Generalised Discount Functions applied to a Monte-Carlo AIμ Implementation

Sean Lamont[1], John Aslanides[1], Jan Leike[2] and Marcus Hutter[1]

[1]Research School of Computer Science, Australian National University. [2]Google Deepmind, London; Future of Humanity Institute, Oxford

Motivation

- General Reinforcement Learning (GRL) : Domain independent Reinforcement Learning agents
- Many theoretical results for GRL, but few examples demonstraring these concretely
- **Our Goal**: Use the platform AIXIjs to experimentally verify theoretical results regarding general discount functions

Key Results

- We have experimentally verified that hyperbolic discounting is time-inconsistent and power discounting causes a growing effective horizon
- We added simulations to the AIXIjs platform which demonstrate these results, and enable future demos incorporating generalised discounting.

Background

Standard model of discounted utility (the goal of RL agents is to maximise this):

$$V_k := \sum_{t=k}^{\infty} \gamma_{t-k} r_t$$

Where γ is the discount function, and r_t is the reward at time t.
Can be extended to include discount functions which can change over time.
This generalised model allows for policies which:

- Are **time inconsistent**, where actions may not always align with previous plans.
- Cause future rewards to become relatively more desirable as time progresses (from a **growing effective horizon**)

The discount functions we investigate are:
Hyperbolic Discounting: $\gamma_t^k = \frac{1}{(1+\kappa(t-k))^\beta}$ Time inconsistent, and models human discounting.
Power Discounting: $\gamma_t^k = t^{-\beta}$ Time consistent, causes a growing effective horizon.

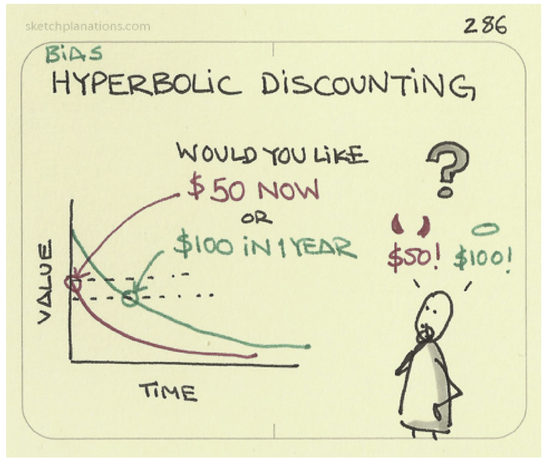

Figure 1: Hyperbolic discounting can explain some irrational human behaviour, such as procrastination and addiction

AIXIjs

- Online, JavaScript based platform showcasing theoretical results from GRL in Gridworld environments.
- **Open Source**, allows researchers to add and modify demos as necessary
- Adapted to include arbitrary discount functions, and to include a simple MDP assessing agent far-sightedness.

AIμ with Monte-Carlo Tree Search

- Informed agent AIμ knows environment dynamics **a priori**
- In combination with the deterministic MDP, ensures any observed change in behaviour is from the discount function, as opposed to any stochasticity in the environment/model.
- Approximate agents expectimax with Monte-Carlo Tree Search
- We derive the number of time inconsistent actions by recording the MCTS plan and comparing this with future actions.

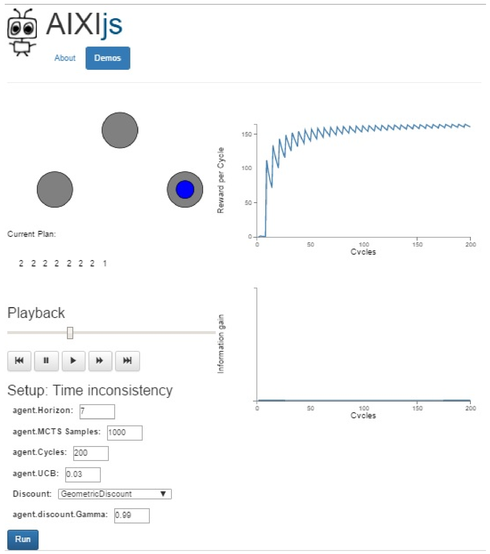

Figure 2: Layout of an AIXIjs web demo

AIXIjs Source Code/ Web Page

- Source: https://github.com/aslanides/aixijs
- Web Page: http://aslanides.io/aixijs/
- Also: http://www.hutter1.net/aixijs/

Environment

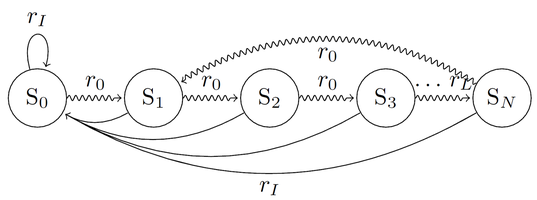

Figure 3: MDP Used for Discounting Experiments

This environment gives the agent 2 actions:

- a_1 (Straight Line): Return a small reward r_I **every** time a_1 is taken
- a_2 (Squiggly Line): Return very large reward $r_L > N r_I$ only if the agent follows a_2 for N **consecutive** steps. Else return 0 reward r_0.

A far sighted agent will ignore the temptation of the low r_I, instead planning ahead to reach the large r_L.

Results: Reward Plot

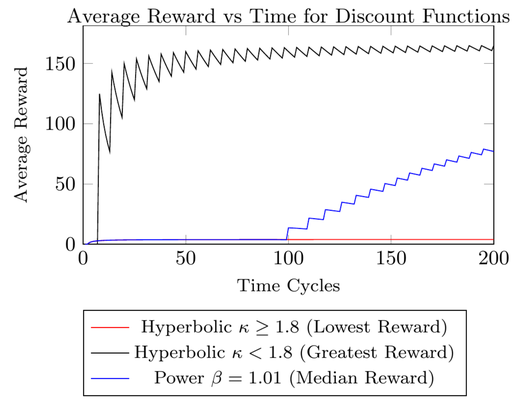

Figure 4: Reward Plot for Discounting Experiments

Summary of Results

- Time consistent results for all trials of power discounting, and for hyperbolic discounting when $\kappa \neq 1.8$
- For $\kappa = 1.8$, the hyperbolic agent was time inconsistent for all time steps, with the plan showing it **procrastinating** the delayed reward.
- From Figure 4, **the power discounting agent changed to a far-sighted policy** around step 100. This is directly caused by the growing effective horizon.

Additional Information

- **Email**: sean.a.lamont@outlook.com
- **Full Paper**: arXiv:1703.01358 [cs.AI]

Australian National University