AUP - Theory

Fabrice Beaumont

28. Mai 2023

Inhaltsverzeichnis

1 Notation **3**

2 Key notes **4**

3 Paper summaries **5**

Literaturverzeichnis **10**

Included papers:

- [1]
-

Papers to include:

- [2]
- [3]
- [4]

1 Notation

- Arbitrary states x, y
  states at time step t: s_t
- Baseline state s'_t
  Starting state baseline s_0, inaction baseline $s_t^{(0)}$, stepwise inaction baseline $s_t^{(t-1)}$
- Noop-action a^{noop}
- $s_m^{(t)}$ - inaction rollout of $m - t$ no-operation actions after state s_t ($m = t + k > t$).

2 Key notes

- Σ - Set of states
- A - Set of actions

Important definitions :

- $C(\tilde{s}, s) \in [0, 1]$ - how easily can we obtain s form $\tilde{s}$ (**coverage**, **reachability**)
- Coverage matrix: $R = C \in \mathbb{R}^{|\Sigma| \times |\Sigma|}$.
  (Written as C or R (if implemented as reachability).)
- $Q \in [0, 1]^{|\Sigma| \times |A|}$ - **Q-table**
  $q_{s,a}$ - if in state s, how rewarding is it to perform action a
- $D \in \mathbb{R}^{|\Sigma|} = \left(d(S_t, S')\right)$ - deviation of state S_t compared to the some baseline state S'

Paper differences : Updates of the [1] paper, compared to the old [2] paper:

- Write R instead of C
- Introduction of the **stepwise inaction baseline**
- Introduction of the scaling parameter beta β
- Scaling the computation of the d_{RR} by the number of states

3 Paper summaries

Consider environments as discounted Markov Decision Processes (MDPs) $(S, \mathcal{A}, r, p, \gamma)$:

- S - set of states
  s'_t - baseline state at time t
- $\mathcal{A}$ - set of actions
  $a^{\text{noop}} \in \mathcal{A}$ - special no-operation action
- $r : S \times \mathcal{A} \to \mathbb{R}$ - reward function
- $p : S \times S \times \mathcal{A} \to [0, 1]$ - transition function
- $\gamma \in (0, 1)$ - discount factor (sometimes written as γ_r - reachability discount factor w.r.t. reward function r)
- $d : S \times \{s'_t\} \to \mathbb{R}$ - deviation
- $\beta \in \mathbb{R}$ - **deviation penalty** (**learned parameter**)

At time step t, the agent receives the current state s_t, outputs the action a_t drawn from its policy $\pi(a_t|s_t)$ and receives reward $r(s_t, a_t)$.

DEF: Desirable properties of XZ

Penalize the agent for effects on the environment if and only if those effects are unnecessary for ac

(Property 1)

Distinguish between agent effects and environment events, and only penalize the agent for the for

(Property 2)

Sensitivity to the reversible-ness of the agent's effects:

Give a higher penalty for irreversible effects than for reversible effects. [2]
(Property 3)

Sensitivity to the magnitude of the agentâ€™s irreversible effects:

(Cumulative penalty) The penalty should accumulate when more irreversible effects occur. [2]
(Property 4)

For example, if the agent starts in state S_0, takes an irreversible action that leads to state S_1, and then takes another irreversible action that leads to state S_2, then $d(S_2; S_0) > d(S_1; S_0)$.

DEF: Baselines

We define different baselines in order to compare the actions of an agent at any state with these baselines.
Starting state: Use the state of the environment at $t = 0$.
Inaction baseline: Simulate the environment as if the agent never spawned. (That is it performs the a^{noop} at every time-step.)
Stepwise-inaction baseline: Simulate the environment as if the agent has done noting, instead of the last chosen action.
It may be useful to not (just) compare the current state and baseline state of the current time-step **inaction-rollout**.

DEF: Intrinsic pseudo-reward

By adding a **penalty for side effects**[a] to the reward function we can implement an intrinsic pseudo-reward. Therefore, we subtract at time t an impact penalty, which is a scaled deviation penalty from the deviation of the current state from the baseline state s'_t.

$$r_\beta(s_t, a_t) := r(s_t, a_t) - \beta \cdot d(s_{t+1}, s'_{t+1})$$

[a] Side effects are impacts to the environment, which are not necessary to complete the main task

DEF: Inaction rollout

An **inaction rollout** from state s_t is a sequence of states obtained by following the inaction policy (a^{noop}) starting from that state. Thus state $s_{t+2}^{(t)}$ denotes the state at time step $t+2$, after arriving at state s_t at time step t and performing the no-operation action for two time steps.
This allows for an easy comparison to environment state after choosing the no-operation action at every time step starting from the baseline state: $s_{t+2}^{\prime(t)}$

DEF: Deviation measure

...
is called **symmetric**, if ...

