Simulación y Optimización

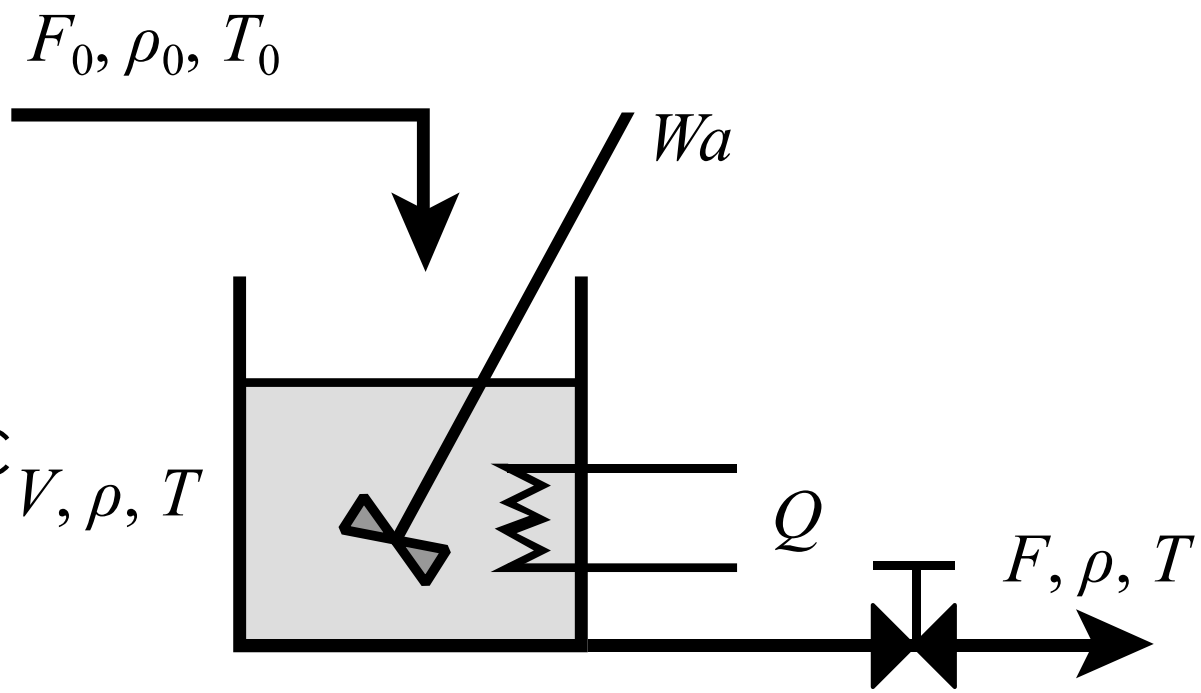
Modelado
Parte IV

Enrique E. Tarifa, Facultad de Ingeniería, UNJu

Tanque calefaccionado

Tanque calefaccionado

- Tanque calefaccionado
- Estado estacionario anterior
- Serpentín en la base
- Vapor saturado a 3 kgf/cm^2
- Wa = 2000 W, T_0 = 25 °C, T_{ini} = 60 °C
- x = 0.25, ¿T cuando rebalse?
- Aumento de 65 % en F_0 (x = 0.5), ¿T cuando rebalse?



Modelo dinámico

Modelo original

$$V\,\rho Cp\frac{dT}{dt}=F_0\,\rho_0\,Cp_0\left(T_0-T\right)+Q+Wa$$

~~$V=AL$~~

$$Q=UA_s\left(T_V-T\right)$$

Modelo simplificado

$$\frac{dT}{dt}=\frac{F_0\,\rho Cp\left(T_0-T\right)+Q+Wa}{AL\,\rho Cp}$$

$$Q=UA_s\left(T_V-T\right)$$

Si el serpentín quedara expuesto parcialmente, el área de intercambio sería función del nivel.

Parámetros en SI

- $F_0 = 2\ l/s = 2\times10^{-3}\ m^3/s$
- $L_{ini} = 1\ m$
- $L_{max} = 2\ m$
- $D = 1\ m \Rightarrow A = 0.785\ m^2$
- $g = 9.81\ m/s^2$
- $\rho = 1\ kg/l = 1000\ kg/m^3$
- $C_v = 4.039\times10^{-5}\ m^{3.5}/kg^{0.5}$
- $T_0 = 25\ °C$
- $T_{ini} = 60\ °C$
- $T_V = 132\ °C$
- $Cp = 4.187\times10^3\ J/(kg\cdot K)$
- $Wa = 2000\ W$

Determinación UA_s

Estado estacionario inicial

$$0 = F_0\,\rho Cp\left(T_0 - T_{ini}\right) + Q_{ini} + Wa$$

$$Q_{ini} = UA_s\left(T_V - T_{ini}\right)$$

Estimación de UA

$$Q_{ini} = F_0\,\rho Cp\left(T_{ini} - T_0\right) - Wa$$

$$UA_s = \frac{Q_{ini}}{T_V - T_{ini}}$$

$$Q_{ini} = 2.91\times10^5\ \text{W}$$

$$UA_s = 4.04\times10^3\ \text{W/K}$$

Simulación dinámica

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

$$\frac{dT}{dt} = \frac{F_0 \rho Cp \left(T_0 - T\right) + Q + Wa}{A L \rho Cp}$$

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

$$Q = UA_s \left(T_V - T\right)$$

Verificar estado estacionario inicial.

Ver Tanque calefaccionado.mmd

```
{Tanque calefaccionado}

METHOD RK4

STARTTIME = 0
STOPTIME = 1100
DT = 10

; Inicialización
INIT L = 1
INIT T = 60

; Sistema ODEs
L'  = (F0-F)/A
T' = (F0*rho*Cp*(T0-T)+Q+Wa)/(A*L*rho*Cp)

; Sistema AEs
F = Cv*x*sqrt(rho*g*L)
Q = UAs*(Tv-T)

; Datos
F0 = 2E-3
A = 0.785
Cv = 4.039E-5
rho = 1000
g = 9.81
x = 0.5

Cp = 4.187E3
UAs = 4.04E3
T0 = 25
Tv = 132
Wa = 2000
```

