Minds & Machines manuscript No.
(will be inserted by the editor)

On *Potential* Cognitive Abilities in the Machine Kingdom

José Hernández-Orallo · David L. Dowe

Received: July 28th, 2012 / Accepted: December 11th, 2012

Abstract Animals, including humans, are usually judged on what they could become, rather than what they are. Many physical and *cognitive* abilities in the 'animal kingdom' are only acquired (to a given degree) when the subject reaches a certain stage of development, which can be accelerated or spoilt depending on how the environment, training or *education* is. The term '*potential* ability' usually refers to how quick and likely the process of attaining the ability is. In principle, things should not be different for the 'machine kingdom'. While machines can be characterised by a set of cognitive abilities, and measuring them is already a big challenge, known as 'universal psychometrics', a more informative, and yet more challenging, goal would be to also determine the potential cognitive abilities of a machine. In this paper we investigate the notion of potential cognitive ability for machines, focussing especially on universality and intelligence. We consider several machine characterisations (non-interactive and interactive) and give definitions for each case, considering permanent and temporal potentials. From these definitions, we analyse the relation between some potential abilities, we bring out the dependency on the environment distribution and we suggest some ideas about how potential abilities can be measured. Finally, we also analyse the potential of environments at different levels and briefly discuss whether machines should be designed to be intelligent or potentially intelligent.

Keywords cognitive abilities · machine intelligence measurement · (universal) Turing machines · universality probability · potential intelligence · (universal) psychometrics

José Hernández-Orallo
DSIC, Universitat Politècnica de València, València, Spain. E-mail: jorallo@dsic.upv.es

David L. Dowe
Computer Science and Software Engineering, Clayton School of Information Technology, Monash University, Vic. 3800, Australia. E-mail: david.dowe@infotech.monash.edu.au

1 Introduction

In about the last fifteen years, there have been several efforts to give formal definitions, measures and tests of intelligence based on computation theory and (algorithmic) information theory [10,8,9,23,15,16,38,33,18,11,21,22]. All of these works have worked on the notion of *actual* intelligence, i.e., the intelligence which is measured over a system at a particular stage of its development (or a particular moment of its life). If this is not already a very difficult question, things become even more complex when we try to evaluate *potential* intelligence, which can be loosely defined (for now) as the capacity that a system has to eventually become intelligent, where the terms 'capacity' and 'eventually' will be understood respectively as 'probability' and 'in a given future time under a range of circumstances'.

Small human children are said to be potentially intelligent even though their actual intelligence is very low compared to an adult's. In fact, an adult rat has higher actual intelligence than a new-born baby, for whom perception and reaction are *still* inoperative or very primitive. Potential intelligence is linked to the notions of development environment and education, and also to the nature vs. nurture dilemma. In other words, having the *potential* does not mean that this potential will ever be attained (or realised). A very talented child can be spoilt with an inappropriate education, while another, less talented, child can be boosted with the appropriate, specialised education. In this natural context, the notion of potential makes sense, as either a limit that a subject can attain or the easiness (in terms of education or environment) to reach a given level.

Things start becoming more interesting (but perhaps counter-intuitive, as well) when we move from biological systems to artificial systems. Let us consider, for the moment, universal Turing machines (UTMs). And let us assume, for the moment, that intelligence is not a score, but a binary property (a system is either intelligent or not, by, e.g., setting a threshold relative to which we consider a system intelligent). Let us also assume that this property is computably realisable[1]. Under these conditions, as we will show (in a more formal, straightforward way), any UTM can become intelligent. If we define potential intelligence as the *possibility* of reaching actual intelligence, then we have that any UTM is potentially intelligent. This is not very informative, since we know that some (universal) machines are potentially more intelligent than others. This shows that the notion needs to be refined to be more useful.

The way-out from here is the definition of potential intelligence as the *probability* that a machine becomes intelligent for a random input (or education, or life, or "possible world"). With this definition we could say that a machine is more potentially intelligent than another if we are able to show (theoretically, or empirically) that it becomes intelligent more frequently (or with shorter inputs).

[1] We use the term 'computably realisable' to express that there is at least one (Turing) machine with the property. This does not mean that determining whether a machine has the property is decidable. All this will be further clarified in section 2.3.

Intelligence is not the only cognitive ability or property we will be interested in. In the past, the probability of UTMs acquiring or preserving a property has been studied a few times. For instance, the probability of a UTM's halting for a random input was first investigated by Zvonkin and Levin[2] [59], and also by Chaitin [5], and very important results about randomness were obtained.

More recently, another property, and the probability of a UTM's preserving it, has been studied. This is the *universality probability*[3], or 1 minus the probability that a UTM loses its universality (becomes non-universal) after feeding it with a random input. It has been shown in [3] that this probability is strictly between 0 and 1 for any UTM. The universality probability was first suggested by Chris Wallace [6, footnote 70][7, sec. 2.5] with the intuition of whether an 'educated machine' could lose its capacity to *learn*. Certainly, universality and capacity to learn are related, but they are not the same thing. In fact, the capacity to learn is more closely related to intelligence than universality.

The notion of universality probability and the results obtained in [3] may have implications concerning (or may be helpful for addressing) the notions of potential cognitive abilities in general, and intelligence in particular. This is the starting point of this paper. This is also an important source of hindrances, but also opportunities, throughout the paper, since universal machines are able to become intelligent machines (with some probability), and intelligent machines are, arguably, able to imitate any other machine and, hence, universal, in a slightly different sense.

This relation between intelligence and universality shows how important it is to realise that it is one thing to *become* a different machine and another thing to *imitate* or *model* another machine *for a while*. Also, it is important to distinguish between an individual agent and a whole system, which may have subsystems inside having a property.

The analysis of all these fundamental questions concerning potential cognitive abilities, universality and resource-bounded machines is the main goal of this paper.

The paper is organised as follows. Section 2 discusses some previous works on measuring actual abilities for machines, like intelligence, several notions of 'potential' intelligence in psychology, the ideas of training sequences and educating Turing machines, the notion of cognitive ability/property, and the notion of universality probability of a Turing machine and the generalisation of the permanent preservation of any property. Section 3 introduces the definition of potential of a property for non-interactive (classical) Turing machines, and highlights the relevance of the input distribution and the number of steps considered (the temporal period), in order to make sense of the definition. From the temporal notion of potential we make the first important distinction between TMs becoming or imitating another TM. Section 4 extends and adapts some of the previous definitions for computable agents, i.e., interac-

[2] Possibly even earlier by Martin-Löf, from Levin's personal communication.

[3] Note the difference with the concept of "universal probability" (distribution), as introduced by Solomonoff [45].

tive Turing machines inside an environment. This brings out a more realistic perspective, also including speed, and more sophisticated relations between potential and the distribution of environments. Section 5 deals with the challenge of evaluating *potential* properties. We are interested in how the potential can be approximated from behaviour, not from internal introspection. Section 6 discusses related, but different, notions, such as the exploration/exploitation dilemma, which is a recurrent issue in reinforcement learning, and the distinction between fluid vs. crystallised intelligence, a crucial concept in psychometrics. Section 7 closes the paper with a discussion of potential in terms of 'emergence' in complex systems, some open questions and the implications for building intelligent machines.

2 Background

The evaluation of cognitive abilities for humans and non-human animals can be traced back to the now consolidated disciplines of psychometrics and comparative psychology. There is also a large and important body of work comparing abilities between humans and non-human animals, and the hybridisation between both disciplines is becoming stronger (see, e.g., [24,25]). However, generalising cognitive abilities for different species is not easy, since the assumptions about the required abilities and the proper interfaces required to evaluate each individual is always at issue. From a scientific point of view, and most especially from an evolutionary stance, it makes sense to evaluate the cognitive abilities of any subject in the 'animal kingdom' (including humans) at any stage of its development.

2.1 Evaluating cognitive abilities in the machine kingdom

If things were not already complex enough for the animal kingdom, there is a diverse new realm that is still unexplored: the 'machine kingdom', i.e., the set of all machines[4]. This uncharted space is much more complex than the animal kingdom, because we can define a machine to behave in virtually any possible way, including emulating any animal. The only constraints are computability and resources. Clearly, in order to assess the behaviour of a plethora of machines, bots, robots, artificial agents, avatars, animats, any other artificial life beasts and hybrids and communities thereof, we require a powerful set of cognitive tests. This is precisely the goal of a proposed new discipline, 'universal psychometrics' [19,21]. While part of the methods and concepts can be borrowed from psychometrics and comparative psychology, universal psychometrics has its formal grounds in the works on machine intelligence evaluation that have taken place in the past fifteen years or so.

[4] In the first part of the paper we will consider non-interactive (classical) Turing machines, while in the second part we will refer to the set of interactive (and resource-bounded) machines, as in [21].

These works are not based on Turing's imitation game [52] or its many extensions (see, e.g., [40]), but on the notions of learning, inductive inference, Turing machines and compression. In particular, Solomonoff's theory of inductive inference [45], the Minimum Message Length (MML) principle [55,56, 54,7], algorithmic information theory [4], Kolmogorov complexity [31,45] and compression theory paved the way in the 1990s for a new approach for defining and measuring intelligence based on algorithmic information theory and two-part compression.

The first proposal introduced an *induction-enhanced* Turing Test [9], where a general inductive ability could be evaluated. The importance was not that any kind of ability could be included in the Turing Test, but that this ability could be formalised in terms of MML and cognate ideas, such as (two-part) compression. Related intelligence tests were also developed, such as the C-test [23] [15], composed of sequences of prediction problems that were generated by a universal distribution [45] and their difficulty assessed by a variant of Kolmogorov complexity. Other cognitive abilities were addressed in [16], by the introduction of other 'factors', and the suggestion of using interactive tasks where "rewards and penalties could be used instead", as in reinforcement learning.

Similar ideas followed relating compression and intelligence, such as [38], and the 'universal intelligence measure' [33], where intelligence is seen as weighted average performance in a range of environments, where the environments are just selected by a universal distribution.

Some more recent works have focussed on the construction of actual tests and their use for evaluating machines and humans in the same way. For instance, the anytime intelligence test in [18] could be applied to any kind of subject: machine, human, non-human animal or a community of these. The term *anytime* was used to indicate that the test could evaluate any agent speed, it would adapt to the intelligence of the examinee, and that it could be interrupted at any time to give an intelligence score estimate. Preliminary tests based on these ideas have since been done and applied to the evaluation of humans [15] and machines [29,34], and the comparison of both humans and machines [28].

For a more comprehensive view of this line of research and its relation with other approaches, such as human psychometrics and the Turing Test, the reader can see [11,21,22].

