Contents

Called Functions

Given Values

The following assigns values given by the problem statement to variables. (The value of the gravitational constant is assumed).

```
% Given Values
w = 15;                         % Weight of bowling ball (lb)
d = 8.5;                        % Diameter of bowling ball (in)
mu_k = 0.12;                    % Kinetic friction coefficient (unitless)
mu_s = 0.14;                    % Static friction coefficient (unitless)
g = 32.174;                     % Gravitational constant (ft/s^2)

% Solved Values
m = w/g;                        % Mass of bowling ball (slug)
r = (d/2)/12;                   % Radius of bowling ball (ft)
I = 2/5*m*r^2;                  % Moment of inertia (slug*ft^2)
```

Solve for the Friction Force

To find the friction force, the following first checks whether the bowling ball is slipping by evaluating the velocity of the contact point. A contact point velocity of 0 corresponds to a ball that rolls without slip; otherwise, the ball is slipping. A deadband is used to avoid chatter. See the attached file for hand calculations.

```
% Easy access to variables
v_x = x(1);                     % Easy access to v_x (ft/s)
v_y = x(2);                     % Easy access to v_y (ft/s)
omega_x = x(3);                 % Easy access to omega_x (rad/s)
omega_y = x(4);                 % Easy access to omega_y (rad/s)

% Velocity at contact point
vc_x = v_x - omega_y*r;         % Velocity at contact point[x] (ft/s)
vc_y = v_y + omega_x*r;         % Velocity at contact point[y] (ft/s)

% If no slip, set friction force to 0
deadband = 0.02;
if abs(vc_x) < deadband && abs(vc_y) < deadband

    F_x = 0;                        % Friction force[x] (lb)
    F_y = 0;                        % Friction force[x] (lb)

% Else, set friction force to kinetic friction
else

    N = w;                          % Normal force (lb)

    theta = atan2(vc_y, vc_x);      % Theta (rad)

    F_x = -mu_k*N*cos(theta);       % Friction force[x] (lb)
```

```
    F_y = -mu_k*N*sin(theta);       % Friction force[y] (lb)

end
```

```
Not enough input arguments.

Error in BowlingBallEOM (line 32)
v_x = x(1);                     % Easy access to v_x (ft/s)
```

Solve for xdot

See the attached file for hand calculations.

```
xdot = [F_x/m;
        F_y/m;
        F_y*r/I;
        -F_x*r/I;
        v_x;
        v_y];
```

```
end
```

Published with MATLAB® R2017a