Can Reinforcement Learning support policy makers? A preliminary study with Integrated Assessment Models

Theodore Wolf
University College London | Carbon Re
theo@carbonre.com

Nantas Nardelli
Carbon Re
nanta@carbonre.com

John Shawe-Taylor
University College London
j.shawe-taylor@ucl.ac.uk

María Pérez Ortiz
University College London
maria.perez@ucl.ac.uk

Abstract

Governments around the world aspire to ground decision-making on evidence. Many of the foundations of policy making — e.g. sensing patterns that relate to societal needs, developing evidence-based programs, forecasting potential outcomes of policy changes, and monitoring effectiveness of policy programs — have the potential to benefit from the use of large-scale datasets or simulations together with intelligent algorithms. These could, if designed and deployed in a way that is well grounded on scientific evidence, enable a more comprehensive, faster, and rigorous approach to policy making. Integrated Assessment Models (IAM) is a broad umbrella covering scientific models that attempt to link main features of society and economy with the biosphere into one modelling framework. At present, these systems are probed by policy makers and advisory groups in a hypothesis-driven manner. In this paper, we empirically demonstrate that modern Reinforcement Learning can be used to probe IAMs and explore the space of solutions in a more principled manner. While the implication of our results are modest since the environment is simplistic, we believe that this is a stepping stone towards more ambitious use cases, which could allow for effective exploration of policies and understanding of their consequences and limitations.

1 Introduction

Climate is a high dimensional dynamical system with strong inter-dependent components and long time dependencies, all of which interact to produce highly non-linear responses and behavior. Climate is also highly conditioned on human behavior – another greatly complex system – such that is now necessary to reason about climate change from a socio-climatic perspective [Moore et al., 2022]. To make progress towards achieving some kind of solution in the face of extreme consequences, policymakers and advisory groups employ *Integrated Assessment Models* (IAMs), state of the art models for climate change that combine knowledge about human development (such as economical theories) together with planetary sciences such as ecology and geophysics [Parson and Fisher-Vanden, 1997]. Exploring and analysing the properties of the IAMs employed for large scale assessments [Pörtner et al., 2022] - to e.g. measure fidelity against the real world – is generally intractable from a computational perspective, which leads researchers to implement poor simplifying assumptions and decrease their effectiveness [Asefi-Najafabady et al., 2021]. Smaller IAM models aim to provide an alternative by employing fewer state variables and simpler sets of dynamics, making them amenable to mathematical probing and analysis [Kittel et al., 2017, Nitzbon et al., 2017]. The

Tackling Climate Change with Machine Learning: workshop at NeurIPS 2023.

literature commonly explores these models with ODE solvers; however recently [Strnad et al., 2019] has shown that it is possible to employ them as environments in standard Reinforcement Learning (RL) [Sutton and Barto, 2020], and explore models using trained policies. These can be used to understand the system, and provide upwards insight towards improving more complex IAMs or even our understanding of climate change policies.

We build on this work, testing more RL algorithms, reward functions, as well as different experimental setups. Among others, we show that (a) modern RL can learn effective policies with a variety of reward functions in this environment (b) that different agents and reward functions generate a significantly diverse set of solutions, thus exploring the IAM in different manners, and (c) that it is necessary to apply care when designing reward functions as they show different success rate in reaching the desirable state for different initialisation points, finally - (d) that RL helps us gain a deeper understanding of the properties and limitations of the applied models.

2 AYS environment, RL reward functions and agents

We employ the *AYS model* [Kittel et al., 2017], which we use to create a Markov Decision Process following Strnad et al. [2019]. AYS is governed by three coupled differential equations:

$$\frac{dA}{dt} = \frac{1}{1+(S/\sigma)^\rho}\frac{Y}{\phi\epsilon} - A/\tau_A, \quad \frac{dY}{dt} = \beta Y - \theta AY, \quad \frac{dS}{dt} = \left(1 - \frac{1}{1+(S/\sigma)^\rho}\right)\frac{Y}{\epsilon} - S/\tau_S, \quad (1)$$

where A is the excess atmospheric carbon, Y society's economic output, and S the renewable knowledge stock. At each step t, the agent observes a vector (A_t, Y_t, S_t), which is a partial representation of the state of the system. This dynamical system contains two attractors, a *green fixed point*, where renewable knowledge and economic output grow forever, and a *black fixed point*, where the economic output is stagnant and there is a large amount of excess carbon in the atmosphere (Figure 1). Broadly speaking, the former is a positive end state (and effectively encodes the high level goal), while the latter is undesirable. See Appendix A for a more detailed technical aspects of the environment. The agent can take four actions, each corresponding to a high level government policy decision and changes to the dynamics of the environment (See Table 1 for a mathematical and textual description of the actions).

