```
C#############################################################################
c                                                                             #
c                    SUBROUTINE PROGRAM                                       #
C                  VERSION 1.0 (28/07/2009)                                   #
C                  AUTHORIZED BY ZHANG JINGXIN                                #
C                               SHANGHAI JIAO TONG UNIVERSITY                 #
C                               SHANGHAI, CHINA                               #
c-----------------------------------------------------------------------------#
c                      computes the velocity                                  #
c                                                                             #
c#############################################################################
      Subroutine PROFW                          垂向流速计算 (qz)
      Include './Include/OCERM_INF'
    Dimension AAAA(KB,KB),BBBB(KB)
c   Dimension IS(KBM),JS(KBM)
c   Dimension VH(KBM),VHP(KBM)
    Dimension AA(3*KB-2), BB(KB)
    Dimension ELFX(IJM),ELFY(IJM)
    IIII = 0
      IJM_B = IJM_DYN_B + IIII
      IJM_E = IJM_DYN_E + IIII
C=============================================================================C
C                       matrix of the equations                               c
C=============================================================================C
      Do K1 = 1, KB
         Do K2 = 1, KB
          AAAA(K1,K2) = 0.0
        Enddo
    Enddo
    Do K1 = 1, 3 * KB - 2
       AA(K1) = 0.0
    Enddo
    Do K1 = 1, KB
       BB(K1) = 0.0
       BBBB(K1) = 0.0
    Enddo
!$OMP PARALLEL DEFAULT(SHARED) PRIVATE(I,J,K,K1,K2,AAAA,BBBB,AA,BB,BU,
!$OMP&          BV,L)
!$OMP DO
      Do I = IJM_B, IJM_E    (1, IJM)
         ELFX(I) = 0.0
```

$$\frac{q_{zi}^{*} - q_{zi}^{n}}{\Delta t} = F q_{zi}^{n} + \left[\frac{\partial}{D\partial\sigma}\left(\frac{\nu_{tv}}{D}\frac{\partial q_i^{v}}{\partial\sigma}\right)\right]_i$$

$$\Rightarrow A_{i2}^{n} Q_{zi}^{*} = G_{zi}^{n}$$

$$q_\sigma = \frac{q_z}{D} - \frac{q_x}{D}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial\xi}{\partial x}\right) - \frac{q_y}{D}\left(\sigma\frac{\partial D}{\partial y} + \frac{\partial\xi}{\partial y}\right) - \left(\sigma\frac{\partial D}{\partial t} + \frac{\partial\xi}{\partial t}\right)$$

```
          ELFY(I) = 0.0
          If(CCM(I) .EQ. 1.0) Then
             Do J = 1, CELL_POLYGEN(I)
                If(CFM(CELL_SIDE(I,J,1)) .EQ. 1.0) Then
                   ELFX(I) = ELFX(I) + CELL_CUV(I,J,6) *
     &                    (ELF(CELL_SIDE(I,J,2)) + ELF(I)) / 2. *
     &                     CELL_CUV(I,J,7)
                   ELFY(I) = ELFY(I) + CELL_CUV(I,J,6) *
     &                    (ELF(CELL_SIDE(I,J,2)) + ELF(I)) / 2. *
     &                     CELL_CUV(I,J,8)
                Else
                  ELFX(I) = ELFX(I) + CELL_CUV(I,J,6) *
     &                   ELF(I) * CELL_CUV(I,J,7)
                  ELFY(I) = ELFY(I) + CELL_CUV(I,J,6) *
     &                   ELF(I) * CELL_CUV(I,J,8)
                Endif
             Enddo
             ELFX(I) = ELFX(I) / AREA(I)
             ELFY(I) = ELFY(I) / AREA(I)
          Endif
       Enddo
!$OMP END DO
!$OMP DO
       Do I = IJM_B, IJM_E
C         Do K1 = 1, KBM
C            Do K2 = 1, KBM
C               AAAA(K1,K2) = 0.0
C            Enddo
C         Enddo
           IF(CCM(I) .EQ. 1.0) Then
            Do K = 2, KB
              AAAA(K,K-1) = -DTI * PORE(I,K-1) * (KM(I,K-1) + UMOL) /
     &                      DC(I) ** 2. / DZ(K-1)
              AAAA(K-1,K) = AAAA(K,K-1)
            Enddo
            Do K = 2, KBM
              AAAA(K,K) = DZZ(K-1) * PORE_VF(I,K) - AAAA(K,K-1) - AAAA(K,K+1)
            Enddo
C           If(KBM .GT. 1) Then
C                AAAA(1,1) = DZ(1) - AAAA(1,2) +
C     &                      DTI * (KM(I,1)+UMOL) / DC(I) ** 2. /(.5*DZ(1))
```

