Potential Properties of Turing Machines

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

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

3 August 2012

Abstract

In this paper we investigate the notion of potential properties for Turing 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 transitory 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 on how potential abilities can be measured.

Keywords: cognitive abilities, machine intelligence measurement, (universal) Turing machines, universality probability, potential intelligence, psychometrics, emergence.

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 [13][11][12][32][25][26][44][39][27][14][30][31]. 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). It 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.

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. 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 counterintuitive, 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 computable. 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 quite counter-intuitive, since our intuition says that some machines are potentially more intelligent than others.

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). 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).

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[1] [74], and also by Chaitin [3], 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, 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 [1] that this probability is strictly between 0 and 1. The universality probability was first suggested by Chris Wallace [6, footnote 70][8, 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 [1] 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. 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 taken, in order to make sense of the definition. From the transitory 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., interactive 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 closes the paper with a discussion of potential in terms of

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

'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., [33, 34]). 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 stake. 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[2]. 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 living or extinct 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 and any other artificial life beasts, we require a powerful set of cognitive tests. This is precisely the goal of a proposed new discipline, 'universal psychometrics' [28, 30]. 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.

These works are not based on Turing's imitation game [60] or its many extensions (see, e.g., [46]), but on the notion of learning, inductive inference, Turing machines and compression. In particular, Solomonoff's theory of inductive inference [53], the Minimum Message Length (MML) principle [64, 65, 71, 68, 63, 8][3] [4], algorithmic information theory [2], Kolmogorov complexity

[2]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 [30].

[3]see also [64, 67, 62, 70][63, sec. 6.8][61] for discussions of MML clustering and mixture modelling; and see also [20, 4, 5, 6, 8] for the theory of MML generalised Bayesian nets (or "inverse learning" [20]) - including the first papers [4, 5] on MML (hybrid) Bayesian nets (also known as causal nets or graphical models) with both continuous and discrete attributes. Combining this work on MML (hybrid) Bayesian networks [20, 4, 5][63, sec. 7.4][6, 8] with MML time series [22] should lead without too much difficulty to MML dynamic Bayesian nets.

[4]following footnote 3 on the Bayesian information-theoretic MML principle, we mention here some applications of MML to angular/directional distributions and data [66, 17, 70]. We also give some applications to biological data of MML (in particular) [15, 16, 73, 10, 21] (some of which [10, 21] uses directional distributions [66, 70] and even finds evidence of the order in which proteins fold [21, sec. 6][10, sec. 5, p253](see also [9])[62, sec. 4.2][6, footnote85][7, p454][8, sec. 4, p929]) and of information theory in general [49, 18, 48, 19]. We now take ideas from [1] in the direction of biology - or, more specificially, immunology. Following [1], we have since found an uncountable set of real numbers in the interval $[0, 1]$ (equivalently, an uncountable set of [infinite] binary strings), B, of measure 0 such that (i) each finite string, s, is a prefix to uncountably many elements of B, and (ii) for all Turing machines M for all $b \in B$ either $M(b)$ halts or $M(b)$ goes into an infinite loop (i.e., for some n_0, $M(b_1...b_{n_0}) = M(b')$ for any string b' for which the string $b_1...b_{n_0}$ is a prefix). Having thus defined B, one can likewise create a related set B' for which we replace condition (ii) above by condition (ii') for B' as follows: (ii') for all Turing machines M for all $b \in B'$ $M(b)$ is non-universal. Note that B' contains every element of B and (by [1, corollary 2.7]) B' is also of measure 0. We mention these notions of B and B' here in the context of biology because of seeming analogies between halting and

[37, 53] 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 [12], 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[5]. Related intelligence tests were also developed, such as the C-test [32] [25], composed of sequences of prediction problems that were generated by a universal distribution [53] and their difficulty assessed by a variant of Kolmogorov complexity. Other cognitive abilities were addressed in [26], 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 [44], and the 'universal intelligence measure' [39], 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 [27] 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 have since been done [35, 36, 40] for comparing human agents with non-human AI agents. 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 [14][30][31].

