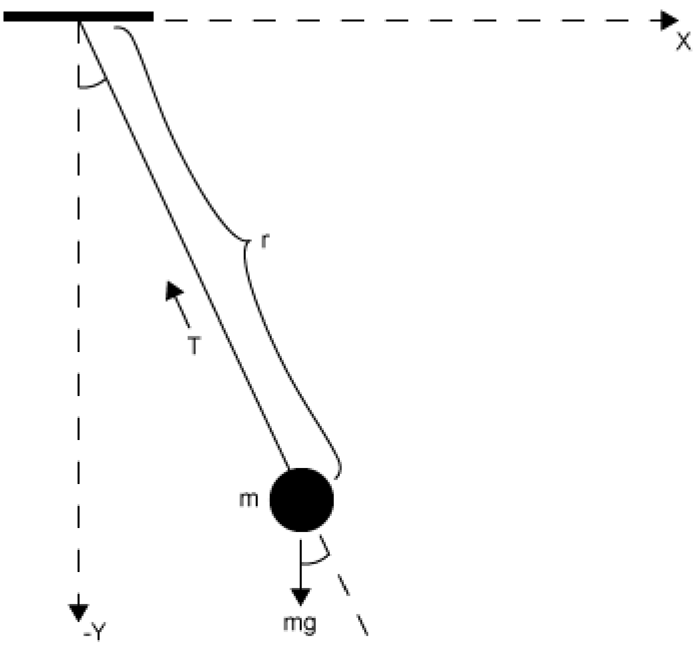
Solve DAE for a simple Pendulum

Example from documentation:

- https://www.mathworks.com/help/symbolic/solve-differential-algebraic-equations.html



State variables

- Horizontal position of pendulum $x(t)$
- Vertical position of pendulum $y(t)$
- Force preventing pendulum from flying away $T(t)$

Parameters

- Pendulum mass m
- Pendulum length r
- Gravitational constant g

DAE system of equations

$$m\frac{d^2 x}{\mathrm{dt}^2} = T(t)\frac{x(t)}{r}$$

$$m\frac{d^2 y}{\mathrm{dt}^2} = T(t)\frac{y(t)}{r} - \mathrm{mg}$$

$$x^2(t) + y^2(t) = r^2$$

```
clear variables

syms x(t) y(t) T(t) m r g

eqn1 = m*diff(x(t), 2) == T(t)/r*x(t)
```

eqn1 =

$$m \frac{\partial^2}{\partial t^2} x(t) = \frac{T(t)\, x(t)}{r}$$

```
eqn2 = m*diff(y(t), 2) == T(t)/r*y(t) - m*g
```

eqn2 =

$$m \frac{\partial^2}{\partial t^2} y(t) = \frac{T(t)\, y(t)}{r} - g\, m$$

```
eqn3 = x(t)^2 + y(t)^2 == r^2
```

eqn3 = $x(t)^2 + y(t)^2 = r^2$

```
eqns = [eqn1 eqn2 eqn3];
vars = [x(t); y(t); T(t)];
origVars = length(vars)
```

origVars = 3

Check Incidence of Variables

```
M = incidenceMatrix(eqns, vars)
```

```
M = 3×3
     1     0     1
     0     1     1
     1     1     0
```

Reduce Differential Order

```
[eqns, vars] = reduceDifferentialOrder(eqns, vars)
```

eqns =

$$\begin{pmatrix} m \frac{\partial}{\partial t} \mathrm{Dxt}(t) - \frac{T(t)\, x(t)}{r} \\ g\, m + m \frac{\partial}{\partial t} \mathrm{Dyt}(t) - \frac{T(t)\, y(t)}{r} \\ -r^2 + x(t)^2 + y(t)^2 \\ \mathrm{Dxt}(t) - \frac{\partial}{\partial t} x(t) \\ \mathrm{Dyt}(t) - \frac{\partial}{\partial t} y(t) \end{pmatrix}$$

vars =

$$\begin{pmatrix} x(t) \\ y(t) \\ T(t) \\ \mathrm{Dxt}(t) \\ \mathrm{Dyt}(t) \end{pmatrix}$$

Check Differential Index of System

```
if ~isLowIndexDAE(eqns, vars)
    disp("Reducing Differential Index...")
    % Reduce Differential Index with reduceDAEIndex
    [DAEs, DAEvars] = reduceDAEIndex(eqns, vars)
    % Eliminate redundant equations and variables
    [DAEs, DAEvars] = reduceRedundancies(DAEs, DAEvars)
    % Check the differential index of the new system
    assert(isLowIndexDAE(DAEs, DAEvars))
else
    DAEs = eqns;
    DAEvars = vars;
end
```

```
Reducing Differential Index...
DAEs =
```

