Promen 2, Problema 3
Embalse retardador de crecidas
Métodos Numéricos
ITBA

Primer autor:
Conrado Mader Blanco - Legajo 51270 (Comisión S, lunes)
cmaderbl@alu.itba.edu.ar

Co-autores:
Betina Cynthia Mamani - Legajo 52310 (Comisión B, lunes)
Federico Ramundo - Legajo 51596 (Comisión S, lunes)

1. Enunciado completo del problema

1.1. Determinación del problema

La determinación del problema se realiza de acuerdo a las pautas determinadas por la cátedra. Para ello, se ejecuta el comando: octave nro_prob.m

El mismo determina que el problema a realizar es el número 3.

1.2. Enunciado

Suponga un embalse que supondremos tiene como principal propósito actuar como retardador de crecidas en un curso de agua. Al ingresar agua al embalse por algún afluente o una tormenta, el nivel de agua en el mismo asciende y si se excede cierta cota H entonces se produce un caudal de salida desde el embalse hacia aguas abajo. Se conoce el flujo de entrada al embalse en función de tiempo (lo denominaremos hidrograma de entrada $E(t)$). Para el hidrograma de salida $S(t)$ (flujo de salida del embalse hacia aguas abajo) se tiene una estructura (el vertedero) en la que $S(t)$ depende de la diferencias de alturas $h?H$, donde ambas alturas se miden desde el punto mas profundo del embalse siendo h la distancia entre la superficie libre y ese punto de mayor profundidad. Del embalse se conoce el área $A(z)$ de cada sección horizontal del mismo como función de la distancia z al punto mas profundo. Se puede plantear un modelo elemental de flujos de entrada y salida para obtener el siguiente problema de valor inicial del cual es solución $h(t)$:

$$h(t) = \frac{(E(t) - S(t))}{A(h(t))}$$
$$t \in [0, T], h(0) = h0$$

De las funciones $A(h)$ y $E(t)$ se dispondrá solo la información de una tabla de valores en las que se podrá interpolar en el caso que hiciesen falta valores no registrados. Se supondrá que

$$S(t) = \begin{cases} C\sqrt{(h(t) - H)^3} & \text{si } h(t) > H \\ 0 & \text{si } h(t) < H \end{cases}$$

con C una constante positiva.

1. Se desea analizar el comportamiento del embalse como atenuador del pico de crecida y como retardador de la aparición de dicho pico. Supondremos conocido el flujo de entrada $E(t)$ con t en miles de segundos y E en m^3/s como indica la siguiente tabla:

t	0.0	1.8	3.6	5.4	7.2	9.0	10.8	12.6	14.4	16.2	18.0
E	0	30	150	400	500	460	350	230	130	60	10

De la función $A(h)$ se conoce la siguiente tabla de valores con h en metros y A en miles de m^2:

h	0.0	2.5	5.0	7.5	10.0	12.5	15.0	17.5	20.0	22.5
A	0.0	0.1	1.4	7.0	18.5	42.0	80.0	140.0	230.0	330.0

h	25.0	27.5	30.0	32.5	35.0
A	480.0	700.0	1000.0	1100.0	1600.0

Obtener $h(t)$ integrando el problema de valor inicial con el método de Runge-Kutta de orden 4 suponiendo un paso de 100 segundos con $T =$ 30000 segundos si $H = 30$ metros, $C = 30$(en el MKS) y $h0 = H$. Para evaluar las funciones E y A se sugiere usar el procedimiento $interp1$ en Octave para implementar una interpolación por splines. Para t mayor que 18000 segundos suponga que $E(t)$ es cero.

2. Representar gráficamente $h(t)$ identificando su valor máximo y el instante en que se produce.

3. En un mismo gráfico representar $E(t)$ y $S(t)$ y comentar el efecto que produce el embalse atenuando y retardando el fenómeno de crecida. Con los resultados obtenidos estimar la razón R entre los picos de E y S así como el retardo D entre la aparición de ambos picos.

4. Completar una tabla con los valores de R y D si los datos de entrada de E se multiplican por: 0.5, 0.75, 1 (que fueron obtenidos),1.25 y 1.5.

