Simulación y Optimización

Modelado
Parte II

Enrique E. Tarifa, Facultad de Ingeniería, UNJu

Mapa curricular de la materia

Simulación

Optimización

Mapa curricular de la materia

Simulación

Optimización

Mapa curricular de Simulación

- Definiciones
- Modelo de espacio de estados
- Resolución de modelos
- Simulación de plantas

Mapa curricular de Simulación

- Definiciones
- Modelo de espacio de estados
- Resolución de modelos
- Simulación de plantas

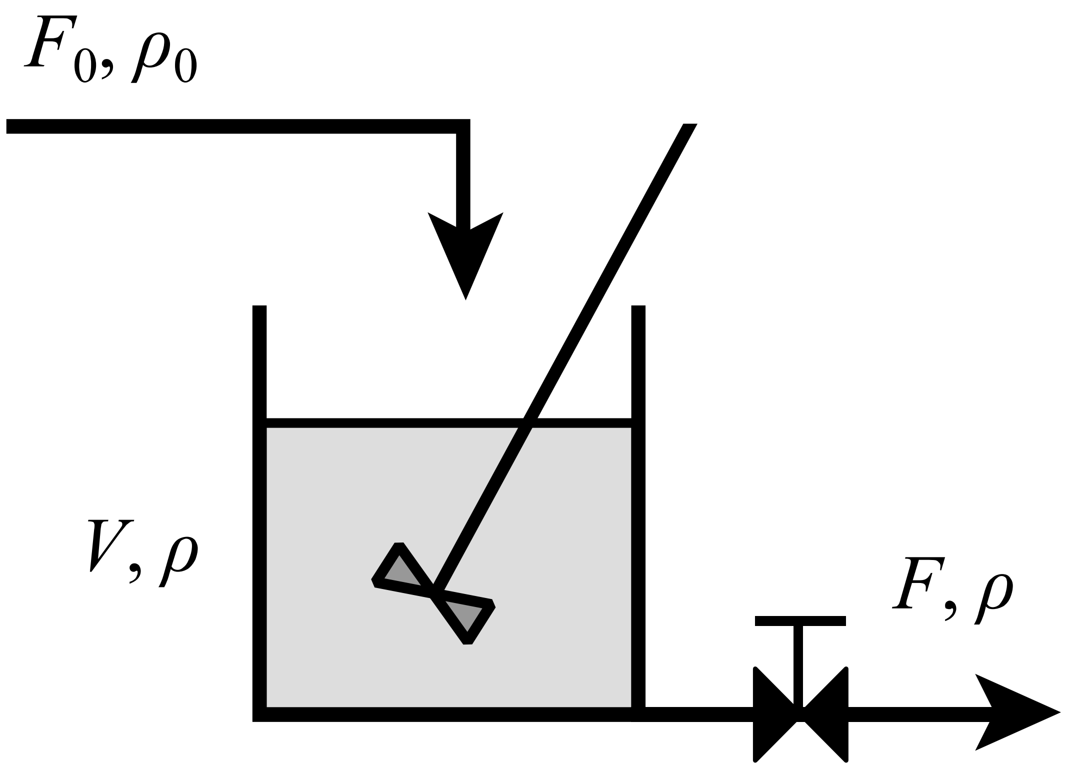
Mapa curricular de tanque con descarga gravitatoria

1. Modelo de espacio de estados
2. Balances
3. Ecuaciones constitutivas
4. Simplificación
5. Modelo estacionario
6. Simulación dinámica

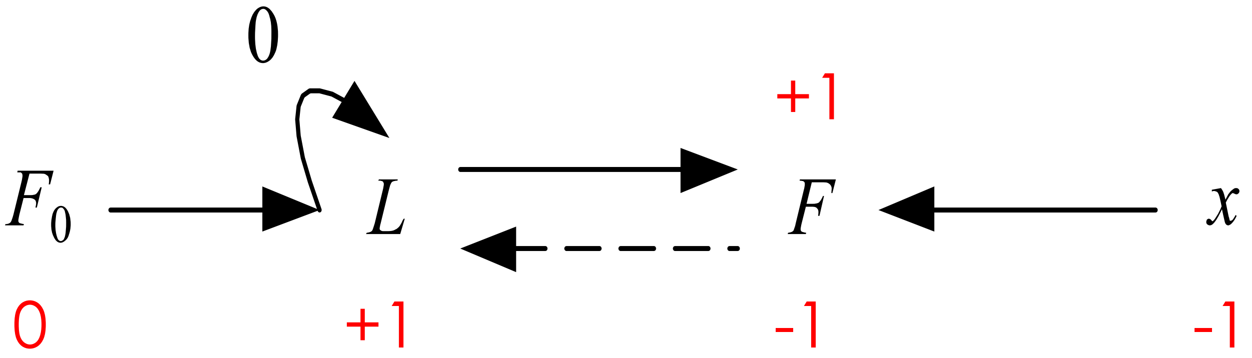
Tanque con descarga gravitatoria

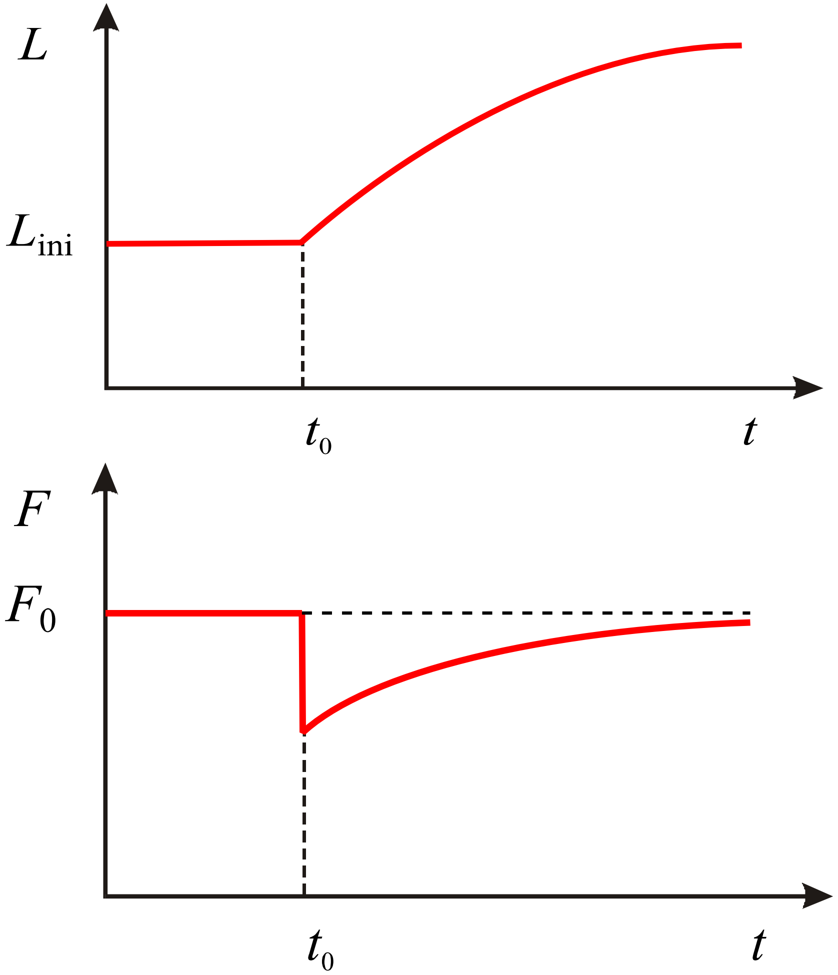
Tanque con descarga gravitatoria

- Descarga gravitatoria
- Estado estacionario
- F_0 = 2 l/s de agua, P = 1 atm y x_{ini} = 0.5
- Válvula de descarga lineal
- L_{ini} = 1 m, L_{max} = 2 m y D = 1 m
- ¿Rebalsará si x_{fin} = 0.25?
- De ser así, ¿cuándo?



Modelo conceptual





Modelo dinámico

Modelo original

$$\frac{d(V\rho)}{dt} = F_0\,\rho_0 - F\,\rho$$

$$\cancel{V} = A\,L$$

$$F = C_v\,x\sqrt{\frac{\Delta P_v}{\rho/\rho_w}}$$

$$\cancel{\Delta P_v} = \rho\,g\,L$$

Modelo simplificado 1

$$A\frac{d\left(L\,\rho\right)}{dt} = F_0\,\rho_0 - F\,\rho$$

$$F = C_v\,x\sqrt{\rho\,g\,L}$$

Pierde información pero no exactitud.

Modelo dinámico simplificado

Modelo simplificado 1

$$A\frac{d(L\rho)}{dt} = F_0\,\rho_0 - F\,\rho$$

$$F = C_v\,x\sqrt{\rho g L}$$

$$\rho_0 = \rho$$

$$\frac{d\rho}{dt} = 0$$

Modelo simplificado 2

$$A\frac{dL}{dt} = F_0 - F$$

$$F = C_v\,x\sqrt{\rho g L}$$

Pierde exactitud.

