Análisis de movimiento en un sistema hidráulico de salmuera a través de EDO

Métodos analíticos y numéricos modelados en MATLAB.

Clemente Serrano

28 de Agosto del 2015

Resumen

En el presente trabajo se estudió cómo resolver ecuaciones diferenciales utilizando el método de Euler de un problema propuesto que corresponde a determinar la tasa de acumulación de una solución de salmuera con el fin de poder comparar este resultado con la resolucin de estas ecuaciones mediante el comando `dsolve()` de MATLAB y calcular los errores que el método de Euler supone. El análisis será visto desde un punto de vista analítico y numérico y con ayuda del software MATLAB.

1 INTRODUCCIÓN

Una salmuera (solución de sal en agua) entra en un tanque a una velocidad $\vec{v}_1$ $(galones\, de\, salmuera/min)$ y con una concentración de c_1 $(lib.\, sal/gal\ \ salmuera)$. Inicialmente el tanque tiene Q galones de salmuera con P libras de sal disueltas. La mezcla bien homogenizada abandona el tanque a una velocidad de $\vec{v}_2$ $(gal\, salmuera/min)$. La relación que se trabaja para determinar las libras de sal que hay en el tanque en cualquier instante t es (entiéndase por T_{ac} como la tasa de acumulación, T_e como la tasa de entrada y T_s como la tasa de salida de soluto):

$$T_{ac} = T_e - T_s \tag{1}$$

Esto quiere decir que si $x(t)$ representa las libras de sal (soluto) presente en el instante t, entonces la ecuación diferencial que rige este comportamiento está dada por:

$$\frac{dx}{dt} = \vec{v_1}c_1 - \frac{\vec{v_2}x}{Q + t(\vec{v_1} - \vec{v_2})} \quad \left(\frac{lib\, sal}{min}\right) \tag{2}$$

donde

$$c_2 = \frac{x}{Q + t(\vec{v_1} - \vec{v_2})} \quad (lib\, sal)$$

En cuanto a esto, se debe modelar el comportamiento de la salmuera en el tanque posicionándose en los siguientes casos:

1. El tanque posee una cantidad inicial de salmuera $Q = 300(lib)$, la concentración uno que es c_1 corresponde a $0.5\,(lib\, sal/gal\, salmuera)$ y velocidades $\vec{v_1} = \vec{v_2} = 2\ (gal\, salmuera/min)$. El proceso inicia normalmente y continúa durante 10 minutos, luego de esto el proceso se detiene y se introduce al tanque agua pura manteniendo las velocidades de entrada y salida.

2. Al igual que en el caso 1, se mantienen las condiciones $Q = 300(lib)$, $c_1 = 0.5\,(lib\, sal/gal\, salmuera)$ y $\vec{v_1} = \vec{v_2} = 2\ (gal\, salmuera/min)$, solo que ahora el proceso inicia normalmente hasta los siete minutos, instante donde el tanque sufre una ruptura y se genera una fuga de agua a una velocidad de 3 $(gal\, salmuera/min)$. Finalmente, luego de estos minutos (o sea a los diez minutos) se soluciona el imprevisto y el proceso vuelve a la normalidad.

En el presente informe se analizará el problema propuesto y se dará a conocer el método seguido para la resolución de este.

El objetivo del laboratorio es resolver las ecuaciones diferenciales propuestas en los dos casos presentados utilizando el método de Euler, respondiendo las preguntas que serán propuestas en el desarrollo de este informe, graficando las soluciones y concluyendo con el fin de poder comparar esta solución obtenida con la solución exacta realizada en el laboratorio anterior mediante el comando MATLAB *dsolve()* y finalmente poder realizar un estudio de errores del método de Euler.

2 REQUISITOS TEÓRICOS PREVIOS

2.1 La salmuera

PAra una buena comprensión del análisis realizado durante el trabajo, es necesario manejar los conceptos básicos que respaldan los fundamentos matemáticos y químicos utilizados.

En cuanto al concepto de salmuera, este se refiere al agua que tiene sal disuelta, centrándose en el ámbito de la química, se puede decir que la salmuera es H_2O que dispone de una elevada concentracin de $NaCl$ que se encuentra disuelta. Esto corresponde en sí a una solución, la cual es una mezcla homogénea de dos o más sustancias. La sustancia disuelta se denomina soluto y la sustancia donde se disuelve se denomina solvente. Existen tres clasificaciones para las soluciones, atendiendo a la cantidad de soluto que se encuentra disuelto. La solubilidad de las sustancias varía generalmente con la temperatura, por lo que se asumirá que la temperatura es constante en este caso. Se dice que una solución es insaturada cuando la capacidad de disolver soluto aún no ha sido rebosada y que se encuentra aun con capacidad de recibir más soluto a una temperatura dada, por lo que es posible agregar más soluto y que este se disuelva. Una solución saturada ocurre cuando está en el punto exacto donde no puede recibir más soluto porque pasaría su capacidad de disolución. Una solución sobresaturada posee mayor cantidad de soluto disuelto que el que admite la solución a esa temperatura. Todo esto se resume en la concentración de una solución ya que esto está referido a la cantidad de soluto que hay en una masa o volumen determinado de solución o solvente.

2.2 El Método de Euler

Las leyes que gobiernan los fenómenos de la naturaleza se expresan habitualmente en forma de ecuaciones diferenciales. Pocas ecuaciones diferenciales tienen una solución analítica sencilla, la mayor parte de las veces es necesario realizar aproximaciones, estudiar el comportamiento del sistema bajo algunas condiciones.

El *Método de Euler* o *De las Tangentes* constituye primer y más sencillo ejemplo de método numérico para la resolución de ecuaciones diferenciales ordinarias con valor inicial. Es decir,

$$\frac{dy}{dx} = f(x,y) \quad , \; y(x_0) = y_0$$

donde existe una solución única para el problema.

Interpretando la ecuacin diferencial $y' = f(x, y)$ como un campo de direcciones en el plano xy y la condicin $y(x_0) = y_0$ como un punto (x_0, y_0) de dicho plano, se puede aproximar la función solución $y(x)$ por medio de la recta tangente a la misma que pasa por ese punto:

$$\frac{dy}{dx} \approx y_0 + f(x, y)(x - x_0)$$

donde se ha utilizado que la pendiente de dicha tangente es: $m = y'(x_0)$ y, en consecuencia: $m = f(x_0, y_0)$.