Reward functions The reward signal that agents receives from taking actions in this environment is based on *planetary boundaries* (PBs), quantitative physical and economical limits which, if crossed, would represent disastrous and irreversible consequences for the biosphere and humans [Rockström et al., 2009, Steffen et al., 2015]. We define these boundaries as $A_{PB} = 600$, $Y_{PB} = 4 \times 10^{13}$, and $S_{PB} = 0$, which form the triplet s_{PB}. Crossing one of these boundaries yields a reward of 0 and ends the episode. Furthermore, in the standard scenario, the agent is rewarded for staying away from these boundaries, i.e. $R_t^{PB} = ||s_t - s_{PB}||^2$. This incentives the agent to maximise economic output while minimising carbon emissions. We call this reward function *PB reward*. A second reward function we employ is the *Policy Cost (PC) reward*, which adds an action-dependent cost to PB reward, simulating the real world cost of implementing and maintaining any significant shift in policy in a running socio-economical system. E.g., throttling growth over years may be challenging for a policymaker [Keyßer and Lenzen, 2021]. Finally, our experiments see the use of a third and simpler reward function (*Sparse reward*, since it significantly lowers the amount of feedback

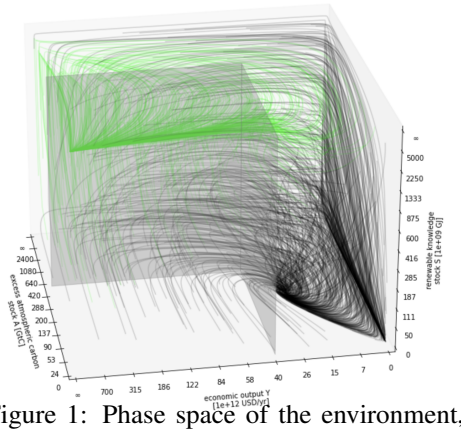


Figure 1: Phase space of the environment, each dimension corresponds to one of the states dimensions. Hair lines show the flows of the system.

that the agent receives on average), which only considers whether the agent reaches the goal or hits any of the planetary boundaries:

$$r(s) = \begin{cases} 1 & \text{if} \quad s_t = s_g \\ -1 & \text{if} \quad (A_t > A_{PB}) \vee (Y_t < Y_{PB}) \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Table 1: Environment actions, and how they relate to the policy cost (PC) reward function.

Action	Parameter Change	Explanation	PC reward
Noop	None	Environment evolves with default parameters.	R_t^{PB}
ET	$\sigma \leftarrow \sigma/\sqrt{\rho}$	Halves relative cost of renewables to fossil fuels.	$0.5 \times R_t^{PB}$
DG	$\beta \leftarrow \beta/2$	Halves rate of growth of the economy.	$0.5 \times R_t^{PB}$
ET+DG	$\beta \leftarrow \beta/2, \sigma \leftarrow \sigma/\sqrt{\rho}$	Combination of the two above actions.	$0.25 \times R_t^{PB}$

Agent settings We are primarily interested in the interplay of RL and AYS. To do so, we learn a diverse set of agents through four different learning algorithms: A2C [Mnih et al., 2016], PPO [Schulman et al., 2017], DQN [Mnih et al., 2015], and Double DQN with dueling networks and prioritized experience replay (D3QN) [Hessel et al., 2017]. All agents employ the same network architecture, bar the final layer, which is dependent on the number and type of heads required by the algorithm. See Appendix B for further details on the network architecture. Agents are trained for 5e5 steps, and tuned separately for best cumulative performance using Bayesian optimization. All experiments show results of 3 random seeds.

3 Results and discussion

Figure 2 shows how most agents are generally able to learn a policy that optimizes the relevant reward function. Interestingly, while DQN and D3QN are broadly consistent across reward functions, A2C struggles to learn policies under the same episodic budget, even under a substantial amount of hyperparameter tuning. PPO on the other hand beats all agents on *PB reward*, and underperforms everywhere else.

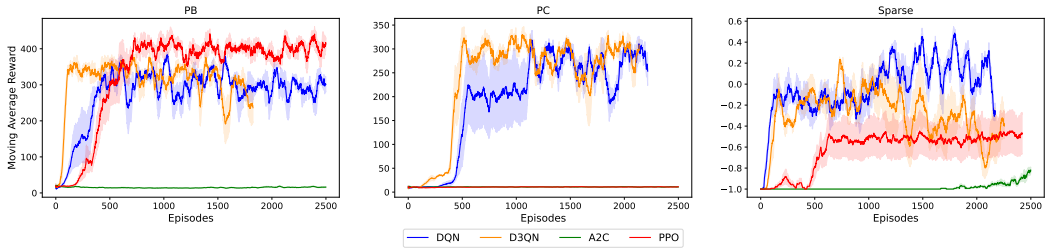


Figure 2: Average cumulative training rewards for each agent by reward function.