tanque_calefaccionado.m

```
% Tanque calefaccionado
% 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 T] = num2cell(X'){1,:};

  % Recupera variables Y
  Y = AEs(t,X);
  [A F0 F rho Cp T0 Q Wa] = num2cell(Y){1,:};

  % Ecuaciones diferenciales
  dL = (F0 - F)/A;
  dT = (F0*rho*Cp*(T0-T)+Q+Wa)/(A*L*rho*Cp);

  dX = [dL dT]'; % 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 T] = num2cell(X'){1,:};

  % Parámetros
  F0 = 2E-3; A = 0.785; Cv = 4.039E-5; rho = 1000; g = 9.81;
  Cp = 4.187E3; UAs = 4.04E3; T0 = 25; Tv = 132; Wa = 2000; % Sistema mks

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

  F = Cv*x*sqrt(rho*g*L);
  Q = UAs*(Tv-T);

  Y = [A F0 F rho Cp T0 Q Wa];
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
  Tini = 60; % °C
  Xini = [Lini Tini]; % Inicializa las variables de estado

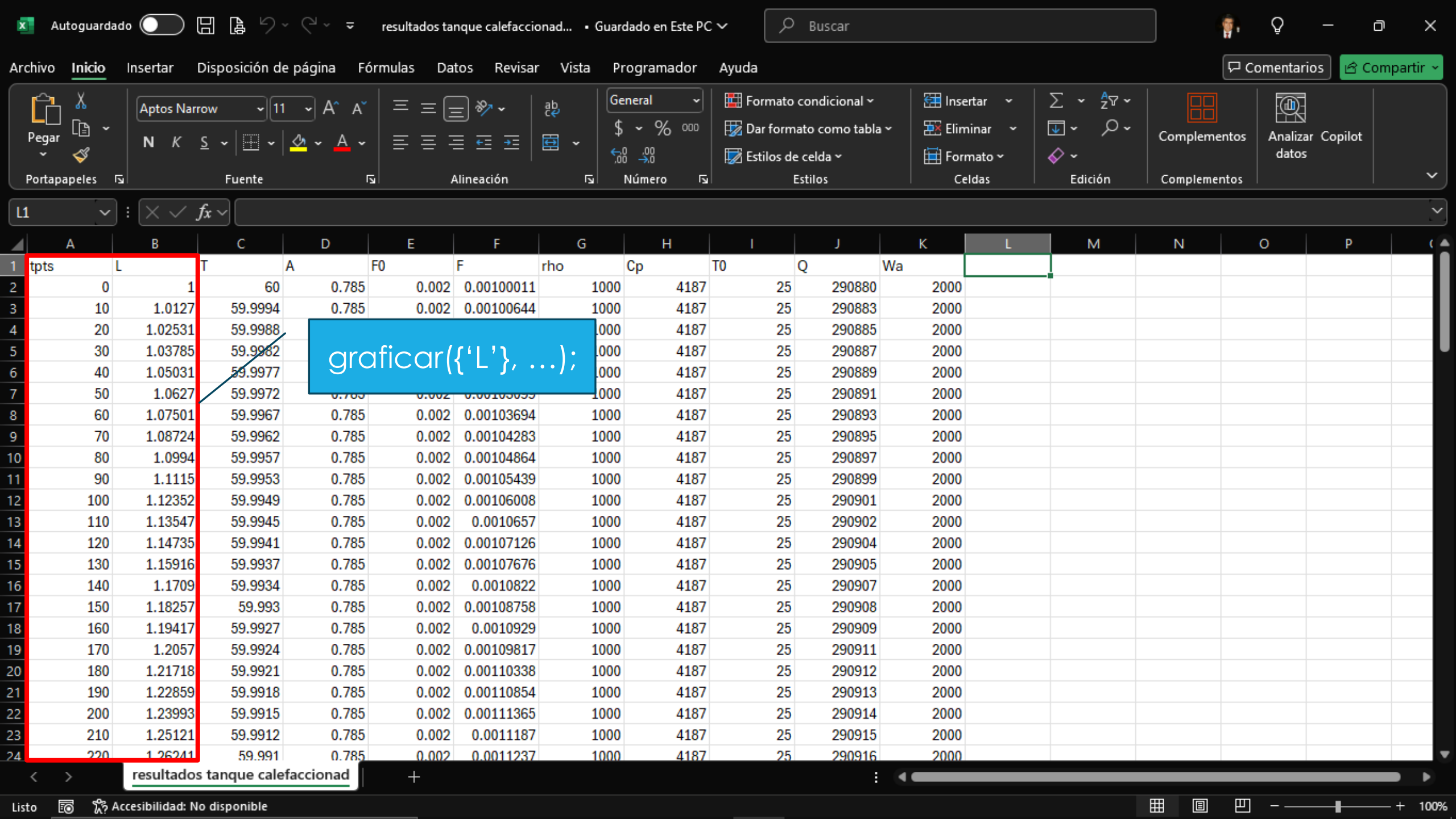
  % Leyendas
  LX = {'L' 'T'}; % Leyendas de las variables X
  LY = {'A' 'F0' 'F' 'rho' 'Cp' 'T0' 'Q' 'Wa'}; % Leyendas de las variables Y
endfunction % inicializar
```

```
% Análisis
function analizar(LX,LY,tpts,X,Y)
  % Análisis de resultados.
  exportar('resultados.csv');

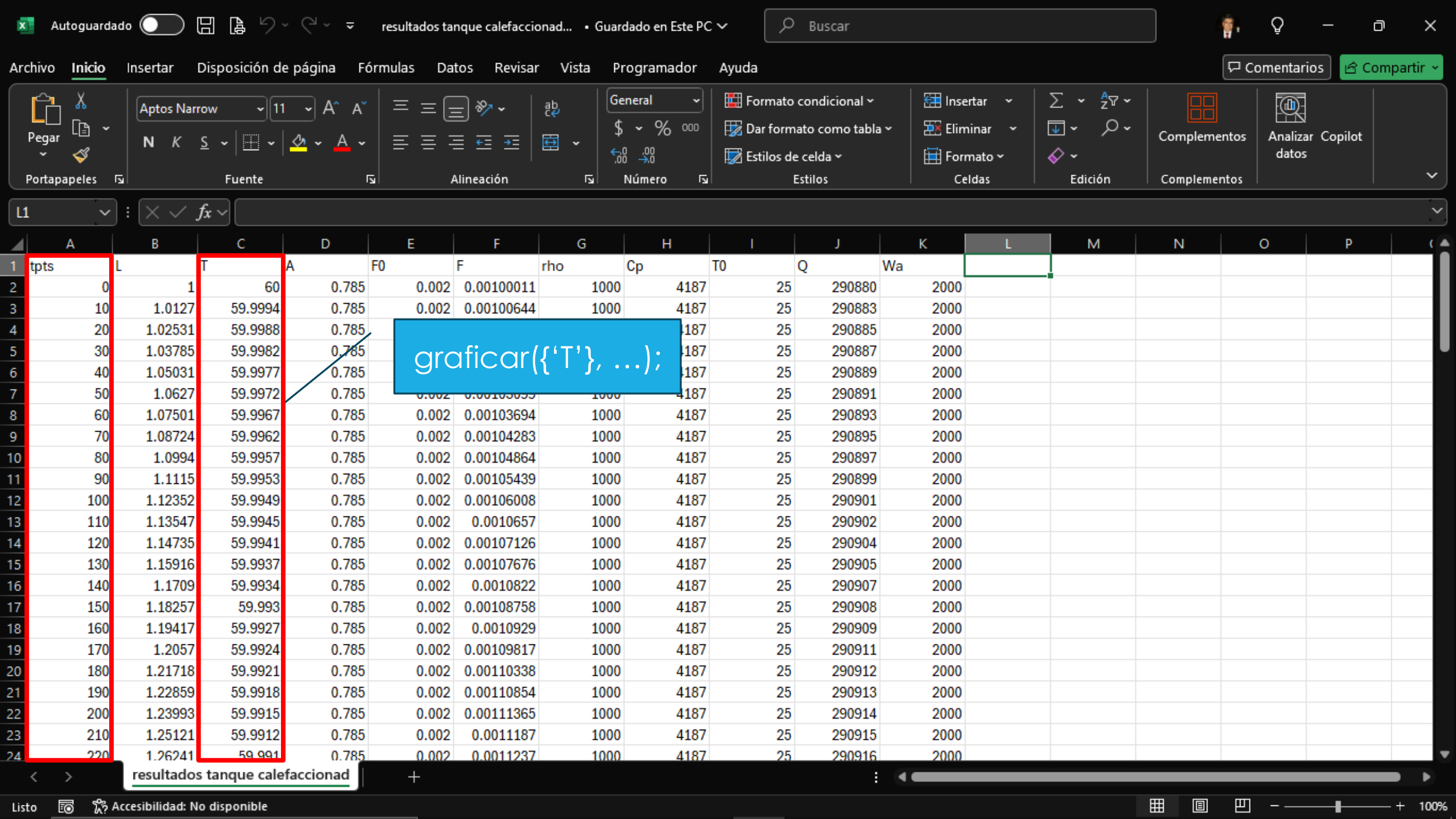
  graficar({'L'}, 'Nivel vs. tiempo', 's', 'm', [0 3]);
  graficar({'F0' 'F'}, 'Caudales vs. tiempo', 's', 'm^3/s', [0 4E-3]);
  graficar({'T'}, 'Temperatura vs. tiempo', 's', '°C', [0 120]);

  % 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.']);

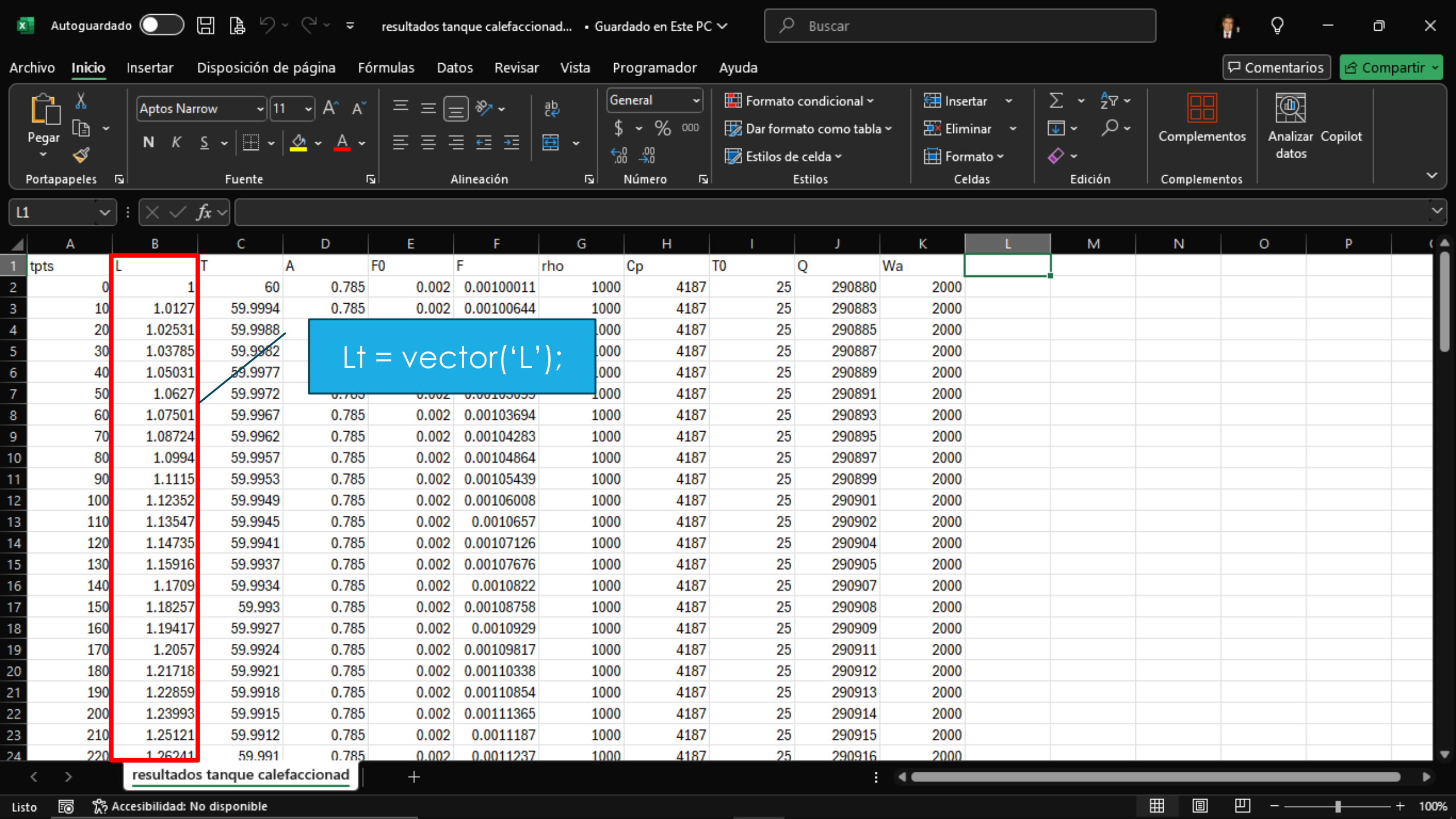
    Tt = vector('T'); % Recupera el vector de temperaturas.
    Tr = interp1(tpts, Tt, tr); % Interpola la temperatura para tr.
    disp(['La temperatura del tanque en ese tiempo fue ' num2str(Tr) ' °C.']);
  endif
endfunction % analizar
```