All the previous approaches focus on actual cognitive abilities, such as induction, deduction, planning, etc. Following [30], 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* selection *of information-processing tasks.*

Note that actual abilities are linked to performance and, ultimately, to the observational demonstration of the ability, and not 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

In contrast, the term 'potential' has already been used in psychology and other disciplines. For instance, 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" [45]. 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

(cell) death, between loss of universality and some depletion of function, and between going into an infinite loop and cancer.

[5] for another motivating example of the merits of compression in inductive (or statistical) learning and intelligence, see the uniqueness results of log(arithm)-loss probabilistic scoring in [6, footnote 175 (and 176)], [7, pp437-438] and [8, sec. 3]. These uniqueness results of log(arithm)-loss probabilistic scoring likewise carry over to Kullback-Leibler divergence [7, p438][8, sec. 3.6]. As per [58, sec. 4.2][6, sec. 0.2.5][7, p436][8, sec. 3.6 (and 7.6)], it is straightforward to define Kullback-Leibler divergence for (hybrid) Bayesian nets (or causal nets), with the log-loss scoring approximation having been used in [4, sec. 9]. This can similarly be done for mixture models [7, p436][8, sec. 3.6].

a great degree of implication and motivation. For instance, a very intelligent subject can score badly at an intelligence test if she is not properly motivated (e.g., rewards are not appropriate or well understood —see also [54, 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, which 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 achievable result leads to considering the result of a cognitive test as a *lower bound* of the actual ability. Note that with this interpretation, the right measure of an ability is exactly that of its potential, and measurements are typically below that value. This has led to approaches to convert this lower bound into a less biased estimate, trying to predict potential intelligence [59] or calculating how far a test can be from the actual measure [43]. Consequently, the difference between actual and potential is really 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 capacity of a system or individual of *acquiring or improving* 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). 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, 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 training. This gives a complementary (and essential) perspective for the notion of potential. Some potential abilities can only be acquired with appropriate training environments.

The study of the so-called training sequences for Turing machines was first discussed by Solomonoff [52] —and 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 [56]. 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 [63, sec. 2.3][6, footnote 70][8, 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 [23] 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

Let us denote by $\mathbb{B}$ the set $\{0,1\}$ and by $\mathbb{B}^*$ the set of 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 domain of strings by their length, where $\mathbb{B}^{m:n}$ denotes all the strings σ such that $m \leq |\sigma| \leq n$. Given a string $\sigma \in \mathbb{B}^*$, we use $\sigma_{a:b}$ to denote the substring between positions a and b inclusive (so having length $b - a + 1$). Given two strings σ and τ, the concatenation is simply denoted by $\sigma\tau$. The (cylinder)

set of all the strings starting with σ is denoted by $\sigma*$.

Any (possibly partial) computable function $M : \mathbb{B}^* \to \mathbb{B}^*$ can be calculated by a Turing machine, which we shall also call M. In order to properly analyse the concept of universal Turing machine, we will 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 prefix of another program)[6]. For the rest of the paper we will work with prefix-free Turing machines. Any Turing machine M *becomes* another Turing machine (denoted by $M[\tau]$) after being fed with an input string τ, i.e., for every finite 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 $\sigma \in \mathbb{B}^*$, $M(\sigma) = \lambda$. A halted machine is a null machine, but there may be null machines which make some other (possibly infinite) calculations and never output anything.

From here, and following, e.g., [1], we can define universality:

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

We can define a probability measure over these sets as follows:

Definition 3. *A probability measure w is a function $w : \mathbb{B}^* \to [0, 1]$ such that:*

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

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

For instance, we will use v to denote the probability measure such that for every natural number $n > 0$ and any pair of sequences $\sigma, \tau \in \mathbb{B}^n$, we have that $v(\sigma*) = v(\tau*) = 2^{-n}$. This measure v is known as the 'uniform' probability measure and represents the probability of strings 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 combinations. For instance, the universal semi-measure derived from UTM U, as the probability of a string being output by U with fair coin tosses as inputs, could be normalised as a probability measure μ_U (see, e.g., [42] for details), and would be a very different way of assigning probabilities to strings.

