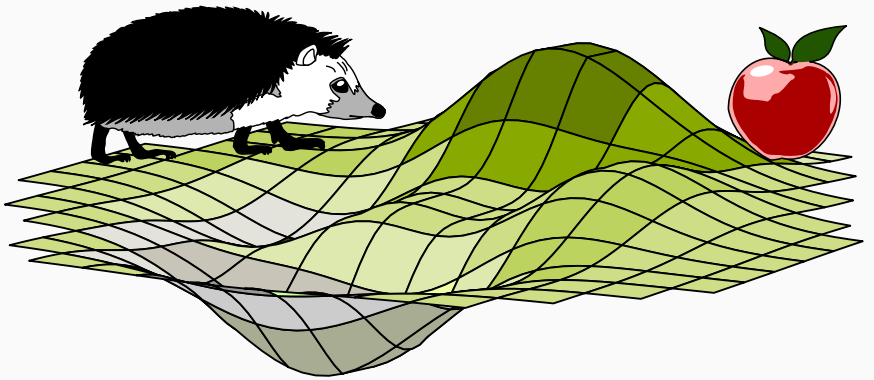
REINFORCEMENT LEARNING

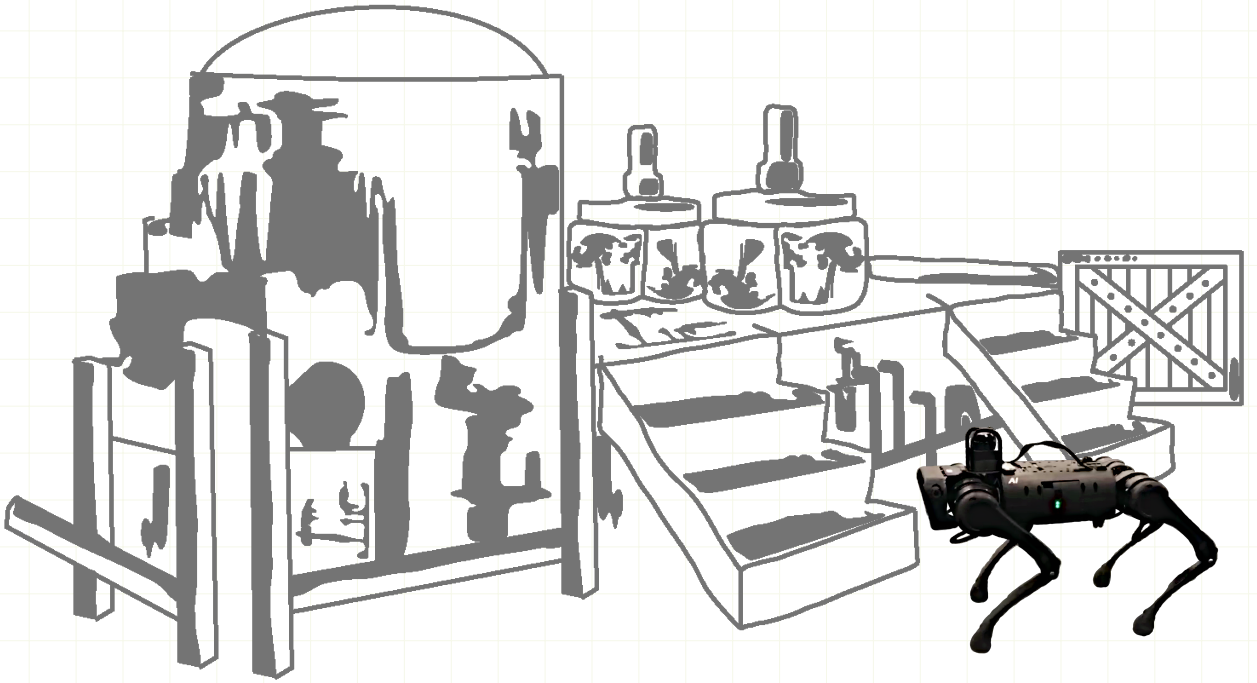
BASICS, ENVIRONMENTS

Pavel Osinenko



Basics, environments

Philosophy of RL → agent – environment interaction



Environment

Surroundings

The „outside" of the agent

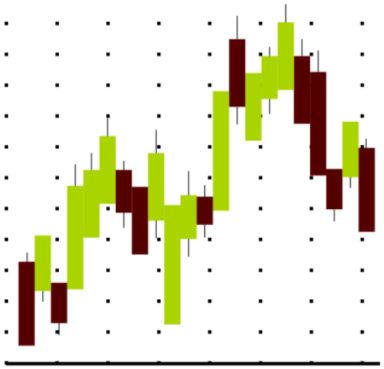
Agent

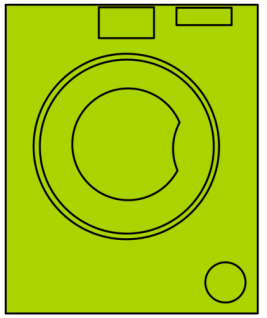
Active entity

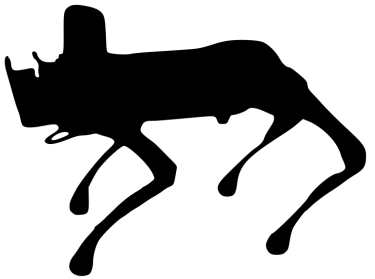
decision maker

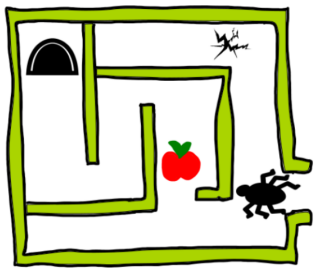
Contexts

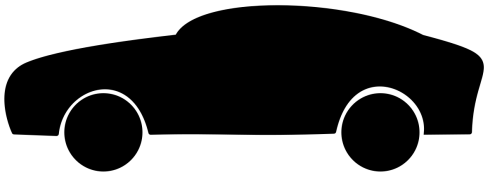
... can be anything, an object, a process, physical or virtual