All the previous approaches focus on actual cognitive abilities, such as induction, deduction, planning, etc. Following [21], we can give the following definition of cognitive ability:

Definition 1 A *cognitive ability* is a property of individuals in the machine kingdom which allows them to perform well in a *class* of information-processing tasks.

A class is a (possibly infinite) set of problems. From each class, in order to construct a *test*, we can sample tasks using a distribution, which does not need to be uniform (in fact, in many cases, it cannot be defined as a uniform

distribution). For instance, we can define the ability of multiplying two natural numbers. If we were to select some specific tasks from there, we would need a distribution to give more or less probability to some numbers. For instance, we could precisely define the ability as the correct multiplication of numbers of 3 digits (where all of them could have the same probability) as a first (example or) case. As a second case, we could precisely define the ability of multiplying numbers of n and m digits, where n and m are taken as the closest natural number from a truncated normal distribution with mean at 3 and standard deviation 1 (and then generating each of the m and n digits respectively in a uniform way[5]). In the first case, the number of possible exercises is finite while it is infinite in the second case. In what follows, we assume that a class is a set of information-processing tasks (or environments) possibly with an associated probability distribution.

Note that actual abilities are linked to performance and, ultimately, to the observational demonstration of the ability, and not determined by any solely intrinsic property of the internal code of the individual (its program). To our knowledge, there has not been any attempt to define potential abilities and consider their evaluation *on machines*, perhaps because, at first sight, this seems to require the inspection of the machine code.

2.2 Previous notions of potential

Unlike artificial intelligence, the term 'potential' has already been used in psychology and other disciplines[6]. For instance, in psychometrics the terms 'potential', 'capacity' and others have been used for "differentiating a measured intelligence score from some higher score which an individual is presumed capable of obtaining" [39]. A 'potential' ability is then understood as the maximum score that an individual can score on a test of that ability. Clearly, this is an issue related to measuring error produced by how tests are conducted. Typically, tests not only require the co-operation of the subject but also a great degree of implication and motivation. For instance, a very intelligent subject can score poorly at an intelligence test if she is not properly motivated (e.g., rewards are not appropriate or well understood —see also [46, sec. 6]) or any other problem with the interface (e.g., language, background knowledge, perception limitations, etc.). This is a great concern in the evaluation of animal abilities, since it is a frequent discovery to find that some animals do have an ability that was previously considered absent in these animals just because no proper test had been devised to accurately measure the ability for that species. This difference between the actual result of a test and the maximum

[5] The first digit of each number would be generated uniformly from 1 to 9 and the remaining digits would each be generated uniformly from 0 to 9.

[6] In fact, the distinction between potentiality and actuality can be traced back to Aristotle's Methaphysics, in book Θ (IX) [2], with his distinction between potentiality (*dunamis*) and actuality (*entelecheia* or *energeia*). In part 6 he says: "potentially, for instance, a statue of Hermes is in the block of wood [...], and we call even the man who is not studying a man of science, if he is capable of studying".

achievable result, and the fact that the former is usually lower than the latter, leads to considering the result of a cognitive test as a *lower bound* of the actual ability. Note that with this interpretation, the 'potential' would be the right (or corrected) value of an ability, while measurements would be approximations, which are typically —but not always— below that value. This has led to approaches to convert this lower bound into a less biased estimate, trying to predict 'potential' intelligence [51] or calculating how far a test score can be from the actual measure [37]. In a nutshell, the difference between actual and 'potential' would really be applied to the measurement, but not to the individual.

The meaning of potential that we will use in this paper differs from the measurement 'potential' described above. We will deal with the *probability* that a system or individual *acquires (or reaches a certain level for)* a given ability. This is clearly a notion related to the *state* of a system and not about the test or measurement error. This state can change by inner mechanisms or can be induced by outer mechanisms (or both). Note that we use the term 'probability' instead of 'capacity', which is a less precise term and usually associated with the *maximum* value, while probability is associated with the average or *expected* value.

Let us think, for a moment, about non-cognitive abilities or traits, because the concept of potential is simpler. For instance, human height growth has been commonly used as a parallel to the development of other cognitive abilities, such as intelligence. Consider that we measure the height of a 6-year-old child and get an actual height of 120 cm. What is her potential height? We would certainly not give 248 cm (the maximum value recorded in history for a woman) as an answer, even though this is physically possible (or even feasible in general, by using drugs). The question is typically understood as the height that she is *expected* to reach as an adult woman *under a range of circumstances*. This is the concept of potential we are using in this paper. Crucially, we need to identify two important things in this concept. First, we are talking about the expectation for a future time and, second, we are considering some particular or general circumstances. For instance, we could just ask a different, more specific (second) question: what is her expected height at age 10 assuming a diet poor in calcium and vitamin D? In this case both the time and the context have been modified, and the answer should be different. This is the parameterised concept of potential we want to explore in this paper.

Remarkably, a different thing is the way to answer these questions. In other words, one thing is to define potential and another thing is to measure it. For instance, for human height we have some tools, such as a growth chart, as shown in Figure 1. Looking at that chart we can get a rough estimation for the answer to the first question (e.g., 170 cm) and perhaps, with some extra knowledge, to the second question as well.

Finally, note that the potential property in a future time does not have to match the actual property at that time once the time has passed. For instance, the potential height of this 6-year-old child under normal circumstances can be said to be around 170 cm. However, if, after 20 years, we measure her height

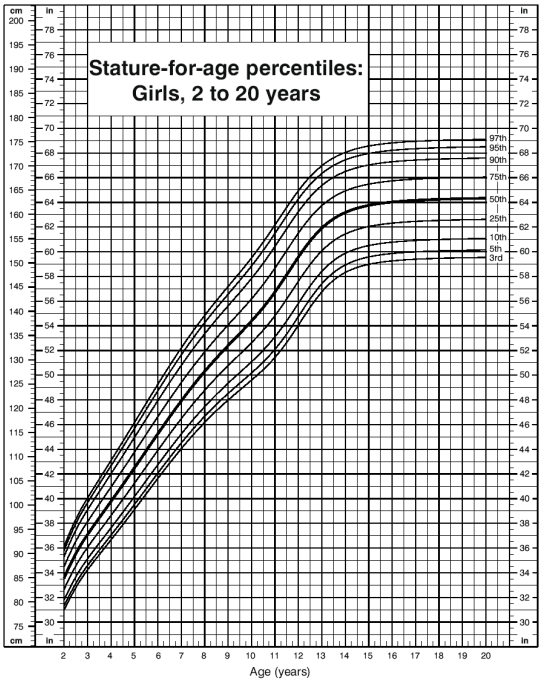

Fig. 1 A growth chart of stature with percentiles of U.S. females aged 2-20 (NCHS 2000). Potential height at a future age under normal circumstances can be roughly estimated from this chart. *Taken under public licence from wikimedia commons from an original U.S. government chart.*

and we record 155 cm, this does not necessarily mean that the potential height of 170 cm was wrong. Perhaps she had a serious illness, or an accident or a bad nutrition record. Even knowing her actual height as an adult we would still say that her potential height was 170 cm, had she grown up under normal circumstances. In fact, we would say that her height is 155 cm, while it was *expected* to be 170 cm.

For cognitive abilities, things become more subtle, especially because we almost always associate some learning process with these kinds of abilities. In animals, there are some cognitive abilities which are not learned (they are innate), such as a frog distinguishing small dark spots from big dark spots. They may even be there from birth, with no variation whatsoever during the animal lifespan. Some others may appear after a time, as a *programmed* development. For instance, a new-born baby may not be able to recognise colours, but this ability may develop in a few weeks' time. If the baby is fed and cared for adequately, this ability will develop without further training or conditions. On the contrary, other abilities are not innately programmed (i.e., have to be acquired). For example, a person may not be able to calculate square roots now, but she can learn to do it and have the ability after some time. Clearly, in this case, acquiring the ability requires some particular specialised training. This gives a complementary (and essential) perspective for the notion

of potential. Some abilities can only be acquired with appropriate training environments.

In fact, things become really interesting when we bring these concepts from the animal kingdom to the machine kingdom. The study of the so-called *training sequences* for Turing machines was first discussed by Solomonoff [44] (possibly influenced by Turing [52, sec. 7][7]). Also, the notion of perfect sequence, as a sequence of exercises that makes a system acquire or learn a concept with the minimum amount of effort or information, was further studied by Solomonoff [48]. Clearly, finding these sequences is not easy, as any teacher knows. Similarly, Wallace also considered the problem of 'educating' Turing machines and several problems related to this issue [54, sec. 2.3][6, footnote 70][7, sec. 2.5], and also gave at least some thought directly to training sequences [6, sec. 0.2.5, p542, col. 1].

While we will mostly deal with *individuals* acquiring, increasing, preserving, decreasing or losing an ability, the term potential can also be applied to systems for which a given property develops or emerges *inside* the system (on some of its parts). For instance, we can ask whether a given initial pattern in Conway's game of life [13] would eventually lead to substructures with self-replicating power. This does not mean that the pattern is self-replicating, but rather that it leads to self-replicating structures. We will go back over these issues later on, but for the moment it is important to be clear how we use the term potential, and the accompanying verb(s) —such as becoming, imitating, emulating, hosting, preserving, etc.

2.3 Properties, universality and preservation

Let us denote by $\mathbb{B}$ the set $\{0,1\}$ and by $\mathbb{B}^*$ the set of finite binary *strings* of any length, including the empty string λ. The length of a string $\sigma \in \mathbb{B}^*$ is denoted by $|\sigma|$. We can restrict the range of strings by their length, where $\mathbb{B}^{m:n}$ denotes all the strings $\sigma \in \mathbb{B}^*$ such that $m \leq |\sigma| \leq n$. We use $\mathbb{B}^n$ as shorthand to denote $\mathbb{B}^{n:n}$. We will also work with infinite *sequences*. The set of all infinite sequences will be denoted by $\mathbb{B}^\infty$. If σ is a finite string in $\mathbb{B}^*$ or an infinite sequence in $\mathbb{B}^\infty$, we use $\sigma_{a:b}$ to denote the finite substring between positions a and b inclusive (so having length $b-a+1$). If $a > b$ then $\sigma_{a:b} = \lambda$. Given a finite string σ and a finite string or infinite sequence τ, the concatenation is simply denoted by $\sigma\tau$. The (cylinder) set of all the infinite sequences in $\mathbb{B}^\infty$ starting with finite string σ is denoted by $\sigma\circ$.