2. Planteo del problema y resultados obtenidos

2.1. item 1

Utilizando el método de Runge-Kutta de orden 4 obtuvimos una aproximación a $h(t)$ cuyos valores no mostramos a continuaci'on dado que al aproximar de $t = 0$ a 30000 con un salto de 100 obtuvimos una tabla de 301 entradas, lo cual resulta muy extenso. El gráfico pertinente a $h(t)$ puede verse en el siguiente item.

2.2. item 2

El gráfico de la aproximación de $h(t)$ puede verse en la figura 1.

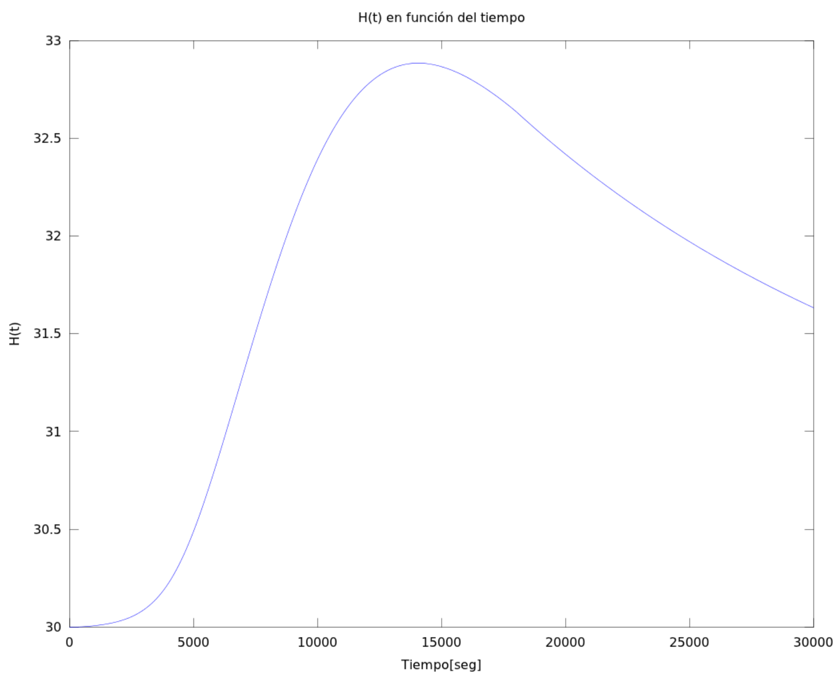

Figura 1: Gráfico de la aproximación de $h(t)$ en función del tiempo.

Los resultados que obtuvimos respecto al máximo de $h(t)$ y al tiempo en el que ocurre son:

$$\text{máximo} = 32{,}884m$$
$$\text{instante} = 14100seg$$

2.3. item 3

En la figura 2 pueden observarse graficadas las funciones $E(t)$ y $S(t)$.