Se calcula así de manera aproximada el valor de la solución y en el punto de abscisa x_1 como:

$$y(x_1) \approx y_1 = y_0 + f(x_0, y_0)(x_1 - x_0)$$

y con este punto aproximado ya calculado se puede repetir el método para obtener el otro punto aproximado (x_2, y_2) de la siguiente forma:

$$y(x_2) \approx y_2 = y_1 + f(x_1, y_1)(x_2 - x_1)$$

y así sucesivamente.

Es habitual en este método calcular la solución aproximada en puntos de la forma: $x_n = x_(n-1) + h = x_0 + nh$, siendo el h el paso del método. De esta forma se obtienen las fórmulas que determinan la solución aproximada en la forma:

$$x_n = x_(n-1) + h$$

$$y_n = y_{(n-1)} + f(x_{(n-1)}, y_{(n-1)})h$$

Por último, si se observa este método del punto de vista geométrico se tiene en definitiva que en el método de Euler la aproximación será tanto peor cuanto mayor sea el número de pasos, esto quiere decir que cuanto más lejos se encuentre del punto inicial, el margen de error será mayor, esto es que el error será mayor cuanto más grande sea el ?paso? del método, h. La siguiente figura ilustra lo anterior:

El error puede ser estudiado y analizado.

En este trabajo se resolverán ecuaciones diferenciales utilizando el método de Euler mediante un algoritmo en MATLAB, un algoritmo es una secuencia

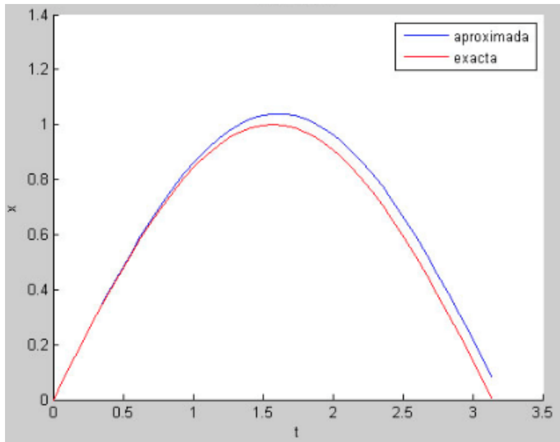


Figure 1: Representación del margen de error en aumento, *Método de Euler*

de códigos de ordenar que permiten implementar el método.

3 DESARROLLO

TAl como anteriormente se mencionó, se analizarán dos casos en específico de la dinámica de la salmuera dentro del tanque. Ya sea para el caso (1) como para el (2), se considerará que en el primer instante que el sistema inicia su actividad, las cantidades de sal contenidas en él son 36 ($lib\,sal$). Esto quiere decir que la condición inicial de la ecuación diferencial que modela el cambio en las cantidades de sal con respecto al tiempo se modifica a lo siguiente:

$$x(0) = 36\ (lib\,sal)$$

Paralelamente, también se considerará para ambos casos que los intervalos de tiempo usados para aplicar el método de Euler (diferencial del tiempo) serán constantes e iguales a 0.1 (min).

3.1 Caso 1

En la actividad de laboratorio anterior (laboratorio II), se determinó que considerando los parámetros iniciales establecidos, la ecuación diferencial de la cantidad de sal respecto al tiempo en la salmuera es:

$$\frac{dx}{dt} = 1 - \frac{x}{150}\left(\frac{lib_{sal}}{min}\right)$$

Ahora, ¿qué ocurre si sobre esta ecuación diferencial se aplica el método de Euler? Realizando los respectivos cálculos y reemplazo en las ecuaciones, se determinó que la solución general estaría dada por la siguiente expresión:

$$x(n+1) = x(n) + dt\left(1 - \frac{x(n)}{150}\right) \quad (lib\, sal)$$

Donde n representará la cantidad de intervalos de tiempo sobre los cuales se evaluará la función.

El caso dice que esta condición se dará hasta diez minutos, por lo tanto la función será evaluada vez veces cada 0.1 unidades ($n = 10$).

Luego, tras empezar a introducirse agua pura al tanque (sin sal), entonces el factor c_1 de la ecuación diferencial desaparece. La ecuación resultante fue calculada también en el laboratorio anterior, dando la siguiente expresión:

$$\frac{dx}{dt} = -\frac{x}{150}\left(\frac{lib_{sal}}{min}\right)$$

Aplicando el método de Euler para resolverla, se obtuvo la siguiente solución general:

$$x(n+1) = x(n) - dt\left(\frac{x(n)}{150}\right) \quad (lib\, sal)$$

Para contrastar los resultados de ambos métodos se evaluarán las soluciones generales tanto del analı́tico como del numérico transcurridos 20 minutos de actividad en el sistema.

Para el caso del método numérico, el resultado fue de 40.5576 ($lib\, sal$). Luego, para el caso del método analítico la concentración de sal fue de 40.5563 ($lib\, sal$). Puede notarse que ambos métodos difieren en una cantidad ´nfima de sal tras transcurrido el mismo tiempo: exactamente 0.0013 ($lib\, sal$).

Según la teoría, a medida que la función avanza, la diferencia entre las imágentes de ambas funciones también lo hacen. Por lo tanto, se evaluarán los mismos resultados pero en un dominio más extendido: 100 minutos transcurridos.

Tras evaluar las soluciones generales, se obtuvo que para el mt́odo numérico la concentración de sal en la salmuera fue de 23.7887 $(lib\, sal)$ y que para el método análitico 23.7922 $(lib\, sal)$. Ahora puede notarse una diferencia más considerable pero aún pequeña: 0.0035 $(lib\, sal)$. Esto quiere decir que el aumento en el margen de error es lento, pues claro, el diferencial sobre el cual se avanza en las sucesiones también lo es.

