A Universal Measure of Intelligence for Artificial Agents*

Shane Legg and **Marcus Hutter**
IDSIA, Galleria 2, Manno-Lugano 6928,
Switzerland
{shane,marcus}@idsia.ch

1 The concept of intelligence

A fundamental difficulty in artificial intelligence is that nobody really knows what intelligence is, especially for systems with senses, environments, motivations and cognitive capacities which are very different to our own. If we look to definitions of human intelligence given by experts, we see that although there is no consensus, most views cluster around a few common perspectives and share many key features.

In all cases, intelligence is a property of an entity, which we will call the *agent*, that interacts with an external problem or situation, which we will call the *environment*. An agent's intelligence is typically related to its ability to succeed in environments, which implies that there is some kind of objective, which we will call the *goal*. The emphasis on learning, adaptation and flexibility common to many definitions implies that the environment is not fully known to the agent. Thus intelligence is the ability to deal with a wide range of possibilities, not just a few specific situations. Putting these things together gives us our informal definition of intelligence:

> *Intelligence measures an agent's general ability to achieve goals in a wide range of environments.*

We are confident that this definition captures the essence of many common perspectives on intelligence. It also describes what we want to achieve in machines: A very general capacity to adapt and perform well in a wide range of situations. In this paper we will try to formalise this view of intelligence.

2 A formal framework

We refer to the signals sent from the agent to the environment as *actions*, and the signals sent back as *perceptions*. Our definition requires there to be a goal for the agent to try to achieve which implies that the agent knows what it is. If the goal was known in advance it could be built into the agent, however this would limit each agent to just one goal. The alternative is to inform the agent of what the goal is. Unfortunately, the possession of a high level of language is too strong an assumption to make. Thus we define a communication channel with very simple semantics: A signal that indicates how good the agent's current situation is, called the *reward*. The agent then simply tries to maximise the amount of reward it receives by learning about the structure of the environment and what it needs to accomplish in order to receive large rewards.

*This work was supported by SNF grant 2100-67712.02.

This is the extremely flexible reinforcement learning framework commonly used in AI, and is equivalent to the controller-plant framework in control theory. In our formulation we will include the reward signal as part of the perception generated by the environment. Thus, as the agent's goal is implicitly defined by the environment, to test an agent in any given way it is sufficient to fully define its environment.

Formally, the agent sends information to the environment by sending *symbols* from some finite set called the *action space*, for example, $\mathcal{A} := \{\texttt{up}, \texttt{down}, \texttt{left}, \texttt{right}\}$. The environment responds with symbols from a finite set called the *perception space*, denoted $\mathcal{P}$. The *reward space*, denoted by $\mathcal{R}$, is a finite subset of $[0,1] \cap \mathbb{Q}$. Every perception consists of two separate parts; a reward and a non-reward part called an *observation*. For symbols being sent we will use the lower case variable names o, r and a for observations, rewards and actions respectively. We index these in the order in which they occur, thus a_3 is the agent's third action. The agent and the environment take turns at sending symbols, starting with the environment. This produces an increasing history of observations, rewards and actions, $o_1 r_1 a_1 o_2 r_2 a_2 o_3 r_3 a_3 o_4 \ldots$.

The agent is a function, denoted by π, which takes the current history as input and chooses the next action as output. We represent this as a probability measure over actions conditioned on the current history, for example, $\pi(a_3|o_1 r_1 a_1 o_2 r_2)$. The workings of the agent is unspecified, though in artificial intelligence the agent will be a machine and thus π will be a computable function. The environment, denoted μ, is similarly defined: $\forall k \in \mathbb{N}$ the probability of $o_k r_k$, given the current history is $\mu(o_k r_k|o_1 r_1 a_1 o_2 r_2 a_2 \ldots o_{k-1} r_{k-1} a_{k-1})$.

The agent must try to maximise the total reward it receives, however what this means depends on how we value rewards at different points in the future. The standard way of expressing this is to weight the future reward at time i by a factor γ_i. Thus the future value is $V^{\pi\mu} := \mathbf{E}\big(\sum_{i=1}^{\infty} \gamma_i r_i\big)$, where r_i is the reward in cycle i of a given history, and the expected value is taken over all possible interaction histories of π and μ. The choice of γ_i is a subtle issue that controls how greedy or far sighted the agent should be. Here we use the near-harmonic $\gamma_i := 1/i^2$ as this produces an agent with increasing farsightedness of the order of its current age [Hutter, 2004].

As we desire an extremely general definition of intelligence for arbitrary systems, our space of environments should be as large as possible. An obvious choice is the space of all probability measures, however this causes serious problems

as we cannot even describe some of these measures in a finite way. The solution is to require the measures to be computable. This allows for an infinite space of possible environments with no bound on their complexity. It also permits environments which are non-deterministic as it is only their distributions which need to be computable. This space, denoted E, appears to be the largest useful space of environments.

