Active World Model Learning with Progress Curiosity

Kuno Kim[1], Megumi Sano[1], Julian De Freitas[2], Nick Haber[3*], and Daniel Yamins[1, 4, 5*]

[1]Department of Computer Science, Stanford University [2]Department of Psychology, Harvard University
[3]Stanford Graduate School of Education [4]Department of Psychology, Stanford University
[5]Wu Tsai Neuroscience Institute, Stanford University [*]Equal contribution.

World models are self-supervised predictive models of how the world evolves. Humans learn world models by curiously exploring their environment, in the process acquiring compact abstractions of high bandwidth sensory inputs, the ability to plan across long temporal horizons, and an understanding of the behavioral patterns of other agents. In this work, we study how to design such a curiosity-driven Active World Model Learning (AWML) system. To do so, we construct a curious agent building world models while visually exploring a 3D physical environment rich with distillations of representative real-world agents. We propose an AWML system driven by γ-Progress: a scalable and effective learning progress-based curiosity signal. We show that γ-Progress naturally gives rise to an exploration policy that directs attention to complex but learnable dynamics in a balanced manner, thus overcoming the "white noise problem". As a result, our γ-Progress-driven controller achieves significantly higher AWML performance than baseline controllers equipped with state-of-the-art exploration strategies such as Random Network Distillation and Model Disagreement.

1 Introduction

Imagine yourself as an infant in your parent's arms, sitting on a playground bench. You are surrounded by a variety of potentially interesting stimuli, from the constantly whirring merry-go-round, to the wildly rustling leaves, to your parent's smiling and cooing face. After briefly staring at the motionless ball, you grow bored. You consider the merry-go-round a bit more seriously, but its periodic motion is ultimately too predictable to keep your attention long. The leaves are quite entertaining, but after watching their random motions for a while, your gaze lands on your parent. Here you find something really interesting: you can anticipate, elicit, and guide your parents' changes in expression as you both engage in a game of peekaboo. Though just an infant, you have efficiently explored and interacted with the environment, in the process gaining strong intuitions about how different things in your world will behave.

The infant appears to have learned a powerful *world model* — a predictor of how the world evolves over time, due both to external physical dynamics and to the infant's actions. Such world models help enable humans to plan across long temporal horizons and to anticipate the behavioral patterns of other agents. They also may play an important role in the self-supervised learning of the high-bandwidth sensory systems that produce compact perceptual abstractions underlying cognition and decision making. Devising algorithms that can efficiently construct such world models is an important goal for the next generation of socially-integrated AI and robotic systems.

A slightly abridged version of this paper was published in the *Proceedings of the 37th International Conference on Machine Learning* (ICML 2020).

A key challenge in world model learning is that real-world environments contain a diverse range of dynamics, generated by a multiplicity of objects and other agents, with varying levels of learnability. The inanimate ball and periodic merry-go-round display dynamics that are easy to learn. On the other hand, stimuli such as falling leaves exhibit unlearnable noise-like dynamics. Lying in a "sweet spot" on the learnability spectrum are animate agents that generate interesting and complex yet rule-driven dynamics, e.g. your parent's expressions and play offerings. Balancing attention to maximize learning progress amidst the blooming and buzzing sea of stimuli is a substantial challenge. Particularly difficult is the *white noise problem* [Schmidhuber, 2010, Burda et al., 2018b, Pathak et al., 2019], i.e. perseverating on unlearnable stimuli rather than pursuing learnable dynamics. Thus, it is a natural hypothesis that behind the infant's ability to learn powerful world models must be an equally powerful *active learning* algorithm that directs its attention to maximize learning progress.

In this work, we formalize and study Active World Model Learning (AWML) – the problem of determining a directed exploration policy that enables efficient construction of better world models in agent-rich contexts. To do so, we construct a progress-driven curious neural agent performing AWML in a custom-built 3D virtual world environment. Specifically, our contributions are as follows:

1. We construct a 3D virtual environment rich with agents displaying a wide spectrum of realistic stimuli behavior types with varying levels of learnability, such as static, periodic, noise, peekaboo, chasing, and mimicry.

2. We formalize AWML within a general reinforcement learning framework that encompasses curiosity-driven exploration and traditional active learning.

3. We propose an AWML system driven by γ-Progress: a novel and scalable learning progress-based curiosity signal. We show that γ-Progress gives rise to an exploration policy that overcomes the white noise problem and achieves significantly higher AWML performance than state-of-the-art exploration strategies — including Random Network Distillation (RND) [Burda et al., 2018b] and Model Disagreement [Pathak et al., 2019].

2 Related Works

2.1 Artificial Intelligence Literature

World Models. A natural class of world models involve forward dynamics prediction. Such models can directly predict future video frames [Finn et al., 2016, Wang et al., 2018, Wu et al., 2019], or latent feature representations such as 3D point clouds [Byravan and Fox, 2017] or object-centric, graphical representations of scenes [Battaglia et al., 2016, Chang et al., 2016, Mrowca et al., 2018]. Action-conditioned forward-prediction models can be used directly in planning for robotic control tasks [Finn and Levine, 2017], as performance-enhancers for reinforcement learning tasks [Ke et al., 2019], or as "dream" environment simulations for training policies [Ha and Schmidhuber, 2018]. In our work, we focus on forward dynamics prediction with object-oriented representations.

Active Learning and Curiosity. A key question the agent is faced with is how to choose its actions to efficiently learn the world model. In the classical *active learning* setting [Settles, 2011], an agent seeks to learn a supervised task with costly labels, judiciously choosing which examples to obtain labels for so as to maximize learning efficiency. More recently, active learning has been implicitly generalized to self-supervised reinforcement learning agents [Schmidhuber, 2010, Oudeyer et al., 2013, Jaderberg et al., 2016]. In this line of work, agents typically self-supervise a world model with samples obtained by curiosity-driven exploration. Different approaches to this general idea exist, many of which are essentially different approaches to estimating future *learning progress*

— e.g. determining which actions are likely to lead to the highest world model prediction gain in the future. One approach is the use of *novelty* metrics, which measure how much a particular part of the environment has been explored, and direct agents into under-explored parts of state-space. Examples include count-based and psuedo-count-based methods [Strehl and Littman, 2008, Bellemare et al., 2016, Ostrovski et al., 2017], Random Network Distillation (RND) [Burda et al., 2018b], and *empowerment* [Mohamed and Rezende, 2015]. Novelty-based approaches avoid the difficult world model progress estimation problem entirely by not depending at all on a specific world model state, and relying on novelty as a (potentially inconsistent) proxy for expected learning progress.

The simplest idea that takes into account the world model is *adversarial* curiosity, which estimates current world model error and directs agents to take actions estimated to maximize this error [Stadie et al., 2015, Pathak et al., 2017, Haber et al., 2018]. However, adversarial curiosity is especially prone to the *white noise problem*, in which agents are motivated to waste time fruitlessly trying to solve unsolvable world model problems, e.g. predicting the dynamics of random noise. The white noise problem can to some degree be avoided by solving the world-modeling problem in a learned latent feature space in which degeneracies are suppressed [Pathak et al., 2017, Burda et al., 2018a].

Directly estimating learning progress [Oudeyer et al., 2007, 2013] or *information gain* [Houthooft et al., 2016] avoids the white noise problem in a more comprehensive fashion. However, such methods have been limited in scope because they involve calculating quantities that cannot easily be estimated in high-dimensional continuous action spaces. *Surprisal* [Achiam and Sastry, 2017] and model disagreement [Pathak et al., 2019] present computationally-tractable alternatives to information gain, at the cost of the accuracy of the estimation. For comprehensive reviews of intrinsic motivation signal choices, see [Aubret et al., 2019, Linke et al., 2019]. In this work, we present a novel method for estimating learning progress that is "consistent" with the original prediction gain objective while also scaling to high-dimensional continuous action-spaces.

2.2 Cognitive Science Literature

Intuitive physics and object-based priors. Humans excel at intuitively predicting object dynamics [Battaglia et al., 2018]. A key framework underlying such abilities is object-centric attention allocation. Humans are able to keep track of objects over time, even as they become occluded or leave the visual frame [Piaget, 1952]. In this work, we include object-based attention and object permanence as neural architectural biases.

Curiosity and active learning. Humans interact with the world to learn how it works. Infants actively gather information from their environment by attending to objects in a highly non-random manner [Smith et al., 2019], devoting more attention to objects that violate their expectations [Stahl and Feigenson, 2015]. They also self-generate learning curricula, preferring stimuli that are complex enough to be interesting but still predictable [Kidd et al., 2012]. We study active learning by means of attention allocation.

Animate attention. From early infancy, humans effectively distinguish between inanimate and animate agents, preferentially paying attention to animate features like faces [Maurer et al., 2002]. Even in the absence of such visual features, infants preferentially attend to spatiotemporal kinematics indicative of animacy, such as efficient movement towards targets [Gergely et al., 1995] and contingent behavior between agents [Frankenhuis et al., 2013]. Such kinematic patterns give rise to an irresistable sense of animacy, even when the moving objects are simple shapes [Heider and Simmel, 1944]. In this work, instead of injecting biases for animate attention, we test whether it emerges naturally, albeit with the right choice of curiosity.

Social prediction and theory of mind. A more sophisticated ability emerging later in develop-

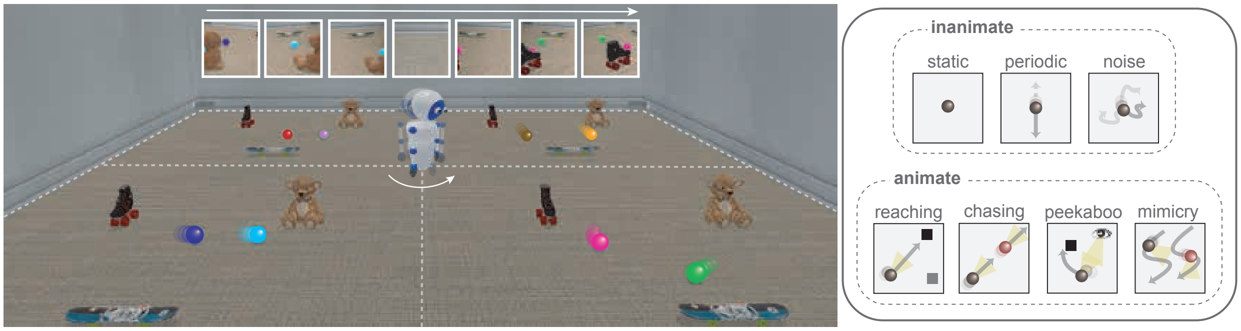


Figure 1: **Virtual environment.** Our 3D virtual environment is a distillation of key aspects of real-world environments. The *curious agent* (white robot) is centered in a room, surrounded by various *external agents* (colored spheres) contained in different quadrants, each with dynamics that correspond to a realistic inanimate or animate behavior (right box). The curious agent can rotate to attend to different behaviors as shown by the first-person view images at the top. See https://bit.ly/31vg7v1 for videos.

ment is understanding other agents' behaviors as consequences of their underlying mental states, aka *theory of mind* [Astington et al., 1990]. Theory of mind allows generating predictions about other agents as a function of their underlying mental states. In this work, our model learns to predict what other agents in the environment will do next through the use of a disentangled architecture that leverages the idea of different agents having different underlying internal states.

