# Research Should Promote Understanding

**Manfred Diaz** [1]

*Basically, I'm not interested in doing research and I never have been... I'm interested in understanding, which is quite a different thing. And often to understand something you have to work it out yourself (...).*

— David Blackwell *1983*

## Abstract

## 1. Introduction

Every scientist, or aspiring scientist, is or should strive to be as curious as a seven years old (Gopnik, 1996). Science is the act of formally asking *why*. In this essay, we start asking *why* there is, at a fundamental level, a difference in reinforcement learning (RL) *research* and *understanding*. To many, this is a subtle, uninteresting, and almost philosophical difference: there exists plenty empirical and, in a particular sense, theoretical evidence that progress is being made.

David Blackwell

The question is how to remain curious in an environment where the incentives structures to do so (**?**). incentive structures in academia, life in general, that beat curiosity out

at

### 1.1. A Recurrent Dichotomy

”Every child is an artist. The problem is how to remain an artist once we grow up.” – Picasso

“Every kid starts out as a natural-born scientist, and then we beat it out of them. A few trickle through the system with their wonder and enthusiasm for science intact.” – Sagan

## References

Gopnik, A. The scientist as child. *Philosophy of science*, 63 (4):485–514, 1996.

# Actionable Measures of Machine Intelligence

**Manfred Diaz** [1]

*Basically, I'm not interested in doing research and I never have been... I'm interested in understanding, which is quite a different thing. And often to understand something you have to work it out yourself (...).*

— David Blackwell *1983*

## Abstract

## 1. Introduction

In the last 10 years, the field of reinforcement learning (RL) (Sutton & Barto, 2018) has produced incredible results in game playing (**?**), robotics (**?**), among others, that were out of reach in the 20 years preceding this revolution. Yet, why do we insist here in asking if (or how much) we truly understand the RL problem? Our interest in this proposition stems from calling into question not only the *objective* for RL research but also its *path forward*.

First, *what is the objective of RL research?* There may exist many ways to articulate an answer to this question, but formally, the Legg and Hutter (Legg & Hutter, 2007a) definition of machine intelligence remains to be one of the few attempts to formalize what RL research should aim for. That is, a generalist agent that *performs well* across a *distribution* of environments. In principle, this is a sensitive choice derived from a collection of definitions of human intelligence (Legg & Hutter, 2007b). Then, why don't optimize for this objective? Firstly, this definition leveraged Solomonoff universal prior (Solomonoff, 1960) to score an agent performance. Thus, the final measure depends on the uncomputable *Solomonoff-Kolmogorov-Chaitin complexity* (Li & Vitányi, 2019) of the environments.

On the other hand, even if we agree in an objective, we may further ask *if RL research is producing RL understanding while it moves forward.* Undoubtedly, major contributions to the field have come together with significant theoretical backing () that have nurture in many ways our understanding of the problem. Yet, most if not all impactful results have had a marked algorithmic nature (Mnih et al., 2015; Schulman et al., 2015; Lillicrap et al., 2015; Schulman et al., 2017; Haarnoja et al., 2018; Fujimoto et al., 2018). Thus, *RL research has been primarily driven by algorithmic innovations*.

Nevertheless, in our perspective, some transcendental questions are lingering. Consider, for instance, that we have measured the performance of an algorithm $\mathrm{A}$ on a task $\mathrm{T}_1$. *How much are we able to say about the performance of* $\mathrm{A}$ *on a second task* $\mathrm{T}_2$*?* We contend that relatively little can be said about this beyond intuitive human-centric notions of task complexity. In our view, the scenario described above dictates how RL research is moving forward, explains why benchmarks have proliferated, and also current trends in how research quality is evaluated. While this problem has garnered some attention (Oller et al., 2020; Martínez-Plumed & Hernández-Orallo, 2020; Furuta et al., 2021b), the field still lacks the understanding of how to measure task complexity decoupled from the performance of algorithmic innovations.

Here, we argue that a path forward for RL research could be based on unlocking the most important scientific tool we could have: the ability to measure (Tal, 2020). In particular, we focus on (Legg & Hutter, 2007a) Universal Measure of Machine Intelligence (UMI) and propose two approaches to address its limitations. First, we expand the UMI from the usually narrow view of agent performance interpreted in the traditional RL sense. When the environments rewards come pre-defined by a designer, the UMI is too narrow to capture other forms of machine (or human) intelligence (Cattell, 1963). In Section 2, we introduce the Generalized Measure of Machine Intelligence (GMI) that offers a more flexible perspective on the *objective* of RL research. Then, we leverage this generalization to make GMI an *actionable measure* when we use it as an optimization objective in Section 4 and derive, from first principles, different objectives for generalization in RL (Bengio et al., 2009) that generate some notions of environment complexity.

This is still, however, insufficient to address the uncomputability of $K(\mu)$ that limits the practical utility of UMI. The complexity of environments The framework still have learning built-in hence it is tightly couple with the performance of learning algorithms. We propose to investigate a relaxation of environment complexity specifically tailored to markovian MDPs (Puterman, 1994). In this relaxation,

[1] Mila, University of Montreal. Correspondence to: Manfred Diaz <diazcabm@mila.quebec>.

we put forward an approach that dissects an MDP into its constituent parts and analyze each component's contribution to the overall MDP complexity.

We believe this combination unlocks a unifying approach to several areas of RL research that study generalization, including curriculum, meta and multi-task learning while further..