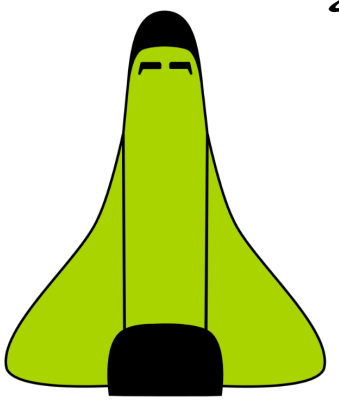


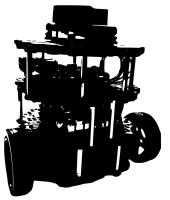




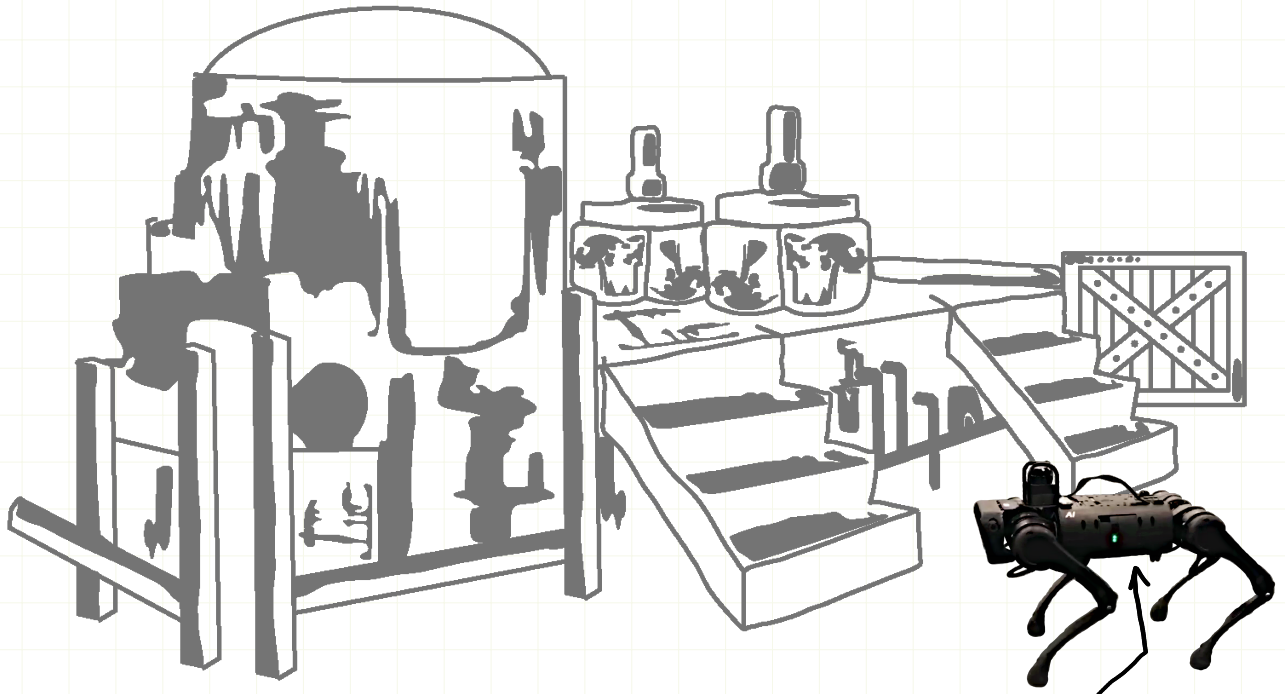




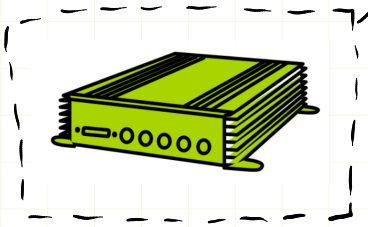