3 Multi-Agent Virtual World Environment

To faithfully replicate real-world algorithmic challenges, we design our 3D virtual environment to preserve the following key properties of real-world environments:

1. *Diverse dynamics.* Agents operate under a diverse set of dynamics specified by agent-specific programs. An agent's actions may depend on those of another agent resulting in complex interdependent relationships.

2. *Partial observability.* At no given time do we have full access to the state of every agent in the environment. Rather, our learning is limited by what lies within our field of view.

3. *Contingency.* How much we learn is contingent on how we, as embodied agents, choose to interact with the environment.

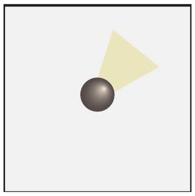
Concretely, our virtual environment consists of two main components, a *curious agent* and various *external agents.*

The *curious agent*, embodied by an avatar, is fixed at the center of a room (Figure 1). Just as a human toddler can control her gaze to visually explore her surroundings, the agent is able to partially observe the environment based on what lies in its field of view (see top of Figure 1). The agent can choose from 9 actions: rotate $12°, 24°, 48°$, or $96°$, to the left/right, or stay in its current orientation.

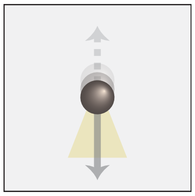
The *external agents* are spherical avatars that each act under a hard-coded policy inspired by real-world inanimate and animate stimuli. An *external agent behavior* consists of either one external agent, e.g reaching, or two interacting ones, e.g chasing. Since external agents are devoid of surface features, the curious agent must learn to attend to different behaviors based on spatiotemporal kinematics alone. We experiment with external agent behaviors (see Figure 1, right) including static, periodic, noise, reaching, chasing, peekaboo, and mimicry. The animate behaviors have deterministic and stochastic variants, where the stochastic variant preserves the core dynamics underlying the behavior, albeit with more randomness. See https://bit.ly/31vg7v1 for video descriptions of the environment and external agent behaviors.

We divide the room into four quadrants, each of which contains various auxiliary objects (e.g teddy bear, roller skates, surfboard) and one external agent behavior. The room is designed such that the curious agent can see at most one external agent behavior at any given time. This design is key in ensuring partial observability, such that the agent is faced with the problem of allocating attention between different external agent behaviors in an efficient manner. Below, we describe all behaviors in detail. Note that a subset of behaviors (peekaboo, reaching, and chasing) is further sub-divided into deterministic and stochastic varieties.

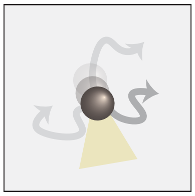
Inanimate behaviors



Static Inspired by stationary objects such as couches, lampposts, and fire hydrants, the *static agent* remains at its starting location and stays immobile.

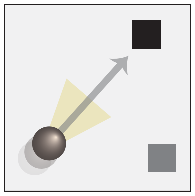


Periodic Inspired by objects exhibiting periodic motion such as fans, flashing lights, and clocks, the *periodic agent* regularly moves back and forth between two specified locations in its quadrant.

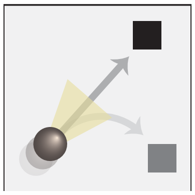


Noise Inspired by random motion in wind, water, and other inanimate elements, the *noise agent* randomly samples a new direction and moves in that direction with a fixed step size while remaining within the boundaries of its quadrant.

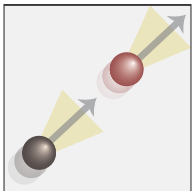
Animate Behaviors



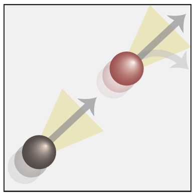
Reaching (deterministic) We often exhibit goal-oriented behavior by interacting with objects. The *reacher agent* approaches each auxiliary object in its quadrant sequentially, such that object positions fully determine its trajectory. Objects periodically shift locations such that predicting agent behavior at any given time requires knowing the current object positions.



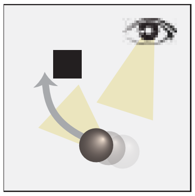
Reaching (stochastic) The order in which the reacher agent visits the objects is stochastic (uniform sampling from the three possible objects). However, once the reacher agent starts moving towards an object, its trajectory for the next few time steps, before it chooses a different object to move to, is predictable.



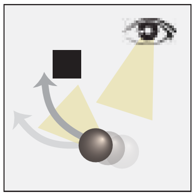
Chasing (deterministic) We often act contingently on the actions of other agents, which in turn depend on our own. In chasing, a *chaser agent* chases a *runner agent*. If the runner is too close to quadrant bounds, it then escapes to one of a few escape locations away from the chaser but within the quadrant. Thus, the chaser's position affects the runner's trajectory and vice versa.



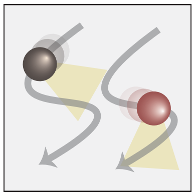
Chasing (stochastic) When the runner agent is too close to the quadrant bounds, it escapes by picking any random location away from the chaser and within the bounds of the quadrant.



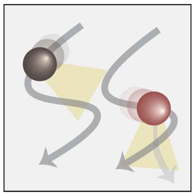
Peekaboo (deterministic) The *peekaboo agent* acts contingently on the curious agent. If the curious agent stares at it, it hides behind an auxiliary object such as a doll. If the curious agent continues to stare, it starts *peeking* out by moving to a close fixed location. Once the curious agent looks away, it stops hiding, returning to an exposed location.



Peekaboo (stochastic) There are multiple peeking locations near the hiding object that the peekaboo agent can visit randomly during its peeking behavior.



Mimicry (deterministic) From an early age, we learn by imitating others. Mimicry consists of an *actor agent* (red) and an *imitator agent* (gray), each staying in one half of the quadrant to avoid collisions. The actor acts identically to the random agent, while the imitator mirrors the actor's trajectory with a delay, such that the past trajectory of the actor fully determines the future trajectory of the imitator.



Mimicry (stochastic) The imitator agent is imperfect and produces a noisy reproduction of the actor agent's trajectory.

4 Theory

In this section we formalize Active World Model Learning (AWML) as a Reinforcement Learning (RL) problem that is a specific form of active learning. We then discuss a number of curiosity signals that can be used to drive AWML, and introduce γ-Progress, a scalable progress-based measure with several algorithmic and computational advantages over previous signals.

4.1 Active World Model Learning

We formalize an agent in environment as the tuple $\mathcal{E} := (\mathcal{S}, \mathcal{A}, P, P_0)$. $\mathcal{S}$ denotes the set of states the agent and environment can be in — in the virtual world environment described in section 3, $\mathcal{S}$ captures the gaze direction of the curious agent, the positions and type of external objects, and the positions and internal states of the external agents*. $\mathcal{A}$ represents the set of actions the agent can take, and are constrained by the physical avatar of the agent — in the virtual world, the choice of how far and where to turn its gaze. Transition dynamics are given by the function $P : \mathcal{S} \times \mathcal{A} \to \Omega(\mathcal{S})$, where $\Omega(\mathcal{S})$ is the set of probability measures on $\mathcal{S}$ (allowing for stochastic environment dynamics). In the case of our virtual world, P captures both the effect of the gaze actions of the agent (e.g. changes in which part of the scene is being observed), as well the dynamics of each of the external

*Our virtual world environment is *partially observable* and hence requires the additional specification of $\mathcal{O}$, the set of observations, and $Q = Q(\mathbf{o}|\mathbf{s}, \mathbf{a})$, the set of conditional observation probabilities. For the sake of simplicity, we suppress this complication in the main text and point out where it is salient in a series of footnotes.

agents. The function $P_0 : \mathcal{S} \to [0, 1]$ describes the probability distribution of initial conditions of states.

In this environment, the agent's overall goal is to learn a target function ω with as few data samples as possible. In general, ω can be any predictor on finite-horizon state-action trajectories sampled from the environment. That is, $\omega : \mathcal{X} \to \Omega(\mathcal{Y})$, where $\mathcal{X} := \mathcal{S}^{i_s} \times \mathcal{A}^{i_a}$ and $\mathcal{Y} := \Omega(\mathcal{S}^{o_s} \times \mathcal{A}^{o_a})$ represent sets of fixed-length observation-action sequences. (The non-negative integers i_s, i_a, o_s, and o_a are the input and output state and action horizons, respectively.) In this work, we work with forward prediction, i.e. the situation where $\mathcal{X} = \mathcal{S} \times \mathcal{A}, \mathcal{Y} = \mathcal{S}$, and $\omega = P$, but a variety of other potentially useful targets, such as inverse prediction, can also be formulated by appropriate choice of $\mathcal{X}, \mathcal{Y}$ and ω.[†]

The agent seeks to estimate a parameterized model ω_θ of ω (e.g θ are parameters deep neural network; see section 5 below). We henceforth refer to ω_θ as the world model. To measure its error during world model optimization, the agent is equipped with a loss function $\mathcal{L} : (x, f, g) \mapsto \mathbb{R}$ such that for any $x \in \mathcal{X}$ and any functions $f, g : \mathcal{X} \to \Omega(\mathcal{Y})$, $\mathcal{L}(x, f, g)$ achieves its minimum whenever $f(x) = g(x)$. A measure μ over $\mathcal{X}$ representing a validation data distribution is also specified, so that the agent's learning goal is to minimize $\mathcal{L}_\mu(\theta) := \mathbb{E}_\mu[\mathcal{L}(\theta)] = \int_\mathcal{X} \mathcal{L}(x, \omega(x), \omega_\theta(x))\mu(x)dx$.

The agent learns the world model from data gathering by acting in the environment. We formally define Active World Model Learning as a Markov Decision Process (MDP) $\mathcal{M} := (\bar{\mathcal{S}}, \bar{\mathcal{A}}, \bar{P}, \bar{P}_0, r)$ with state and action spaces $\bar{\mathcal{S}}, \bar{\mathcal{A}}$, dynamics and initial conditions $\bar{P}, \bar{P}_0$, and reinforcement reward function r. Because intrinsically-motivated policies (such as progress curiosity) will critically depend on states of the agent's world model, $\mathcal{M}$ is an augmentation of the environment $\mathcal{E}$ that is constructed by adding the data-collection and model parameter history of the agent itself.