## 2. A Generalized Measure of Machine Intelligence

The Universal Measure of Intelligence (UMI) (Legg & Hutter, 2007a) is the defined by the expected performance of an agent $\pi$ with respect to the Solomonoff universal prior (**?**) over the space of all computable environments $\mu \in E$. An environment is, in the *Markov Decision Process* framework (**?**) sense, a probability distribution $\mu(s_{t+1}, r_{t+1}|h_{t-1})$ conditioned on the history of agent-environment interactions $h_{t-1} = \{s_{t-1}, r_{t-1}, a_{t-1}, \ldots, s_1, r_1, a_1, s_0\}$. Thus, the Legg & Hutter UMI of an agent (or a policy) $\pi$, denoted by $\Gamma(\pi)$, is defined by the expression:

$$\Gamma(\pi) = \sum_{\mu \in E} 2^{-K(\mu)} V_\mu^\pi \tag{1}$$

where $K(\cdot)$ is the Solomonoff-Kolmogorov-Chaitin complexity (**?**) of the environment $\mu \in E$ defined as the length $l(p)$ of the shortest program $p$ that computes $\mu$ in some reference Universal Turing Machine (UTM) $\mathcal{U}$, such that:

$$K(\mu) = \min_p \{l(p) : \mathcal{U}(p) = \mu\} \tag{2}$$

and $V_\mu^\pi$ is a measure of performance of agent $\pi$ when the environment $\mu$ is reward-summable such that:

$$V_\mu^\pi = \mathbb{E}\left(\sum_{t=1}^{\infty} r_t\right) \leq 1 \tag{3}$$

### 2.1. A Recurrent Dichotomy

[Legg] (pp.10), discusses an earlier work by Horst that goes beyond the performance-based view of intelligence. Horst defines native intelligence as a form of intelligence ”potential”, and I think we can tie this notion nicely with empowerment and a few other ideas. The [Cattell-Horn-Carroll theory]explains the existence of fluid and crystallized intelligences. [Legg] (pp. 4 and 8) discusses that these two types intelligence may be related to children/adolescents and adults different intelligences, having chlidren/adolescents more malleable (fluid) and adults more rigid (crystallized) types of intelligences. Also, Woodrow (see [Legg](pp. 4 and 8) talks about intelligence being ”the capacity to acquire capacity” as another form/definition of intelligence.

These ideas are very close to the discussion in [Klyubin et al, 2005] that justifies the role of empowerment as a universal utility. In the absence of an specified utility function: ”Empowerment can be seen as the agent's potential to change the world, that is, how much the agent could do in principle”. Recent works [Eysenbach et al] have shown that information-theoretic objectives can create, in Cattell's sense, fluid or malleable policies and representations, that are, in many ways, good priors for learning more crystallized policies for downstream tasks. This is similar to how Cattell describes that ”fluid intelligence is a determining factor in the speed with which crystallised knowledge is accumulated” [Cattell, 1963]. If we look at pre-training methods in RL, this fluid to crystal analogy has already gained some ground.

### 2.2. A Generalized Measure of Intelligence

Let's start by generalizing Legg and Hutter measure of general intelligence as:

$$\Gamma(\pi) = \sum_{\mu \in E} p(\mu) F_\mu^\pi$$

#### 2.2.1. Intelligence Beyond

The second component of the generalized measure of intelligence is the measure of an agent's performance on a particular environment instance.

1. Performance-based Intelligence [[Legg & Hutter]](https://arxiv.org/abs/0712.3329)

$$G_\mu^\pi = V_\mu^\pi = \mathbb{E}_{\pi,\mu}\left[\sum_{i=1}^{\infty} r_i\right] \tag{4}$$

2. Native Intelligence [[OURS]]

$$G_\mu^\pi = \max_{p(z)} I_\pi(S, Z) \tag{5}$$

$$G_\mu^\pi = I_\pi(A, S') \tag{6}$$

### 2.3. What can we measure with Intrinsic Intelligence?

1. Natives (Intrinsic) vs Performance: How well does a fluid agent do on the performance-based test, how well does a performance-trained agent does in the fluid test? 1. The Fluid & Crystal Trade-Off: Imagine starting with a policy that maximizes empowerment $\pi^{\&}$ and we start training to optimize for performance. On the other end, imagine we start with an optimal policy $\pi^*$ and start to optimize for empowerment. How do the curves of each learning procedure would look like for each of the intrinsic and extrinsic measures would look like. 1. Pre-training: There have been many methods that have stated to achieve some form of unsupervised/supervised pretrained policies. Can we relate pre-training with native intelligence? Is a high native intelligence a good predictor of posterior fine-tuning performance (i.e., potential)?

## 3. Measures of Complexity

Not all novel benchmarks are not environmental innovations.

, not . In general, it setting the tone

MANFRED: We may be able to argue that it is unclear how much progress is being made, in part, because these measures have not been *actionable* relaxation. However, when approximated they can be linked with

it still has a real impact on RL research and unless we address it every future algorithmic innovation will introduce almost as many questions as it resolves.

This leads to algorithm-environment co-adaptation, also leads to proliferation of benchmarks. How do we know how useful a new benchmark is? Why should we expend compute time

triad environment-algorithm-implementation co-adaptation.

Every known algorithm or the ones that could be created in the future will posit the same question.

we contend that if we truly understand RL we would be able understand with relative ease to answer questions of this nature.