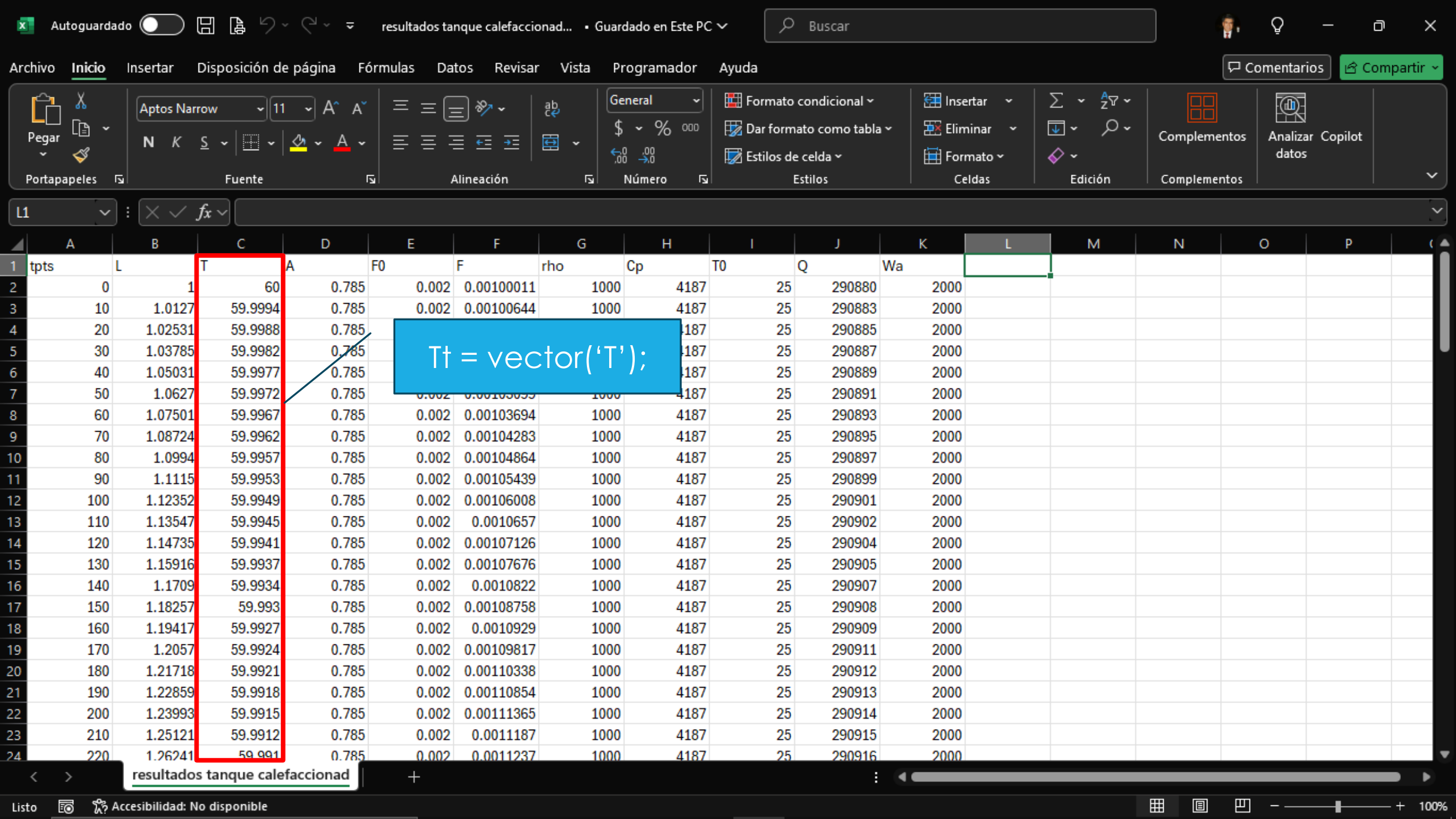
	A	B	C	D	E	F	G	H	I	J	K	L
1	tpts	L	T	A	F0	F	rho	Cp	T0	Q	Wa	
2	0	1	60	0.785	0.002	0.00100011	1000	4187	25	290880	2000	
3	10	1.0127	59.9994	0.785	0.002	0.00100644	1000	4187	25	290883	2000	
4	20	1.02531	59.9988				1000	4187	25	290885	2000	
5	30	1.03785	59.9982				1000	4187	25	290887	2000	
6	40	1.05031	59.9977				1000	4187	25	290889	2000	
7	50	1.0627	59.9972				1000	4187	25	290891	2000	
8	60	1.07501	59.9967	0.785	0.002	0.00103694	1000	4187	25	290893	2000	
9	70	1.08724	59.9962	0.785	0.002	0.00104283	1000	4187	25	290895	2000	
10	80	1.0994	59.9957	0.785	0.002	0.00104864	1000	4187	25	290897	2000	
11	90	1.1115	59.9953	0.785	0.002	0.00105439	1000	4187	25	290899	2000	
12	100	1.12352	59.9949	0.785	0.002	0.00106008	1000	4187	25	290901	2000	
13	110	1.13547	59.9945	0.785	0.002	0.0010657	1000	4187	25	290902	2000	
14	120	1.14735	59.9941	0.785	0.002	0.00107126	1000	4187	25	290904	2000	
15	130	1.15916	59.9937	0.785	0.002	0.00107676	1000	4187	25	290905	2000	
16	140	1.1709	59.9934	0.785	0.002	0.0010822	1000	4187	25	290907	2000	
17	150	1.18257	59.993	0.785	0.002	0.00108758	1000	4187	25	290908	2000	
18	160	1.19417	59.9927	0.785	0.002	0.0010929	1000	4187	25	290909	2000	
19	170	1.2057	59.9924	0.785	0.002	0.00109817	1000	4187	25	290911	2000	
20	180	1.21718	59.9921	0.785	0.002	0.00110338	1000	4187	25	290912	2000	
21	190	1.22859	59.9918	0.785	0.002	0.00110854	1000	4187	25	290913	2000	
22	200	1.23993	59.9915	0.785	0.002	0.00111365	1000	4187	25	290914	2000	
23	210	1.25121	59.9912	0.785	0.002	0.0011187	1000	4187	25	290915	2000	
24	220	1.26241	59.991	0.785	0.002	0.0011237	1000	4187	25	290916	2000	