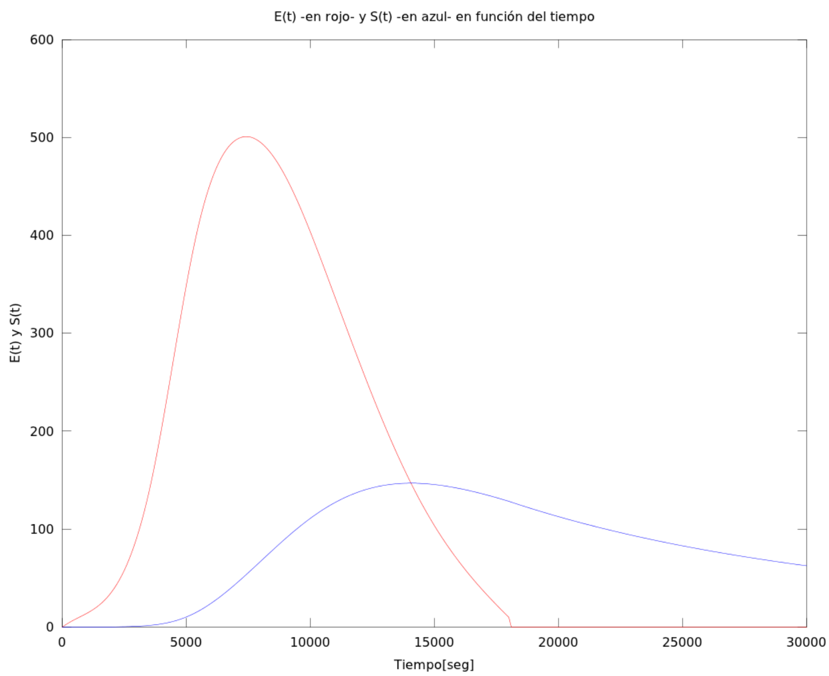

Figura 2: Gráfico de $E(t)$ y $S(t)$ en función del tiempo.

El efecto que produce el embalse atenuando y retardando el fenḿeno de crecida, como se observa en el gráfico, es una fuerte subida en la entrada del embalse ($E(t)$) seguida de una fuerte caída, mientras que la salida ($S(t)$) no describe un pico tan pronunciado y lo alcanza con cierto retarde respecto de la entrada (dado que para que salga agua del embalse primero debe entrar).
Mientras que el máximo de la salida no es tan elevado como el de la entrada, su crecimiento y su decrecimiento son más suaves si se comparan con los de la entrada ($E(t)$).
Cabe destacar que tanto en al entrada como en al salida, una vez alcanzado el pico, la pendiente de decrecimiento es menos pronunciada que en la de crecimiento.
La razón R entre los picos de E y S, y el retardo D entre ambos picos, según los cálculos obtenidos es el siguiente:

$$R = 3{,}4088$$
$$D = 6700 seg$$

2.4. item 4

Multiplicando $E(t)$ por los siguientes factores, los resultados que obtuvimos de la relación R entre los picos de $E(t)$ y $S(t)$ y la distancia D corresponden a la siguiente tabla:

Factor	R	$D[seg]$
0.50	1.7044	6700
0.75	2.5566	6700
1.00	3.4088	6700
1.25	4.2610	6700
1.50	5.1133	6700

3. Códigos Octave

```
function ret = nro_prob( legajos )

%
% Determina el n\'umero de problema a
% realizar por el grupo
%

    ret = rem( min(legajos), 4 ) + 1;

end

disp( nro_prob( [ 51270 52310 51596 ] ) );
```

- Ítem 1

```
%
% Interpola por spline utilizando el método
%  interp1 sugerido
%

function y = interpolar(X, Y, x)

    [y] = interp1(X, Y, [x], 'spline');

end

%
% Evalua la función E de entrada de agua
% al embalse
%

function y = E(t, factor)

    if(t > 18000)
        y = 0;

    else
        y = interpolar([0: 1800: 18001], [0, 30, 150,
              400, 500, 460, 350, 230, 130, 60, 10] .*
              factor, t);
    end
```

```
end

%
% Evalua la función A de sección horizontal
% del embalse
%

function y = A(t)

    a = [0: 2.5: 35];
    b = [0, 100, 1400, 7000, 18500, 42000, 80000,
        140000, 230000, 330000, 480000, 700000,
        1000000, 1100000, 1600000];

    y = interpolar(a, b, t);

end

%
% Evalua la función S de salida de agua del embalse
%

function y = S(h, h0, C)

    if(h <= h0)
        y = 0;

    else
        y = C * (h - h0) ^ (3/2);
    end

end

%
% Función f(t_k, y_k) utilizada por el método de
% Runge-Kutta para aproximar h(t).
% Equivale a la función h'(t) del enunciado.
%

function y = f(tk, yk, h0, C, factor)

    y = (E(tk, factor) - S(yk, h0, C)) / A(yk);

end

%
```

```
% Ejecuta el método de Runge-Kutta
% de orden 4
%

function Y = runge_kutta4(ti, tf, step, h0, C, factor
    )

    Y = [h0];

    for t = ti:step:(tf - step)

        yk = Y(length(Y));

        k1 = step * f(t, yk, h0, C, factor);
        k2 = step * f(t + step/2, yk + k1/2, h0, C,
            factor);
        k3 = step * f(t + step/2, yk + k2/2, h0, C,
            factor);
        k4 = step * f(t + step, yk + k3, h0, C,
            factor);

        Y = [Y, yk + (k1 + 2*k2 + 2*k3 + k4)/6];

    end

end
```