Modelo estacionario

Modelo dinámico

$$A\frac{dL}{dt} = F_0 - F$$

$$F = C_v\, x\sqrt{\rho g L}$$

Modelo estacionario

$$F = F_0$$

$$L = \frac{1}{\rho g}\left(\frac{F}{C_v\, x}\right)^2$$

Parámetros en SI

- F_0 = 2 l/s = 2×10^{-3} m^3/s
- L_{ini} = 1 m
- L_{max} = 2 m
- D = 1 m $\Rightarrow$ A = $\pi/4$ m^2 = 0.785 m^2
- g = 9.81 m/s^2
- ρ = 1 kg/l = 1000 kg/m^3

Determinación del C_V

Estado estacionario inicial

$$F_{ini} = F_0$$

$$L_{ini} = \frac{1}{\rho g}\left(\frac{F_{ini}}{C_v\, x_{ini}}\right)^2$$

Estimación de C_v

$$C_v = \frac{F_0}{x_{ini}\sqrt{\rho g L_{ini}}}$$

$C_v = 4.039\times10^{-5}\ m^{3.5}/kg^{0.5}$

Determinación del nivel final

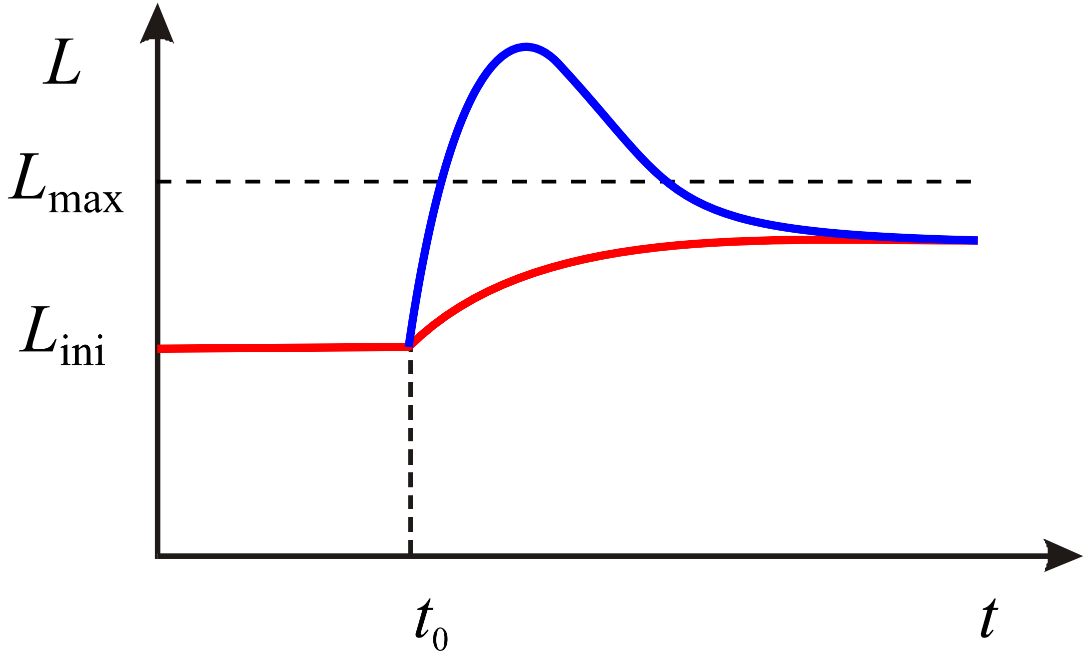
Modelo estacionario

$$F_{fin} = F_0$$

$$L_{fin} = \frac{1}{\rho g}\left(\frac{F_{fin}}{C_v\, x_{fin}}\right)^2$$

$$L_{fin} = 4\ \text{m} > L_{max}$$

Estado estacionario final



Determinación de la apertura mínima

Modelo estacionario

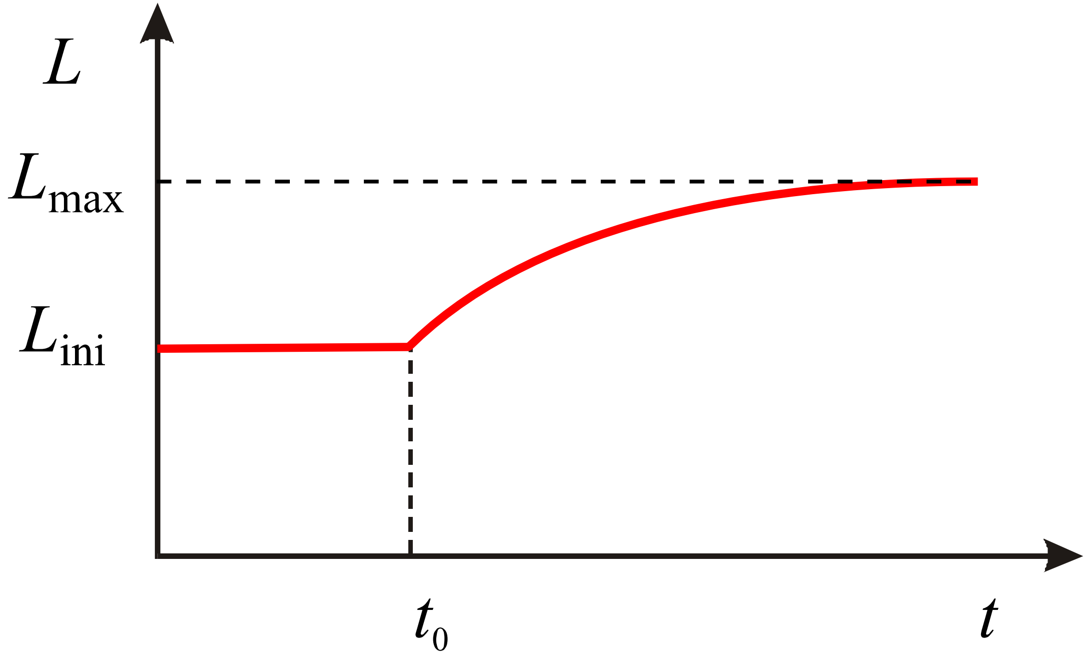
$$F_{fin} = F_0$$

$$L_{max} = \frac{1}{\rho g}\left(\frac{F_{fin}}{C_v\, x_{min}}\right)^2$$

$$x_{min} = \frac{F_0}{C_v \sqrt{\rho g L_{max}}}$$

$$x_{min} = 0.35$$

Estado estacionario final



Simulación dinámica

$$\frac{dL}{dt} = \frac{F_0 - F}{A}$$

$$F = C_v \, x \sqrt{\rho g L}$$

Derivada en forma normal

Incógnitas despejadas

Verificar estado estacionario inicial.

Ver Tanque con descarga gravitatoria.mmd

```
{Tanque con descarga gravitatoria}

METHOD RK4

STARTTIME = 0
STOPTIME = 1100
DT = 10

; Inicialización
INIT L = 1

; Sistema ODEs
L' = (F0-F)/A

; Sistema AEs
F = Cv*x*sqrt(rho*g*L)

; Datos
F0 = 2E-3
A = 0.785
Cv = 4.039E-5
rho = 1000
g = 9.81
x = 0.25
```

Construcción del modelo

1. Modificar `ODEs`:
    1. Escribir las derivadas despejadas.
    2. Escribir los vectores `dX`, `X` e `Y`.
2. Modificar `AEs`:
    1. Copiar el vector `X` de `ODEs`.
    2. Copiar el vector `Y` de `ODEs`.
    3. Escribir parámetros y ecuaciones algebraicas.

Construcción del modelo

3. Modificar `inicialización`:
    1. Copiar el vector `X` de `ODEs` en `LX` y agregar apóstrofos.
    2. Copiar el vector `Y` de `ODEs` en `LY` y agregar apóstrofos.
    3. Escribir parámetros de simulación `tfin` y `dt`.
    4. Inicializar las variables de estado `X`.
4. Modificar `analizar`.

tanque.m

```
% Tanque con descarga gravitatoria
% En X están las variables de estado.
% En Y deben ir las variables que se requieren en
las ODEs o que se quieren graficar.

clear all; close all; clc;

%================ Modelo ==================
```

