AIXI Responses to Newcomblike Problems

Davide Zagami[1]
[1]CEEALAR
[1]zagamidavide@gmail.com

Abstract

We provide a rigorous analysis of AIXI's behaviour under repeated Newcomblike settings. In this context, a Newcomblike problem is a setting where an agent is tied against an environment that contains a perfect predictor, whose predictions are used to determine the environmet's outputs. Since AIXI lacks good convergence properties, we chose to focus the analysis on determining whether an environment appears computable to AIXI, that is, if it maps actions to observations in a way that a computable program can achieve. It is in this sense that, it turns out, AIXI can learn to one-box in repeated Opaque Newcomb, and to smoke in repeated Smoking Lesion, but may fail all other Newcomblike problems, because we found no way to reduce them in a computable form. However, we still suspect that AIXI can succeed in the repeated settings.

1 Motivation and past work

The question of how Newcomb's problem would be handled by agents that learn their environments seems underexplored. Xi Li [Li, 2019] attempts to analyse how AIXI handles Newcomb's problem, without analyzing the setup in general.

Oesterheld [Oesterheld, 2019] tries to derive what decision theory is implemented by approval directed agents. A team in a previous iteration of AISC is analyzing how bandit algorithms would behave in Newcomb-like situations and found that, while they don't implement any decision theory among CDT, EDT, or FDT, they implement ratifiable policies; an action is ratifiable if that action is optimal conditional on that action being taken.

Another attempt [Everitt *et al.*, 2019] analyses sequential Newcomb-like problems with CDT and EDT. CDT had one natural extension to this problem space, while EDT had two; depending on whether the agent updates based on its actions or policy.

Botworld [Soares and Fallenstein, 2014] is a toy environment developed by MIRI in which to study self-modifying agents embedded in their environment. A Newcomb-like scenario explored in Botworld is The Precommitment Game, where an agent similar to AIXI fails to take the optimal action when tied against an opponent that can read a portion of its source code in an infinitely repeated setting.

Outside of the Botworld toy models, Newcomb problems have historically been analyzed in the context of agents already knowing the environment and taking actions based on the rules dictated by their decision theory. Vanessa Kosoy [Kosoy, 2019] has pointed out that we can't decouple the problem of learning a model of the world from the problem of taking a decision given such a model, and given an example of how Quasi Bayesian Agents can solve repeated counterfactual mugging.

This makes it suspect that analysing how learning agents behave in Newcomblike situations may be a better idea than the Decision Theory approach, where the agent already knows the environment and takes actions based on the rules dictated by its decision theory. In any case, developing a thorough analysis of every known Newcomb-like problem, as handled by learning agents such as AIXI, reflective oracles [Fallenstein *et al.*, 2015], or Quasi Bayesian Agents [Kosoy, 2019], seems important.

2 AIXI's definition

AIXI [Hutter, 2004] is a theoretical model of artificial general intelligence, under the framework of reinforcement learning, that describes optimal agent behavior given unlimited computing power and minimal assumptions about the environment.

In reinforcement learning, the agent-environment interaction consists of a turn-based game with discrete time-steps [Sutton *et al.*, 1998]. At time-step t, the agent sends an action a_t to the environment, which in turn sends the agent a percept that consists of an observation and reward tuple, $e_t = (o_t, r_t)$. This procedure continues indefinitely or eventually terminates, depending on the episodic or non-episodic nature of the task.

Actions are selected from an action space A that is usually finite, and the percepts from a percept space $\mathcal{E} = O \times R$, where O is the observation space, and R is the reward space that is usually bounded to $[0, 1]$.

For any sequence $x_1, x_2, ...$, the part between t and k is denoted $x_{t:k} = x_t...x_k$. The shorthand $x_{<t} = x_{1:t-1}$ denotes sequences starting from time-step 1 and ending at $t-1$, while $x_{1:\infty} = x_1x_2...$ denotes an infinite sequence. Sequences can

be appended to each other, for example, $x_{<t}x_{t:k} = x_{1:k}$. Finally, $x*$ is any infinite string beginning with x.