$\frac{V_w \Delta t}{D^2 \Delta \sigma}$

$\Delta \sigma - 2 \cdot \frac{V_w \Delta t}{D^2 \Delta \sigma}$

```
C        Else
C           AAAA(1,1) = DZ(1)
C        Endif
ccc           AAAA(1,1) = 1.0
ccc           AAAA(1,2) = 0.0
           Do K = 1, KB
            BBBB(K) = WF(I,K) / AREA(I)
           Enddo
C-----     On the free surface
CC           AAAA(1,1) = DZ(1) -AAAA(1,2)
           AAAA(1,1) = 1.0
           AAAA(1,2) = 0.0
           BBBB(1) = U(I,1) * ELFX(I) * PORE(I,1) + V(I,1) * ELFY(I) *
     &          PORE(I,1) + DC(I) * PORE(I,1) * (ELF(I) - EL(I)) / DTI
           AAAA(KB,KB) = 1.0
           AAAA(KB,KBM) = 0.0
           BBBB(KB) = 0.0
C        If(KBM .GT. 1) Then
C        AAAA(KBM,KBM) = DZ(KBM) - AAAA(KBM,KBM-1) + DTI *
C     &                     (KM(I,KBM) + UMOL) / DC(I) ** 2. / DZZ(KBM)
C        Else
C           AAAA(KBM,KBM) = DZ(KBM) + DTI *
C     &                        (KM(I,KBM)+UMOL) / DC(I) ** 2. / DZZ(KBM)
C        Endif
C        Do K = 1, KBM
C           BBBB(K) = WF(I,K) / AREA(I)
C          Enddo
C          BBBB(1) = BBBB(1) + DTI*(KM(I,1)+UMOL)/DC(I)**2./(.5*DZ(1))*
C     &               (U(I,1) * ELFX(I) + V(I,1) * ELFY(I) +
C     &                DC(I) * (ELF(I) - EL(I)) / DTI)
c          BBBB(1) =
c     &               (U(I,1) * ELFX(I) + V(I,1) * ELFY(I) +
c     &                DC(I) * (ELF(I) - EL(I)) / DTI)
C        BBBB(KBM) = BBBB(KBM) - DHT(I) * DTI * (KM(I,KBM) + UMOL) /
C     &                 DC(I) ** 2. / DZZ(KBM)
C=============================================================================C
C                forward method for velocity solution                         c
c=============================================================================c
         AA(1) = AAAA(1,1)
         If(KBM .GT. 1) AA(2) = AAAA(1,2)
           Do K = 2, KBM
```

```
              AA(2*(K-1)+K-1) = AAAA(K,K-1)
             AA(2*(K-1)+K)   = AAAA(K,K)
              AA(2*(K-1)+K+1) = AAAA(K,K+1)
            Enddo
          If(KBM .GT. 1) AA(3*KB-3) = AAAA(KB,KB-1)
          AA(3*KB-2) = AAAA(KB,KB)
            Do K = 1, KB
             BB(K) = BBBB(K)
          Enddo
            Call ATRDE(AA,KB,3*KB-2,BB,L)
          Do K = 1, KB
               QZ(I,K) = BB(K)
          Enddo
C         VH(1) = -AAAA(1,2) / AAAA(1,1)
C         VHP(1) = BBBB(1) / AAAA(1,1)
C         Do K = 2, KBM
C            VH(K) = -AAAA(K,K+1) / (AAAA(K,K) + AAAA(K,K-1) * VH(K-1))
C            VHP(K) = (BBBB(K) - AAAA(K,K-1) * VHP(K-1)) /
C     &                   (AAAA(K,K) + AAAA(K,K-1) * VHP(K-1))
C         Enddo
C         QZ(I,KBM) = VHP(KBM)
C         Do K = 1, KBM - 1
C            KI = KBM - K
C            QZ(I,KI) = VH(KI) * QZ(I,KI+1) + VHP(KI)
C         Enddo
        Endif
      Enddo
!$OMP END DO NOWAIT
!$OMP END PARALLEL
      Return
      End
```

$Q_{zi}^{*} = A_{iz}^{n\ -1} G_{zi}^{n}$