MuJoCo: Hybrid Systems (1)

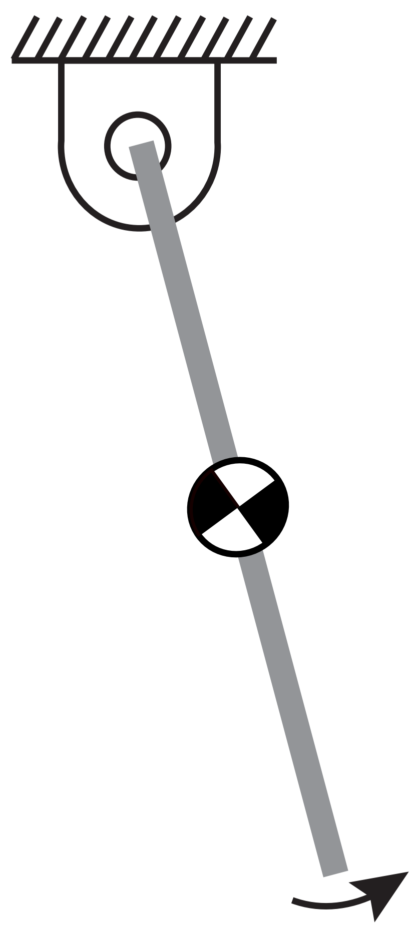
Hybrid Systems: Systems with continuous and discrete dynamics/modes

Examples

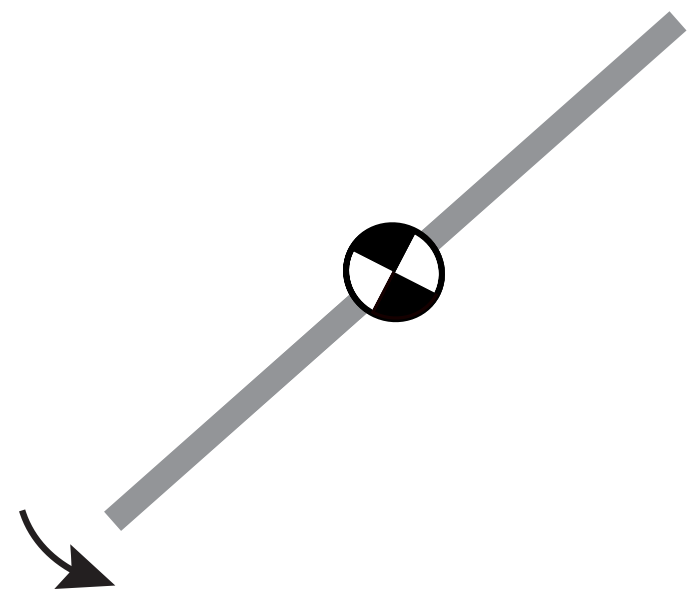
1. Bouncing Ball
2. Juggling
3. Locomotion (walking, running, trot, bound, …)
4. Manipulating an object.

MuJoCo: Hybrid Systems (2)

Horizontal bar: Pendulum swing + Free fall



Pendulum swing



Free Fall

MuJoCo: Hybrid Systems (3)