	A	B	C	D	E	F	G	H	I	J	K	L
1	tpts	L	T	A	F0	F	rho	Cp	T0	Q	Wa	
2	0	1	60	0.785	0.002	0.00100011	1000	4187	25	290880	2000	
3	10	1.0127	59.9994	0.785	0.002	0.00100644	1000	4187	25	290883	2000	
4	20	1.02531	59.9988	0.785				4187	25	290885	2000	
5	30	1.03785	59.9982	0.785				4187	25	290887	2000	
6	40	1.05031	59.9977	0.785				4187	25	290889	2000	
7	50	1.0627	59.9972	0.785				4187	25	290891	2000	
8	60	1.07501	59.9967	0.785	0.002	0.00103694	1000	4187	25	290893	2000	
9	70	1.08724	59.9962	0.785	0.002	0.00104283	1000	4187	25	290895	2000	
10	80	1.0994	59.9957	0.785	0.002	0.00104864	1000	4187	25	290897	2000	
11	90	1.1115	59.9953	0.785	0.002	0.00105439	1000	4187	25	290899	2000	
12	100	1.12352	59.9949	0.785	0.002	0.00106008	1000	4187	25	290901	2000	
13	110	1.13547	59.9945	0.785	0.002	0.0010657	1000	4187	25	290902	2000	
14	120	1.14735	59.9941	0.785	0.002	0.00107126	1000	4187	25	290904	2000	
15	130	1.15916	59.9937	0.785	0.002	0.00107676	1000	4187	25	290905	2000	
16	140	1.1709	59.9934	0.785	0.002	0.0010822	1000	4187	25	290907	2000	
17	150	1.18257	59.993	0.785	0.002	0.00108758	1000	4187	25	290908	2000	
18	160	1.19417	59.9927	0.785	0.002	0.0010929	1000	4187	25	290909	2000	
19	170	1.2057	59.9924	0.785	0.002	0.00109817	1000	4187	25	290911	2000	
20	180	1.21718	59.9921	0.785	0.002	0.00110338	1000	4187	25	290912	2000	
21	190	1.22859	59.9918	0.785	0.002	0.00110854	1000	4187	25	290913	2000	
22	200	1.23993	59.9915	0.785	0.002	0.00111365	1000	4187	25	290914	2000	
23	210	1.25121	59.9912	0.785	0.002	0.0011187	1000	4187	25	290915	2000	
24	220	1.26241	59.991	0.785	0.002	0.0011237	1000	4187	25	290916	2000	



	A	B	C	D	E	F	G	H	I	J	K	L
1	tpts	L	T	A	F0	F	rho	Cp	T0	Q	Wa	
2	0	1	60	0.785	0.002	0.00100011	1000	4187	25	290880	2000	
3	10	1.0127	59.9994	0.785	0.002	0.00100644	1000	4187	25	290883	2000	
4	20	1.02531	59.9988				1000	4187	25	290885	2000	
5	30	1.03785	59.9982				1000	4187	25	290887	2000	
6	40	1.05031	59.9977				1000	4187	25	290889	2000	
7	50	1.0627	59.9972				1000	4187	25	290891	2000	
8	60	1.07501	59.9967	0.785	0.002	0.00103694	1000	4187	25	290893	2000	
9	70	1.08724	59.9962	0.785	0.002	0.00104283	1000	4187	25	290895	2000	
10	80	1.0994	59.9957	0.785	0.002	0.00104864	1000	4187	25	290897	2000	
11	90	1.1115	59.9953	0.785	0.002	0.00105439	1000	4187	25	290899	2000	
12	100	1.12352	59.9949	0.785	0.002	0.00106008	1000	4187	25	290901	2000	
13	110	1.13547	59.9945	0.785	0.002	0.0010657	1000	4187	25	290902	2000	
14	120	1.14735	59.9941	0.785	0.002	0.00107126	1000	4187	25	290904	2000	
15	130	1.15916	59.9937	0.785	0.002	0.00107676	1000	4187	25	290905	2000	
16	140	1.1709	59.9934	0.785	0.002	0.0010822	1000	4187	25	290907	2000	
17	150	1.18257	59.993	0.785	0.002	0.00108758	1000	4187	25	290908	2000	
18	160	1.19417	59.9927	0.785	0.002	0.0010929	1000	4187	25	290909	2000	
19	170	1.2057	59.9924	0.785	0.002	0.00109817	1000	4187	25	290911	2000	
20	180	1.21718	59.9921	0.785	0.002	0.00110338	1000	4187	25	290912	2000	
21	190	1.22859	59.9918	0.785	0.002	0.00110854	1000	4187	25	290913	2000	
22	200	1.23993	59.9915	0.785	0.002	0.00111365	1000	4187	25	290914	2000	
23	210	1.25121	59.9912	0.785	0.002	0.0011187	1000	4187	25	290915	2000	
24	220	1.26241	59.991	0.785	0.002	0.0011237	1000	4187	25	290916	2000	



	A	B	C	D	E	F	G	H	I	J	K	L
1	tpts	L	T	A	F0	F	rho	Cp	T0	Q	Wa	
2	0	1	60	0.785	0.002	0.00100011	1000	4187	25	290880	2000	
3	10	1.0127	59.9994	0.785	0.002	0.00100644	1000	4187	25	290883	2000	
4	20	1.02531	59.9988	0.785				4187	25	290885	2000	
5	30	1.03785	59.9982	0.785				4187	25	290887	2000	
6	40	1.05031	59.9977	0.785				4187	25	290889	2000	
7	50	1.0627	59.9972	0.785				4187	25	290891	2000	
8	60	1.07501	59.9967	0.785	0.002	0.00103694	1000	4187	25	290893	2000	
9	70	1.08724	59.9962	0.785	0.002	0.00104283	1000	4187	25	290895	2000	
10	80	1.0994	59.9957	0.785	0.002	0.00104864	1000	4187	25	290897	2000	
11	90	1.1115	59.9953	0.785	0.002	0.00105439	1000	4187	25	290899	2000	
12	100	1.12352	59.9949	0.785	0.002	0.00106008	1000	4187	25	290901	2000	
13	110	1.13547	59.9945	0.785	0.002	0.0010657	1000	4187	25	290902	2000	
14	120	1.14735	59.9941	0.785	0.002	0.00107126	1000	4187	25	290904	2000	
15	130	1.15916	59.9937	0.785	0.002	0.00107676	1000	4187	25	290905	2000	
16	140	1.1709	59.9934	0.785	0.002	0.0010822	1000	4187	25	290907	2000	
17	150	1.18257	59.993	0.785	0.002	0.00108758	1000	4187	25	290908	2000	
18	160	1.19417	59.9927	0.785	0.002	0.0010929	1000	4187	25	290909	2000	
19	170	1.2057	59.9924	0.785	0.002	0.00109817	1000	4187	25	290911	2000	
20	180	1.21718	59.9921	0.785	0.002	0.00110338	1000	4187	25	290912	2000	
21	190	1.22859	59.9918	0.785	0.002	0.00110854	1000	4187	25	290913	2000	
22	200	1.23993	59.9915	0.785	0.002	0.00111365	1000	4187	25	290914	2000	
23	210	1.25121	59.9912	0.785	0.002	0.0011187	1000	4187	25	290915	2000	
24	220	1.26241	59.991	0.785	0.002	0.0011237	1000	4187	25	290916	2000	