3.2 Caso 2

En este caso los lapsus de tiempo se dividieron en tres. Para el primero (de los 0 a 7 minutos) al igual que en el caso anterior, la ecuación diferencial de la cantidad de sal respecto al tiempo en la salmuera ya fue determinada. Considerando las condiciones dadas, dicha ecuación tiene la forma:

$$\frac{dx}{dt} = 1 - \frac{x}{150} \left(\frac{lib_{sal}}{min} \right)$$

Aplicando el método de Euler, la solución general de la ecuación diferencial es la siguiente (nótese que es la misma que en el primer lapsus de tiempo del caso (1) ya que las condiciones sobre las cuales se parten también son iguales):

$$x(n+1) = x(n) + dt \left(1 - \frac{x(n)}{150} \right) \quad (lib\, sal)$$

Luego de 7 minutos de actividad, se genera una ruptura en el tanque, lo cual provoca una fuga de salmuera que termina por aumentar la velocidad $\vec{v}_2$ con la que la mezcla abandona la estructura. Si cambian las condiciones físicas del sistema en análisis, también lo harán las matem´ticas que lo modelan. Por lo tanto, la nueva razón de cambio de las cantidades de sal en la salmuera con respecto al tiempo es:

$$\frac{dx}{dt} = 1 - \frac{5x}{300 - 3t} \left(\frac{lib_{sal}}{min} \right)$$

Tras aplicar el método de Euler, se obtuvo que la solución general de esta ecuación es:

$$x(n+1) = x(n) + dt\left(1 - \frac{5x(n)}{300 - 3t(n)}\right) \ \ (lib\, sal)$$

Por último, tras arreglarse el inconveniente el sistema vuelve a la normalidad por lo tanto la ecuación diferencial junto a su respectiva solución también.

Si nuevamente se contrastan los resultados de ambos métodos mientras pasa el tiempo, ¿ocurrirá también que el margen de error acumulado en el mt́odo de Euler tiene una razón de aumento práctivamente despreciable?

Transcurridos 20 minutos desde que el sistema comienza a funcionar, el método numérico arrojó que la concentración de sal en la salmuera sería de 48.9013 $(lib\, sal)$. Paralelamente el método analítico determinó que dicha cantidad de sal sería 48.8958 $(lib\, sal)$. Ha ocurrido algo inesperado: aumentó casi cinco veces el margen de error con respecto al caso (1), ahora la diferencia entre las cantidades de sal evaluadas en ambos métodos es de 0.0055 $(lib\, sal)$.

Ahora, ¿qué ocurre si se deja pasar más tiempo? ¿Cuánto más aumenta el margen de error?

Si tal como anteriormente se hizo, se dejan pasar 100 minutos de proceso, el mt́odo análitico afirma que la concentración de sal en la salmuera estaría siendo exactamente 90.6876 $(lib\, sal)$. Sin embargo el mt́odo numérico afirma que dicha cantidad de sal sería 90.7014 $(lib\, sal)$. La diferencia entre ambos mt́odos es de 0.0138 $(lib\, sal)$, lo que claramente refleja una diferencia con respecto al caso (1) (donde la diferencia entre las cantidades de sal presentes en la salmuera luego de 100 minutos fue de 0.0035 $(lib\, sal)$).

Lo anterior termina por demostrar que efectivamente la taza de aumento del margen de error en el caso (2) es mayor a la del caso (1).

Nótese que comparando las cantidades de sal del primer caso con el segundo, este último presenta valores notablemente mayores con respecto al otro independientemente del mt́odo empleado para resolver la ecuación diferencial. Esto se debe simplemente a que la función que describe el comportamiento de la sal en la salmuera en el segundo caso siempre es creciente, por lo tanto sea cual sea el tiempo transcurrido, las cantidades de sal irán aumentando en la mezcla. Esto no ocurre para el primer caso. El hecho de que se empiece a adicionar agua sin sal, genera automáticamente que sus índices de abundancia decrezcan en el tiempo, es decir, que la funcin solucin de la ecuacin diferencial tome pendientes negativas (sea decreciente).

¿Qué significa esto? Pues que la función del caso (1) decrecerá y la función del caso (2) crecerá, ambos hasta el infinito. Físicamente esto no es posible. Si el proceso no se detiene y se deja andar por un tiempo indefinido, llegará un momento en el que el agua alcance el denominado "límite de saturación", condición en la que el agua ya no aguantará más sal (para el caso (2)) o que definitivamente los índices de sal en el agua se hagan cero. En definitiva, las cantidades de sal no puede sobrepasar el límite ni adquirir valores negativos (no existe el concepto de $-\ (lib\, sal)$).

Entonces surge la siguiente pregunta: ¿cuándo la salmuera estará 100% purificada? Pues en el momento T tal que $x(T) = 0$. Con la ayuda del software MATLAB, pudo analizarse gráfica y numéricamente esta condición según el mt́odo analítico y el numérico.

A través del método analítico pudo determinarse que la cantidad de tiempo T que tardar'ia la salmuera en purificarse quizás no en un 100%, pero al menos más de un 90% (al orden de 10^{-2}), serían 1100 minutos, es decir, aproximadamente 18 horas.

En el caso del mt́odo numérico, este tiempo T también equivaldría aproximadamente a 1100 minutos, ya que genera casi el mismo orden de acercamiento a $x(T) = 0$: 10^{-2}.

Específicamente, en $t = 1100$ la función $x(t)$ obtenida por el método analítico afirmó que las cantidades de sal presentes en la salmuera serían 0.0303 $(lib\, sal)$. Por su lado, la función $x(t)$ obtenida por el método de Euler definió que las cantidades de sal en la mezcla luego de 1100 minutos serían 0.0302 $(lib\, sal)$.

3.3 Algoritmos MATLAB para los cálculos

Quizás resolver las ecuaciones diferenciales propuestas en el análisis de de la dinámica de la salmuera en el tanque no es una tarea tediosa. Pero, ¿qué ocurre al momento de tratar de determinar qué pasa con las funciones al minuto 100, o averiguar cuándo la función se hace prácticamente cero (alrededor de 1100 minutos)?

Tan solo considérece que en el uso del método de Euler los saltos en el tiempo para hacer las aproximaciones (diferenciales dt) son de 0.1 en 0.1. Eso indica que para haber determinado que la función al cero recién en los 1100 minutos hubo que previamente hacer 11000 iteraciones.

Debido a lo anterior, a través del software MATLAB se han creado (acorde al caso en análisis) una serie de algoritmos capaces de desempeñar dichas tareas. A continuación se explicará cómo funcionan y cuáles son los comandos claves utilizados.

