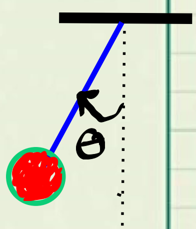
Summary Lagrangian Dynamics



"Lagrange's Method for deriving ODEs for Mechanical Systems"

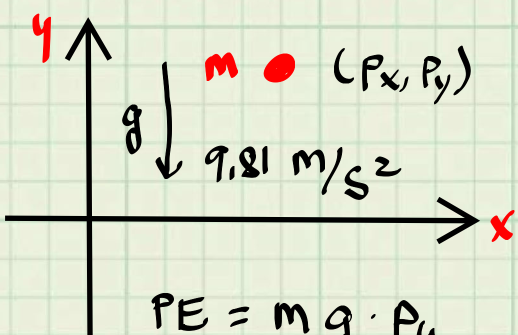
$\ddot{\theta} = -\frac{g}{L}\sin(\theta)$

where $\ddot{\theta} = \frac{d^2\theta(t)}{dt^2}$

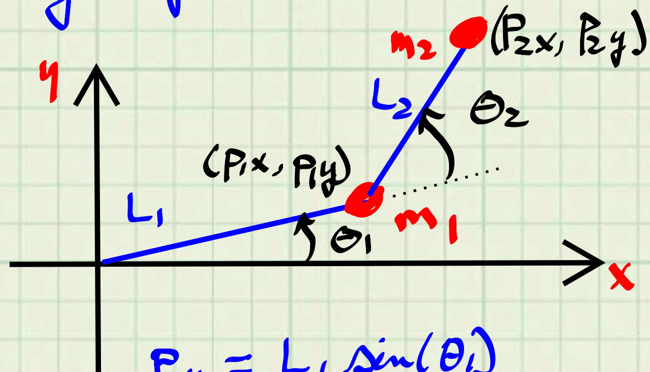
Differential Equation := Equation containing a variable along with one or more derivatives.

Key Parts of Lagrange's Method

- PE := Potential Energy (position)
- KE := Kinetic Energy (motion)
- $\mathcal{L}$:= KE − PE "Lagrangian"



$PE = m g \cdot p_y$



$p_{1y} = L_1 \sin(\theta_1)$

$p_{2y} = p_{1y} + L_2 \sin(\theta_1 + \theta_2)$

$PE = m_1 \cdot g \cdot p_{1y} + m_2\, g\, p_{2y}$

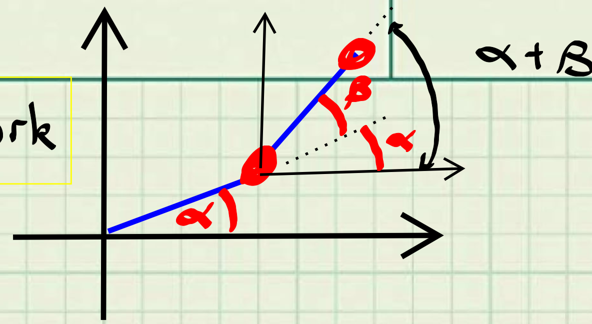
(let Julia do the algebra!)

Not yet revealed:
"Lagrange's Equation"

θ_1 is an absolute angle (measured w.r.t. world frame)

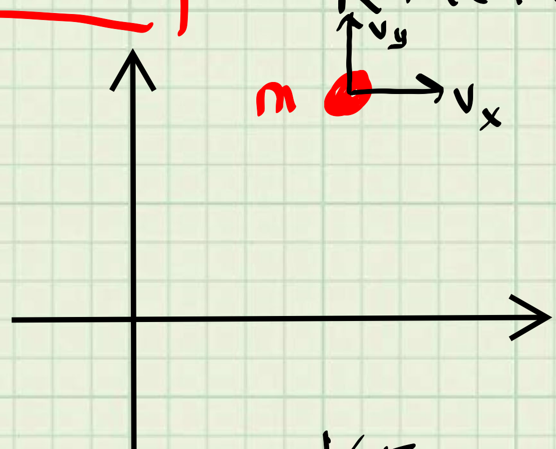
θ_2 is a relative angle (measured w.r.t. previous link)

Video: How Encoders Work



absolute
Needed for trig calc's.