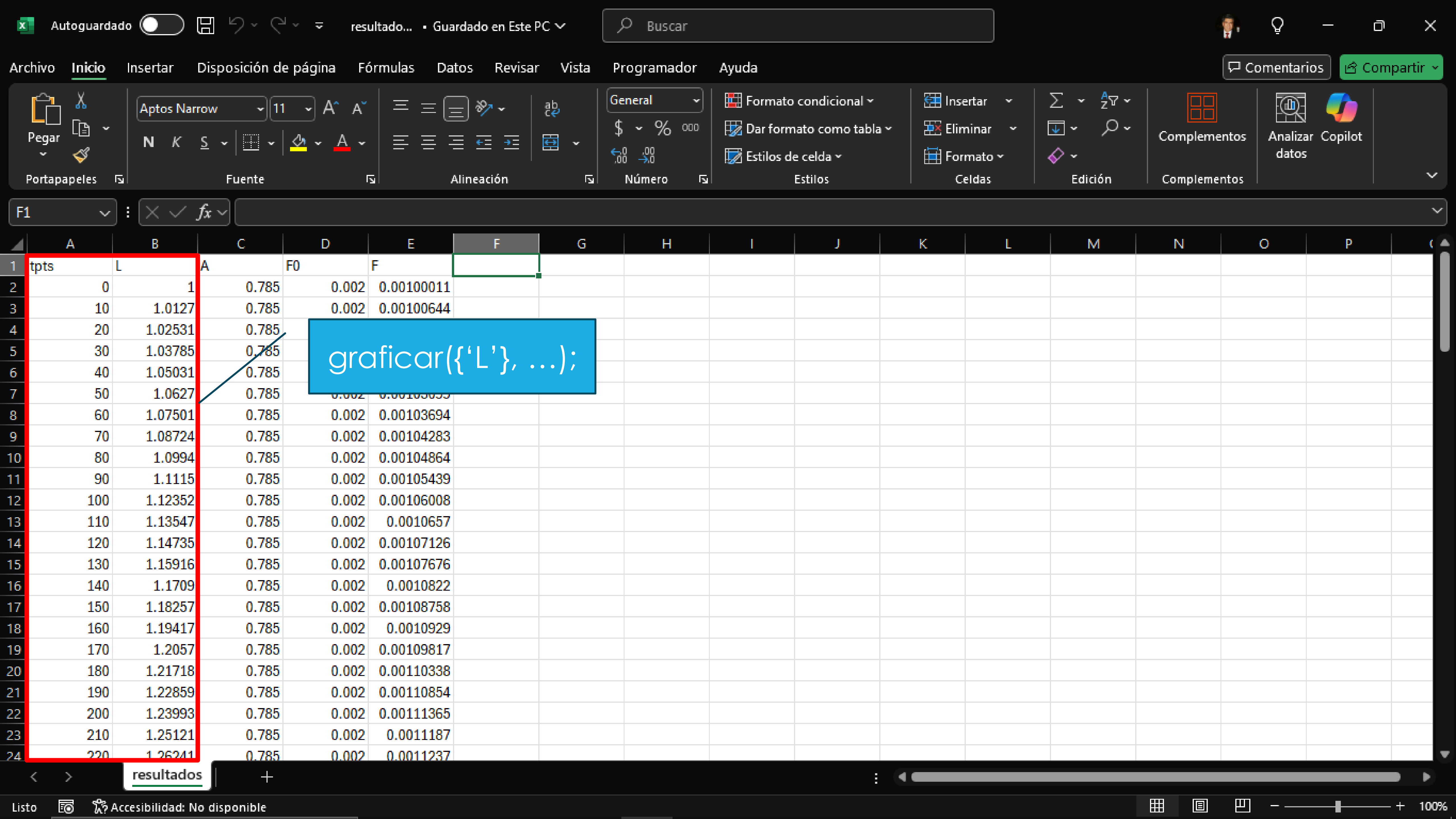
```
% ODEs
function dX = ODEs(t,X)
  % En dX devuelve el vector columna de derivadas

  % Recupera variables X
  [L] = num2cell(X'){1,:};

  % Recupera variables Y
  Y = AEs(t,X);
  [A F0 F] = num2cell(Y){1,:};

  % Ecuaciones diferenciales
  dL = (F0-F)/A;

  dX = [dL]'; % vector columna
endfunction % ODEs
```

```
% AEs
function Y = AEs(t,X)
  % En Y devuelve el vector fila de variables requeridas por ODEs o a graficar.

  % Recupera variables X
  [L] = num2cell(X'){1,:};

  % Parámetros
  F0 = 2E-3; A = 0.785; Cv = 4.039E-5; rho = 1000; g = 9.81; % Sistema SI

  % Ecuaciones algebraicas
  if t < 0
    x = 0.5;
  else
    x = 0.25;
  endif

  F = Cv*x*sqrt(rho*g*L);

  Y = [A F0 F];
endfunction % AEs
```

```
% Inicialización
function [tfin dt Xini LX LY] = inicializacion
  % Inicializa la simulación

  % Parámetros de simulación
  tfin = 1100; % tiempo final
  dt = 10; % paso temporal

  % Inicialización
  Lini = 1; % m
  Xini = [Lini]; % Inicializa el nivel con Lini

  % Leyendas
  LX = {'L'}; % Leyendas de las variables X
  LY = {'A' 'F0' 'F'}; % Leyendas de las variables Y
endfunction % inicializar
```

```
% Análisis
function analizar(LX,LY,tpts,X,Y)
  % Análisis de resultados. Funciones disponibles:
  % exportar('resultados.csv',[{leyendas}])
  % graficar({leyendas}, 'título', 'rótulo x', 'rótulo y', [limitesy])
  % vector(leyenda)
  exportar('resultados.csv');

  graficar({'L'}, 'Nivel vs. tiempo', 's', 'm');
  graficar({'F0' 'F'}, 'Caudales vs. tiempo', 's', 'm^3/s', [0 4E-3]);

  % Control de rebalse
  Lmax = 2; % m Altura del tanque
  Lt = vector('L'); % Recupera el vector de niveles.
  if Lt(end) <= Lmax % Verifica el último nivel porque es el mayor.
    disp('El tanque no rebalsó.');
  else
    tr = interp1(Lt, tpts, Lmax); % Se puede usar interp1 porque Lt es creciente.
    disp(['El tanque rebalsó en el tiempo igual a ' num2str(tr) ' s.']);
  endif
endfunction % analizar
```

```
% Análisis
function analizar(LX,LY,tpts,X,Y)
  % Análisis de resultados. Funciones disponibles:
  % exportar('resultados.csv',[{leyendas}])
  % graficar({leyendas}, 'título', 'rótulo x', 'rótulo y', [limitesy])
  % vector(leyenda)
  exportar('resultados.csv');

  graficar({'L'}, 'Nivel vs. tiempo', 's', 'm');
  graficar({'F0' 'F'}, 'Caudales vs. tiempo', 's', 'm^3/s', [0 4E-3]);

  % Control de rebalse
  Lmax = 2; % m Altura del tanque
  Lt = vector('L'); % Recupera el vector de niveles.
  if Lt(end) <= Lmax % Verifica el último nivel porque es el mayor.
    disp('El tanque no rebalsó.');
  else
    tr = interp1(Lt, tpts, Lmax); % Se puede usar interp1 porque Lt es creciente.
    disp(['El tanque rebalsó en el tiempo igual a ' num2str(tr) ' s.']);
  endif
endfunction % analizar
```



tpts	L	A	F0	F
0	1	0.785	0.002	0.00100011
10	1.0127	0.785	0.002	0.00100644
20	1.02531	0.785		
30	1.03785	0.785		
40	1.05031	0.785		
50	1.0627	0.785		
60	1.07501	0.785	0.002	0.00103694
70	1.08724	0.785	0.002	0.00104283
80	1.0994	0.785	0.002	0.00104864
90	1.1115	0.785	0.002	0.00105439
100	1.12352	0.785	0.002	0.00106008
110	1.13547	0.785	0.002	0.0010657
120	1.14735	0.785	0.002	0.00107126
130	1.15916	0.785	0.002	0.00107676
140	1.1709	0.785	0.002	0.0010822
150	1.18257	0.785	0.002	0.00108758
160	1.19417	0.785	0.002	0.0010929
170	1.2057	0.785	0.002	0.00109817
180	1.21718	0.785	0.002	0.00110338
190	1.22859	0.785	0.002	0.00110854
200	1.23993	0.785	0.002	0.00111365
210	1.25121	0.785	0.002	0.0011187
220	1.26241	0.785	0.002	0.0011237

tpts	L	A	F0	F
0	1	0.785	0.002	0.00100011
10	1.0127	0.785	0.002	0.00100644
20	1.02531	0.785	[illegible]	[illegible]
30	1.03785	0.785	[illegible]	[illegible]
40	1.05031	0.785	[illegible]	[illegible]
50	1.0627	0.785	[illegible]	[illegible]
60	1.07501	0.785	0.002	0.00103694
70	1.08724	0.785	0.002	0.00104283
80	1.0994	0.785	0.002	0.00104864
90	1.1115	0.785	0.002	0.00105439
100	1.12352	0.785	0.002	0.00106008
110	1.13547	0.785	0.002	0.0010657
120	1.14735	0.785	0.002	0.00107126
130	1.15916	0.785	0.002	0.00107676
140	1.1709	0.785	0.002	0.0010822
150	1.18257	0.785	0.002	0.00108758
160	1.19417	0.785	0.002	0.0010929
170	1.2057	0.785	0.002	0.00109817
180	1.21718	0.785	0.002	0.00110338
190	1.22859	0.785	0.002	0.00110854
200	1.23993	0.785	0.002	0.00111365
210	1.25121	0.785	0.002	0.0011187
220	1.26241	0.785	0.002	0.0011237

Lt = vector('L');

GNU Octave

- `end`: Valor de índice que refiere al último elemento de un vector o dimensión.
- `disp(x)`: Muestra el valor de `x`. Si `x` es un vector, se pueden concatenar varias cadenas en él.
  - `disp(['El tanque rebalsó en el tiempo igual a ' num2str(tr) ' s.']);`
- `num2str(x)`: Convierte el número `x` en una cadena.

GNU Octave

- `yi = interp1(x, y, xi)`: Determina por interpolación lineal el valor de `yi` correspondiente a `xi` considerando la tabla de valores de `y` en función de `x`. El vector `x` debe ser creciente. El vector `y` debe ser creciente o decreciente.

x	y
x_1	y_1
x_2	y_2
x_3	y_3
…	…
x_n	y_n

Recomendaciones

1. Distinguir entre mayúsculas y minúsculas.
2. Seguir el orden recomendado para la construcción del modelo.
3. Escribir las derivadas en forma normal.
4. Despejar una incógnita de cada ecuación algebraica.
5. Ordenar las ecuaciones para que puedan ser calculadas.