Specifically, the augmented state space $\bar{\mathcal{S}} := \mathcal{S} \times \mathcal{H} \times \Theta$, so that $\bar{\mathbf{s}} \in \bar{\mathcal{S}}$ has the form $\bar{\mathbf{s}} = (\mathbf{s}, H, \theta)$. $\mathbf{s} \in \mathcal{S}$ is an environment state, $H = (\mathbf{s}_0, \mathbf{a}_0, \mathbf{s}_1, \mathbf{a}_1 \ldots) \in \mathcal{H}$ is the history of environment state-actions visited so far, and $\theta \in \Theta$ is the current model parameters. The action space $\bar{\mathcal{A}} := \mathcal{A}$ is simply the same set of actions available to the agent in the environment[‡]. The dynamics are described by $\bar{P} : \bar{\mathcal{S}} \times \mathcal{A} \to \Omega(\bar{\mathcal{S}})$, which step $\mathbf{s}$ according to the environment dynamics P, augment the history with new data, and updates the world model ω_θ on the augmented history. Formally this is described by the sampling procedure:

$$(\mathbf{s}', H', \theta') \sim \bar{P}(\cdot|\bar{\mathbf{s}} = (\mathbf{s}, H, \theta), \mathbf{a}) \;\; \text{where } \mathbf{s}' \sim P(\mathbf{s}, \mathbf{a}),\; H' = H \cup \{\mathbf{a}, \mathbf{s}'\},\; \theta' \sim P_\ell(H', \theta)$$

where $P_\ell : H \times \Theta \to \Omega(\Theta)$ is a (stochastic) update rule for the world model parameters, e.g. a (stochastic) learning algorithm which updates the parameters on the history of data. The initial conditions $\bar{P}_0(\bar{\mathbf{s}} = (\mathbf{s}, H, \theta)) = P_0(\mathbf{s})\mathbb{1}(H = \{\})q(\theta)$ is the augmented initial-distribution where $\mathbb{1}$ is the indicator function and $q(\theta)$ is a prior distribution over the model parameters.

The function r encodes the learning objective of the agent as an RL reward. A policy is a map $\pi : \mathcal{S} \to \Omega(\mathcal{A})$ from states to action distributions. In general, the infinite-horizon RL problem is to find an optimal policy $\pi^* = \arg\max_\pi J(\pi)$, where $J(\pi) = \mathbb{E}_\pi[\sum_{t=0}^\infty \beta^t r_t]$ and $0 \le \beta < 1$ is a discount factor. The goal of AWML in specific is to make effective data-collection decisions to minimize world model loss. This can in theory be accomplished by taking the reward function of AWML to be

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = -\mathcal{L}_\mu(\theta'), \tag{1}$$

[†] Actually, in partial observable case such as ours, the agent predicts observations from observations rather than raw states from raw states. Observations can contain additional information, such as the direction an external agent is moving, that is relevant to predicting future observations. However, they are also also typically quite incomplete, e.g. if an external agent is invisible until the observation to be predicted. This partial observability leads to what can be thought of as additional white noise, or *degeneracy* in the world model problem [Haber et al., 2018].

[‡] In the partial observability case, the action choice determines not only the state transition but also what is observable each timestep, and hence the agent should keep the interesting in view. The MDP becomes a POMDP, where we assume that the agent has full access to its internal state and history, so augmented observations $\bar{o} \in \bar{\mathcal{O}} = \mathcal{O} \times \mathcal{H} \times \Theta$ has the form $\bar{o} = (o, H, \theta)$, where $o \in O$ (augmented conditional observation probabilities $\bar{Q}$ are similarly derived from Q).

where $\bar{\mathbf{s}} = (\mathbf{s}, H, \theta), \bar{\mathbf{s}}' = (\mathbf{s}', H', \theta')$ and $\theta' = P_\ell(H \cup \{\mathbf{a}, \mathbf{s}'\}, \theta)$ is the updated model parameters after collecting new data $\{\mathbf{a}, \mathbf{s}'\}$.

It is useful to note that, given that the definition of total reward J is a telescoping geometric sum, optimizing for eq 1 is essentially equivalent to optimizing for the reward function:

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = \mathcal{L}_\mu(\theta) - \mathcal{L}_\mu(\theta'). \tag{2}$$

Thus we can see that, $r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}')$ essentially measures the reduction in world model loss as a result of obtaining new data $\{\mathbf{a}, \mathbf{s}'\}$, i.e the *prediction gain*.

By appropriately constructing $\mathcal{M}$, different variants of traditional active learning can be recovered as AWML problems. For example, Query Synthesis Active Learning [Settles, 2011] is obtained by taking $\mathcal{S} = \mathcal{Y}, \mathcal{A} = \mathcal{X}$, and $P(\cdot|\cdot, \mathbf{a} = \mathbf{x}) = \omega(\mathbf{x})$. In words, the agent proposes a synthetic data query $\mathbf{a}$ and the oracle P provides a label $\mathbf{s}'$. Other traditional active learning tasks can also be derived, including pool-based and stream active learning (see Appendix A for details).

However, there are several complications making it challenging to use eq. 2 directly. First, μ can be a rather diffuse distribution which makes it intractable to compute eq. 2 at every environment step. This is especially problematic in the types of environments of interest here and in other recent works on curiosity-driven learning, relative to the more constrained situations of traditional active learning. Secondly, in cases in which an agent explores an unknown environment, μ is not even known prior to interacting with the environment. These bottlenecks necessitate an efficiently-computable heuristic reward function that will typically promote the same learning goal of eq. 2 — constructing a learning dataset that minimizes the loss $\mathcal{L}_\mu$ — while being independent of any particular choice of μ. The literature on algorithmic curiosity has explored many variants of such heuristic "curiosity signals", which achieve consistency with the learning goals of eq. 2 with varying degrees of accuracy and efficiency. A spectrum of such ideas, including our novel proposal (γ-Progress), are described in the next section.

4.2 Curiosity Signals

We now motivate γ-Progress by outlining the limitations of previously proposed curiosity signals and highlighting the computational and algorithmic advantages of our method.

Information Gain [Houthooft et al., 2016, Linke et al., 2019] based methods seek to minimize uncertainty in the Bayesian posterior distribution over model parameters:

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = D_{\mathrm{KL}}(p(\theta')||p(\theta)) \tag{3}$$

where $p(\theta') = p(\theta|H \cup \{\mathbf{a}, \bar{\mathbf{s}}'\})$ and $p(\theta) = p(\theta|H)$. Note that, information gain is a lower bound to the prediction gain under weak assumptions [Bellemare et al., 2016]. If the posterior has a simple form such as Laplace or Gaussian, information gain can be estimated by weight change $|\theta' - \theta|$ [Linke et al., 2019], and otherwise one may resort to learning a variational approximation q to approximate the information gain with $D_{\mathrm{KL}}(q(\theta')||q(\theta))$ [Houthooft et al., 2016]. The former weight change methods require a model after every step in the environment and is thus impractical in many settings where world model updates are expensive, e.g. backpropagation through deep neural nets. The latter family of variational methods require maintenance of a parameter distribution and an interlaced evidence lower bound optimization and are thus impractical to use with modern deep nets [Achiam and Sastry, 2017].

Adversarial [Stadie et al., 2015, Pathak et al., 2017, Haber et al., 2018] curiosity assumes prediction gain is proportional to the current world model loss, which, for forward prediction AWML with negative log likelihood loss, is

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = -\log \omega_\theta(\mathbf{s}'|\mathbf{s}, \mathbf{a}). \tag{4}$$

This assumption holds when the target function ω is learnable by the model class Θ and the learning algorithm P_ℓ makes monotonic improvement without the need for curriculum learning. However, adversarial reward is perpetually high when the target is unlearnable by the model class, e.g. deterministic model ω_θ cannot match stochastic target ω on inputs $\mathbf{x}$ for which $\omega(\mathbf{x})$ is not a Dirac-delta function. As a result, the curious agent suffers from the white noise problem [Schmidhuber, 2010], i.e it endlessly fixates on unlearnable stimuli.

Disagreement [Pathak et al., 2019] assumes future world model loss reduction is proportional to the prediction variance of an ensemble of N world models $\{P_{\theta_j}\}_{j=1}^N$.

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = \mathrm{Var}(\{\omega_{\theta_j}(\mathbf{s}'|\mathbf{s}, \mathbf{a})\}_{j=1}^N) \tag{5}$$

This approximation is reasonable when there exists a unique optimal world model. As we will show, for complex target functions all members of the ensemble do not converge to a single model and as a result the white noise problem persists. A key limitation of this method is that memory usage grow linearly with size of the model ensemble. Disagreement-based curiosity is known as query by committee sampling [Seung et al., 1992] in active learning.

Novelty [Bellemare et al., 2016, Dinh et al., 2016, Burda et al., 2018b] methods reward transitions with a low visitation count $\mathcal{N}(s, a, s')$. The prototypical novelty reward is:

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = \mathcal{N}(s_t, a_t)^{-1/2} \tag{6}$$

Bellemare et al. [2016] generalize visitation counts to pseudocounts for use in continuous state, action spaces. Novelty is a good surrogate reward when one seeks to maximize coverage over the transition space regardless of the learnability of the transition. This characteristic makes novelty reward prefer noisy data drawn from a high entropy distribution. Novelty reward is not adapted to the world model and thus has a propensity to be inefficient at reducing world model loss.

Progress [Schmidhuber, 2010, Achiam and Sastry, 2017, Graves et al., 2017] The key idea is to simply approximate the expectation involving μ in eq. 2 with the prediction gain on the history.

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = \mathcal{L}_{H'}(\theta) - \mathcal{L}_{H'}(\theta') \tag{7}$$

where H' is the augmented history after adding $(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}')$. There is no guarantee the optimal policy with respect to eq. 7 is also an optimal policy with respect to eq. 2 for every choice of μ. However, we expect this history-based approximation of prediction gain to generate a data distribution that will be suitable for a wide array of μ. If we think of the target ω as having easy, hard but doable, and impossible instances $(\mathbf{x}, \mathbf{y})$, we expect such an agent to spend some time sampling easy, a good deal of time sampling the hard but doable, and little time on the impossible. For μ with support on easy data, little sampling is needed; for support on hard but doable, the greater proportion of samples is useful; and support on the impossible does not contribute to eq. 2 in the first place. Intuitively (if not formally), the progress curiosity approach should thus yield a data distribution[§] that is proportionate to the intrinsic learnability of the target ω.

Unlike eq. 2, policies based on eq. 7 are subject to sampling and robustness tradeoffs. New data gathered after parameter update from θ to θ' is expected to generate the most useful information for distinguishing $\mathcal{L}_{H'}(\theta)$ from $\mathcal{L}_{H'}(\theta')$. The fewer such samples there are in the history, the less well the empirical difference approximates true progress. It is thus useful to compute the empirical difference in eq. 7 and perform the next parameter update after multiple samples from θ' have been

[§] Actually, there is no guarantee that the stochastic process described by eq. 7 will converge in distribution at all, but the occupancy measure distribution will exist and the intuition presented here can be applied to that.

gathered. On the other hand, the number of new samples between θ and θ' cannot be allowed to be too large, since less frequent progress updates would slow down the improvement of the dataset and thus, presumably, limit the efficiency of world model improvement. A natural compromise is to smooth the empirical progress measurement over multiple consecutive parameter changes, pooling datapoints to better approximate progress. This comes at the cost, however, of requiring access to model parameters at those multiple timepoints and some method for combining progress trajectory information as the model itself changes. All this is further complicated by the fact that the number of gradient descent steps taken when computing θ' from θ will itself be limited for computational reasons, making even the empirical progress computations noisier. Ensuring reliable and efficient approximation of progress thus requires careful choices of how often to update θ' and how to integrate information across multiple updates.