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

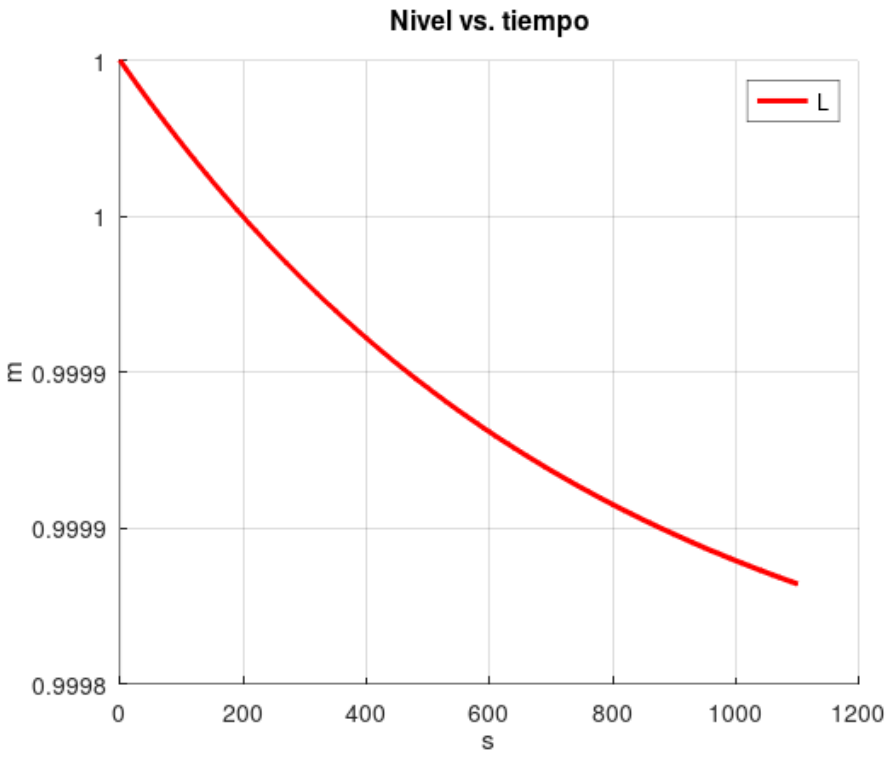
Estado estacionario para x = 0.5

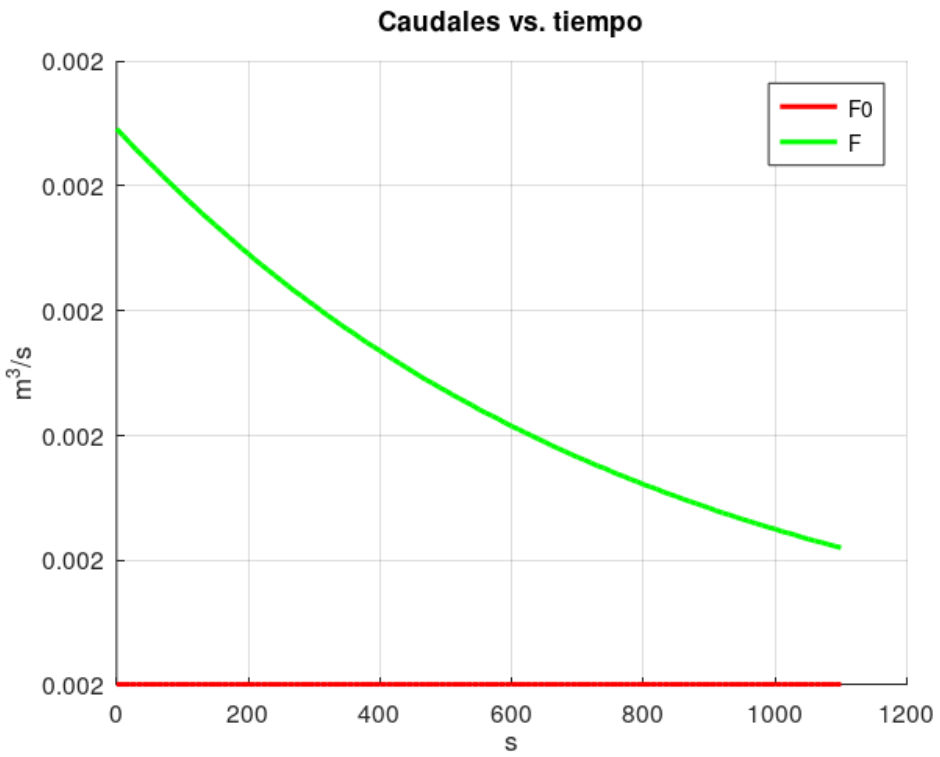
```
% AEs
function Y = AEs(t,X)
  % En Y devuelve el vector fila de variables requeridas por ODEs o a graficar.
...

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

...
endfunction % AEs
```