We hypothesize these differences may be related to the use of experience replay for the DQN agents, as well as the action sampling method in the case of A2C and PPO. We will later see that a robust class of successful policies in this environment corresponds to getting as quickly as possible to a spot under a green "current", and letting the system converge to the goal. In such scenario, the action distribution for large subsets of the state space needs to converge onto specific actions (such as *noop* in this particular case). This is a straightforward outcome for DQN-based agents, however as PPO relies on softmax parametrization to output actions (and learn) in a discrete space, it becomes more difficult for the agent to converge to such a policy given its dependency on both entropy regularization and Boltzmann-based exploration [Ahmed et al., 2019, Mei et al., 2020].

Figure 3 shows that agents converge to significantly different policies under different learning algorithms and reward functions. We note that for a significant part of the initial state space, the first few actions for all the agents that reach the green point are generally similar.

In Figure 4, we see that agents that reach the green point prefer picking the action *ET+DG* during the first 23 steps of the episode, and then go on to exhibiting more variety later in the episode. We notice the significant difference between the distribution of actions for the different reward functions. As expected under the PC reward, the agent maximises the number of *noop* actions taken in order to collect the highest reward. What is important to note is that the agent prefers taking on the cost of

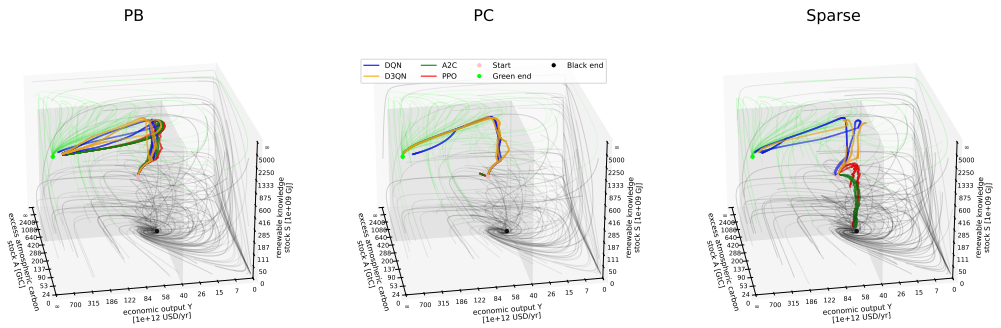


Figure 3: Sample trajectories of the four agents initialized from a fixed state with 3 reward functions. The PB reward function helps the agents solve the environment more consistently.

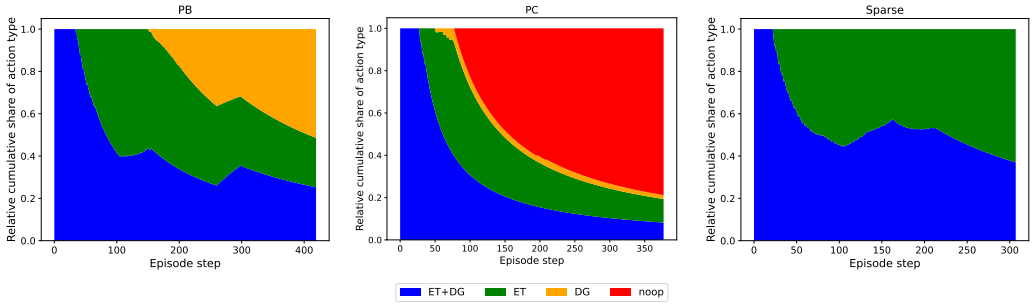


Figure 4: Sample cumulative relative share of actions taken by DQN agents trained with different reward functions.

taking actions in the beginning of the episode. More generally, we see that the distribution of the first 50 actions taken are the same across all reward functions (75% *DG+ET* and 25% *ET*).

This implies that the environment has some form of natural bottleneck, which can also explain the loss in general performance for some of the agents in the sparse reward case, since agents don't have a state or dynamics-driven exploration system [Burda et al., 2018]. Ultimately, we see that the different reward functions yield qualitative changes in the resulting trained policy of the agents, with a generally higher difference for on-policy agents.

Figure 5 shows how agents solve the environment from different initial states when trained with different reward functions. The experiments suggest that policies are indeed highly dependent on the employed learning algorithm, but we also see commonalities across reward functions, particularly when looking at initial states. If the agents are initialized in the top-left of the starting grid, is is much easier for them to reach the goal. On the other hand if they start in the bottom-right corner, they seem guaranteed to fail. Therefore when initialising a new episode, not all starts are equal in outcome. We hypothesize that a strictly optimal policy may not be able to do better, but further work is necessary to establish the exact failure conditions. We note that under this model, we can explain this particular result by looking at the modeled dynamics of real the environment: the economic output grows exponentially (as long as atmospheric carbon is low) and needs exponentially more energy to then sustain itself. The underlying equations dictate that if not enough renewable knowledge stock is available, then this energy will come from fossil fuels. This consequently increases the