δ-**Progress**. One approach to such choices is given by δ-progress [Achiam and Sastry, 2017, Graves et al., 2017], measures how much better the current "new" model θ_{new} is compared to an old model θ_{old}, which, for forward prediction AWML, is

$$r(\bar{\mathbf{s}}, \mathbf{a}, \bar{\mathbf{s}}') = \log \frac{\omega_{\theta'}(\mathbf{s}'|\mathbf{s}, \mathbf{a})}{\omega_{\theta}(\mathbf{s}'|\mathbf{s}, \mathbf{a})} \simeq \log \frac{\omega_{\theta_{new}}(\mathbf{s}'|\mathbf{s}, \mathbf{a})}{\omega_{\theta_{old}}(\mathbf{s}'|\mathbf{s}, \mathbf{a})}. \tag{8}$$

Recall that μ is ideally a distribution whose support is learnable data with respect to model class Θ. There are two steps of approximation in eq. 8. The first step assumes that training on a sample $(\mathbf{s}, \mathbf{a}, \mathbf{s}')$ affects the total validation loss on learnable data μ only through the reduction in loss on that particular sample. The second step assumes that future prediction gain is close to past prediction gain measured with respect to $\theta_{new}, \theta_{old}$. The choice of $\theta_{new}, \theta_{old}$ is crucial to the efficacy of the progress reward. A popular approach [Achiam et al., 2017, Graves et al., 2017] is to choose

$$\theta_{new} = \theta_k, \quad \theta_{old} = \theta_{k-\delta}, \quad \delta > 0 \tag{9}$$

where θ_k is the model parameter after k update steps using P_ℓ. Intuitively, if the progress horizon δ is too large, we obtain an overly optimistic approximation of future progress. However if δ is too small, the agent may prematurely give up on learning hard transitions, e.g. where the next state distribution is very sharp. In practice, tuning the value δ presents a major challenge. Furthermore, the widely pointed out [Pathak et al., 2019] limitation of δ-Progress is that the memory usage grows $\mathcal{O}(\delta)$, i.e one must store δ world model parameters $\theta_{k-\delta}, ..., \theta$. As a result it is intractable in practice to use $\delta > 3$ with deep neural net models.

γ-**Progress**. Here we propose γ-Progress, the following choice of $\theta_{new}, \theta_{old}$ to overcome both hurdles faces by δ-progress:

$$\boxed{\theta_{new} = \theta, \quad \theta_{old} = (1-\gamma)\sum_{i=1}^{k-1} \gamma^{k-1-i}\theta_i} \tag{10}$$

In words, the old model is a weighted mixture of past models where the weights are exponentially decayed into the past. γ-Progress can be interpreted as a noise averaged progress signal. Conveniently, γ-Progress can be implemented with a simple θ_{old} update rule:

$$\boxed{\theta_{old} \leftarrow \gamma\theta_{old} + (1-\gamma)\theta_{new}} \tag{11}$$

Similar to eq. 9, we may control the level of optimism towards expected future loss reduction by controlling the progress horizon γ, i.e a higher γ corresponds to a more optimistic approximation. γ-Progress has key practical advantages over δ-Progress: γ is far easier to tune than δ, e.g. we use a single value of γ throughout all experiments, and memory usage is constant with respect to γ. Crucially, the second advantage enables us to tune the progress horizon so that the model does not prematurely give up on exploring hard transitions. The significance of these practical advantages will become apparent from our experiments.

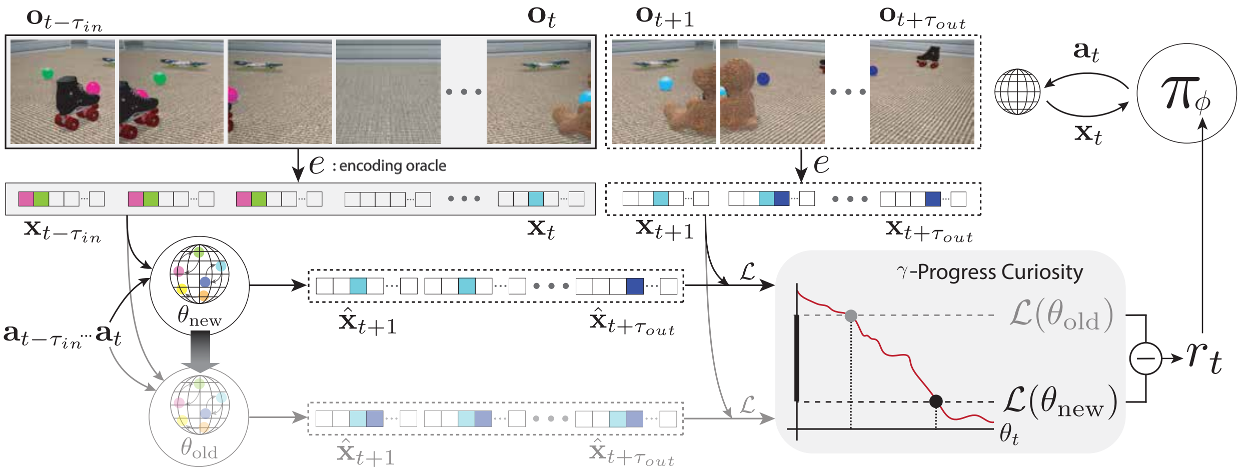

$\mathbf{o}_{t-\tau_{in}}$
$\mathbf{o}_t$
$\mathbf{o}_{t+1}$
$\mathbf{o}_{t+\tau_{out}}$
$\mathbf{a}_t$
$\mathbf{x}_t$
π_ϕ
e : encoding oracle
e
$\mathbf{x}_{t-\tau_{in}}$
$\mathbf{x}_t$
$\mathbf{x}_{t+1}$
$\mathbf{x}_{t+\tau_{out}}$
$\mathbf{a}_{t-\tau_{in}} \cdots \mathbf{a}_t$
θ_{new}
θ_{old}
$\hat{\mathbf{x}}_{t+1}$
$\hat{\mathbf{x}}_{t+\tau_{out}}$
$\mathcal{L}$
γ-Progress Curiosity
$\mathcal{L}(\theta_{\text{old}})$
$\mathcal{L}(\theta_{\text{new}})$
θ_t
$-$
r_t

Figure 2: **Active World Model Learning with γ-Progress** The curious agent consists of a *world model* and a *progress-driven controller*. The curious agent's observations $\mathbf{o}_t$ are passed through an encoding oracle e that returns an object-oriented representation $\mathbf{x}_t$ containing the positions of external agents that are in view, auxiliary object positions, and the curious agent's orientation. Both the new (opaque) and old (translucent) models take as input $\mathbf{x}_{t-\tau_{in}:t}$ and predict $\hat{\mathbf{x}}_{t:t+\tau_{out}}$. The old model weights, θ_{old}, are slowly updated to the new model weights θ_{new}. The controller, π_ϕ, is optimized to maximize γ-Progress reward: the difference $\mathcal{L}(\theta_{old}) - \mathcal{L}(\theta_{new})$.

5 Methods

In this section we describe practical instantiations of the two components in our AWML system: a *world model* which fits the forward dynamics and a *controller* which chooses actions to maximize γ-Progress reward. See Appendix B for full details on architectures and training procedures.

World Model As the focus of this work is not to resolve the difficulty of representation learning from high-dimensional visual inputs, we assume that the agent has access to an oracle encoder $e : \mathcal{O} \to \mathcal{X}$ that maps an image observation $\mathbf{o}_t \in \mathcal{O}$ to a disentangled object-oriented feature vector $\mathbf{x}_t = (\mathbf{x}_t^{ext}, \mathbf{x}_t^{aux}, \mathbf{x}_t^{ego})$ where $\mathbf{x}_t^{ext} = (\tilde{\mathbf{c}}_t, \mathbf{m}_t) = (\tilde{\mathbf{c}}_{t,1}, \ldots \tilde{\mathbf{c}}_{t,n_{ext}}, \mathbf{m}_{t,1}, \ldots, \mathbf{m}_{t,n_{ext}})$ contains information about the external agents; namely the observability masks $\mathbf{m}_{t,i}$ ($\mathbf{m}_{t,i} = 1$ if external agent i is in curious agent's view at time t, else $\mathbf{m}_{t,i} = 0$) and masked position coordinates $\tilde{\mathbf{c}}_{t,i} = \mathbf{c}_{t,i}$ if $\mathbf{m}_{t,i} = 1$ and else $\tilde{\mathbf{c}}_{t,i} = \hat{\mathbf{c}}_{t,i}$. Here, $\mathbf{c}_{t,i}$ is the true global coordinate of external agent i and $\hat{\mathbf{c}}_{t,i}$ is the model's predicted coordinate of external agent i where $i = 1, \ldots, n_{ext}$. Note that the partial observability of the environment is preserved under the oracle encoder since it provides coordinates only for external agents in view. $\mathbf{x}_t^{aux}$ contains coordinates of auxiliary objects, and $\mathbf{x}_t^{ego}$ contains the ego-centric orientation of the curious agent.

Our world model ω_θ is an ensemble of component networks $\{\omega_{\theta^k}\}_{k=1}^{N_{\text{cc}}}$ where each ω_{θ^k} independently predicts the forward dynamics for a subset $I_k \subseteq \{1, ..., \dim(\mathbf{x}^{ext})\}$ of the input dimensions of $\mathbf{x}^{ext}$ corresponding to a minimal behaviorally interdependent group. For example, $\mathbf{x}_{t:t+\tau,I_k}^{ext}$ may correspond to the masked coordinates and observability masks of the chaser and runner external agents for times $t, t+1, ..., t+\tau$. We found that such a "disentangled" architecture outperforms a simple entangled architecture (see Discussion and Fig. 5). We assume $\{I_k\}_{k=1}^{N_{\text{cc}}}$ is given as prior knowledge but future work may integrate dependency graph learning into our pipeline. A component network ω_{θ^k} takes as input $(\mathbf{x}_{t-\tau_{in}:t,I_k}^{ext}\ ,\ \mathbf{x}_{t-\tau_{in}:t}^{aux}\ ,\ \mathbf{x}_{t-\tau_{in}:t}^{ego}\ ,\ \mathbf{a}_{t-\tau_{in}:t+\tau_{out}})$, where $\mathbf{a}$ denotes the curious agent's actions, and outputs $\hat{\mathbf{x}}_{t:t+\tau_{out},I_k}^{ext}$. The outputs of the component network are concatenated to get the final output $\hat{\mathbf{x}}_{t:t+\tau_{out}}^{ext} = (\hat{\mathbf{c}}_{t:t+\tau_{out}}, \hat{\mathbf{m}}_{t:t+\tau_{out}})$. The world model loss is:

$$\mathcal{L}(\theta, \mathbf{x}_{t-\tau_{in}:t+\tau_{out}}, \mathbf{a}_{t-\tau_{in}:t+\tau_{out}}) = \sum_{t'=t}^{t+\tau_{out}} \sum_{i=1}^{N_{ext}} \mathbf{m}_{t',i} \cdot \|\hat{\mathbf{c}}_{t',i} - \tilde{\mathbf{c}}_{t',i}\|_2 + \mathcal{L}_{ce}(\hat{\mathbf{m}}_{t',i}, \mathbf{m}_{t',i})$$