Any (possibly partial) computable function $M : \mathbb{B}^* \to \mathbb{B}^*$ can be calculated by a Turing machine (TM), which we shall also call M. In order to properly analyse the concept of universal Turing machine (UTM), we can, without

[7] Section 7 of Turing (1950), entitled "Learning Machines", says: "Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain. [...] We have thus divided our problem into two parts. The child programme and the education process. [...] This process could follow the normal teaching of a child."

loss of generality, work with prefix-free machines. A prefix-free machine is a machine such that the domain is a prefix-free code on $\mathbb{B}^*$, or in other words, programs are self-delimited (no program can be a prefix of another program). A way to ensure that machines are prefix-free is by using *self-delimiting*[8] Turing machines, where inputs can only be read sequentially and outputs are also considered to be written sequentially. Input is hence not delimited and the Turing machine can stop reading eventually. The work tapes are bidirectional. We say that M *halts* on input $\sigma \in \mathbb{B}^*$ with output $\tau \in \mathbb{B}^*$, and we write $M(\sigma) = \tau$, if σ is on the left of the input head and τ is on the left of the output head after M halts. For the rest of the paper we will assume self-delimiting Turing machines. In fact, the notion of self-delimiting Turing machine is closer to the way we understand machines here as having a training input sequence (or 'life', or 'possible world'), where it is not possible to go back in time.

Any Turing machine M *becomes* another Turing machine (denoted by $M[\tau]$) after being fed with an input τ, i.e., for every string σ, $M(\tau\sigma) = M[\tau](\sigma)$. The set of all Turing machines is denoted by Ω, and known as the 'machine kingdom'. Two Turing machines M_1 and M_2 are *equivalent* iff for every $\sigma \in \mathbb{B}^*$, $M_1(\sigma) = M_2(\sigma)$. A machine M is said to be *null* iff for every $\sigma \in \mathbb{B}^*$, $M(\sigma) = \lambda$ or is not defined (M is partial). A halted machine is a null machine, but there may be null machines that may halt after reading non-empty strings σ.

From here, and following, e.g., [3], we can define universality:

Definition 2 A Turing machine[9] U is called universal (a Universal Turing Machine, UTM) if for every machine M there is a string τ such that for every string σ we have that $M(\sigma) = U(\tau\sigma)$ (i.e., we have that $M = U[\tau]$).

Following, e.g., [27], we can define a probability measure over *infinite sequences* as follows:

Definition 3 A probability measure w is defined over the sample space of infinite sequences $\mathbb{B}^\infty$ using cylinder sets $\sigma\circ$ (with σ being a finite string) and their countable union and complement as event space, as given by the values over the cylinder sets with the following properties:

$$w(\lambda\circ) = 1$$

$$\forall \sigma \in \mathbb{B}^* \quad w(\sigma\circ) = w(\sigma 0\circ) + w(\sigma 1\circ)$$

From here, and more intuitively, $w(\sigma\circ)$ is the probability that an infinite sequence starts with finite string σ. Because of this, and somewhat loosely, we will say that w is a probability measure (or distribution) on strings.

One special and important case of a probability measure is the *uniform* probability measure, denoted by v:

[8] We follow the definition of self-delimiting machines in [36, p.201] and the equivalent definition of prefix TM in [27, p.35].

[9] Technically, in the general case, both U and M must be prefix-free. However, since we are assuming self-delimiting Turing machines, all this is ensured.

Definition 4 The uniform probability (or Lebesgue) measure v is a probability measure for sets of infinite sequences defined as $v(\sigma\circ) = 2^{-|\sigma|}$ for any finite string σ.

This measure v represents the probability of sequences being constructed with 0s and 1s by tossing a fair coin. Other measures can of course behave differently, by giving more or less probability (or even zero) to some string prefixes. For instance, the universal semi-measure derived from UTM U (note that we would need monotone Turing machines here), as the probability of a sequence being output by U with fair coin tosses as inputs, could be normalised as a probability measure μ_U (see, e.g., [36] for details), and would be a very different way of assigning probabilities to sets of sequences starting with a given string.

As discussed in the introduction, we are interested in cognitive properties of machines, which are generally defined as follows:

Definition 5 A *property* is a real-valued function $\phi : \Omega \to [0, 1]$.

Higher values returned by the function ϕ imply a higher accomplishment of the property. We will now enumerate several kinds of properties:

- A *computably realisable* property ϕ is any property such that there is at least one Turing machine M for which $\phi(M) > 0$. Cognitive abilities are assumed to be computably realisable, especially because we expect that some machines may have them (while others not).
- A *decidable* property ϕ is any property such that there is an effective procedure to calculate $\phi(M)$ (to arbitrary precision) for every M. Precisely deciding $\phi(M)$ will be impossible for many properties. Consequently, empirical approximation through measurement, as discussed in section 5, will be necessary.
- A *Boolean* property is a property where the domain is restricted to $\{0, 1\}$, i.e., not having or having the property. For instance, the property of being universal, as per definition 2, denoted by ζ, is Boolean. Gradual properties (not restricted to $\{0, 1\}$) are said to be *non-Boolean*.
- A *non-vanishing* property is a property ϕ for which there is at least one Turing machine M and two constants $k, c \in \mathbb{R}$, with $1 \geq k > 0$ and $c > 0$ such that for every $n \in \mathbb{N}$ there are at least $\lceil k2^n \rceil$ strings $\sigma \in \mathbb{B}^n$ for which $\phi(M[\sigma]) > c$. This means that there is at least one machine with the property and that it can keep a non-zero value (bounded below) of the property indefinitely for a proportion of k of the inputs (the proportion is bounded below). Clearly, non-vanishing implies computably realisable.
- A *genuine* property is a non-vanishing property ϕ such that for every null machine M, $\phi(M) = 0$. In other words, the property does not hold for any null machine, but there is at least another non-null machine M' which makes ϕ non-vanishing as well.
- A property ϕ is *observable* iff for any two equivalent machines M_1 and M_2 we have that $\phi(M_1) = \phi(M_2)$.

In this paper, since we want to evaluate properties (and ultimately cognitive abilities) by the behaviour of an individual, we will be especially interested in observable genuine properties.

Now, let us analyse the *universality* property, denoted by ζ, and formally defined as $\zeta = 1$ if U is a UTM, and 0 otherwise. Universality is genuine[10]. The analysis of the universality property is of utmost importance for computer science and will be crucial for the proper understanding of the notion of potential ability, since UTMs are capable of becoming any other machine, and hence are eventually able to have any computably realisable property. Universality has almost always been studied as an actual property, i.e., a machine is either universal or not. This perspective has been challenged a few times in the past, and the interesting notion of probability makes the issue a matter of degree, rather than an absolute thing. As said in the introduction, the *probability* of a UTM halting for a random input was first investigated by Zvonkin and Levin [59] and Chaitin [5]. This is a first notion of potential, because two different machines may eventually halt but some machines may have a higher probability of doing so than others. No less interesting is the universality probability, the probability that a UTM preserves its universality (forever) after being fed with a random input —i.e., for a sequence for which each bit is i.i.d. with probability 0.5 of being 0 (and ditto of being 1). Formally, the notion of property preservation, as taken from [3], is:

Definition 6 An infinite sequence τ preserves a Boolean property ϕ with respect to machine M, denoted by $preserves(\phi, \tau, M)$, if all machines M_n^τ defined from M, τ and $n \in \mathbb{N}$ as $M_n^\tau \triangleq M[\tau_{1:n}]$ also have property ϕ ($\phi(M_n^\tau) = 1$).

From here we define the preserving probability of a property ϕ:

Definition 7 The ϕ-preserving probability for machine M, denoted by P_M^ϕ, is the measure of the set of all infinite sequences] which preserve property ϕ with respect to M.

If we take ζ (the property of universality), this probability was first suggested by Chris Wallace [6, footnote 70][7, sec. 2.5], and conjectured to be always 0. However, it has been shown in [3] that this probability is strictly between 0 and 1 for any UTM[11]. There is a special thing about ζ, then; once it is acquired it cannot be lost for *all* sequences (even though it must necessarily

[10] Since universality is 0 for any null machine, to show that it is genuine we need only to show that it is non-vanishing. This follows from the definition of universality probability as a limit (from [6, footnote 70]) which we know (from [3, Theorem 2.4] or alternatively from text in and surrounding our footnote 11) to have a lower bound greater than 0.

[11] A simpler proof of [3, Theorem 2.4 and Corollary 2.7] was given by Leonid Levin [3, p3499], but an even simpler proof is based on the fact that a 1-dimensional fair (50%:50%) random walk will pass any given point infinitely often from either direction. From there, we sketch this proof. Consider a recursive enumeration of UTMs $T_1, \ldots, T_i, \ldots$ (which might or might not be identical) and a monotonically increasing recursive function $g : \mathbb{N} \to \mathbb{N}$ such that $g(1) \geq 1$ and for all $i \geq 1$ we have $g(i{+}1) > g(i){+}1$. We define a UTM, U, as follows. For a given string, x, let $j_{x,1} < j_{x,2} < \ldots$ be the smallest values of j (in ascending order) such

be lost for *some* sequences). And if there is just a single string which makes a Turing machine M become a UTM then M is a UTM, and the probability of preserving it will always be greater than 0.

Conversely, if we consider certain other properties which are preserved indefinitely (become permanent), this means that the machine cannot be universal.

Proposition 1 *For any genuine property ϕ, if a machine M preserves ϕ indefinitely for any input, then M cannot be universal.*

Proof Assume M is universal. Then, it has a non-zero probability of halting. Since a halted machine is a null machine, and property ϕ is 0 for a null machine since ϕ is genuine, then M has a non-zero probability of not preserving property ϕ. But we have assumed that M preserves ϕ indefinitely for any input. So, by contradiction, M is not universal. □

Being universal implies that properties can be lost, since a machine can become a different machine. This proposition is closer to Wallace's intuition [6, footnote 70], and can be seen as a companion result to [3], especially if we consider that ϕ is intelligence, learning ability or some other significant cognitive ability, such as 'being educated'. If a system acquires an interesting (or genuine) ability and keeps it forever for any input, then it has to renounce its universality. In this way, an 'educated' machine must lose universality, as Wallace conjectured.

In any case, the concept of preserving (forever) a given property is much too specific. For many other properties, unlike universality, a machine M may not have the property ϕ, but may develop it after some inputs[12]. And for some other properties, we are not always interested in cases where the property is kept forever. In other words, we are interested in a notion of potential such that properties can be acquired and lost (or held to a higher or lower degree), and the probability of this happening (and when it happens) is what potential should really represent. This is what we address next.