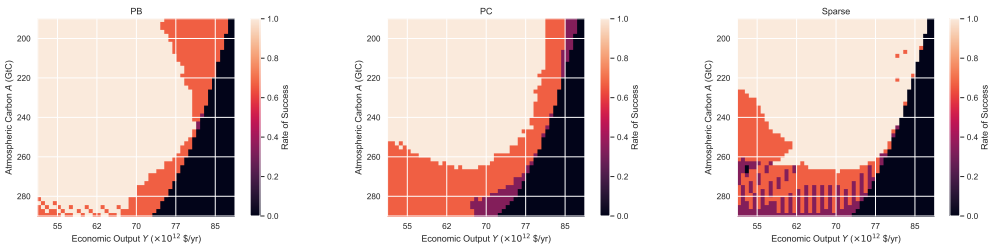


Figure 5: Average success rate of three seeds of the trained DQN agent for different reward functions given different initialisation states. Success is defined as reaching the green point.

excess atmospheric carbon which accumulates very fast and causes the agent to cross the atmospheric carbon boundary. Such emergent behavior is also not immediately obvious when considering the system equations.

Conclusions The question of whether algorithms could support policy making is of current interest, where more than ever we find governments having to make high-stake decisions with regards to fast changing complex, global and interconnected challenges, which are difficult to understand and tackle without relevant datasets, scientific evidence and scenario-analysis tools. Public policy-making is often a cyclical process, with stages such as identification of societal needs, formulation of agendas, scenarios and policy alternatives, adoption of policy decisions, implementation in the real world, and finally, evaluation of their effectiveness, with subsequent improvements. Our experiments aim to understand whether RL can be used to formulate policy alternatives and evaluate their effectiveness in a simulation environment. Specifically, we have shown that a variety of RL algorithms produce well-behaved policies in the AYS environment under different, more or less sparse, reward functions. The combination of different reward functions, RL algorithms, and the space of initial states produce diverse policies that can successfully explore and solve the underlying AYS model. This enables the study of the emergent properties of the system without needing to encode much knowledge about the model into the learning process. This is an interesting result, as it shows the potential of applying RL as a general debugging and analysis toolkit for IAM models.

We note that the solution to solving the AYS model relates to *early action* [MacCracken, 2008], as the agent solves the environment more consistently when implementing the more aggressive policy position early on in the episode. Early action dictates that fast and aggressive climate change mitigation has cumulative benefits. The concept of *early action* stems from the *time value of carbon* [Cornelis van Kooten et al., 2021]. Future work should look into exploring whether agents can be trained across multiple model-environments, to understand whether some kind of "common exploration strategy" emerges as a result, or whether agents could be trained to explore small, simplified models, and behave in a reasonable manner in computationally bigger, and thus more expensive IAMs. Appendix D presents further experiments, conclusions and future work.

References

Frances C. Moore, Katherine Lacasse, Katharine J. Mach, Yoon Ah Shin, Louis J. Gross, and Brian Beckage. Determinants of emissions pathways in the coupled climate–social system. *Nature 2022 603:7899*, 603(7899):103–111, 2 2022. ISSN 1476-4687. doi: 10.1038/s41586-022-04423-8. URL `https://www.nature.com/articles/s41586-022-04423-8`.

Edward A Parson and Karen Fisher-Vanden. Integrated assessment models of global climate change. *Annual Review of Energy and the Environment*, 22(1):589–628, 1997.

Hans-O Pörtner, Debra C Roberts, Helen Adams, Carolina Adler, Paulina Aldunce, Elham Ali, Rawshan Ara Begum, Richard Betts, Rachel Bezner Kerr, Robbert Biesbroek, et al. *Climate change 2022: Impacts, adaptation and vulnerability*. IPCC Geneva, Switzerland:, 2022.

Salvi Asefi-Najafabady, Laura Villegas-Ortiz, and Jamie Morgan. The failure of integrated assessment models as a response to 'climate emergency'and ecological breakdown: the emperor has no clothes. *Globalizations*, 18(7):1178–1188, 2021.

Tim Kittel, Finn Müller-Hansen, Rebekka Koch, Jobst Heitzig, Guillaume Deffuant, Jean Denis Mathias, and Jürgen Kurths. From lakes and glades to viability algorithms: Automatic classification of system states according to the Topology of Sustainable Management. *European Physical Journal: Special Topics*, 230(14-15):3133–3152, 6 2017. ISSN 19516401. doi: 10.48550/arxiv.1706.04542. URL `https://arxiv.org/abs/1706.04542v4`.

Jan Nitzbon, Jobst Heitzig, and Ulrich Parlitz. Sustainability, collapse and oscillations in a simple World-Earth model. *Environmental Research Letters*, 12(7): 074020, 7 2017. ISSN 1748-9326. doi: 10.1088/1748-9326/AA7581. URL `https://iopscience.iop.org/article/10.1088/1748-9326/aa7581https://iopscience.iop.org/article/10.1088/1748-9326/aa7581/meta`.