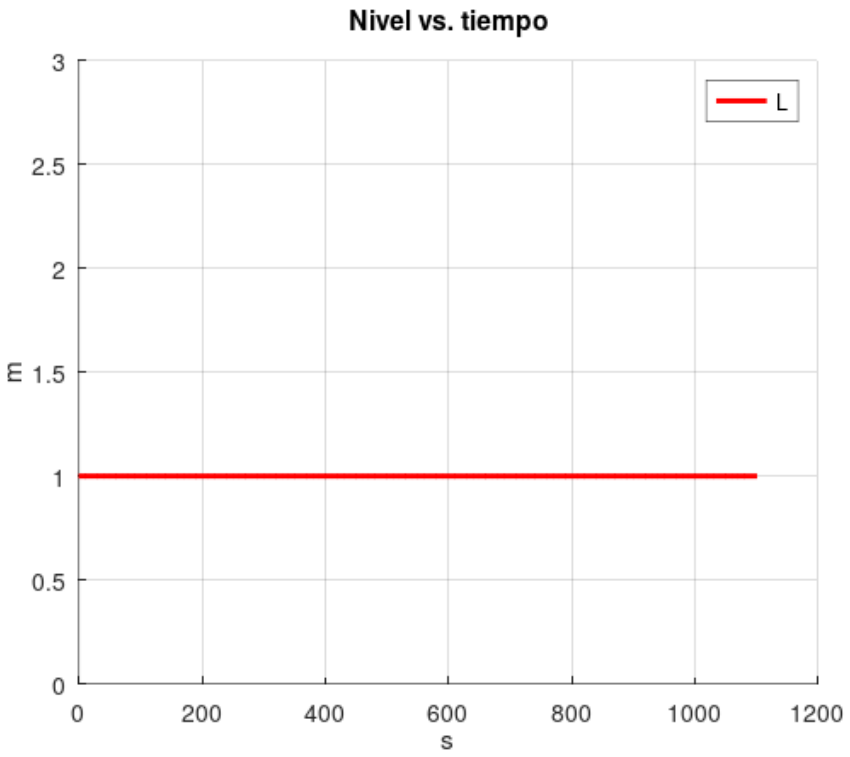
Estado estacionario para x = 0.5

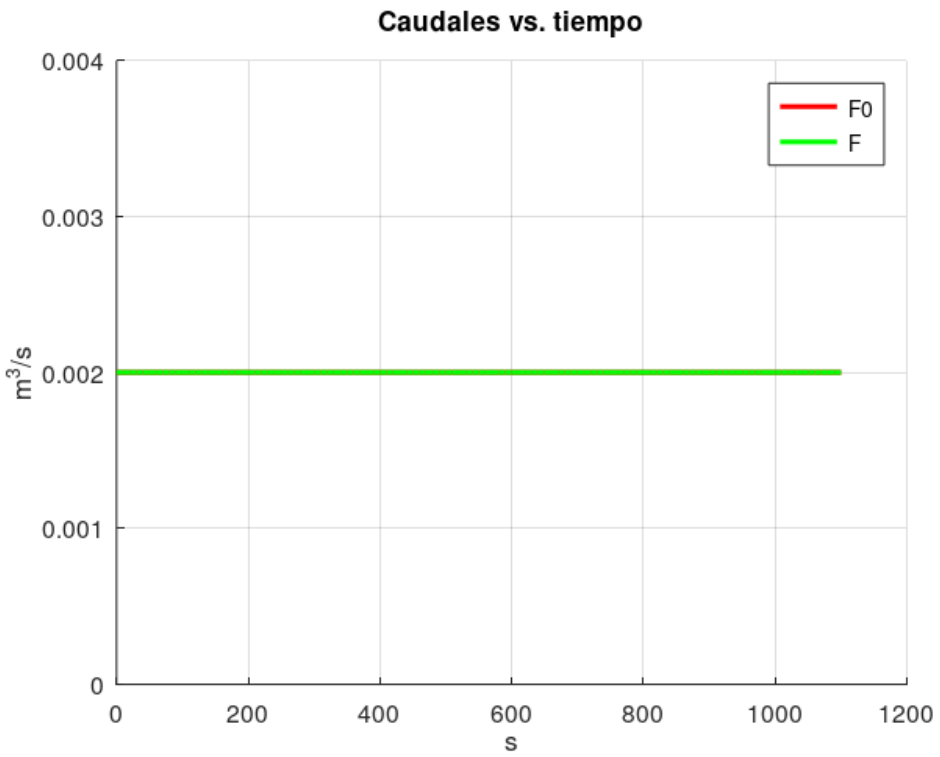





```
graficar({'L'},'Nivel vs. tiempo', 's', 'm');
graficar({'F0' 'F'},'Caudales vs. tiempo', 's', 'm^3/s');
```

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]);
```

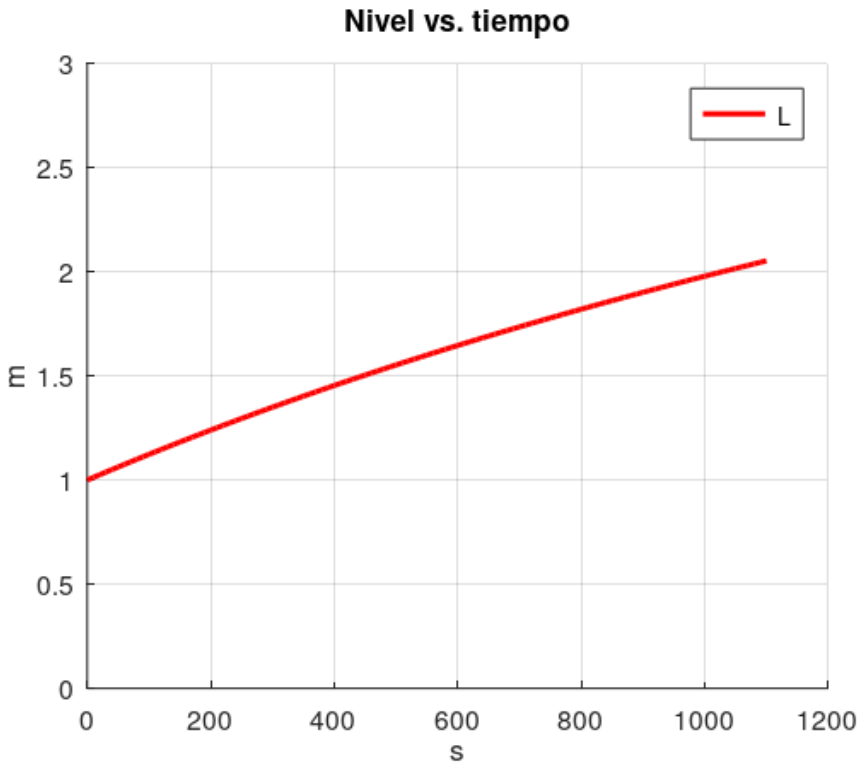
Simulación para x = 0.25

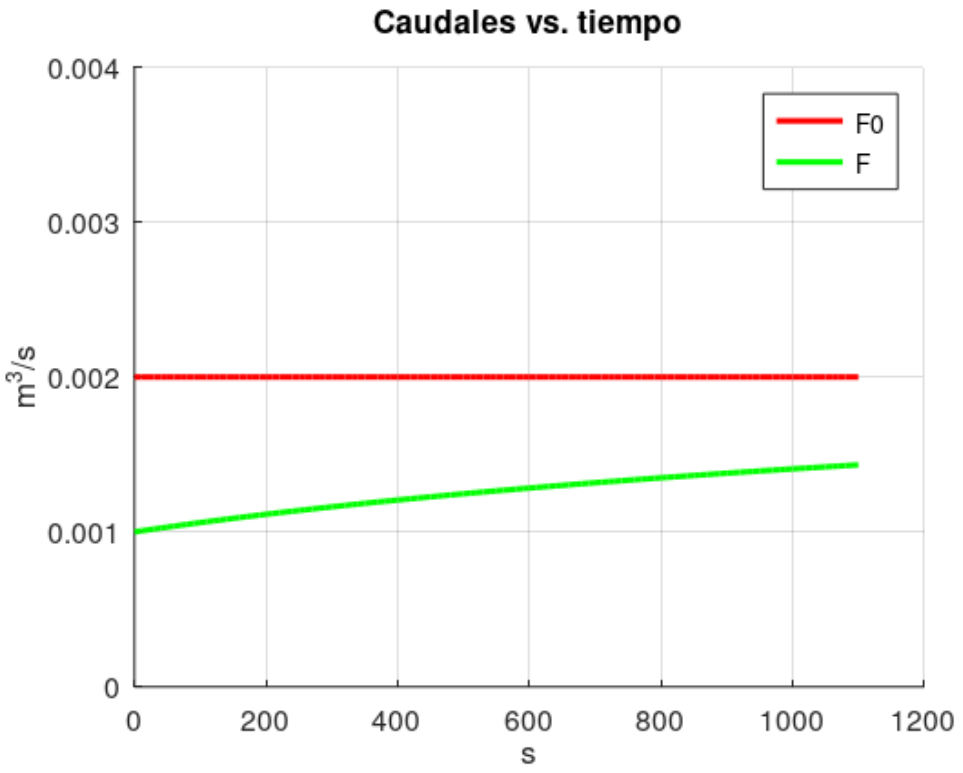
```
% AEs
function Y = AEs(t,X)
  % En Y devuelve el vector fila de variables a requeridas por ODEs o a graficar.
...

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

...
endfunction % AEs
```

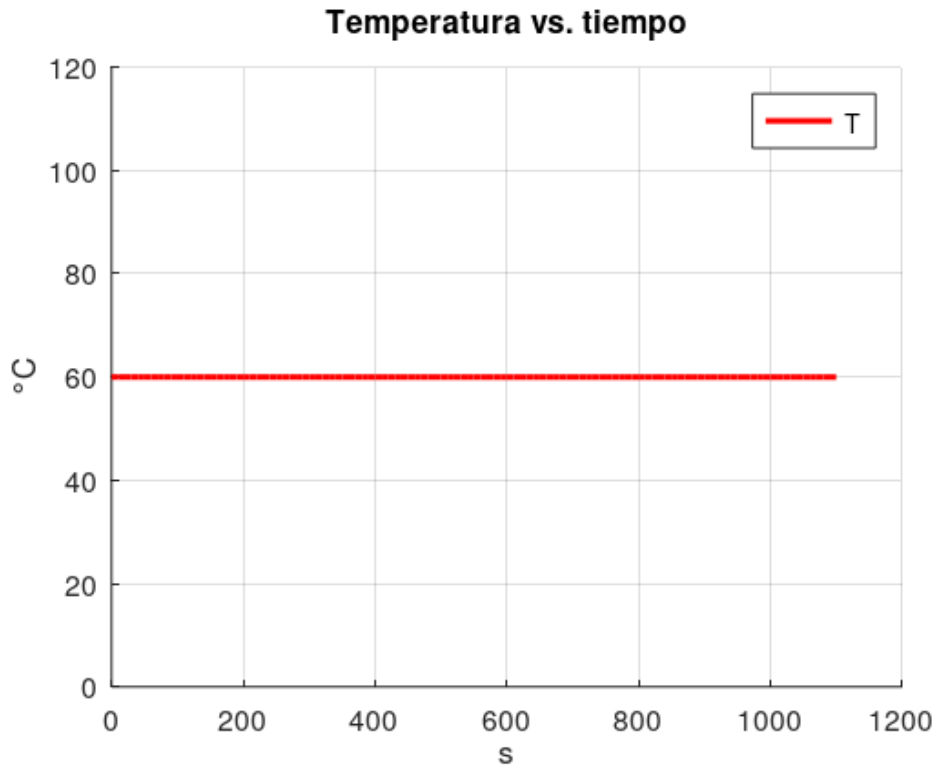
Resultados para x = 0.25