3 Potential for properties of Turing machines

Let us start with sequential, deterministic machines, which is the most classical approach. Remember that $\phi(M)$ denotes the degree of M for the *actual*

that the first $2j$ bits of x contain j 0s and j 1s. If there is a k and an i such that $j_{x,k} = g(i)$, then choose the smallest such k and i, and the first $2j_{x,k}$ bits of x are used to get U to emulate/become T_i, and the subsequent bits of x are the input to T_i. (This defines UTM, U.) We can make the universality probability of U arbitrarily close to 1 by setting $g(1)$ large enough and having g grow sufficiently rapidly. Since the set $\{m/2^n : 1 \leq n, 1 < m < 2^n\}$ is dense in the open interval $(0, 1)$, it also follows that the set of universality probabilities of UTMs is dense in $[0, 1]$.

[12] The universality probability and the halting probability are very special cases. Once a machine has halted, it can never re-start. Once a machine has lost its universality, it can never again be universal.

property ϕ. As we have seen in the previous section we are interested in describing how this property changes after some inputs to M. But what inputs? If we just consider a single input sequence, we may draw an evolution of the property as shown in Figure 2 (left). However, this is not very informative because other input sequences are also possible (and even more likely).

An alternative is to consider all the sequences, which means that a distribution or measure over them must be determined. In fact, this is what we have done in definition 7 above. This definition takes one important thing implicitly, the measure of all sequences is the 'uniform' probability measure v (definition 4). It assumes that the probability of the ability is calculated with respect to input sequences such that 0 and 1 are equally likely, i.e., inputs are just 0s and 1s by tossing a fair coin. As already said, the 'uniform' probability measure assumes a uniform weight on all the input sequences of a given length. There are infinitely many other possibilities for this weight. For instance, we might assume that input sequences are generated by another (possibly universal) *monotone* Turing machine fed by the uniform probability measure v, which would lead to a different weight for each input sequence and, consequently, a different overall result in definition 7. In fact, this weighting given by each probability measure is at the core of some fundamental results in inductive inference, such as the expected value of squared prediction error, given in [47, Theorem 3 (17)].

3.1 Point potential and period potential

Continuing from above, the idea now is to parameterise the expected (potential) value (of a property) with different probability measures. Also, instead of considering how probable a Boolean property will be in the limit (i.e., permanently), we can just define the expected value of a non-Boolean property for a *point* t (i.e., temporally) as follows:

Definition 8 The *point* potential of property ϕ for machine M at point t, under a probability measure w, is given by:

$$\dot{\Pi}(M, \phi, t, w) \triangleq \sum_{\tau \in \mathbb{B}^t} \phi(M[\tau]) \cdot w(\tau\circ)$$

We can better understand the meaning of potential graphically. Figure 2 (right) shows a figurative evolution of point potential $\dot{\Pi}(M, \phi, t, w)$ for increasing values of t. Note that the curve looks smooth because it is an average of many (all) input sequences (using a probability measure), but it does not have to be so (or continuous) in general.

We may also be interested in the potential for a period. A period is just an input size range $[a, b]$ of positive integers, where $a \leq b$, by considering all the input strings σ of size $a \leq |\sigma| \leq b$. From here, we give our first definition of period potential:

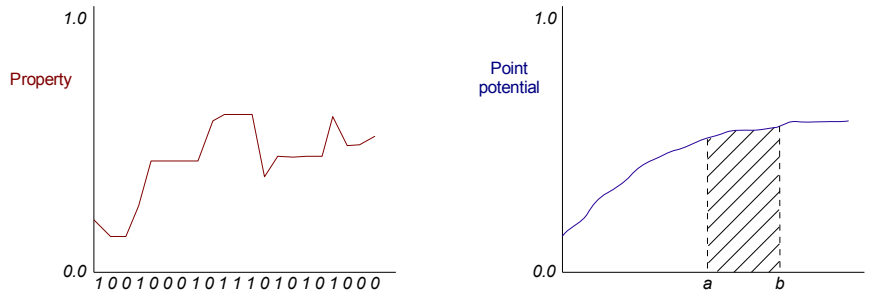

Fig. 2 Left: Figurative evolution of a property for a given sequence (on the x-axis). Right: Figurative evolution of point potential $\dot{\Pi}(M, \phi, t, w)$. The period potential $\Pi(M, \phi, a, b, w)$ is just the average of point potentials for the period $t \in [a, b]$.

Definition 9 The *period* potential of property ϕ for machine M for a period $[a, b]$ $(a \leq b)$, under a probability measure w, is given by:

$$\Pi(M, \phi, a, b, w) \triangleq \sum_{t=a}^{b} \frac{\dot{\Pi}(M, \phi, t, w)}{b - a + 1}$$

Clearly, $\Pi(M, \phi, t, t, w) = \dot{\Pi}(M, \phi, t, w)$. Potential is then the (suitably weighted) expected value of property ϕ after each and every input string σ of size $a \leq |\sigma| \leq b$ under the measure w. More precisely, it is the average value of the property for all the machines that originate from M after feeding all prefixes of sizes in $[a, b]$ of sequences according to measure w, where each *size* in $[a, b]$ (rather than each sequence) is given an equal weight in the average. Note that definition 9 also works for Boolean properties and the result is an estimated probability.

Figure 2 (right) shows the figurative evolution of point potential $\dot{\Pi}(M, \phi, t, w)$ for increasing values of t. A period potential is just the average of any portion in this curve (such as $[a, b]$ in the figure).

The information given by potential is an expected value. On some occasions, we may be interested in the whole distribution, i.e., how the property ϕ is distributed. While this is clearly much more informative, it also makes things much more complicated. Alternatively, we could calculate the expected value for a given distribution on the x-axis, as follows:

Definition 10 The *generalised* potential of property ϕ for machine M for a time distribution $p(t)$, under a probability measure w, is given by:

$$\tilde{\Pi}(M, \phi, w, p) \triangleq \sum_{t=1}^{\infty} \dot{\Pi}(M, \phi, t, w) p(t)$$

where $p(t)$ is a discrete probability distribution on positive integers.

This would make sense when we want to weight some periods more than others. Clearly a period potential between a and b is equal to the generalised

potential using a uniform distribution p between a and b (and 0 elsewhere). For the rest of the paper, we will just stick to definition 9 as an expected value on a period (weighted uniformly).

Clearly, the ϕ-preserving probability for machine M introduced in definition 7, denoted by P_M^ϕ, is just equal to $\lim_{t\to\infty} \dot{\Pi}(M, \phi, t, v)$ where v is the uniform probability measure (definition 4). The universality probability is just a special case: $\lim_{t\to\infty} \dot{\Pi}(M, \zeta, t, v)$, which we know is strictly between 0 and 1 for any UTM. In fact, with a similar rationale, we can show that this holds for any *genuine* property, which can be seen as an extension (or corollary) of Theorem 2.4 in [3].

Proposition 2 *For any genuine property ϕ, the uniform probability measure v and any UTM U, we have that* $\liminf_{t\to\infty} \dot{\Pi}(U, \phi, t, v) > 0$ *and* $\limsup_{t\to\infty} \dot{\Pi}(U, \phi, t, v) < 1$.

Proof The < 1 case is directly derived from the non-halting probability [5], since some programs make a UTM halt (and this set is known to have measure > 0) and a genuine property is 0 for these programs. The > 0 case is just a consequence of a genuine property being a non-vanishing property, which means that there is at least one Turing machine M and $k > 0$, $c > 0$ such that there are at least $\lceil k2^n \rceil$ strings $\tau \in \mathbb{B}^n$ for every $n > 0$ such that we have $\phi(M[\tau]) > c$. Since U is a UTM, it can become M for at least one input string σ, so $\liminf_{t\to\infty} \dot{\Pi}(U, \phi, t, v) \geq v(\sigma\circ) \liminf_{t\to\infty} \dot{\Pi}(M, \phi, t, v) = 2^{-|\sigma|} \liminf_{t\to\infty} \sum_{\tau\in\mathbb{B}^t} v(\tau\circ)\phi(M[\tau]) \geq 2^{-|\sigma|}kc > 0$.

In fact, by properly choosing the UTM, we can get values arbitrarily higher or lower (in the range of possible values for the property ϕ). Nonetheless, it is important to say that —for some properties ϕ— there can be some non-universal Turing machines for which the potential for property ϕ is much higher.

The previous argument brings out (again) that the mere possibility of a property being achieved is not really useful, especially if we are thinking about observable properties. The important thing about the notion of potential is how frequently the property can be achieved and when.

The use of a period in the notion of potential intelligence makes this explicit. It also makes the definition more precise. Actually, we can distinguish between permanent potentials (a property is preserved forever) and temporal potentials (a property is expected during a finite period). In practice we are interested in temporal potentials. For instance, when we say that a baby is potentially intelligent, we mean that it will probably have a certain degree of intelligence during a period of her life, say, between 20 years and 60 years, provided she has a somewhat typical education. We are not assuming that intelligence will be preserved eternally.

Some derived notions can be defined as well. For instance, we can define the speed that a machine takes to acquire[13] an ability as follows:

[13] We have used point potential here, but this could be extended to period potential.

Definition 11 The acquisition speed that a machine M takes to reach a threshold value c for an ability ϕ and distribution w is given by $min\{t : \dot{\Pi}(M, \phi, t, w) \geq c\}$. Note that all these are expected values for a given measure w.

We can also take the perspective of the distribution and calculate, e.g., $\{w : \dot{\Pi}(M, \phi, t, w) \geq c\}$ for some c, which means all the distributions of training sequences such that at least a degree of c in property ϕ is achieved after t input bits. Note that this may return some distributions which assign 0 to many sequences, or may even assign all the probability mass to sequences starting with one finite string. In the latter case, this could be understood as perfect training sequences. We will get back on this issue later on, when we define the notion of optimistic speed (definition 16).

3.2 Emulation

A Turing machine may be able to emulate another machine temporarily. This raises the issue that the difference between a system having a property and a system emulating another system having the property is indistinguishable from a behavioural point of view. In fact, as already said, we are interested in observable properties which are measurable by observing their behaviour, so the distinction may be important from a conceptual point of view but not really significant in many applications.

Our definition of potential above is parameterised for a given period, so if we only consider a finite period, it is irrelevant whether the machine stops emulating after the period and resumes some previous state, or keeps the property forever. It is interesting to analyse the notion of emulation related to the universality property ζ. For instance, from definition 2 we can just give the following refined definition:

Definition 12 A Turing machine U is n-universal if for every prefix-free machine M there is a string τ such that for every string σ whose length is less than or equal to n we have that $M(\sigma) = U(\tau\sigma)$.