3.3.1 Para el caso 1

La situación que aquí se propone dice que se inicia un proceso con ciertos parḿetros iniciales como velocidades de entrada y salida, concentraciones de sal, etc. Este proceso se mantiene invariable hasta diez minutos desde su inicio. Luego de ese tiempo, cambia una de las condiciones iniciales (la concentracin de sal presente en la salmuera) por lo que la ecuación diferencial que modela las concentraciones de sal también lo hace. La finalidad del análisis consistió en ver cćomo se comportaba la cantidad de sal en la salmuera conforme pasaba el tiempo desde dos puntos de vista: a partir de un método analt́ico y uno numérico. Para estudiar este caso, lo que se hizo fue evaluar las funciones en diferentes tiempos para ver cómo diferían unas de otras.

Este algoritmo se encarga de evaluar dichos tiempos y de mostrar gráficamente el comportamiento de ambas funciones (visualizar el comportamiento de la salmuera contribuye a un mejor estudio de resultados).

El procedimiento para obtener la solución de la ecuación diferencial por el método analítico es el mismo utilizado en el laboratorio anterior: se definen las ecuaciones diferenciales de ambas fases del caso, luego a través del comando *dsolve()* se obtienen sus soluciones y se transforman a funciones en línea (capaces de evaluarse fácilmente, es decir, sin tener que hacer extensos procesamientos de iteración). Finalmente se definen los intervalos de tiempo (dominio de la función para ambas fases, recordando que las funciones son diferentes) para luego evaluarlos en la función y graficar.

Lo diferente está ahora en el mt́odo numérico. Se parte por definir el diferencial de tiempo $dt = 0.1$ y dos matrices nulas (compuestas nada más que de ceros). Cada una de ellas representa una especie de base de datos sobre las cuales se irán guardando los valores de las cantidades de sal propias de cada uno de los tiempos. Para eso se inicia un ciclo iterativo for (propio de cada fase del caso) el cual a través de un contador recorre el dominio de la función (de 0.1 en 0.1 partiendo desde cero) guardando cada uno de los valores en una de las matrices nulas antes definidas. A partir de este contador el ciclo toma la función cantidad de sal v/s tiempo la evalúa en cada uno de los valores del dominio para definir el recorrido, rango de valores que se guarda en la otra matriz nula. El procedimiento se repite para ambos casos. Finalmente se grafica la relación dominio/ recorrido entre ambas matrices para generar la representación visual del método.

El algoritmo es el siguiente:

```
clear all
clear figure
clc
% Variables generales
%--------
c1=0.5;
Q=300;
v1=2;
v2=2;
%--------
% Funcion por metodo analitico:
%--------------------------
syms x(t)
t1=0:1:10; % Primer caso
sol1=dsolve(diff(x)== v1*c1 - v2*(x/(Q+t*(v1-v2))), x(0)==36);
f=inline(sol1);
t2=10:1:20; % Segundo caso
sol2=dsolve(diff(x)== - v2*(x/(Q+t*(v1-v2))), x(10)==f(10));
f2=inline(sol2);
%--------------------------
% Funcion por metodo numerico:
%-----------------------------------
dx=0.1;
%Primer caso:
x1=zeros(10,1);
y1=zeros(10,1);
x1(1)=0;
y1(1)=36;
for i=1:100
   x1(i+1)=x1(i)+dx;
   y1(i+1)=y1(i)+dx*(v1*c1 - v2*(y1(i)/(Q+x1(i)*(v1-v2))));
   y1f=y1(i+1);
end
%Segundo caso:
```

```
25. x2=zeros(10,1);
26. y2=zeros(10,1);
27. x2(1)=10;
28. y2(1)=y1f;
29. for i=1:100
30.    x2(i+1)=x2(i)+dx;
31.    y2(i+1)=y2(i)+dx*(- v2*(y2(i)/(Q+x2(i)*(v1-v2))));
32.    y2f=y2(i+1);
33. end
%-----------------------------------
%Graficacion:
%--------------------------
%Metodo analitico:
34. grafico_analitico1=plot(t1,f(t1))
35. grafico_analitico1.LineWidth = 1.5
36. hold on
37. grafico_analitico2=plot(t2,f2(t2));
38. grafico_analitico2.LineWidth = 1.5
39. hold on
%Metodo numerico:
40. grafico_numerico1=plot(x1,y1);
41. grafico_numerico1.LineWidth = 2
42. hold on
43. grafico_numerico2=plot(x2,y2);
44. grafico_numerico2.LineWidth = 2
%--------------------------
% Formalidades:
%---------------------------------------------------------
45. titulo=title({' ';
'\textbf{Cambio en las cantidades de sal respecto al tiempo I}
: '; ' '; '\, \, \,  \, \, \, \,  \, \, \, \,  \,
\textit{Metodo Analitico v/s Numerico} '});
46. set(titulo,'Interpreter','latex', 'fontsize', 17);
47. eje_x=xlabel('\textbf{Tiempo (min)}') ;
48. set(eje_x,'Interpreter','latex', 'fontsize', 15);
49. eje_y=ylabel('\textbf{Cantidad de sal (\textit{lib})}') ;
50. set(eje_y,'Interpreter','latex', 'fontsize', 15);
51. axis([0 20 0 50])
%---------------------------------------------------------
```

Luego de establecer los comandos $clearall$,$clear figure$ y clc (útiles para eliminar historiales de datos y graficaciones anteriores, permitiendo un mejor flujo de datos para los cálculos) desde la línea 4 hasta la 7 se definen los parḿetros iniciales del caso: $c1$, Q, $v1$ y $v2$ (constante de concentración de sal, cantidad de salmuera inicial y velocidades de entrada y salida de la salmuera respectivamente). Puede observarse que los métodos de cálculo han sido separados. Primero se inicia con el mt́odo analítico definiendo simbólicamente la variable x que depende de t (en la línea 8). Tras definir el primer intervalo de tiempo del proceso $t1$ (desde cero a diez minutos), se guarda en la variable $sol1$ la solución de la ecuación diferencial correspondiente a la primera fase del proceso. Desde las líneas 12 a la 14 se repite el proceso pero considerando que la ecuación diferencial ha cambiado: ya no se considera la concentración $c1$. Finalmente, a través del comando $inline()$ las dos soluciones de las ecuaciones diferenciales $sol1$ y $sol2$ son transformadas a funciones en línea para posteriormente evaluarlas con mayor facilidad.