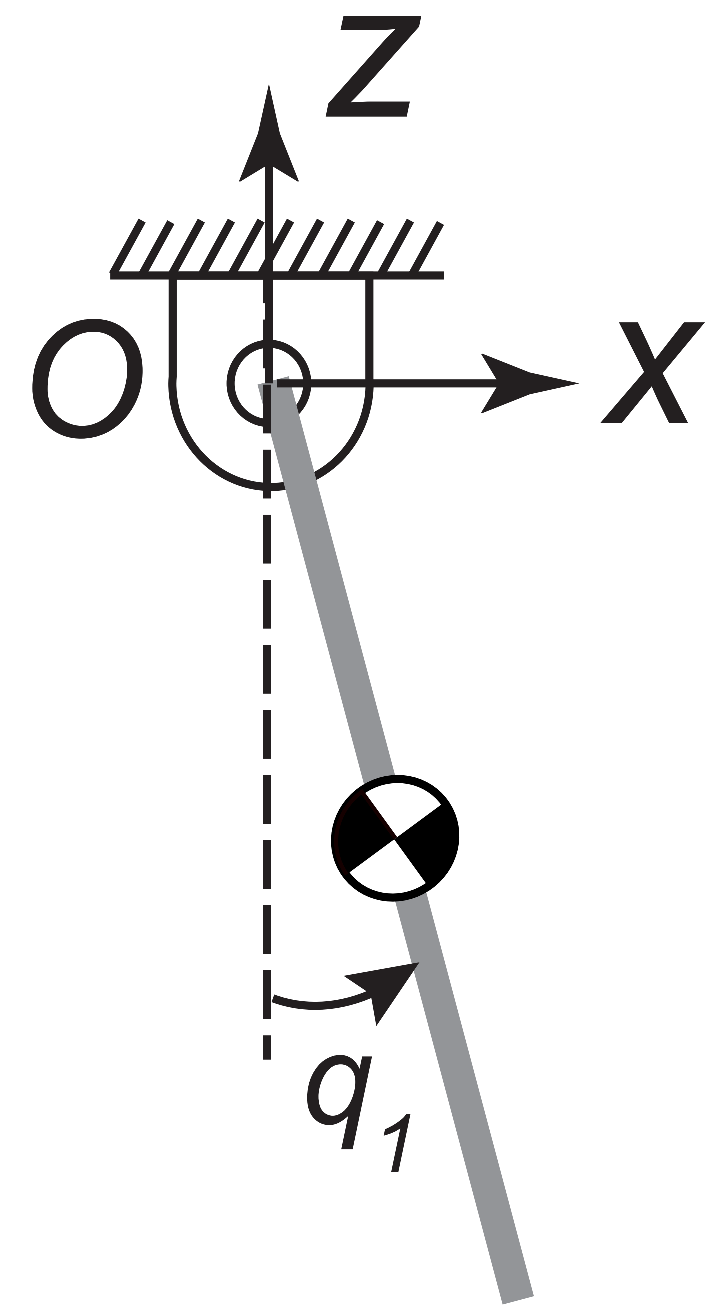
Using template_pendulum.zip to get started

1. From tiny.cc/mujoco download template_pendulum.zip and unzip in myproject
2. Rename folder template_pendulum to hybrid_pendulum
3. Make these three changes
    1. main.c — line 28, change template_pendulum/ to hybrid_pendulum/
    2. makefile — change ROOT = template_writeData to ROOT = hybrid_pendulum also UNCOMMENT (del #) appropriate to your OS
    3. run_unix / run_win.bat change <template_pendulum> to <hybrid_pendulum>
4. In the *shell, navigate to hybrid_pendulum and type ./run_unix (unix) or run_win (windows); *shell = terminal for mac/linux / x64 for win

MuJoCo: Pendulum with floating base (1)