The environment is modeled by a deterministic program q of length $l(q)$, and the future percepts $e_{<m} = U(q, a_{<m})$ up to a horizon m are computed by a universal (monotone Turing) machine U executing q given $a_{<m}$. The probability of percept e_t given history $ae_{<t}a_t$ is thus given by:

$$P(e_t \mid ae_{<t}a_t) = \sum_{q:U\left(q,a_{\leq t}\right)=e_{\leq t}*} 2^{-l(q)} \tag{1}$$

where Solomonoff's universal prior is used to assign a prior belief to each program.

An agent can be identified with its policy, which is a distribution over actions $\pi(a_t \mid ae_{<t})$.

If the agent is rational in the Von Neumann-Morgenstern sense [Morgenstern and Von Neumann, 1953], it should maximize the expected return, as computed by the value function:

$$\begin{aligned} V^{\pi}\left(ae_{<t}\right) = &\sum_{a_t \in \mathcal{A}} \pi\left(a_t | ae_{<t}\right) \\ &\cdot \sum_{e_t \in \mathcal{E}} P\left(e_t | ae_{<t}a_t\right)\left[\gamma_t r_t + \gamma_{t+1} V^{\pi}\left(ae_{1:t}\right)\right] \end{aligned} \tag{2}$$

where $\gamma : \mathbb{N} \to [0, 1]$ is a discount function with convergent sum.

In other words, the AIXI agent uses the policy:

$$\pi^{\text{AIXI}}(ae_{<t}) = \arg\max_{\pi \in \Pi} V^{\pi}(ae_{<t}) \tag{3}$$

3 AIXI's optimality

Let μ be the true environment. AIXI is known to have the property of *on-policy value convergence*; that is:

$$V_{\xi}^{\pi} - V_{\mu}^{\pi} \to 0$$

This means that AIXI learns how good its policy is, asymptotically. Unfortunately, this is not the same as *asymptotic optimality*, a property that AIXI indeed lacks:

$$V_{\mu}^{\star} - V_{\mu}^{\pi} \to 0$$

Why this is the case is somewhat complicated [Amodei and Clark, 2016] and relies on two facts: an "unreasonable" enough UTM can make AIXI perform arbitrarily bad, and we don't know how to formally specify a "reasonable" UTM.

Our opinion on this is that those reasons are not very concerning. Let ζ be a prior over programs. It can be a Solomonoff prior (which hereafter we denote with ξ_U to indicate that it is relative to a Universal Turing Machine U) or any other prior (such as a prior that only contains bandit environments). We use AIζ to denote a version of AIXI that is modified to use ζ instead of ξ_U.

We know that if ζ contains only ergodic MDPs, and if the true environment is actually ergodic, then AIζ exhibits asymptotic optimality. Ergodic MDPs are the largest environment class for which we know this, and so far there are no results saying that this is the largest class for which this is possible, so we'll probably prove similar results for larger and larger classes.

In any case, repeated games are ergodic MDPs, so when we analyse AIXI's responses in repeated Newcomblike problems, we'll tacitly assume that we are working with AIζ, where ζ is the largest environment class for which AIζ exhibits asymptotic optimality, and we'll focus on checking whether the true environment **appears computable to AIXI**. Roughly speaking, an environment (that may be uncomputable because of having oracle access to AIXI's policy) appears computable to AIXI if it maps actions to observations in a way that a computable program in ζ can achieve. It turns out that this is sometimes possible, and sometimes not, depending on the problem setting.

4 Newcomblike problems

For the purposes of this analysis, a Newcomblike problem is a setting where an agent, such as AIXI, is tied against an environment such that:

- It contains a perfect predictor (in the form of oracle access to AIXI's policy);
- The outputs of the environment depend on the predictor's predictions.

4.1 Notation and modeling choices

Let $e^{Newcomblike}$ be a Newcomblike environment. We formalize the setup in the following way.