Desde la línea 15 a la 33 se programa el mt́odo de Euler. Para eso se parte definiendo el diferencial de tiempo $dx = 0.1$ y las matrices nulas $x1$ e $y1$ a través del comando $zeros(a, b)$ donde a es el número de filas y b el de columnas de la matriz. En este caso $x1$ estaría representando la matriz del dominio e $y1$ la del recorrido. En las líneas 18 y 19 se define que en el tiempo 0 la cantidad de sal en la salmuera son 36 *lib*. Finalmente en la línea 20 se inicializa el ciclo for, con un contador i que va desde 1 hasta 100 (se busca evaluar desde los cero hasta los diez minutos y como el avance en el dominio se da cada 0.1 unidades, entonces para llegar desde el cero al diez se necesitan 100 iteraciones). En la líneas 21 y 22 se inicia la asignación de valores para las matrices $x1$ e $y1$ respectivamente. El método empleado para realizar dicha asignación consiste en: *el valor a introducir en la siguiente componente $i + 1$ de la matriz depende de una serie de términos y el valor actual de la componente i de la matriz.* A partir de este concepto se realiza la iteración hasta llegar a $i = 100$, momento donde termina la iteración (línea 24). Este mismo procedimiento se repite para la segunda fase del caso (línea 25 a la 33), solo que utilizando la ecuación diferencial que no depende de la concentración $c1$.

A partir de las líneas 34 a la 44 se grafican las funciones generadas por ambos métodos a través del comando $plot(x, y)$ donde x es el conjunto de valores correspondientes al dominio de la función e y la función (para el m 'etodo analítico) o el conjunto de valores correspondientes al recorrido (para el método de Euler).

Finalmente, desde la línea 45 a la 51 se definen algunas formalidades propias de la ventana de gráficos (títulos y nombre de los ejes coordenados).

3.3.2 Para el caso 2

La forma en la que este algoritmo se programó es prácticamente la misma que para el caso 1, con la diferencia de que en este caso el proceso dinámico de la salmuera se divide en tres fases: de cero a siete minutos con los mismos parḿetros iniciales del caso 1, de los siete a diez minutos considerando que producto de una ruptura en el tanque se genera una fuga de salmuera por ende el factor de velocidad de salida $\vec{v_2}$ aumenta y de diez a veinte minutos el proceso vuelve a la normalidad. Las ecuaciones diferenciales que modelan el proceso siguen siendo dos, con la diferencia de que ahora se distribuyen en tres intervalos de tiempo diferentes.

Esto es lo único que cambia con respecto al caso 1, por lo que el algoritmo sigue dividiéndose según el mt́odo empleado: en el analítico se buscan soluciones a partir del comando $dsolve()$, se generan los intervalos de tiempo $t1$, $t2$ y ahora $t3$ y las soluciones obtenidas se linealizan con el comando $inline()$. En el método numérico también se generan matrices nulas (solo que ahora tres por cada fase del proceso) correspondientes al dominio y recorrido de las funciones como así también los ciclos for que les asignan los valores correspondientes. Las formas de graficación de definición de formalidades sigue siendo igual.

El algoritmo de este caso es el siguiente:

```
1. clear all
2. clear figure
3. clc
% Variables generales
%--------
4. c1= 0.5;
5. v1=2;
6. v2=2;
7. Q= 300;
%--------
% Funcion por metodo analitico:
%--------------------------------------------------
8. syms  x(t)
9. t1=0:1:7; % Primer caso
10. sol1=dsolve(diff(x)== v1*c1 - v2*(x/(Q+t*(v1-v2))), x(0)==36);
11. f=inline(sol1);
12. t2=7:1:10; % Segundo caso
13. sol2=dsolve(diff(x)== v1*c1- 5*(x/(Q+t*(v1-5))), x(7)==f(7));
14. f2=inline(sol2);
15. t3=10:1:20; % Tercer caso
16. sol3=dsolve(diff(x)== v1*c1 - v2*(x/(Q+t*(v1-v2))), x(10)==f2(10))
17. f3=inline(sol3);
%--------------------------------------------------
% Funcion por metodo numerico:
%--------------------------------
18. dx=0.1;
%Primer caso:
19. x1=zeros(10,1);
20. y1=zeros(10,1);
21. x1(1)=0;
22. y1(1)=36;
23. for i=1:70
24.    x1(i+1)=x1(i)+dx;
25.    y1(i+1)=y1(i)+dx*(v1*c1 - v2*(y1(i)/(Q+x1(i)*(v1-v2))));
```

```
26.    y1f=y1(i+1);
27. end
%Segundo caso:
28. x2=zeros(10,1);
29. y2=zeros(10,1);
30. x2(1)=7;
31. y2(1)=y1f;
32. for i=1:30
33.    x2(i+1)=x2(i)+dx;
34.    y2(i+1)=y2(i)+dx*(v1*c1- 5*(y2(i)/(Q+x2(i)*(v1-5))));
35.    y2f=y2(i+1);
36. end
%Tercer caso:
37. x3=zeros(10,1);
38. y3=zeros(10,1);
39. x3(1)=10;
40. y3(1)=y2f;
41. for i=1:100
42.    x3(i+1)=x3(i)+dx;
43.    y3(i+1)=y3(i)+dx*(v1*c1 - v2*(y3(i)/(Q+x3(i)*(v1-v2))));
44.    y3f=y3(i+1);
45. end
%------------------------------------
%Graficacion:
%---------------------------
%Metodo analitico:
46. grafico_analitico1=plot(t1,f(t1))
47. hold on
48. grafico_analitico2=plot(t2,f2(t2))
49. hold on
50. grafico_analitico3=plot(t3,f3(t3));
%Metodo numerico:
51. hold on
52. grafico_numerico1=plot(x1,y1)
53. hold on
54. grafico_numerico2=plot(x2,y2)
55. hold on
56. grafico_numerico3=plot(x3,y3)
%---------------------------
% Formalidades:
%---------------------------------------------------------
57. titulo=title({’ ’;
’\textbf{Cambio en las cantidades de sal en funcion del tiempo II}
: ’;’ ’; ’\, \, \,  \,\, \, \,  \, \, \, \,  \, \, \, \,  \,
\textit{Metodo Analitico v/s Numerico} ’})
58. set(titulo,’Interpreter’,’latex’, ’fontsize’, 17)
59. eje_x=xlabel(’\textbf{Tiempo (min)}’)
60. set(eje_x,’Interpreter’,’latex’, ’fontsize’, 15)
61. eje_y=ylabel(’\textbf{Cantidad de sal (\textit{lib})}’)
62. set(eje_y,’Interpreter’,’latex’, ’fontsize’, 15)
63. axis([0 20 0 50])
%---------------------------------------------------------
```