If we do not, we can claim that our measure of progress is relative to

We don't even agree on how to measure algorithmic-based p

(Schaul et al., 2011; Bellemare et al., 2012; Perez-Liebana et al., 2016; Brockman et al., 2016; Beattie et al., 2016; Kempka et al., 2016; Ahn et al., 2019; Cobbe et al., 2019; Juliani et al., 2019; James et al., 2019; Yu et al., 2019; Kurach et al., 2019; Crosby et al., 2020; Samvelyan et al., 2021; Kannan et al., 2021; Fan & Zhu; Freeman et al., 2021)

Recent work (Henderson et al., 2017) has pointed to a crisis of reproducibility in RL research that could be partly blamed on an algorithm-implementation co-adaptation ().

There is hardly a guarantee that a re-implementation of algorithm A performs well in $\mathrm{T}_1$ (Henderson et al., 2017; Engstrom et al., 2020). While Furuta et al. (2021a) argued that there is a co-adaptation of mathematical and algorithmical innovations in RL. Unfortunately, this co-adaptation goes beyond algorithm and implementation: algorithmic innovations are oftentimes co-adapted to the environments or tasks they are tested on.

The original definition of $\Gamma(\cdot)$ proposes an algorithmic probability to weight the importance of each environment. We can generalize this notion to any environment complexity measure (in bits?) we may be able to define, such as:

$$p(\mu) \propto \frac{1}{C(\mu)}$$

For Legg & Hutter, this probability was given by the universal distribution:

$$p(\mu) = 2^{-C(\mu)}$$

where the complexity of the environment $C(\mu)$ could any of the following.

1. Kolmogorov: $C(\mu) = K(\mu)$ [[Legg & Hutter]] 2. Levin: $C(\mu) = K_t(\mu)$

1. I wonder if, beyond what the Universal Solomonoff distribution based on Kolmogorov, we can also use some other approach. For instance, let $C(\mu)$ be a general measure of complexity for $\mu \in M$ where M is the set of MDP environments defined by the transition probability distributions $\mu(s_{t+1}, r_{t+1}|s_t, a_t)$. If we define:

$$p(u) = e^{-C(\mu)}$$

where $C(\mu) \geq 0$, this defines a probability distribution that also decays exponentially with the complexity of the environments. This may not be universal, but it may be universal at least in the problems modelled by MDPs.

2. Doesn't the approach described above also introduce an Occam's razor prior? Simpler environments are weighted more than more complex environments.

3. Measuring vs Training: While I intuitively understand that for evaluation (a measure) of intelligence it is required to weight less complex environments more (i.e., because the performance of the agent is expected to drop), isn't it the contrary when we are training for generalization?

All the project ideas are focused on finding answers for the question: what environment is harder to solve, left or right?. Markov Decision Processes are the de-facto tool to model Reinforcement Learning problems. **Can we turn this intution into a formal explanation of complexity of the environment?** Even if not formal, **can we turn this intuition into some actionable measure of complexity of the environment?**

### 3.1. Analysis of The Complexity of MDPs

We follow a part-to-whole approach to explaining the complexity of Markov Decision Process. The analysis that follows present a decomposition of the MDP problem into its constituent parts: states, action, and rewards. In an MDP definition, the relationship among these three quantities is given by the full transition probability distribution:

$$p(s', r|s, a)$$

from which we can derive any other definitions of MDP functions.

#### 3.1.1. State Space

Provided a fixed reward function (at least semantically: ex. stepping into a free space gives reward 0, bump into a wall

-1, stepping into the goal gives 10) and a fixed action space (semantically: N, S, W, E, with return when stepping into a wall). What impact does the state space structure has over a learning process / complexity of the environment?

![](https://i.imgur.com/cwH1K7W.png)
![](https://i.imgur.com/us2YQ3I.png)

In the figures above, both gridworlds have the same state space size, yet, our intuition would tell us that would be harder for ”any” RL algorithm to solve the environment on the left, than it would be to solve the one on the right.

Subproblem: Exploration

We already investigated this issue in the ICLR 2021 Workshop paper. The environment on the left is harder to explore (under the uniform policy)

#### 3.1.2. Action Space

![](https://i.imgur.com/cwH1K7W.png)

**The von Neumann Agent**

$$A = \{U, D, L, R\}$$

**The Noop Action**

$$A = \{U, D, L, R, O\}$$

where $O$ is the noop (do nothing)

In this case, there is, for instance, no gains in exploration by executing the *noop* action. So, intuitively, a von Neumann agent with a noop action should have a harder problem given the rest of the elements are held constant.

**The Moore Agent**

$$A = \{U, D, L, R\} \cup \{UR, UL, DL, DR\}$$

This is an interesting case because, while the number of actions get duplicated, the ”connectivity” of the state space is significantly increased. For instance, the initial state and the second state of the large corridor (from left to right) are connected by a single action DR, while in the von Neumann agent, two steps were required (D, R) to reach that state from the initial state.

Also, think, for instance, of an optimal policy from the initial state. For the von Neumann agent, an optimal (under a common reward function) policy would required the following sequence:

$$\{S, E, E, E, S\}$$

while, for the Moore agent, an optimal policy would look like:

$$\{SE, E, SE\}$$

So this is an interesting problem. While the increase in number of actions would make the MDP intuitively harder to explore (more actions to try on each state), the actions may re-structure the connectivity of the state space. What does this mean? How can we quantify this problem?