Today: Kinetic Energy



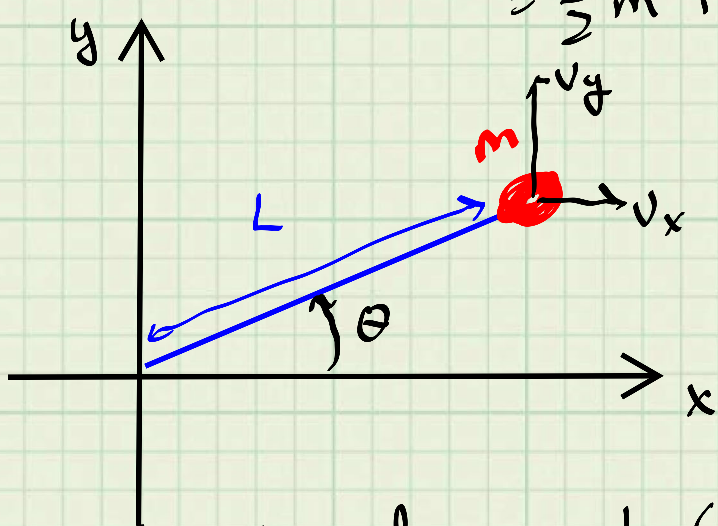
$$v_x = \frac{d}{dt} p_x = \dot{p}_x$$

$$v_y = \frac{d}{dt} p_y = \dot{p}_y$$

$$KE_0 = \frac{1}{2} m \left((v_x)^2 + (v_y)^2 \right)$$

$$= \frac{1}{2} m \; v^T \cdot v \qquad v = \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

$$= \frac{1}{2} m \; \|v\|^2$$



$$p_x = L \cos(\theta)$$

$$p_y = L \sin(\theta)$$

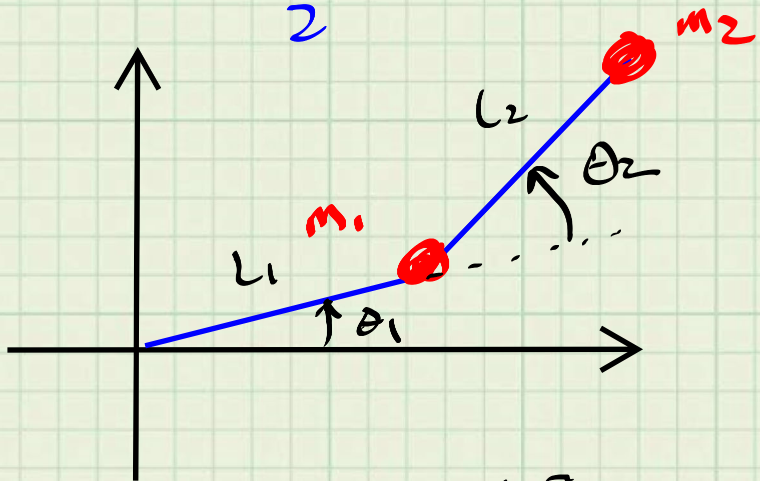
$$v_x = \frac{d}{dt} p_x = L \left(-\sin(\theta)\right) \cdot \dot{\theta}$$

$$v_y = \frac{d}{dt} p_y = L \; \cos(\theta) \cdot \dot{\theta}$$

$$KE = \frac{1}{2} m\left[(v_x)^2 + (v_y)^2\right]$$

$$= \frac{1}{2} m\left[L^2 \sin^2(\theta)\cdot(\dot{\theta})^2 + L^2\cos^2(\theta)(\dot{\theta})^2\right]$$

$$= \frac{m}{2} L^2 (\dot{\theta})^2$$



$$V_{1,x} = \frac{d}{dt}\left[L_1 \cos(\theta_1)\right] = L_1\left[-\sin(\theta_1)\right]\cdot\dot{\theta}_1$$

$$V_{1,y} = \frac{d}{dt}\left[L_1 \sin(\theta_1)\right] = L_1 \cos(\theta_1)\dot{\theta}_1$$

$$V_{2,x} = \frac{d}{dt}\left[L_1 \cos(\theta_1) + L_2 \cos(\theta_1+\theta_2)\right]$$

$$= L_1\left[-\sin(\theta_1)\right]\dot{\theta}_1 + L_2\left[-\sin(\theta_1+\theta_2)\right]\cdot(\dot{\theta}_1+\dot{\theta}_2)$$

$V_{2,y}$ = similar (Will do in code shortly)

$$KE = \frac{1}{2} m_1 \|v_1\|^2 + \frac{1}{2} m_2 \|v_2\|^2$$

Leads to messy algebra

$\mathcal{L} := KE - PE$

How do we get the "equations of motion" (aka, the ODEs) from the Lagrangian?

$$\boxed{\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q} = \Gamma}$$

Lagrange's Eqn. comes from the "Calculus of Variations"

q = vector of position coordinates

e.g. $q = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ for the 2-link arm

$\dot{q}$ = vector of velocity coordinates,

e.g. $\dot{q} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$ for the 2-link arm

