETURN                               getting converged solution.
*
```

Module GAMSOR

```
      ENTRY GAMSOR
      IF(ITER == 0) THEN       ! constant thermo-properties, call once only
      DO 500 J=1,M1
      DO 501 I=1,L1
      GAM(I,J)=1.
501 ENDDO
500 ENDDO
       ELSE
       ENDIF
      RETURN
      END
```

! The zero initial values of S_c, S_p have been set in "RESET". Only Γ is set up here.

11-1-3 Analysis of results

COMPUTATION IN CARTESIAN COORDINATES
****

ITER	T(4,4)	T(5,3)
0	0.000000	0.000000
1	1.999978	1.720364
2	2.000000	1.720001
3	2.000000	1.720000
4	2.000000	1.720000
5	2.000000	1.720000
6	2.000000	1.720000
7	2.000000	1.720000
8	2.000000	1.720000
9	2.000000	1.720000
10	2.000000	1.720000

```
      ENTRY OUTPUT
      IF(ITER==0) THEN
      PRINT 401
      WRITE(8,401)
401   FORMAT(1X,'  ITER',13X,'T(4,4)',14X,'T(5,3)')
      ELSE
      PRINT 403, ITER,T(4,4),T(5,3)
      WRITE(8,403) ITER,T(4,4),T(5,3)
403   FORMAT(1X,I5,2F20.6)
      ENDIF
      IF(ITER==LAST) CALL PRINT
      RETURN
```

401 FORMAT(1X,' ITER',13X,'T(4,4)',14X,'T(5,3)')

403 FORMAT(1X,I5,2F20.6)

2F20.6

2F-two floating-point number

20.6-Every data take 20 places;
after decimal (小数点) there are 6 digits

```
LPRINT(4)=.TRUE.
TITLE(4)=' TEMP '
```

```
IF(ITER==LAST) CALL PRINT
RETURN
```

Node numbers: 7 * 7

*************************** .TEMP. ***************************

I =	1	2	3	4	5	6	7
J							
7	2.00E+00	2.30E+00	2.90E+00	3.50E+00	4.10E+00	4.70E+00	5.00E+00
6	1.80E+00	2.08E+00	2.64E+00	3.20E+00	3.76E+00	4.32E+00	4.60E+00
5	1.40E+00	1.64E+00	2.12E+00	2.60E+00	3.08E+00	3.56E+00	3.80E+00
4	1.00E+00	1.20E+00	1.60E+00	2.00E+00	2.40E+00	2.80E+00	3.00E+00
3	6.00E-01	7.60E-01	1.08E+00	1.40E+00	1.72E+00	2.04E+00	2.20E+00
2	2.00E-01	3.20E-01	5.60E-01	8.00E-01	1.04E+00	1.28E+00	1.40E+00
1	0.00E+00	1.00E-01	3.00E-01	5.00E-01	7.00E-01	9.00E-01	1.00E+00

From initial field

$$T = x + y + xy$$

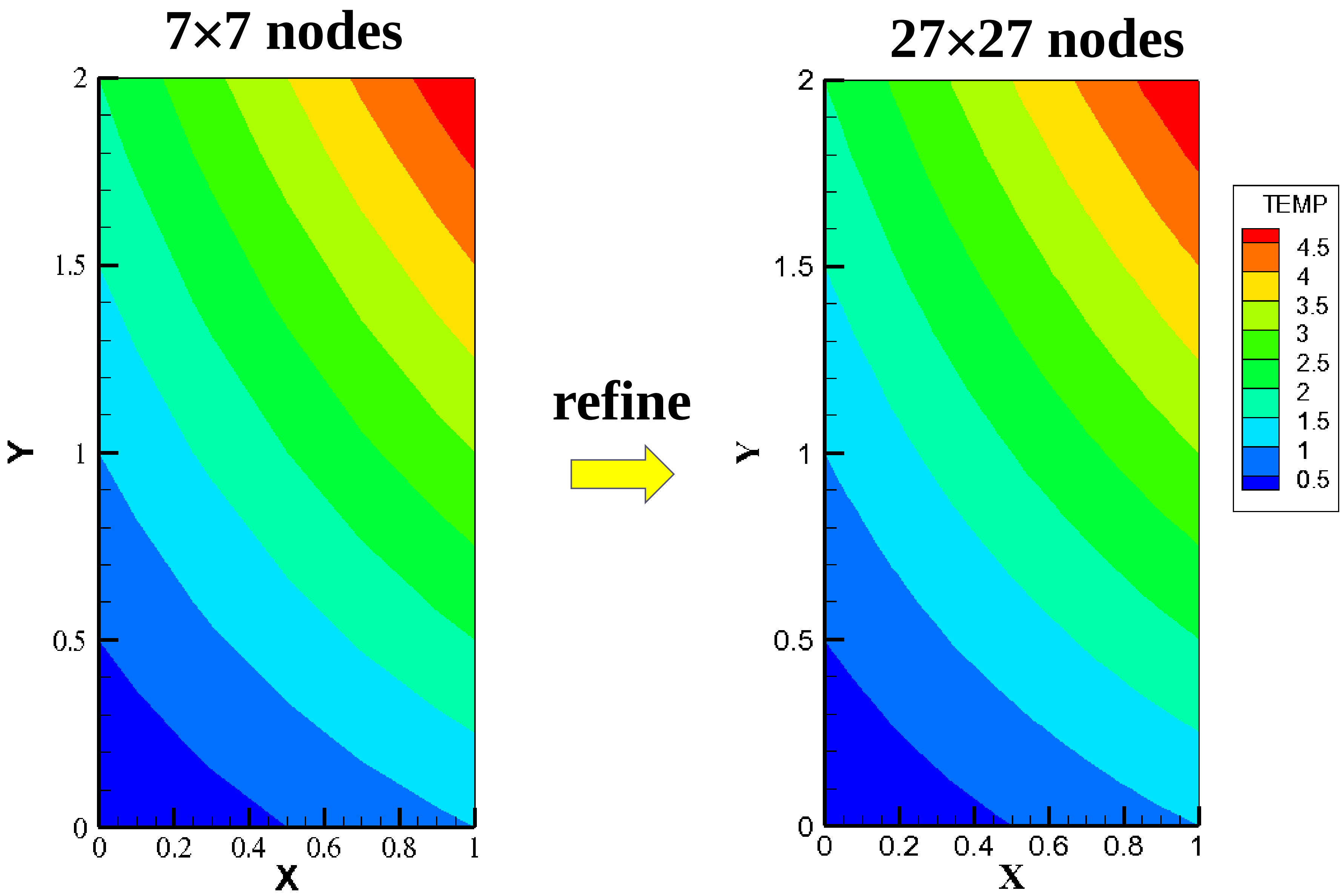


Fig. 2 Isotherms from TECPLOT

11-2 Steady heat conduction in a hollow cylinder ---ASTM for 2nd and 3rd boundary conditions

11-2-1 Physical problem and its math formulation

Known: Steady heat conduction in a hollow cylinder with variable property and source term shown in Fig. 1 has following boundary conditions:

Left boundary---given temperature:

$$T = 100(1+y)$$

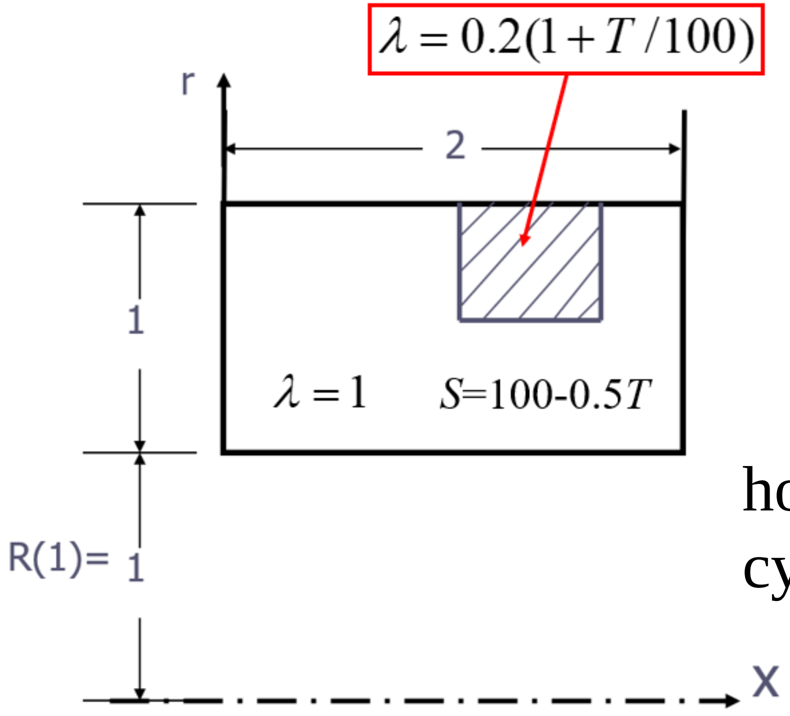
Right boundary---convective heat transfer:

Heat transfer coefficient $h = 5$;

Fluid temperature $T_f = 100$.

Top boundary---adiabatic;

Bottom boundary---given heat flux: $q = 50$

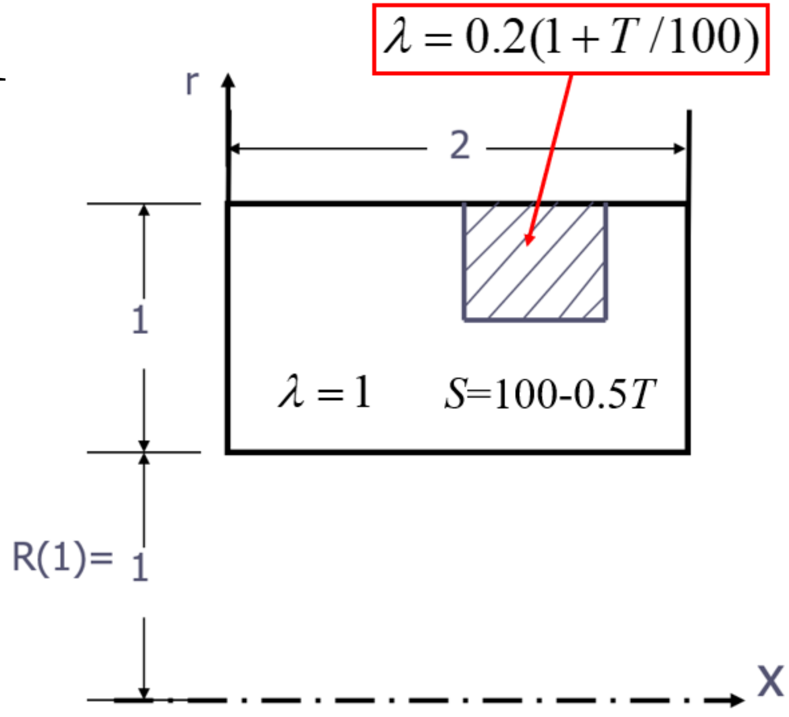


hollow cylinder

Variable thermo-properties:

Thermal conductivity---for most region, $\lambda = 1$

in a local region $\lambda = 0.2(1 + T/100)$



Sources term:

Entire region--- S =100 - 0.5T

Find: temperature distribution in the domain.

Solution:

Steady conduction problem with given Γ and source term: Γ_ϕ S_ϕ.

$$div(\Gamma_\phi grad\phi) + S_\phi = 0$$

Steady heat conduction in a hollow cylinder

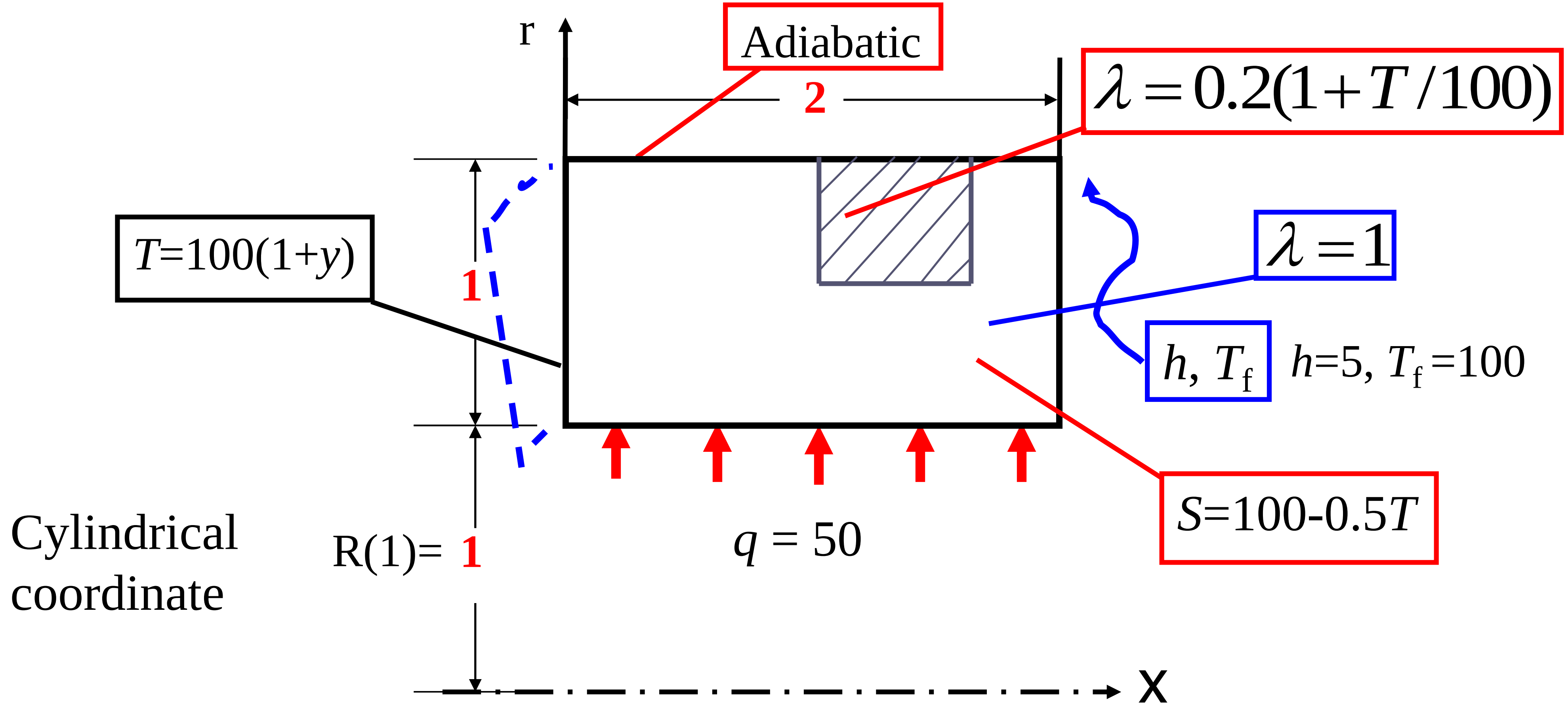


Fig.1 Diagram

B.C. treatment: ASTM or updating method

Method 1: updating method

- Top boundary---adiabatic;

$$\frac{\partial T}{\partial y}=0 \Rightarrow T(i,\mathrm{M1})=T(i,\mathrm{M1}-1)=T(i,\mathrm{M2})$$

- Bottom boundary---given heat flux: $q=50$

$$-\lambda\frac{\partial T}{\partial y}=q \Rightarrow \lambda\frac{T(i,1)-T(i,2)}{\mathrm{Y}(2)-\mathrm{Y}(1)}=q$$

Taking heat transferring into region as positive!

$$\Rightarrow T(i,1)=T(i,2)+q\frac{YDIF(2)}{\lambda}$$

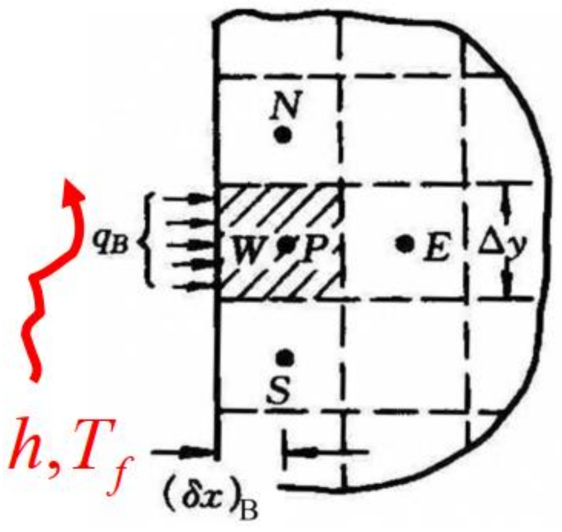
- Right boundary---convective heat transfer: $-\lambda\frac{\partial T}{\partial x}=h(T_{L1}-T_f)$

$$h(T_f-T_{L1})=\frac{\lambda}{XDIF(L1)}(T_{L1}-T_{L2})=GY(T_{L1}-T_{L2}) \qquad GY=\frac{\lambda}{XDIF(L1)}$$

$$\Rightarrow \boxed{T(L_1,j)=(hT_f+GY\cdot T(L_2,j))/(h+GY)}$$

Method 2: ASTM (additional source term method)

Implementing procedure of ASTM



(1) Determining $S_{C,ad}$, $S_{P,ad}$ for CV neighboring to boundary

(2) Adding them into source term of related CV

$$S_C \leftarrow S_C + S_{C,ad} \qquad S_P \leftarrow S_P + S_{P,ad}$$

(3) Setting the conductivity of boundary node as zero

$$\lambda = 0 \rightarrow a_W = \frac{\lambda \Delta y}{\delta x} = 0$$

2nd BC. $S_{C,ad} = \dfrac{q_B \Delta y}{\Delta V}$ $\quad S_{P,ad} = 0$

(4) Solving the algebraic Eqs. for inner nodes

3rd BC. $S_{C,ad} = \dfrac{\Delta y \bullet T_f}{\Delta V[\frac{1}{h} + \frac{(\delta x)_B}{\lambda_B}]}$

(5) Using Newton' law of cooling or Fourier eq. to get the boundary temperatures when converged

$$S_{P,ad} = -\frac{\Delta y}{\Delta V \bullet [1/h + (\delta x)_B / \lambda_B]}$$

11-2-2 Program reading

MODULE

USER_L

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       MODULE USER_L
C*************************************************************
       INTEGER*4 METHOD, I, J
       REAL*8 HTC, TF, GAM1, GY, RES, ARES
C*************************************************************
       END MODULE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE USER
C**************************************************************
      USE START_L
      USE USER_L
      IMPLICIT NONE
C***************************************************************
C--------------------------PROBLEM  TWO----------------------------------------
C    Two–dimensional steady-state heat conduction in a hollow cylinder
C-----Implementation of ASTM and comparison with updating method------
C-------------------------------------------------------------------------------------
C**************************************************************
```