#### 3.1.3. Reward Functions

Provided a fixed state and action space topology. What impact does the reward function has over the learning process?

**Rewards and State Space** The figures below depict two gridworlds (representing two underlying MDPs) for a goal-based RL task whose state and action spaces are identical. If we keep the state and action spaces fixed but vary the reward structure (zero everywhere except at the goal) by simply moving the position of the goal state: which environment would be easier to solve?

![](https://i.imgur.com/uPGId3h.png)
![](https://i.imgur.com/WDQaA5w.png)

Technically, for both MDPs, the reward function could have the same structure:

$$r_{wall} = -1, r_{free} = 0, r_{goal} = 10$$

Intuition tells us that the environment on the left should be easier, as the goal is more ”central” position w.r.t to the rest of the states.

**Density and Sparsity of Reward Functions** Seldom, if ever, the sparsity of the reward function is properly define in RL. What does it mean for a reward function to be sparse? To be 0 almost everywhere? I’d argue that sparsity means some form of ”lack of information”. Wouldn’t be the same if the reward is zero everywhere than if it is -1 everywhere?

Take, for instance, the gridworld depicted on the left figure above (the one with the goal in the center). We can introduce two simple modifications that, intuitively, would reduce to an MDP with the same complexity:

1. When the agent bumps into a wall the reward function returns 0, instead of -1.

$$r_{wall} = 0, r_{free} = 0$$

1. When the agent steps into free space, return -1 instead of 0

$$r_{wall} = -1, r_{free} = -1$$

### 3.2. Intrinsic Complexity of Markov Decision Process

The problems above can be seen as simplifications of a more general problem: that of stablishing a complexity order for Markov Decision Processes. The pictures below propose two gridworld that have a commom acton

![](https://i.imgur.com/7anZ2AB.png)![](https://i.imgur.com/mZhHYKS.png)

#### 3.2.1. Candidate Solutions

**Redefining MDP Observation Space** Let $O = S \times R$ denote the observation space, and let $\pi(|o)$ denote an observation-conditioned policy. Let redefine a trajectory of the policy $\pi$ as:

$$\tau = \{o_0, a_0, o_1, a_1, \dots\}$$

the probability of the trajectory $p(\tau)$ under the $\pi$ is given by:

$$p(\tau) = p(o_0) \prod_{t=0}^{T} \pi(a_t|o_t) p(o_{t+1}|a_t, o_t)$$

What's different? Regularly, the policy $\pi$ is a function $\pi : S \to \Delta(A)$. Now, with this modification, the policy domain space changes $\pi : S \times R \to \Delta(A)$. Also, the transition function changed to $p : S \times R \times A \to \Delta(S \times R)$ when before it was defined as $p : S \times A \to \Delta(S \times R)$.

Why is this change advantageous? Well, now, a policy $\pi$ induces a Markov Chain over the cross-product (observation space) $O = S \times R$ such that:

$$p(o_{t+1}|o_t) = \sum_a \pi(a_t|o_t) p(o_{t+1}|a_t, o_t)$$

**Diversity and Complexity** One of the main problem that plagues Reinforcement Learning research is that of measuring the complexity of the problem an MDP proposes, regardless of the algorithm utilize to solve it. In **Diversity as Complexity** we would like to measure an environment complexity by the number of possible diverse behaviors that can be generated on a given environment.

Complexity of ecosystems.

Each skill is a specie in the ecosystem?

Measuring Diversity of Skills (Behavior)

*Approach 1.* Let $\pi_{z_1}$ and $\pi_{z_2}$ be the policies induced by two skills $p(z = z_1)$ and $p(z = z_2)$, the distance between the two skills could be given by:

$$d(\pi_{z_1}, \pi_{z_2}) = \mathbb{E}_{s \sim p_\pi(s)}[D_f(\pi_{z_1}(\cdot|s) \| \pi_{z_2}(\cdot|s))]$$

where $D_f(\cdot|\cdot)$ is a metric? between probability distributions and $p_\pi(s)$ is the state marginal distribution under a policy. * Q1. Under which policy to take this expectation? This should be a policy that induces a high-entropy state marginal distribution. Tentative solutions: uniform, one of the skills, The problem of the state marginal is that if it is not a high entropy distribution the estimate may be way off (e.g., imagine that the visited states are a hand full). * I1. Can this distribution be learned? If we would like to find the state visitation marginal distribution that maximizes $d(\cdot, \cdot)$: learn $\pi_\lambda$ such that $s \sim p_{\pi_\lambda}(s)$ is such distribution:

$$\max_\lambda \mathbb{E}_{p_{\pi_\lambda}(s)}[D_f(\pi_{z_1}(\cdot|s) \| \pi_{z_2}(\cdot|s))]$$

* I2. We can use Lee et al, [Efficient Exploration via State Marginal Matching](https://arxiv.org/abs/1906.05274) to learn an exploration policy. Then, we compute the expection under the learn policy for any pair of skills. This idea seems more plausible than to learn a policy to discriminate a pair of skills. Is the distribution obtained by SMM close to uniform?

#### 3.2.2. Complexity as Information Diameter and Radius

Why is Jensen-Shannon called information radius [(wiki)](https://en.wikipedia.org/wiki/Jensen

#### 3.2.3. Learning as Contraction and Expansions of Policies

Speed of change in the probability simplices.

Notice how the space of (finite) stochastic matrices is generated from MxN are the cross-product in the product of probility simplices, projection of stochastic matrices into the $[0, 1] \subset \mathbb{R}$.