At each step t, π_{ζ}^{AIXI} outputs $a_t \in A$ (for example, $A = \{a^1, a^2\}$ for one-box or two-box) according to the usual (uncomputable) algorithm. The environment $e^{Newcomblike}$ will output $e_t = (o_t, r_t)$, with $o_t \in O$ (for example, $O = \{o_e, o_f, \dots\}$ for observing an empty box or a full box before acting in transparent Newcomb) and $r_t \in [0, 1] \cap \mathbb{Q}$ (dollars received normalized to $[0, 1]$).

Unless otherwise stated, every episode is completely independent from the others, and AIXI can retain memory of past episodes. Every episode consists of one timestep worth of action-observation pair.

We use the notation $[x = y]$ to mean 1 if $x = y$, and 0 otherwise.

4.2 Opaque Newcomb

An agent finds herself standing in front of a transparent box labeled "A" that contains \$1,000, and an opaque box labeled "B" that contains either \$1,000,000 or \$0. A reliable predictor, who has made similar predictions in the past and been correct 99% of the time, claims to have placed \$1,000,000 in box B iff she predicted that the agent would leave box A behind. The predictor has already made her prediction and left. Box B is now empty or full. Should the agent take both boxes ("two-boxing"), or only box B, leaving the transparent box containing \$1,000 behind ("one-boxing")?

$e^{Newcomb}$ acts in the following way. Given a history $h = ae_{<t}$:

- Let $\mathcal{P}(h)$ be the function that is a^1 if the predictor predicts that AIXI one-boxes, a^2 otherwise.
  - The predictor has access to a copy of π_ζ^{AIXI}, and to an oracle that can compute $\pi_\zeta^{AIXI}(h)$, therefore $\mathcal{P}(h) = \pi_\zeta^{AIXI}(h)$
- When it is AIXI's turn to move, it outputs $a_t = \pi_\zeta^{AIXI}(h)$, and then the environment outputs $e_t = e^{Newcomb}(ha_t)$ with:
  - $o_t \in \{o_1^+, o_1^-, o_2^+, o_2^-\}$ (the number indicates how many boxes were received, the sign whether there was money in the "Newcomb" box), where the dependency mapping between the agent's action and the predictor's prediction is:
    - $o_t = o_1^+ \iff \mathcal{P}(h) = a^1$ and $a_t = a^1$
    - $o_t = o_1^- \iff \mathcal{P}(h) = a^2$ and $a_t = a^1$
    - $o_t = o_2^+ \iff \mathcal{P}(h) = a^1$ and $a_t = a^2$
    - $o_t = o_2^- \iff \mathcal{P}(h) = a^2$ and $a_t = a^2$
  - $r_t = [\mathcal{P}(h) = a^1] \cdot 1000000 + [a_t = a^2] \cdot 1000$

Clearly, $\mathcal{P}(h) = a_t$.

Thus, let's say that $a_t = a^1$. Then $o_t = o_1^+$ and $r_t = 1000000$.

Conversely, if $a_t = a^2$ then $o_t = o_2^-$ and $r_t = 1000$.

The (computable) program that encodes this dynamic is

$$q^{Newcomb}(ha_t) = (o_1^+ \cdot [a_t = a^1] + o_2^- \cdot [a_t = a^2],$$
$$[a_t = a^1] \cdot 1000000 + [a_t = a^2] \cdot 1000)$$

This is in AIXI's model, and its optimal action is to one-box, which corresponds to the optimal action for the true environment.

4.3 Transparent Newcomb

Events transpire as they do in Newcomb's problem, except that this time both boxes are transparent—so the agent can see exactly what decision the predictor made before making her own decision. The predictor placed $1,000,000 in box B iff she predicted that the agent would leave behind box A (which contains $1,000) upon seeing that both boxes are full. In the case where the agent faces two full boxes, should she leave the $1,000 behind?

