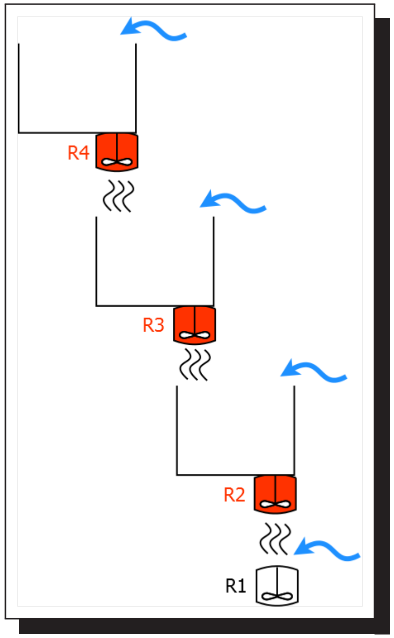
Learning Decisions from Preferences

Doina Precup

COMP579, Lecture 22, part 2

Example: Power Plant Control



- 3 turbines to control (continuous variables), one per reservoir
- turbine R1 is controlled by the water flow
- (stochastic) ground water inflows
- weekly time steps
- objective: maximize average annual power production while satisfying constraints (see below)

Cf. Grinberg et al, 2014; collaboration with Hydro Quebec

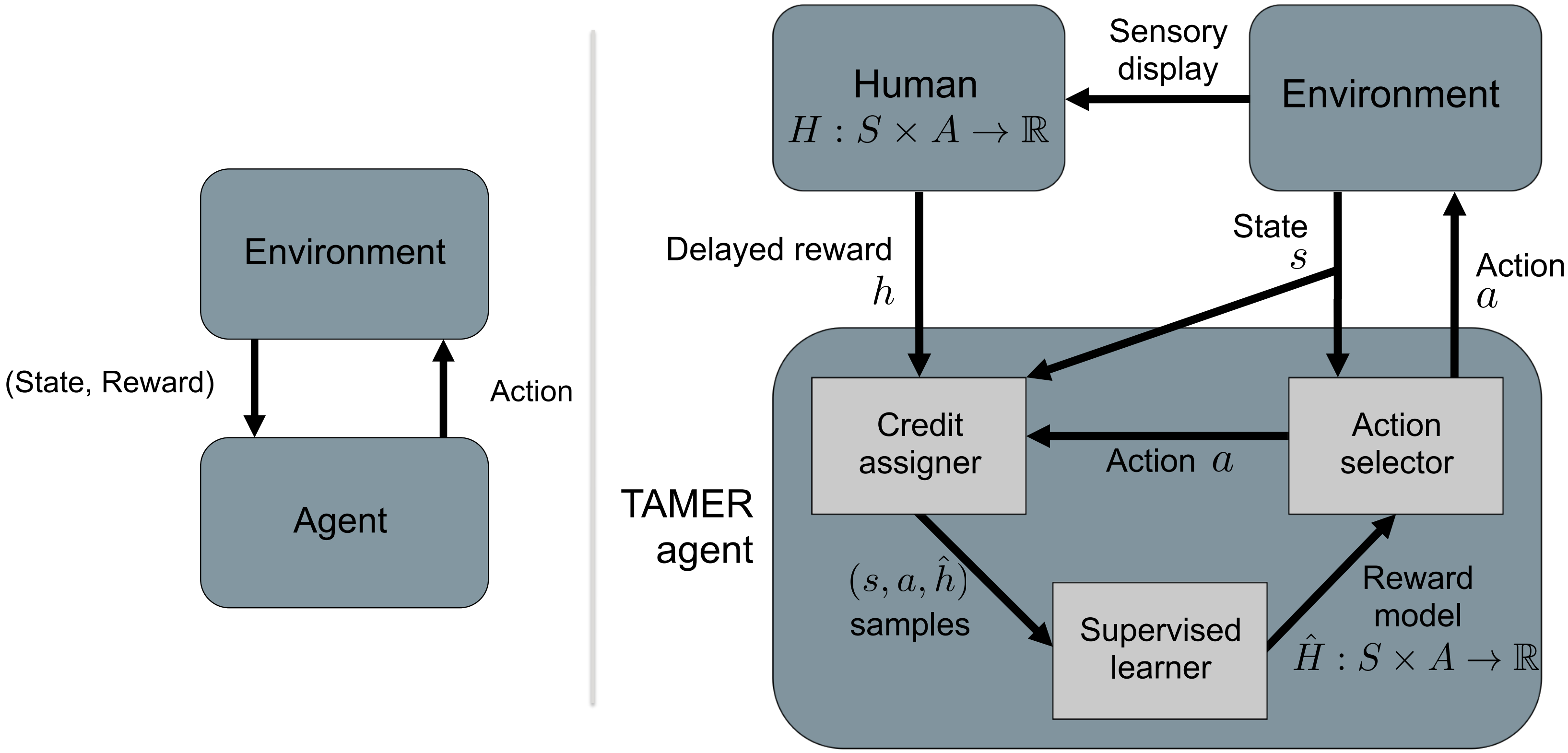
- Major: sufficient flow needs to be maintained to allow easy passage for fish
- Major: stable turbine speed throughout weeks 43-45 to allow fish spawning
- Minor: amount of water in second reservoir should be above a minimum