Clearly, $(n + 1)$-universality implies n-universality, and ∞-universality is the original definition of universality (definition 2). Note also that the result of proposition 1 does not apply for n-universality[14].

It can be argued whether humans are potentially universal, in an informal sense. There are some training sequences which are able to persuade a human to do whatever the persuader wants (brain washing), and this is easier for n-universality, since humans can be instructed (i.e., programmed) to do a given cognitive task, as a job, especially under punishment or under threat. It is true that humans cannot do every task, because of space and time limitations, since human brains have finite memory (they are not ideal Turing machines)

[14] There are n-universal machines which can resume their original 'state' after n input bits.

and some tasks have time constraints for which some (or all) humans may be too slow. In any case, it is clear that humans can be instructed to do many cognitive tasks. In a voluntary way, (resource-bounded) n-universality is a common phenomenon for which we may have many examples, as actors imitating other behaviours or a person knowing computer programming and 'mentally executing' any program, even without realising what the program is actually calculating[15]. In fact, the first (algorithm for a) computer chess player was written by Turing and emulated by Turing himself (who had no machine to run it on) in order to play matches with some friends (and quite possibly also himself), although it has been estimated that the algorithm took him approximately 15 minutes to execute per move.

It is not coincidental that the father of the concept of Turing-completeness, i.e., UTMs, introduced the first intelligence test for machines as an imitation game. A computer being able to emulate (or just simulate) a human temporarily could pass the test. The relation between the ability of imitating, in general, as a UTM, and the Turing test has recently been explored in [22].

In fact, we can imagine what Turing could have been able to do had he designed a better computer chess player, memorised its code and emulated it himself. This is, actually, what any human assisted by (or emulating) Chinook (an optimal computer draughts player) can do, and become a perfect draughts player [43]. Also, consider now any other property, such as having an IQ of 200. If we were able to devise a program which can score 200 on regular IQ tests, or the program for a super-intelligent agent [32], then, by emulating it (and ignoring computational resources), any human could become super-intelligent. This imitation (from good and not-so-good sources) is, in fact, a great part of human learning, also present in other animals. Small children learn by imitation, so acquiring the abilities that other humans have.

We will go back to some of these issues later on, but the concept of emulation spurs two important issues: (1) resources (space and time) have to be taken into account, and (2) interaction is an important feature that we have been neglecting so far, since cognitive abilities are properties which are better represented by interactive information-processing skills.

4 Potential abilities of interactive agents

Turing machines are useful for understanding some basic relations between universality and other properties. Despite the original conception of Turing machines as systems which read and write on the same tape (which is slightly different to our use of self-delimiting Turing machines), the tape is not further altered by any external agent once the computation has started. Consequently, Turing machines are not interactive. However, cognitive abilities are usually associated with interactive systems, embedded in a world where actions have to be taken according to previous observations. Also, cognitive abilities have

[15] Searle's Chinese room elaborates on this for a different (and arguably misleading) purpose.

to be measured on resource-bounded interactive agents. Otherwise we may get counter-intuitive results, since any UTM would ultimately be able to emulate other agents if speed and space considerations were not taken into account. From here on (recalling footnote 4), we will re-interpret the machine kingdom Ω as the set of all resource-bounded interactive agents, as in [21]. We see this below.

4.1 Considering computational resources

Reinforcement learning [50,58] is an appropriate setting for considering agents interacting in an environment. Although the setting typically uses a discrete (alternating) interaction scheme (actions and observations alternate with no time delays), we can extend the notion of interactive system considering time. Over the next two pages or so, we outline a formalisation for deterministic asynchronous resource-bounded agents and environments, where they can issue no outputs for a while (or take some time to 'respond'), whereas the peer may issue several outputs during this time:

Definition 13 An interactive system is defined as a tuple $\langle \mathcal{T}, \mathcal{S}, \mathcal{O}, \mathcal{I}, \dot{s}, \dot{o} \rangle$, where $\mathcal{T}$ is the time space provided with a strict order relation $<$, $\mathcal{S}$ is the state space, $\mathcal{O}$ is the output space, $\mathcal{I}$ is the input space, $\dot{s}(s, i)$ is a transition rate function: $\mathcal{S} \times \mathcal{I} \rightarrow \Delta\mathcal{S}$, and $\dot{o}(s, i)$ is an output function: $\mathcal{S} \times \mathcal{I} \rightarrow \mathcal{O}$. There is an initial state s_0 and an initial start time t_0.

We will consider that the sets $\mathcal{T}$, $\mathcal{S}$, $\mathcal{O}$ and $\mathcal{I}$ are recursively enumerable and the transition rate and output functions are computable. In what follows, we will assume that $\mathcal{T}$ is the set of positive rational numbers (including zero, with $t_0 = 0$). By coupling this domain with actual (physical) or virtual time, we can turn the definition into a time-bounded one. We will assume that systems are deterministic[16]. *Agents* are interactive systems where outputs are called actions, and inputs are called perceptions or observations. Similarly, *environments* are also interactive systems, where outputs are called observations and inputs are called actions. The set of agents is a r.e. set. The set of environments is a r.e. set.

We now adapt the definitions in the previous section (sec. 3) to interactive systems. Instead of input sequences, we consider interaction histories between the environment and the agent. An interaction between an agent α and an environment μ is possible if $\mathcal{I}_\alpha = \mathcal{O}_\mu$ and $\mathcal{I}_\mu = \mathcal{O}_\alpha$ (directly, or through an appropriate interface). Given deterministic agent α and deterministic environment μ, the interaction history $H(\alpha, \mu)$ is the set of all triplets $\langle t, x, y \rangle$ where $t \in \mathcal{T}$, $x \in \mathcal{I}_\alpha = \mathcal{O}_\mu$ and $y \in \mathcal{I}_\mu = \mathcal{O}_\alpha$, such that the input and output of agent α at time t are x and y respectively (also, the input and output of environment μ at time t are y and x respectively). From here, since $\mathcal{T}$ is a strictly ordered

[16] For probabilistic environments and agents the notion of emulation (which we will see next) would be somewhat more challenging, understood as a probabilistic expectation rather than in terms of exact values.

set, we denote by $H_{a:b}(\alpha, \mu)$ the triplets $\langle t, x, y\rangle \in H(\alpha, \mu)$ where $a \leq t \leq b$. Environments can modify x and agents can modify y so generally only one of them will change for two consecutive triplets (which do not need to alternate since agents and environments are decoupled) except for the case where both agent and environment act at the very same time. We cannot have two triplets with the same value t. We assume that both the agent and the environment are time-bounded, so for every a and b, $H_{a:b}$ is a finite set, whose size is bounded by $(b-a)\beta + 1$, with β being a fixed (usually large) constant (typically β will be chosen as $1/q$ where q is a time quantum which sets the minimal time resolution for the interaction). Finally, given an agent α we denote the agent that results from α after interacting with μ during a time t as $\alpha[\mu, t]$ (starting from t_0). Actually, this is α with a different initial state.

Universality is easily understood in this setting as the property of an agent behaving like any other agent from a given time t, after an interaction history in the appropriate environment. Environments are then seen as programs (the notation $\alpha[\mu, t]$ makes this explicit), although this is not the usual way of programming an agent in the field of artificial intelligence (but it may become a more common option in the future, as we will discuss at the end of the paper).

We can give a more formal definition of interactive emulation as follows:

Definition 14 An agent α_1 after interacting (or being 'raised up') in an environment μ emulates α_2 during a period $[a, b]$, $a, b \in \mathcal{T}$, $(a \leq b)$, if $\forall \mu' : H_{0:(b-a)}(\alpha_1[\mu, a], \mu') = H_{0:(b-a)}(\alpha_2, \mu')$.

If $b = \infty$ we say that agent α_1 *becomes* α_2 after time a. The first difference with section 3 appears because we consider time (and possibly other resources), so there is no universal agent[17]. Secondly, it also becomes more visible that exact emulation is perhaps an idealistic view.

While the concept of universality is elusive when considering computational resources, the notion of potential can be easily adapted from the version (namely, definition 8) for non-interactive TMs.

Since the distribution of interaction histories depends on both the agent and the environment, it is easier to work with environment distributions. Let us consider $\mathbb{M}$ as the set all computable environments (defined as interactive systems), or any other subset (or class) of environments we would like to consider. Over this set, we can define a distribution, $\rho(\mu)$. This has been done, e.g., in [33,32], by using a universal distribution $\rho(\mu) = 2^{-K(\mu)}$ (with probabilistic environments and a more classical alternating interaction scheme), but many other possibilities exist. From here:

Definition 15 The point potential of property ϕ for agent α for time t, $t \in \mathcal{T}$ under an environment distribution ρ over a class $\mathbb{M}$, denoted by $\dot{\Pi}(\alpha, \phi, t, \rho)$,

[17] The same seems to apply to environments. While the notion of universal environment is appealing, the inclusion of time makes this notion more general (but also infeasible).

is the expected value of ϕ for α at time t for all the environments in class $\mathbb{M}$ weighted by ρ. Formally:

$$\dot{\Pi}(\alpha, \phi, t, \rho) \triangleq \sum_{\mu \in \mathbb{M}} \phi(\alpha[\mu, t]) \cdot \rho(\mu)$$

The period potential $\Pi(\alpha, \phi, a, b, \rho)$ can be defined from point potential as we did in definition 9.

4.2 Environments and kinds of potential

The use of a class or distribution of environments emphasises that potential abilities represent expected values over an astronomical range of possibilities. In the case of interactive environments considering that interaction is asynchronous and may take time, we may have a huge amount of environments which are either too slow or too fast. For the classical, alternating discrete time, they may still be repetitive or apparently random, so they will have almost no effect on a potential ability. Actually, in many cases, the ability will develop (or not) if the agent's program tells it to do (or not do) so within (or at the end of) a given period.

The notion of *speed*, seen in the previous section (definition 11) as the time that an agent takes to acquire a minimum value c for an ability ϕ and distribution w, suffers the same concerns. Any definition of potential which considers a broad sample of environments might be unrealistic. This would be like considering that a baby is not potentially intelligent, because we calculate the expectation of letting her grow in a random place in the universe, where, if she survives [57, p335, sec.3], would have no interesting stimuli. As a result, the expectation should be calculated with an appropriate (presumably much concentrated) distribution or class, which accounts for those 'lives' we are interested in or we expect the agent to face. One solution for this is the so-called Darwin-Wallace distribution for environments, as introduced in [20]. This distribution is conceived for measuring (social) intelligence, but other distributions could be used for other abilities. These distributions could then be properly adapted for the calculation of potential or acquisition speed, as we did for definition 11.