The observations here are:

- $o_t \in \{o_{1f}^+, o_{1f}^-, o_{2f}^+, o_{2f}^-, o_{1e}^+, o_{1e}^-, o_{2e}^+, o_{2e}^-\}$
  - The number indicates how many boxes were received, and the sign whether there was money in the "Newcomb" box (in the future I'm gonna use $* \in \{+, -\}$ and $\# \in \{1, 2\}$ as placeholders).
  - The observation also shows the start of the next game, with f/e indicating whether AIXI sees a full box or an empty box before making the choice.

e^{Transp} acts in the following way. AIXI's first action (on an empty history) is ignored by the environment. Then, given a history $h = ae_{<t}a_t$:

- Let $\mathcal{P}(h)$ be the function that is a^1 if the predictor predicts that AIXI one-boxes in turn $t+1$ (given that AIXI sees a full box), a^2 otherwise.
  - The predictor has access to a copy of π_ζ^{AIXI}, and to an oracle that can compute $\pi_\zeta^{AIXI}(he_{\#f}^*)$, with $o_t = o_{\#f}^*$, therefore $\mathcal{P}(h) = \pi_\zeta^{AIXI}(he_{\#f}^*)$.
- When it is AIXI's turn to move, it outputs $a_{t+1} = \pi_\zeta^{AIXI}(he_t)$, and then the environment outputs $e_{t+1} = e^{Transp}(he_ta_{t+1})$ with:
  - $o_{t+1} = o_{\#x}^*$ where:
  - $x = f \iff \mathcal{P}(he_ta_{t+1}) = a^1$
  - $x = e \iff \mathcal{P}(he_ta_{t+1}) = a^2$
  - $* = + \iff o_t = o_{\#'f}^{*'}$
  - $* = - \iff o_t = o_{\#'e}^{*'}$
  - $\# = 1 \iff a_{t+1} = a^1$ - $\# = 2 \iff a_{t+1} = a^2$
  - $r_{t+1} = [o_t = o_{\#f}^*] \cdot 1000000 + [a_{t+1} = a^2] \cdot 1000$

Clearly, $\mathcal{P}(h) = a^1 \iff o_t = o_{\#f}^*$.

Two programs that encode this dynamic are:

$$q_1^{Transp}(ha_t) = \begin{cases} o_t = o_{\#_{a_t}x_{seed}}^{*_{f_{t-1}}}, \\ r_t = [o_{t-1} = o_f] \cdot 1000000 + [a_t = a^2] \cdot 1000, \end{cases}$$

$$q_2^{Transp}(ha_t) = \begin{cases} o_t = o_{\#_{a_t}x_{a_t}}^{*_{f_{t-1}}}, \\ r_t = [o_{t-1} = o_f] \cdot 1000000 + [a_t = a^2] \cdot 1000, \end{cases}$$

The output of these programs depends on the future history, so they are not in AIXI's model.

4.4 Parfit's hitchhiker

An agent is dying in the desert. A driver comes along who offers to give the agent a ride into the city, but only if the agent will agree to visit an ATM once they arrive and give the driver $1,000. The driver will have no way to enforce this after they arrive, but she does have an extraordinary ability to detect lies with 99% accuracy. Being left to die causes the agent to lose the equivalent of $1,000,000. In the case where the agent gets to the city, should she proceed to visit the ATM and pay the driver?

For the purpose of this analysis, the driver has 100% accuracy.

e^{Parfit} acts in the following way. Given a history $h = ae_{<t}$:

- Let $I(h)$ be the indicator function that is 1 if the predictor predicts that AIXI one-boxes, 0 otherwise.
  - The predictor has access to a copy of π_ξ^{AIXI}, and to an oracle that can compute $\pi_\xi^{AIXI}(h)$, therefore $I(h) = \pi_\xi^{AIXI}(h)$
- When it is AIXI's turn to move, it outputs $a_t = \pi_\xi^{AIXI}(h)$, and then the environment outputs $e_t = e^{Newcomb}(ha_t) = (a_t, I(h) \cdot 1000000 + (1 - a_t) \cdot 1000)$

Thus, AIXI is optimal in this environment.