ENTRY GRID

LAST=100 ! A relatively large value for non-linear problems

LSOLVE(4)=.TRUE. ! Solve the energy equation

TITLE(4)= ‘ .TEMP. ‘ ! Title for temperature field print out

LPRINT(4)=.TRUE.

TITLE(13)=‘ .COND. ‘ ! Title for variable conductivity print out

LPRINT(13)=.TRUE. ! Regarding Γ as the13th variable,

```
MODULE START_L
 PARAMETER(NI=100,NJ=200,NIJ=NI,NFMAX=10,NFX4=NFMAX+4)
REAL*8,DIMENSION(NI,NJ,NFX4)::F
```

NF =	1	2	3	4	……	11	12	13	14
Variable	U	V	p_c	T	……	p	ρ	Γ	C_p

MODE=2

R(1)=1.

XL=2.

YL=1.

L1=7

M1=7

Specify lengths and node numbers of domain

CALL UGRID ! Generate interface position of CV

RETURN

```
      ENTRY START
      METHOD=1           ! Boundary temperature updated method;
      DO 100 J=1,M1        While METHOD= 2 is ASTM method
      DO 101 I=1,L1
      T(I,J)=200.        ! Initial values
      IF(I == 1) T(I,J)=100.*(1.+Y(J))    ! Specify left boundary temperature
101 ENDDO
100 ENDDO
      HTC=5.
      Q=50.              Specify boundary condition parameters
      TF=100.
      GAM1=1.    ! Set up conductivity value for main body
      RETURN
*
      ENTRY DENSE       ! Empty, but keep it
      RETURN
```

