

Apéndice C

! TITULO: Absorción simultánea de CO₂ y H₂S

!----- Gonzalo Rocha Aguilera

!----- Mayo 2005

! Programa complementario a la tesis que presenta para obtener el título de Maestría

! en Ingeniería Química, en la Universidad de las Américas - Puebla

! DESCRIPCIÓN:

! Este programa calcula las composiciones interfaciales, los coeficientes de transferencia

! y el factor de aceleración para un sistema en el cual la fase gas está formada por tres

! componentes de los cuales uno no se absorbe.

! El programa está basado en el algoritmo propuesto por Al-Ghawas, H. A. y Sandall, O. C.

! 1991, Simultaneous absorption of carbon dioxide, carbonyl sulfide and hydrogen sulfide

! in aqueous methyldiethanolamine. Chem. Engng Sci. 46 665-676

! La documentación del programa puede encontrarse en Rocha G. 2005, Estudio de los fenómenos

! de transporte en el endulzamiento del gas natural. Tesis de Maestría. Universidad de las Américas

! Puebla.

! Antes de usar el presente programa es necesario modificar ciertos valores según el sistema

! a calcular. Los valores son:

! - Ecuación para el cálculo de factor de aceleración E

! - Cálculo de las constantes de Henry H

! - Valores de coeficientes convectivos de transferencia de masa kg y kl

! - Cálculo o valores de presión de vapor P_v

PROGRAM AB_SIM_CO2_H2S

USE MSIMSL

INTEGER cont1, cont2, cont3 !contadores auxiliares

INTEGER valor, cont

REAL suma !sumador auxiliar

REAL y0 (3) !Matriz de composiciones en la fase global gas

REAL y0Prima (2) !Ajuste de y0 para calcular Fluxes totales

REAL yInter (3) !Matriz de composiciones en la interface gas-líquido

REAL yInterPrima (2) !Ajuste de yInter para calcular residuales

REAL yInter1 (2) !yInter de N-1 para método de secante

REAL yInter2 (2) !yInter de N-2 para método de secante

REAL yInterNuevo (2) !Variable auxiliar para cálculos por el método de secante

REAL LambdaG (2, 2) !Matriz tal que LambdaG*Flux difusional=Flux total

REAL Delta !Función Delta de Kronecker

REAL M (2, 2) !Matriz cuyos elementos están dados por $M(i,i)=H_i \cdot CL \div P$

REAL P !Presión

REAL Hi (2) !Constante de Henry del componente i

REAL, PARAMETER :: CL = 18 !Densidad molar de la fase líquida

REAL MBeta (2, 2) !Inverso de la matriz de los coeficientes de transferencia de masa sin reacción

REAL KGlobal (3, 3) !Coeficiente de transferencia global

REAL kG (3, 3) !Matriz de coeficiente de transferencia en la fase gas sin reacción

REAL kL (3, 3) !Matriz de coeficiente de transferencia en la fase líquida, sin reacción