pendulum.xml

- pendulum with three joints: x, z, q
- Enforce x=z=0 using <equality> in xml



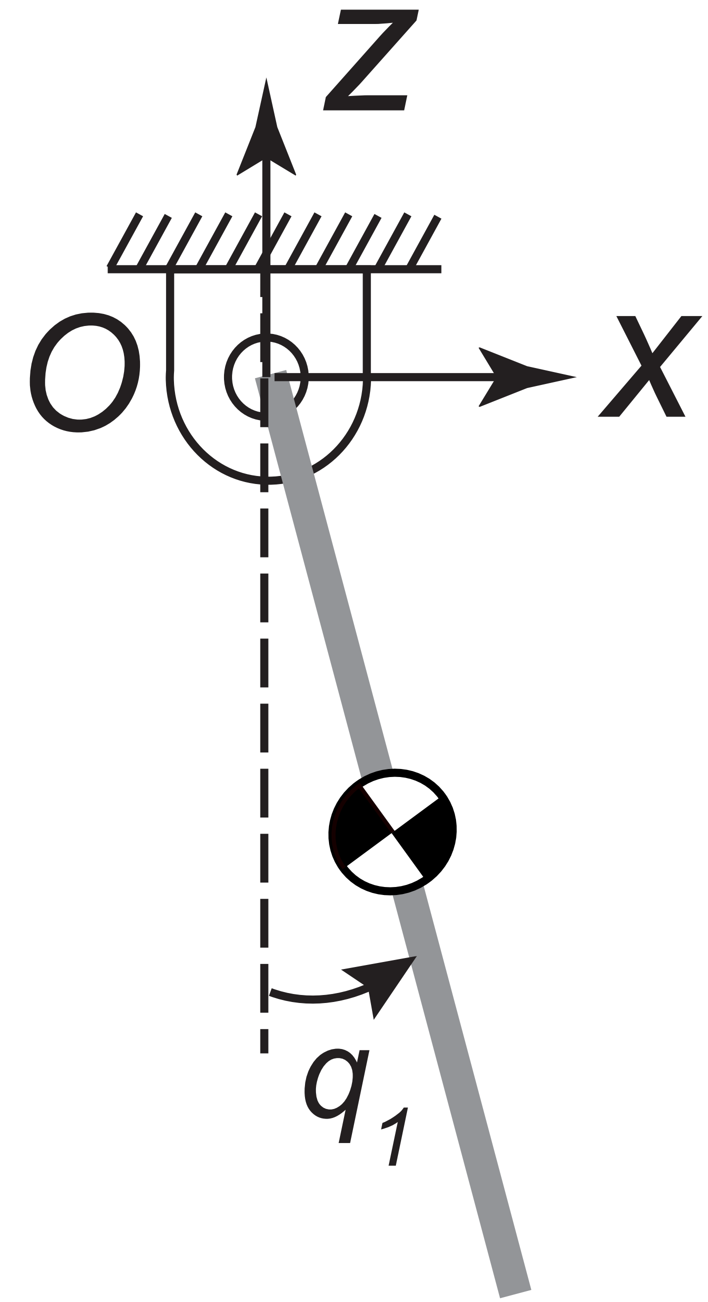
MuJoCo: Pendulum with floating base (2)

Equations of floating pendulum

$$M\ddot{q} + f = \tau$$

Expanded out

$$\begin{bmatrix} M[0] & M[1] & M[2] \\ M[3] & M[4] & M[5] \\ M[6] & M[7] & M[8] \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{z} \\ \ddot{q}_1 \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} F_x \\ F_z \\ \tau_y \end{bmatrix}$$