- Ítem 2

```
%
% Grafica la función h(t)
%

function graficar_item2(H, T, nombre)

    plot(T, H);
    title('H(t) en funci\'on del tiempo');
    xlabel('Tiempo[seg]');
    ylabel('H(t)');

    print(nombre, '-dpng');

end

%
```

```
% Calcula el maximo en h(t)
%

function [M, tM] = calcular_maximo(H, T)

    [M, tM] = max(H);

    tM = T(tM);

end
```

- Ítem 3

```
source("item1.m");

%
% Calcula dos vectores
% equivalentes a E(t) y a S(t)
%

function [vectorE, vectorS] = calcular_vectores(H, T,
    C, factor)

    vectorS = [];

    for h=H

        vectorS = [vectorS, S(h, H(1), C)];

    end

    vectorE = [];

    for t=T

        vectorE = [vectorE, E(t, factor)];

    end

end

%
% Grafica E(t) y S(t)
%

function graficar_item3(H, T, C, nombre, factor)
```

```
    [vectorE, vectorS] = calcular_vectores(H, T, C,
        factor);

    plot(T, vectorE, 'r', T, vectorS, 'b');
    title('E(t) -en rojo- y S(t) -en azul- en funci\'
        on del tiempo');
    xlabel('Tiempo[seg]');
    ylabel('E(t) y S(t)');

    print(nombre, '-dpng');

end

%
% Calcula los picos y el retardo entre ellos
%

function [R, D] = calcular_picos(H, T, C, factor)

    [vectorE, vectorS] = calcular_vectores(H, T, C,
        factor);

    [maxE, posMaxE] = max(vectorE);
    [maxS, posMaxS] = max(vectorS);

    R = maxE / maxS;

    D = abs(T(posMaxE) - T(posMaxS));

end
```

- Ítem 4

```
source("item3.m");

%
% Crea la tabla con los valores de R y D
% multiplicando E por 0.5, 0.75, 1, 1.25 y 1.5
%

function tabla_item4(H, T, C)

    tabla = [];

    for factor = 0.5:0.25:1.5
```

```
        [R, D] = calcular_picos(H, T, C, factor);
        tabla = [tabla, transpose([factor, R, D])];

    end

    printf("\n\tFactor\tR\tD\n");

    transpose(tabla)

end
```

4. Conclusiones

Podemos concluir que el método de Runge-Kutta de orden 4 resulta muy útil a la hora de aproximar una función, conociendo su derivada y un valor inicial. Permite aproximarse al valor real de la función con un error mucho menor que otros métodos como el de Heun (orden 2) o el de Euler (orden 1), y el hecho de automatizar el proceso evita la parte engorrosa de calcular todos los términos intermedios K1, K2, K3 y K4 que se necesitan para pasar de un y_k a un y_{k+1}.

Además, consideramos de gran utilidad la utilización del Software Octave con el objetivo de resolver problemas físico-matemáticos. El lenguaje utilizado por Octave no es tipado, lo que resulta muy cómodo a la hora de realizar cálculos y programar funciones, a lo que si a esto se le suma la potencia del GNU Plot, resulta realmente práctico dado que es posible computar y graficar todos los resultados de los problemas planteados.

Por último, cabe destacar que la precisión con la que se obtuvo el resultado puede ser ajustado por el usuario, obviamente dentro de las limitaciones de la arquitectura del sistema en que se está trabajando.

Este trabajo nos resultó interesante como equipo dado que refleja el hecho de que los conocimientos adquiridos en la materia pueden ser aplicados perfectamente a problemas de la vida real, ayudando a minimizar costos, agilizar procesos, etc.