Environment causes large expansions on the policy. More complex environments causes larger expansions. (see Relative Entropy Policy Search)

#### 3.2.4. Playing Sequential Games

An MDP is a collection of games between two players that can be analyzed in at least this two ways:

1. Two players sequential game where one of the players decides what game (state) is played next. 2. Two players sequential game where the next game(state) played is a move by nature (an external? player with no? interest in the outcome of the game).

The goal of an agent in this game is to accrue tha maximum payoff in all the games played.

Playing sequential games where one agent has the ability to "decide" (or has a private preference) on which game to play next. This is basically an MDP.

Pareto analysis (or equilibrium): a the optimal policy is conditionally in Pareto equilibrium, a change on the conditional distribution on one state can make performance worse in another state.

### 3.3. Generalization

We can derive from the measure of intelligence, several approaches to agents generalization when we convert the measure in the objective (this sounds bad, how's the name of this?)

$$\mathcal{F}(\pi) = \sum_{\mu \in E} p(\mu) G_{\mu}^{\pi} \tag{7}$$

$$\mathcal{F}(\pi) = \mathbb{E}_{u \sim p_{\phi}(u)} \left[ G_{u}^{\pi} \right] \tag{8}$$

## 4. Optimizing for Intelligence

### 4.1. Background

#### 4.1.1. $\mu$-ARMED BANDITS

Fixed a policy $\pi$, the GMI induces a $\pi$-bandit problem over the space of environments. Let the tuple $(\mathcal{A}, \mathcal{R})$ define a multi-arm bandit problewm where $\mathcal{A}$ is the action (or arms) space, and $\mathcal{R}$ is an unknown reward with probability $\mathcal{R}(a) = P[r|a]$. We can convert the GMI into a $\pi$-bandit problem where $\mu \in E$ are actions in the action space $E$ and the unknown reward is the performance $F_{\mu}^{\pi}$ of the policy $\pi$ in the environment $\mu$. Due to the stochasticity of the evaluation of $F_{\mu}^{\pi}$, we can say $r = F_{\pi}^{\mu}$ is a random varaible and we will be interested in the distribution of unknown reward $\mathcal{R}(\mu) = p(r|\mu)$.

In this $\pi$-bandit problem, the action-value function for is given by:

$$Q(\mu) = \mathbb{E}[r|\mu] = \mathbb{E}[F_{\mu}^{\pi}] \tag{9}$$

and computes the expected value of evaluating $\pi$ on the environment $\mu$. Similarly, the maximal action-value function for this problem is given by:

$$V^{+} = \max_{\mu \in E} Q(\mu) \tag{10}$$

and $\mu^{+}$ denotes the environment under which this value is maximized. In a similar vein, we could define the minimal value function such as:

$$V^{-} = \min_{\mu \in E} Q(\mu) \tag{11}$$

Then, we let $\mu^{-}$ denoting the environment for which this value is achieved.

Interaction Protocol

For a fixed policy $\pi$:

1. Sample $\mu \sim \rho(\mu)$ 2. Compute $r = T_{\mu}^{\pi}$ 3. Optimize $\rho(\mu)$

Interpretation: For a fixed policy $\pi$, the optimal environment $\mu^{+}$ is the environment for which the agent performs best, an environment designed to make the agent thrive. This could be a useful tool to retrieve what's the "belief" of the agent with respect to the environment it is executing. On the other hand, the environment $\mu^{-}$ that minimizes the action-value function is the environment in $E$ that it is harder for the agent policy.

#### 4.1.2. CONTEXTUAL BANDIT [WIP]

If we consider the evaluation of multiple policies, we can propose an alternative to the $\pi$-bandit framework under the umbrella of a contextual bandit problem. Let the tuple $(\mathcal{A}, \mathcal{S}, \mathcal{R})$ denote a contextual bandit problem, we could establish an equivalent problem over environment space $(E, \Pi, \mathcal{R})$ where the space of all possible environments $E$ is the action space, the space of all feasible policies $\Pi$ is the context space and with a context-depedent probability reward probability function $R = P(r|\mu, \pi)$, where $r = F_{\mu}^{\pi}$. Also, we should define a probability distribution $\pi \sim p(\pi)$ over the contexts, and a function $\mu \sim \psi(\mu|\pi)$ that specifies a contextual bandit policy.

As we did before, we can defined the action-value function:

$$Q(\pi, \mu) = \mathbb{E}[r|\mu, \pi] = \mathbb{E}[F_{u}^{\pi}] \tag{12}$$

where, in contrast with the $\pi$-bandit problem, the policy $\pi$ is no longer fixed, but sampled i.i.d from the distribution $p(\pi)$

There are a couple of interesting optimization objectives here. First, let $\dot{\pi}$ be a fixed policy, the objective

$$\max_{\mu} Q(\dot{\pi}, \mu) \tag{13}$$

reduces to a $\dot{\pi}$-bandit problem

Interaction Protocol

1. Generate $\pi \sim p(\pi)$ 1. Sample $\mu \sim \psi(\mu|\pi)$ 1. Compute $r = T_{\mu}^{\pi}$ 1. Optimize $\psi(\mu|\pi)$

#### 4.1.3. MARKOV DECISION PROCESS

We can think of extending the previous framework to a full MDP $(\mathcal{A}, \mathcal{S}, \mathcal{R}, \mathcal{P}, \rho)$ with action space $\mathcal{A} = E$, state space $\mathcal{S} = \Pi$, transtion function $P(\pi'|\pi, \mu)$, and a reward distribution $\mathcal{R} = \mathbb{E}[r|\pi, \mu]$, with a reward function $r = F_{\mu}^{\pi}$.

