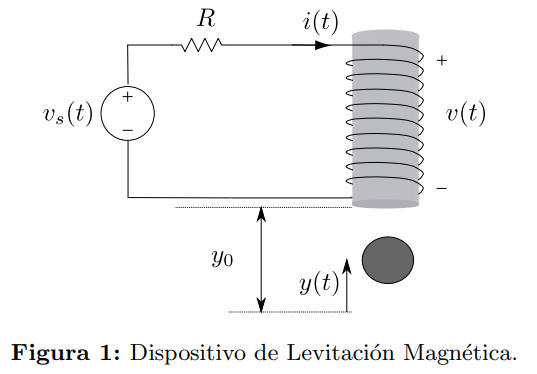
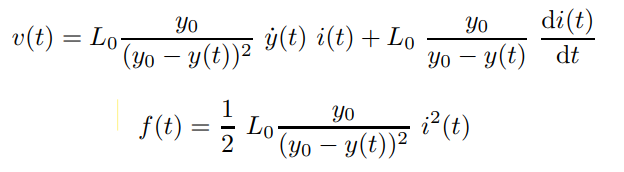
Problema 1. Modelado de la Interacción Electromagnética-Mecánica

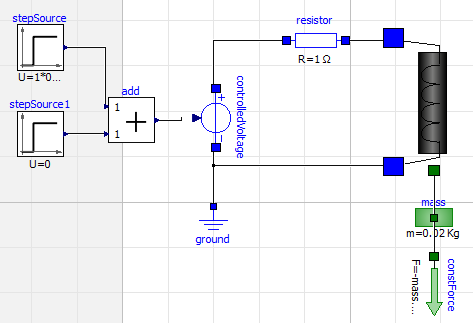




|  |
| --- |
| model **CoillBall**  DSFLib.Mechanical.Translational.Interfaces.Flange flange  extends DSFLib.Circuits.Interfaces.OnePort;  DSFLib.Mechanical.Translational.Units.Position y;  DSFLib.Mechanical.Translational.Units.Force f;  parameter Real Lo = 5.5181e-3;  parameter Real yo = 9e-3;  equation  y = flange.s;  f = flange.f;  f = -(Lo\*(yo/((yo - y)^2))\*(i^2))/2;  v = Lo\*(yo/((yo - y)^2))\*der(y)\*i + (Lo\*(yo/(yo - y)))\*der(i);  end CoillBall; |

Problema 2. Modelo del sistema sin control (Lazo abierto)

Donde será la entrada de control y es el voltaje de equilibrio para que la pelotita quede en la posición .

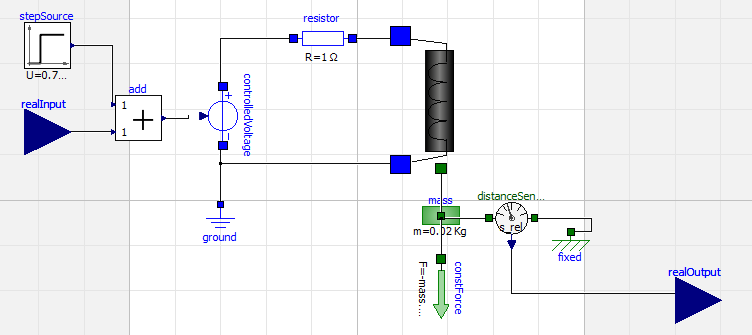


model **MaglevSys**  
DSFLib.Circuits.Components.Resistor resistor(R = 1)  
DSFLib.Mechanical.Translational.Components.Mass mass(m = 0.02, s(start = 0))   
DSFLib.ControlSystems.Actuators.Circuits.ModulatedVoltageSource controlledVoltage  
DSFLib.Circuits.Components.Ground ground  
DSFLib.Mechanical.Translational.Components.ConstForce constForce(F = -mass.m\*9.8)  
DSFLib.ControlSystems.Blocks.Components.StepSource stepSource(U = 1\*0.79959396013)  
DSFLib.ControlSystems.Blocks.Components.Add  
TP4Bellini.CoillBall coillBall(i(start = 0.79959396013))  
DSFLib.ControlSystems.Blocks.Components.StepSource stepSource1(U = 0)

Problema 3. Linealización y Análisis del Modelo

Se agregan conectores de señal al modelo para y para la salida Entonces se linealiza el modelo utilizando OMShell:

linearize(TP4Bellini.MaglevSys,stopTime=0.0)



Matrices Sistema Linealizado:

Analizamos la estabilidad del modelo linealizado.

2 autovalores reales negativos y uno real positivo. Por teorema de Hartman-Grobman el sistema linealizado puede ser utilizado para aproximar al sistema no linealizado en las cercanías del punto de equilibrio ya que no posee autovalores con parte real nula, por ende, las trayectorias serán similares.

Con respecto a la estabilidad del punto de equilibrio, por el Método Indirecto de Lyapunov, el punto de equilibrio es inestable ya que un autovalor presenta parte real positiva.

Problema 4. Control Lineal de Posición

Se obtiene la función transferencia del sistema linealizado:

Sys1 = ss(A,B,C,D); G = tf(sys1);

Función transferencia del controlador PID:

Se pide analizar la posición de los polos para distintos valores de K, eligiendo finalmente un valor no muy grande de K que permita que la constante de tiempo más lenta sea menor que 60 ms.

Esto es equivalente a solicitar que la parte real del polo mas lento sea mayor a la inversa del tau, en este caso 1/60ms. Además, el polo más lento es el que se encuentra más cercano al eje imaginario pues su exponencial decaerá mas lentamente. Entonces se busca un K que cumpla con la condición parte real (negativa) mayor a 1/60ms = 16.667.

Se utiliza un algoritmo para encontrar el K óptimo:

|  |
| --- |
| % Búsqueda iterativa de K  %Se itera en varios valores de K para encontrar el mínimo valor de %cumpla que el tau mas lento sea menor a 60 ms\n  K\_values = logspace(-1, 4, 1000);  K\_opt = 0; % Inicializar K óptimo  poles\_opt = []; % Inicializar polos óptimos  for K = K\_values  sys\_cl = feedback(K \* Gc \* G, 1); % Sistema en lazo cerrado  poles = pole(sys\_cl);  if all(abs(real(poles)) > 16.67)  K\_opt = K;  poles\_opt = poles;  break;  end  end  fprintf('\nValor de K que cumple con la condición: %.5f\n', K\_opt);  disp(poles\_opt); |

El K óptimo para este sistema es **K = 3843** y los polos para el sistema a lazo cerrado con dicho K son:

Todos cumplen con la condición solicitada.