Alternatively, we can define the notion of *optimistic speed* as follows:

Definition 16 The optimistic speed of an agent α for showing a threshold level c for property ϕ during some time span s is given by: $min\{t : \exists\rho \ \Pi(\alpha, \phi, t, t+s, \rho) \geq c\}$. If this minimum value does not exist, the optimistic speed is infinite.

The idea behind the above definition has many possible alternative formulations, meaning different things, such as $argmax_\rho \dot{\Pi}(\alpha, \phi, t, \rho)$ which is the distribution of environments which gets the highest value for the property in a given time t.

All this is again related to the perfect training sequence problem originally introduced by Solomonoff [48] (and touched upon by Wallace [6, sec. 0.2.5, p542, col.1]), but in the more realistic setting of interactive agents considering time. In the end, definition 16 can be seen as an optimisation problem of what environment (i.e., education) should be given to α to get a degree c in ability ϕ as fast as possible. Interestingly, those agents which are easily 'programmed' by the environment would be able to acquire the property faster than other agents which are less malleable. However, and much more interestingly, for some abilities, an agent which is able to learn would possibly require fewer bits of information to construct the program it needs to run to get the ability than if given the program itself. In other words, some agents could learn (be programmed) by example rather than by direct programming, and this may be more efficient in many cases. This is in fact the approach taken by machine learning and, most especially, by general reinforcement learning settings such as AIXI [27,53]. This supports the study of perfect training sequences for machines using Levin's optimal search [35] (or any other learning machine, e.g., AIXI), as Solomonoff did [48], rather than for UTMs.

It is also enlightening (and perhaps necessary) to consider that environments can also include some other agents, and these agents may have some properties. The use of environments which are able to host some other agents is seen as a requirement for many cognitive abilities which are characterised by interacting with other agents. Recently, it has been argued that in order to measure intelligence we require environments full of other agents of similar degrees of intelligence, and that only under a distribution of environments that takes this into account, such as the already mentioned Darwin-Wallace distribution [20], does it make sense to measure intelligence as an average performance over a distribution of environments.

The existence of other agents in an environment which may have many different kinds of abilities opens up many possibilities for the notion of potential. For instance, the potential of an agent can increase if we just consider environments where other agents having the property abound, since the ability can be acquired, i.e., learnt, from other agents, by *imitation*. Other levels of interaction can boost potential, such as having some agents transmitting knowledge or acting as teachers. Also, some agents can acquire an ability by controlling (or taking advantage of) other agents, such as a person with a calculator or with an advisor.

Finally, as we will further mention in section 7, we may consider several agents as a group and think about some members of the group having a property or the whole group having a property. In fact, we could think of properties of environments (instead of agents), and ask questions such as, "would this environment develop life?", "would this environment develop intelligence?"[6, sec. 0.2.7, p545, col. 1]. A proper formalisation of these questions is much more difficult than the notion of individual potential seen in this paper.

5 Measuring potential abilities

A definition of potential ability and its adaptation to a classical and an interactive setting are useful to have a precise account of the concept, and to discuss the relations between a property and its potential. It is also useful to better analyse conceptually some properties such as universality or intelligence, which are usually associated with their potential counterpart. By making explicit that some issues have to be considered for an appropriate notion of potential, such as the distribution of environments and the (future) period that the potential refers to, we have also understood that some other simplistic views have flaws or are simply counter-intuitive.

In section 2.3, we introduced the term 'decidable' for those properties for which we can find a procedure (by introspection or observation) to calculate the exact value of that property for any possible machine. Clearly, most properties will be undecidable in general. Nonetheless, even for undecidable properties, it may be possible to approximate the value for all (or a great proportion of) machines. In fact, decidability does not even suggest the difficulty of measurement, since some decidable properties (e.g., machines always outputting a 0 after 2^{1000} *years*) might be difficult to measure.

In the end, the usefulness of the very concept of potential abilities makes full sense if we are able to *measure* them. And, by measuring them, we mean the observation of their behaviour, using tests or related mechanisms, and not by analysing the code (or the DNA) of the system.

Apparently, the evaluation of potential abilities will be generally more difficult than the evaluation of actual abilities. The evaluation of actual abilities is already a difficult issue, as shown by disciplines like psychometrics and comparative psychology, and the efforts already made to develop tests for machines.

The specific problems of measuring potential abilities are not (but certainly add up to) the problems of cognitive tests, where the result is usually a lower bound of the actual ability of the subject, because of inappropriate rewards, insubordination [46, sec. 6], bad interfaces, etc. It is not then the problem referred to by (other uses of) the term potential in psychology [39, 51, 37], as discussed in section 2. Instead, there are two specific problems for measuring potential cognitive abilities. First, we are trying to measure something that has not happened yet. So we need to infer future results. This means that we need an accurate model (or a very good estimator) of the individual and also of the environments which are considered for the expected value. Second, we do not have repeatability for the same individual. Inferring the potential at a given development stage of an individual can only be done once, so we cannot have enough evidence to properly extrapolate. In fact, this second problem suggests that when we talk about potential of an individual (e.g., a 3-month old baby), we use that individual as a prototype of a bigger population (e.g., all 3-month old babies). This is an interesting concept, because we can define the potential of a population of individuals. For instance, if we consider a set of possible

agents[18] Ω, we can define a distribution over them, κ (so $\sum_{\alpha\in\Omega}\kappa(\alpha) = 1$), and extend the notion of potential as follows:

Definition 17 The period potential of a distribution of agents κ is:

$$\Pi^{\diamond}(\phi, a, b, \rho, \kappa) \triangleq \sum_{\alpha\in\Omega} \Pi(\alpha, \phi, a, b, \rho) \cdot \kappa(\alpha)$$

This means that we can infer potential abilities by considering populations of agents which are similar. This is what psychometrics does. We can infer, for instance, how intelligent a particular 6-year old child will be by the age of 20, by taking some variables and comparing them with the evolution of other humans for similar periods and conditions. For machines this seems to be easier, because we can replicate agents and environments. All this suggests three general approaches for measuring potential intelligence:

1. An analysis of the *evolution* of an ability, i.e., its curve, can give information about saturation points and when they will be reached (and how long they can be expected to be sustained for). On some occasions, this is possible by looking at a single individual and a single environment, especially when the curve is expected to have a particular shape, and we can infer a plateau. For instance, negatively-accelerated curves or S-shaped learning curves are frequently observed for many different tasks and abilities for animals and AI algorithms [12,50,58,1], and can be approximated by logistic or (cumulative) Weibull functions. This is more powerful if we can extrapolate the potential for a similar individual for a similar class or distribution of environments. This is exactly what the growth chart in Figure 1 does for stature. Also, from plots such as the one on the right of Figure 2 (or even from a partial view of the plot) we can infer the potential ability for *similar* agents (or the same agent). Of course, this estimation is easier if we only want to calculate the potential ability for a few environments. In the general case of an infinite class of environments, sampling is necessary.
2. Another way of estimating potential relies on the *actual* ability of a similar subject. This does not rely on the curves of one or the other but on point measurements. For animals, including humans, this can be done with 'relatives', i.e., other subjects which share part of the genotype. For instance, we can give a rough estimate of the intelligence that a child might have as an adult by evaluating the actual intelligence of her parents. Obviously, this use of the genotype as a predictor will only be eventually used for machines (algorithm instead of genotype) when we have that the subject to be evaluated is a small variation (or evolution) of another machine whose properties we know well. The knowledge about one machine can be used to infer estimations for the related machine.

[18] Extending this for *all* possible computable agents would depend on whether they are probabilistic or deterministic, and resource-bounded or not. We can just consider a distribution over all computable agents as defined in section 4.

3. A third approach is to determine whether a specific actual property correlates with the potential of *another* property, i.e. $\dot{\Pi}(\alpha, \phi, t, w) \approx \phi'(\alpha)$. If we have effective tests for ϕ', then the problem is solved. Obviously, in order to establish this correlation we need a thorough experimental analysis of the evolution of ϕ for similar agents (or replications of the agent) and the class (or distribution) of environments of interest. For some abilities, we could even establish some relations theoretically, such as some actual learning abilities being related to the potential of acquiring some other cognitive abilities. In fact, for some abilities, we may even develop theoretical bounds, very much in the same way that some error bounds are given for some learning algorithms or paradigms (see, e.g., [47,30]).

In the case of machines, it will be more common (at least in the following years) to be interested in the potential abilities of *algorithms* rather than actual agents with a given state. The goal will be to analyse whether a given algorithm will develop a property. The previous discussion (and the bulk of this paper) has excluded the analysis of the code (for observable properties), but it is obviously a possibility to combine some experimental measurement with some estimations given by theoretical results about the algorithm itself (when possible, since many properties cannot be determined theoretically even for simple algorithms). Although the analysis of algorithms is independent of the underlying physical machine, there are of course algorithms for which the notion of speed (as per definition 16) and the computational cost of each step are issues.

Finally, we have to pay attention to the notion of reward, since many cognitive abilities, e.g., intelligence, require a way to persuade an individual to do a task. For human adults, this is usually taken for granted, since we can give orders and make the subject complete the test (although this does not mean that the subject is always motivated and does her best). For children and animals in general, the choice of appropriate rewards and interfaces is crucial. The notion of potential and its estimation must be linked to environments which include rewards. Also, the interface (or the range of interfaces) should be the same for all, because otherwise the results would not be extrapolable (for a related discussion see, e.g., [46, sec. 6][21]). The notion of universality is also partially at odds with the idea of interface. For instance, redundant Turing machines are machines that work with a special coding of the input (such as 00 for 0 and 11 for 1, and are null for inputs not following the code) [6, sec. 0.2.7][18, p1514, footnote 6]. Some non-universal redundant machines could essentially become universal (redundant) machines with a proper interface — with the redundant machine carrying out the redundantly coded version of the original (non-redundant) calculation[19].

[19] For example, if we redundantly code 0 and 1 as 01 and 10 respectively, if M is a machine and M_R is its redundant counter-part, and (say) $M(100) = 0110$, then $M_R(100101) = 01101001$.

6 Related notions

The concept of potential suggests possible relations with other notions, especially in the context of cognitive abilities that can be learned. Actually, since potential deals with the development of an ability, there have been many studies about how cognitive abilities develop in humans, with age. For instance, some verbal abilities grow quickly during infanthood and then more slowly during adulthood, while other cognitive abilities or traits that cannot be learned (e.g., reaction time) start a small decrease at the age of 20. Some more specific abilities, e.g., playing chess, usually reach their peak in their thirties (or thereabouts) for professional players, although the exact point and the development curve depends on many factors. Some abilities increase with time because they depend on knowledge and skills that are learned over the years.