$\tau = \{\pi_0, \mu_0, \pi_1, \mu_1\}$

$$p(\tau) = p(\pi_0) p(\mu_o) \prod_{t=1}^{T} \psi(\mu_t|\pi_t) p(\pi_{t+1}, r_{t+1}|\mu_t, \pi_t) \tag{14}$$

Interaction Protocol 1. Sample $\pi_0 \sim p(\pi)$, $\mu_0 \sim p(\mu)$ 2. For $t \in [0 \dots \infty]$ 2.1 Sample $\mu_t \sim \psi(\mu_t|\pi_t)$ 2.2 Sample $\pi_{t+1}, r_{t+1} \sim p(\pi_{t+1}, r_{t+1}|\mu_t, \pi_t)$ 2.3 Update $\psi(\mu_t|\pi_t)$

Objective

$$G_t = \sum_{t'=t}^{\infty} \gamma^{t-t'} r_t \tag{15}$$

$$Q_\psi(\pi, \mu) = \mathbb{E}_{\pi\sim d_\psi(\pi)\mu\sim\psi(\mu|\pi)} \left[G_t|\mu_t = \mu, \pi_t = \pi\right] \tag{16}$$

$$V_\psi(\pi) = \mathbb{E}_{\pi\sim d_\psi(\pi)} \left[G_t|\right] \tag{17}$$

Let the probability of selecting the environment may be given by a latent structure defined in parameter space $\phi \in \Phi$ such that the probability of each environment is given by $p_\phi(\mu)$.

$$\begin{align}
\mathcal{F}(\pi) &= \sum_{\mu\in E} p_\phi(\mu) G_\mu^\pi \tag{18}\\
\nabla_\phi \mathcal{F}(\pi) &= \sum_{u\in E} \nabla_\phi p_\phi(\mu) G_\mu^\pi \tag{19}\\
&= \sum_{\mu\in E} p_\phi(\mu) \nabla_\phi \log p_\phi(\mu) G_\mu^\pi \tag{20}\\
&= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\phi \log p_\phi(\mu) G_\mu^\pi\right] \tag{21}
\end{align}$$

where this last objective was obtained from applying the [log-derivative trick](https://blog.shakirm.com/2015/11/machine-learning-trick-of-the-day-5-log-derivative-trick/). If we look closer, this objective is an gradient-based bandit objective.

**Control Variates** If, from the last result, we apply the control variates using the performance of the agent on a reference environment $G_{\mu_r}^\pi$ (a constant), we recover a method that is, more general but, in spirit similar to [Active Domain Randomization](https://arxiv.org/abs/1904.04762).

$$\nabla_\phi \mathcal{F}(\pi) = \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\phi \log p_\phi(\mu) (G_\mu^\pi - G_{\mu_r}^\pi)\right] \tag{22}$$

### 4.2. First-order Differentiable General-Sum Game

Let $\pi_\theta(a|s)$ be a parametrized policy on a Markov Decision Process defined by the transition function $\mu(s_t, r_t|s_{t-1}, a_{t-1})$.

$$\max_\theta \max_\phi \mathcal{F}(\phi, \theta) \tag{23}$$

$$\begin{align}
\phi_k &= \phi_{k-1} + \eta_1 \nabla_\phi \mathcal{F}(\phi, \theta) \tag{24}\\
\theta_k &= \theta_{k-1} + \eta_2 \nabla_\theta \mathcal{F}(\phi, \theta) \tag{25}\\
& \tag{26}
\end{align}$$

Simultaneous Gradient Ascent [[Mescheder et al, 2017]](https://arxiv.org/abs/1705.10461), [Ratliff et al, 2014](https://arxiv.org/abs/1411.2168)

#### 4.2.1. Reward-based Reinforcement Learning

The performance of the policy $\pi_\theta$ on the environment $\mu$ is given by:

$$G_\mu^{\pi_\theta} = \mathbb{E}_{s\sim d_\mu^{\pi_\theta} a\sim\pi_\theta} \left[Q_\mu^{\pi_\theta}(s, a)\right] \tag{27}$$

Then, we can re-write the gradient of the general $\mu$-bandit problem for RL as follows:

$$\begin{align}
\nabla_\phi \mathcal{F}(\phi, \theta) &= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\phi \log p_\phi(\mu) G_\mu^{\pi_\theta}\right] \tag{28}\\
&= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\phi \log p_\phi(\mu) \left(\mathbb{E}_{s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[Q_\mu^{\pi_\theta}(s, a)\right]\right)\right] \tag{29}\\
&= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\mathbb{E}_{s\sim d_\mu^{\pi_\theta} a\sim\pi_\theta} \left[\nabla_\phi \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s, a)\right]\right] \tag{30}\\
&= \mathbb{E}_{\mu\sim p_\phi(\mu), s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[\nabla_\phi \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s, a)\right] \tag{31}
\end{align}$$