Algorithm 1: AWML with γ-Progress

```
**Require:** progress horizon $\gamma$, step sizes $\eta_\omega, \eta_Q$
Initialize $\theta_{new}, \phi$
**for** $k = 1, 2, ...$ **do**
    Update policy: $\pi_\phi \leftarrow \epsilon\text{-}greedy(Q_\phi, \epsilon - 0.0001)$
    Sample $(\mathbf{x}, \mathbf{a}, c) \sim \pi_\phi$ and place in Buffer $\mathcal{B}$
    where $c = \mathcal{L}(\theta_{new}, \mathbf{x}, \mathbf{a}) - \mathcal{L}(\theta_{old}, \mathbf{x}, \mathbf{a})$
    **for** $j = 1, ..., M$ **do**
        Sample batch $b_j \sim \mathcal{B}$
        Update new world model: $\theta_{new} \leftarrow \theta_{new} - \text{ADAM}(\theta_{new}, b_j, \eta_\omega, \mathcal{L})$
        Update old world model: $\theta_{old} \leftarrow \gamma\theta_{old} + (1 - \gamma)\theta_{new}$
        Update Q-network with DQN [Mnih et al., 2015]: $\phi \leftarrow \text{DQN}(\phi, b_j, \eta_Q)$
    **end**
**end**
```

where $\mathcal{L}_{ce}$ is cross-entropy loss. We parameterize each component network ω_{θ^k} with a two-layer Long Short-Term Memory (LSTM) network followed by two-layer Multi Layer Perceptron (MLP). The number of hidden units are adapted to the number of external agents being modeled.

The Progress-driven Controller Our controller π_ϕ is a two-layer fully-connected network with 512 hidden units that takes as input $\mathbf{x}_{t-2:t}$ and outputs estimated Q-values for 9 possible actions which rotate the curious agent at different velocities. π_ϕ is updated with the DQN [Mnih et al., 2013] learning algorithm using the cost:

$$c(\mathbf{x}_t) = \mathcal{L}(\theta_{new}, \mathbf{x}_{t-\tau_{in}-\tau_{out}:t}, \mathbf{a}_{t-\tau_{in}-\tau_{out}:t}) - \mathcal{L}(\theta_{old}, \mathbf{x}_{t-\tau_{in}-\tau_{out}:t}, \mathbf{a}_{t-\tau_{in}-\tau_{out}:t}) \tag{12}$$

with $\gamma = 0.9995$ across all experiments.

6 Experiments

We evaluate the AWML performance of γ-Progress on two metrics: *end performance* and *sample complexity*. End performance is the inverse of the the final world loss after a larger number of environment interactions, and intuitively measures the “consistency” of the proxy reward with respect to the true reward. Sample complexity measures the rate of reduction in world model loss $\mathcal{L}_\mu(\theta)$ with respect to the number of environment interactions. The samples from the validation distribution μ correspond to core validation cases we crafted for each behavior. On the reaching behaviors, for example, we validate the world model loss with objects spawned at new locations. For details on each behavior-specific validation case and metric computation, we refer readers to Appendix C.

Experiments are run in two virtual worlds: Mixture and Noise world. In the Mixture world, the virtual environment is instantiated with external agents spanning four representative types: static, periodic, noise, and animate. This set up is a natural distillation of a real-world environment containing a wide spectrum of behaviors. In the Noise world, the environment is instantiated with three noise agents and one animate agent. This world stress-tests the noise robustness of γ-Progress. For each world, we run separate experiments in which the animate external agents are varied amongst the deterministic and stochastic versions of reaching, chasing, peekaboo, and mimicry agents (see Section 3).

We compare the AWML performance of the following methods:

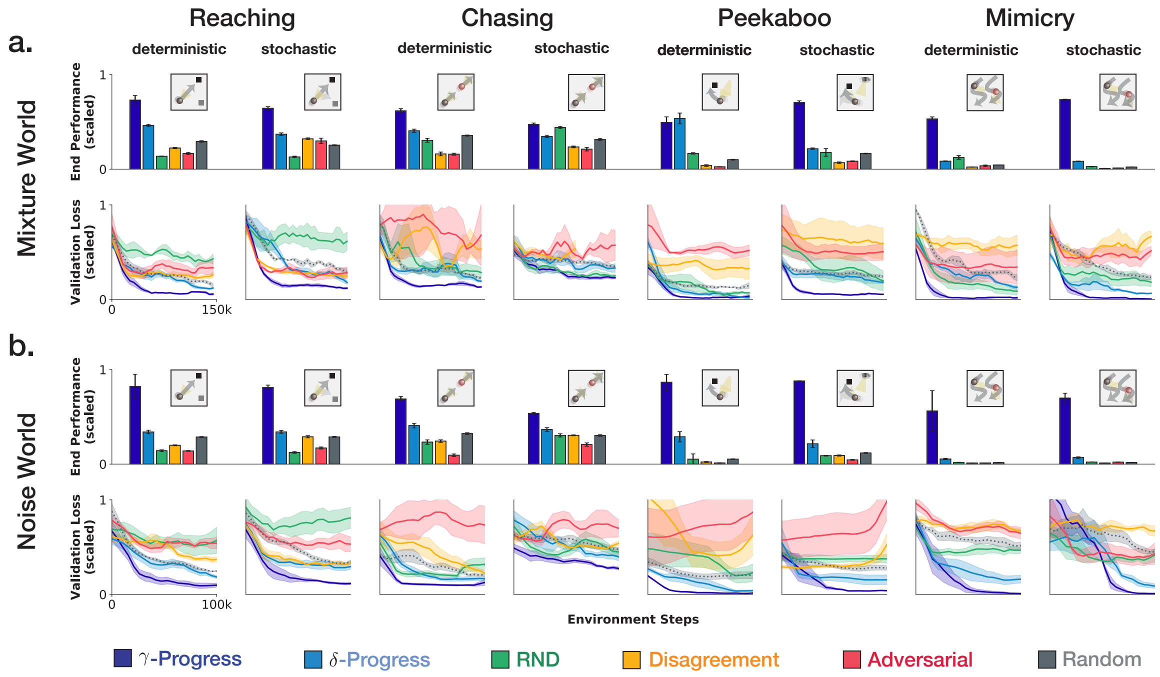


Figure 3: **AWML Performance and Sample Complexity**. The animate external agent is varied across experiments according to the column labels. End performance is the mean of the last five validation losses. Sample complexity plots show validation losses every 5000 environment steps. Error bars/regions are standard errors of the best 5 seeds out of 10. (a). *Mixture World*: γ-Progress achieves lower sample complexity than all baselines on 7/8 behaviors. Notably, γ-Progress also outperforms all baselines in end performance on 6/8 behaviors. (b). *Noise World*: γ-Progress is more robust to white noise than baselines and achieves lower sample complexity and higher end performance on 8/8 behaviors. Baselines frequently perform worse than random due to noise fixation

Table 1: **AWML Performance Summary.** Mean ratio of baseline end performance over Random baseline end performance (standard error in parentheses)

	Mixture World	Noise World
γ-Progress	**7.83** (3.57)	**13.79** (5.29)
δ-Progress	2.2 (0.51)	2.46 (0.55)
RND	1.25 (0.25)	0.85 (0.10)
Disagreement	0.62 (0.10)	0.76 (0.06)
Adversarial	0.62 (0.09)	0.59 (0.10)

- **γ-Progress (Ours)** is our proposed variant of progress curiosity which chooses θ_{old} to be a geometric mixture of all past models as in Eq. 10.
- **δ-Progress [Achiam and Sastry, 2017, Graves et al., 2017]** is the δ-step learning progress reward from Eq. 9 with $\delta = 1$. We found that any $\delta > 3$ is impractical due to memory constraints.
- **RND [Burda et al., 2018b]** is a novelty-based method that trains a predictor neural net to match the outputs of a random state encoder. States for which the predictor networks fails to match the random encoder are deemed "novel", and thus receive high reward.
- **Disagreement [Pathak et al., 2019]** is the disagreement based method from Eq. 5 with $N = 3$ ensemble models. We found that $N > 3$ is impractical due to memory constraints.
- **Adversarial [Stadie et al., 2015, Pathak et al., 2017]** is the prediction error based method from Eq. 4. We use the ℓ_2 prediction loss of the world model as the reward.
- **Random** chooses actions uniformly at random among the 9 possible rotations.

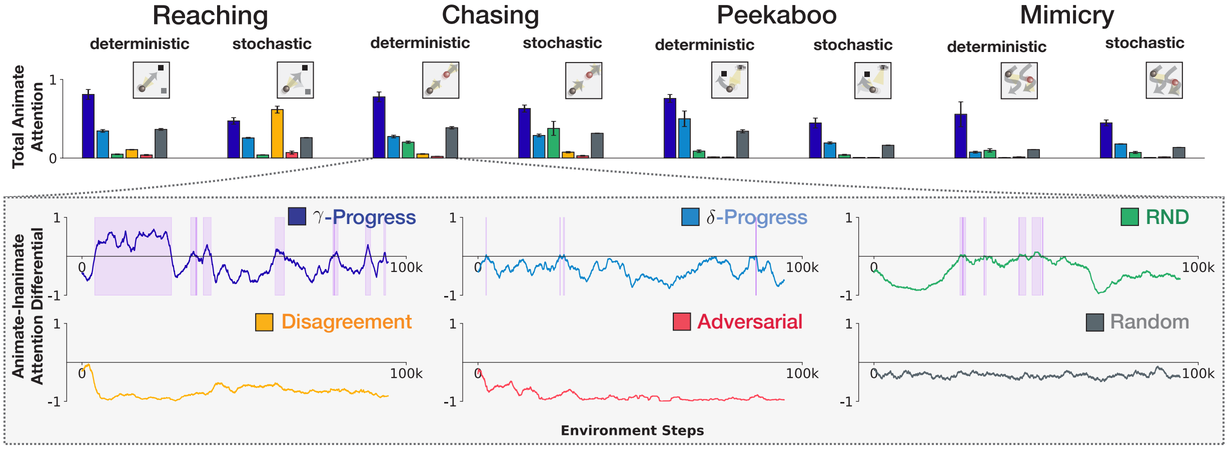


Figure 4: **Attention Patterns**. The bar plot shows the total animate attention, which is the ratio between the number of time steps an animate external agent was visible to the curious agent, and the time steps a noise external agent was visible. The time series plots in the zoom-in box show the differences between mean attention to the animate external agents and the mean of attention to the other agents in a 500 step window, with periods of animate preference highlighted in purple. Results are averaged across 5 runs. γ-Progress displays strong animate attention while baselines are either indifferent, e.g δ-Progress, or fixating on white noise, e.g Adversarial.