This is not to be confused with the distinction between fluid and crystallised intelligence [26], where fluid intelligence can be said to be the ability of creating knowledge by identifying patterns, while crystallised intelligence is defined as the ability of using previously-acquired skills or knowledge. Fluid intelligence is usually constant from the age of 20 (or with a small decline), while crystallised intelligence still increases slightly in adulthood. Since both are components of usual IQ tests, general intelligence is said to be relatively constant for most adulthood. Note that, despite their names, crystallised intelligence is not the "crystallised" form of fluid intelligence, so crystallised intelligence does not represent the knowledge a subject has of a specific domain. Also, even though fluid and crystallised intelligence are correlated, it is debatable whether fluid intelligence can be seen as a predictor for *future* crystallised intelligence or, more precisely, for the future increase in crystallised intelligence. So, it is not clear whether one can be seen, even roughly, as the potential of the other. In fact, for example, both the fluid and crystallised intelligence of a small baby are 0. Nevertheless, fluid intelligence can be seen as a measure of the flexibility of a mind in terms of the use of basic deductive and, especially, inductive abilities. As a result, high degrees of fluid intelligence correlate with the potential of many other specific cognitive abilities. However, this is also true, perhaps to a lesser extent, for crystallised intelligence. For instance, we can estimate the potential ability of playing chess well for an adult person who has never played chess by performing a test of fluid and crystallised intelligence, since both fluid and crystallised intelligence are necessary for learning and playing a game such as chess well.

Related to the above distinction between fluid and crystallised intelligence we find the *exploration vs. exploitation dilemma*. Many systems behave more exploratively in unknown environments where risks and needs are low, while they exploit what they have learned during previous explorative stages whenever they need to achieve a goal. During exploration, systems typically acquire knowledge while, during exploitation, systems apply this knowledge. This seems to relate fluid intelligence with exploration, and crystallised intelligence with exploitation. This relation means that fluid intelligence would be

more beneficial during exploration stages and crystallised intelligence should be more beneficial during exploitation stages. For instance, a system with high fluid intelligence but low crystallised intelligence may be able to find complex patterns and construct elaborate concepts (good at exploration) but it then may fail to apply them on practical problems (bad at exploitation). This does not mean, however, that a system that is very explorative must have high fluid intelligence or vice versa. Similarly, a system which tries to exploit the environment from the very beginning is not necessarily a system with high crystallised intelligence, or vice versa. Actually, the relation becomes more subtle when we think about measuring an ability. Typically, if a subject is more explorative, i.e., more playful (such as children or some animals), it is very difficult to measure some abilities (including fluid intelligence), and the scores would be an under-estimation, because the subject does not focus on the task. This is related to the notion of potential of a test in psychometrics, as we described in section 2.2 but not directly to the notion of potential in this paper. In the case of machines, we can, for some abilities, hard-wire or condition the system to be in 'exploitation mode', whenever an evaluation is to be made.

The connection of our notion of potential with the exploration vs. exploitation dilemma appears when the abilities are related to learning. In reinforcement learning, e.g., we usually employ a discount rate parameter. In order to solve a task, we can distinguish between learning from the environment (possibly moving randomly or using the actions that get more information) and exploiting the knowledge (using the actions which get higher expected reward). Many experimental settings in reinforcement learning distinguish these stages. However, when evaluating a learning ability, we cannot split these two stages, since we want to evaluate how quickly the system is able to learn and exploit a given environment. Systems that take too much time exploring would score poorly but systems that try to exploit from the very beginning can be stuck in local optima or never reach some important information about the environment. Loosely, we could say that if a system is too biased towards exploitation then it will never reach its full potential. This interpretation is misleading, because it is the *technique* (e.g., Q-learning) which is below its full potential for that problem, but not the system. The system has a fixed, evolving or adaptable discount rate and, according to that configuration, the system must be evaluated. The potential of a subject is actually related to considering a future time after a variety of environments, but not considering variations of the subject. Finally, and quite differently, we could define the cognitive ability of 'being explorative' or abilities for which the result in the limit (or in the long term) is more important than having a good result quickly. In that case, the testing device could be tuned to give more value to late rewards than to initial rewards and to be mild with mistakes. In tests using RL-like environments and rewards (such as [33,18]), this could be done by modifying the discount rate of the test or the way rewards are averaged [17,28,29,34].

In general, it is important to consider the notion of potential as an expected value. The *definition* depends on the individual, the ability, the environment

and the period, but the *measurement* may introduce some new issues, since potential abilities cannot be measured directly and must be indirectly estimated from other (similar) individuals, other (related) abilities, other (distributions of) environments and other (equivalent) time periods, as has been examined in section 5.

7 Discussion

In this paper we have focussed on the potential of individuals that acquire or lose a given ability (or increase or decrease its score). Along the way, we have also seen that the notion of potential could also be applied at many levels. For instance, we discussed that an environment could contain agents which may have some abilities. While it is clear that, in this case, the environment (as a closed system) does not have any of these abilities, it *hosts* agents having them. In the literature, this view of potential (usually referred to as 'emergence') has been used for three particular properties (arguably the three most important properties of all): universality, life and intelligence, which are also deeply related.

For instance, given a real or artificial world, we may be interested in determining whether the world can contain universal computers (possibly with resource limitations). It is clear that the universe, as we know it, holds resource-bounded universal computers. So, the universe, at the Big Bang (or however it might have begun), was a potential universality-holder[20]. The universe has developed life (at least on Earth), and DNA is another example of a Turing-complete machine. Some programs 'written' with this DNA (i.e., some genome) 'generate' humans, who are able to reason and think. Humans are able to emulate and hold models of the world. Natural language is a powerful Turing-complete language [54, sec. 9.3]. So universality emerges once again. Finally, humans have created computers, yet again sources of local universality. This trip across levels also occurs for life and intelligence, and will be more and more frequent in the future, with the growth and development of artificial life, the technological singularity and other self-replicating and self-improving systems.

All this can be studied in terms of being possible, being probable or just estimating when it will take place. For instance, Solomonoff [49] studied several stages in the process towards the technological singularity, and gave predictions of when these stages could come, and not their probability. It is clearly more difficult to estimate when a property will appear than just estimating its probability. Also, it is important to estimate whether it will endure. For instance, current DNA ensures universality and self-replication, but it seems that self-replication came much before.

[20] In fact, Conway's game of life [13] is a very simple 'universe' and can contain universal computers. Given an appropriate 'big bang' (i.e., a start-up configuration of the cells), it has been shown that Conway's game of life can contain a universal Turing machine.

There is also a strong parallelism between an artificial life system running on a computer and a fantastic world running on a human's mind, such as figuring out a different fictional world by imagination or after reading a novel [7, sec. 7.7, p968]. Intelligence, especially with natural language, is able to host other worlds (thoughts, situations, memories, etc.) which can have or host some other properties. In fact, a natural language and many Turing-complete computer languages share the property of being able to describe any effectively computable function. Minds and machines are just the substrate to execute them. Since natural language can be considered a universal programming language, the ability of learning a language may be seen as the ability of becoming universal (Wallace makes this point as well in [54, sec. 9.3]), in the restricted sense above.

While all of this has been left out from the analysis in this paper, it is relevant for the concept of potential, and most especially for the properties of universality and intelligence. Some recent works have also explored related ideas in the context of intelligence, such as 'self-modification', 'mortality', 'delusion' and 'survival' [41, 42].

Back to the mainstream view of potential in this paper, i.e., the capacity of individuals, we have seen that two important properties, universality and intelligence, are intertwined. Absolute universality (i.e., becoming universal) is incompatible with preserving intelligence for all inputs. Also, it implies that the machine can halt (i.e., die). However, other more restrictive views of the universality property are more compatible with (or even intrinsic to) intelligence, such as n-universality (temporal emulation), resource-bounded universality, etc.

The notion of potential ability is not only useful to clarify the relation between some properties. Having a clear definition of the concept is crucial for the characterisation of individuals, since humans, non-human animals and other machines are usually classified by their potential abilities rather than by their abilities. For instance, a baby is classified as a human being, even though it has none of the cognitive abilities an adult human being has. In fact, the very concept of 'person' is, in part, potential, and this and other concepts should be very clear before we want to extend them to AI artefacts. Actually, these notions lead to highly controversial ethical issues, such as the set and degree of potential properties required to make the processes of replication, abortion and/or euthanasia ethically unacceptable for machines.

The notion of potential intelligence, in particular, and any procedure that could be used to estimate it can be crucial for the field of artificial intelligence. It is quite unlikely that we can construct an algorithm such that it makes a machine intelligent the first day. Surely, the machine will require an appropriate environment (such as a playschool [14]) and some training (optimal training sequences [44])[21]. How long the training is (and how difficult finding a good training sequence is) will of course be related to potential abilities. In fact, in the worst case, training a machine to be intelligent would be like taking a

[21] Recall the related quotations from Turing (1950) [52, sec. 7] in our footnote 7.

UTM and finding the program that makes it intelligent. It looks like a trade-off between these two extremes (a very intelligent machine from scratch or a UTM to be given a good training sequence) needs to be found. This is the direction of the field of artificial general intelligence, which tries to split from the task-specific view of mainstream artificial intelligence. More orientated, the roadmap for machine intelligence may lie on the gestation of *potentially* intelligent systems and the construction of optimal training environments for intelligence.

Acknowledgments

We thank the anonymous reviewers for their comments, which have helped to significantly improve this paper. This work was supported by the MEC-MINECO projects CONSOLIDER-INGENIO CSD2007-00022 and TIN 2010-21062-C02-02, GVA project PROMETEO/2008/051, the COST - European Cooperation in the field of Scientific and Technical Research IC0801 AT. Finally, we thank three pioneers ahead of their time(s). We thank Ray Solomonoff (1926-2009) and Chris Wallace (1933-2004) [54,6] for all that they taught us, directly and indirectly. And, in his centenary year, we thank Alan Turing (1912-1954), with whom it perhaps all began.

