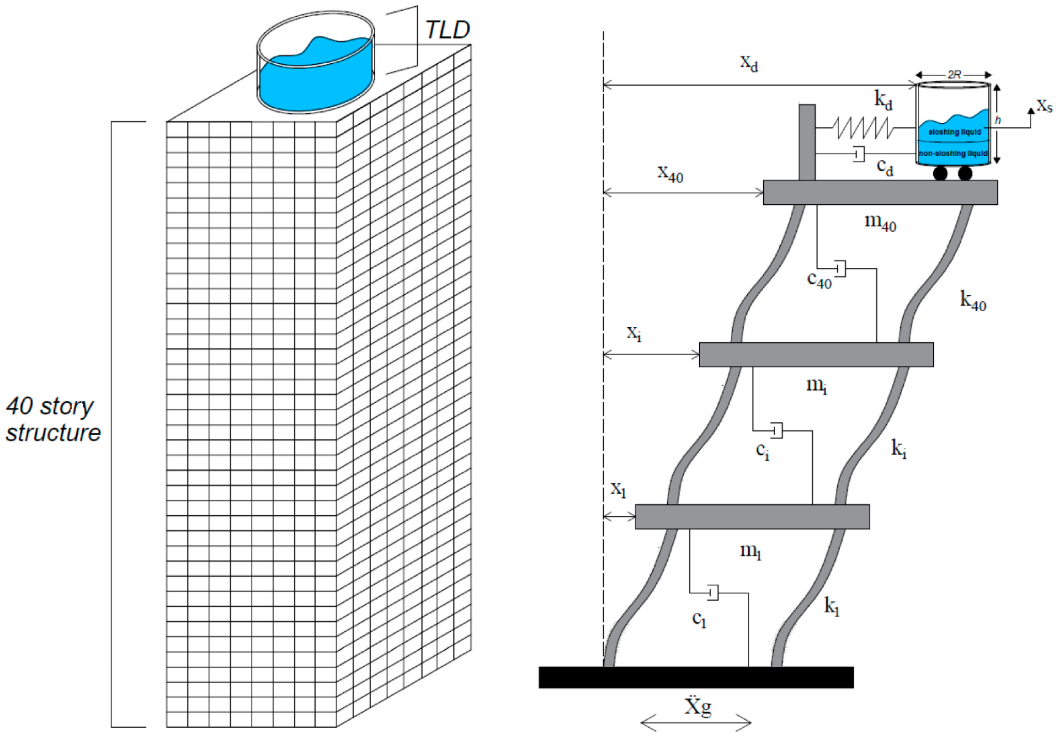
Final Presentation Planning

Requirements:

- From lecture, project needs to have some element of uniqueness - can be based on an existing example but needs to have some kind of “creative wrinkle”

Ideas:

- Balancing inverted pendulum
    - Would probably need to add some twist, bc PyDy already has this example
        - Conical pendulum, instead of revolute?
        - Reaction-wheel pendulum?
        - Regular single-link pendulum, but with swing-up method (like this)
    - Demo-wise, I think this idea is a good candidate for an interesting live demo
    - Practicality-wise, I think the single-link pendulum with swing-up would be the easiest to do, but the biggest effort would be designing the controller that does the swing-up process. Modeling the EoMs and simulating the pendulum motion has already been done in examples.
- Track/field simulation
    - Obtain data about the physical builds of various track/field athletes (leg/arm/torso length) and try to create a model that will show if certain builds have advantages in certain events (long jump/high jump/dash)
    - Ex: For long jump something along the lines of which combo of reasonable measurements generates the longest jump. Set base “muscle mass” to relate length to mass and use PyDy to simulate jumps maybe.
    - Cons: Not sure if it would be too difficult, or if data would be hard to come by. Also might have to integrate more packages than just PyDy to do anything useful with the data used/collected.
    - Pros: I think this would be interesting/fun to give a presentation on and people would like it. Also there’s room for relatively easy rescoping under this general idea. If we find it too easy/difficult we can reduce/increase the sample size of athletes or breadth of events covered.
- Three story building model (earthquake)

- Multiple mass spring damper systems on top of each other. Derive symbolic equations of motion for a building model. We can calculate if a certain story's velocity or deflection is too large and determine whether an earthquake of certain magnitude will cause a building to collapse (it would depend on criteria that we set).
- I think this would be a really useful example for this idea
- 

- Tricycle model
  - Might be tough to get approved in terms of uniqueness, bc PyDy already has bicycle model
- Sliding pendulum
  - Model a pendulum's motion when oscillating as well as on a sliding, oscillating base. How does the sliding base affect the pendulum's position and motion?
- Dynamic analysis of car engine (single cylinder internal combustion mechanism)
  - A single cylinder internal combustion mechanism is a fourbar slider-crank mechanism

Project objective:

The objective of this project is to do a dynamic analysis of a basic 4-stroke engine model composed of 4 piston assemblies. This will involve some force inputs to the pistons, and speed and torque outputs of the system.