Felix M. Strnad, Wolfram Barfuss, Jonathan F. Donges, and Jobst Heitzig. Deep reinforcement learning in World-Earth system models to discover sustainable management strategies. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 29(12):123122, 12 2019. ISSN 1054-1500. doi: 10.1063/1.5124673. URL https://aip.scitation.org/doi/abs/10.1063/1.5124673.

Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. 2 edition, 2020. URL http://incompleteideas.net/book/the-book.html.

Johan Rockström, Will Steffen, Kevin Noone, Asa Persson, F. Stuart Chapin, Eric F. Lambin, Timothy M. Lenton, Marten Scheffer, Carl Folke, Hans Joachim Schellnhuber, Björn Nykvist, Cynthia A. De Wit, Terry Hughes, Sander Van Der Leeuw, Henning Rodhe, Sverker Sörlin, Peter K. Snyder, Robert Costanza, Uno Svedin, Malin Falkenmark, Louise Karlberg, Robert W. Corell, Victoria J. Fabry, James Hansen, Brian Walker, Diana Liverman, Katherine Richardson, Paul Crutzen, and Jonathan A. Foley. A safe operating space for humanity. *Nature 2009 461:7263*, 461 (7263):472–475, 9 2009. ISSN 1476-4687. doi: 10.1038/461472a. URL https://www.nature.com/articles/461472a.

Will Steffen, Katherine Richardson, Johan Rockström, Sarah E. Cornell, Ingo Fetzer, Elena M. Bennett, Reinette Biggs, Stephen R. Carpenter, Wim De Vries, Cynthia A. De Wit, Carl Folke, Dieter Gerten, Jens Heinke, Georgina M. Mace, Linn M. Persson, Veerabhadran Ramanathan, Belinda Reyers, and Sverker Sörlin. Planetary boundaries: Guiding human development on a changing planet. *Science*, 347(6223), 2 2015. ISSN 10959203. doi: 10.1126/SCIENCE.1259855. URL http://dx.doi.

Lorenz T Keyßer and Manfred Lenzen. 1.5 c degrowth scenarios suggest the need for new mitigation pathways. *Nature communications*, 12(1):2676, 2021.

Volodymyr Mnih, Adria Puigdomenech Badia, Lehdi Mirza, Alex Graves, Tim Harley, Timothy P. Lillicrap, David Silver, and Koray Kavukcuoglu. Asynchronous Methods for Deep Reinforcement Learning. *33rd International Conference on Machine Learning, ICML 2016*, 4:2850–2869, 2 2016. URL https://arxiv.org/abs/1602.01783v2.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *ArXiv*, abs/1707.06347, 2017.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature 2015 518:7540*, 518(7540):529–533, 2 2015. ISSN 1476-4687. doi: 10.1038/nature14236. URL https://www.nature.com/articles/nature14236.

Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining Improvements in Deep Reinforcement Learning. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 3215–3222, 10 2017. ISSN 2159-5399. doi: 10.48550/arxiv.1710.02298. URL https://arxiv.org/abs/1710.02298v1.

Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. Understanding the impact of entropy on policy optimization. In *International conference on machine learning*, pages 151–160. PMLR, 2019.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning*, pages 6820–6829. PMLR, 2020.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Michael C. MacCracken. Prospects for future climate change and the reasons for early action. *Journal of the Air & Waste Management Association*, 58(6):735–786, 2008. doi: 10.3155/1047-3289.58.6.735. URL https://doi.org/10.3155/1047-3289.58.6.735.

G. Cornelis van Kooten, Patrick Withey, and Craig M.T. Johnston. Climate urgency and the timing of carbon fluxes. *Biomass and Bioenergy*, 151:106162, 2021. ISSN 0961-9534. doi: https://doi.org/10.1016/j.biombioe.2021.106162. URL https://www.sciencedirect.com/science/article/pii/S0961953421001987.

Thomas M. Bury, Chris T. Bauchid, and Madhur Anand. Charting pathways to climate change mitigation in a coupled socio-climate model. *PLOS Computational Biology*, 15(6):e1007000, 6 2019. ISSN 1553-7358. doi: 10.1371/JOURNAL.PCBI.1007000. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1007000.

Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Van Hasselt, Marc Lanctot, and Nando De Frcitas. Dueling Network Architectures for Deep Reinforcement Learning. *33rd International Conference on Machine Learning, ICML 2016*, 4:2939–2947, 11 2015. doi: 10.48550/arxiv.1511.06581. URL https://arxiv.org/abs/1511.06581v3.

Scott M Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In I Guyon, U Von Luxburg, S Bengio, H Wallach, R Fergus, S Vishwanathan, and R Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

Zoe Juozapaitis, Anurag Koul, Alan Fern, Martin Erwig, and Finale Doshi-Velez. Explainable Reinforcement Learning via Reward Decomposition. 2019. URL https://web.engr.oregonstate.edu/~erwig/papers/ExplainableRL_XAI19.pdf.