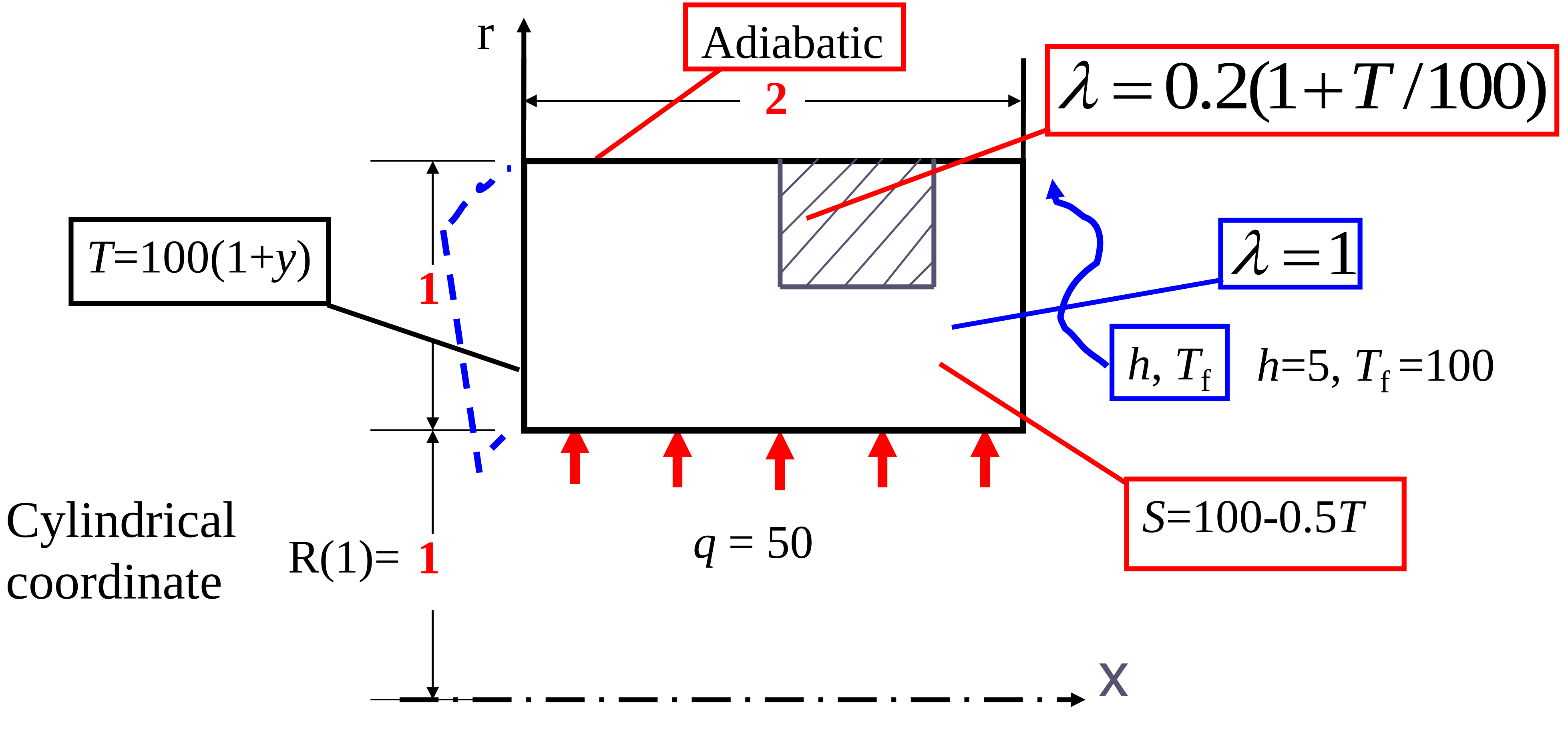


Fig.1 Computational domain

```
      ENTRY BOUND
      DO 300 I=2,L2
      T(I,M1)=T(I,M2)                     !  updated temperature (top boundary)
      T(I,1)=T(I,2)+Q*YDIF(2)/GAM1        !  updated temperature (bottom boundary)
300 ENDDO
      GY=GAM1/XDIF(L1)                    !  Temporary variable for right boundary
      DO 301 J=2,M2
      T(L1,J)=(HTC*TF+GY*T(L2,J))/(HTC+GY)   !  right boundary, updated temperature
301 ENDO
      RETURN
```

$$q=\lambda\frac{T(i,1)-T(i,2)}{YDIF(2)}$$

Heat transferring into the region is taken as positive!

$$T(i,1)=T(i,2)+q\frac{YDIF(2)}{\lambda}$$

$$h(T_f-T_{L1})=\frac{\lambda}{XDIF(L1)}(T_{L1}-T_{L2})=GY(T_{L1}-T_{L2})$$

$$hT_f+GYT_{L2}=T_{L1}(h+GY)$$

$$T_{L1}=(hT_f+GYT_{L2})/(h+GY)$$

```
     ENTRY OUTPUT
     IF(ITER==0) THEN
     PRINT 403, METHOD
     WRITE(8,403) METHOD
403 FORMAT(1X,'   METHOD =‘, I1)
     PRINT 401
     WRITE(8,401)
401 FORMAT(1X,'   ITER',11X, 'T(4,5)’, 14X, 'T(5,3)')
     ENDIF
     IF (ITER>0) PRINT 402, ITER, T(4,5), T(5.3)
     WRITE(8,402) ITER, T(4,5), T(5,3)
402 FORMAT(1X, I6, 2F20.6)
     IF(ITER==LAST) CALL PRINT
     RETURN
```

METHOD is an indicator for boundary condition treatment for 2nd and 3rd kinds

“I1” shows that the value of METHOD is expressed by an integer with one digit

! Integer has at most six digits;, 2 floating-point data with 6 digits after decimal and total length of 20 places.

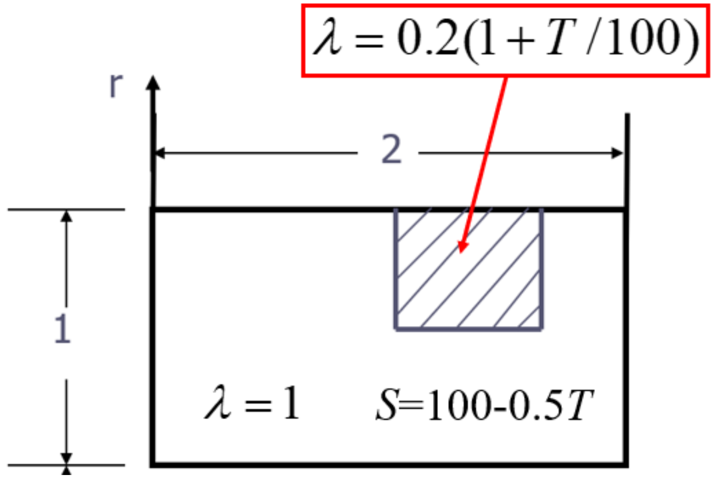
```
      ENTRY GAMSOR
      DO 500 J=1,M1
      DO 501 I=1,L1
      GAM(I,J)=GAM1
501 ENDDO
500 ENDDO
      DO 503 J=4,7
      DO 504 I=4,5
      GAM(I,J)=0.2*(1.+T(I,J)/100.)
504 ENDDO
503 ENDDO
      DO 510 J=2,M2
      DO 511 I=2,L2
      CON(I,J)=100.
      AP(I,J) = -0.5
511 ENDDO
510 ENDDO
```

! Specify Γ for whole domain $\Gamma = \lambda = 1$

! Specify variable λ in a small region

! Specify source term S =100-0.5T

$CON(I,J) = S_C$; $AP(I,J) = S_P$



$$-\lambda \frac{\partial T}{\partial y} = 0$$

Top B.: Adiabatic

Bottom B: $q = 50$

Right B: $-\lambda \frac{\partial T}{\partial x} = h \Delta t$

```
      IF(METHOD==1) RETURN ! Following is for ASTM
      DO 520 I=2,L2
      GAM(I,M1)=0.
      GAM(I,1)=0.  ! Set λ = 0
      CON(I,2)=CON(I,2)+Q*R(1)/ARX(2)
 520  ENDDO                ! Accumulative
      RES=1./HTC+1./GY
      ARES=1./(RES*XCV(L2))
      DO 521 J=2,M2
      GAM(L1,J)=0.  ! Set λ = 0
      CON(L2,J) = CON(L2,J)+ARES*TF
      AP(L2,J) = AP(L2,J)-ARES
 521  ENDDO !Accumulative
      RETURN
      END
```

Bottom B:

$$S_{c,ad} = \frac{qA}{\Delta V} = \frac{q \cdot XCV(i) \cdot R(1)}{ARX(2) \cdot XCV(i)} = \frac{q \cdot R(1)}{ARX(2)}$$

! CVs next to boundary

$$\frac{A}{\Delta V} = \frac{ARX(j)}{ARX(j) \cdot XCV(i)} = \frac{1}{XCV(i)}$$

$$S_{c,ad} = \frac{A}{\Delta V} \frac{T_f}{\delta x/\Gamma + 1/h} = \underbrace{\frac{1}{XCV(i)} \underbrace{\frac{1}{\delta x/\Gamma + 1/h}}_{\text{RES}}}_{\text{ARES}} T_f$$

$$S_{P,ad} = -\frac{1}{XCV(i)} \frac{1}{\delta x/\Gamma + 1/h}$$

11-2-3 Results analysis

COMPUTATION FOR AXISYMMETRICAL SIMULATION

****

METHOD =1 ! For updating method

ITER	T(4,5)	T(5,3)
0	200.000000	200.000000
1	196.503891	193.806549
2	194.450150	190.325912
3	192.184113	187.114395
4	189.861618	184.072250
5	187.567535	181.222870
6	185.361771	178.597488
7	183.282364	176.208923
8	181.350449	174.055115
9	179.575180	172.125107
10	177.957458	170.403229

Initial field

11	176.492798	168.871887
12	175.173325	167.513016
13	173.989273	166.309189
14	172.930008	165.243973
15	171.984665	164.302246
16	171.142624	163.470215
17	170.393753	162.735428
18	169.728561	162.086731
19	169.138290	161.514206
20	168.614944	161.008957
21	168.151245	160.563156
22	167.740601	160.169846
23	167.377090	159.822830
24	167.055481	159.516693
25	166.770981	159.246658

26	166.519409	159.008408
27	166.296982	158.798203
28	166.100388	158.612778
29	165.926620	158.449173
30	165.773102	158.304855
31	165.637451	158.177505
32	165.517609	158.065186
33	165.411758	157.966049
34	165.318222	157.878601
35	165.235626	157.801422
36	165.162720	157.733337
37	165.098282	157.673233
38	165.041412	157.620209
39	164.991196	157.573425
40	164.946838	157.532135
41	164.907684	157.495712
42	164.873108	157.463547

43	164.842590	157.435181
44	164.815643	157.410141
45	164.791870	157.388062
46	164.770844	157.368561
47	164.752319	157.351334
48	164.735947	157.336151
49	164.721497	157.322754
50	164.708740	157.310913
51	164.697495	157.300476
52	164.687561	157.291245
53	164.678772	157.283127
54	164.671051	157.275940
55	164.664200	157.269608
56	164.658157	157.264008
57	164.652847	157.259094
58	164.648148	157.254730
59	164.643982	157.250885
60	164.640289	157.247482

61	164.637070	157.244492
62	164.634201	157.241837
63	164.631683	157.239502
64	164.629471	157.237442
65	164.627502	157.235626
66	164.625778	157.234024
67	164.624268	157.232590
68	164.622894	157.231339
69	164.621689	157.230225
70	164.620636	157.229279
71	164.619736	157.228409
72	164.618896	157.227646
73	164.618179	157.226990
74	164.617538	157.226379
75	164.616974	157.225861
76	164.616486	157.225418
77	164.616058	157.225021
78	164.615662	157.224655
79	164.615341	157.224350
80	164.615036	157.224060

81	164.614746	157.223816
82	164.614517	157.223587
83	164.614304	157.223389
84	164.614120	157.223236
85	164.613968	157.223068
86	164.613815	157.222931
87	164.613693	157.222839
88	164.613571	157.222717
89	164.613495	157.222641
90	164.613403	157.222549
91	164.613312	157.222488
92	164.613251	157.222412
93	164.613205	157.222382
94	164.613159	157.222321
95	164.613113	157.222275
96	164.613037	157.222229
97	164.613007	157.222214
98	164.612976	157.222168
99	164.612946	157.222153
100	164.612930	157.222137
(! ITER	T(4,5)	T(5,3))

The 1st three digits after decimal unchanged during 5 iterations!

