CHBE 553 - Assignment 2 – 01/24/2019

Chemical equilibrium:

The chemical equilibrium state describes concentrations of reactants and products in a reaction taking place in a closed system, which no longer changes with time. The equilibrium state is said to be dynamic, meaning that the reaction is continuously in motion. This consistency, however, does not mean that the reactions have stopped, but rather that the rates of the two opposing reactions have become equal.

At the beginning of the reaction, the rate forward reaction is greater than the rate of reverse reaction, as the reaction consists of only pure reactants. As reactants are converted to products, and it is only until a large enough concentration of products are available, that the reverse reaction becomes a factor. With sufficient time, both these rates become equal reaching the equilibrium state (fig1).

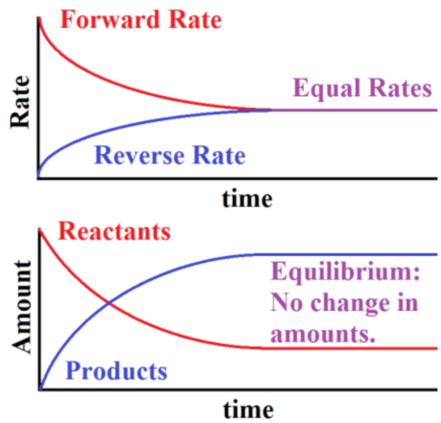


Fig 1: chemical equilibrium state [1]

After this point is reached the concentrations of each species in the reactor becomes fixed and displays no further propensity to change unless propelled by any externally imposed "disturbance" (say, by the provision of heat) [2]. The magnitude of all measurable macroscopic variables (T, P, and composition) characterizing the reaction remains constant.
Clearly, under the equilibrium state, the percentage conversion of the reactants to products must be the maximum possible at the given temperature and pressure. Or else the reaction would progress further until the state of equilibrium is achieved. The choice of the reaction conditions thus depends on the maximum (or equilibrium) conversion possible. Further, the knowledge of equilibrium conversions is essential to the intensification of a process. Finally, it also sets the limit that can never be crossed in practice regardless of the process strategies. This forms a primary input to the determination of the economic viability of a manufacturing process [2].

MATLAB CODE:

Consider two reactions in a batch reactor:
$2A + B \leftrightarrow C$, $K_1 = 5 \times 10^{-4}$
$A + D \leftrightarrow C$, $K_2 = 4 \times 10^{-2}$

Given the equilibrium concentration for the reactants:

$C_A = C_{A0} – 2x_1C_{B0} – x_2C_{D0}$,
$C_B = C_{B0} – x_1C_{B0}$,
$C_C = C_{C0} + x_1C_{B0} + x_2C_{D0}$,
$C_D = C_{D0} – x_2C_{D0}$

The initial concentrations are:
$C_{A0} = 40$ kmol/m^3,
$C_{B0} = 15$ kmol/ m^3
$C_{C0} = 0$,
$C_{D0} = 10$ kmol/ m^3

Given the equilibrium constant:
$K_1 = C_C / C_A^2C_B = 5 \times 10^{-4}$
$K_2 = C_C / C_AC_D = 4 \times 10^{-2}$

The equilibrium constant is related to the forward and backward rate constants, k_f and k_r of the reactions involved in reaching equilibrium as [3],

$$K = k_f/k_r \qquad (1)$$

For reaction 1,
$C_C = 5 \times 10^{-4} \times C_A^2C_B$
For reaction 2,
$C_C = 4 \times 10^{-2} \times C_AC_D$

These two equations provide two nonlinear equations with two unknowns.
$Eqn_1 = f_1 = C_C - 5 \times 10^{-4} \times C_A^2C_B = 0$ (2)
$Eqn_2 = f_2 = C_C - 4 \times 10^{-2} \times C_AC_D = 0$ (3)

For solving these two non-linear equations, Newton’s iterative method is used [4] [5].

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \mathbf{J}^{-1}(\mathbf{x}_n)F(\mathbf{x}_n) \qquad (4)$$

Where,

$$\mathbf{J_f}(x,y) = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix} \qquad (5)$$

Central difference formula was used to evaluate the partial derivatives [6]:
Given by:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}. \qquad (6)$$

Newton's method, calculates the new value of x, by calculating the Jacobian matrix 'J' and function 'f' at the old value of x. The iteration is continued until the difference between the new value of x and the old value of x is less significant (<machine epsilon). For fast convergence, initial guess must be chosen appropriately that is close to the actual value of x. The initial guess for conversion of the system is chosen between 0 and 1. In the first trial, 1 was chosen as the initial guess for both x_1 and x_2. The system didn't converge even after 100 iterations. In the next trails, the conversion was assumed to be 0, 0.1 and 0.5. It was seen that the system converges after 6, 5 and 7 iterations respectively. Hence, 0.1 was chosen as the initial guess, as it provided faster convergence.