Using a symmetric deviation measure implies, that all actions are reversible.

We define an asymmetric deviation measure called **reachability**.

DEF: Reachability

We define a **reachability** $R : S \times S \to [0,1]$ as a measure of difficulty to get from state x to state y ($x \neq y$). We use the parameter $\gamma_r \in (0,1]$ to define the importance of time. For low γ, it is expensive to need more time-steps. For high γ, it is cheaper to need more time-steps. The special case $\gamma = 1$, where times does not matter is discussed below.

$$R(x,y) := \max_{\pi} \gamma^{N_\pi(x,y)} \qquad \left(= \max_{\pi} \mathbb{E}\left[\gamma^{N_\pi(x,y)]}\right]\right)$$

(Use the $\mathbb{E}$ notation only for the proof of the statement below for undiscounted reachability.)
A recursive computation can be done like this:

$$\begin{aligned} R(x,y) &:= \gamma_r \max_{a} \sum_{z \in S} p(z|\ x,a) R(z,y) \\ &\stackrel{?}{=} \gamma_r^n \max_{a_1} \sum_{z_1 \in S} p(z_1|\ x,a_1) \Big(\max_{a_2} \sum_{z_2 \in S} p(z_2|\ z_1,a_2) \dots \max_{a_n} \sum_{z_n \in S} (0 + p(y|\ z_n,a_n) * 1) \Big) \end{aligned}$$

where $n = N_\pi(x,y)$. And it is $R(y,y) = 1$.

Special case: **Undiscounted reachability** ($\gamma_r = 1$), which computes whether y is reachable in any number of steps. In this cased it is (see paper for proof):

$$R(x,y) = \max_{\pi} \mathbb{P}\big(N_\pi(x,y) < \infty\big) = \begin{cases} 1 \; y \text{ is reachable from } x \\ 0 \text{ otherwise} \end{cases}$$

The **unreachability** (UR) **deviation measure** $d_{\mathrm{UR}} : S \times S \to [0,1]$ is then defined as:

$$d_{\mathrm{UR}}(x,y) := 1 - R(x,y)$$

$d_{\mathrm{UR}}(x,y)$ close to 1 means low reachability, high unreachability.
Note: The undiscounted unreachability measure only penalizes irreversible transitions[a], while the discounted measure also penalizes reversible transitions.

[a] $d_{\mathrm{UR}}(x,y) = 1$ if unreachable, 0 else.

The unreachability deviation measure is often used to compute the unreachability to the baseline state s'_t from a state s_t: $d_{\mathrm{UR}}(s_t, s'_t)$.

DEF: Relative reachability

The **relative reachability** (**RR**) measure $d_{\mathrm{RR}} : S \times S \to [0,1]$ is the average reduction in reachability of all states s from the current state s_t compared to the baseline s'_t:

$$d_{\mathrm{RR}}(x,y) := \frac{1}{|S|} \sum_{s \in S} \max\big(R(s'_t, s) - R(s_t, s),\ 0\big)$$

DEF: Attainable utility

The **attainable utility measure** $d_{\mathrm{VD}} : S \times S \to \mathbb{R}$ denotes the average gain in reward by obtaining a state x compared to a state y. To define it, we define the **value** $V_r : S \to \mathbb{R}$ of a state x according to a reward function r. Therefore let x_t^π denote the state obtained from x by following policy π for t steps. It is

$$V_r(x) := \max_{\pi} \sum_{t=0}^{\infty} \gamma_r^k \, r(x_t^\pi)$$

TODO: Was ist k? x is die reward function r? Die ist aber fuer States UND Aktionen definiert. Ist die Summe aller rewards fuer alle States und Aktionen gemaess π gemeint?] $r(x_t^\pi) = \sum_{t_1=0}^{\infty} r(x_{t_1}, a_{t_1})$ sodass $(x_{t_1}, a_{t_1}) \in \pi$. TODO: Vermutung fuer k: Wie beim inaction rollout time step difference zum aktuellen time step. Frage: Sollte das values eines states nicht abhaengig vom time step des states sein?

Literaturverzeichnis

[1] Victoria Krakovna, Laurent Orseau, Ramana Kumar, Miljan Martic, and Shane Legg. Penalizing side effects using stepwise relative reachability. arXiv preprint arXiv:1806.01186, 2018.

[2] Victoria Krakovna, Laurent Orseau, Miljan Martic, and Shane Legg. Measuring and avoiding side effects using relative reachability. arXiv preprint arXiv:1806.01186, 2018.

[3] Alex Turner, Neale Ratzlaff, and Prasad Tadepalli. Avoiding side effects in complex environments. Advances in Neural Information Processing Systems, 33:21406–21415, 2020.

[4] Alexander Matt Turner, Dylan Hadfield-Menell, and Prasad Tadepalli. Conservative agency via attainable utility preservation. In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, pages 385–391, 2020.