Project approach:

- Derive the equations of motion for a single piston assembly - **Van**
- Define a forcing function that applies force to the piston assembly based on the 4-stroke cycle - **Bhargav**
    - Need to look at examples for how exactly force is applied to the system
    - I think compression and combustion strokes are of particular importance here
    - Function should change based on parameter inputs (geometric, chemical, etc.)
    - Function will be used for simulation of one piston assembly
- Calculate the torque and angular velocity of the crankshaft - **Nick**
    - Should probably be presented as 2 plots
        i. Torque vs crankshaft angle / time
        ii. Angular velocity vs crankshaft angle / time
    - I think these can just be static plots, not necessarily live (e.g. synchronized w/ animation; maybe could be a nice-to-have)
- Animate motion of single piston assembly - **Christian**
- Nice-to-have:
    - Live torque and angular velocity plots (synchronized with animation)
    - Allow modularity by making # of pistons an input parameter, and analyzing the effects on torque and angular velocity output

Packages to be utilized:

SymPy

- Modeling equations of motion
- Defining forcing function
- Assigning masses and inertias to components of piston assembly

PyDy/matplotlib

- Simulating trajectory of piston assemblies
- Plotting trajectories and animating motion

Similar existing projects:

PyDy 4-bar linkage model

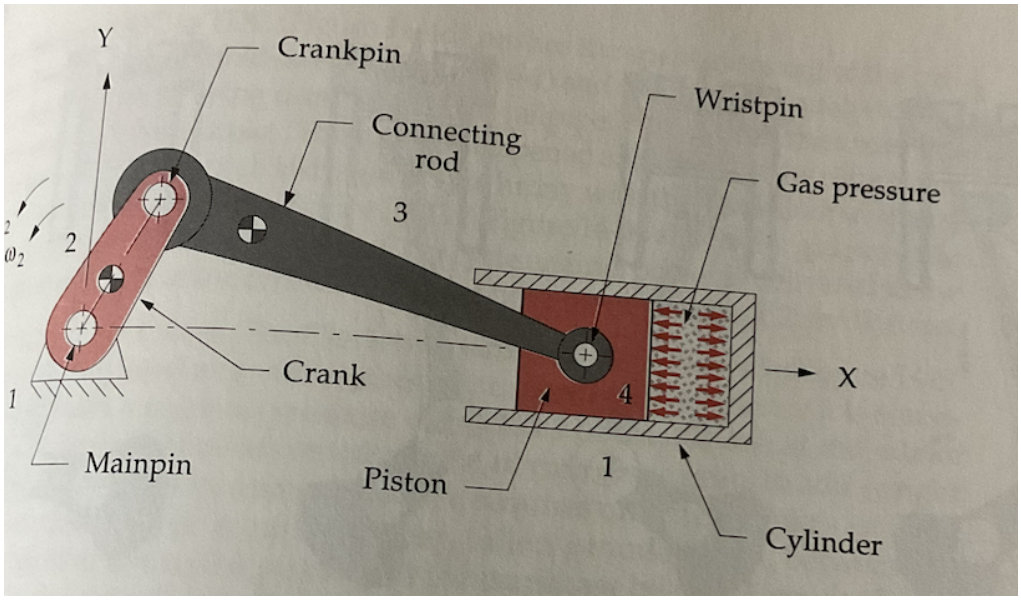
- https://nbviewer.org/github/pydy/pydy/blob/master/examples/four_bar_linkage/four_bar_linkage_with_motion_constrained_link.ipynb