Reward function can be quite hard to formulate!

How to Solve Power Plant Control?

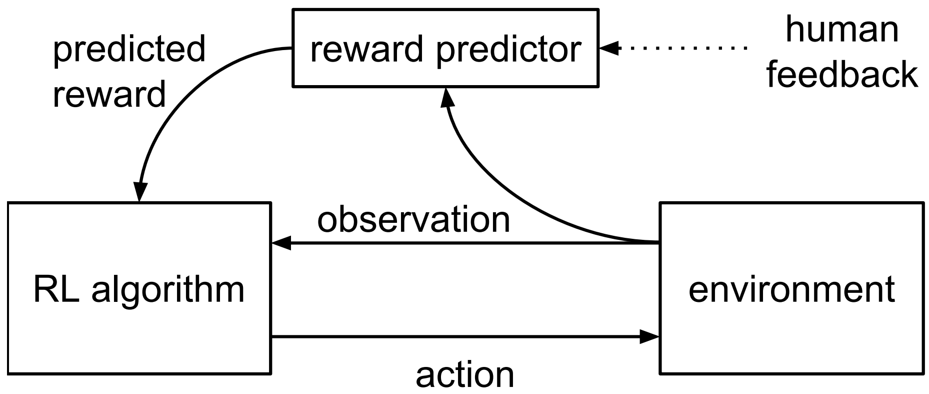
- Spent a lot of time trying to craft a reward function that captures the objective
- *Reward hacking is a major issue*
- Tried various constrained and risk-sensitive optimization (hyper-parameter tuning is no better than fitting rewards)
- Ended up doing *randomized policy search*!

Learning from Human Feedback (Knox, 2012)



- Numerical reward is a high-variance signal even when learned

Deep RL from Human Feedback (Christiano et al, 2017)



- People provide a *preference* among two choices
- Assuming there is a latent variable explaining the choice, reward is fit using maximum likelihood (Bradley-Terry model)
- Cf. https://arxiv.org/pdf/1706.03741.pdf

Bradely-Terry reward model

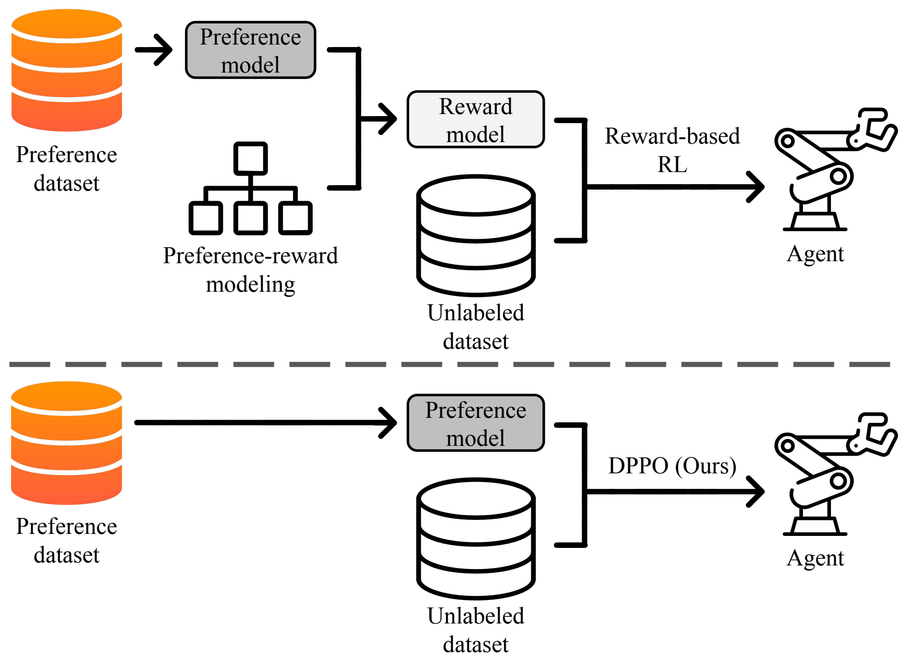
- Collect data from human raters (pairs of y_w, y_l responses to a prompt x)
- Optimize the expected value of:

$$-\log(\sigma(r_\theta(x, y_w) - r_\theta(x, y_l)))$$

  wrt reward parameter vector θ

- Cf. Ouyang et al, InstructGPT
- Corresponds to maximum likelihood fitting of binomial preference function if reward is linear over the variables

Direct Preference Optimization



- Cf. An et al, NeurIPS'2023 (https://arxiv.org/pdf/2301.12842.pdf)
- Direct preference optimization (Rafailov et al, NeurIPS'2023, https://arxiv.org/pdf/2305.18290.pdf)
- Several other almost-concurrent papers in this space

Optimizing Preferences: Setup

- An agent interacting with an environment receives observations for a set $\mathcal{O}$ and performs action from set $\mathcal{A}$
- A *history* h_t is a sequence of observation-action pairs $\langle o_0, a_0, o_1, a_1, \ldots o_t \rangle$
- A *policy* π is a mapping from histories to actions: $\pi : \mathcal{H} \rightarrow \mathcal{A}$
- Consider a *binary relation over trajectory distributions* $\preceq$
- A policy π in an environment e induces a probability distribution over trajectories, D^π
- See Colaco-Carr et al, AISTATS'2024 (https://arxiv.org/abs/2311.01990)

Preference Relations and Their Properties

- We will formalize preference relations through pre-orders
- For trajectory distributions A and B, $A \preceq B$ means is that B is at least as preferred as A
- $\preceq$ is a *pre-order* if it satisfies:
  - *Reflexivity*: $A \preceq A$
  - *Transitivity*:if $A \preceq B$ and $B \preceq C$ the $A \preceq C$
- A pre-order is *total* if for and A, B, $A \preceq B$ and $B \preceq A$

Direct Preference Process

- A *Direct Preference Process* is a tuple $\mathcal{O}, \mathcal{A}, T, e, \preceq$ where:
  - $\mathcal{O}$ is an observation set
  - $\mathcal{A}$ is an action set
  - T is a time horizon
  - e is an environment (transition function from achievable history-action pairs to the next observation)
  - $\preceq$ is a binary (preference) relation over trajectory distributions
- $\preceq$ is *expressible through a reward function* $r : \mathcal{H} \rightarrow \mathbb{R}$ if:

$$\forall A, B, A \preceq B \text{ if and only if } \mathbb{E}_A \left[\sum_{t=0}^{T} r(H_t) \right] \leq \mathbb{E}_B \left[\sum_{t=0}^{T} r(H_t) \right]$$

Preference Relations and Their Properties

- A total pre-order is *consistent* if

$$\forall\alpha \in (0,1), \forall A, B, C, A \preceq B \implies \alpha A + (1-\alpha)C \preceq \alpha B + (1-\alpha)C$$

- A total pre-order is *convex* if

$\forall\alpha \in (0,1), \forall A, B, C, A \preceq B$. if and only if $\alpha A{+}(1{-}\alpha)C \preceq \alpha B{+}(1{-}\alpha)C$

- A total pre-order has the *interpolation property* if

$\forall A, B, C, A \preceq B$ and $B \preceq C$ implies $\exists\alpha \in (0,1), \alpha A + (1-\alpha)C \sim B$

- Von Neumann-Morgenstern theorem: if all the above hold, $\preceq$ can be expressed by a utility function

When Are Preferences Representable By Reward Functions?