Nótese que nuevas variables han sido introducidas. Primero, la ecuación diferencial generada por el cambio físico (en este caso la fuga de salmuera) ya no es la misma que en el caso anterior, lo que significa que las partes de algoritmo correspondientes a la segunda fase del proceso cambian la ecuación que manejan. Luego, en las líneas 15, 16 y 17 (correspondientes a la programación del método analítico) se introduce un nuevo intervalo de tiempo $t3$, correspondiente al conjunto de valores representantes de la tercera fase del proceso y la variable $sol3$, solución de la ecuación diferencial original del proceso solo que ahora considerando valores iniciales diferentes: desde el minuto diez en adelante. Luego, desde la línea 37 a la 45 se define el método de Euler pero para la tercera fase del proceso, es decir, las mismas matrices nulas y procedimiento del ciclo for, solo que ahora considerando tiempos de dominio diferentes (desde los diez minutos hasta los veinte).

Finalmente, puede observarse que en las líneas 50 y 56 se incluyen los gráficos de las terceras fases del proceso.

3.3.3 Para los márgenes de error

Tal como anteriormente se mencionó, el margen de error representa cuán desviada (o alejada) de la solución real está la aproximación realizada con el mt́odo numérico. Dicho margen estará dado por el valor absoluto de la diferencia de los valores de ambas soluciones (analítica y numérica) ya sea en un punto en específico o en una sucesión de puntos. Considerando que tal como anteriormente se mencionó, el margen de error causado por el método numérico aumenta progresivamente acorde avanza la función, si se quiere evaluar cómo se comporta este margen de error conviene generar un rango extenso de puntos.

En virtud de lo anterior, se necesita un algoritmo que tome registro de las imágenes de ambas soluciones (método analítico y numérico), obtenga su diferencia y luego la grafique para analizar más ilustrativamente cómo se comporta dicho margen de error.

Este algoritmo es práctivamente igual a los anteriores, con la diferencia de que toma una serie de matrices vacías las cuales las llena con el conjunto de imágenes generadas tras evaluar las soluciones de ambos métodos en los tiempos que le correspondan. Luego de producidas dos matrices, una con las concentraciones de sal correspondientes al método numérico y otra con las concentraciones del método analítico, se genera una tercera matriz resultante de la sustracción de las dos anteriores. Finalmente esa matriz se grafica con respecto a los tiempos correspondientes (para cada fase del proceso se genera un trío de matrices).

Los algoritmos para el cálculo del margen de error en ambos casos son los siguientes:

- **Caso 1**:

```
1. clear all
2. clear figure
3. clc
% Variables generales
%--------
4. c1=0.5;
5. Q=300;
6. v1=2;
7. v2=2;
%--------
% Funcion por metodo analitico:
%--------------------------
8. syms x(t)
9. t1=0.1:0.1:10; % Primer caso
10. sol1=dsolve(diff(x)== v1*c1 - v2*(x/(Q+t*(v1-v2))), x(0)==36);
11. f=inline(sol1);
12. M_analitica1=transpose([f(t1)]);
13. t2=10:0.1:19.9; % Segundo caso
14. sol2=dsolve(diff(x)== - v2*(x/(Q+t*(v1-v2))), x(10)==f(10));
15. f2=inline(sol2);
16. M_analitica2=transpose([f2(t2)]);
%--------------------------
% Funcion por metodo numerico:
%-----------------------------------
17. dx=0.1;
%Primer caso:
18. M_numerica1=[];
19. x1=zeros(10,1);
20. y1=zeros(10,1);
21. x1(1)=0;
22. y1(1)=36;
23. for i=1:100
24.    x1(i+1)=x1(i)+dx;
25.    y1(i+1)=y1(i)+dx*(v1*c1 - v2*(y1(i)/(Q+x1(i)*(v1-v2))));
26.    y1f=y1(i+1);
27.    M_numerica1(i)=y1(i+1); %Calculo de error
28. end
%Segundo caso:
29. M_numerica2=[];
30. x2=zeros(10,1);
31. y2=zeros(10,1);
32. x2(1)=10;
33. y2(1)=y1f;
34. for i=1:100
35.    x2(i+1)=x2(i)+dx;
36.    y2(i+1)=y2(i)+dx*(- v2*(y2(i)/(Q+x2(i)*(v1-v2))));
37.    y2f=y2(i+1);
38.    M_numerica2(i)=y2(i+1);
39. end
%-----------------------------------
```

```
%Calculo de error:
%--------------------------
40. M_Error1=[];
41. M_Error2=[];
42. for i=1:100
43.    M_Error1(i)=abs(M_numerica1(i)-M_analitica1(i));
44.    M_Error2(i)=abs(M_numerica2(i)-M_analitica2(i));
45. end
%--------------------------
%Graficacion:
%--------------------------
46. plot(t1,M_Error1)
47. hold on
48. plot(t2,M_Error2)
%--------------------------
% Formalidades:
%---------------------------------------------------------
49. titulo=title({' ';
'\, \, \,  \, \,  \textbf{Margenes de error I}: '; ' '; '
\, \textit{Metodo Analitico v/s Numerico} '});
50. set(titulo,'Interpreter','latex', 'fontsize', 17)
51. eje_x=xlabel('\textbf{Tiempo (min)}') ;
52. set(eje_x,'Interpreter','latex', 'fontsize', 15)
53. eje_y=ylabel('\textbf{Cantidad de sal (\textit{lib})}');
54. set(eje_y,'Interpreter','latex', 'fontsize', 15)
55. axis([0 20 0 0.08])
%---------------------------------------------------------
```