$$\begin{align}
\nabla_\theta \mathcal{F}(\phi, \theta) &= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\theta G_\mu^{\pi_\theta}\right] \tag{32}\\
&= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\mathbb{E}_{s\sim d_\theta^\pi, a\sim\pi_\theta} \left[Q_\mu^{\pi_\theta}(s, a) \nabla_\theta \log \pi_\theta(a|s)\right]\right] \tag{33}\\
&= \mathbb{E}_{\mu\sim p_\phi(\mu), s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[\nabla_\theta \log \pi_\theta(a|s) Q_\mu^{\pi_\theta}(s, a)\right] \tag{34}\\
& \tag{35}
\end{align}$$

where we took the gradient of the parametrized policy w.r.t to its parameters such that:

$$\begin{align}
\nabla_\theta G_\mu^{\pi_\theta} &= \nabla_\theta \mathbb{E}_{s\sim d_\theta^\pi, a\sim\pi_\theta} \left[Q_\mu^{\pi_\theta}(s, a)\right] \tag{36}\\
&= \mathbb{E}_{s\sim d_\theta^\pi, a\sim\pi_\theta} \left[Q_\mu^{\pi_\theta}(s, a) \nabla_\theta \log \pi_\theta(a|s)\right] \tag{37}
\end{align}$$

NOTE: Interpretation of REINFORCE and the score function.

Questions

1. What happens in this case with ADR and the baseline? Does scoring the baseline higher?

1. Formulate PAIRED in this terms.

### 4.2.2. Second-Order Analysis (WIP)

$$H(\mathcal{F}) = \begin{bmatrix} \nabla_\phi^2 \mathcal{F} & \nabla_\phi \nabla_\theta \mathcal{F} \\ \nabla_\theta \nabla_\phi \mathcal{F} & \nabla_\theta^2 \mathcal{F} \end{bmatrix} \tag{38}$$

NOTE: Not sure if this step is valid. I use linearity of expectation w.r.t. $\nabla_\phi \log p_\phi(\mu)$.

$$\nabla_\phi^2 \mathcal{F} = \mathbb{E}_{\mu\sim p_\phi(\mu), s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[\nabla_\phi^2 \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s,a)\right] \tag{39}$$

$$\nabla_\theta^2 \mathcal{F} = \mathbb{E}_{\mu\sim p_\phi(\mu), s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[\nabla_\theta^2 \log \pi_\theta(a|s) Q_\mu^{\pi_\theta}(s,a)\right] \tag{40}$$

$$\begin{aligned} \nabla_\theta \nabla_\phi \mathcal{F}(\pi) &= \nabla_\theta \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\phi \log p_\phi(\mu) G_\mu^{\pi_\theta}\right] && (41) \\ &= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\phi \log p_\phi(\mu) \nabla_\theta G_\mu^{\pi_\theta}\right] && (42) \\ &= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\nabla_\phi \log p_\phi(\mu) \left(\mathbb{E}_{s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[Q_\mu^{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a|s)\right]\right)\right] && (43) \\ &= \mathbb{E}_{\mu\sim p_\phi(\mu)} \left[\mathbb{E}_{s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[\nabla_\phi \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a|s)\right]\right] && (44) \\ &= \mathbb{E}_{\mu\sim p_\phi(\mu), s\sim d_\mu^{\pi_\theta}, a\sim\pi_\theta} \left[\nabla_\phi \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a|s)\right] && (45) \end{aligned}$$

$$H(\mathcal{F}) = \mathbb{E}_{\mu\sim p_\phi(\mu) s\sim d_\mu^{\pi_\theta} a\sim\pi_\theta} \begin{bmatrix} \nabla_\phi^2 \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s,a) & \nabla_\phi \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a|s) \\ \nabla_\phi \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a|s) & \nabla_\theta^2 \log \pi_\theta(a|s) Q_\mu^{\pi_\theta}(s,a) \end{bmatrix} \tag{46}$$

$$H(\mathcal{F}) = \mathbb{E}_{\mu\sim p_\phi(\mu) s\sim d_\mu^{\pi_\theta} a\sim\pi_\theta} \left[Q_\mu^{\pi_\theta}(s,a) \begin{bmatrix} \nabla_\phi^2 \log p_\phi(\mu) & \nabla_\phi \log p_\phi(\mu) \nabla_\theta \log \pi_\theta(a|s) \\ \nabla_\phi \log p_\phi(\mu) Q_\mu^{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a|s) & \nabla_\theta^2 \log \pi_\theta(a|s) Q_\mu^{\pi_\theta}(s,a) \end{bmatrix}\right] \tag{47}$$

## 4.3. Reward-free Reinforcement Learning

### 4.3.1. Empowerment

$$\mathcal{E}(s) = I(A, S'|S = s) \tag{48}$$

$$G_\mu^\pi = \mathbb{E}_{s\sim d_\mu^\pi} \left[\mathcal{E}(s)\right] \tag{49}$$

## References

Ahn, M., Zhu, H., Hartikainen, K., Ponte, H., Gupta, A., Levine, S., and Kumar, V. ROBEL: Robotics benchmarks for learning with Low-Cost robots. September 2019.

Beattie, C., Leibo, J. Z., Teplyashin, D., Ward, T., Wainwright, M., Küttler, H., Lefrancq, A., Green, S., Valdés, V., Sadik, A., Schrittwieser, J., Anderson, K., York, S., Cant, M., Cain, A., Bolton, A., Gaffney, S., King, H., Hassabis, D., Legg, S., and Petersen, S. DeepMind lab. December 2016.

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. The arcade learning environment: An evaluation platform for general agents. July 2012.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. Curriculum learning. In *Proceedings of the 26th International Conference on Machine Learning*, Montreal, Canada, 2009.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. OpenAI gym. June 2016.