- Main result
  - *If convexity and/or interpolation do not hold, $\preceq$ is NOT is expressible through a reward function*
  - *However, total consistent pre-orders have deterministic optimal policy!*
- The latter situation is not exotic or rare!

Examples when Optimal Policies Exist Without Rewards

- *Total consistent convex pre-order not satisfying interpolation: tie-breaking criteria*
  - Use a first criterion, if tied go to a second criterion
  - See not flooding vs water in second reservoir in power plant example
- *Total consistent pre-order that is non-convex: excess risk*
  - If risky event does not occur, linear utility
  - Risky event occurring entails exponential penalty
  - No flooding neighbouring areas in power plant example

How Do We Compute Optimal Policies?

- If $\preceq$ is a total consistent pre-order and a policy π satisfies the following for any attainable history $h_t, t < T$ and any action a_t:

$$D^\pi(h_t \cdot a_t) \preceq D^\pi(h_t)$$

  then π is $\preceq$-optimal
- So we are *justified to do policy search*!
- If $\preceq$ is expressible through a reward function, value iteration is a direct consequence of this result

Discussion

- Nice to know that aproaches such as direct preference optimization are justified
- Our results are currently on distributions - working on sample-based extensions
- If we can fit a reward function, should we?
  - Bias-variance trade-off? Sample complexity considerations?
- *What can we do if other properties of pre-orders are violated?*

Learning with non-transitive preferences: NashLLM

- Objective:find a policy π^* which is preferred over any other policy

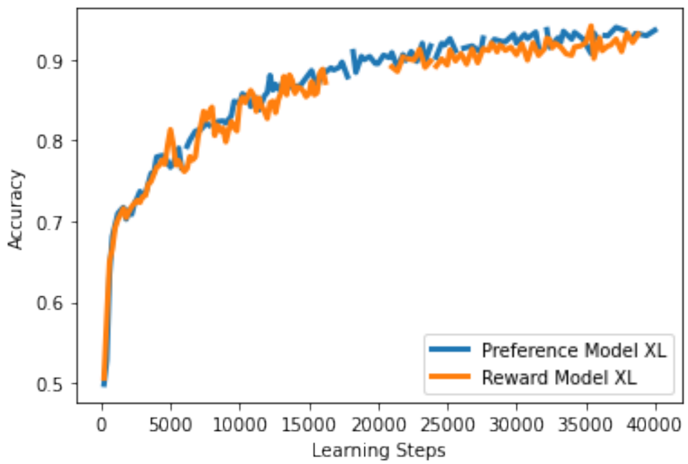
$$\pi^* = \arg\max_{\pi} \min_{\pi'} \mathbb{P}(\pi' \preceq \pi)$$

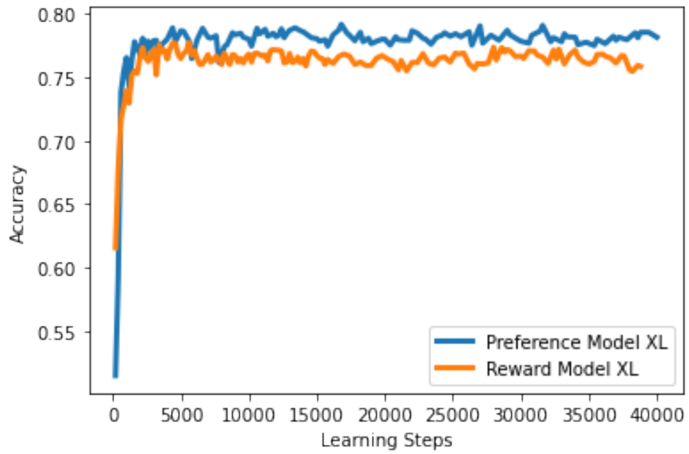
- Think of this as a game: one player picks π the other picks π'
- When both players use π^* this is a *Nash equilibrium* for the game
- For this game an equilibrium exists (even if eg preferences are not transitive)
- Cf. Munos et al, 2024 (https://arxiv.org/pdf/2312.00886.pdf)

NashLLM-style algorithms

- Fit a *two-argument preference function* by supervised learning
- Decide what is the *set of opponent policies*
- Ideally, the max player should play against a mixture of past policies
- *Optimize* using eg online mirror descent, convex-concave optimization...
- A lot of algorithmic variations to explore!

NashLLM results





Using preferences instead of rewards leads to less overfitting