REAL Pv (3) !Presiones de vapor de los componentes puros al equilibrio

REAL xe (3) !Concentraciones interfaciales en el líquido

REAL E (2) !Factor de aceleración

```

REAL kasteriL (2, 2) ! Matriz de coeficientes de transferencia con reacción fase líquida
REAL AL (2,2) !Inverso de la matriz de resistencia en la fase líquida
REAL MInversa (2,2) !Matriz auxiliar para la inversión
REAL MPsi (2,2) !Matriz de factores de corrección

REAL MPsiAnterior (2,2) !Matriz auxiliar para ciclo de factores de corrección
REAL MPorcentaje (2,2) !Matriz auxiliar para ciclo de factores de corrección
REAL MPrecision (2,2) !Matriz auxiliar para ciclo de factores de corrección

REAL Phi (2,2) !Matriz para cálculo de factores de corrección
REAL AG (2,2) !Inverso de matriz de resistencias en la fase gas
REAL AOG (2,2) !Inverso de matriz de resistencias globales
REAL N (2) !Fluxes totales

REAL MAux (2,2) !Matriz auxiliar para cálculos
REAL MAux2 (2,2)
REAL MIdentidad (2,2) !Matriz identidad

REAL ResidualCalculada (2) !Es igual a y0-yI obtenida con el vector de Fluxes totales
REAL ResidualPaso (2) !Es igual a las composiciones en la fase global - la composición interfacial estimada
REAL ResidualTotal (2) !Es igual a ResidualCalculada - ResidualPaso
REAL PrecisionResidual (2) !Auxiliar para ciclo exterior
REAL ResidualTotal1 (2) !ResidualTotal de N-1 para método de secante
REAL ResidualTotal2 (2) !ResidualTotal de N-2 para método de secante
REAL ResidualTotalNuevo (2) !Auxiliar para cálculos del método de la secante

    !Inicialización de matrices
    DO cont1 = 1, 3, 1
        y0(cont1) = 0
        yInter(cont1) = 0
        yInter1(cont1) = 0
        yInter2(cont2) = 0
    END DO
    DO cont1 = 1, 2, 1
        yInter1(cont1) = 0
        yInter2(cont1) = 0
        ResidualTotal1(cont1)=0
        ResidualTotal2(cont1)=0
    END DO

    CALL MatrizIdentidad(MIdentidad)
    CALL MatrizPrecision(MPrecision, PrecisionResidual)

    !Asignación de valores para la constante de Henry
    CALL Henry(Hi)
    !Asignación de valores para kG, kL y KGlobal
    CALL CoefTrans (kG, kL, KGlobal)
    !Asignación de presiones de vapor
    CALL PresionesV (Pv)

    !Lectura de presión
    PRINT *, 'Proporcione la presión: '
    READ *, P

    !Lectura de composiciones en la fase global gas
    PRINT *, 'Proporcione las composiciones en la fase gas global en el siguiente orden: '

```

```

PRINT *, '1- CO2 '
PRINT *, '2- H2S '
PRINT *, '3- CH4 '
CALL lectura(y0)

```

!Suposición de la concentración interfacial y normalización

```

DO cont1 = 1, 2, 1
    yInter(cont1) = y0(cont1) * 0.9
END DO
cont1 = 3
yInter(cont1) = y0(cont1)
suma = 0
DO cont1 = 1, 3, 1
    suma = suma + yInter(cont1)
END DO
DO cont1 = 1, 3, 1
    yInter(cont1) = yInter(cont1) / suma
END DO
!Así, la suma de yInter debe dar 1.

```

!Ajustando yInter para calcular ResidualPaso

```

DO cont1 = 1, 2, 1
    yInterPrima(cont1) = yInter(cont1)
END DO

DO cont2= 1, 2, 1
    DO cont3= 1, 2, 1
        CALL calculaDelta(Delta,cont2, cont3)
        LambdaG(cont2,cont3)=Delta + y0(cont2)/y0(3)
    END DO
END DO

```

!Calculamos la matriz M

```

DO cont2= 1, 2, 1
    DO cont3= 1, 2, 1
        SELECT CASE (cont2-cont3)
            CASE (0)
                M(cont2, cont3) = (Hi(cont2) * CL)/P
            CASE DEFAULT
                M(cont2, cont3) = 0
        END SELECT
    END DO
END DO

```

!Calculamos la matriz MBeta

```

DO cont2= 1, 2, 1
    DO cont3= 1, 2, 1
        SELECT CASE (cont2-cont3)
            CASE (0)
                suma = y0(cont2)/KGlobal(cont2,3)
                DO cont1 = 1, 3, 1
                    suma = suma + (y0(cont1)/kG(cont2,cont1))
                END DO
                MBeta(cont2, cont3) = suma
            CASE DEFAULT

```

```

                suma = 1/kG(cont2,cont3)
                suma = suma - (1/kG(cont2,3))
                MBeta(cont2, cont3) = -y0(cont2)*suma
            END SELECT
        END DO
    END DO

cont = 1
valor = 1
!Calcular concentraciones en la interface lado líquido
DO
    DO cont1 = 1, 3, 1
        xe(cont1) = yInter(cont1)*P/Pv(cont1)
    END DO

    !Calcular factores de aceleración La ecuación requiere ser modificada según el sistema
    !a usar. la ecuación es  $E = \sqrt{D \text{ del gas en el líquido} / \text{Difusividad del reactante en el líquido}}$ 
    !+ (concentración del reactante en el líquido/Concentración de gas en interfase)*
    !sqrt(Difusividad del reactante en el líquido/D del gas en el líquido)
    ; ver Haimour y Sandall (1987)
    DO cont1 = 1, 2, 1
        E(cont1) = (sqrt ((1.3347e-8)/(5.3820e-9))+((141.58/xe(cont1))*sqrt((5.3820e-
9)/(1.3347e-8))))
    END DO

    !Calcular matriz de coeficientes de transferencia con reacción fase líquida
    DO cont2= 1, 2, 1
        DO cont3= 1, 2, 1
            SELECT CASE (cont2-cont3)
                CASE (0)
                    kasteriL(cont2, cont3) = kL(cont2, cont2)*CL*E(cont2)
                CASE DEFAULT
            END SELECT
        END DO
    END DO

    !Calcular la matriz AL
    !AL = .INV. kasteriL .x. M
    CALL LINRG(2, kasteriL, 2, MInversa, 2)
    AL = MATMUL (M, MInversa)

    !Iteración para obtener la matriz de factores de corrección
    !Calcular Matriz Phi
    CALL LINRG(2, LambdaG, 2, MInversa, 2)
    AG = MATMUL (MBeta, MInversa)

    !Calcular AOG
    AOG = AG + AL

    !Ajustando y0 para calcular N
    DO cont1 = 1, 2, 1
        y0Prima(cont1) = y0(cont1)
    END DO

    CALL LINRG(2, AOG, 2, MInversa, 2)
    N = MATMUL (MInversa, y0Prima)      !N es igual a la inversa de AOG por y0