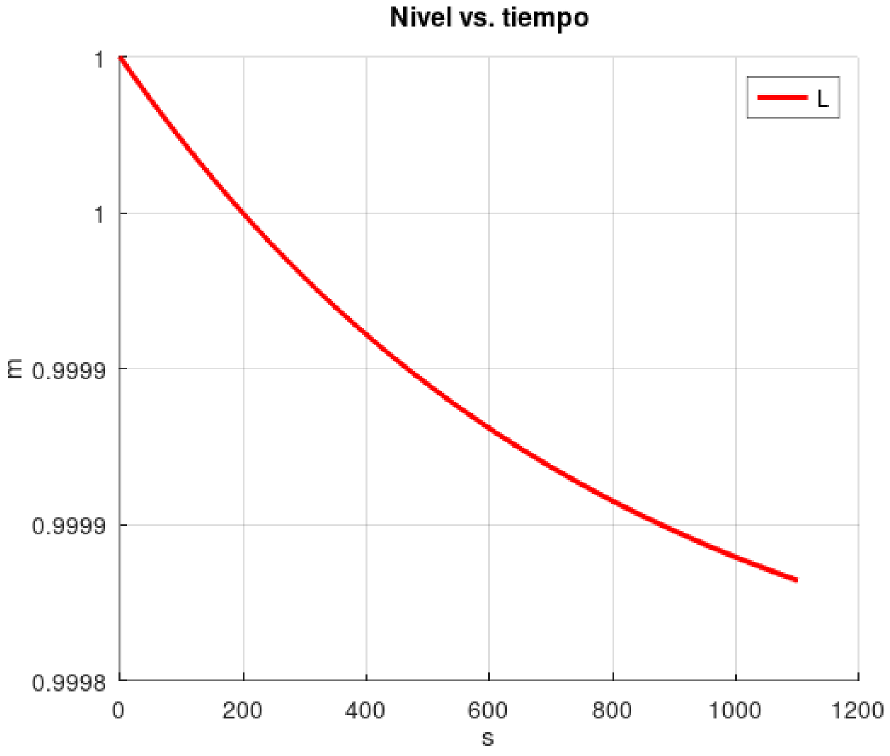
Recomendaciones

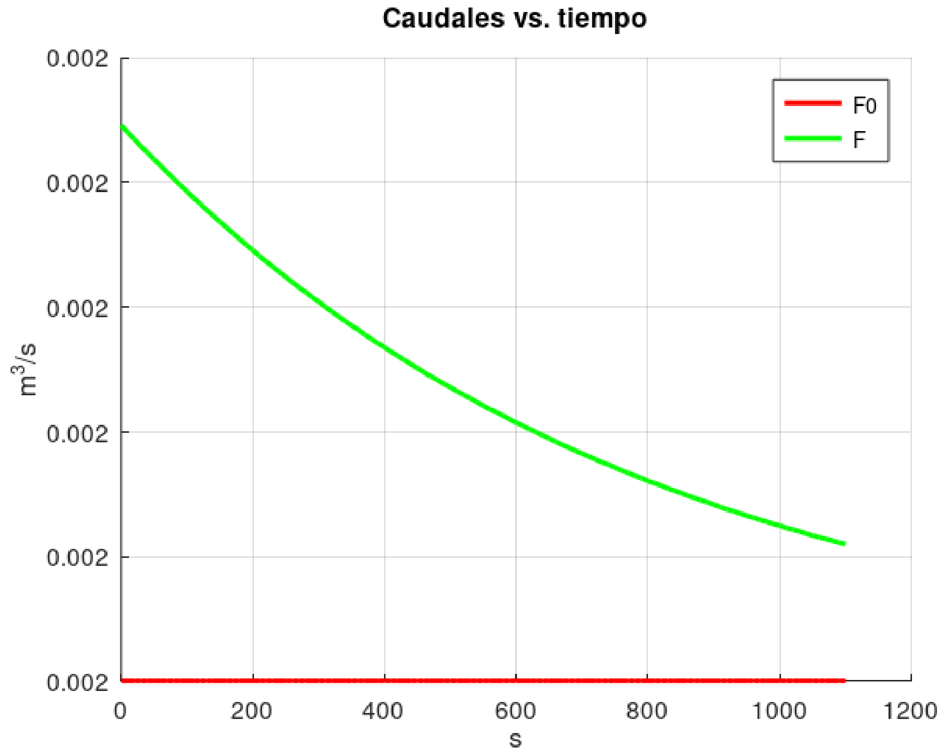
6. Elegir variables a graficar.
7. Elegir escalas de ejes.
8. Verificar estado estacionario inicial.
9. Leer atentamente los mensajes de la ventana de comandos.

Estado estacionario para x = 0.5

```
% AEs
function Y = AEs(t,X)
  % En Y devuelve el vector fila de variables requeridas por ODEs o a graficar.
...

  % Ecuaciones algebraicas
  if t < 10000
    x = 0.5;
  else
    x = 0.25;
  endif

...
  Y = [A F0 F];
endfunction % AEs
```

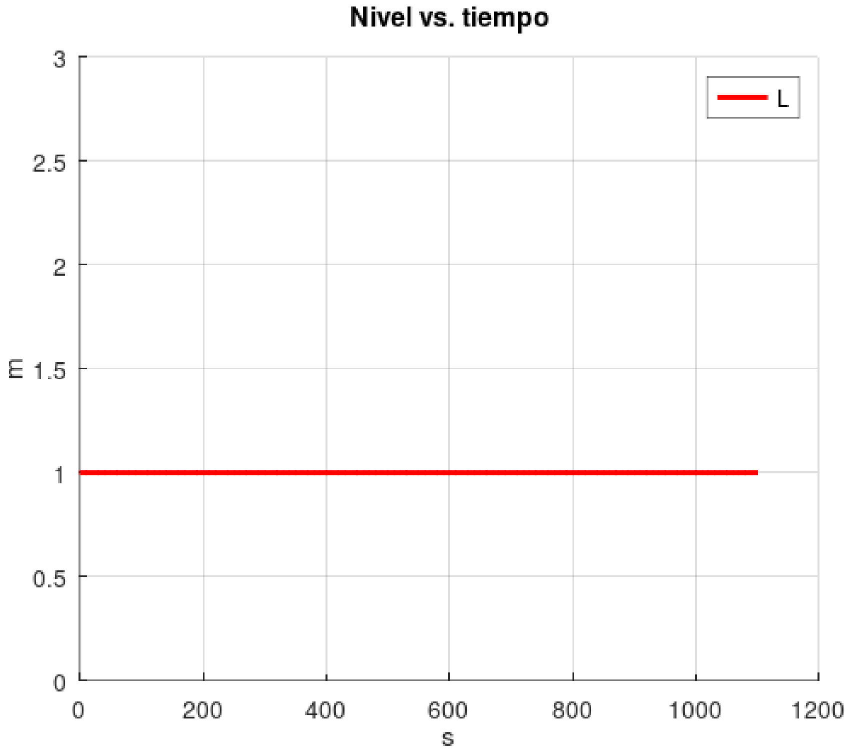
Estado estacionario para x = 0.5

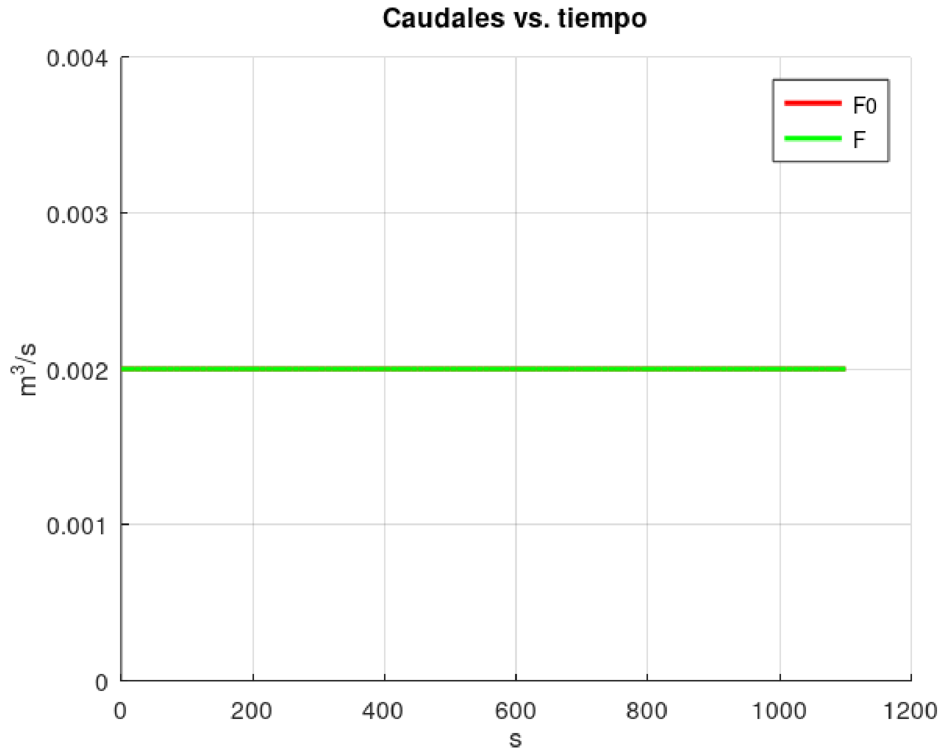





```
graficar({'L'},'Nivel vs. tiempo', 's', 'm');
graficar({'F0' 'F'},'Caudales vs. tiempo', 's', 'm^3/s');
```