A closer look



Environment = robot body + its interaction with the surroundings



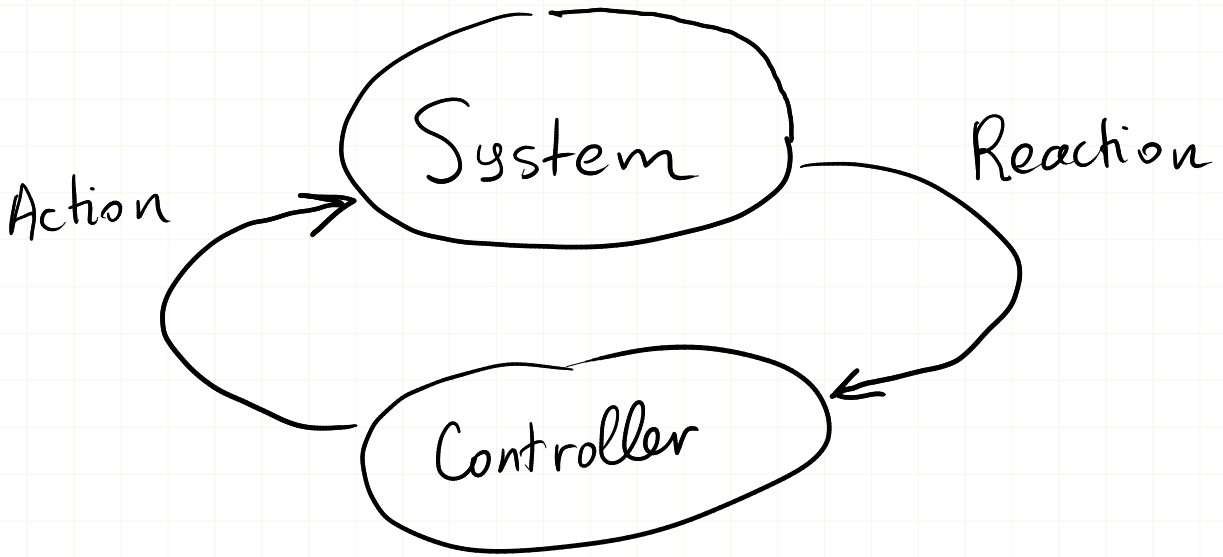
μC

What is the actual decision maker?