Jasper van der Waa, Jurriaan van Diggelen, Karel van den Bosch, and Mark Neerincx. Contrastive Explanations for Reinforcement Learning in terms of Expected Consequences. 7 2018. doi: 10.48550/arxiv.1807.08706. URL https://arxiv.org/abs/1807.08706v1.

A A broader look at the AYS environment

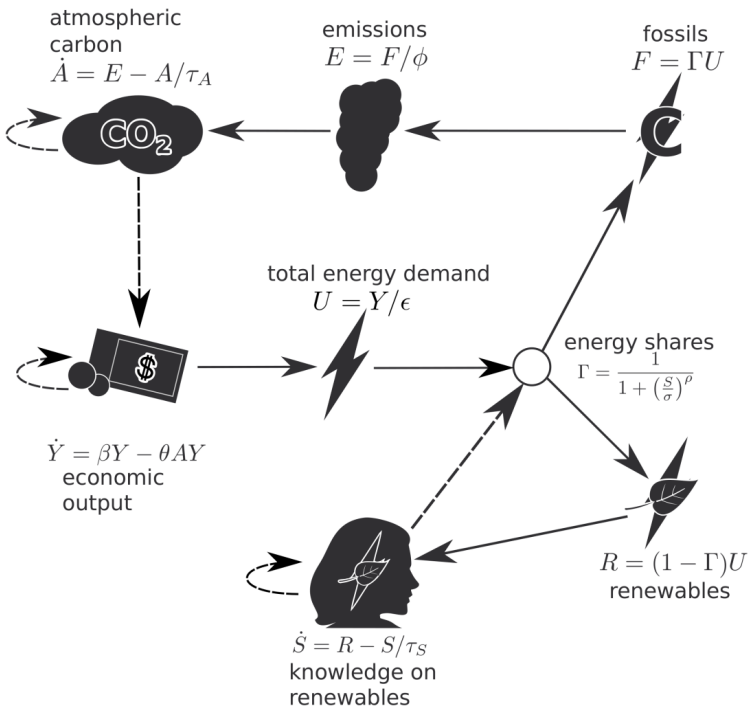


Figure 6: Schematic from Kittel et al. [2017] summarizing the interactions in the AYS model. The full lines are positive interactions and the dotted lines are negative interactions.

A.1 Observables

The model has three observed variables:

- Excess atmospheric carbon, A, in Gigaton of Carbon (GtC).
- Economic output, Y, in US dollars per year (\$/yr).
- Renewable knowledge stock, S, in GigaJoules (GJ).

This low dimensional environment enables us to test our framework's limits as well as plot trajectories in phase space for tractable interpretability and analysis. Furthermore, the dynamics of model are also kept relatively simple compared to other more complex models Nitzbon et al. [2017], Bury et al. [2019], Moore et al. [2022]. We note that out of all three variables, the last one is less "tangible" than other ones, as it represents some knowledge metric humans have about renewable energy; to make it relevant to available quantitative data, this model chooses to represent it as energy.

A.2 Equations

The system is governed by three differential equations, one for each observed variable:

$$\frac{dA}{dt} = E - A/\tau_A, \tag{3}$$

$$\frac{dY}{dt} = \beta Y - \theta AY, \tag{4}$$

$$\frac{dS}{dt} = R - S/\tau_S. \tag{5}$$

With R and E the energy extracted from renewables and fossil fuels respectively. These are defined with the energy demand U in GJ/year, which is proportional to the economic output:

$$U = \frac{Y}{\epsilon}, \tag{6}$$

where ϵ is the efficiency of energy. Energy is either produced from renewable sources or from fossil fuel sources:

$$R = (1 - \Gamma)U, \tag{7}$$
$$F = \Gamma U, \tag{8}$$
$$E = F/\phi. \tag{9}$$

Here, ϕ is the fossil fuel combustion efficiency in GJ/GtC. The share of fossil fuel energy Γ is calculated as an inverse response to the renewable knowledge:

$$\Gamma = \frac{1}{1 + (S/\sigma)^{\rho}}. \tag{10}$$

With σ being the break-even knowledge, which corresponds to the state where renewable and fossil fuel costs become equal, and ρ is the renewable knowledge learning rate. As knowledge on renewables (S) increases, $\Gamma \to 0$ and the total energy share produced by renewables increases. If $S \to 0$, then $\Gamma \to 1$ and more energy is produced from fossil fuels. The interactions are summarized in Figure 6. The parameter values are summarized in Table 2.

Table 2: Table summarising the parameters of the AYS model from Kittel et al. [2017].