```

```

!Calculando Phi
DO cont2= 1, 2, 1
    DO cont3= 1, 2, 1
        SELECT CASE (cont2-cont3)
            CASE (0)
                suma = N(cont2)/kG(cont2,3)
                DO cont1 = 1, 2, 1
                    suma = suma + (N(cont1)/kG(cont2,cont1))
                END DO
                Phi(cont2, cont3) = suma
            CASE DEFAULT
                suma = 1/kG(cont2,cont3)
                suma = suma - (1/kG(cont2,3))
                Phi(cont2, cont3) = -N(cont2)*suma
        END SELECT
    END DO
END DO

!Calculando MPsi
MAux = EXP(Phi)- MIdentidad
CALL LINRG(2, MAux, 2, MAux2, 2)
MPsi = MATMUL (Phi, MAux2)

!Una vez obtenida la primera matriz Psi, iteramos hasta que converja
DO
    MPsiAnterior = MPsi
    CALL LINRG(2, LambdaG, 2, MInversa, 2)
    AG = MATMUL (MBeta, MInversa)
    AG = MATMUL (AG, MPsi)
    !Calcular AOG
    AOG = AG + AL
    !Ajustando y0 para calcular N
    DO cont1 = 1, 2, 1
        y0Prima(cont1) = y0(cont1)
    END DO
    CALL LINRG(2, AOG, 2, MInversa, 2)
    N = MATMUL (MInversa, y0Prima)
    !Calculando Phi
    DO cont2= 1, 2, 1
        DO cont3= 1, 2, 1
            SELECT CASE (cont2-cont3)
                CASE (0)
                    suma = N(cont2)/kG(cont2,3)
                    DO cont1 = 1, 2, 1
                        suma = suma + (N(cont1)/kG(cont2,cont1))
                    END DO
                    Phi(cont2, cont3) = suma
                CASE DEFAULT
                    suma = 1/kG(cont2,cont3)
                    suma = suma - (1/kG(cont2,3))
                    Phi(cont2, cont3) = -N(cont2)*suma
            END SELECT
        END DO
    END DO

```

```

        END DO
        !Calculando MPsi
        MAux = EXP(Phi)- MIdentidad
        CALL LINRG(2, MAux, 2, MAux2, 2)
        MPsi = MATMUL (Phi, MAux2)
        MPorcentaje = ABS((MPsi - MPsiAnterior)/MPsiAnterior)
        CALL ComparaMatrices (MPorcentaje, MPrecision, valor)
        IF (valor<1) EXIT

        DO cont2= 1, 2, 1
            DO cont3= 1, 2, 1
                PRINT *, MPorcentaje(cont2, cont3)
            END DO
        END DO
    END DO

    !Cálculo de residuales
    ResidualCalculada = MATMUL(AG, N)

    ResidualPaso = y0Prima - yInterPrima

    ResidualTotalNuevo = ResidualCalculada - ResidualPaso
    CALL CompararPresRes (ResidualTotalNuevo, PrecisionResidual, valor) !cuando valor = 0
puedo salir del ciclo
    IF (valor <1) EXIT

    ResidualTotal2 = ResidualTotal1
    ResidualTotal1 = ResidualTotal
    ResidualTotal = ResidualTotalNuevo

    IF (cont < 3) THEN
        yInterNuevo = yInterPrima + 0.01
    ELSE
        yInterNuevo = yInter1 - ResidualTotal1 * (ResidualTotal1-ResidualTotal2)/(yInter1-yInter2)
    END IF
    yInter2 = yInter1
    yInter1 = YInterPrima
    yInterPrima = yInterNuevo

    cont = cont + 1

END DO

Print *, 'Composiciones en la interfase lado gas'
Print *, ' CO2      H2S      CH4'
Print *, Yinter
Print *, 'Composiciones en la interfase lado liquido'
Print *, ' CO2      H2S      CH4'
Print *, Xe
Print *, 'Factores de Aceleracion'
Print *, 'CO2'
Print *, 'H2S'
DO cont3= 1, 2, 1
    PRINT *, E (cont3)
END DO
! En esta sección se pueden imprimir los resultados que se crean relevantes