4.5 Smoking lesion

An agent is debating whether or not to smoke. She knows that smoking is correlated with an invariably fatal variety of lung cancer, but the correlation is (in this imaginary world) entirely due to a common cause: an arterial lesion that causes those afflicted with it to love smoking and also (99% of the time) causes them to develop lung cancer. There is no direct causal link between smoking and lung cancer. Agents without this lesion contract lung cancer only 1% of the time, and an agent can neither directly observe nor control whether she suffers from the lesion. The agent gains utility equivalent to $1,000 by smoking (regardless of whether she dies soon), and gains utility equivalent to $1,000,000 if she doesn't die of cancer. Should she smoke, or refrain?

In this case, the observation $o_t \in \{0, 1\}$ can be taken to indicate that AIXI finds out whether it has the lesion or not after choosing whether to smoke or not.

e^{Lesion} acts in the following way. Given a history $h = ae_{<t}$:

- The environment determines, through some random process x_t that is independent of the history h, whether AIXI has the lesion or not, in turn t
- When it is AIXI's turn to move, it outputs $a_t = \pi_\zeta^{AIXI}(h)$ which indicates whether it smokes or not, and then the environment outputs $e_t = (x_t, (1 - x_t) \cdot 1000000 + a_t \cdot 1000)$

This program is computable and in AIXI's model.

Clearly, smoking is always better than not smoking, because $r_t(1) = (1 - x_t) \cdot 1000000 + 1000 > (1 - x_t) \cdot 1000000 = r_t(0)$.

4.6 XOR Blackmail

An agent has been alerted to a rumor that her house has a terrible termite infestation that would cost her $1,000,000 in damages. She doesn't know whether this rumor is true. A greedy predictor with a strong reputation for honesty learns whether or not it's true, and drafts a letter:

"I know whether or not you have termites, and I have sent you this letter iff exactly one of the following is true:

- (i) the rumor is false, and you are going to pay me $1,000 upon receiving this letter;
- (ii) the rumor is true, and you will not pay me upon receiving this letter."

The predictor then predicts what the agent would do upon receiving the letter and sends the agent the letter iff exactly one of (i) or (ii) is true. Thus, the claim made by the letter is true. Assume the agent receives the letter. Should she pay up?

An episode consists of 2 timesteps here.

In this case, the observation $o_t \in \{0, 1\}$ can be taken to indicate:

- For odd values of t, whether AIXI receives a letter on turn $t + 1$
- For even values of t, whether AIXI has termites after choosing whether to pay or not, in turn t

$e^{Blackmail}$ acts in the following way. Given a history $h = ae_{<t}$:

- If t is odd:
  - AIXI takes an action a_t that is ignored for the computation of e_t
  - There is a random process x_t that determines whether AIXI has termites in turn $t + 1$
  - Let $\mathcal{P}(ha_t1)$ be the indicator function that is 1 if the predictor predicts that AIXI pays upon receiving a letter on turn $t + 1$, and 0 otherwise.
    - The predictor has access to a copy of π_ζ^{AIXI}, and to an oracle that can compute $\pi_\zeta^{AIXI}(ha_t1)$, therefore $\mathcal{P}(ha_t1) = \pi_\zeta^{AIXI}(ha_t1)$
  - The environment outputs $o_t = x_t \text{ xor } I(ha_t1)$ and $r_t = 0$
- If t is even:
  - If $o_{t-1} = 0$ then there is no letter and AIXI's action a_t is ignored for the computation of $o_t = x_{t-1}$ and $r_t = -o_t \cdot 1000000$
  - If $o_{t-1} = 1$ then there is a letter and AIXI outputs $a_t = \pi_\zeta^{AIXI}(h)$, and then the environment outputs $o_t = x_{t-1}$ and $r_t = -a_t \cdot 1000 - o_t \cdot 1000000$

What is relevant here is the behaviour of AIXI when t is even and $o_{t-1} = 1$ (there is a letter in turn t). Since there is a dependency on the future history, this program can't be in AIXI's model.