! LAST = 100

```
LPRINT(4)=.TRUE.
TITLE(4)=' TEMP '
```

```
IF(ITER==LAST) CALL PRINT
RETURN
```

Node numbers: 7 * 7

Temperature field

**************************** TEMP ****************************

I =	1	2	3	4	5	6	7
J							
7	2.00E+02	1.75E+02	1.70E+02	1.64E+02	1.48E+02	1.25E+02	2.00E+02
6	1.90E+02	1.75E+02	1.70E+02	1.64E+02	1.48E+02	1.25E+02	1.12E+02
5	1.70E+02	1.69E+02	1.69E+02	1.65E+02	1.49E+02	1.26E+02	1.13E+02
4	1.50E+02	1.60E+02	1.68E+02	1.66E+02	1.52E+02	1.28E+02	1.14E+02
3	1.30E+02	1.52E+02	1.68E+02	1.70E+02	1.57E+02	1.33E+02	1.16E+02
2	1.10E+02	1.49E+02	1.72E+02	1.75E+02	1.63E+02	1.39E+02	1.19E+02
1	1.00E+02	1.54E+02	1.77E+02	1.80E+02	1.68E+02	1.44E+02	2.00E+02

Node numbers: 7 * 7

Thermal conductivity

```
TITLE(13)=' COND '
LPRINT(13)=.TRUE.
```

```
IF(ITER==LAST) CALL PRINT
RETURN
```

**************************** COND ****************************

I =	1	2	3	4	5	6	7
J							
7	1.00E+00	1.00E+00	1.00E+00	5.28E-01	4.95E-01	1.00E+00	1.00E+00
6	1.00E+00	1.00E+00	1.00E+00	5.28E-01	4.95E-01	1.00E+00	1.00E+00
5	1.00E+00	1.00E+00	1.00E+00	5.29E-01	4.98E-01	1.00E+00	1.00E+00
4	1.00E+00	1.00E+00	1.00E+00	5.33E-01	5.05E-01	1.00E+00	1.00E+00
3	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00
2	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00
1	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00

COMPUTATION FOR AXISYMMETRICAL SIMULATION
****

METHOD =2

ITER	T(4,5)	T(5,3)
0	200.000000	200.000000
1	163.633240	156.107574
2	164.603409	157.204285
3	164.612839	157.222092
4	164.612747	157.221954
5	164.612747	157.221954
6	164.612747	157.221954
7	164.612747	157.221970
8	164.612747	157.221954
9	164.612747	157.221970
10	164.612747	157.221954
11	164.612747	157.221970
12	164.612747	157.221954

！ For ASTM

In order to keep the 1st three digits after decimal unchanged during 5 iterations, Method 1 needs 90 iterations, while Method 2 only needs 8 iterations! Convergence speed of Method 2 is 10 times of Method 1!

ASTM is recommended.

13	164.612747	157.221970
14	164.612747	157.221954
15	164.612747	157.221970
16	164.612747	157.221954
17	164.612747	157.221970
18	164.612747	157.221954
19	164.612747	157.221970
20	164.612747	157.221954
21	164.612747	157.221970
22	164.612747	157.221954
23	164.612747	157.221970
24	164.612747	157.221954
25	164.612747	157.221970
26	164.612747	157.221954
27	164.612747	157.221970
28	164.612747	157.221954
29	164.612747	157.221970
30	164.612747	157.221954

31	164.612747	157.221970
32	164.612747	157.221954
33	164.612747	157.221970
34	164.612747	157.221954
35	164.612747	157.221970
36	164.612747	157.221954
37	164.612747	157.221970
38	164.612747	157.221954
39	164.612747	157.221970
40	164.612747	157.221954
41	164.612747	157.221970
42	164.612747	157.221954
43	164.612747	157.221970
44	164.612747	157.221954
45	164.612747	157.221970
46	164.612747	157.221954
47	164.612747	157.221970
48	164.612747	157.221954

49	164.612747	157.221970
50	164.612747	157.221954
51	164.612747	157.221970
52	164.612747	157.221954
53	164.612747	157.221970
54	164.612747	157.221954
55	164.612747	157.221970
56	164.612747	157.221954
57	164.612747	157.221970
58	164.612747	157.221954
59	164.612747	157.221970
60	164.612747	157.221954
61	164.612747	157.221970
62	164.612747	157.221954
63	164.612747	157.221970
64	164.612747	157.221954
65	164.612747	157.221970
66	164.612747	157.221954

67	164.612747	157.221970
68	164.612747	157.221954
69	164.612747	157.221970
70	164.612747	157.221954
71	164.612747	157.221970
72	164.612747	157.221954
73	164.612747	157.221970
74	164.612747	157.221954
75	164.612747	157.221970
76	164.612747	157.221954
77	164.612747	157.221970
78	164.612747	157.221954
79	164.612747	157.221970
80	164.612747	157.221954
81	164.612747	157.221970
82	164.612747	157.221954
83	164.612747	157.221970
84	164.612747	157.221954

85	164.612747	157.221970
86	164.612747	157.221954
87	164.612747	157.221970
88	164.612747	157.221954
89	164.612747	157.221970
90	164.612747	157.221954
91	164.612747	157.221970
92	164.612747	157.221954
93	164.612747	157.221970
94	164.612747	157.221954
95	164.612747	157.221970
96	164.612747	157.221954
97	164.612747	157.221970
98	164.612747	157.221954
99	164.612747	157.221970
100	164.612747	157.221954

For diffusion problems, further iterations after convergence will not change the results! But it is not for convective problems!

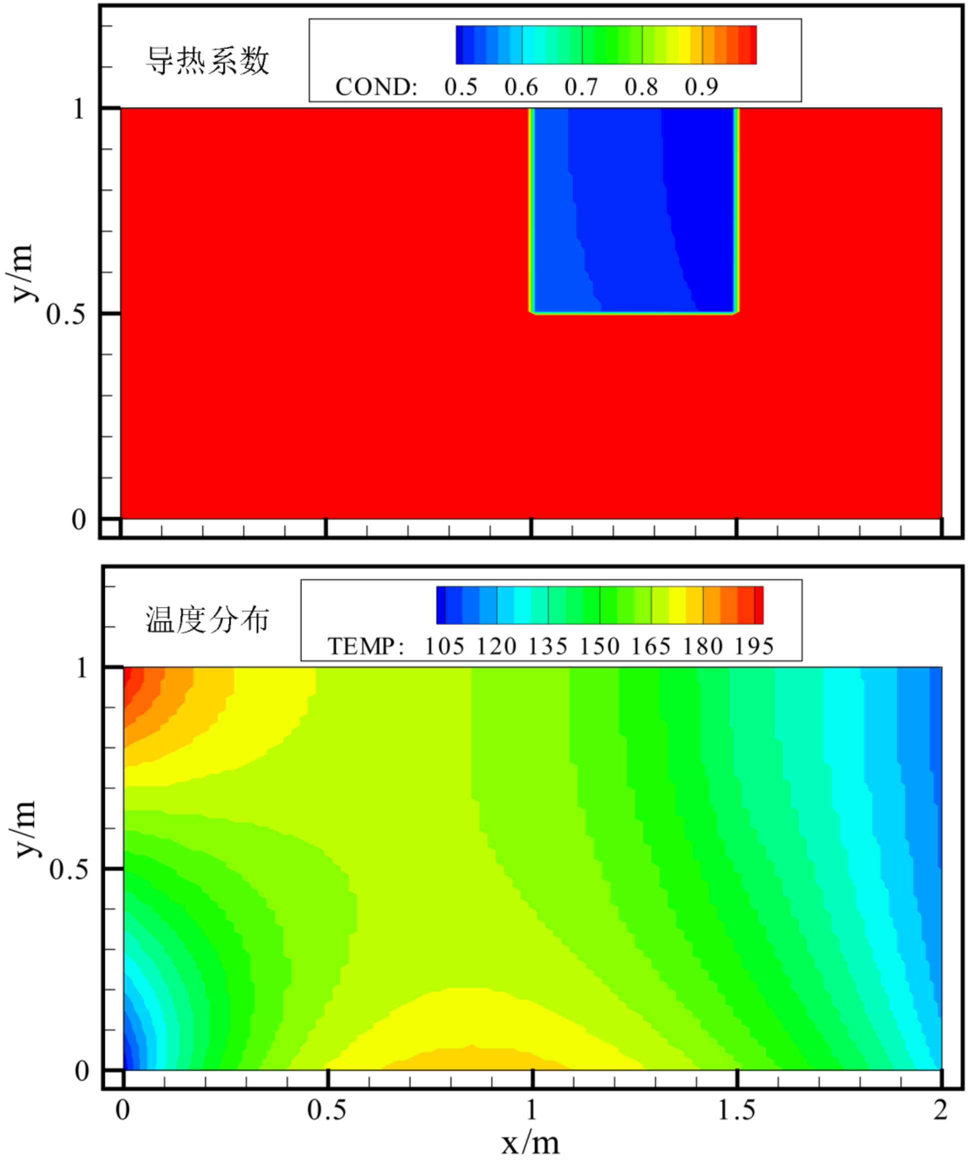
！ For METHOD＝2, all boundary temperatures will be printed only after convergence.

```
    ENTRY BOUND
    DO 300 I=2,L1
    T(I,M1)=T(I,M2)
    T(I,1)=T(I,2)+Q*YDIF(2)/GAM1
300 ENDDO
```

```
T(L1,J)=(HTC*TF+GY*T(L2,J))/(HTC+GY)
```

