Human-robot interaction

[ ] Dynamical system
[ ] Joint state and relative state
[ ] What is HRI hard?

Dynamical Systems

- model the motion of human and robot agents as a *dynamical system*

What is a dynamical system?
A set of quantities (ie. states) whose values evolve overtime.
The future values depend on the current value & potentially other external inputs (eg. controls, disturbances).

- describe dynamics using state-space representation.
(aka using vectors / Linear algebra)

<u>State</u> $x \in \mathcal{X} \subseteq \mathbb{R}^n, n \in \mathbb{Z}_+$ $\quad x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$

a <u>minimum</u> set of quantities needed to describe the system of interest.

eg. position, velocity heading

<u>Control</u> $u \in \mathcal{U} \subseteq \mathbb{R}^m, m \in \mathbb{Z}_+$, $\quad u = \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$

a set of quantities whose values can be selected by, eg, a human or algorithm

eg. acceleration, torque, yaw rate.

NOTE: in UG controls $u = k(x)$, k was a PID or leadlag.. proportional etc.

and focused on SISO

but in this class, we are interested in MIMO $\Rightarrow$ applying LA.

& also look at optimisation-based control.

<u>disturbance</u> $d \in D \subseteq \mathbb{R}^k$ $\quad d = \begin{bmatrix} d_1 \\ \vdots \\ d_k \end{bmatrix}$ disturbances that can affect the system dynamics, state & controls

eg- wind, ice, noise, anything that is not explicitly modeled.

<u>observations</u> $y \in Y \subseteq \mathbb{R}^l$ $\quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_l \end{bmatrix}$ Observations that can be measured by sensors on the system.

eg- IMU, accelerometer - GPS, etc

NOTE: assume system is fully observable $y = x$.

<u>dynamics</u>

$x = x(t)$, $\dot{x} = \frac{dx}{dt}$

$$\left.\begin{aligned} \dot{x} &= f_c(x, u, d, \cancel{t}) \\ y &= g_c(x, u, d, t) \end{aligned}\right\} \text{continuous-time dynamics.}$$

$t \in \mathbb{R}$

we can also have discrete-time dynamics

$x_{t+1} = f_d(x_t, u_t, d_t, t)$ $\quad t$: time step.

- get CT dyn from physics. eg. Newton's 2nd law

$\downarrow$ integrate

DT dynamics.

Properties of f_c / f_d

- typically dynamics are nonlinear. (AA/ME/EE 583)
- f can be linear $\dot{x} = Ax + Bu$, $x_{t+1} = Ax_t + Bu_t$ (547, 548)
- control affine systems $\dot{x} = f(x) + g(x)u + h(x)d$

Trajectories a sequence/signal of states (& controls)

CT: $\xi^{f,u(\cdot),d(\cdot)}_{x_0,t_0}(t)$ = trajectory starting at x_0, t_0, following dynamics f, executing $u(\cdot)$ & disturbance $d(\cdot)$ up to time t.

\xi

$X^{f,u(\cdot),d(\cdot)}_{x_0,t_0}(t)$ = state " " " "

" " " "

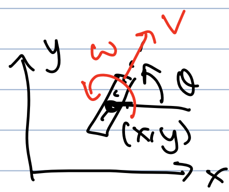
$\tau(x_0, t_0, f, u(\cdot), d(\cdot))$

→ make sure you define it clearly.

Simulating trajectories → INTEGRATE (CT)
STEP FORWARD (DT)

For CT, you can use ODE45, scipy.RK4(?)
scipy.integrate? | $\boxed{}$

eg. Simple unicycle



state: $\vec{x} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$

$$\dot{\vec{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ \omega \end{bmatrix} \qquad u = \begin{bmatrix} v \\ \omega \end{bmatrix}$$

$\dot{\vec{x}} = f(\vec{x}) + g(\vec{x})u$? Yes!

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}_{f(\vec{x})} + \underbrace{\begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix}}_{g(\vec{x})} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

add an integrator state

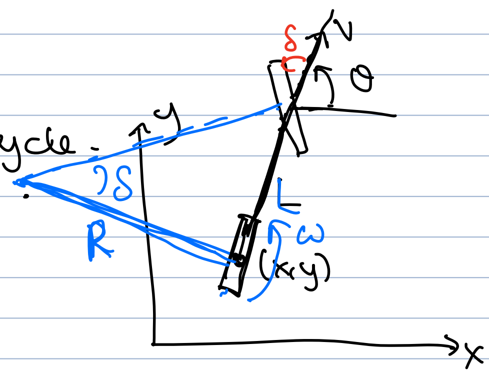
$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ \omega \\ a \end{bmatrix} \qquad u = \begin{bmatrix} \omega \\ a \end{bmatrix}$$

$$\begin{bmatrix} v\cos\theta \\ v\sin\theta \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w \\ a \end{bmatrix}$$

Q: given a general nonlinear dyn. sys, can we do some "trick" to make it control affine?