3 A formal measure of intelligence

We want to compute the general performance of an agent in unknown environments. As there are an infinite number of environments, we cannot simply take a uniform distribution over them. If we consider the agent's perspective on the problem, this is the same as asking: Given several different hypotheses which are consistent with the data, which hypothesis should be considered the most likely? This is a standard problem in inductive inference for which the usual solution is to invoke Occam's razor: *Given multiple hypotheses which are consistent with the data, the simplest should be preferred.* As this is generally considered the most intelligent thing to do, we should test agents in such a way that they are, at least on average, rewarded for correctly applying Occam's razor. This means that our a priori distribution over environments should be weighted towards simpler environments.

As each environment is described by a computable measure, one way of measuring the complexity of an environment is by taking its Kolmogorov complexity. If $\mathcal{U}$ is a prefix-free universal Turing machine then the Kolmogorov complexity of an environment μ is the length of the shortest program on $\mathcal{U}$ that computes μ, formally $K(\mu) := \min_p\{l(p) : \mathcal{U}(p) = \mu\}$. Unfortunately, K is not computable and is provably difficult to approximate. For the purposes of Occam's razor, it also seems philosophically unnatural to consider short programs which require an enormous amount of time to compute to be "simple". We can address both of these problems by using a notion of complexity that takes execution time into account, such as Kt complexity [Levin, 1973]. Formally, $Kt(\mu) := \min_p\{l(p) + \log t(p) : \mathcal{U}(p) = \mu\}$ where $t(p)$ is the number of steps required to compute μ on $\mathcal{U}$. This gives us a computable distribution $2^{-Kt(\mu)}$ over our space of possible environments which is consistent with the notion that very simple algorithms should be short and fast to compute. Another alternative is the Speed Prior [Schmidhuber, 2002].

We can now define the *universal intelligence* of an agent π to simply be its expected performance when faced with an unknown environment sampled from this distribution,

$$\Upsilon(\pi) := \sum_{\mu \in E} 2^{-Kt(\mu)} V^{\pi\mu}.$$

4 Properties of universal intelligence

It is clear by construction that universal intelligence measures the general ability of an agent to perform well in a very wide range of environments, as required by our informal definition of intelligence given earlier. The definition places no restrictions on the internal workings of the agent; it only requires that the agent is capable of generating output and receiving input which includes a reward signal. Universal intelligence also reflects Occam's razor in a natural way that respects both the minimal description and computation time of an environment. Indeed it is similar to intelligence tests for humans which usually define the correct answer to a question to be the simplest consistent with the given information. Although our space of possible environments E is infinite, as Kt is computable, $\Upsilon(\pi)$ can also be computed.

By considering $V^{\pi\mu}$ for a number of basic environments, such as small MDPs, and agents with simple but very general optimisation strategies, it is clear that Υ correctly orders the relative intelligence of these agents in a natural way. If we consider a highly specialised agent, for example IBM's Deep-Blue chess super computer, then we can see that this agent will be ineffective outside of one very specific and complex environment, and thus would have a very low universal intelligence value. This is consistent with our view of intelligence as being a highly adaptable and general ability.

A very high value of Υ would imply that an agent was able to perform well in many environments. Such a machine would obviously be of large practical significance. If we replace near-harmonic discounting with a finite length horizon and ignore computation time, it is possible to define an order relation between agents over interaction histories, known as the Intelligence Order Relation (IOR) [Hutter, 2004]. The maximal agent with respect to this order relation is AIXI, and with minor adjustments, it would also be maximal with respect to Υ. AIXI has been shown to have many optimality properties, including the ability to be self-optimising in environments in which this is at all possible [Hutter, 2004]. These results demonstrate the power of agents which rate highly with respect to the IOR and the related Υ defined here.

Clearly Υ spans simple adaptive agents right up to super intelligent agents like AIXI, unlike the pass-fail Turing test which is useful only for agents with near human intelligence. Moreover, the Turing test is highly anthropomorphic, with many suggesting that it is a test of humanness rather than intelligence. We have avoided this problem of human bias, as well as the need for human judges, by basing our definition on the fundamentals of information and computation theory.

The only related work to ours is the C-Test [Hernández-Orallo, 2000]. While Υ is an interactive test, the C-Test is a static sequence prediction test which always ensures that each question has an unambiguous answer. We believe that these are unrealistic and unnecessary assumptions. The C-Test was able to compute a number of usable test problems which were shown to correlate with real IQ test scores for humans.

References

[Hernández-Orallo, 2000] J. Hernández-Orallo. Beyond the Turing test. *Journal of Logic, Language and Information*, 9(4):447–466, 2000.

[Hutter, 2004] M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2004.

[Levin, 1973] L. A. Levin. Universal sequential search problems. *Problems of Information Trans*, 9:265–266, 1973.

[Schmidhuber, 2002] J. Schmidhuber. The Speed Prior: A new simplicity measure yielding near-optimal computable predictions. In *Proc. COLT 2002*, Lecture Notes in Artificial Intelligence, pages 216–228, July 2002. Springer.