

Práctica 4: Sistemas de ecuaciones no lineales.

1 Introducción.

Sea un conjunto de ecuaciones de la forma:

$$f_i(x_1, x_2, \dots, x_N) = 0, \quad i = 1, 2, \dots, N \quad (1)$$

o en notación matricial:

$$\mathbf{f}(\mathbf{x}) = 0 \quad (2)$$

cuya solución es el vector $\mathbf{X} = (X_1, X_2, \dots, X_N)$. Para resolver un sistema de ecuaciones no lineales vamos a emplear la generalización del método de Newton-Raphson. Se cumple:

$$\mathbf{f}(\mathbf{x}) \simeq \mathbf{f}(\mathbf{x}^k) + \mathbf{F}(\mathbf{x} - \mathbf{x}^k) \quad (3)$$

siendo \mathbf{x}^k una aproximación a la raíz buscada y \mathbf{F} , la matriz jacobiana:

$$\mathbf{F}_{ij} = \frac{\partial f_i}{\partial x_j} \quad (4)$$

Despejando \mathbf{x} se obtiene la fórmula iterativa:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{F}^{-1}(\mathbf{x}^k)\mathbf{f}(\mathbf{x}^k) \quad (5)$$

Sistemas altamente no lineales pueden diverger rápidamente. Para evitarlo se introduce un factor de relajación ρ ($0 < \rho < 1$) en la ecuación (5) en la forma:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \rho\mathbf{F}^{-1}(\mathbf{x}^k)\mathbf{f}(\mathbf{x}^k) \quad (6)$$

Este algoritmo se ha programado en el archivo **Newt.m**. Abrir este archivo y comprobar que se ha programado la expresión (6). Contestar la primera pregunta de la hoja de resultados.

Como primera aplicación del método consideramos el sistema de dos ecuaciones:

$$\begin{aligned} f_1(x, y) &= x^2 - y - 0.2 = 0 \\ f_2(x, y) &= y^2 - x - 0.3 = 0 \end{aligned} \quad (7)$$

Los programas **f1.m** y **f2.m** evalúan los primeros miembros de (7). Se dispone asimismo del programa **curvas.m** para representar gráficamente estas funciones; éste hace uso del comando MATLAB **contour(x,y,z)** que dibuja curvas de nivel de una función $z = f(x, y)$ y en él hemos seleccionado los contornos con $z = 0$ para representar las curvas definidas en (7). Escribimos:

```
curvas ('f1', 'f2', -2, 2)
```

que nos proporcionará la figura de las curvas correspondientes a (7). De la figura podemos estimar que las soluciones del sistema son aproximadamente (-0.2,-0.2) y (1.2,1.2), que emplearemos como valores de partida para resolver numéricamente el sistema (7). Empleamos el programa **Newt**:

```
x0=[-0.2,-0.2];  
tol=1e-4;  
[x,iter]=Newt('sistpru', x0, tol)
```

Donde:

- **x** es la solución buscada.
- **iter** es el número de iteraciones efectuadas.
- **sistpru** es el nombre de la función donde se evalúan las funciones $f_{1,2}$.
- **x0** es el *initial guess*.
- **tol** es la tolerancia.

Repetir con $x_0=(1.2,1.2)$ y copiar las soluciones en la hoja de resultados.

Como segundo ejemplo, considerar el sistema de ecuaciones:

$$\begin{aligned}x^2 + y^2 - 2 &= 0 \\ xy - 1 &= 0\end{aligned}\tag{8}$$

Sustituyendo en (8), se comprueba que (1,1) es una solución de este sistema. Crear una función **sistpru2.m**, modificando la función **sistpru** y comprobar el funcionamiento del método numérico:

```
x0=[1,1]  
tol=1e-4  
[x1,iter]=Newt('sistpru2', x0, tol)
```

Discutir el resultado (pregunta 3 de la hoja de resultados).