Parameter	**Value**	**Description**
τ_A	50 years	Carbon decay out of the atmosphere.
τ_S	50 years	Decay of renewable knowledge stock.
β	3 %/year	Economic output growth.
σ	4×10^{12} GJ	Break-even knowledge: knowledge at which fossil fuel and renewables have equal cost.
ϕ	4.7×10^{10} GJ/GtC	Fossil fuel combustion efficiency.
ϵ	147 \$/GJ	Energy efficiency parameter.
θ	8.57×10^{-5}	Temperature sensitivity parameter.
ρ	2	Learning rate of renewable knowledge.

A.3 States

The initial state is:

$$s_{t=0} = \begin{pmatrix} 240\,GtC \\ 7 \times 10^{13}\,\$/yr \\ 5 \times 10^{11}\,GJ \end{pmatrix}, \tag{11}$$

which aims to represent the current state of the Earth in this model. There are two attractors in this model,

$$s_b = \begin{pmatrix} \beta/\theta \\ \frac{\phi\beta\epsilon}{\theta\tau_A} \\ 0 \end{pmatrix} = \begin{pmatrix} 350\,GtC \\ 4.84 \times 10^{13}\,\$/yr \\ 0\,GJ \end{pmatrix}. \tag{12}$$

This is denoted as the *black fixed point*: roughly half of the current economic production, and this economic production is stagnant. Furthermore, in the black fixed point, there is an excess of 350 GtC in the atmosphere with no renewable energy production. The other point we are interested in is located at the boundaries of the state space,

$$s_g = \begin{pmatrix} 0 \\ +\infty \\ +\infty \end{pmatrix}, \tag{13}$$

where economic growth and renewable energy knowledge grow forever. We label this the *green fixed point*, and we note that it corresponds to the ideal scenario.

The dynamics of this environment do not allow for more fixed points. Therefore, any point in the space will be naturally drawn to one of these fixed points.

Just as in Kittel et al. [2017], we normalize the state variables A, Y and S between 0 and 1. This prevents any numerical issues from arising in unexpected ways. The normalization scheme employed is the following:

$$\bar{s_t} = \frac{s_t}{s_t + s_{t=0}}. \tag{14}$$

This leads to the initial state being $(0.5, 0.5, 0.5)^T$ and the green fixed point to be $s_g = (0, 1, 1)^T$.

A.4 Episode Description

The AYS model is a deterministic environment. The enable non-trivial learning, we initialize each new episode n a random state sampled under a fixed distribution:

$$s_{t=0} = \begin{pmatrix} 0.5 + \mathcal{U}(-0.05, 0.05) \\ 0.5 + \mathcal{U}(-0.05, 0.05) \\ 0.5 \end{pmatrix}, \tag{15}$$

where $\mathcal{U}$ is the uniform distribution. We do not add noise the third varialbe, as we notice that it dramatically reduces the ability for the agent to learn Strnad et al. [2019].

Each step corresponds to a difference of 1 year. The environment uses an Ordinary Differential Equation numerical solver to calculate the state for the next step given the action-specific parameters.

B Neural Network Architecture

We attempt to normalize the network architecture across all our experiments (and agents). The "torso" of the policy corresponds to three linear layers intercalated with ReLU activation functions. The input layer has three units, the hidden layer has 256 units and the output layer has four units. In the case of D3QN, we use a dueling network [Wang et al., 2015] such that there are two output layers connected to the hidden layer: one with four units and one with a single unit.

C Action distribution by reward function

We observe how different reward functions produce significantly different types of trajectories (Figure 5).

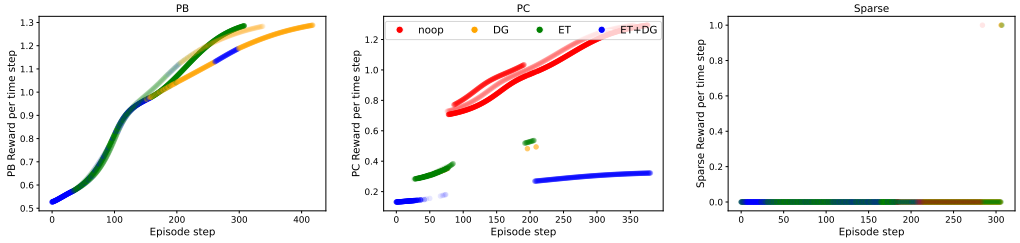


Figure 7: Reward obtained and action taken per timestep of different sucessful DQN agents trained with different reward functions.

This shows that there are many qualitatively different pathways towards achieving the goal in the AYS environment, and that reward signals can easily be used as a way to embed structure into policy space. However, finding a strategy for precisely tuning the trajectories such that they may "evolve" in some specific manner is still an open problem, and we believe it to be robust. Nonetheless, the effect is noticeable across all our experiments.

D Further experiments and conclusions

In our experiments we observed significant issues with sensitivity to hyperparameters, which were very difficult to tune. The off-policy agents were much more flexible with learning the environment in different experiments, which is clear from their consistency across the experiments. The on-policy agents were lacking in exploration, which significantly hurt their performance when using the cost reward function.