6.1 AWML Performance.

Fig. 3a shows end performance (first row) and sample complexity (second row) in the Mixture world, and Fig. 3b shows the same for the Noise World. In the Mixture world, we see that γ-Progress has lower sample complexity than δ-Progress, Disagreement, Adversarial, and Random baselines on all 8/8 behaviors and outperforms RND on 7/8 behaviors while tying on stochastic chasing. In the Noise world, we see that γ-Progress has lower sample complexity than all baselines on all 8/8 behaviors. See Table 1 for aggregate end performance, and https://bit.ly/31vg7v1 for visualizations of model predictions.

6.2 Attention control analysis

Figure 4 shows the ratio of attention to animate vs other external agents for each behavior in the Mixture world as well as example animate-inanimate attention differential timeseries (for the Noise world, see Appendix D). The γ-Progress agents spend substantially more time attending to animate agents than do alternative policies. This increased animate-inanimate attention differential often corresponds to a characteristic attentional "bump" that occurs early as the γ-Progress curious agent focuses on animate external agents quickly before eventually "losing interest" as prediction accuracy is achieved. Strong animate attention emerges for 7/7 behaviors when using γ-Progress. Please see appendix E for a more in-depth analysis of how attention, and particular early attention, predicts performance and how curiosity signal predicts attention.

Baselines display two distinct modes that lead to lower performance (Table 2). The first is *attentional indifference*, in which the curious agent finds no particular external agent interesting — more precisely, we say that a curiosity signal choice displays attentional indifference if its average animate/inanimate ratio in the Mixture world is within two standard deviations of the Random policy's. δ-Progress frequently had attentional indifference as the new and old world model, separated by a fixed time difference, were often too similar to generate a useful curiosity signal.

The second failure mode is *white noise fixation*, where the observer is captivated by the noise external agents — more precisely, we say that a curiosity signal choice displays white noise fixation if its average animate/inanimate ratio in the Noise world is more than two standard deviations below the Random policy's. RND suffers from white noise fixation due to the fact that our noise

Table 2: **Failure modes** Fraction of indifference and white noise failures, out of eight external agent behaviors.

	Indifference	Noise Fixation
γ-Progress	**0/8**	**0/8**
δ-Progress	7/8	0/8
RND	2/8	4/8
Disagreement	0/8	7/8
Adversarial	0/8	8/8
Random	8/8	0/8

behaviors have the most diffuse visited state distribution. We also observe that for noise behaviors, a world model ensemble does not collectively converge to a single mean prediction, and as a result Disagreement finds the noise behavior highly interesting. Finally, the Adversarial baseline fails since noise behaviors yield the highest prediction errors. The white noise failure mode is particularly detrimental to sample complexity for RND, Disagreement, and Adversarial, as evidenced by their below-Random performance in the Noise world.

7 Discussion and Future Directions

In this work, we propose an Active World Model Learning agent that observes and interacts with an agent-rich 3D environment. The AWML agent learns a predictive world model of this environment, combining an agent-centric disentangled world model with a curiosity-driven action policy. A main contribution of this work is introducing γ-Progress, a computationally-tractable approximate estimator of expected information gain. γ-Progress is sensitive and robust enough to discover *de novo* a simple form of animate attention without having to have this concept built in, "realizing" that animate agents are more interesting to focus on than inanimate alternatives across a variety of animate agent types and inanimate distractors. The curious neural agent equipped with γ-Progress is better able to allocate scarce attentional resources in the partially observable environment, and is thus substantially more effective at learning world models in our agent-rich environment.

More realistic embodiments, richer tasks and behaviors, and real-world input streams. While our environment does capture some key features of proto-social interactions, it is lacking in several important ways. First, our AWML agent only has gaze-driven interactions with the enviroment, and external agents have no complex effectors or physical features capable of expressing social cues. Extending our current work with a more realistic embodiment (including agents with full motility, articulated effectors, and gaze cue markers) is an important direction, especially because some types of important proto-social behaviors — such as mutual gaze coordination, gaze following, and pointing — can only be expressed using richer avatars. In this work we have only targeted improved external agent prediction as the success metric, but an important next step, enabled by improved embodiment, will be to extend to the case of imitation learning, where the observing agent not only learns about others but also about how to do things itself. Another limitation is that only the single central observer is implemented as an AWML agent, with all the external-agent behaviors limited to hard-coded routines. We seek to investigate true multi-agent scenarios where all agents are running a curious policy (perhaps at different stages of learning).

In the present work, we have chosen to avoid the complication of having the learned components of our AWML system work directly with visual inputs, so as to focus on the challenging policy learning problems. However, forcing our agents (both observer and the external) to use pixel-based visual inputs will be an important next step. We expect that filling in this gap will be a meaningful challenge, especially given the need to integrate a working memory system to handle partial observability (determining e.g. when an agent has gone out of view and identifying it when it returns). Once these improvements in embodiment, input realism, and behavioral richness are made, we will seek to deploy AWML on a real-world robotic platform.

Disentangled world models and theory of mind. To produce effective world models, we found it helpful to use an "agent-centric" disentangled architecture. However, is this choice necessary? Do architectures not based on some form of agent-centric disentangling always fail to solve social multi-agent prediction problems? This is not obvious, since disentangling has proved a non-optimal (or at least non-necessary) strategy in some domains [Locatello et al., 2018, Hong et al., 2016]. As an initial investigation of this issue, we performed a pilot investigation of the effect of world model disentanglement on external-agent prediction performance (Fig. 5). To ensure that this evaluation is independent of the choice of the policy controller (e.g. which type of curiosity, if any), our evaluation uses offline training datasets, one for each task in our current environment.* We compare performance between the agent-centric disentangled world model and an "entangled" or "joint" LSTM architecture that instead takes as input and predicts all external agents together. The disentangled architecture significantly outperforms the joint version: in fact, we originally began our investigation attempting to use the joint model, and only switched to the more complicated disentangled version when the former failed to work. Intuitively, the disentangled architecture performs better because it ignores spurious correlations between causally-unrelated events in the agent's data stream. Formalizing this intuition mathematically and explaining why it may be particularly relevant in our current environment, in contrast to some other situations [Locatello et al., 2018, Hong et al., 2016], is an important future direction. If it turns out that an agent-centric disentangled architecture is indeed robustly necessary, a natural question that will need to be resolved is how the interaction graph describing agent-agent dependencies can be estimated from observations, rather than known oracularly as here.

Interestingly, this agent-centric disentangled architecture shares a key feature with the concept from cognitive science known as Theory of Mind (ToM). ToM describes the ability of one person to predict the behaviors of other people by inferring the others' mental states, such as beliefs, desires, and goals ([Astington et al., 1990, Premack and Woodruff, 1978, Wellman, 1992]). A core, though often implicit, assumption of ToM is that separate predictive models are individually constructed for each non-self agent (or group of interacting non-self agents), and inferences about mental states are performed on a per-external-agent basis. Our disentangled model builds this as an inherent part of the architecture, and the performance improvements we observe from that choice loosely suggest that at least one possible function of ToM may be to enable statistical disentangling in highly partially-observed settings. Obviously, full ToM involves many other features beyond mere disentanglement that should enable effective external-agent model-making (e.g. model sharing and recursive representation of belief states) [De Freitas et al., 2019], so the connection to the present work is at best partial. However, making this connection more concrete, and understanding how to build other key ToM features into an improved world model afford interesting avenues for future work.

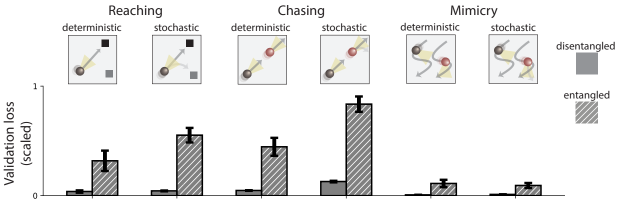


Figure 5: **Asymptotic Model Performance** Final validation loss of the disentangled world model and entangled ablation on fixed dataset.

Toward quantitative models of human behavior and developmental variability. Aside from AI uses, we hope that AWML might provide the basis of a quantitative model of human behavior. As a preliminary gesture in this direction, we have run a pilot human subject experiment (Fig. 6a) in which we exposed twelve adult human participants to static, periodic, animate, and noise stimuli using a display in which external agents were embodied with featurally-uniform spherical robots known as Spheros ([Kurkovsky, 2013]). Patterns of participants' attention were measured via a mobile eye tracker. We found that human adults display a clear animacy preference, quickly

[*] Excluding peekaboo, since because the behavior is dependent on the observer's choices, no policy-independent offline training dataset can be constructed.

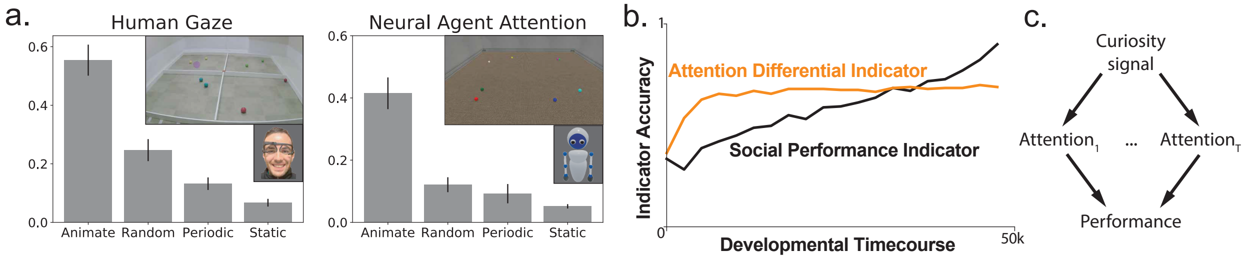


Figure 6: **Modeling human behavior**. (a) Attentional preference in a pilot human behavior study and corresponding model preference. (b) Accuracy of early indicators of final performance, as a function of time. Initially, the Attention Differential Indicator better predicts final performance, but as the time approaches final performance time, the Social Performance Indicator better predicts it. (c) Hypothesized factor diagram: curiosity signal determines attention, which determines final performance.

and reliably directing their gaze to focus on Spheros executing the more complex animate behaviors at the expense of predictable static/period or unpredictable random motion behaviors. They also exhibited a reliable pattern of relative attention across conditions. Comparing these measurements to those from our AWML agent, we found that the attention pattern generated by the γ-Progress policy is similar to that of the humans. While this pilot is too limited to draw any solid conclusion as to which model describes the human data best, it is illustrative of the type of comparisons we aim to make at much finer grain and greater scale in future work.

Eventually, we would like to use the AWML agent as a model of intrinsically-motivated learning in early childhood. Under this interpretation, the learning curves of the AWML agent should correspond to empirical developmental timecourses for the emergence of social attention in children. Improvements in external-agent prediction over time would be compared to empirical measures of changes in child social acuity, while attention allocation timecourses would be compared to how gaze patterns change developmentally.