Estado estacionario para x = 0.5



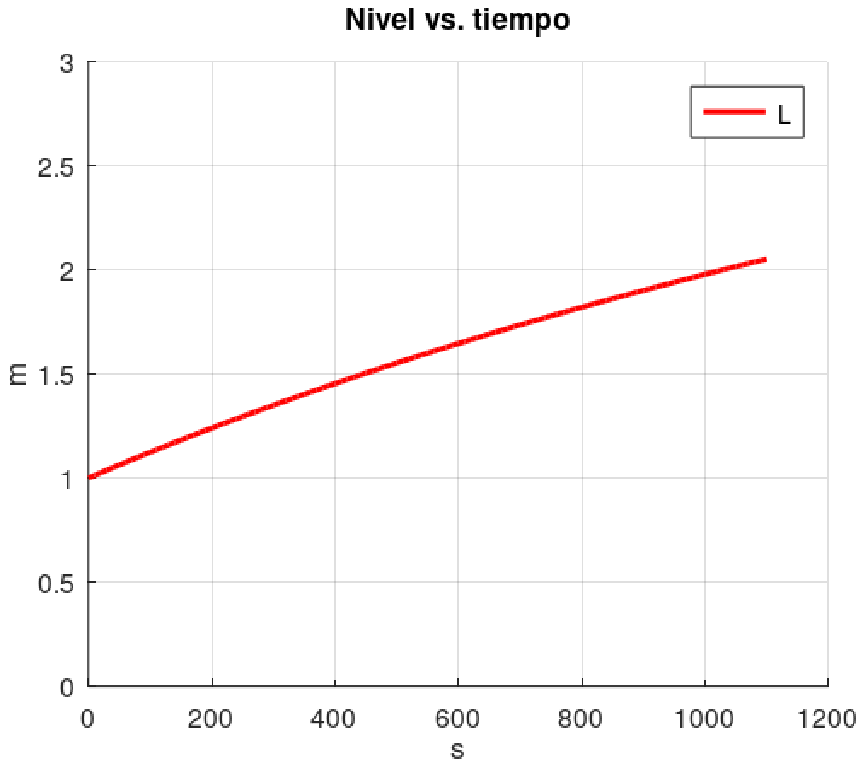


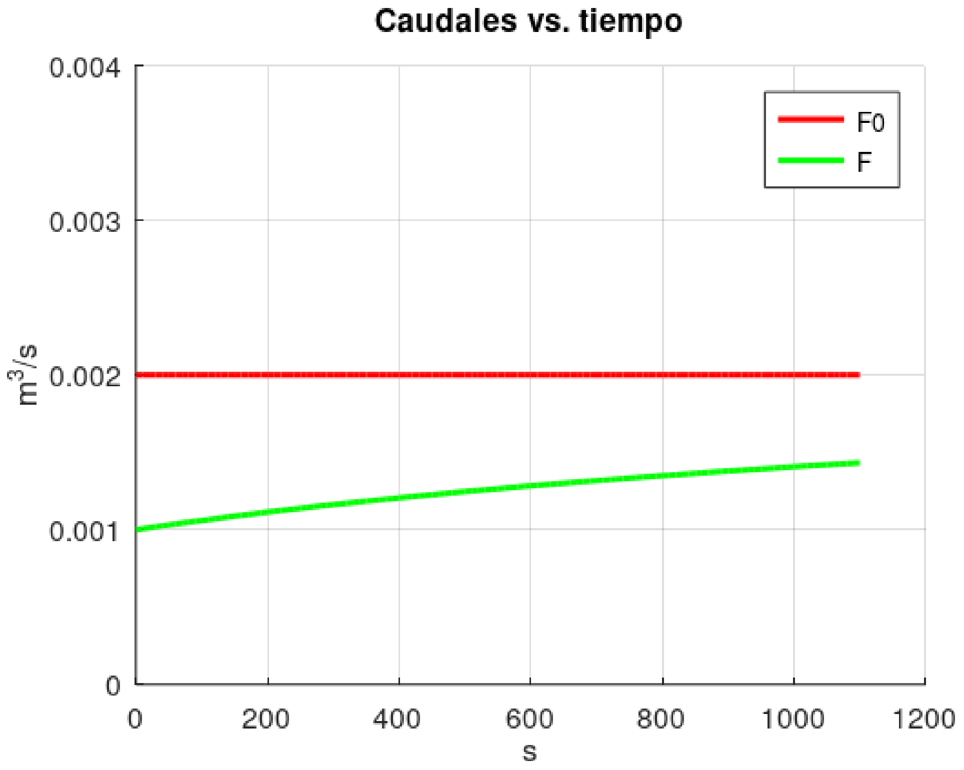

```
graficar({'L'}, 'Nivel vs. tiempo', 's', 'm', [0 3]);
graficar({'F0' 'F'}, 'Caudales vs. tiempo', 's', 'm^3/s', [0 4E-3]);
```

Simulación para x = 0.25

```
% AEs
function Y = AEs(t,X)
  % En Y devuelve el vector fila de variables a requeridas por ODEs o a graficar.
...

  % Ecuaciones algebraicas
  if t < 0
    x = 0.5;
  else
    x = 0.25;
  endif

...
  Y = [A F0 F];
endfunction % AEs
```

Resultados para x = 0.25



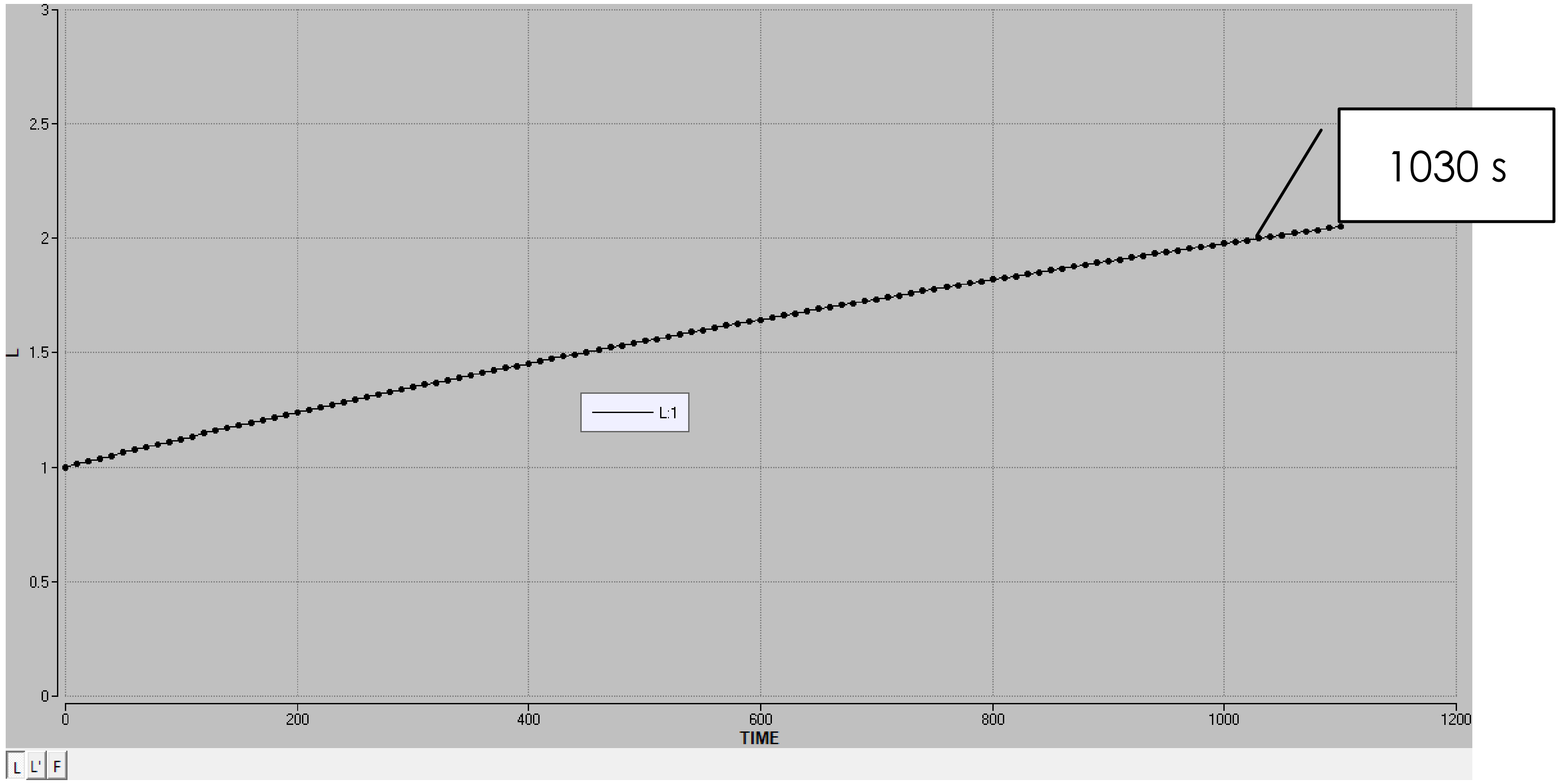



```
graficar({'L'}, 'Nivel vs. tiempo', 's', 'm', [0 3]);
graficar({'F0' 'F'}, 'Caudales vs. tiempo', 's', 'm^3/s', [0 4E-3]);
```

Resultados para x = 0.25

```
Resolvedor v01, 2025

Resolviendo el modelo...
Archivo exportado como "resultados.csv" en el directorio
de trabajo.
El tanque rebalsó en el tiempo igual a 1028.7738 s.

Simulación finalizada.
>>
```

Rebalse para x = 0.25



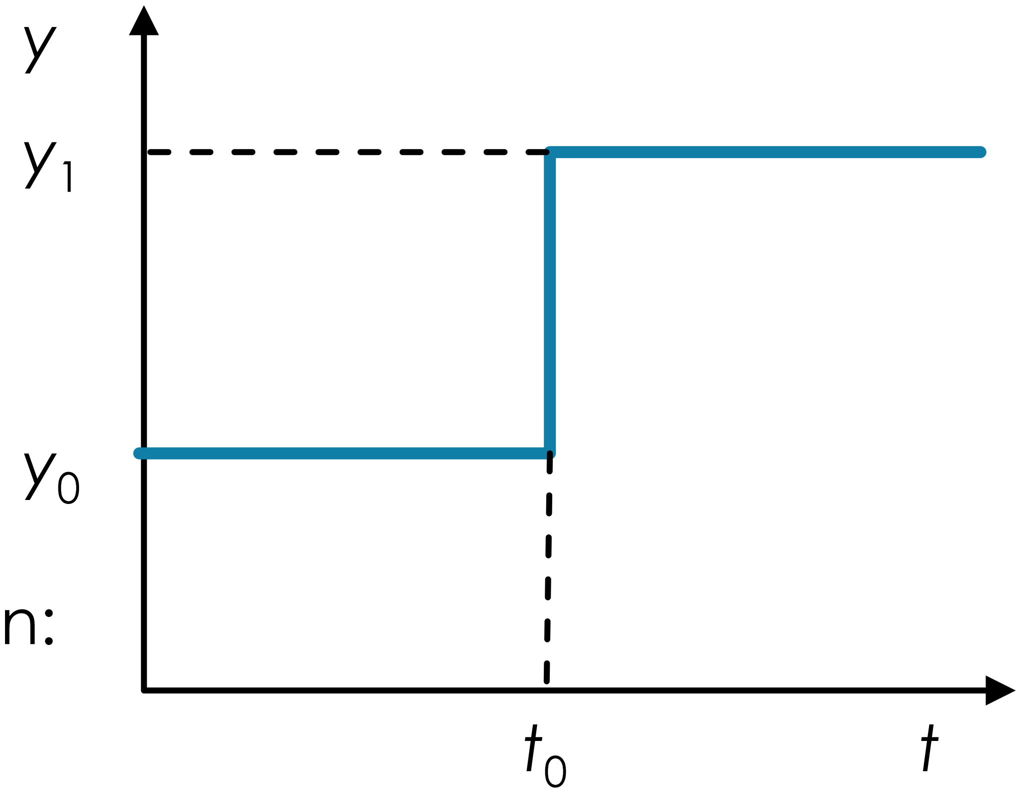
Perturbaciones comunes

- Escalón de y_0 a y_1 en t_0:

```
if t < t0
  y = y0;
else
  y = y1; % Se modifica con m%
endif;
```

- Escalón de $m\%$ de y_0 en t_0, modificación:

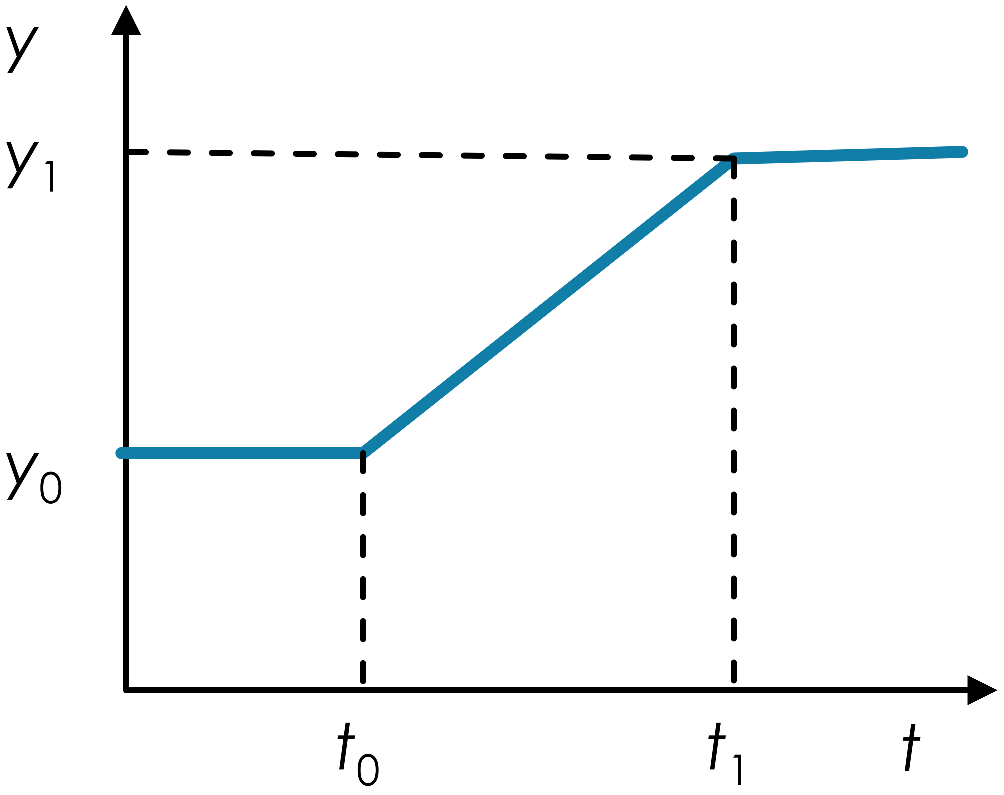
```
y1 = y0*(1+m%/100);
```



Perturbaciones comunes

- Rampa creciente de (t_0,y_0) a (t_1,y_1):

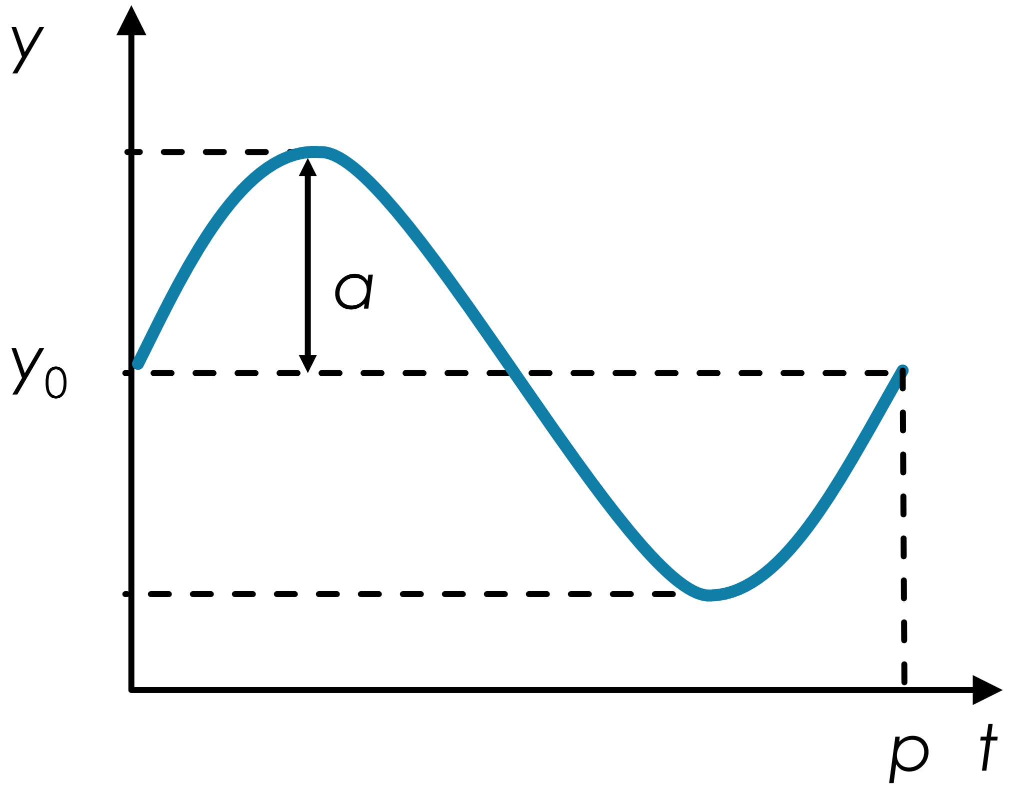
```
if t < t0
  y = y0;
elseif t < t1
  y = y0+(y1-y0)/(t1-t0)*(t-t0);
else
  y = y1;
endif;
```



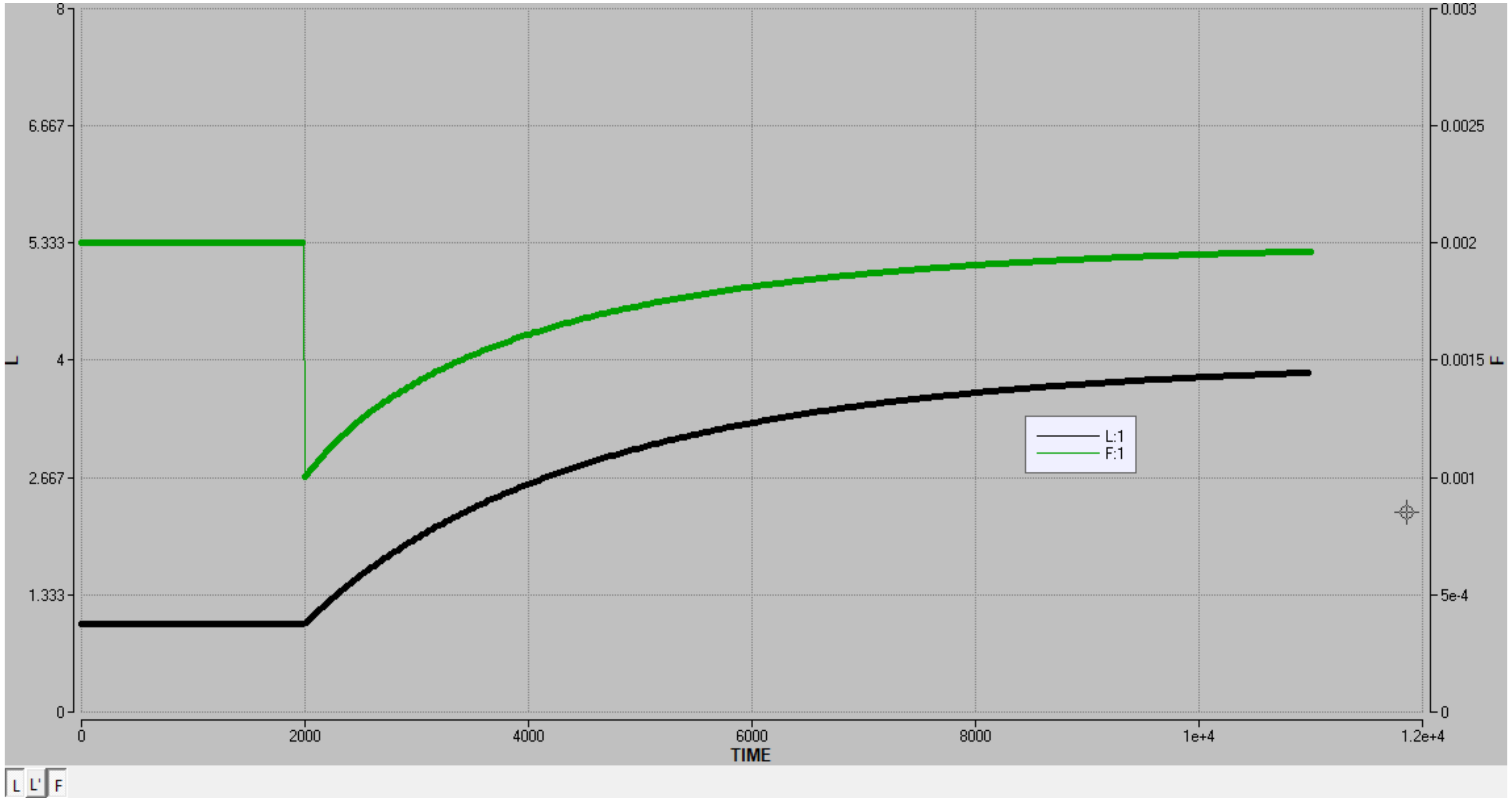
Perturbaciones comunes

- Senoidal con amplitud *a*, media y_0 y periodo *p*:

```
y = y0+a*sin(2*pi*t/p);
```