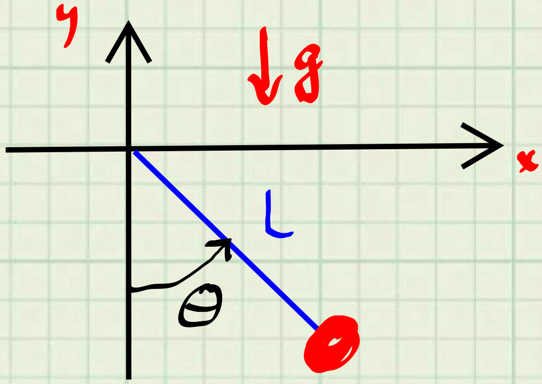
Γ = vector of external forces, e.g., motor torques, joint friction, air resistance

Fact: Robot Equations

Lagrange's Equations can always be written as

$$\underbrace{D(q)}_{\substack{n\times n \text{ mass} \\ \text{inertia matrix}}}\ddot{q} + \underbrace{C(q,\dot{q})}_{\substack{n\times n \text{ Coriolis} \\ \text{matrix}}}\cdot\dot{q} + \underbrace{G(q)}_{\substack{n\times 1 \\ \text{Gravity} \\ \text{vector}}} = \overset{\substack{n\times 1 \text{ external} \\ \text{forces}}}{\Gamma}$$

"ma = F on steroids"



$$P = \begin{bmatrix} P_x \\ P_y \end{bmatrix} = \begin{bmatrix} L\sin(\theta) \\ -L\cos(\theta) \end{bmatrix}$$

$$PE = -mgL\cos(\theta)$$

$$V = \begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} L\cos(\theta)\cdot\dot{\theta} \\ L\sin(\theta)\dot{\theta} \end{bmatrix} \leftarrow (-)(-) = +$$

$$KE = \frac{1}{2}mL^2(\dot{\theta})^2$$

$$q = [\theta]$$

$$\dot{q} = [\dot{\theta}]$$

$\Gamma = 0$ because we will assume no friction and no motor torque.

$$\mathcal{L} = KE - PE = \frac{1}{2} m L^2 (\dot{\theta})^2 \underbrace{-(mg)(-L)}_{+} \cos(\theta)$$

$$= \frac{1}{2} m L^2 (\dot{\theta})^2 + mgL\cos(\theta)$$

$$\frac{\partial \mathcal{L}}{\partial \dot{q}} = \frac{\partial \mathcal{L}}{\partial \dot{\theta}} = m L^2 \dot{\theta}$$

$$\frac{\partial \mathcal{L}}{\partial q} = \frac{\partial \mathcal{L}}{\partial \theta} = -mgL\sin(\theta)$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}}\right) = \frac{d}{dt}\left[m L^2 \dot{\theta}\right] = m L^2 \ddot{\theta}$$

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q} = \Gamma = 0_{1\times 1}$$

$$\cancel{m} L^{\cancel{2}} \ddot{\theta} \underbrace{- [-}_{+} \cancel{m} g \cancel{L} \sin(\theta)] = 0$$

$$L\ddot{\theta} + g\sin(\theta) = 0$$

$$\boxed{\ddot{\theta} = -\frac{g}{L}\sin(\theta)}$$

Equation of motion for a pendulum

"Integrating the Robot Equations"

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = 0_{n\times 1}$$

```julia
for i in 2:nRow
    # Compute the velocity trajectory
    vMat[i, :] = vMat[i-1, :] + 0.5*(aMat[i-1, :] + aMat[i, :])*dt

    # Compute the position trajectory
    pMat[i, :] = pMat[i-1, :] + 0.5*(vMat[i-1, :] + vMat[i, :])*dt
end
```

(turn into Riemann sum)

$$D(q)\ddot{q} = -C(q,\dot{q})\dot{q} - G(q)$$

$$\ddot{q} = D(q) \backslash \left[-C(q,\dot{q})\dot{q} - G(q)\right]$$

angular acceleration

insert acceleration calculation here for equivalent of aMat[i-1,:]

see DEMO.

```julia
for i in 2:nRow
    # Compute the velocity trajectory
    vMat[i, :] = vMat[i-1, :] + 0.5*(aMat[i-1, :] + aMat[i, :])*dt

    # Compute the position trajectory
    pMat[i, :] = pMat[i-1, :] + 0.5*(vMat[i-1, :] + vMat[i, :])*dt
end
```

not available

Problem Setup

We consider the robot equations without input, namely

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = 0_{n\times 1},$$

where, as before,

- q typically represents joint angles,
- $\dot{q}$ typically represents angular velocities, and
- $\ddot{q}$ represents the angular accelerations.

Moreover,

- $D(q)$ is the $n \times n$ **inertia matrix**,
- $C(q,\dot{q})$ represents the $n \times n$ **Coriolis and centrifugal effects**,
- $G(q)$ is the $n \times 1$ **gravity vector**.

To be extra clear, we are not including motor torques at the joints.