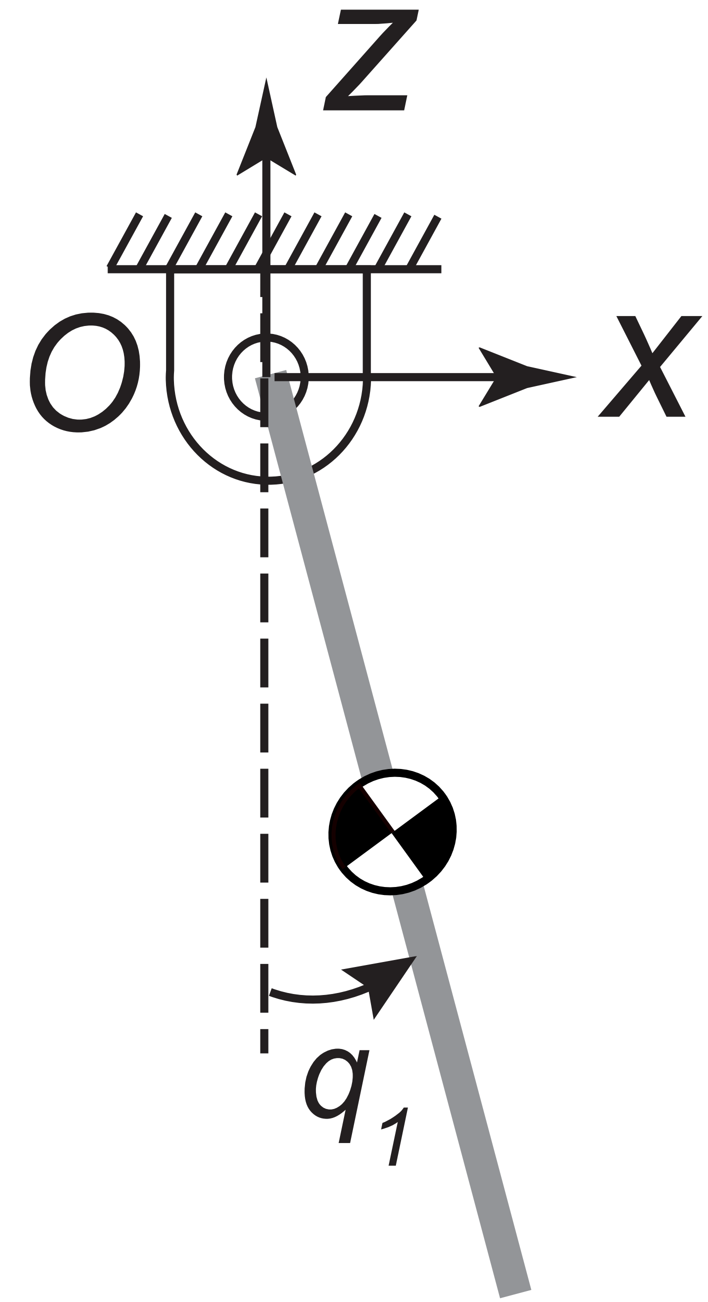


MuJoCo: Pendulum with floating base (3)

Equations of constrained pendulum

$$M\ddot{q} + f = \tau + J_O^T F_0$$

F_0 are the constraint forces at O



Expanded out

$$\begin{bmatrix} M[0] & M[1] & M[2] \\ M[3] & M[4] & M[5] \\ M[6] & M[7] & M[8] \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{z} \\ \ddot{q}_1 \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \ldots$$

$$\begin{bmatrix} F_x \\ F_z \\ \tau_y \end{bmatrix} + \begin{bmatrix} J[0] & J[1] & J[2] \\ J[3] & J[4] & J[5] \\ J[6] & J[7] & J[8] \end{bmatrix}^T \begin{bmatrix} F_0[0] \\ F_0[1] \\ F_0[2] \end{bmatrix}$$

Let us verify these equations in MuJoCo

MuJoCo: Pendulum with floating base (4)