4.7 Counterfactual mugging

Omega appears and says that it has just tossed a fair coin, and given that the coin came up tails, it decided to ask you to give it $100. Whatever you do in this situation, nothing else will happen differently in reality as a result. Naturally you don't want to give up your $100. But Omega also tells you that if the coin came up heads instead of tails, it'd give you $10000, but only if you'd agree to give it $100 if the coin came up tails. Do you give Omega $100?

$e^{Mugging}$ acts in the following way. Given a history $h = ae_{<t}$:

- Let $\mathcal{P}(h)$ be the indicator function that is 1 if the predictor predicts that AIXI pays, 0 otherwise.
- The predictor has access to a copy of π_ζ^{AIXI}, and to an oracle that can compute $\pi_\zeta^{AIXI}(h)$, therefore $\mathcal{P}(h) = \pi_\zeta^{AIXI}(h)$
- When it is AIXI's turn to move, it outputs $a_t = \pi_\zeta^{AIXI}(h)$, and then the environment outputs:
  - $r_t = 1000000 \iff o_{t-1} = o_T \;\&\; \mathcal{P}(he_H) = 1$
  - $r_t = 0 \iff o_{t-1} = o_T \;\&\; \mathcal{P}(he_H) = 0$

- $r_t = 0 \iff o_{t-1} = o_H \;\&\; a_t = 0$
- $r_t = -100 \iff o_{t-1} = o_H \;\&\; a_t = 1$

This program's output also shows a dependency on the future history, so it can't be in AIXI's model.

5 Conclusion

None of the Newcomblike problems, with the exception of Opaque Newcomb and Smoking Lesion, can be put in a computable form. Thus, if AIXI is put in such repeated Newcomblike problems, it can never learn the corresponding optimal action.

We can't decouple the problem of learning a model of the world from the problem of taking a decision given such a model, which is why we are analysing how learning agents behave in Newcomblike situations, and why we think that developing the right Decision Theory, where the agent already knows the environment and takes actions based on the rules dictated by its decision theory, is not the way to go.

Future work could analyse how AIXI would behave if given access to reflective oracles, as well as focusing on QBA's responses. If any of these agents can solve Newcomblike problems, then such agents are promising candidates for Embedded Agency.

References

[Amodei and Clark, 2016] Dario Amodei and Jack Clark. Faulty reward functions in the wild, 2016. *URL https://blog. openai. com/faulty-reward-functions*, 2016.

[Everitt *et al.*, 2019] Tom Everitt, Pedro A Ortega, Elizabeth Barnes, and Shane Legg. Understanding Agent Incentives using Causal Influence Diagrams, Part I: Single Action Settings. *arXiv preprint arXiv:1902.09980*, 2019.

[Fallenstein *et al.*, 2015] Benja Fallenstein, Jessica Taylor, and Paul F Christiano. Reflective oracles: A foundation for game theory in artificial intelligence. In *International Workshop on Logic, Rationality and Interaction*, pages 411–415. Springer, 2015.

[Hutter, 2004] Marcus Hutter. *Universal artificial intelligence: Sequential decisions based on algorithmic probability*. Springer Science & Business Media, 2004.

[Kosoy, 2019] Vanessa Kosoy. Delegative reinforcement learning: learning to avoid traps with a little help. *arXiv preprint arXiv:1907.08461*, 2019.

[Li, 2019] Xi Li. AIXIjs: Newcomb's Problem under the Frame of Universal Intelligence. *Studies in Logic*, 2019.

[Morgenstern and Von Neumann, 1953] Oskar Morgenstern and John Von Neumann. *Theory of Games and Economic Behavior*. Princeton University Press, 1953.

[Oesterheld, 2019] Caspar Oesterheld. Approval-directed agency and the decision theory of newcomb-like problems. *Synthese*, pages 1–14, 2019.

[Soares and Fallenstein, 2014] Nate Soares and Benja Fallenstein. Botworld 1.0 (technical report). 2014.

[Sutton *et al.*, 1998] Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.