What are environments and agents really?

Answer: Environment = (dynamical) system

Agent = controller



Systems

Systems are entities that posses an internal state and a definite law of state evolution

Controllers

Controllers are entities that interact with systems. They perceive information about the system (its state in the simplest case) and produce control actions that are fed back to the system

Systems

- State x (or s)
- Input u (or a)
- Output y (or o)
- * Disturbance w (uncontrolled input)

Systems

- State transition law (might be implicit)

$$x_{next} \leftarrow f(x_{current}, u)$$

- Output map

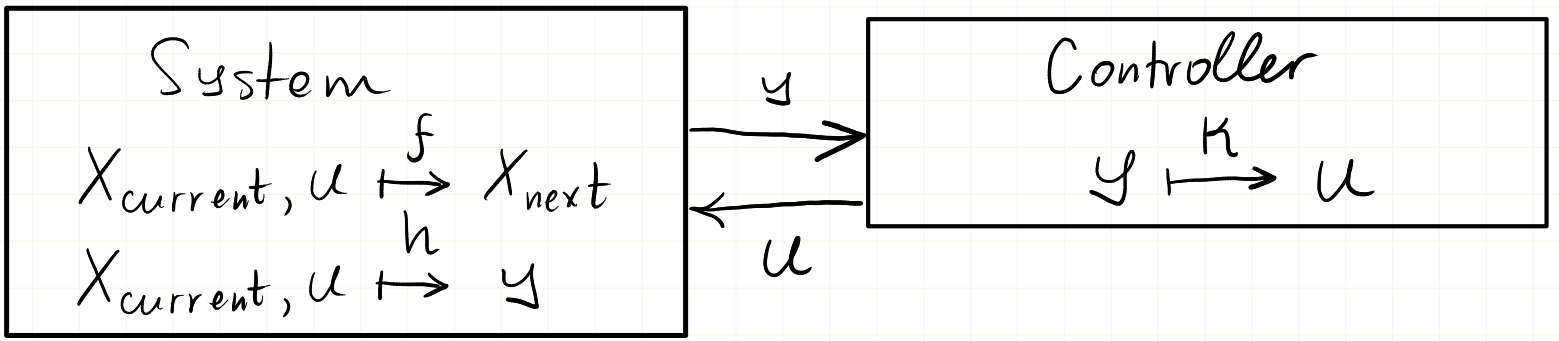
$$y \leftarrow h(x_{current}, u)$$

- f, h may depend on (system) parameters and disturbance w

Systems

A (feedback) law that transforms states into actions is called policy

The controller (agent) is the link associated with the policy



The simplest form of a system is a discrete-time deterministic one:

$$x_{k+1} = f(x_k, u_k)$$

$$y_k = h(x_k, u_k)$$

$$k \in \mathbb{N}_{+0}, \quad x \in X \subseteq \mathbb{R}^n, u \in U \subseteq \mathbb{R}^m, y \in Y \subseteq \mathbb{R}^l$$

These domains are also called state, action and output space, resp.

Policy: $u_k = K(x_k)$

Short name: DT system

Examples of DT systems

- A digital device
- A digital filter
- Population dynamics in generations

$$x_{k+1} = x_k + b_k + i_k - d_k - e_k$$

b_k: birth rate, say $b x_k$, $b = \text{const}$

i_k: # of immigrants

d_k: death rate, say $d x_k$, $d = \text{const}$

e_k: # of emigrants

Q: what could be controlled here?

Basics, environments

A particular probabilistic variant of a DT system is a **Markov decision process** (MDP) which is frequently used as the starting point in RL.