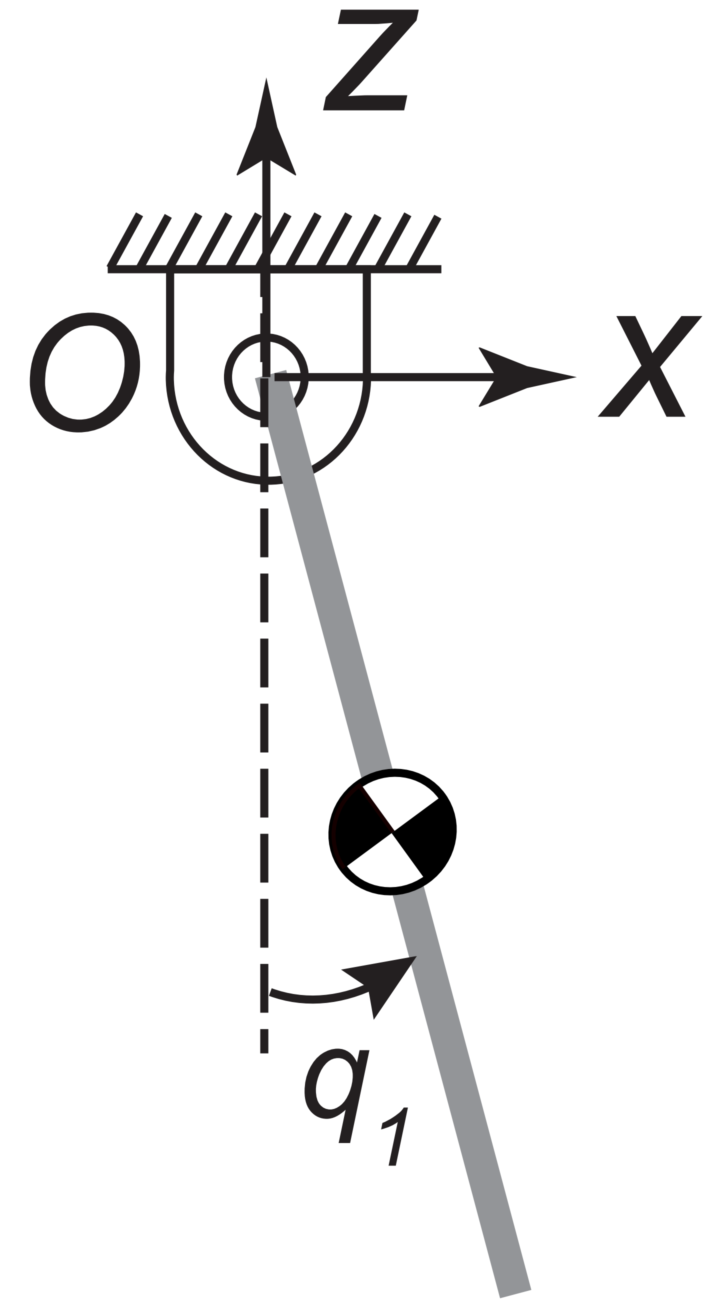
Equations of constrained pendulum

$$M\ddot{q} + f = \tau + J_O^T F_0$$

3 equations

6 unknowns (xddot, zddot, q1ddot, F0x, F0y,F0z)

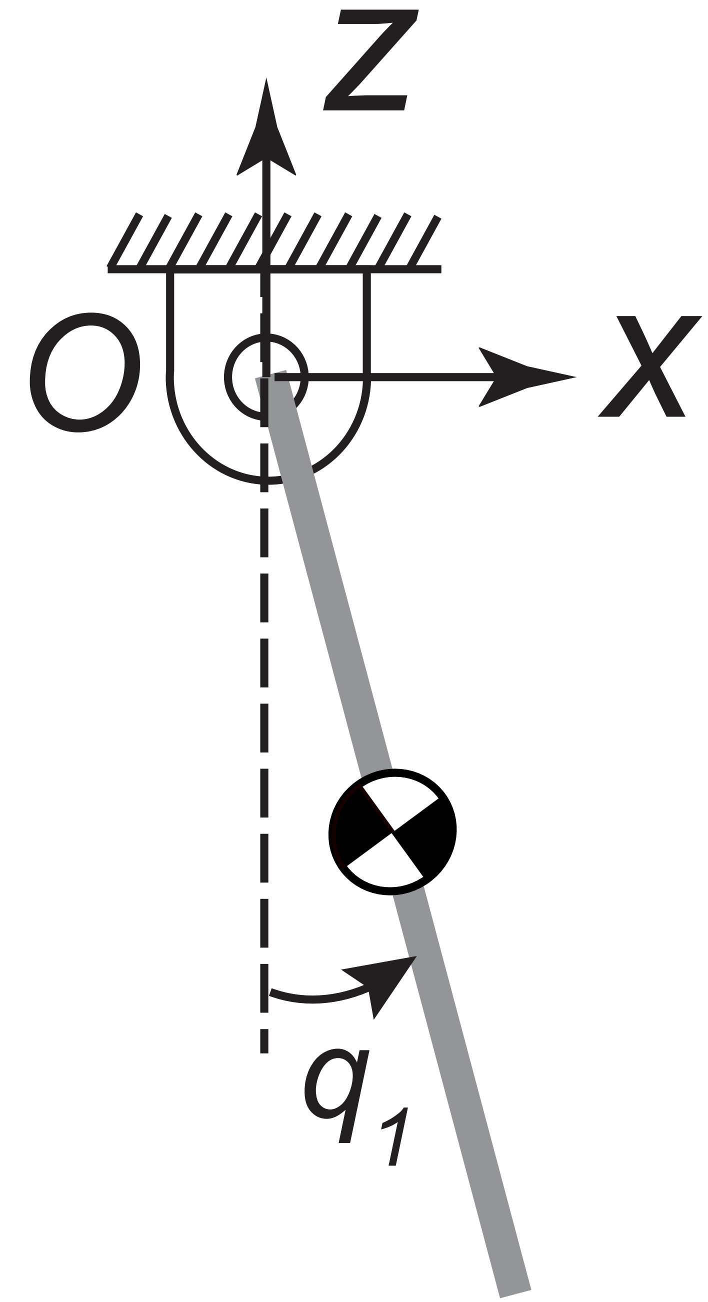
How to compute F0?