Comparing AYS to noisy AYS environment We now test injecting Gaussian noise to the parameters of the environment at each new episode. Each new episode is then slightly different. This is significant, as in dynamical systems small changes early on can radically change the outcome, known in mathematics as *chaos*. This tests the agents' robustness to different environment parameters to emulate the fact that such real-world parameters are never perfectly known. The PPO and A2C agents struggle significantly more in this environment but the DQN-based agents are relatively unaffected by the introduction of noise. This is promising as it shows that off-policy agents can learn in this noisy environment. This brings them one step closer to real-world application.

Figure 8: Moving average and standard deviation reward of the agents in the *Noisy* AYS environment. The noise standard deviation is set at 10^{-3}.

Comparing fully vs partially observable environments We also test making the environment fully observable to the agent by giving the velocities of the variables as observable features. This enables us to show whether a partially observed environment is a significant hurdle to learning, as the real world is almost certainly only partially observed. We find that some of the agents can still learn equivalently well in a partially observed environment if fine-tuned to it (the agents' hyperparameters were optimised for the partially observed environment). We can also contrast how the agent leverages the information from each observable in both environments by analysing the trained neural network parameters. We use SHAP values [Lundberg and Lee, 2017] as a proxy for feature importance, see Figure 10b below. We see that that the state velocity is used more by the agent to infer expected reward.

Future extensions There are many extensions that have not been explored in this work. Changes that were not looked at were changes in the number of actions that can be taken per year, we set this to one throughout this work, but there is no particular reason for this, apart from the easily interpretable idea of one policy per year. In this work, we focused more on the interpretable aspect and thus aimed

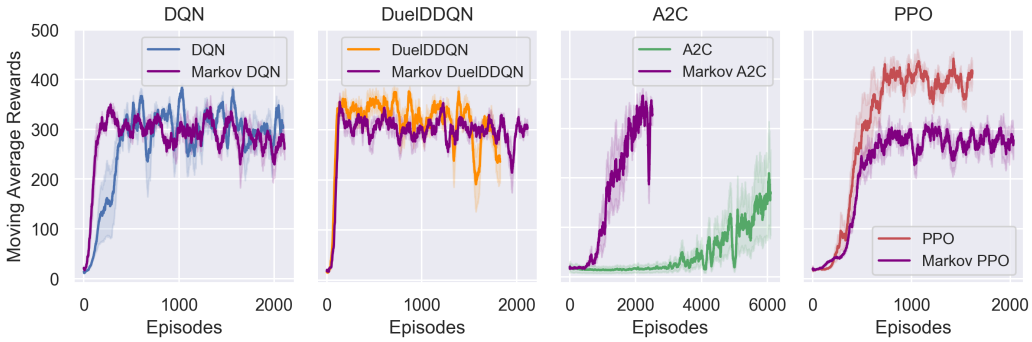
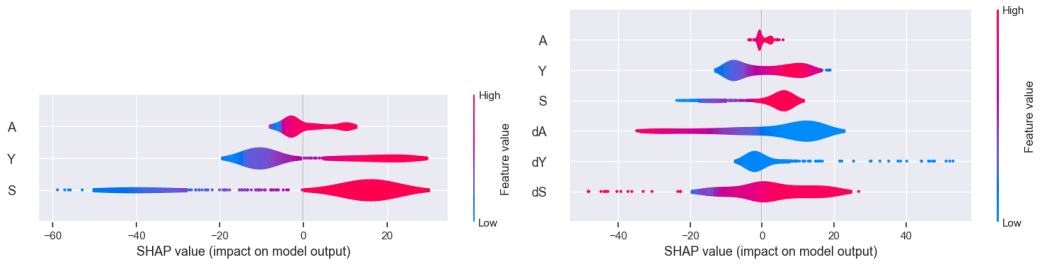


Figure 9: Moving average and standard deviation reward of the four agents with a partially observed environment and with a fully observed one (labelled *Markov*), all trained over 500000 time steps. It is worth noting that the hyperparameters were not re-tuned for the fully observable environment such that we accurately compare identical agents in different environments.



(a) D3QN

(b) Markov D3QN

Figure 10: SHAP values of two D3QN agents, one trained in the partially observed environment and one trained in the fully observed environment. These plots use 500 states randomly sampled from the replay buffer after training.

to leave the fundamental dynamics of the model from Kittel et al. [2017] untouched. Additional actions or continuous actions are a clear avenue for probing the environment in different ways. There is also research in *Explainable Artificial Intelligence* (XAI) that could be integrated in this framework, specifically: explainable RL [Juozapaitis et al., 2019, van der Waa et al., 2018]. This would help with explainability of the agents and interpreting their decisions. Multi-agent RL may also be promising, simulating different drivers of different nations through differentiated reward functions.