$$\begin{pmatrix} m\,\mathrm{Dxtt}(t) - \frac{T(t)\,x(t)}{r} \\ g\,m + m\,\mathrm{Dytt}(t) - \frac{T(t)\,y(t)}{r} \\ -r^2 + x(t)^2 + y(t)^2 \\ \mathrm{Dxt}(t) - \mathrm{Dxt}_7(t) \\ \mathrm{Dyt}(t) - \mathrm{Dyt}_7(t) \\ 2\,\mathrm{Dxt}_7(t)\,x(t) + 2\,\mathrm{Dyt}_7(t)\,y(t) \\ 2\,y(t)\,\frac{\partial}{\partial t}\,\mathrm{Dyt}_7(t) + 2\,\mathrm{Dxt}_7(t)^2 + 2\,\mathrm{Dyt}_7(t)^2 + 2\,\mathrm{Dxt7t}(t)\,x(t) \\ \mathrm{Dxtt}(t) - \mathrm{Dxt7t}(t) \\ \mathrm{Dytt}(t) - \frac{\partial}{\partial t}\,\mathrm{Dyt}_7(t) \\ \mathrm{Dyt}_7(t) - \frac{\partial}{\partial t}\,y(t) \end{pmatrix}$$

```
DAEvars =
```

$$\begin{pmatrix} x(t) \\ y(t) \\ T(t) \\ \mathrm{Dxt}(t) \\ \mathrm{Dyt}(t) \\ \mathrm{Dytt}(t) \\ \mathrm{Dxtt}(t) \\ \mathrm{Dxt}_7(t) \\ \mathrm{Dyt}_7(t) \\ \mathrm{Dxt7t}(t) \end{pmatrix}$$

DAEs =

$$\begin{pmatrix} -\frac{T(t)\,x(t) - m\,r\,\mathrm{Dxtt}(t)}{r} \\ \frac{g\,m\,r - T(t)\,y(t) + m\,r\,\mathrm{Dytt}(t)}{r} \\ -r^2 + x(t)^2 + y(t)^2 \\ 2\,\mathrm{Dxt}(t)\,x(t) + 2\,\mathrm{Dyt}(t)\,y(t) \\ 2\,\mathrm{Dxt}(t)^2 + 2\,\mathrm{Dyt}(t)^2 + 2\,\mathrm{Dxtt}(t)\,x(t) + 2\,\mathrm{Dytt}(t)\,y(t) \\ \mathrm{Dytt}(t) - \frac{\partial}{\partial t}\,\mathrm{Dyt}(t) \\ \mathrm{Dyt}(t) - \frac{\partial}{\partial t}\,y(t) \end{pmatrix}$$

DAEvars =

$$\begin{pmatrix} x(t) \\ y(t) \\ T(t) \\ \mathrm{Dxt}(t) \\ \mathrm{Dyt}(t) \\ \mathrm{Dytt}(t) \\ \mathrm{Dxtt}(t) \end{pmatrix}$$

Convert DAE system to MATLAB function

```
pDAEs = symvar(DAEs);
pDAEvars = symvar(DAEvars);
extraParams = setdiff(pDAEs, pDAEvars)
```

extraParams = $\begin{pmatrix} g & m & r \end{pmatrix}$

Create the function handle

```
f = daeFunction(DAEs, DAEvars, g, m, r);
```

Set parameter values

```
g = 9.81;
m = 1;
```

```
r = 1;
```

Create function for ode15i

```
F = @(t, Y, YP) f(t, Y, YP, g, m, r);
```

Find initial condition

```
DAEvars
```

DAEvars =

$$\begin{pmatrix} x(t) \\ y(t) \\ T(t) \\ \mathrm{Dxt}(t) \\ \mathrm{Dyt}(t) \\ \mathrm{Dytt}(t) \\ \mathrm{Dxtt}(t) \end{pmatrix}$$

Note that `Dxt(t)`, `Dthetat(t)`, ... etc. are the first derivatives of `x(t)` ... etc.

Provide an estimate of the initial condition

```
% Variables
x0est = [r*sin(pi/6); -r*cos(pi/6); 0; 0; 0; 0; 0];
% Their derivatives
xp0est = zeros(7, 1);
```

Set tolerances and do numerical search.

```
opt = odeset('RelTol', 10.0^(-7), 'AbsTol', 10.0^(-7));
[x0, xp0] = decic(F, 0, x0est, [], xp0est, [], opt)
```

```
x0 = 7×1
    0.4771
   -0.8788
   -8.6214
         0
    0.0000
   -2.2333
   -4.1135
xp0 = 7×1
         0
    0.0000
         0
         0
   -2.2333
         0
         0
```

Solve DAEs Using ode15i

```
[tSol, xSol] = ode15i(F, [0 0.5], x0, xp0, opt);
```

Plot solution

```
plot(tSol, xSol(:, 1:origVars), 'LineWidth', 2)
xlabel("Time ($t$)", 'Interpreter', 'latex')
labels = arrayfun(@(i) char(DAEvars(i)), 1:origVars, 'UniformOutput', false);
legend(labels, 'Location', 'Best', 'Interpreter', 'latex')
grid on
```