By extension, the AWML framework might also be used to describe the mechanisms underlying inter-subject variability in social development. In this interpretation, variability in type of curiosity signal could represent a latent cause of developmental variability controlling both social acuity and attention allocation observables. This connection is potentially plausible, since major causes of variability in social development, such as Autism Spectrum Disorder (ASD), are linked to differences in both low-level gaze preferences ([Jones and Klin, 2013]) and high-level social acuity ([Hus and Lord, 2014]). We hypothesize that this apparently wide and disparate spectrum of empirical variability might be accounted for by AWML-based computational model variants — i.e. that ASD might in part be caused by systematic (and potentially genetically-linked [Constantino et al., 2017]) differences between different children's mechanisms for intrinsically-motivated self-supervision.

If correct, such "computational etiology" models could allow the design of model-driven diagnostics. To see what such a possibility might look like at a theoretical level, we consider the problem of how to predict, from early timepoint observations only, what the late (end-of-training) timepoint external-agent prediction performance of an AWML agent will be. Specifically, we estimate a statistical Attention Differential Indicator model, ATT$_{\leq T}$, which takes the agent's animate-inanimate attention differential (the quantity in Fig. 4a) up to time T as input, and outputs predictions for the end-of-training performance (the quantity in Fig. 3 barplots). The agent's curiosity policy is a latent source of variability that is hidden from the ATT$_{\leq T}$. As a baseline, we also trained a Social Performance Indicator model PERF$_{\leq T}$, which takes performance before time T as input. As seen in Figure 6b, ATT$_{\leq T}$ achieves reasonable prediction early in time, and throughout most of the "developmental timecourse" is actually a more accurate indicator of late performance than PERF$_{\leq T}$, the direct measurement of early-stage performance itself. The underlying reason why this occurs is possibly that attention differential is the very mechanism that eventually leads to better performance in better-performing models (e.g. γ-pogress), and (as seen in Fig. 4a) often manifests as an early "bump" in animate attention, allowing the $\mathrm{ATT}_{\leq T}$ predictor to have high SNR. The overall structural equation model is conveyed by the factor diagram Figure 6c — for further details, see Appendix E.1.

Currently, ASD diagnosis is done by expert clinicians, using observations of high-level behaviors ([Hus and Lord, 2014]). This method is subjective, expensive, and often too late — the average diagnosis comes after 4 years of age, often preventing interventions during a critical period of development. It would be of substantial utility if a simple and comparatively easy-to-estimate metric (such as gaze preference), measured early in development, could be used to predict social acuity in late childhood, which is highly salient for ASD outcomes. Translating the above computational analysis into a real-world experimental population could lead to substantial improvements in diagnostics of developmental variability.

Acknowledgements

This work was supported by the McDonnell Foundation (Understanding Human Cognition Award Grant No. 220020469), the Simons Foundation (Collaboration on the Global Brain Grant No. 543061), the Sloan Foundation (Fellowship FG-2018- 10963), the National Science Foundation (RI 1703161 and CAREER Award 1844724), the DARPA Machine Common Sense program, the IBM-Watson AI Lab, and hardware donation from the NVIDIA Corporation.

References

Joshua Achiam and Shankar Sastry. Surprise-based intrinsic motivation for deep reinforcement learning. *arXiv preprint arXiv:1703.01732*, 2017.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. *arXiv preprint arXiv:1705.10528*, May 2017.

Janet W Astington, Paul L Harris, and David R Olson. *Developing theories of mind*. CUP Archive, 1990.

Arthur Aubret, Laetitia Matignon, and Salima Hassas. A survey on intrinsic motivation in reinforcement learning. *arXiv preprint arXiv:1908.06976*, 2019.

Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. In *Advances in neural information processing systems*, pages 4502–4510, 2016.

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, Caglar Gulcehre, Francis Song, Andrew Ballard, Justin Gilmer, George Dahl, Ashish Vaswani, Kelsey Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matt Botvinick, Oriol Vinyals, Yujia Li, and Razvan Pascanu. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, June 2018.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, pages 1471–1479, 2016.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A. Efros. Large-scale study of curiosity-driven learning. In *arXiv:1808.04355*, 2018a.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation, 2018b.

Arunkumar Byravan and Dieter Fox. Se3-nets: Learning rigid body motion using deep neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 173–180. IEEE, 2017.

Michael B Chang, Tomer Ullman, Antonio Torralba, and Joshua B Tenenbaum. A compositional object-based approach to learning physical dynamics. *arXiv preprint arXiv:1612.00341*, 2016.

J. N. Constantino, S. Kennon-McGill, C. Weichselbaum, N. Marrus, A. Haider, A. L. Glowinski, S. Gillespie, C. Klaiman, A. Klin, and W. Jones. Infant viewing of social scenes is under genetic control and is atypical in autism. *Nature*, 547(7663):340–344, Jul 2017.

Julian De Freitas, Kyle Thomas, Peter DeScioli, and Steven Pinker. Common knowledge, coordination, and strategic mentalizing in human social life. *Proceedings of the National Academy of Sciences*, 116(28):13751–13758, 2019.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. *arXiv preprint arXiv:1605.08803*, May 2016.

Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 2786–2793. IEEE, 2017.

Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in Neural Information Processing Systems*, pages 64–72, 2016.

Willem E Frankenhuis, Bailey House, H Clark Barrett, and Scott P Johnson. Infants' perception of chasing. *Cognition*, 126(2):224–233, 2013.

György Gergely, Zoltán Nádasdy, Gergely Csibra, and Szilvia Bíró. Taking the intentional stance at 12 months of age. *Cognition*, 56(2):165–193, 1995.

Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1311–1320. JMLR. org, 2017.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Nick Haber, Damian Mrowca, Stephanie Wang, Li Fei-Fei, and Daniel LK Yamins. Learning to play with intrinsically-motivated self-aware agents. In *Advances in Neural Information Processing Systems*, 2018.

Fritz Heider and Marianne Simmel. An experimental study of apparent behavior. *The American journal of psychology*, 57(2):243–259, 1944.

Ha Hong, Daniel LK Yamins, Najib J Majaj, and James J DiCarlo. Explicit information for category-orthogonal object properties increases along the ventral stream. *Nature neuroscience*, 19 (4):613, 2016.

Rein Houthooft, Xi Chen, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1109–1117. Curran Associates, Inc., 2016.

Vanessa Hus and Catherine Lord. The autism diagnostic observation schedule, module 4: revised algorithm and standardized severity scores. *Journal of autism and developmental disorders*, 44(8): 1996–2012, 2014.

Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. *arXiv preprint arXiv:1611.05397*, 2016.

Warren Jones and Ami Klin. Attention to eyes is present but in decline in 2–6-month-old infants later diagnosed with autism. *Nature*, 504(7480):427–431, 2013.

Nan Rosemary Ke, Amanpreet Singh, Ahmed Touati, Anirudh Goyal, Yoshua Bengio, Devi Parikh, and Dhruv Batra. Learning dynamics model in reinforcement learning by incorporating the long term future. *arXiv preprint arXiv:1903.01599*, 2019.

Celeste Kidd, Steven T Piantadosi, and Richard N Aslin. The goldilocks effect: Human infants allocate attention to visual sequences that are neither too simple nor too complex. *PloS one*, 7(5): e36399, 2012.

Stan Kurkovsky. Android+ sphero: teaching mobile computing and robotics in a single course. In *Proceeding of the 44th ACM technical symposium on Computer science education*, pages 765–765, 2013.

Cam Linke, Nadia M Ady, Martha White, Thomas Degris, and Adam White. Adapting behaviour via intrinsic reward: A survey and empirical study. *arXiv preprint arXiv:1906.07865*, 2019.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. *arXiv preprint arXiv:1811.12359*, 2018.

Daphne Maurer, Richard Le Grand, and Catherine J Mondloch. The many faces of configural processing. *Trends in cognitive sciences*, 6(6):255–260, 2002.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, pages 2125–2133, 2015.

Damian Mrowca, Chengxu Zhuang, Elias Wang, Nick Haber, Li Fei-Fei, Joshua B Tenenbaum, and Daniel L K Yamins. Flexible neural representation for physics prediction. *arXiv preprint arXiv:1806.08047*, June 2018.

Georg Ostrovski, Marc G Bellemare, Aäron van den Oord, and Rémi Munos. Count-based exploration with neural density models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2721–2730. JMLR. org, 2017.

Pierre-Yves Oudeyer, Frdric Kaplan, and Verena V Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 11(2):265–286, 2007.

Pierre-Yves Oudeyer, Adrien Baranes, and Frédéric Kaplan. Intrinsically motivated learning of real-world sensorimotor skills with developmental constraints. In *Intrinsically motivated learning in natural and artificial systems*, pages 303–365. Springer, 2013.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. *arXiv preprint arXiv:1705.05363*, 2017.

Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. *arXiv:1906.04161*, 2019.

J. Piaget. The origins of intelligence in children. 8, 1952.

David Premack and Guy Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences*, 1(4):515–526, 1978.

J. Schmidhuber. Formal theory of creativity, fun, and intrinsic motivation (1990 – 2010). *IEEE Transactions on Autonomous Mental Development*, 2(3):230–247, Sept 2010.

Burr Settles. *Active Learning*, volume 18. Morgan & Claypool Publishers, 2011.

H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294. ACM, 1992.

Kevin Smith, Lingjie Mei, Shunyu Yao, Jiajun Wu, Elizabeth Spelke, Josh Tenenbaum, and Tomer Ullman. Modeling expectation violation in intuitive physics with coarse probabilistic object representations. In *Advances in Neural Information Processing Systems*, pages 8983–8993, 2019.

Bradly Stadie, Sergey Levine, and Pieter Abbeel. Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814*, 2015.

Aimee E Stahl and Lisa Feigenson. Observing the unexpected enhances infants' learning and exploration. *Science*, 348(6230):91–94, 2015.

Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and Philip S Yu. Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. *arXiv preprint arXiv:1804.06300*, 2018.

Henry M Wellman. *The child's theory of mind.* The MIT Press, 1992.

Yueh-Hua Wu, Ting-Han Fan, Peter J. Ramadge, and Hao Su. Model imitation for model-based reinforcement learning. *ArXiv*, abs/1909.11821, 2019.

Appendix

A Connections between General & Conventional Active Learning

Pool-based Active Learning is the same as Query Synthesis Active Learning with the only difference being $\mathcal{A} = \mathcal{D}_{pool}$ where $\mathcal{D}_{pool}$ is the initial pool of unlabelled data.

Stream Active Learning is obtained by choosing $\mathcal{S} = \mathcal{X} \times \mathcal{Y}, \mathcal{A} = \{0, 1\}, P(\cdot|\mathbf{s} = (\mathbf{x}, \mathbf{y}), \mathbf{a}) = \omega(\mathbf{x})$ if $\mathbf{a} = 1$ else $\delta(\mathbf{y}_{dum})$, and $c(\bar{\mathbf{s}} = (\mathbf{s}, H, \theta), \mathbf{a}, \bar{\mathbf{s}}' = (\mathbf{s}', H', \theta')) = \mathcal{L}_{val}(\theta) - \mathcal{L}_{val}(\theta')$, where δ is the Dirac-delta function and $\mathbf{y}_{dum}$ is a dummy label that denotes the case when no label is returned by the oracle.