Cattell, R. B. Theory of fluid and crystallized intelligence: A critical experiment. *Journal of Educational Psychology*, 54(1):1–22, 1963.

Cobbe, K., Klimov, O., Hesse, C., Kim, T., and Schulman, J. Quantifying generalization in reinforcement learning. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1282–1289. PMLR, 09–15 Jun 2019.

Crosby, M., Beyret, B., Shanahan, M., Hernández-Orallo, J., Cheke, L., and Halina, M. The Animal-AI testbed and competition. In Escalante, H. J. and Hadsell, R. (eds.), *Proceedings of the NeurIPS 2019 Competition and Demonstration Track*, volume 123 of *Proceedings of Machine Learning Research*, pp. 164–176. PMLR, 2020.

Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Janoos, F., Rudolph, L., and Madry, A. Implementation matters in deep policy gradients: A case study on PPO and TRPO. May 2020.

Fan, L. and Zhu, Y. SURREAL: Open-source reinforcement learning framework and robot manipulation benchmark. https://surreal.stanford.edu/img/surreal-corl2018.pdf. Accessed: 2021-9-1.

Freeman, C. D., Frey, E., Raichuk, A., Girgin, S., Mordatch, I., and Bachem, O. Brax – a differentiable physics engine for large scale rigid body simulation. June 2021.

Fujimoto, S., van Hoof, H., and Meger, D. Addressing function approximation error in Actor-Critic methods. February 2018.

Furuta, H., Kozuno, T., Matsushima, T., Matsuo, Y., and Gu, S. S. Co-Adaptation of algorithmic and implementational innovations in inference-based deep reinforcement learning. *arXiv preprint arXiv:arXiv:2103. 17258*, 2021a.

Furuta, H., Matsushima, T., Kozuno, T., Matsuo, Y., Levine, S., Nachum, O., and Gu, S. S. Policy information capacity: Information-Theoretic measure for task complexity in deep reinforcement learning. March 2021b.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft Actor-Critic: Off-Policy maximum entropy deep reinforcement learning with a stochastic actor. January 2018.

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D. Deep reinforcement learning that matters. September 2017.

James, S., Ma, Z., Arrojo, D. R., and Davison, A. J. RLBench: The robot learning benchmark & learning environment. September 2019.

Juliani, A., Khalifa, A., Berges, V.-P., Harper, J., Teng, E., Henry, H., Crespi, A., Togelius, J., and Lange, D. Obstacle tower: A generalization challenge in vision, control, and planning. February 2019.

Kannan, H., Hafner, D., Finn, C., and Erhan, D. RoboDesk environment v0. `https://github.com/google-research/robodesk`, 2021.

Kempka, M., Wydmuch, M., Runc, G., Toczek, J., and Jaśkowski, W. ViZDoom: A doom-based AI research platform for visual reinforcement learning. May 2016.

Kurach, K., Raichuk, A., Stanczyk, P., Zajkac, M., Bachem, O., Espeholt, L., Riquelme, C., Vincent, D., Michalski, M., Bousquet, O., and Gelly, S. Google research football: A novel reinforcement learning environment. July 2019.

Legg, S. and Hutter, M. Universal intelligence: A definition of machine intelligence. December 2007a.

Legg, S. and Hutter, M. A collection of definitions of intelligence. June 2007b.

Li, M. and Vitányi, P. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer, Cham, 2019.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning. September 2015.

Martínez-Plumed, F. and Hernández-Orallo, J. Dual indicators to analyze AI benchmarks: Difficulty, discrimination, ability, and generality. *IEEE Transactions on Computational Intelligence in AI and Games*, 12(2):121–131, June 2020.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, February 2015.

Oller, D., Glasmachers, T., and Cuccu, G. Analyzing reinforcement learning benchmarks with random weight guessing. April 2020.

Perez-Liebana, D., Samothrakis, S., Togelius, J., Schaul, T., Lucas, S. M., Couëtoux, A., Lee, J., Lim, C.-U., and Thompson, T. The 2014 general video game playing competition. *IEEE Transactions on Computational Intelligence in AI and Games*, 8(3):229–243, September 2016.

Puterman, M. L. *Markov decision processes : discrete stochastic dynamic programming*. Wiley, 1994.

Samvelyan, M., Kirk, R., Kurin, V., Parker-Holder, J., Jiang, M., Hambro, E., Petroni, F., Kuttler, H., Grefenstette, E., and Rocktäschel, T. MiniHack the planet: A sandbox for Open-Ended reinforcement learning research. June 2021.

Schaul, T., Togelius, J., and Schmidhuber, J. Measuring intelligence through games. September 2011.

Schulman, J., Levine, S., Moritz, P., Jordan, M. I., and Abbeel, P. Trust region policy optimization. February 2015.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. July 2017.

Solomonoff, R. J. A preliminary report on a general theory of inductive inference. Technical report, Zator Company, February 1960.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. The MIT Press, 2nd edition, 2018.

Tal, E. *Measurement in Science*. Metaphysics Research Lab, Stanford University, fall 2020 edition, 2020.

Yu, T., Quillen, D., He, Z., Julian, R., Narayan, A., Shively, H., Bellathur, A., Hausman, K., Finn, C., and Levine, S. Meta-World: A benchmark and evaluation for Multi-Task and meta reinforcement learning. October 2019.