Description: $X_{k+1} \sim P_X(x_{k+1} \mid x_k, u_k)$

$U_k \sim P_U^{\vartheta}(u_k \mid x_k)$

Where: P_X, P_U^{ϑ} are probability distribution (or density if X, U are continuous) functions (PDF);
P_X is called **transition PDF**;
policy is represented here by a pdf with parameters ϑ, but may be a definite function (Markov policy) $u_k = \kappa(x_k)$

Let's clarify the notation:

$$X_{k+1} \sim P_X(x_{k+1} \mid x_k, u_k)$$

$$U_k \sim P_U^{\varphi}(u_k \mid x_k)$$

If $\mathcal{X}, \mathcal{U}$ are discrete, $P_X(x_{k+1} \mid x_k, u_k)$ can simply be interpreted as the probability of transitioning into x_{k+1} from the state x_k under action u_k, i.e.,

$$P_X(x_{k+1} \mid x_k, u_k) = \mathbb{P}\left[X_{k+1} = x_{k+1} \mid X_k = x_k, U_k = u_k\right]$$

If the state and action spaces are continuous, $P_X(x_{k+1} \mid x_k, u_k)$ is to be interpreted as conditional probability density $p_{X_{k+1} \mid X_k, U_k}(\bullet \mid x_k, u_k)$ where we used the function argument x_{k+1} at $\bullet$ for notational purposes

A practical example of an MDP is:

$$X_{k+1} \sim \mathcal{N}(f(X_k, U_k), \Sigma_W)$$

$$U_k \sim \mathcal{N}(K^\vartheta(X_k), \Sigma_U),$$

where $\mathcal{N}(\mu, \Sigma)$ is the normal PDF with mean μ and covariance Σ.

Informally, „$x_{k+1} = f(x_k, u_k) + w_k$", where w_k is a normal Gaussian noise with (system noise) covariance Σ_W;

„$u_k = K^\vartheta(x_k) + \sigma_k$",

where K^ϑ is parametrized function and σ_k is also a normal Gaussian noise with covariance Σ_U, which can in turn be seen as a policy parameter along ϑ

Concrete examples of MDPs

- A turn-based video game with RNG
- A chat bot
- Dayly trading on a stock market

$x = (\text{bond}, \text{share})$, k - day

$$x_{1,k+1} = (1+r)\,x_{1,k} + u_{1,k}, \quad r - \text{bond yield}$$

$$x_{2,k+1} = x_{2,k} + u_{2,k} + w_k, \quad w - \text{share price fluctuation}$$

$$y_k = x_{1,k} + x_{2,k}$$

Problem. Consider:

Chess 

x – state of all squares

e.g. $x = ($  $_{a1}$,  $_{a2}$, $\underbrace{\varnothing}_{a3}$, $\ldots)^T$

u – pair of squares

e.g. $u = (e2e4)$

What kind of a system is it? Why?
Can you come with a description?

Hint: consider the opponent as disturbance source

Most physical objects and processes are continuous-time though (modulo microscopic effects such as Plank time, perhaps)

We need to enrich our classification of environments

So, DT systems correspond to continuous-time (or CT) systems, MDPs — to stochastic CT systems

A CT system is described by differential equations (DEs)

General description:

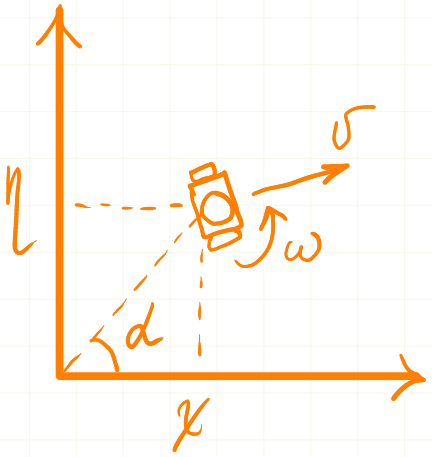
$$\dot{x} = f(x, u)$$
$$y = h(x, u)$$
$$x \in \mathbb{R}^n, \; u \in \mathbb{R}^m, \; y \in \mathbb{R}^l$$