B Training Details

As shown in Algorithm 13, we interleave world model and policy updates while interacting with the environment. Specifically we update the both the world model and Q-network with 10 gradient steps per 40 environment steps. Both model updates begin after the buffer is filled with 1000 samples.

World Model: We parameterize each component network $\omega_{\theta k}$ with a two-layer Long Short-Term Memory (LSTM) network with 256 hidden units if $|I_k| = 1$ i.e., the causal group k contains a single external agent, and 512 if $|I_k| \geq 2$ to ensure that the size of the parameter space scales with the input and output size. All networks are train using Adam with a learning rate of $1e$-4, $\beta_1 = 0.9, \beta_2 = 0.999$ and batch size 256.

The old model is synchronized with the new model weights once after 100 world model updates. This "warm starts" the old model and prevents unreasonable large progress rewards at the start. We use a fixed value of the progress horizon $\gamma = 0.9995$ across all experiments. We found that any $0.9995 \leq \gamma \leq 0.9999$ attains similar results.

Policy Learning: For Q-network Q_ϕ updates we use the DQN algorithm [Mnih et al., 2015] with a discount factor of $\beta = 0.99$, a boostrapping horizon of 200, a buffer size of $2e5$. Same as the world model, we train the Q-network using Adam with a learning rate of $1e$-4, $\beta_1 = 0.9, \beta_2 = 0.999$ and batch size 256. The policy π_ϕ is an ϵ-greedy exploration strategy with respect to Q_ϕ. Specifically, ϵ is linearly decayed from 1.0 to 0.025 at a rate of 0.0001 per environment step.

C Validation Cases

Here we describe validation protocol for each behavior. As data for the world model must be generated by interacting with the environment, what policy to use during validation is an important choice. As some behaviors are "interactive", i.e the external agent dynamics depend on the curious agent's actions, a naive policy that simply stares at the external agent may not elicit the core dynamics underlying the behavior. Thus, we hard-code the policy during validation to elicit the core dynamics for behavior and subsequently measure world model loss on the collected data.

Peekaboo: The validation policy looks at the peekaboo external agent until it hides. The policy then keeps the peekaboo external agent in view so that when the agent "peeks" it immediately hides again. The validation loss measures the world model performance on predicting the dynamics of this peeking behavior which is representative of the core “interactive” nature of peekaboo.

Reaching: At the start of validation, auxiliary objects are spawned at new locations which changes the trajectory of the reaching external agent. The validation policy then stares at the reaching external agent and validation loss is measured on the collected samples. This validation loss measures how well the world model has learned the contingency between the auxiliary object locations and the reaching external agent's movements. For example, a world model that has overfit to the external agent's trajectory for a particular set of auxiliary object locations will fail to generalize when auxiliary objects are spawned at new locations.

Chasing, Mimicry, Periodic, Static, Noise: The validation policy simply stares at the external agents and validation loss is measured on the collected samples.

The validation losses shown in Figure 3a for the Mixture world is an average of the validation losses on the static, periodic, and animate external agents. The random agent is excluded from evaluation as there is virtually no learnable patterns in the behavior and averaging the large world model loss incurred on the random external agent could occlude the learning performance differences between curiosity signals on the other learnable external agents. For the Noise World, the shown validation losses in Figure 3b represent only the validation loss on the animate external agent.

D Noise World Attention

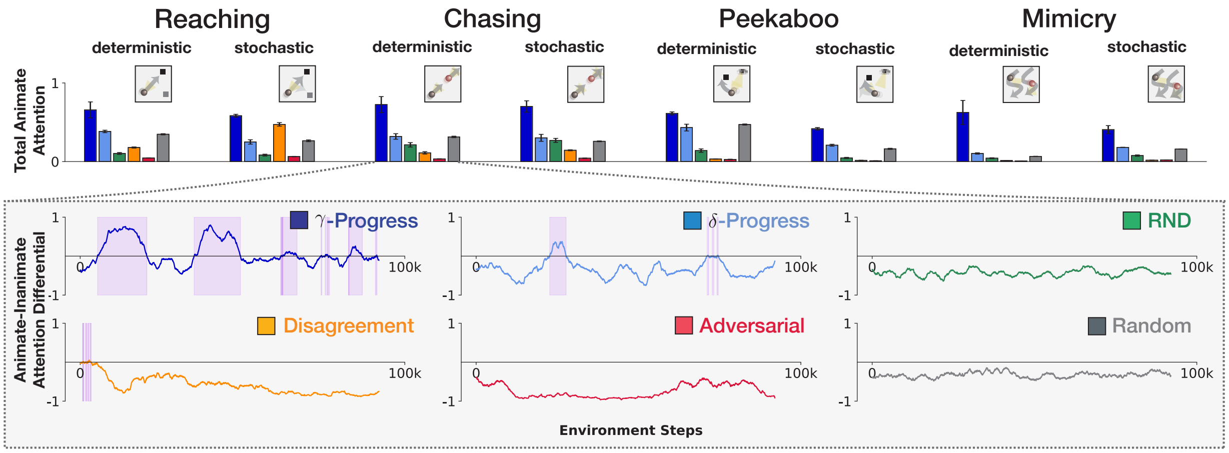


Figure 7: **Attention Patterns in Noise World**. The bar plot shows the total animate attention, which is the ratio between the number of time steps an animate external agent was visible and the number of time steps a noise external agent was visible. The zoom-in box plots show the differences between mean attention to the animate external agents and the mean of attention to the other agents in a 500 step window, with periods of animate preference highlighted in purple. Results are averaged across 5 runs. γ-Progress displays strong animate attention while baselines are either indifferent, e.g δ-Progress, or fixating on white noise, e.g Adversarial.

E Further attention analyses

Here we provide details of the early indicator analysis (Section 7) and a regression of what factors (curiosity signal, architecture, external agent behavior) best predict animate/inanimate attention ratios.

E.1 Details of early indicator analysis

We look to predict final performance P_{final} of a given agent, which we take to be the average of the final four validation runs. To make the modeling problem simple, we discretize this into a classification task by dividing validation performance into 3 equal-sized classes ("high", "medium", and "low", computed separately for each external agent behavior), intuitively chosen to reflect performance around, at, and below that of random policy.

We consider two predictive models of final performance, one that takes as input early attention of the agent, and the other, early performance. Early performance may be quantified simply: given time T ("diagnostic age") during training, let $P_{\leq T}$ be the vector containing all validation losses measured up to time T. Early attention, however, is very high-dimensional, so we must make a dimensionality-reducing choice in order to tractably model with our modest sample size. Hence, we "bucket" average. Given choice of integer B, let

$$A_{\leq T,B} = (f^{\text{anim}}_{0:\frac{T}{B}}, f^{\text{rand}}_{0:\frac{T}{B}}, f^{\text{anim}}_{\frac{T}{B}:\frac{2T}{B}}, f^{\text{rand}}_{\frac{T}{B}:\frac{2T}{B}}, \ldots f^{\text{anim}}_{\frac{(B-1)T}{B}:T}, f^{\text{rand}}_{\frac{(B-1)T}{B}:T}), \tag{13}$$

where $f^{\text{anim}}_{a:b}$ and $f^{\text{rand}}_{a:b}$ are the fraction of the time $t = a$ and $t = b$ spent looking at the animate external agent and random external agents respectively (so $A_{\leq T,B}$ is the attentional trajectory up to time T discretized into B buckets).

Finally, both models must have knowledge of the external agent behavior to which the agent is exposed — we expect this to both have an effect on attention as well as the meaning of early performance and expected final performance as a result. Let χ_{BHR} be the one-hot encoding of which external animate agent behavior is shown.

We then consider models

1. $\text{PERF}_{\leq T}$, which takes as input $P_{\leq T}$ and χ_{BHR}, and

2. $\text{ATT}_{\leq T}$, which takes as input $A_{\leq T,B}$ and χ_{BHR}.

Figure 6b shows the plot of $\text{PERF}_{\leq T}$ and $\text{ATT}_{\leq T}$ accuracy as T varies. We see that, up to a point, $\text{ATT}_{\leq T}$ makes a better predictor of final performance, and then $\text{PERF}_{\leq T}$ dominates. This confirms the intuition that attention patterns precede performance improvements. Intuitively, early attention predicts performance by being able to predict the sort of curiosity signal the agent is using, which predicts the full timecourse of attention (see E.2), which in turn predicts performance.

E.2 Determinants of attention pattern

To gain a finer-grained understanding of what, of the factors we vary (curiosity signal, world model architecture, and stimulus type) drives the attentional behavior of these active learning systems, we perform a linear regression. Specifically, we regress

$$R_{\text{animate/noisy}} = a + b \cdot \chi_{\text{CS}} + c\chi_{\text{causal}} + d \cdot \chi_{\text{BHR}} + \chi_{\text{causal}} * e \cdot \chi_{\text{IM}} + \epsilon \tag{14}$$

Here $R_{\text{animate/noisy}}$ is the ratio of animate to noisy attention, χ_{CS} is a one-hot encoding of curiosity signal (all zeros if random policy), χ_{causal} is an indicator set to 1 if the architecture is causal, χ_{BHR} is a one-hot encoding of animate external agent behavior shown (all zeros if deterministic reaching), and a, b, c, d, e are fixed effects (e measures an interaction effect).

Over 371 individual active learning runs, an ordinary least squares regression achieves an adjusted R^2 of .44. Please see Table 3 for details. We found that γ-Progress receives significant positive weight, while Disagreement and Adversarial receive significant negative weight, with the other curiosity signals having an effect close to that of random policy. In addition, we fail to find a significant effect due to architecture and most external agent behaviors, with two external agent behavior exceptions. In sum, we find that, of the architectural and curiosity signal variations we tested, curiosity signal strongly drives behavior whereas architecture plays an insignificant role.

Table 3: **Attention regression.** Regression model of animate/noisy attention, according to Equation 14. Coefficient values found, and uncorrected p-value for 2-sided t-tests, with significance at the .05 level in bold.

Coefficient	value	P > |t|
constant	.80	.001
γ-Progress	**2.24**	.000
δ-Progress	.08	.788
RND	-.53	.064
Disagreement	**-.70**	.014
Adversarial	**-.79**	.006
Causal architecture	.014	.959
stochastic reaching	.14	.493
deterministic chasing	.25	.222
stochastic chasing	**.45**	.029
deterministic peekaboo	-.08	.682
stochastic peekaboo	.02	.920
mimicry	**.56**	.006
causal $\times\gamma$-Progress	-.32	.408
causal, $\times\delta$-Progress	.06	.868
causal × RND	.03	.935
causal × Disagreement	.23	.555
causal × Adversarial	-.09	.813