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \\ \dot{u}_1 \\ \dot{u}_2 \\ \vdots \\ \dot{u}_m \end{bmatrix} = \begin{bmatrix} f\left(\begin{bmatrix} x \\ u \end{bmatrix}\right) \\ \hline 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \hline \begin{matrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{matrix} \end{bmatrix} \begin{bmatrix} \tilde{u}_1 \\ \vdots \\ \tilde{u}_m \end{bmatrix}$$

eg| simple bicycle.



$$\vec{x} = \begin{bmatrix} x \\ y \\ \theta \\ v \end{bmatrix}$$

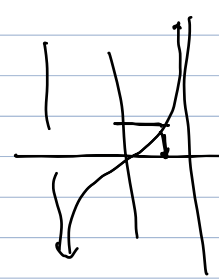
$v = \omega R$

$$\dot{\theta} = \omega = \frac{V}{R} = \frac{V}{L}\tan\delta$$

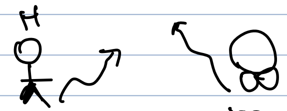
$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ \frac{v}{L}\tan\delta \\ a \end{bmatrix} \quad \tan\delta \to \hat{\delta} \qquad u = \begin{bmatrix} \delta \\ a \end{bmatrix}$$

control affine? No: $\tan\delta$ in nonlinear

if $\hat{\delta} = \tan\delta$



What about a human agent & robot together?



x_H, u_H, f_H $\quad$ x_R, u_R, f_R

$$\begin{bmatrix} \dot{x}_R \\ \dot{x}_H \end{bmatrix} = \begin{bmatrix} f_R(x_R, u_R) \\ f_H(x_H, u_H) \end{bmatrix}$$ done?

well... u_R depends on
- x_R & x_H (x_H relative to x_R)
- u_H - what the human will do.
- potentially future states / controls. knowing u_H future, f_H

$$u_R = \pi_R(x_R, x_H, u_H, \ldots)$$

similarly u_H depends on
- x_R & x_H (x_H relative to x_R)
- u_R - what the human will do.
- potentially future states / controls. knowing u_R future, f_R

$$u_H = \pi_H(x_R, x_H, u_R, \ldots)$$

INTERACTIONS!!!

what if there n-agents?

Q: how to model these interactions?

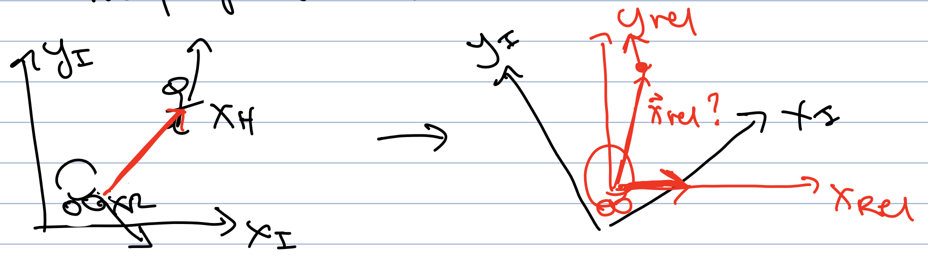
1. learn from data.
   eg. predict what u_H will be (prediction module #1)

2. Come up with a model to explicitly describe the interactions. (model-based)
   - game theory (guest speaker) (module #2, planning)

Relative dynamics

we can consider the relative state between two agents.

may get some reduction in state size



$$\begin{bmatrix} x_{rel} \\ y_{rel} \end{bmatrix} = \begin{bmatrix} \cos\theta_R & \sin\theta_R \\ -\sin\theta_R & \cos\theta_R \end{bmatrix} \begin{bmatrix} x_H - x_R \\ y_H - y_R \end{bmatrix}$$

$$x_{rel} = \cos\theta_R (x_H - x_R) + \sin\theta_R (y_H - y_R)$$

$\dot{x}_{rel} = ?$

$$\Rightarrow \vec{x}_{rel} = \begin{bmatrix} x_{rel} \\ y_{rel} \\ \theta_{rel} \\ v_H \\ v_R \end{bmatrix}$$

$\theta_{rel} = \theta_H - \theta_R$

v_{rel} ? $\cancel{v_H - v_R}$

v_H

v_R