！ To save time, following IF statement can be added before DO–loop 300：

IF(METHOD==2 .AND. ITER < LAST) RETURN

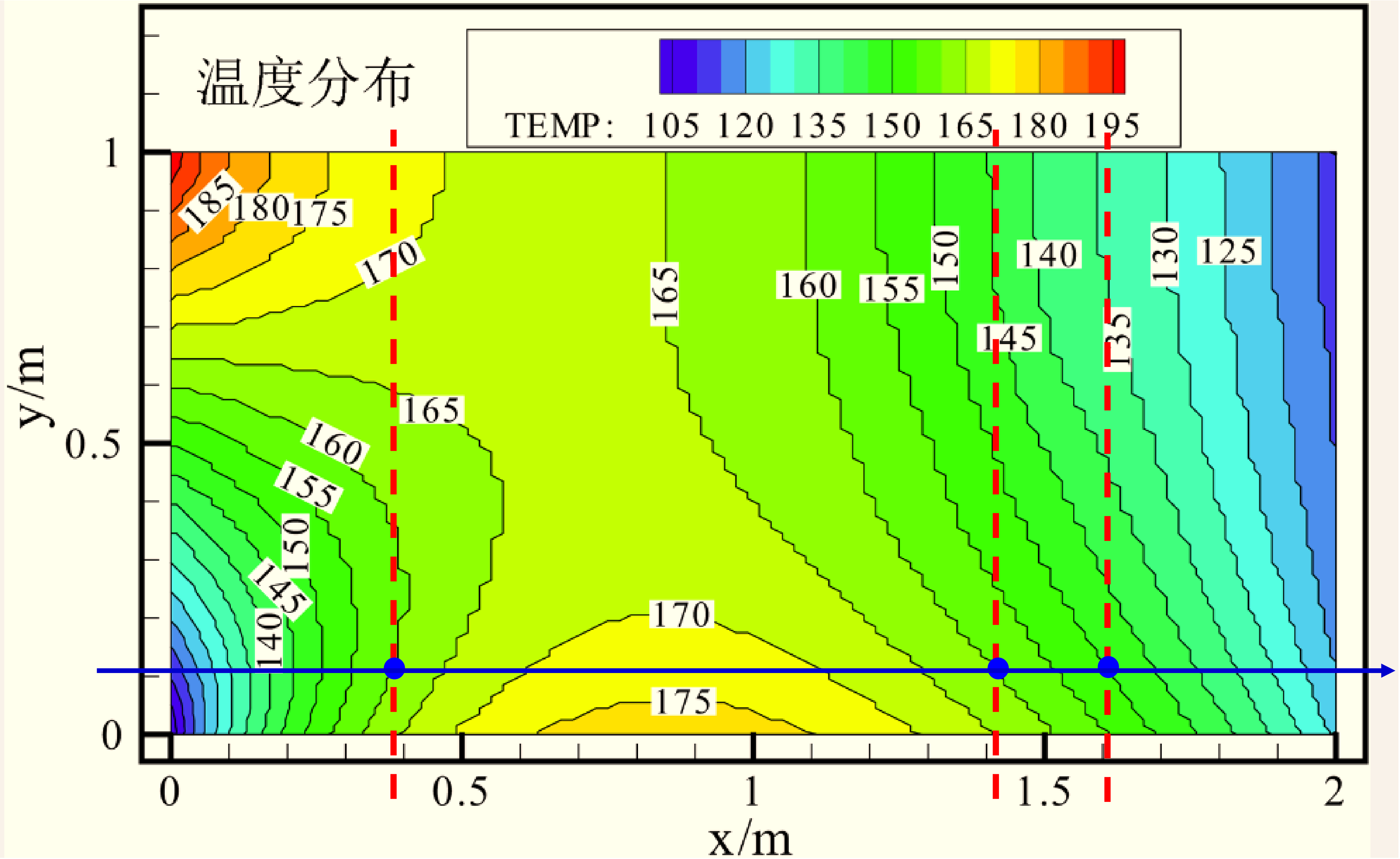


Thermal conductivity field

Temperature field

Fig.2 Computational results

Fig.3 Isotherms



Temperature field

$$q = \lambda \frac{\Delta T}{\Delta y}$$

***q* = 50**

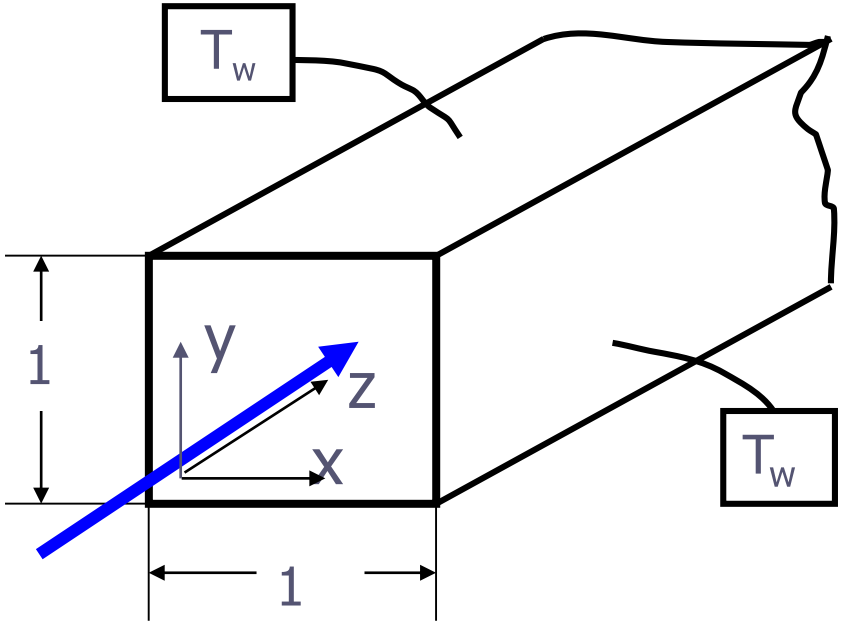
Five degrees difference in the same vertical distance (~ 0.1) for **uniform heat flux**

11-3 Example 3 Fully-developed heat transfer in a square duct －Numerical techniques for FDHT

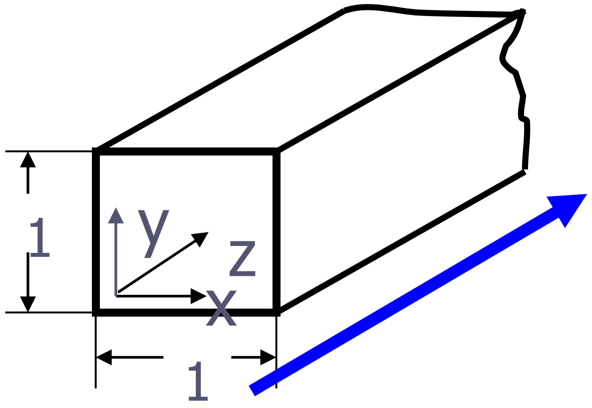
11-3-1 Physical problem and its math formulation

Known： Fully developed laminar heat transfer of fluid with constant properties (Fig. 1) in a square duct. The wall temperatures are uniform.

Find： Velocity and temperature distribution in cross section and *fRe* and *Nu*.



Solution: For fully developed laminar flow in a straight duct, $u = 0$, $v = 0$, and $\partial w/\partial z = 0$



$$\rho(\underbrace{\cancel{u}\frac{\partial w}{\partial x}}_{0}+\underbrace{\cancel{v}\frac{\partial w}{\partial y}}_{0}+\underbrace{w\cancel{\frac{\partial w}{\partial z}}}_{0})=-\frac{\partial p}{\partial z}+\eta(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\underbrace{\cancel{\frac{\partial^2 w}{\partial z^2}}}_{0})$$

$$\eta(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2})-\frac{\partial p}{\partial z}=0 \quad \xrightarrow{\text{Neglecting cross section variation of } p} \quad \eta(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2})-\frac{dp}{dz}=0$$

GE: $\eta(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2})-\frac{dp}{dz}=0$ Axial velocity w

GGE: $\frac{\partial(\rho^*\Phi)}{\partial t}+div(\rho^*\vec{u}\Phi)=div(\Gamma_\Phi grad\Phi)+S_\phi^*$

Compared with standard form, w-eq. is of conduction type.

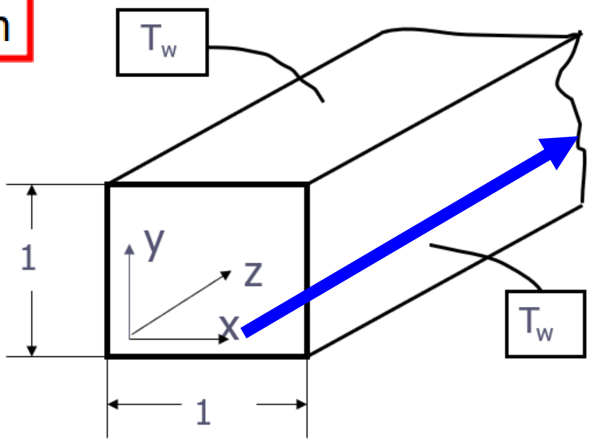
Thus, $\Gamma_\phi = \eta$ $S_C = -dp/dz$

Governing equation for fluid temperature:

$$\rho c_p(\cancelto{0}{u}\frac{\partial T}{\partial x}+\cancelto{0}{v}\frac{\partial T}{\partial y}+w\frac{\partial T}{\partial z})=\frac{\partial}{\partial x}(\lambda\frac{\partial T}{\partial x})+\frac{\partial}{\partial y}(\lambda\frac{\partial T}{\partial y})+\frac{\partial}{\partial z}(\lambda\cancelto{0}{\frac{\partial T}{\partial z}})$$

Neglecting axial heat conduction

Thus: $$\rho c_p w\frac{\partial T}{\partial z}=\frac{\partial}{\partial x}(\lambda\frac{\partial T}{\partial x})+\frac{\partial}{\partial y}(\lambda\frac{\partial T}{\partial y})$$



- In summary, the total Governing Eqs.:

$$\left\{\begin{array}{l}\eta(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2})-\frac{dp}{dz}=0 \\ \frac{\partial}{\partial x}(\lambda\frac{\partial T}{\partial x})+\frac{\partial}{\partial y}(\lambda\frac{\partial T}{\partial y})=\rho c_p w\frac{\partial T}{\partial z}\end{array}\right.$$

B.C.: no slip at walls

B.C.: T_w at walls

11-3-2 Numerical methods

(1) Dimensionless temperature

Define dimensionless temperature $\Theta = \dfrac{T - T_w}{T_b - T_w}$

T_b: average bulk temperature

Then： $T = \Theta(T_b - T_w) + T_w$, $\dfrac{\partial \Theta}{\partial z} = 0$

$$\Rightarrow \frac{\partial T}{\partial z} = \Theta \frac{dT_b}{dz}$$

Energy eq. is transformed into following conduction equation with source term:

$$\frac{\partial}{\partial x}(\lambda \frac{\partial T}{\partial x}) + \frac{\partial}{\partial y}(\lambda \frac{\partial T}{\partial y}) - \rho c_p w \frac{\partial T}{\partial z} = 0 \Rightarrow \frac{\partial}{\partial x}(\lambda \frac{\partial T}{\partial x}) + \frac{\partial}{\partial y}(\lambda \frac{\partial T}{\partial y}) - \underline{\rho c_p w \Theta \frac{dT_b}{dz}} = 0$$

Compared with the standard form： $\Gamma_\phi = \lambda$ $\quad S_C = -\rho c_p w \Theta \dfrac{dT_b}{dz}$

(2) Numerical methods

1.This flow problem is governed by two conduction-type equations with source term;

$$\eta\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}\right)-\frac{dp}{dz}=0 \qquad \frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda\frac{\partial T}{\partial y}\right)-\rho c_p w\Theta\frac{dT_b}{dz}=0$$