The MATLAB code consists of 3 blocks:

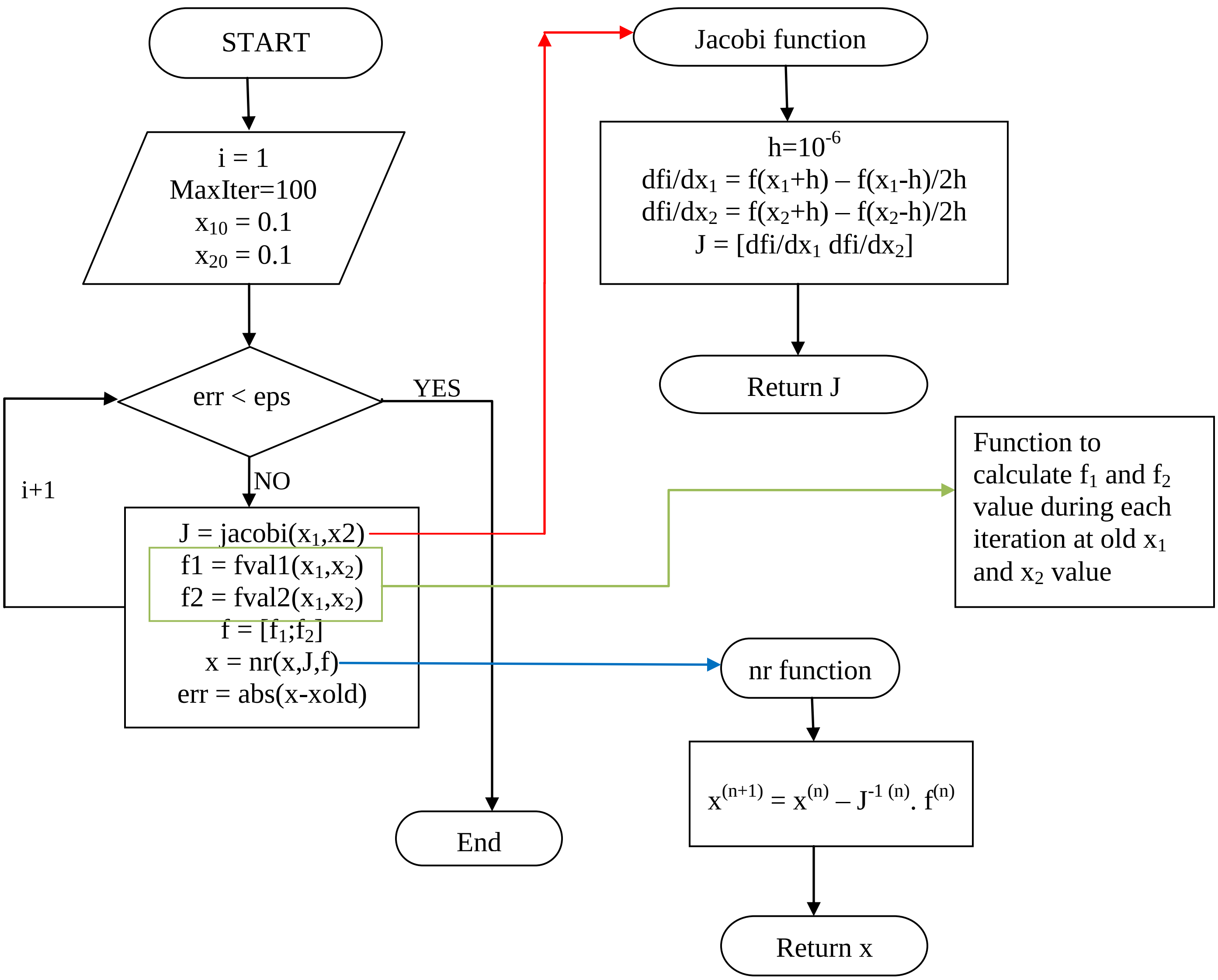
1. **Function evaluation block**: to evaluate 'f' at each x value.
2. **Jacobian block**: to perform central difference to evaluate the value of derivatives and construct the Jacobi matrix
3. **Newton's solver block**: to evaluate the new value of x using Newton's method

The following conversion values were obtained after convergence:
x_1 = 0.1203
x_2 = 0.4787

REFERENCES:

1. https://socratic.org/questions/what-determines-when-a-system-reaches-equilibrium-what-observations-can-be-made- (date accessed: 01/24/2019)
2. https://nptel.ac.in/courses/103101004/downloads/chapter-8.pdf (date accessed: 01/24/2019)
3. https://en.wikipedia.org/wiki/Equilibrium_constant (date accessed: 01/24/2019)
4.http://support.sas.com/documentation/cdl/en/imlug/66112/HTML/default/viewer.htm#imlug_genstatexpls_sect004.htm (date accessed: 01/24/2019)
5. https://en.wikipedia.org/wiki/Jacobian_matrix_and_determinant (date accessed: 01/24/2019)
6. http://mathfaculty.fullerton.edu/mathews/n2003/differentiation/numericaldiffproof.pdf (date accessed: 01/24/2019)

FLOWCHART:


START
i = 1
MaxIter=100
x10 = 0.1
x20 = 0.1
err < eps
YES
NO
i+1
J = jacobi(x1,x2)
f1 = fval1(x1,x2)
f2 = fval2(x1,x2)
f = [f1;f2]
x = nr(x,J,f)
err = abs(x-xold)
End
Jacobi function
h=10-6
dfi/dx1 = f(x1+h) – f(x1-h)/2h
dfi/dx2 = f(x2+h) – f(x2-h)/2h
J = [dfi/dx1 dfi/dx2]
Return J
Function to calculate f1 and f2 value during each iteration at old x1 and x2 value
nr function
x(n+1) = x(n) – J-1 (n). f(n)
Return x