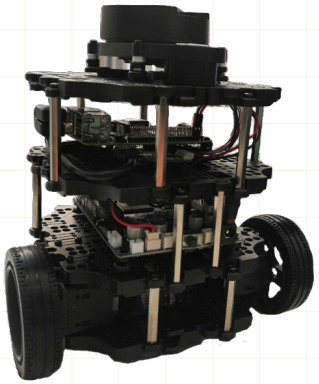
Policy: $u = k(x)$

Issues*: solution existence

Examples of CT systems

- A mobile wheeled robot





$$\dot{x} = v\cos\alpha$$
$$\dot{y} = v\sin\alpha$$
$$\dot{\alpha} = \omega$$
$$\dot{v} = \frac{1}{m}F$$
$$\dot{\omega} = \frac{1}{J}M$$

$$x = \begin{pmatrix} x \\ y \\ \alpha \\ v \\ \omega \end{pmatrix} \quad u = \begin{pmatrix} F \\ M \end{pmatrix}$$

A stochastic CT system is described by stochastic DEs (SDEs)

General description:

$$dX_t = f(X_t, U_t)dt + G(X_t, U_t)dB_t$$

drift (function) — diffusion (function) — Brownian motion

$$Y_t = h(X_t, U_t)$$

Policy (Markov): $U_t = K(X_t)$

Issues*: strong solution existence

Examples of stochastic CT systems

- Standard Black-Scholes model

$dS_t = r S_t dt$ — risk-free asset with rate of return r

$dR_t = \mu R_t dt + \sigma dB_t$ — risky asset with mean rate of return μ and volatility σ

X^S, X^R — investor wealth in bank account and stock

$X = X^S + X^R$ — total wealth

$\Pi_t := X^R / X$, C_t — consumption

Define $U_t := (\Pi_t, C_t)^T$

- Standard Black-Scholes model (cont.)

  The portfolio evolves as

  $$dX_t = ((r + \Pi_t(\mu - r))X_t - C_t)dt + \sigma \Pi_t X_t dB_t$$

  $$\mathbb{P}[X_0 = x_0] = 1 \quad \text{(initial capital)}$$

Further types of systems

- those described by partial differential equations (PDEs)
- hybrid systems: those that combine DT and CT dynamics

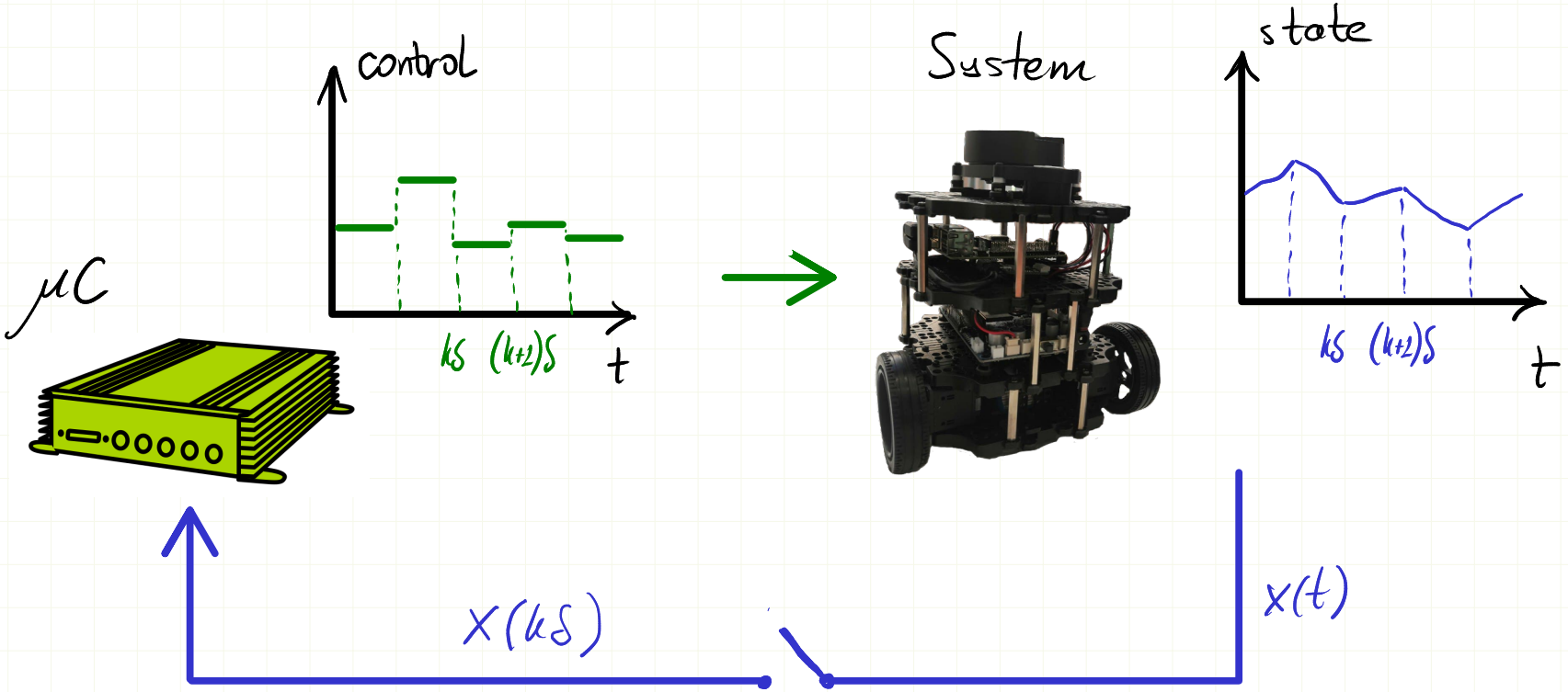
Important example:
sample-and-hold (SH) system