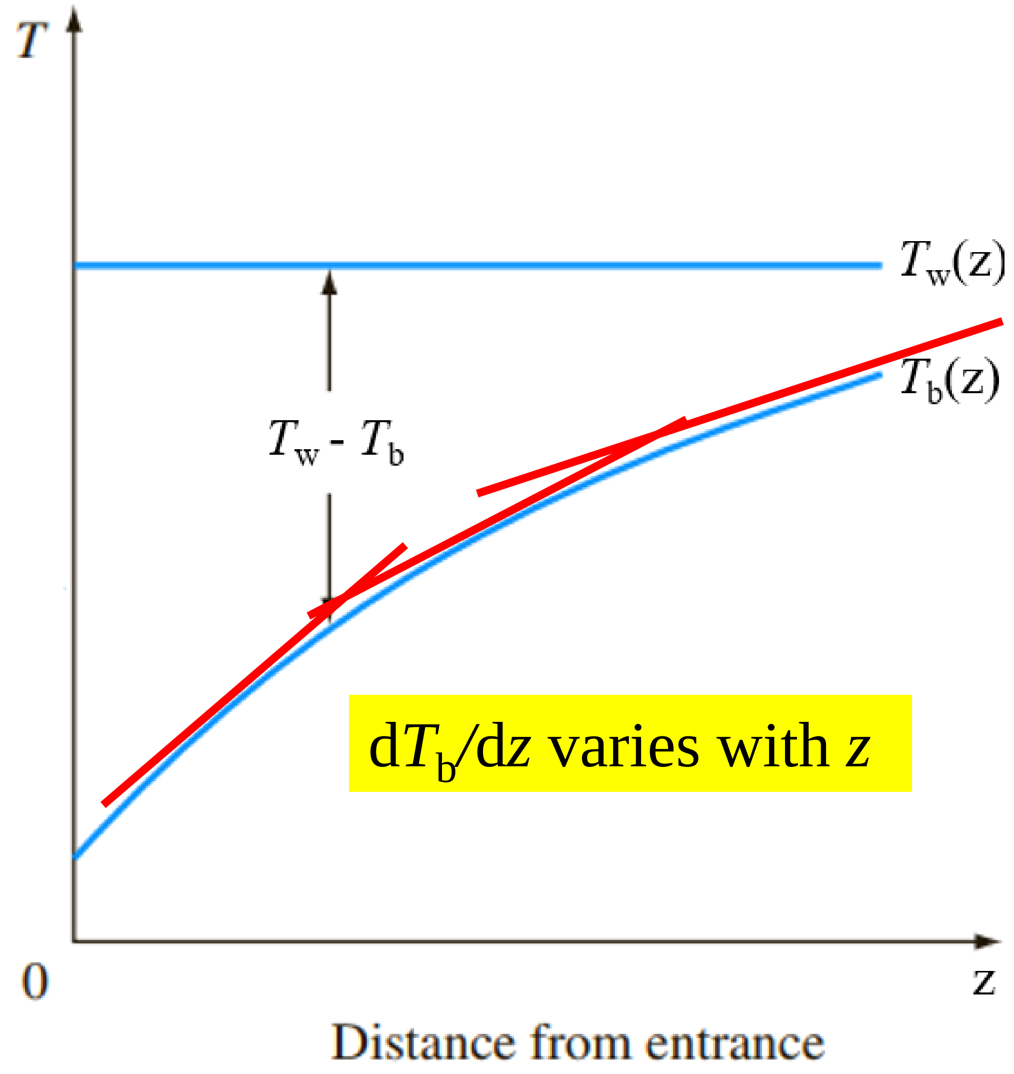
2.The two equations are partially coupled: Velocity *w* is in the source term of temperature. However, temperature is not included in *w*-equation. Thus *w*-eq. should be solved first;

3.For uniform wall temperature, dT_b/dz does not equal constant and an assumed value can be used for simulation. During iteration, the dimensionless temperature Θ (which is included in the source term of temperature) should be updated.

$$S_C=-\rho c_p w\Theta\frac{dT_b}{dz}$$

During simulation, the value of dT_b/dz is assumed and Θ is updated iteratively.

$$S_C = -\rho c_p w \Theta \frac{dT_b}{dz}$$



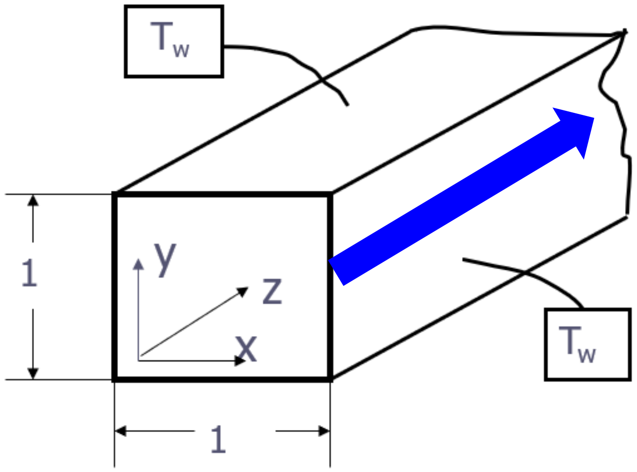


Fig. 2 Streamwise variation of fluid temperature at uniform wall temperature condition

11-3-3 Program reading

MODULE USER_L

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
     MODULE USER_L
C***********************************************************************
     INTEGER*4 I,J
     REAL*8 AMU, DEN, RHOCP, DPDZ, DTBDZ, ASUM, TSUM, AR,
   1 WR, WBAR, TB, DH, RE, FRE, ANU, TW, QW, THETA, DTDZ
     END MODULE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    SUBROUTINE USER
C***********************************************************************
    USE START_L
    USE USER_L
    IMPLICIT NONE
C***********************************************************************
C------------------------------    PROBLEM THREE------------------------------------------
     Fully developed laminar fluid flow and heat transfer in a square duct
C-----------------------------------------------------------------------------------------
C***********************************************************************
```

ENTRY GRID

TITLE(4)=' .THETA. '   ! Title of dimensionless temperature for output

TITLE(5)='. W/WBAR. '   ! Title of dimensionless velocity for output

LSOLVE(5)=.TRUE.   ! *w* (5th variable) solved first, temperature is not solved temporarily

LPRINT(4)=.TRUE.

LPRINT(5)=.TRUE.

LAST=22

XL=0.5   ! Symmetry, only 1/4 domain needs to be solved

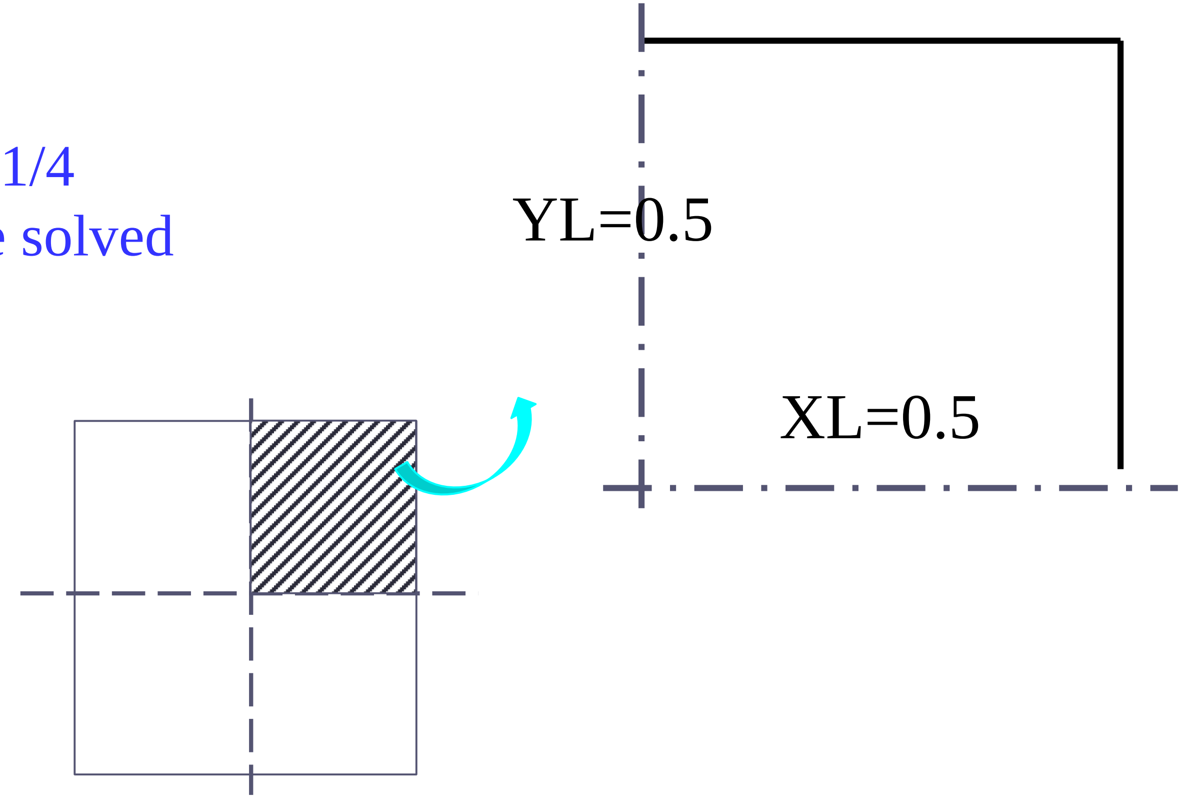
YL=0.5

L1=7

M1=7

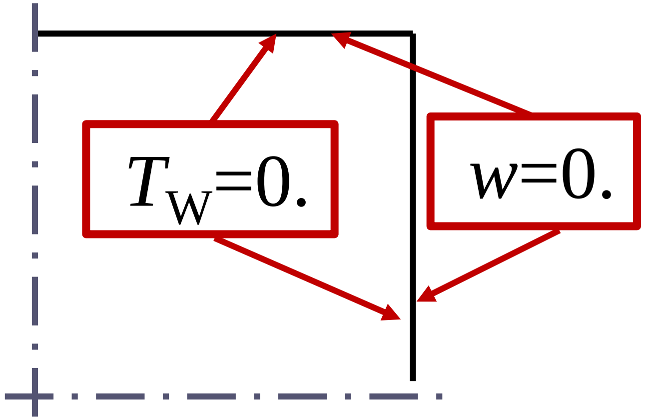
CALL UGRID

RETURN



ENTRY START