MuJoCo: Pendulum with floating base (5)

There are 3 more equations

$$\ddot{x}_O = 0 = J_0 \ddot{q} + \dot{J}_0 \dot{q}$$



All equations

$$M\ddot{q} - J_O^T F_0 = \tau - f$$
$$J_0 \ddot{q} = -\dot{J}_0 \dot{q}$$

All equations, written more compactly

$$\begin{bmatrix} M & -J_0^T \\ J_0 & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ F_0 \end{bmatrix} = \begin{bmatrix} \tau - f \\ -\dot{J}\dot{q} \end{bmatrix}$$

6 equations, 6 unknowns —> Solve for qddot, F0 at each time step

MuJoCo: Pendulum with floating base (6)

Trick: Use MuJoCo to do the computations.

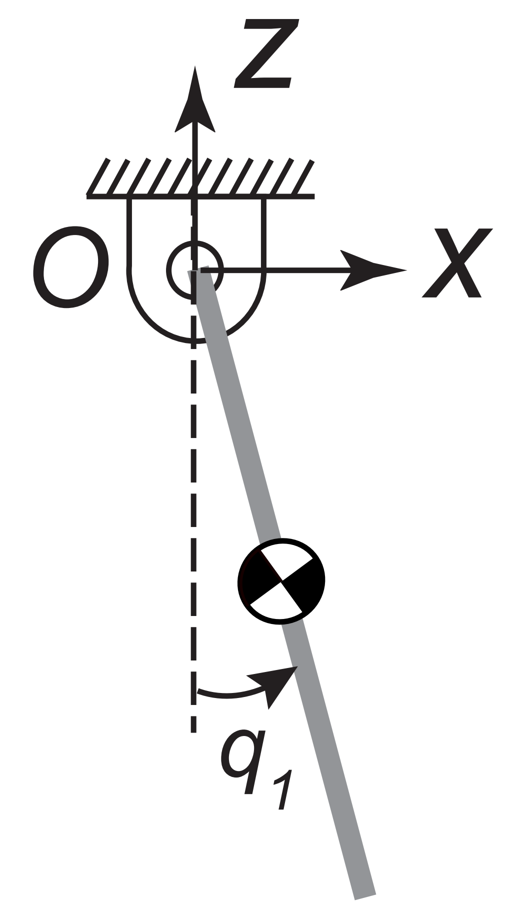