Efecto integral del tanque con x = 0.25



Berkeley Madonna - 02 Tanque con descarga gravitatoria.mmd

File Edit Flowchart Model Compute Graph Parameters Window Help

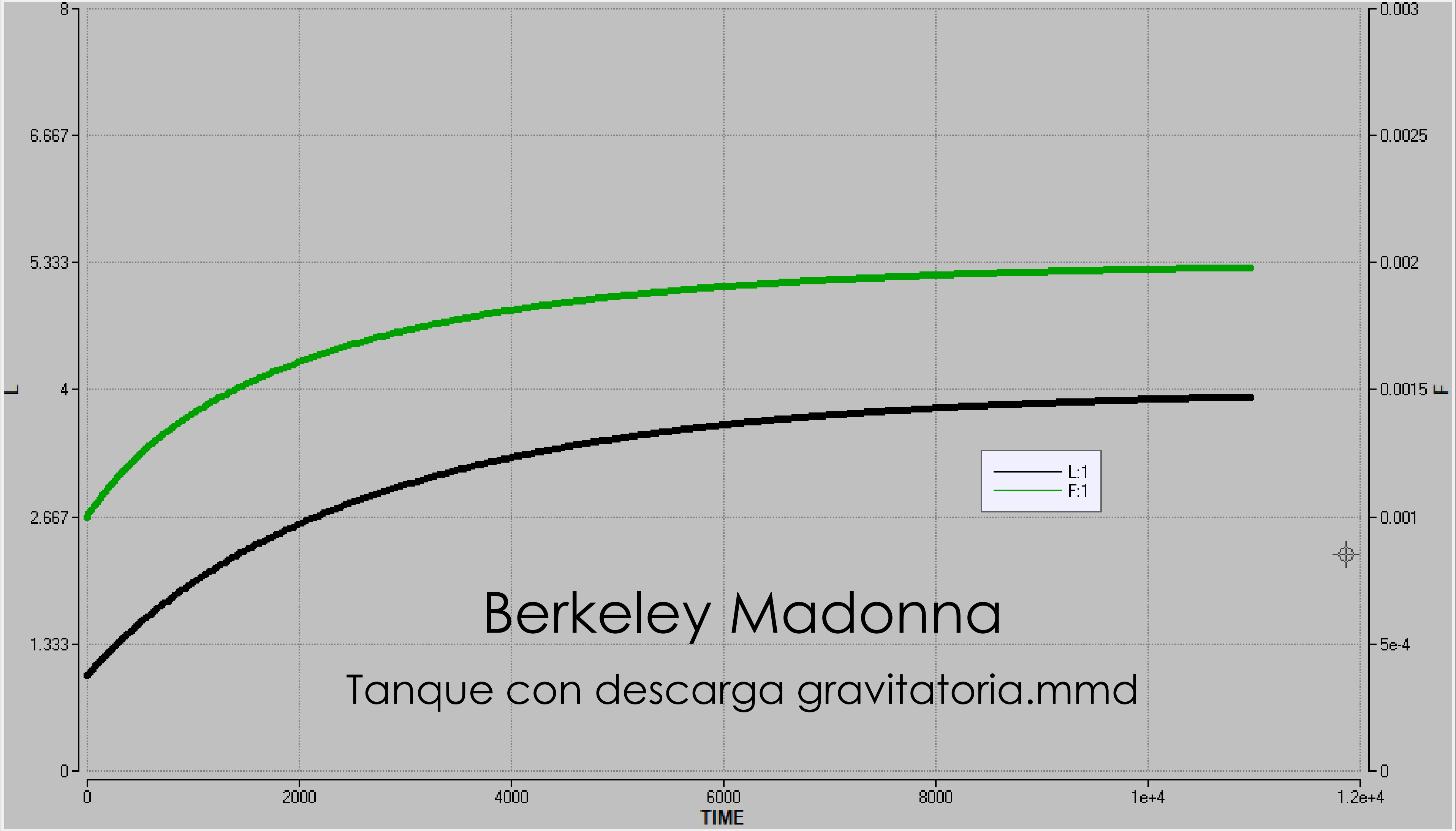
02 Tanque con descarga gravit...

```
{Tanque con descarga
gravitatoria}

METHOD RK4

STARTTIME = 0
STOPTIME = 11000
DT = 10

; Inicialización
INIT L = 1

; Sistema ODEs
L' = (F0-F)/A

; Sistema AEs
F = Cv*x*sqrt(rho*g*L)

; Datos
F0 = 2E-3
A = 0.785
Cv = 4.039E-5
rho = 1000
g = 9.81
x = 0.25
```

02 Tanque con descarga gravitatoria.mmd - Run 1: L, F vs. TIME

Page 1

TIME = 1.186e+4 L = 2.268 F = 8.504e-4



Ready

JSim version 2.21: Project ...Users\eetar\OneDrive\Documentos\Enrique\Materias\Simulación y Optimización\Ejemplos\Simulación\JSim\Tanque con descarga gravitatoria.proj

Project | Tanque | Message | L_vs_t | Sr_F_x

Tanque: File Edit Compile Help

```
/* Tanque con descarga gravitatoria
Sin unidades */

math tanque {

  // Parámetros de simulación
  realDomain t; // tiempo
  t.min   = 0;
  t.max   = 1100;
  t.delta = 10;

  // Parámetros
  real  F0 = 2E-3, A = 0.785, Cv = 4.039E-5,
        rho = 1000, g = 9.81, x = 0.25;

  // Variables
  real F(t);
  real L(t); // Variable de estado

  // Condiciones iniciales
  when(t=t.min) {
    L = 1;
  }

  // ODEs
  L:t = (F0-F)/A;

  // AEs
  F - Cv*x*sqrt(rho*g*L) = 0;
}
```

Source | Run Time | Browser | Debug | Loops | Sensitivity

Optimizer | Monte Carlo | Notes | SemSim | Images | Memory

line 1 terminated normally

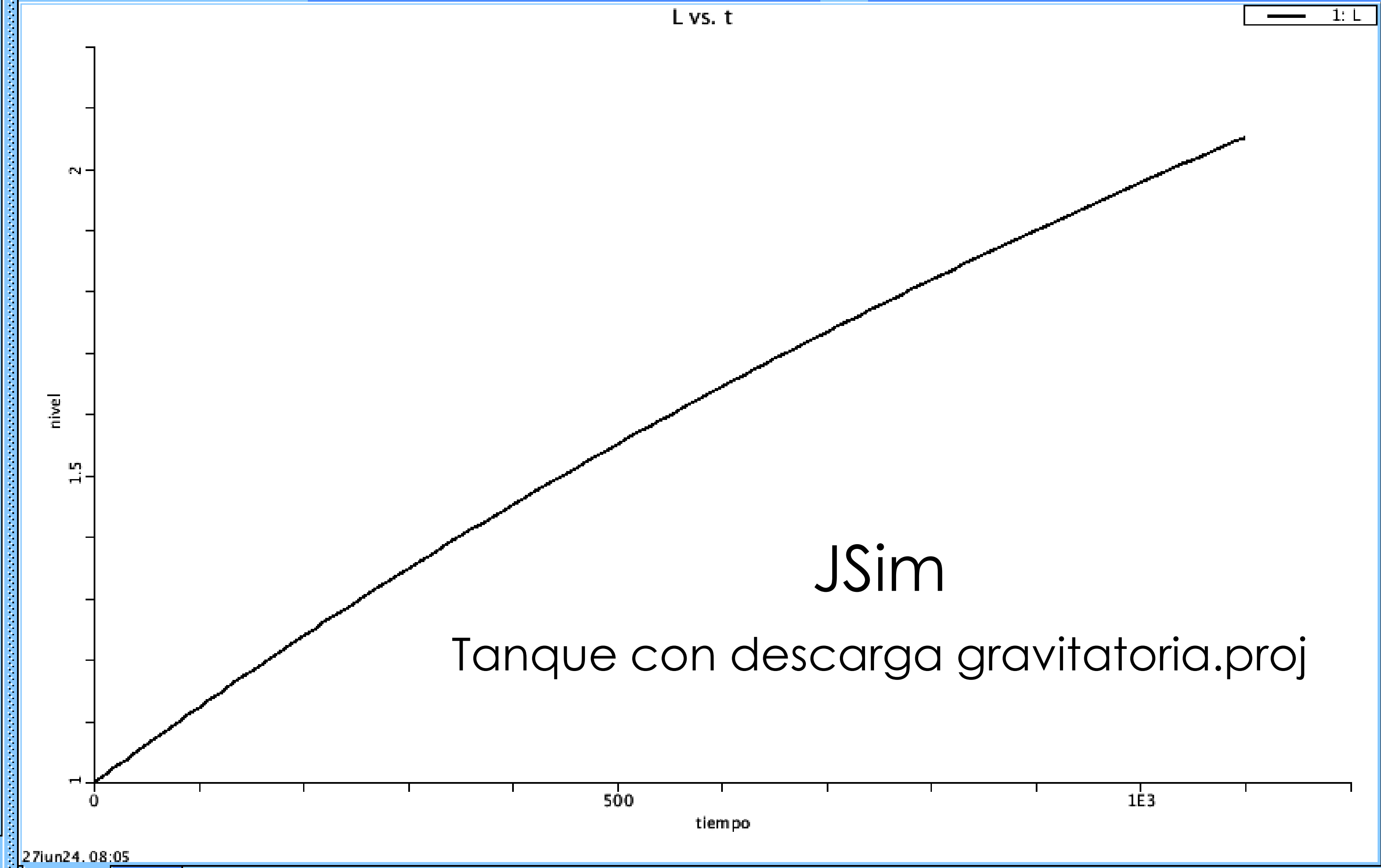
L_vs_t: File View Zoom Help

XY plot | update after run

Data Tanque L | Curve 1 show

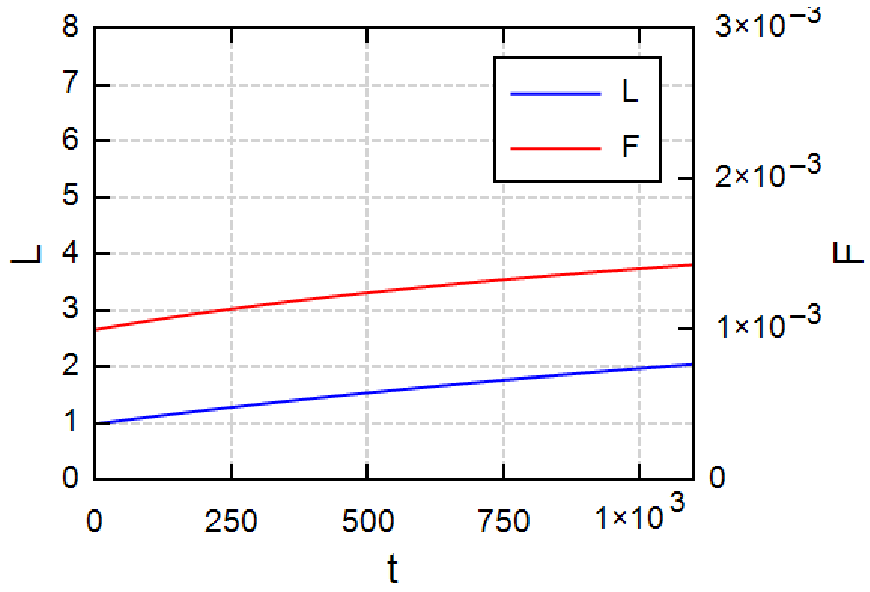
X-Expr

Y Log Autoscale Min 0 Max 1



Graph | Text

SMath Studio



$$\begin{cases} \text{augment}(tv, Lv) \\ \text{augment}(tv, Fv) \end{cases}$$

Tanque con descarga gravitatoria.sm

Modelo del sistema

$$x(t) := \text{if } t < 0 \;\; 0.5 \;\; \text{else} \;\; 0.25$$

AEs

$$F(t, L) := Cv \cdot x(t) \cdot \sqrt{rho \cdot g \cdot L}$$

ODEs

$$dL(t, L) := \frac{F0 - F(t, L)}{A}$$

Simulación del tanque

$$D(t, X) := \begin{vmatrix} [L] := X \\ [dL(t, L)] \end{vmatrix}$$

$$X0 := [1]$$

$$t_0 := 0 \qquad t_f := 1100 \qquad N := 500$$

$$y := \text{Rkadapt}(X0, t_0, t_f, N, D)$$

Prompt para el tanque

Actúa como un experto en Ingeniería Química. Crea el modelo de espacio de estados para un tanque abierto a la atmósfera y perfectamente agitado. El tanque recibe una corriente de alimentación de agua con caudal volumétrico F_0 y densidad \rho_0. El tanque descarga por su base, por gravedad, una corriente con caudal volumétrico F y densidad \rho. La descarga se realiza a través de una válvula lineal con apertura x y coeficiente de tamaño Cv que descarga a la atmósfera. En el tanque, el nivel del fluido es L. El área transversal del tanque es A.