$$\dot{x} = f(x, u^\delta)$$

$$u^\delta(t) = u_k \text{ for } t \in [\delta k, \delta(k+1))$$

The control is piece-wise constant

SH systems are most adequate for description of digital control of physical objects and processes



Let's talk about agents (controllers).
What is their purpose?

Controllers serve to fulfil a control goal

In RL, this goal is optimization of some objective function

Thus, RL is a type of optimal control

The starting point here is to define such an objective for a moment of time, i.e., define instantaneous objective

Basics, environments

This instantaneous objective may have many names, depending on the problem, such as:

- reward, utility $\Rightarrow$ to be maximized
- stage, running or instantaneous cost $\Rightarrow$ to be minimized

Let $\mathcal{X}$ be the state space, $\mathcal{U}$ — action space

Then, an instant. objective is defined as a function $\rho: \mathcal{X} \times \mathcal{U} \longrightarrow \mathbb{R}$

Basics, environments

$\rho: \mathcal{X} \times \mathcal{U} \longrightarrow \mathbb{R}$

In most cases when ρ is a cost, its codomain is actually $\mathbb{R}_{\geq 0}$

In MDPs, we regard values of the instant. obj. as random variables (RVs), whereas ρ is stressed as, say, reward function

Now, from an instant. objective we proceed to the actual objective which in RL is the accumulated (expected, discounted) instant. objective

DT

$$\sum_{k=0}^{\infty} \gamma^k \rho(x_k, u_k)$$

↖ discount. factor $\gamma \in (0, 1]$

MDP

$$\mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k \rho(X_k, U_k)\right]$$

CT

$$\int_0^{\infty} e^{-\gamma t} \rho(x(t), u(t))\, dt$$

SDE

$$\mathbb{E}\left[\int_0^{\infty} e^{-\gamma t} \rho(X_t, U_t)\, dt\right]$$

The ∞ in the objective is also regarded as infinite horizon

Thus, RL is (commonly) a type of infinite horizon optimal control

Denote the objective of a policy π as $J(x_0 | \pi)$, where x_0 is the initial state

* Infinite horizon in the objective makes $J(x_0 | \pi)$ dependent only on x_0 and π.
  Why is this?

The optimal policy reads

$$K^*(x_0) := \arg \operatorname{opt} J(x_0 | K),$$

where „opt" is min or max

It is just a function of the initial state and regarded as the globally optimal controller

In contrast, finite-horizon optimal control gives rise to locally optimal controllers

* Finite-horizon optimal control follows the local formalism of Euler-Lagrange, while RL follows the Hamilton-Jacobi-Bellman formalism

Another names for the objective:

- accumulated reward
- cost-to-go

The optimum of the objective is called value function:

$$V(x_0) = \operatorname*{opt}_K J(x_0 | K)$$

Exercise: come up with instant. objectives for the environment examples of this lecture