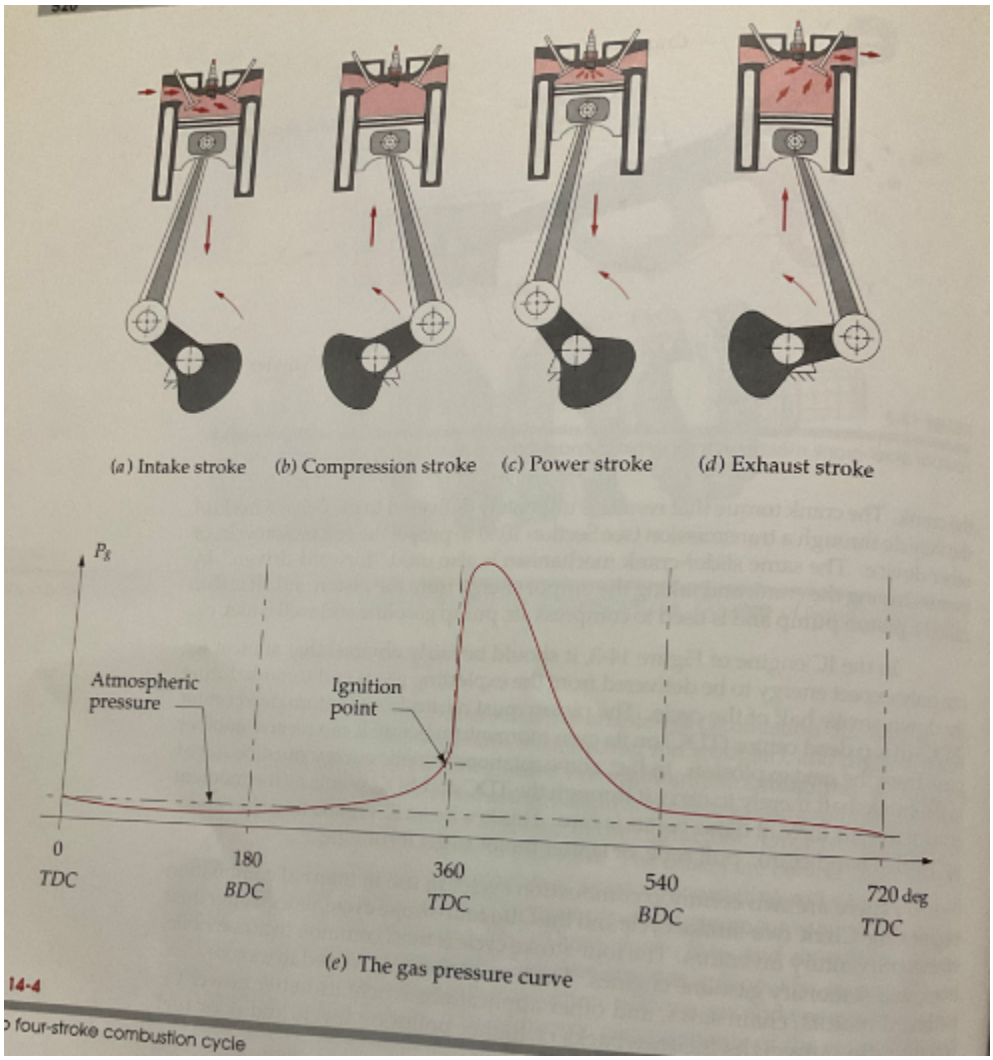
PyDy multi-DoF holonomic system model

- https://pydy.readthedocs.io/en/stable/examples/multidof-holonomic.html

What makes this project unique:

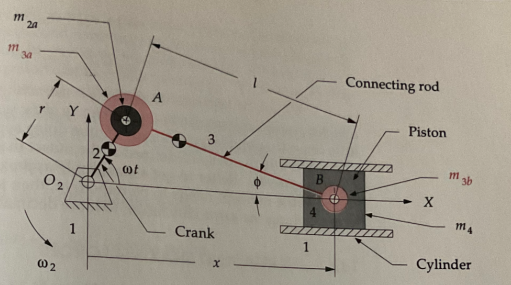
- Goes a step further than the pendulum examples that have already been completed with PyDy; incorporates a forcing function reflective of the 4-stroke cycle
- Models the interaction between pistons during a four stroke cycle and outputs useful information regarding this interaction



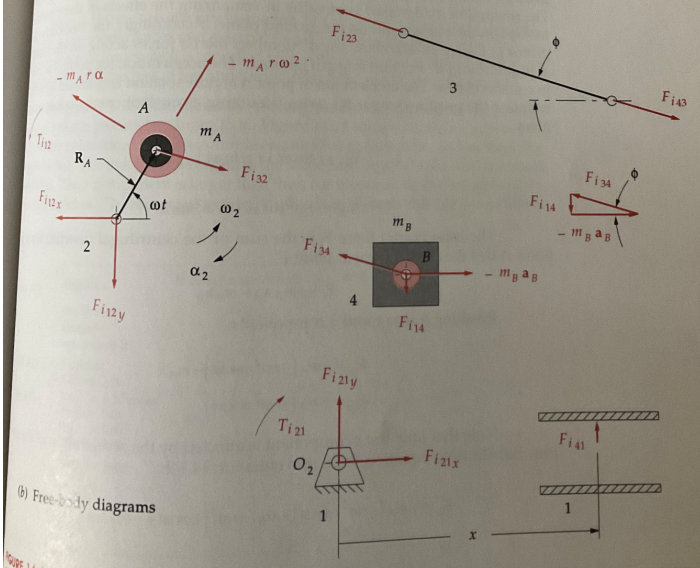


14-4

four-stroke combustion cycle



(a) Dynamic model



(b) Free-body diagrams

FIGURE 14-3
Exploded mass dynamic model of the slider-crank