1) Pendulum 1: Has equality constraints

2) Pendulum 1: Obtain J_0 and F_0

3) Pendulum 2: No equality constraints

4) Pendulum 2: Add forces at 0; J^T_0 F_0

5) (optional) only display Pendulum 2

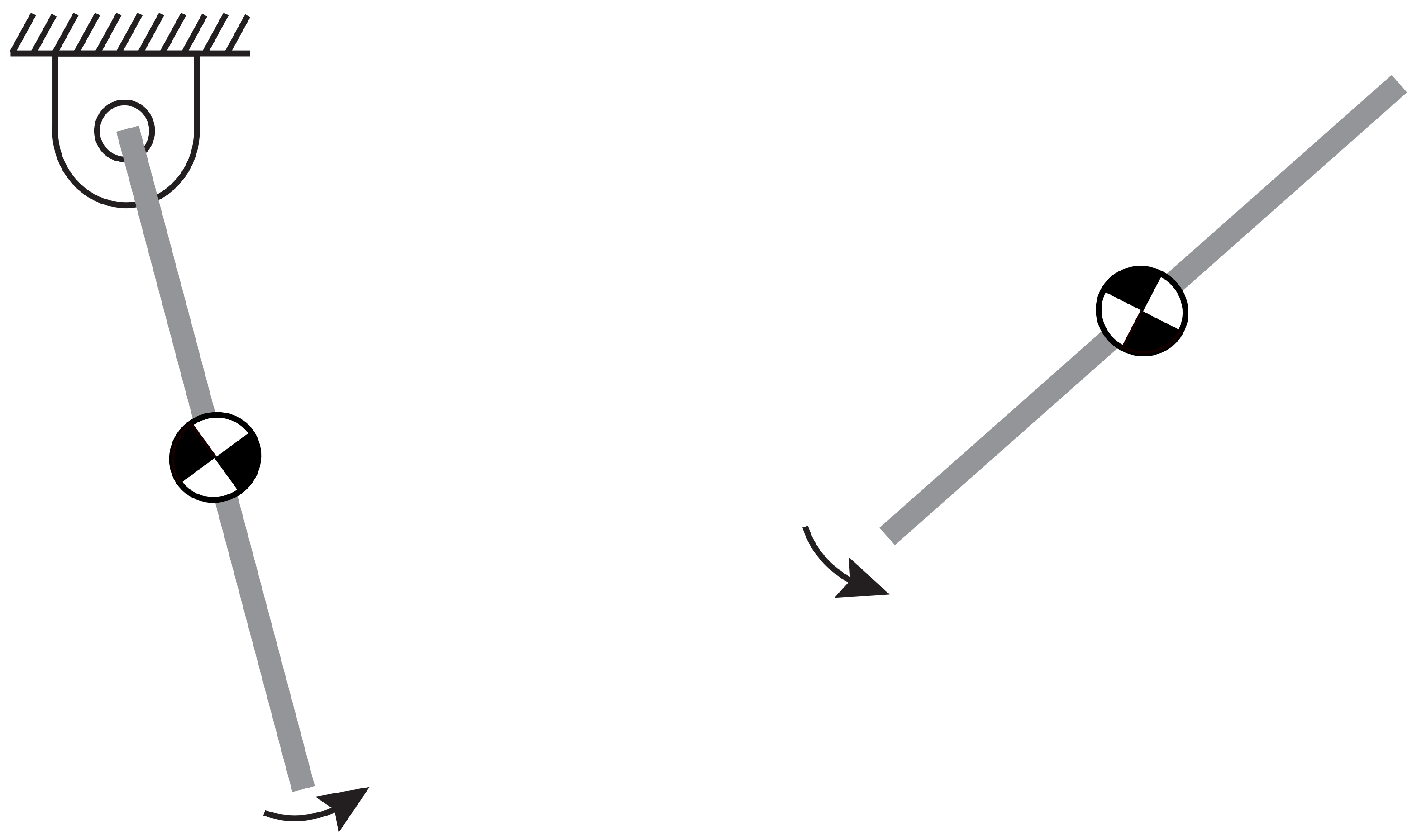


$$M\ddot{q} + f = \tau + J_O^T F_0$$

Lets do this.

MuJoCo: Hybrid Systems (3)

Horizontal bar: Pendulum swing + Free fall



Lets use a finite state machine to program this logic

MuJoCo: Hybrid Systems (4)

- Summary of functions learnt
  - Enforce constraints (in xml): equality
  - Jacobian of constraints: efc_J
  - Constraint force: efc_force