optimization and learning for control

- this course will focus on the following stochastic optimal control problem:

$$(\text{SOCP}) \quad \underset{u}{\text{minimize}} \ \text{cost}(x, u)$$
$$\text{subject to } x_{t+1} \sim \text{dynamics}(t, x_t, u_t)$$

$x_t \in X$ is the <u>state</u> — we'll consider $X = \mathbb{R}^d$ and $|X| < \infty$ (finite set)

$u_t \in \mathcal{U}$ is the <u>control input</u> / <u>decision variable</u>

$x : [0, T) \to X$
$u : [0, T) \to \mathcal{U}$
"x" or "u" without subscript $t \in [0, T)$ denote entire time history

↳ $x|_{[0,t]}$ is the "restriction" to times $\{0, 1, \dots, t\}$

$a \sim b$ means "a is a random variable drawn from distribution b"

ex: $$\min_{u} \sum_t c(x_t, u_t)$$ — sum of <u>running cost</u>

$$\text{s.t. } x_{t+1} = F(x_t, u_t)$$ — <u>difference equation</u> (DE)

* clearly an optimization problem; where's the learning?

→ in the real world, cost c and dynamics F aren't known exactly — must be "learned" through "experience", either by expert humans or expert algorithms

who is the expert?

human ⟷ algorithm

human

1° start from 1st principles
- Newton's Laws
- Maxwell's Eqns
- Navier-Stokes

2° employ domain expertise
- operating regime
- parameter estimate

3° leverage the model

$$u^+ = u - \gamma \cdot D\,c(u)$$

algorithm

1° start from data
- historical logs
- automated experiments
- simulations

2° employ domain expertise
- feature detection
- neural network architecture

3° leverage the data

$$u^+ \sim u - \gamma \cdot g(u)$$

$$E[g(u)] = D\,c(u)$$

← these just summarize these →

BLACK MAGIC !

(stoch.) grad. desc.

* takeaway: these two "extremes" are closely related

→ neither approach is "right"; best approach bridges both