- **Caso 2**:

```
1. clear all
2. clear figure
3. clc
% Variables generales
%--------
4. c1= 0.5;
5. v1=2;
6. v2=2;
7. Q= 300;
%--------
% Funcion por metodo analitico:
%----------------------------------------------------
8. syms  x(t)
9. t1=0.1:0.1:7; % Primer caso
10. sol1=dsolve(diff(x)== v1*c1 - v2*(x/(Q+t*(v1-v2))), x(0)==36);
11. f=inline(sol1);
12. M_analitica1=transpose([f(t1)]);
13. t2=7.1:0.1:10; % Segundo caso
14. sol2=dsolve(diff(x)== v1*c1- 5*(x/(Q+t*(v1-5))), x(7)==f(7));
15. f2=inline(sol2);
16. M_analitica2=transpose([f2(t2)]);
17. t3=10.1:0.1:20; % Tercer caso
18. sol3=dsolve(diff(x)== v1*c1 - v2*(x/(Q+t*(v1-v2))), x(10)==f2(10));
19. f3=inline(sol3);
20. M_analitica3=transpose([f3(t3)]);
%----------------------------------------------------
% Funcion por metodo numerico:
%--------------------------------
```

```
dx=0.1;
%Primer caso:
x1=zeros(10,1);
y1=zeros(10,1);
x1(1)=0;
y1(1)=36;
for i=1:70
   x1(i+1)=x1(i)+dx;
   y1(i+1)=y1(i)+dx*(v1*c1 - v2*(y1(i)/(Q+x1(i)*(v1-v2))));
   y1f=y1(i+1);
   M_numerica1(i)=y1(i+1);
end
%Segundo caso:
x2=zeros(10,1);
y2=zeros(10,1);
x2(1)=7;
y2(1)=y1f;
for i=1:30
   x2(i+1)=x2(i)+dx;
   y2(i+1)=y2(i)+dx*(v1*c1- 5*(y2(i)/(Q+x2(i)*(v1-5))));
   y2f=y2(i+1);
   M_numerica2(i)=y2(i+1);
end
%Tercer caso:
x3=zeros(10,1);
y3=zeros(10,1);
x3(1)=10;
y3(1)=y2f;
for i=1:100
   x3(i+1)=x3(i)+dx;
   y3(i+1)=y3(i)+dx*(v1*c1 - v2*(y3(i)/(Q+x3(i)*(v1-v2))));
   y3f=y3(i+1);
   M_numerica3(i)=y3(i+1);
end
%-----------------------------------
%Calculo de error:
%--------------------------
M_Error1=[];
M_Error2=[];
M_Error3=[];
for i=1:70
   M_Error1(i)=abs(M_numerica1(i)-M_analitica1(i));
end
for i=1:30
   M_Error2(i)=abs(M_numerica2(i)-M_analitica2(i));
end
for i=1:100
   M_Error3(i)=abs(M_numerica3(i)-M_analitica3(i));
end
%--------------------------
%Graficacion:
%--------------------------
plot(t1,M_Error1)
hold on
plot(t2,M_Error2)
hold on
plot(t3,M_Error3)
%--------------------------
% Formalidades:
```

```
%-----------------------------------------------------------
69. titulo=title({' ';
'\, \, \,  \, \,  \textbf{Margenes de error II}: '; ' '; '
\, \textit{Metodo Analitico v/s Numerico} '});
70. set(titulo,'Interpreter','latex', 'fontsize', 17)
71. eje_x=xlabel('\textbf{Tiempo (min)}') ;
72. set(eje_x,'Interpreter','latex', 'fontsize', 15)
73. eje_y=ylabel('\textbf{Cantidad de sal (\textit{lib})}');
74. set(eje_y,'Interpreter','latex', 'fontsize', 15)
75. axis([0 20 0 0.08])
%-----------------------------------------------------------
```

Puede observarse que en ambos casos los códigos siguen los mismos patrones. Las diferencias están efectivamente en la definición de las matrices de apoyo. En la programación del método analítico, las matrices que almacenan los datos se obtienen luego de evaluar las funciones solución el los intervalos que le corresponden. Por ejemplo, para el caso 1, en las línea 12 se define la matriz $M\_analitica$ evaluando la función f (linealización de la solución $sol1$) en el intervalo de tiempo $t1$ (correspondiente a los diez minutos de proceso inicial del caso 1). En definitiva, todas las imágenes generadas tras evaluar la función pasan a formar las componentes de la matriz $M\_analitica$.

Paralelamente, se realiza lo mismo pero en el método numérico. Por ejemplo, en la línea 30 del algoritmo del caso 2 se define dentro del ciclo for la matriz $M_numerica1$, la cual según el valor del contador i, se irá llenando con los datos correspondientes de la matriz $y1(i+1)$.

Definidas las matrices de imágenes para ambos mt́odos se crea la matriz final, la cual contiene los márgenes de errores en cada uno de los puntos por los que pasó la función a medida que transcurría el tiempo. Esto puede verse en las líneas 52 y 56 del algoritmo del caso 2. Se crea la matriz vacía M_Error1 la cual se define dentro de un ciclo for como el valor absoluto ($abs()$) de la resta entre la matriz numérica con la analítica propias de la fase del caso.

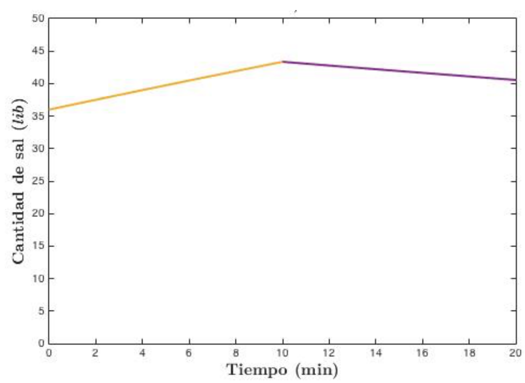
Finalmente, a través de la sentencia $plot()$ se grafican cada una de las matrices de error generadas según la fase del caso con respecto al rango de tiempo que les corresponde (véase como ejemplo las líneas 46, 47 y 48 del algoritmo del caso 1).