ChatGPT 4o

Actúa como un experto en Ingeniería Química. Crea el modelo de espacio de estados para un tanque abierto a la atmósfera y perfectamente agitado. El tanque recibe una corriente de alimentación de agua con caudal volumétrico F_0 y densidad \rho_0. El tanque descarga por su base, por gravedad, una corriente con caudal volumétrico F y densidad \rho. La descarga se realiza a través de una válvula lineal con apertura x y coeficiente de tamaño Cv que descarga a la atmósfera. En el tanque, el nivel del fluido es L. El área transversal del tanque es A.

🧾 Resumen del modelo en notación estándar

$$\boxed{\begin{aligned} \dot{L}(t) &= \frac{1}{A}\left(F_0(t) - C_v\, x(t)\, \sqrt{g\, L(t)}\right) \\ y(t) &= L(t) \end{aligned}}$$

Falta la densidad

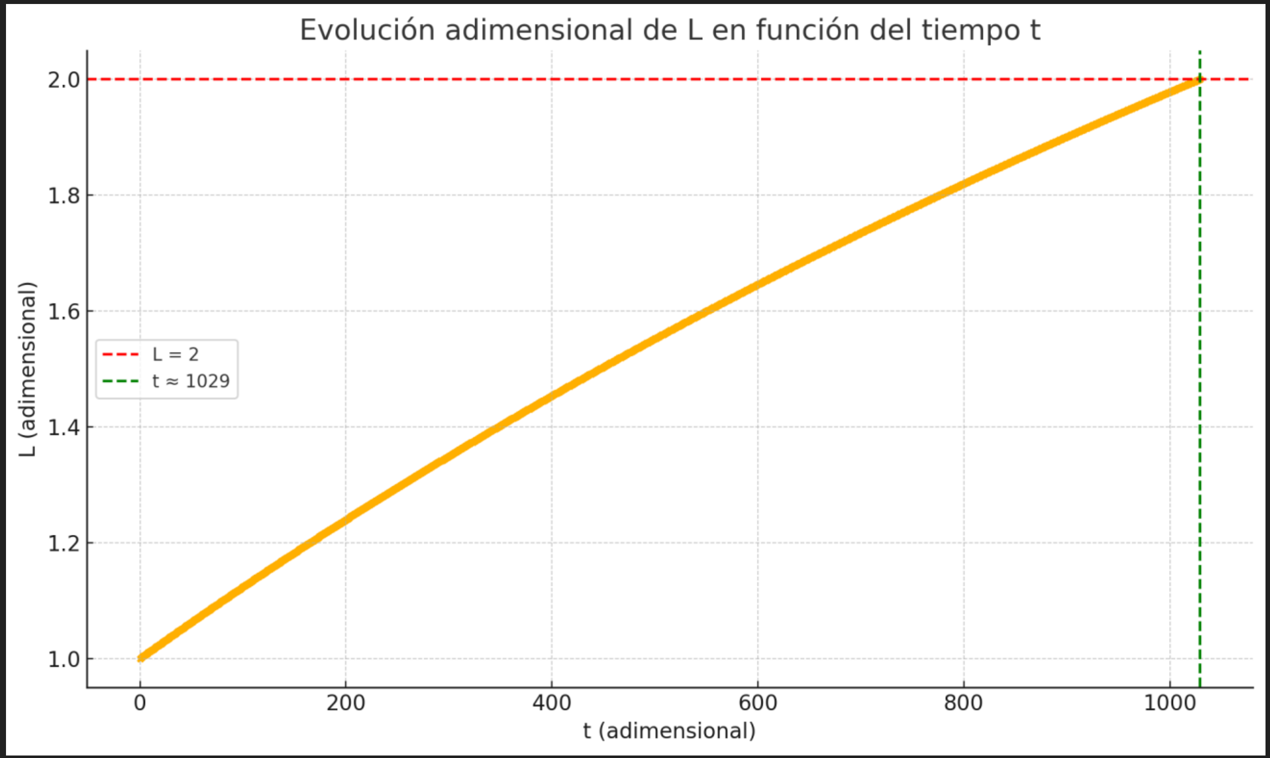
- **Estado:** $L(t)$
- **Entradas:** $F_0(t), x(t)$
- **Salida:** $y(t) = L(t)$

Prompt para el tanque

Para la ecuación diferencial dL/dt = (F_0 - Cv*x*sqrt(\rho*g*L))/A, con L = 1 para t = 0, se tienen los siguientes datos: F_0 = 0.002, \rho = 1000, g = 9.81, A = 0.785, x = 0.25 y Cv = 4.039*10^-5. Crea una gráfica de dispersión donde muestres la evolución de L en función de t. También, determina en qué valor de t, L alcanza el valor de 2. Considera que todos los datos y variables son adimensionales.

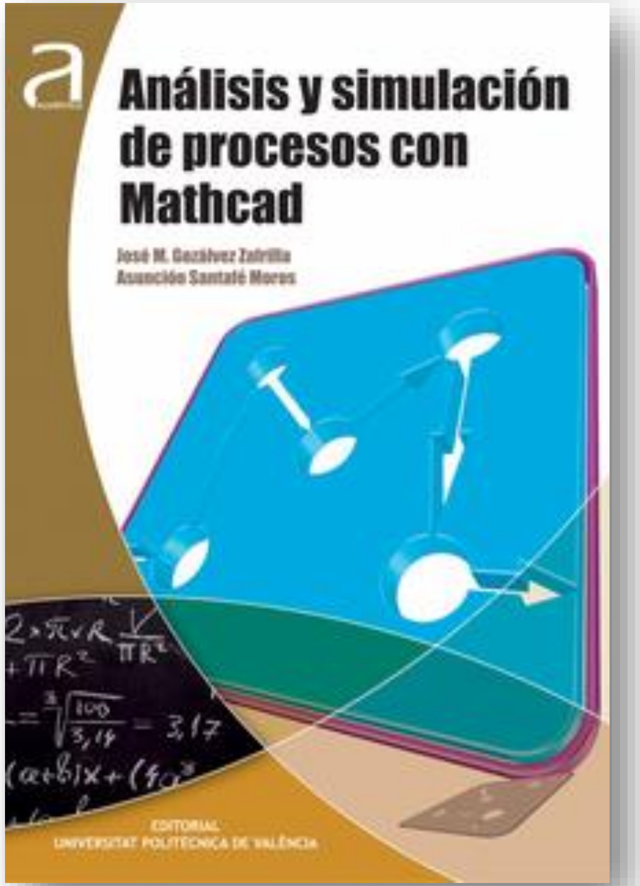
ChatGPT 4o

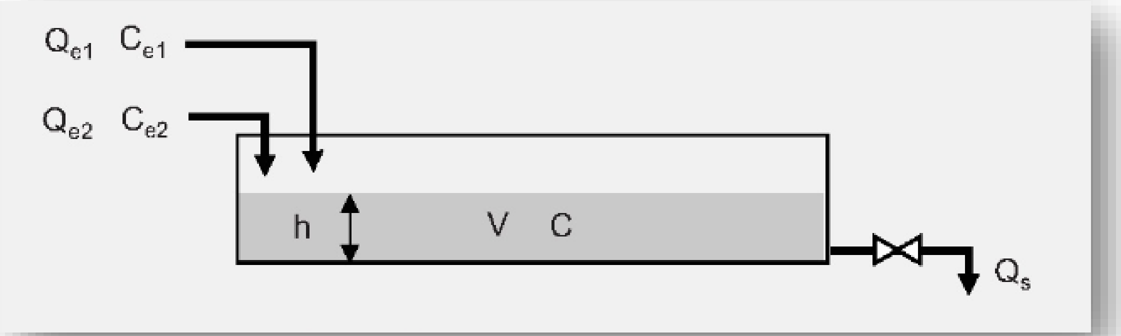
Para la ecuación diferencial dL/dt = (F_0 - Cv*x*sqrt(\rho*g*L))/A, con L = 1 para t = 0, se tienen los siguientes datos: F_0 = 0.002, \rho = 1000, g = 9.81, A = 0.785, x = 0.25 y Cv = 4.039*10^-5. Crea una gráfica de dispersión donde muestres la evolución de L en función de t. También, determina en qué valor de t, L alcanza el valor de 2. Considera que todos los datos y variables son adimensionales.



La gráfica muestra la evolución adimensional de L en función del tiempo t. En este caso, el valor $L = 2$ se alcanza aproximadamente en $t = 1029$ (adimensional). ¿Deseás que lo exprese con otra resolución temporal o te gustaría ver la tabla de valores generados? [>_]

Piscina





Ver archivo Piscina.xmcd

Mapa curricular de tanque con descarga gravitatoria

1. Modelo de espacio de estados
2. Balances
3. Ecuaciones constitutivas
4. Simplificación
5. Modelo estacionario
6. Simulación dinámica