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

Definition 4. *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 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-void* property is a property ϕ for which there is at least one machine M and a constant $k \in \mathbb{R}$, with $0 < k \leq 1$ such that for every $n \in \mathbb{N}$ there are at least $\lceil k2^n \rceil$ strings $\sigma \in \mathbb{B}^n$ for which $\phi(M[\sigma]) > 0$. This means that there is at least one machine with the property and that it can keep the property indefinitely for a non-negligible set of inputs.

[6] A way to ensure that machines are prefix-free is that inputs can only be read sequentially. Input strings are hence not delimited and the Turing machine can stop reading eventually.

- A *genuine* property is a non-void 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-void 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. This analysis 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 (computable) 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 [74] and Chaitin [3]. 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 an infinite input 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 by [1], is:

Definition 5. *A string (or real) τ preserves property ϕ with respect to a prefix-free machine M, denoted by $preserves(\phi, \tau, M)$ if all machines M_n defined from M, τ and $n \in \mathbb{N}$ as $M_n \triangleq M[\tau_{1:n}]$, are such that all M_n also have property ϕ.*

However, the preserving probability needs to be slightly extended from [1] if we want to generalise it to any property. First, we may need to separate *lower* and *upper* ϕ-preserving probability:

Definition 6. *The* lower *ϕ-preserving probability for machine M, denoted by $\check{P}^{\phi}_{M}$, is defined as:*

$$\liminf_{t\to\infty} \frac{1}{2^t} card\{\tau \in \mathbb{B}^t : preserves(\phi, \tau, M)\}$$

where card is the cardinality of a set.

Definition 7. *The* upper *ϕ-preserving probability for machine M, denoted by $\hat{P}^{\phi}_{M}$, is defined as:*

$$\limsup_{t\to\infty} \frac{1}{2^t} card\{\tau \in \mathbb{B}^t : preserves(\phi, \tau, M)\}$$

where card is the cardinality of a set.

Proposition 1. *For every ϕ, the upper ϕ-preserving probability and lower ϕ-preserving probability are equal, i.e., the limit exists.*

Proof. We see that for every $t \in \mathbb{N}$ we have:

$$\frac{1}{2^t} card\{\tau \in \mathbb{B}^t : preserves(\phi, \tau, M)\} \geq \frac{1}{2^{t+1}} card\{\tau \in \mathbb{B}^{t+1} : preserves(\phi, \tau, M)\}$$

since whenever a string τ does not preserve the property for M, then for every $\sigma \in \mathbb{B}^*$ the string $\tau\sigma$ also does not preserve the property (for M), because (otherwise) definition 5 would force all prefices (such as the prefix τ of $\tau\sigma$) to lead to machines that have the property. Consequently, since the value is non-increasing, the limit (as $t \to \infty$) exists. □

Since the two probabilities (from definitions 6 and 7) are equal, the limit exists and leads to the ϕ-preserving probability:

Definition 8. *The ϕ-preserving probability for machine M, denoted by P_M^ϕ, is the measure of the set of all reals (strings) which preserve property ϕ with respect to M.*

If we take ζ, this probability was first suggested by Chris Wallace [6, footnote 70][8, sec. 2.5], and conjectured to be always 0. However, it has been shown in [1] that this probability is strictly between 0 and 1 for any UTM. There is a special thing about ζ, then; once it is acquired it cannot be lost for all sequences (even though it must necessarily 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 other properties which are permanent, this means that the machine cannot be universal.

Proposition 2. *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. 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 [1], 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 ability and keeps it forever, 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 ζ, but may develop it after some inputs. 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* 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, we may draw an evolution of the property as shown in Figure 1 (left).