4 ANÁLISIS DE RESULTADOS

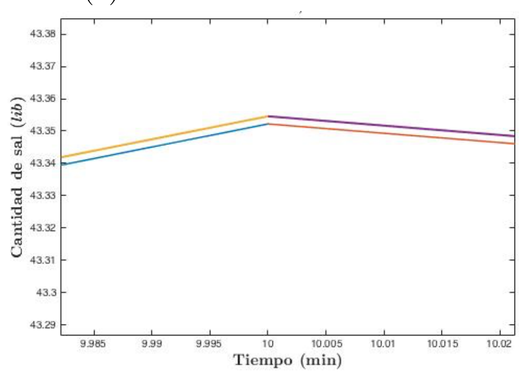
Con claridad respecto a los procedimientos matemáticos para determinar las soluciones de las ecuaciones diferenciales, los valores de las cantidades de sal presentes en la salmuera en tiempos específicos y los márgenes de error del método numérico respecto al analítico, es que se puede pasar a analizar los resultados gráficos de los análisis.

Respecto al caso 1, las variaciones de las cantidades de sal en la mezcla a medida que pasaba el tiempo se ven en la figura 2.(a). Como puede notarse, en un principio estas cantidades de sal aumentan en la solución (hasta los diez minutos, la curva toma pendientes positivas). Al empezar a adicionarse agua pura, la función empieza a descender, esto es claro, ya que tal como anteriormente se dijo, el factor concentración de sal c_1 desaparece, quedando una función de signo negativo.

La figura 2.(b) ilustra ccómo se ven las funciones solución de ambos métodos (la función de arriba es la obtenida por métodos numŕicos y la de abajo por métodos analíticos). Se reafirma la observación de que a escalas muy pequeñas (del orden 10^{-2}) recién se empiezan a ver diferencias entre los valores resultantes de ambos métodos. Los gráficos del caso 1 son:



(a) Gráfico a escala normal.

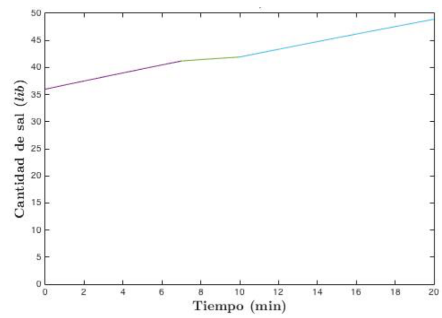


(b) Gráfico a escala pequeña.

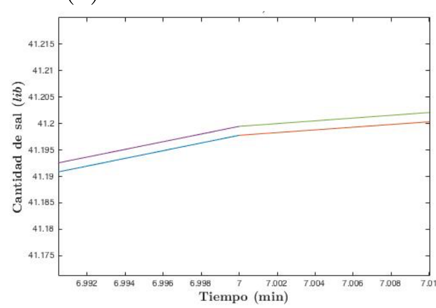
Figure 2: Cantidades de sal v/s tiempo I, métodos analítico y numérico.

Respecto al caso 2, el hecho de que luego de los siete minutos la función tome una pendiente más inclinada (se suman velocidades, por lo tanto los valores de los factores que multiplican a las variables crecen) puede apreciarse en la figura 3.(a). Nótese que luego de que se arregla la fuga (a los diez minutos), la curva solución vuelve a adquirir las pendientes iniciales.

Luego, la figura 3.(b) permite ver que en este caso la aplicación de ambos métodos también difiere en valores no considerables. Para este caso también la curva de arriba representa la solución del método numŕico y la de abajo la del método analítico. A continuación se muestran ambas figuras:



(a) Gráfico a escala normal.



(b) Gráfico a escala pequeña.

Figure 3: Cantidades de sal v/s tiempo II, métodos analítico y numérico.

Finalmente, respecto a los márgenes de error, los resultados gráficos fueron para los casos 1 y 2:

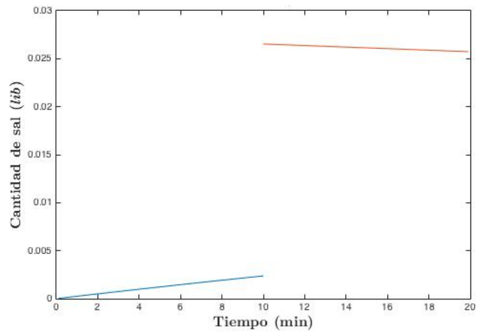


Figure 4: Margen de error, caso 1.

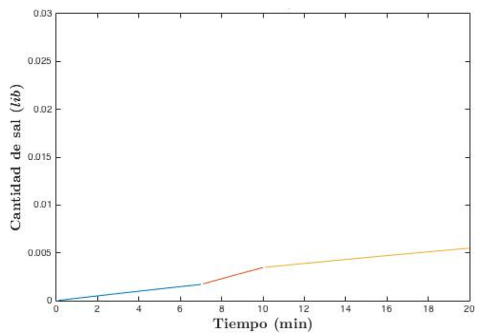


Figure 5: Margen de error, caso 2.

Tras observar estos gráficos puede notarse lo pequeño que es el margen de error en ambos casos. A medida que tiempo avanza en escalas de uno en uno, las diferencias entre las cantidades de sal en la salmuera resultantes de la la aplicación de ambos métodos son prácticamente nulas. Esto se debe a que el nivel de precisión del método de Euler fue muy alto: un diferencial de avance $dt = 0.1$ es muy pequeño y efectivo para encontrar una solución muy aproximada a las ecuaciones diferenciales propuestas.

5 CONCLUSIONES

Se concluye que el método de Euler puede realizar muy buenas aproximaciones a la solución real de una ecuación diferencial.

Se propusieron dos casos totalmente diferentes de flujo de salmuera en un tanque. Ambos fueron modelados a través de EDO, pero tratadas de diferente forma. Por un lado las ecuaciones fueron resueltas por el método clásico analítico y luego por el método numérico de Euler. Se recolectaron una serie de datos pertenecientes a los valores de cantidades de sal presentes en la mezcla mientras pasaba el tiempo y tras estudiarlos numérica y gráficamente los resultados fueron muy aproximados: las diferencias entre ambos métodos fueron del orden 10^{-2}, o sea, una cantidad muy pequeña.

Esto se le atribuye principalmente a que el diferencial dt para realizar las aproximaciones del método numérico fueron de valor bajo: tan solo 0.1 unidades.