```
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




```
graficar({'T'}, 'Temperatura vs. tiempo', 's', '°C', [0 120]);
```

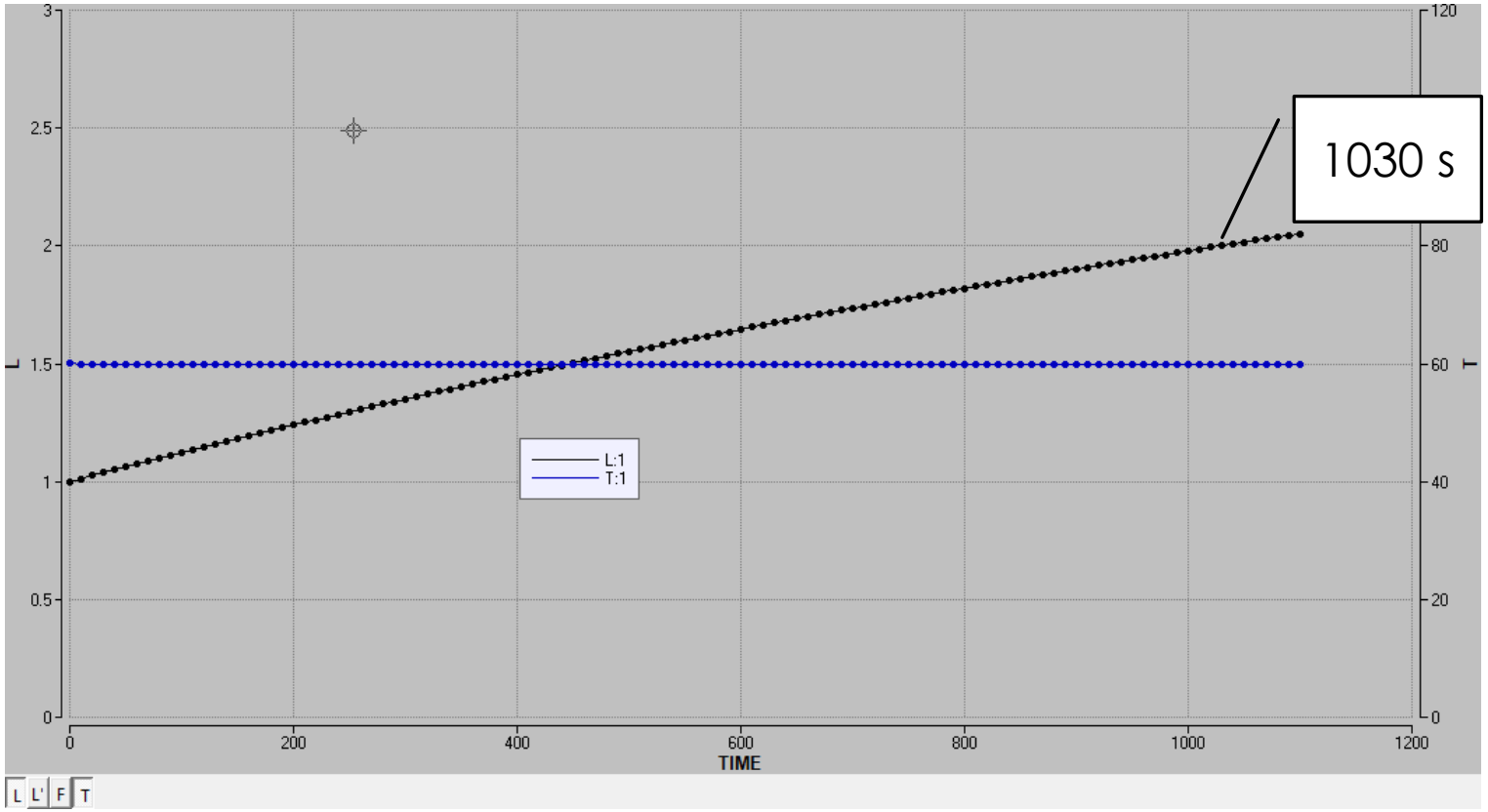
Resultados para x = 0.25

```
Resolvedor v01, 2025

Resolviendo el modelo...
Archivo exportado como "resultados.csv" en el directorio de trabajo.
El tanque rebalsó en el tiempo igual a 1028.4852 s.
La temperatura del tanque en ese tiempo fue 59.9843 °C.

Simulación finalizada.
>>
```