2 Ejemplo de aplicación: Equilibrio químico en un sistema con dos reacciones.

Como aplicación, consideramos los equilibrios químicos



cuyas constantes de equilibrio son:

$$k_1 = \frac{C_C}{C_A^2 C_B}; k_2 = \frac{C_C}{C_A C_D} \quad (11)$$

Consideremos el equilibrio (9), y llamemos x_1 a la fracción molar de B que reacciona. Para una concentración inicial C_{B0} , la concentración de B en el equilibrio es:

$$C_B = C_{B0}(1 - x_1) \quad (12)$$

Análogamente, para el equilibrio (10), llamamos x_2 a la fracción molar de D que reacciona, con lo cual la concentración de D es

$$C_D = C_{D0}(1 - x_2) \quad (13)$$

El número de moles de C presentes en la mezcla de equilibrio es igual al número inicial de moles más los formados en ambas reacciones:

$$C_C = C_{C0} + C_{B0}x_1 + C_{D0}x_2 \quad (14)$$

y, por último:

$$C_A = C_{A0} - 2C_{B0}x_1 - C_{D0}x_2 \quad (15)$$

donde C_{i0} en las ecuaciones (12)-(15) son las concentraciones iniciales. Sustituyendo (12)-(15) en (11) obtenemos:

$$\begin{aligned} k_1 - \frac{C_{C0} + C_{B0}x_1 + C_{D0}x_2}{(C_{A0} - 2C_{B0}x_1 - C_{D0}x_2)^2 C_{B0}(1 - x_1)} &= 0 \\ k_2 - \frac{C_{C0} + C_{B0}x_1 + C_{D0}x_2}{(C_{A0} - 2C_{B0}x_1 - C_{D0}x_2) C_{D0}(1 - x_2)} &= 0 \end{aligned} \quad (16)$$

Las concentraciones en el equilibrio de las cuatro sustancias se obtienen resolviendo el sistema de ecuaciones (16). Para ello disponemos de los siguientes programas:

- **dosreac.m**: lee los datos del problema (C_{i0} , $k_{1,2}$, valores iniciales de las variables $x_{1,2}$).
- **Newt.m**: resuelve un sistema de ecuaciones empleando el método de Newton-Raphson.
- **equilib.m**: programa que evalúa los primeros miembros de las ecuaciones (16).

Basta escribir:

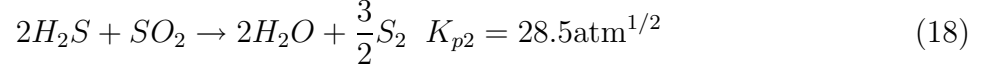
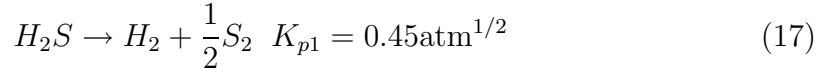
dosreac

para que comience el proceso de resolución. El programa irá pidiendo los datos que requiere para resolver el problema. Probar los siguientes conjuntos de valores:

- Concentraciones iniciales: **[40,15,0,10]**
- **K1=5e-4, K2=4e-2**
- Aproximación inicial: **[0.1,0.9]**
- El archivo que calcula los primeros miembros del sistema de ecuaciones es **'equilib'**
- Repetir con aproximaciones iniciales: **[0.1,0.5]**, **[0.5, 0.5]** y **[0.1,0.1]**

3 Aplicación: Reacciones en fase gas

Resolver el siguiente problema: A alta temperatura y baja presión H_2S y SO_2 experimentan las siguientes reacciones en fase gas:



La mezcla gaseosa inicial contiene 45 % H_2S , 25% SO_2 y el gas inerte N_2 con una presión total de 1.2 atm. Calcular las fracciones molares de equilibrio de todos los componentes a una presión constante de 1.2 atm. Seguimos los siguientes pasos:

1. Las constantes de equilibrio tienen la forma:

$$K_{p1} = \frac{P_{H_2} P_{S_2}^{1/2}}{P_{S_2H}} = \frac{n_{H_2} n_{S_2}^{1/2}}{n_{S_2H}} n^{-1/2} P^{1/2}$$

$$K_{p2} = \frac{P_{H_2O}^2 P_{S_2}^{3/2}}{P_{S_2H}^2 P_{SO_2}} = \frac{n_{H_2O}^2 n_{S_2}^{3/2}}{n_{S_2H}^2 n_{SO_2}} n^{-1/2} P^{1/2} \quad (19)$$

siendo n_i el número de moles de cada componente, n el número total de moles y P la presión total.

2. Escribimos el número de moles de cada componente y el número total de moles en términos de los avances x_1 , x_2 de las reacciones:

$$n_{H_2S} = n_{H_2S}^0 - n_{H_2S}^0 x_1 - 2n_{SO_2}^0 x_2 \quad (20)$$

$$n_{SO_2} = n_{SO_2}^0 (1 - x_2) \quad (21)$$

$$n_{H_2} = n_{H_2S}^0 x_1 \quad (22)$$

$$n_{S_2} = \frac{1}{2} n_{H_2S}^0 x_1 + \frac{3}{2} n_{SO_2}^0 x_2 \quad (23)$$

$$n_{H_2O} = 2n_{SO_2}^0 x_2 \quad (24)$$

$$n_{N_2} = n_{N_2}^0 \quad (25)$$

3. Expresamos las constantes de equilibrio K_p en función de las dos variables $x_{1,2}$, como en el ejemplo anterior.
4. Hemos modificado el programa **equilib**, escribiendo las ecuaciones anteriores y el nuevo programa se denomina **equilib2**. Este programa comienza con la sentencia

function f = equilib2(x, n0, K1, K2, P)

donde $n0$ es un vector de 6 componentes que contiene los números de moles iniciales de los seis gases (SH_2 , H_2 , S_2 , SO_2 , H_2O y N_2) que constituyen la mezcla de reacción; $K1$ y $K2$ son las constantes de equilibrio y P la presión total.

Para acelerar la convergencia del método se emplean formas cercanas a expresiones lineales; por ejemplo, si:

$$K = \frac{f(x_1, x_2)}{g(x_1, x_2)}, \quad (26)$$

es más conveniente escribirla en la forma:

$$Kg(x_1, x_2) - f(x_1, x_2) = 0 \quad (27)$$

que en la forma:

$$K - \frac{f(x_1, x_2)}{g(x_1, x_2)} = 0 \quad (28)$$

5. Ejecutar el programa de la siguiente forma:

n0= [45, 25, 0, 0, 0, 30] (El orden de los valores debe ser coherente con la programación y estamos suponiendo que las dos primeras componentes son los números de moles iniciales de H_2S y SO_2 , respectivamente, y la sexta componente el número de moles del gas inerte).

K1= 0.45

K2= 28.5

P= 1.2

Tomamos:

x0=[0,0.3]

[x,iter]=Newt('equilib2',x0, 1.e-4, n0, K1, K2, P)

Repetir el cálculo con los valores iniciales: **x0=[0.3,0.3]**, **x0=[0.3,0.7]**, **x0=[0.5,0.5]**, **x0=[0.7,0.3]** y con **n0= [25, 45, 0, 0, 0, 30]**

4 Práctica 4 Resultados.

NOMBRE Y APELLIDOS:

1. Copiar la instrucción MATLAB que corresponde a la fórmula (6) en el programa **Newt.m**. ¿Cómo se calcula la matriz jacobiana en este programa? ¿Cuál es el valor del factor de relajación que se ha empleado?
2. Copiar los valores obtenidos para las raíces del sistema (7)
3. Al emplear el método de Newton para obtener la raíz (1,1) del sistema (8) aparece un mensaje de precaución (*warning*) ¿A qué se debe?