```
      TW=0.                    ! Wall temperature
      DO 100 J=1,M1
      DO 100 I=1,L1
      W(I,J)=0.                ! Set up initial fields, and w=0 at walls
      T(I,J)=1.
      T(I,M1)=TW               ! Set up wall temp. for right and top walls
      T(L1,J)=TW
  100 CONTINUE
      AMU=1.
      DEN=1.
      COND=1.
      CP=1.
      RHOCP=DEN*CP
      DPDZ=-100.
      DTBDZ=5.
      RETURN
```



! Set up properties; $\eta = 1$ (very large), to ensure laminar flow.

! This is not a true flow problem, and there is no convection. RHOCP here is for the source term in conduction equation.

! This value must be less than zero

! Fluid is heated. The value is arbitrarily assumed

$$\frac{\partial(\rho^* \Phi)}{\partial t} + div(\rho^* \vec{u} \Phi) = div(\Gamma_\Phi grad \Phi) + S_\phi^*$$

```
ENTRY DENSE        ! Empty, but keep it
RETURN

ENTRY BOUND
ASUM=0.
WSUM=0.            ! Initial values for summation
TSUM=0.
DO 300 J=2,M2
DO 301 I=2,L2
AR=XCV(I)*YCV(J)
WR=W(I,J)*AR
WSUM=WSUM+WR
ASUM=ASUM+AR
TSUM=TSUM+WR*T(I,J)
301 ENDDO
300 ENDDO
```

Element area $A_{i,j}$

$\iint w(i,j)dA_{i,j}$

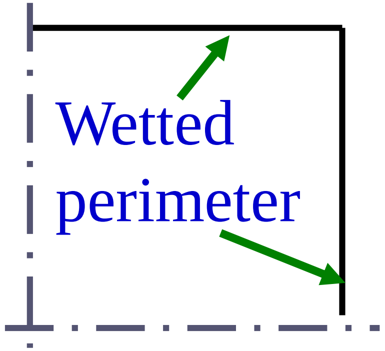
$\iint dA_i$

$\iint w(i,j)T(i,j)dA_{i,j}$

$$T_b = \frac{\iint w(i,j)T(i,j)dA_{i,j}}{\iint w(i,j)dA_{i,j}}$$

$$w_m = \frac{\iint w(i,j)dA_{i,j}}{\iint dA_{i,j}}$$

$$w_m = \frac{\iint w(i,j)\,dA_{i,j}}{\iint dA_{i,j}}$$



WBAR=WSUM/ASUM

$$T_b = \frac{\iint w(i,j)T(i,j)\,dA_{i,j}}{\iint w(i,j)\,dA_{i,j}}$$

TB=TSUM/(WSUM+1.E-30)

! To avoid overflow, a small value is added.

DH=4.*XL*YL/(XL+YL)

RE=DEN*WBAR*DH/AMU

$$f\,\mathrm{Re} = -\frac{(dp/dx)D_h}{\frac{1}{2}\rho w_m^2}\mathrm{Re}$$

FRE=-2.*DPDZ*DH/(DEN*WBAR**2+1.E-30)*RE

QW=DTBDZ*RHOCP*WSUM/(XL+YL)

$$q_w = \frac{dT_b}{dz}\rho c_p \sum (w_{i,j}A_{i,j})\frac{1}{XL+YL}$$

ANU=QW*DH/(COND*(TW-TB)+1.E-30)

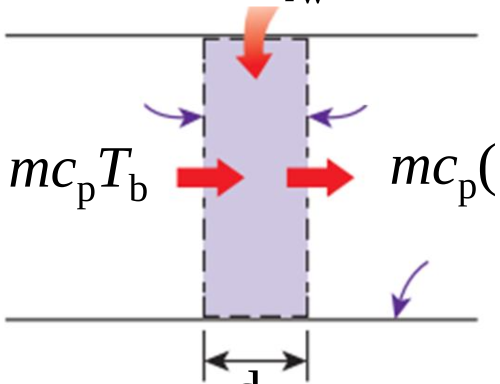
$$Nu = \frac{hD_h}{\lambda} = \frac{D_h}{\lambda}\frac{q_w}{(T_w - T_b)}$$

Energy balance:

$$Q = mc_p dT_b = q_w L dz$$

$$m = \sum (\rho w_{i,j} A_{i,j})$$

$Q = q_w L dz$

$mc_p T_b$  $mc_p(T_b + dT_b)$

dz

```
      IF(ITER>10)   LSOLVE(5)=.FALSE.
      LSOLVE(4)=.TRUE.
      CONTINUE
      RETURN
*
      ENTRY OUTPUT
      IF(ITER==0) THEN
      PRINT 401
      WRITE(8,401)
401 FORMAT(1X,' ITER',12X,'F.RE',17X,'NU')
      ELSE
      PRINT 402, ITER,FRE,ANU
      WRITE(8,402) ITER,FRE,ANU
402 FORMAT(1X,I6,1P2E20.4)
       ENDIF
      IF(ITER/=LAST) RETURN
      DO 410 J=1,M1
      DO 411 I=1,L1
      W(I,J)=W(I,J)/WBAR
      T(I,J)=(T(I,J)-TW)/(TB-TW)
411 ENDDO
410 ENDDO
      CALL PRINT
      RETURN
```

! Switch of solved variable; solve *T* when ITER >=10

1P2E20.4, Scientific expression of data

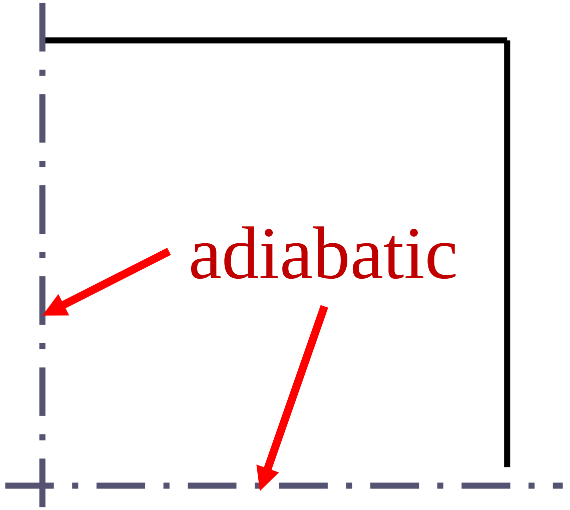
! Dimensionless to make the result more general

$$\bar{w} = \frac{w}{w_m}$$

$$\Theta = \frac{T - T_w}{T_b - T_w}$$

Bottom, Left