Definition 8 above takes one important thing implicitly, the measure of all reals (or strings) v, the 'uniform' probability measure. 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 strings are generated by another (possibly universal) Turing machine, which would lead to a different weight for each input sequence and, consequently, a different overall result in definition 8. In fact, this weighting over a probability measure is at the core of some fundamental results in inductive inference, such as the expected value of squared prediction error, given in [55, Theorem 3 (17)].

3.1 Point potential and period potential

Now, instead of considering how probable a Boolean property will be in the limit, we can just define the expected value of a non-Boolean property for a *point* t as follows:

Definition 9. *The* point *potential of property ϕ for machine M at point t, under a string measure w, is given by:*

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

We can better understand the meaning of potential graphically. Figure 1 (right) shows a figurative evolution of point potential $\dot{\Pi}(M, \phi, t, w)$ for increasing values of t. 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:

Definition 10. *The* period *potential of property ϕ for machine M for a period $[a, b]$ ($a \leq b$), under a string 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. Note that definition 10 also works for Boolean properties and the result is an estimated probability.

Figure 1 (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 segment 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. For the rest of the paper, we will just stick to definition 10 as an expected value.

Clearly, the ϕ-preserving probability for machine M introduced in definition 8, denoted by P_M^ϕ is just equal to $\lim_{t\to\infty} \dot{\Pi}(M, \phi, t, v)$. And the universality probability is just $\lim_{t\to\infty} \dot{\Pi}(M, \zeta, t, v)$, which we know is strictly between 0 and 1 for M a UTM. In fact, with a similar rationale, we can show that this holds for any *genuine* property[7], which can be seen as an extension (or corollary) of Theorem 2.4 in [1].

[7] As we did for the definitions of upper and lower ϕ-preserving probabilities, we use 'lim inf' and 'lim sup' below because, for some properties, the limit may not exist, but potential may still be bounded.

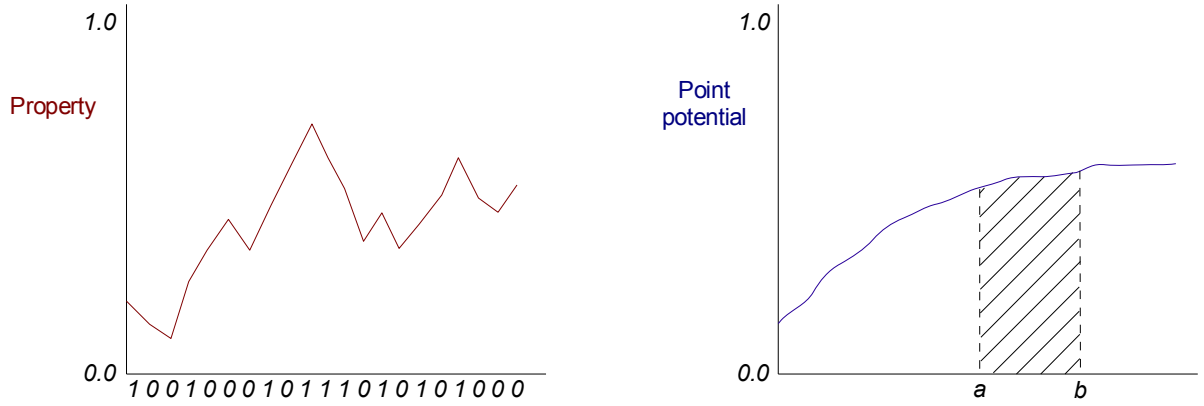

Figure 1: Left: Figurative evolution of a property for a given string (on the x-axis). Right: Figurative evolution of point potential $\dot{\Pi}(M, \phi, t, w)$. The period potential $\Pi(M, \phi, a, b, w)$ is just the average of point potentials for the period $t \in [a, b]$.