References

1. S. Amari, N. Fujita, and S. Shinomoto. Four types of learning curves. *Neural Computation*, 4(4):605–618, 1992.
2. Aristotle (Translation, Introduction, and Commentary by Ross, W.D.). *Aristotle's Metaphysics*. Oxford, Clarendon Press, 1924.
3. G. Barmpalias and D. L. Dowe. Universality probability of a prefix-free machine. *Philosophical Transactions of the Royal Society A [Mathematical, Physical and Engineering Sciences] (Phil Trans A), Theme Issue The foundations of computation, physics and mentality: the Turing legacy' compiled and edited by Barry Cooper and Samson Abramsky*, 370:3488–3511, 2012.
4. G. J. Chaitin. On the length of programs for computing finite sequences. *Journal of the Association for Computing Machinery*, 13:547–569, 1966.
5. G. J. Chaitin. A theory of program size formally identical to information theory. *Journal of the ACM (JACM)*, 22(3):329–340, 1975.
6. D. L. Dowe. Foreword re C. S. Wallace. *Computer Journal*, 51(5):523 – 560, Sept 2008. Christopher Stewart WALLACE (1933-2004) memorial special issue.
7. D. L. Dowe. MML, hybrid Bayesian network graphical models, statistical consistency, invariance and uniqueness. In P. S. Bandyopadhyay and M. R. Forster, editor, *Handbook of the Philosophy of Science - Volume 7: Philosophy of Statistics*, pages 901 – 982. Elsevier, 2011.
8. D. L. Dowe and A. R. Hajek. A computational extension to the Turing Test. *Technical Report #97/322, Dept Computer Science, Monash University, Melbourne, Australia, 9pp, http://www.csse.monash.edu.au/publications/1997/tr-cs97-322-abs.html*, 1997.
9. D. L. Dowe and A. R. Hajek. A non-behavioural, computational extension to the Turing Test. In *Intl. Conf. on Computational Intelligence & multimedia applications (ICCIMA'98), Gippsland, Australia*, pages 101–106, February 1998.
10. D. L. Dowe and A. R. Hajek. A computational extension to the Turing Test. *in Proceedings of the 4th Conference of the Australasian Cognitive Science Society, University of Newcastle, NSW, Australia*, page 9pp, September 1997.

11. D. L. Dowe and J. Hernández-Orallo. IQ tests are not for machines, yet. *Intelligence*, 40(2):77 – 81, 2012.
12. C. R. Gallistel, S. Fairhurst, and P. Balsam. The learning curve: Implications of a quantitative analysis. *Proceedings of the national academy of Sciences of the united States of america*, 101(36):13124–13131, 2004.
13. M. Gardner. Mathematical games: The fantastic combinations of John Conway's new solitaire game "life". *Scientific American*, 223(4):120–123, 1970.
14. B. Goertzel and S. V. Bugaj. AGI preschool: a framework for evaluating early-stage human-like AGIs. In *Proceedings of the Second International Conference on Artificial General Intelligence (AGI-09)*, pages 31–36, 2009.
15. J. Hernández-Orallo. Beyond the Turing Test. *J. Logic, Language & Information*, 9(4):447–466, 2000.
16. J. Hernández-Orallo. On the computational measurement of intelligence factors. In A. Meystel, editor, *Performance metrics for intelligent systems workshop*, pages 1–8. National Institute of Standards and Technology, Gaithersburg, MD, U.S.A., 2000.
17. J. Hernández-Orallo. On evaluating agent performance in a fixed period of time. In M. Hutter et al., editor, *Artificial General Intelligence, 3rd Intl Conf*, pages 25–30. Atlantis Press, 2010.
18. J. Hernández-Orallo and D. L. Dowe. Measuring universal intelligence: Towards an anytime intelligence test. *Artificial Intelligence*, 174(18):1508 – 1539, 2010.
19. J. Hernández-Orallo and D. L. Dowe. Mammals, machines and mind games. Who's the smartest?. *The Conversation, http://theconversation.edu.au/*, April 2011.
20. J. Hernández-Orallo, D. L. Dowe, S. España-Cubillo, M. V. Hernández-Lloreda, and J. Insa-Cabrera. On more realistic environment distributions for defining, evaluating and developing intelligence. In J. Schmidhuber, K.R. Thórisson, and M. Looks (eds), editors, *Artificial General Intelligence 2011*, volume 6830, pages 82–91. LNAI series, Springer, 2011.
21. J. Hernández-Orallo, D. L. Dowe, and M. V. Hernández-Lloreda. Measuring cognitive abilities of machines, humans and non-human animals in a unified way: towards universal psychometrics. *Technical Report 2012/267, Faculty of Information Technology, Clayton School of I.T., Monash University, Australia*, March 2012.
22. J. Hernández-Orallo, J. Insa, D. L. Dowe, and B. Hibbard. Turing tests with Turing machines. In Andrei Voronkov, editor, *The Alan Turing Centenary Conference, Turing-100, Manchester*, volume 10 of *EPiC Series*, pages 140–156, 2012.
23. J. Hernández-Orallo and N. Minaya-Collado. A formal definition of intelligence based on an intensional variant of Kolmogorov complexity. In *Proc. Intl Symposium of Engineering of Intelligent Systems (EIS'98)*, pages 146–163. ICSC Press, 1998.
24. E. Herrmann, J. Call, M. V. Hernández-Lloreda, B. Hare, and M. Tomasello. Humans have evolved specialized skills of social cognition: The cultural intelligence hypothesis. *Science*, Vol 317(5843):1360–1366, 2007.
25. E. Herrmann, M. V. Hernández-Lloreda, J. Call, B. Hare, and M. Tomasello. The structure of individual differences in the cognitive abilities of children and chimpanzees. *Psychological Science*, 21(1):102–110, 2010.
26. J.L. Horn and R.B. Cattell. Refinement and test of the theory of fluid and crystallized general intelligences. *Journal of educational psychology*, 57(5):253, 1966.
27. M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, 2005.
28. J. Insa-Cabrera, D. L. Dowe, S. España, M. V. Hernández-Lloreda, and J. Hernández-Orallo. Comparing humans and AI agents. In *AGI: 4th Conference on Artificial General Intelligence - Lecture Notes in Artificial Intelligence (LNAI)*, volume 6830, pages 122–132. Springer, 2011.
29. J. Insa-Cabrera, D. L. Dowe, and J. Hernández-Orallo. Evaluating a reinforcement learning algorithm with a general intelligence test. In *CAEPIA - Lecture Notes in Artificial Intelligence (LNAI)*, volume 7023, pages 1–11. Springer, 2011.
30. M. Kearns and S. Singh. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 49(2):209–232, 2002.
31. A. N. Kolmogorov. Three approaches to the quantitative definition of information. *Problems of Information Transmission*, 1:4–7, 1965.

32. S. Legg. *Machine Super Intelligence*. Department of Informatics, University of Lugano, June 2008.
33. S. Legg and M. Hutter. Universal intelligence: A definition of machine intelligence. *Minds and Machines*, 17(4):391–444, 2007.
34. S. Legg and J. Veness. An Approximation of the Universal Intelligence Measure. In *Proceedings of Solomonoff 85th memorial conference*. Springer, 2012.
35. L. A. Levin. Universal sequential search problems. *Problems of Information Transmission*, 9(3):265–266, 1973.
36. M. Li and P. Vitányi. *An introduction to Kolmogorov complexity and its applications (3rd ed.)*. Springer-Verlag, 2008.
37. V. L. Little and K. G. Bailey. Potential intelligence or intelligence test potential? a question of empirical validity. *Journal of Consulting and Clinical Psychology*, 39(1):168, 1972.
38. M. V. Mahoney. Text compression as a test for artificial intelligence. In *Proceedings of the National Conference on Artificial Intelligence, AAAI*, pages 486–502. John Wiley & Sons Ltd, 1999.
39. A. R. Mahrer. Potential intelligence: a learning theory approach to description and clinical implication. *The Journal of General Psychology*, 59(1):59–71, 1958.
40. G. Oppy and D. L. Dowe. The Turing Test. In Edward N. Zalta, editor, *Stanford Encyclopedia of Philosophy*. Stanford University, 2011. http://plato.stanford.edu/entries/turing-test/.
41. L. Orseau and M. Ring. Self-modification and mortality in artificial agents. In *AGI: 4th Conference on Artificial General Intelligence - Lecture Notes in Artificial Intelligence (LNAI)*, volume 6830, pages 1–10. Springer, 2011.
42. M. Ring and L. Orseau. Delusion, survival, and intelligent agents. In *AGI: 4th Conference on Artificial General Intelligence - Lecture Notes in Artificial Intelligence (LNAI)*, volume 6830, pages 11–20. Springer, 2011.
43. J. Schaeffer, N. Burch, Y. Bjornsson, A. Kishimoto, M. Muller, R. Lake, P. Lu, and S. Sutphen. Checkers is solved. *Science*, 317(5844):1518, 2007.
44. R. J. Solomonoff. Training sequences for mechanized induction. *Self-organizing systems, eds., M. Yovits, G. Jacobi, and G. Goldsteins*, 7:425–434, 1962.
45. R. J. Solomonoff. A formal theory of inductive inference. *Information and Control*, 7:1–22, 224–254, 1964.
46. R. J. Solomonoff. *Inductive Inference Research: Status, Spring 1967*. RTB 154, Rockford Research, Inc., 140 1/2 Mt. Auburn St., Cambridge, Mass. 02138, July 1967, 1967.
47. R. J. Solomonoff. Complexity-based induction systems: comparisons and convergence theorems. *Information Theory, IEEE Transactions on*, 24(4):422–432, 1978.
48. R. J. Solomonoff. Perfect training sequences and the costs of corruption - a progress report on induction inference research. *Oxbridge Research*, 1984.
49. R. J. Solomonoff. The Time Scale of Artificial Intelligence: Reflections on Social Effects. *Human Systems Management*, 5:149–153, 1985.
50. R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. The MIT press, 1998.
51. T. R. Thorp and A. R. Mahrer. Predicting potential intelligence. *Journal of Clinical Psychology*, 15(3):286–288, 1959.
52. A. M. Turing. Computing machinery and intelligence. *Mind*, 59:433–460, 1950.
53. J. Veness, K.S. Ng, M. Hutter, and D. Silver. A Monte Carlo AIXI Approximation. *Journal of Artificial Intelligence Research, JAIR*, 40:95–142, 2011.
54. C. S. Wallace. *Statistical and Inductive Inference by Minimum Message Length*. Springer-Verlag, 2005.
55. C. S. Wallace and D. M. Boulton. An information measure for classification. *Computer Journal*, 11:185–194, 1968.
56. C. S. Wallace and D. L. Dowe. Minimum message length and Kolmogorov complexity. *Computer Journal*, 42(4):270–283, 1999.
57. C. S. Wallace and D. L. Dowe. Refinements of MDL and MML coding. *Computer Journal*, 42(4):330–337, 1999.
58. F. Woergoetter and B. Porr. Reinforcement learning. *Scholarpedia*, 3(3):1448, 2008.
59. A. K. Zvonkin and L. A. Levin. The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. *Russian Mathematical Surveys*, 25:83–124, 1970.