```

```

        PAUSE 'Presione <Enter> para terminar'

END PROGRAM

SUBROUTINE calculaDelta(d, c2, c3)
!Calcula la función Delta de Kronecker
REAL d
INTEGER c2, c3
!Si ambas variables son iguales, d=1
        SELECT CASE (c2-c3)
            CASE (0)
                d = 1
            CASE DEFAULT
                d = 0
        END SELECT
END SUBROUTINE

SUBROUTINE lectura(a)
REAL a (3)
    READ *, a
END SUBROUTINE

SUBROUTINE Henry (H)
!Asignación de valores para la constante de Henry. Modificar según el sistema a calcular
REAL H (2)
    H(1) = 0.15
    H(2) = 0.15
END SUBROUTINE

SUBROUTINE CoefTrans (kG, kL, KGlobal) !Esta subrutina calcula los coeficientes de transferencia
REAL kG (3, 3)
REAL kL (3, 3)
REAL KGlobal (3, 3)
    kG(1,1) = 0.241
    kG(1,2) = 0.001
    kG(1,3) = 0.001
    kG(2,1) = 0.001
    kG(2,2) = 0.241
    kG(2,3) = 0.001
    kG(3,1) = 0.001
    kG(3,2) = 0.001
    kG(3,3) = 0.001

    kL(1,1) = 0.205
    kL(1,2) = 0.001
    kL(1,3) = 0.001
    kL(2,1) = 0.001
    kL(2,2) = 0.205
    kL(2,3) = 0.001
    kL(3,1) = 0.001
    kL(3,2) = 0.001
    kL(3,3) = 0.205

    KGlobal = 1/((1/kG)+(1/kL))
END SUBROUTINE

```

```

SUBROUTINE PresionesV (Pv) !Esta subrutina fija las presiones de vapor a las condiciones de
REAL Pv(3)                                !operación. Modificar según el sistema a calcular
    Pv (1) = 8.4581
    Pv (2) = 8.4581
    Pv (3) = 8.4581
END SUBROUTINE

```

```

SUBROUTINE MatrizIdentidad(M) !Esta subrutina asigna la matriz identidad
REAL M(2,2)
    M(1,1) = 1
    M(1,2) = 0
    M(2,1) = 0
    M(2,2) = 1
END SUBROUTINE

```

```

SUBROUTINE MatrizPrecision(M, PR) !Esta subrutina asigna la precisión con la cual se
REAL M(2,2)                                !detienen las iteraciones
REAL PR (2)
    M(1,1) = 0.001
    M(1,2) = 0.001
    M(2,1) = 0.001
    M(2,2) = 0.001

    PR(1) = 0.0005
    PR(2) = 0.0005
END SUBROUTINE

```

```

SUBROUTINE ComparaMatrices (MPorcentaje, MPrecision, valor) !Esta subrutina decide si se continua
REAL MPorcentaje (2,2)
    !iterando o se detiene el ciclo
REAL MPrecision (2,2)
INTEGER valor, c1, c2
    valor = 0
    DO c1 = 1, 2, 1
        DO c2 = 1, 2, 1
            IF (MPorcentaje(c1,c2) > MPrecision(c1,c2)) valor=1
        END DO
    END DO
END SUBROUTINE

```

```

SUBROUTINE CompararPresRes (PR, MP, valor) !Esta subrutina decide si se continua
REAL PR (2)                                !iterando o se detiene el ciclo
REAL MP (2)
INTEGER valor, c1
    valor = 0
    DO c1 = 1, 2, 1
        IF (PR(c1) > MP(c1)) valor=1
    END DO
END SUBROUTINE

```