Angular Acceleration

For later use, the angular acceleration can be solved for and used later to update the velocity term.

$$\ddot{q} = D(q) \backslash \left(-C(q,\dot{q})\dot{q} - G(q)\right).$$

Note that we always choose to avoid matrix inverses whenever possible.

Numerical Integration Approach

We solve this system over a finite time interval T using a **discrete time step**:

$$t = \{0, \Delta t, 2\Delta t, \ldots, T\}$$

where $\Delta t = \frac{T}{N}$ is the time step, and N is the total number of steps.
The numerical integration proceeds as follows:

1. **Notation for the code**
   - $q[:,i]$ is q (angular position) at time $t[i]$.
   - $dq[:,i]$ is $\dot{q}$ (angular velocity) at time $t[i]$.
   - $ddq[:,i]$ is $\ddot{q}$ (angular acceleration) at time $t[i]$.
2. **Compute angular acceleration** $\ddot{q}$ at the current time step using:

$$ddq[:,i] = D(q[:,i]) \backslash (-C(q[:,i], dq[:,i]) \cdot dq[:,i] - G(q[:,i$$

3. **Update velocity** $\dot{q}$ using a (left) **Riemann Sum**:

$$dq[:,i+1] = dq[:,i] + \Delta t \cdot ddq[:,i]$$

4. **Update position** q using the **Trapezoidal Rule**:

$$q[:,i+1] = q[:,i] + \Delta t \cdot \left(\frac{dq[:,i] + dq[:,i+1]}{2}\right)$$

End of Chapter 6

Chapter 7: Antiderivatives and the Fundamental Thms of Calculus.

Def. $F(x)$ is an <u>antiderivative</u> of $f(x)$ if $F'(x) = f(x)$.

Elementary Functions: Composing, multiplying, raising to powers or roots, the standard monomials, trig functions, exponentials, and logs.

Too many students are taught that Integration and finding Antiderivatives are the same thing. This is FALSE and leads some students to hate integration.

Chapter 7:

- Full of intellectually fun material.
- You will rarely use it as a STEM professional.
- You will need it in UG Courses that depend upon Math 116 until computing antiderivatives fades away as slide rules did!

Secrets of the Arcane 7.39: Elementary Functions and Antiderivatives

In mathematics, when we talk about **elementary functions**, we are referring to functions that are constructed using basic arithmetic operations (addition, subtraction, multiplication, and division) and compositions of,

1. **Polynomials**: Functions like $f(x) = x^2$, $g(x) = 3x^3 - 2x + 1$.
2. **Rational Functions**: Ratios of two polynomials, e.g., $R(x) = \frac{x^2+1}{x-2}$.
3. **Root Functions**: Functions involving square roots, cube roots, etc., like $h(x) = \sqrt{x}$.
4. **Trigonometric Functions**: The standard sine, cosine, tangent, etc., and their inverses.
5. **Exponential and Logarithmic Functions**: Functions like e^x, $\ln(x)$.

These functions are termed "elementary" because they are among the first functions introduced in mathematics education and are foundational in calculus and analysis. They are well-understood and have known properties, such as **the derivative of an elementary function is also elementary**. When you combine elementary functions using operations like addition, multiplication, or composition, you get a wide variety of functions that can model many different phenomena, but they still retain the "elementary" label because they're built from these basic components.

Most elementary functions do not have antiderivatives that are also elementary functions.[a] Even the very common function, e^{-x^2}, used in Radial Basis Functions and Gaussian Probability Distributions, does not have an antiderivative that can be expressed in terms of anything we would recognize. The same is true for the antiderivative of $\cos(x^2)$ in Example 7.4-(d). In these cases, the antiderivative is literally defined through the First Fundamental Theorem of Calculus, that is,

$$\text{the antiderivative of } f(x) \text{ is } \; F(x) = \int_a^x f(t)\,dt + C,$$

which is the integral you were trying to compute in the first place. Fortunately, for functions that arise sufficiently often, numerical analysts seek fast and accurate means to compute the new antiderivative. If you are lucky, your antiderivative may be one of the non-elementary antiderivatives that have been tabulated. Are you ready to roll the dice?

The coup de grâce in all of this: the hand methods for determining antiderivatives allow you to find antiderivatives that are already available in what used to be called **Integral Tables**. You can simply look up your antiderivative. Back in the day, engineering students purchased books with thousands of antiderivatives in them. Today, no one needs to do that. We have ChatGPT+Wolfram, which provides access to more integrals than all of the books we students of yore used to lug around.

Further Reading: **Non-elementary Integrals**; **Proving the Non-existence of Elementary Anti-derivatives for Certain Elementary Functions**; **Impossibility Theorems for Elementary Integration**; How WolframAlpha defines these nonelementary integrals; and **Closure of the Set of Elementary Functions**.

[a] In plain words, seeking an antiderivative that you can express in understandable terms can be a fool's errand.