```
      ENTRY GAMSOR
      DO 500 I=1,L1
      DO 500 J=1,M1
      GAM(I,J)=AMU                     ! Γ for velocity w
      IF(NF== 4) GAM(I,J)=COND         ! Γ for temperature T
      GAM(I,1)=0.
      GAM(1,J)=0.                      ! Symmetric=adiabatic for both w and T.
 500  CONTINUE
      IF(NF== 4) GOTO 511
      DO 510 J=2,M2
      DO 510 I=2,L2
      CON(I,J)=-DPDZ  ! Source term of w
 510  CONTINUE
      RETURN
 511  DO 520 J=2,M2
      DO 520 I=2,L2
      THEAT=(T(I,J)-TW)/(TB-TW+1.E-30) ! Updating Θ
      DTDZ=THEAT*DTBDZ
 520  CON(I,J)=-RHOCP*W(I,J)*DTDZ      ! Source term S_c of temp.
      RETURN
      END
```



$$\boxed{S_C = -\rho c_p w \Theta \frac{dT_b}{dz}} \qquad \frac{\partial T}{\partial z} = \Theta \frac{dT_b}{dz}$$

11-3-4 Results analysis

COMPUTATION IN CARTESIAN COORDINATES
****

ITER	F.RE	NU
0	0.0000E+00	0.0000E+00
1	6.5168E+01	-3.8363E+00
2	5.6545E+01	-4.4212E+00
3	5.5151E+01	-4.5330E+00
4	5.4891E+01	-4.5545E+00
5	5.4841E+01	-4.5587E+00
6	5.4831E+01	-4.5595E+00
7	5.4829E+01	-4.5596E+00
8	5.4829E+01	-4.5596E+00
9	5.4829E+01	-4.5596E+00
10	5.4829E+01	-4.5596E+00
11	5.4829E+01	4.5875E+00
12	5.4829E+01	3.3408E+00
13	5.4829E+01	3.0894E+00
14	5.4829E+01	3.0361E+00
15	5.4829E+01	3.0257E+00

Energy eq. has not been solved. The values are meaningless

Switch of solved variable

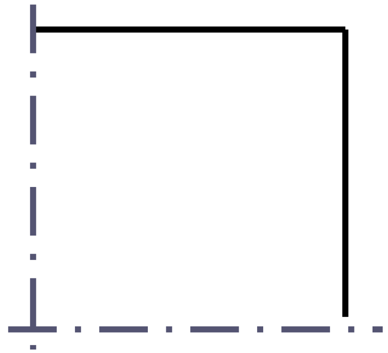
```
IF(ITER>10) LSOLVE(5)=.FALSE.
LSOLVE(4)=.TRUE.
```

16	5.4829E+01	3.0240E+00
17	5.4829E+01	3.0238E+00
18	5.4829E+01	3.0237E+00
19	5.4829E+01	3.0237E+00
20	5.4829E+01	3.0238E+00
21	5.4829E+01	3.0238E+00
22	5.4829E+01	3.0238E+00

Four digits after decimal remain unchanged in successive 6 iterations

*************************** .W/WBAR. ***************************

I =	1	2	3	4	5	6	7
J							
7	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
6	0.00E+00	4.58E-01	4.34E-01	3.83E-01	2.95E-01	1.44E-01	0.00E+00
5	0.00E+00	1.12E+00	1.06E+00	9.12E-01	6.72E-01	2.95E-01	0.00E+00
4	0.00E+00	1.58E+00	1.48E+00	1.26E+00	9.12E-01	3.83E-01	0.00E+00
3	0.00E+00	1.87E+00	1.74E+00	1.48E+00	1.06E+00	4.34E-01	0.00E+00
2	0.00E+00	2.00E+00	1.87E+00	1.58E+00	1.12E+00	4.58E-01	0.00E+00
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00



No decoration before output (未作修饰)

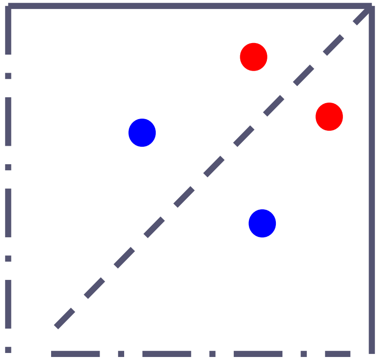
(initial values)

*************************** .THETA. ***************************

I =	1	2	3	4	5	6	7
J							
7	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
6	-6.63E-01	2.41E-01	2.14E-01	1.65E-01	1.02E-01	3.41E-02	0.00E+00
5	-6.63E-01	7.38E-01	6.53E-01	5.00E-01	3.07E-01	1.02E-01	0.00E+00
4	-6.63E-01	1.22E+00	1.08E+00	8.19E-01	5.00E-01	1.65E-01	0.00E+00
3	-6.63E-01	1.61E+00	1.42E+00	1.08E+00	6.53E-01	2.14E-01	0.00E+00
2	-6.63E-01	1.84E+00	1.61E+00	1.22E+00	7.38E-01	2.41E-01	0.00E+00
1	-6.63E-01	-6.63E-01	-6.63E-01	-6.63E-01	-6.63E-01	-6.63E-01	0.00E+00

No decoration(未作修饰) (initial values)

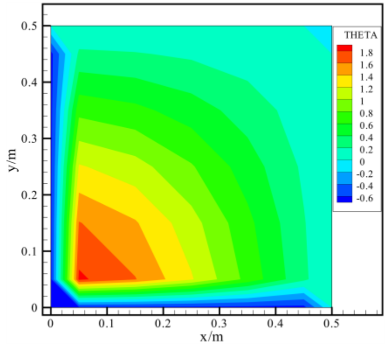
Decoration: before output, set:
THETA (1， j) = THETA(2,j)
THETA (i,1) = THETA(i,2)



Symmetry about diagonal

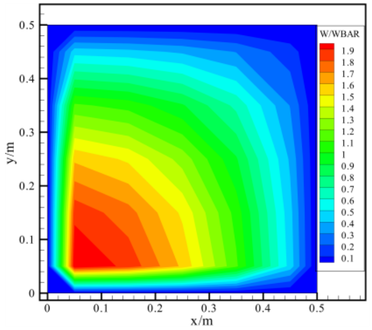
No decoration

Θ

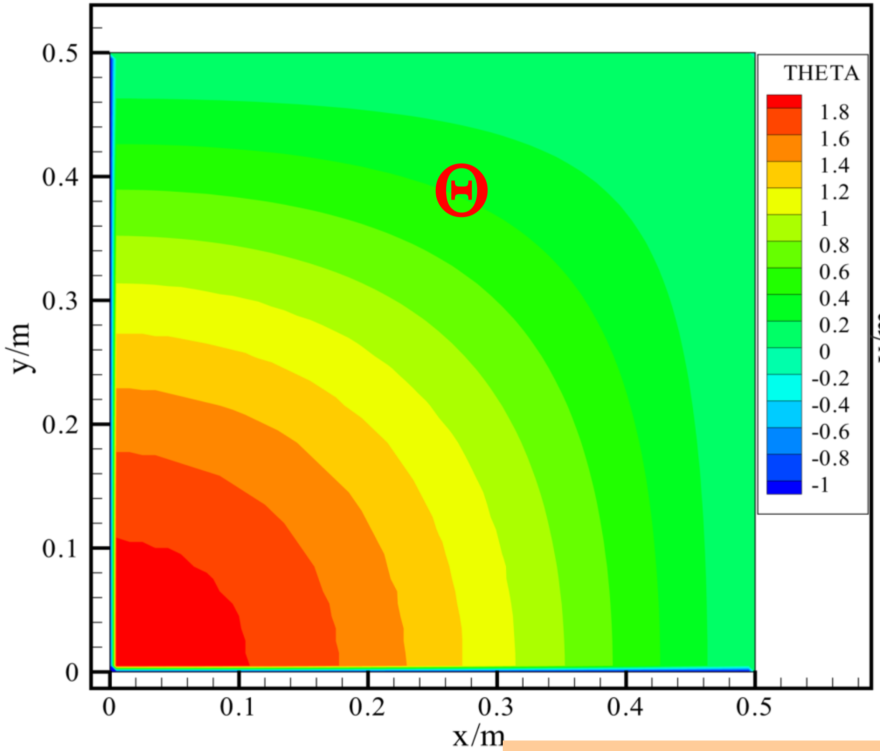


No decoration

W/WBAR



With decoration



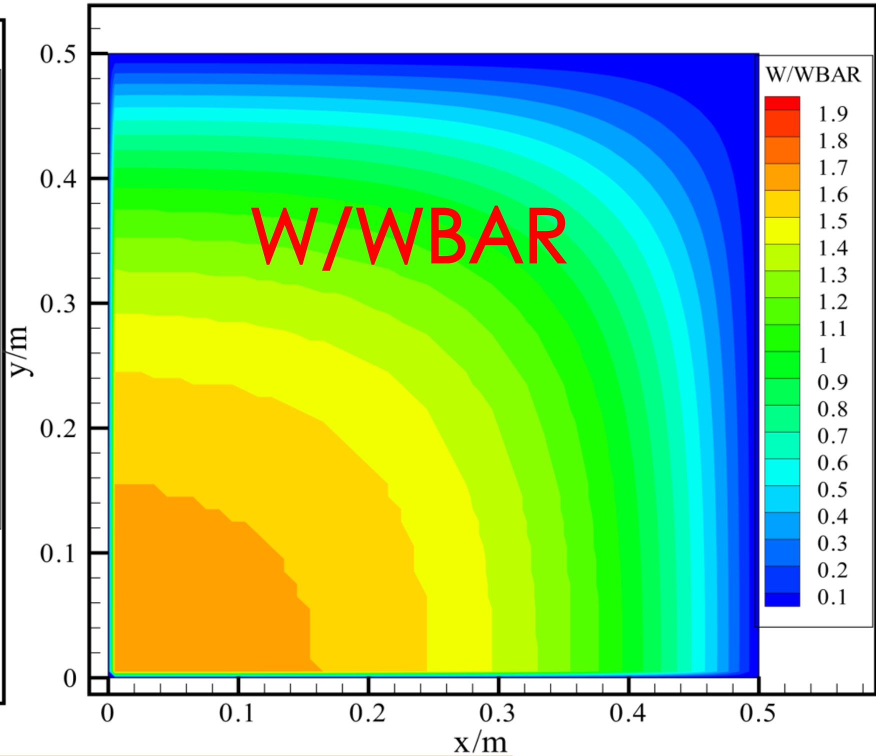


Fig. 3 Results of Problem 3

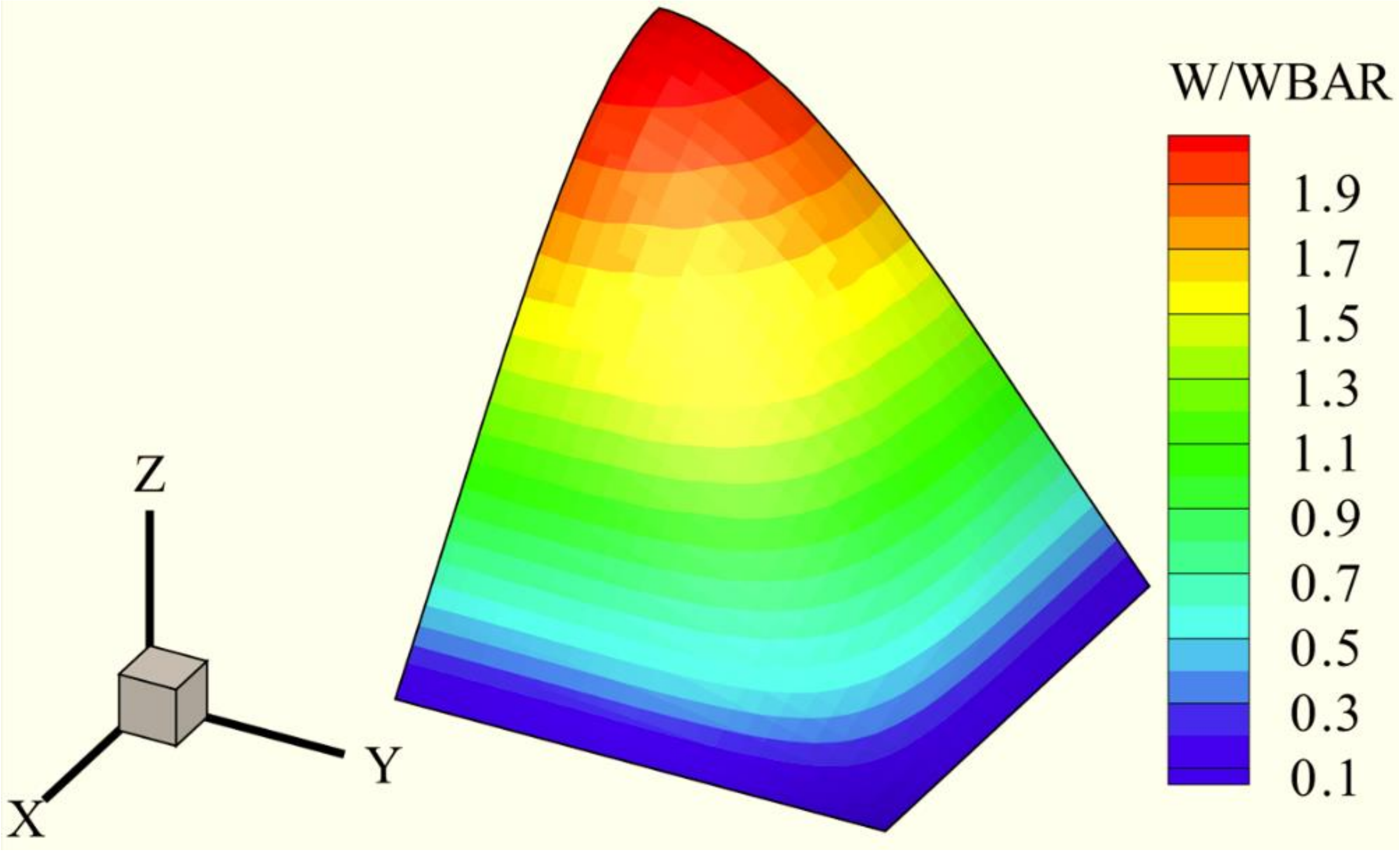


Fig. 4 Pictorial (立体)view of axial velocity distribution

Do the assumed values of dp/dz= -100, dT_b/dz=5 affect fRe and Nu?

GE: $$\eta\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}\right)-\frac{dp}{dz}=0 \qquad \lambda\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right)-\rho c_p w\Theta\frac{dT_b}{dz}=0$$

Introducing characteristic length X_L, characteristic velocity w_m , the above Eqs. can be dimensionless

$$\frac{\partial^2 \overline{w}}{\partial \overline{x}^2}+\frac{\partial^2 \overline{w}}{\partial \overline{y}^2}+\frac{1}{8}\left(1+\frac{X_L}{Y_L}\right)^2 f\,\mathrm{Re}=0 \qquad \frac{\partial^2 \Theta}{\partial x^2}+\frac{\partial^2 \Theta}{\partial y^2}+Nu\cdot\overline{w}\cdot\Theta=0$$

where $\overline{w}=w/w_{\mathrm{m}}$ $\quad \Theta=\dfrac{T-T_w}{T_b-T_w}$

Thus, the assumed values of dp/dz, dT_b/dz do not affect the calculated fRe and Nu

本组网页地址：http://nht.xjtu.edu.cn 欢迎访问！

Teaching PPT will be loaded on our website

同舟共济

渡彼岸！

People in the same boat help each other to cross to the other bank, where….