Cierre de válvula, $x = 0.25$



Aumento del 65 % en *F0*, x = 0.5

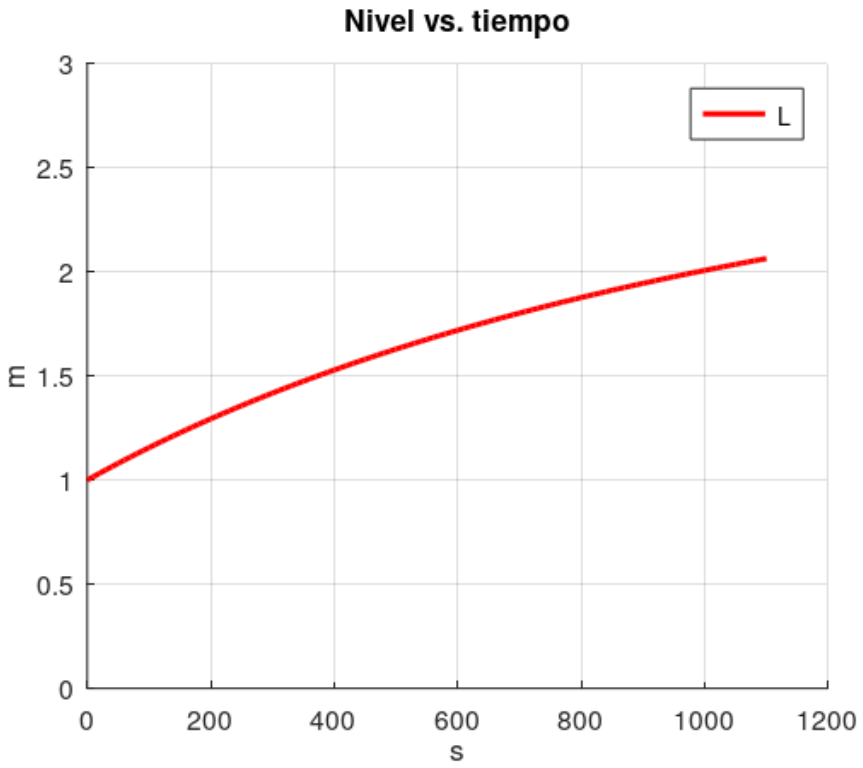
```
% AEs
function Y = AEs(t,X)
  % En Y devuelve el vector fila de variables requeridas por ODEs o a graficar.
...

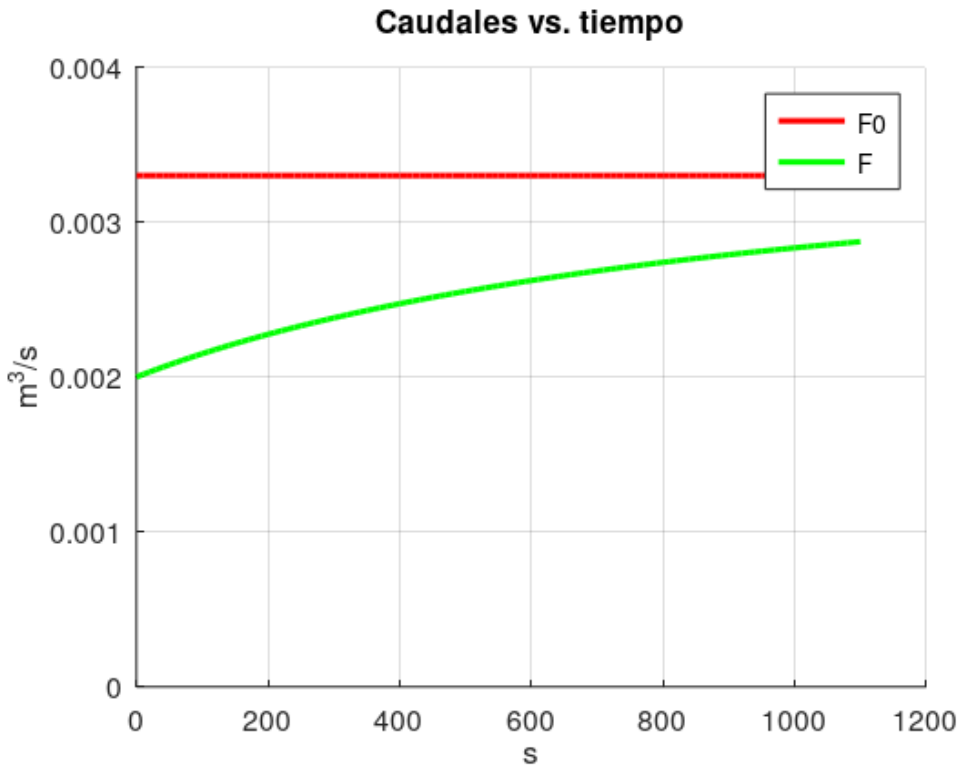
  x = 0.5; % Se fija la apertura de la válvula de descarga.

  % Ecuaciones algebraicas
  if t >= 0
    F0 = 1.65*F0; % Incremento del 65 %
  endif

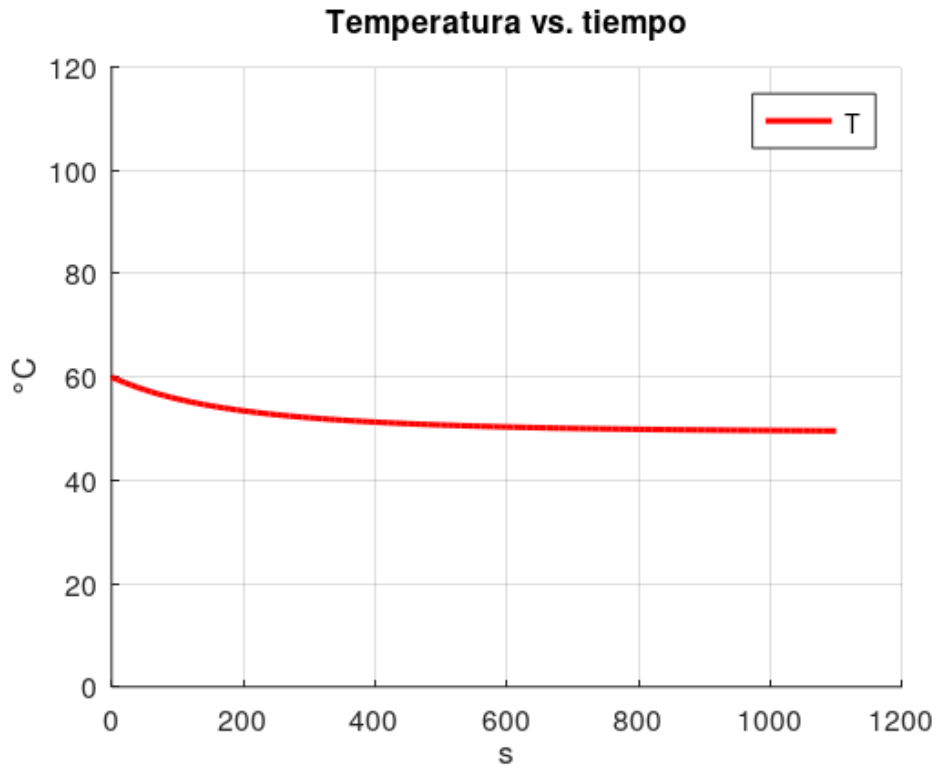
...
endfunction % AEs
```

Aumento del 65 % en *F0*, x = 0.5





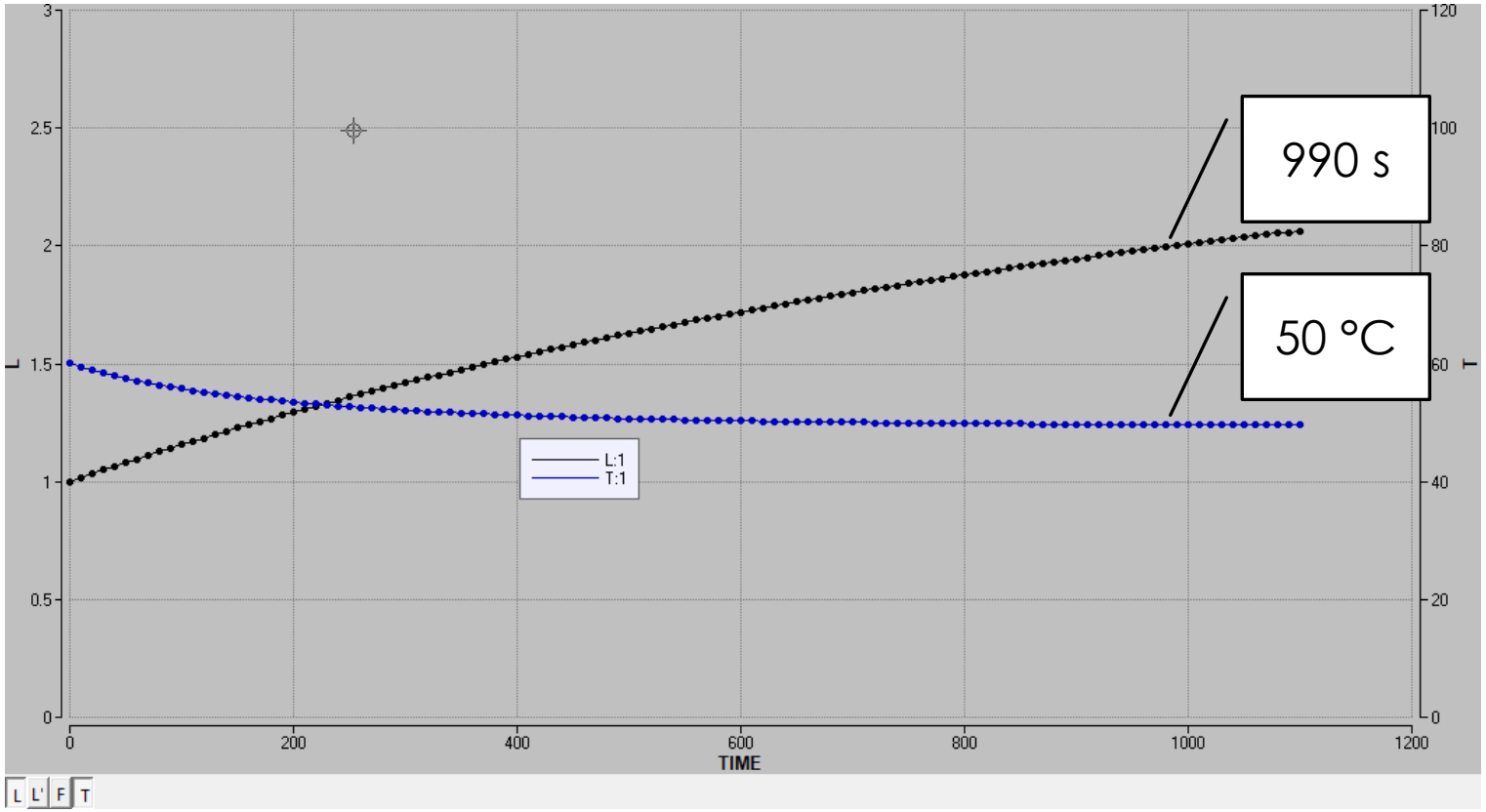
Aumento del 65 % en *F0*, x = 0.5



Aumento del 65 % en *F0*, x = 0.5

```
Resolvedor v01, 2025

Resolviendo el modelo...
Archivo exportado como "resultados.csv" en el directorio de trabajo.
El tanque rebalsó en el tiempo igual a 988.8789 s.
La temperatura del tanque en ese tiempo fue 49.6462 °C.

Simulación finalizada.
>>
```

Aumento del 65 % en *F0*, x = 0.5