Proposition 3. *For any genuine property ϕ 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 [3], since some programs make a UTM halt and a genuine property is 0 for these programs. Let us choose one of these programs, p, of length $|p| = t$, and the machine $M = U[p]$. By construction, M is a null machine. A genuine property is 0 for null machines and clearly kept 0 forever. Consequently there is a value (at most $t + 1$) from which the point potential cannot be 1. The > 0 case is just a consequence of a genuine property being a non-void property, which means that there is at least one Turing machine M and a $k > 0$ such that there are $\lceil k2^n \rceil$ strings $\sigma \in \mathbb{B}^n$ for every $n > 0$ such that we have $\phi(M[\sigma]) > 0$. This, and the use of v, ensures that the value of the property for M in the limit is strictly greater than 0. Since U is a UTM, it can become M for at least one input string τ (of, say, length t), so there is a positive real constant c (which depends on k and $v(\tau*)$) such that potential is always greater than c from inputs of size $t + 1$. □

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 frequent the property can be achieved and when.

In fact, 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 and transitory potentials. In practice we are interested in (or implicitly assume) transitory potentials. For instance, when we say that a baby is potentially intelligent, we mean that it will have a certain degree of intelligence during a period of her life, say, between 20 years and 60 years, provided she has a somewhat common education.

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

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 $min\{t : \Pi(M, \phi, t, t, w) \geq c\}$. 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., $\{w : \Pi(M, \phi, t, t, w) \geq c\}$ 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 one sequence. In the latter case, this could be understood as *a* perfect training sequence. We will get back on this issue later on.

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 observation, so the distinction may be important from a conceptual point of view but not really significant in a practical way.

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 period potential for 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 finite 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 2 does not apply for n-universality.

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 any given task, as a job, especially under punishment or under threat. In a voluntary way, 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. 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 [31].

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 [51]. 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 [38], then, by emulating it (and again 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, 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 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 2, we will re-interpret the machine kingdom Ω as the set of all resource-bounded interactive agents, as in [30].

4.1 Considering computational resources

Reinforcement learning [72] 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), we can easily extend the notion of interactive agent considering 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,* $\mathcal{S}$ *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:* $\mathcal{S} \times \mathcal{I} \to \Delta\mathcal{S}$*, and* $\dot{o}(s, i)$ *is an output function:* $\mathcal{S} \times \mathcal{I} \to \mathcal{O}$*.*

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. '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 can adapt the definitions in the previous section to interactive systems. Instead of input strings, we consider interaction histories between the environment and the agent. Since the distribution of interaction histories depends on both the agent and the environment, it is easier to work with environment distributions.

Universality is easily understood in this setting as the property of an agent behaving like any other given agent from a given time t, after an interaction history in the appropriate environment. Environments are then seen as programs, although this is not the usual way of programming an agent in AI (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 in an environment μ emulates α_2 during a period $[a, b]$, $a, b \in \mathcal{T}$, $(a \leq b)$, if the behaviour of α_1 after time a in μ equals the behaviour of α_2 for every possible (continuation) environment ν, during a time $b - a$.*

If $b = \infty$ we say that agent α_1 *becomes* α_2. The first difference with section 3 appears because, when we consider time and other resources, there is no universal agent[8].

While the concept of universality is elusive when considering computational resources, the notion of potential can be easily adapted from the version for non-iterative TMs:

Definition 15. *The potential of property ϕ for agent α for a period $[a, b]$, $a, b \in \mathcal{T}$, $(a \leq b)$ under an environment distribution w, denoted by $\Pi(\alpha, \phi, a, b, w)$, is the expected value of ϕ between times a and b in the distribution of environments w.*

4.2 Environments and kinds of potential

The use of a distribution of environments emphasises that potential abilities represent expected values over an astronomical range of possibilities. In the case of interactive environments considering time, we may have a huge amount of environments which are either too slow or too fast, are repetitive or apparently random, so they will have almost no effect on a potential ability. Actually, in many cases, the ability will develop if the agent is programmed to do it after a given period.

The notion of *speed*, seen in the previous section 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 (more precisely, an exhaustive distribution) of environments will be clearly 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 [69, p335, sec.3], would have no interesting stimuli. As a result, the expectation should be calculated with an appropriate (presumably much concentrated) distribution, which accounts for those 'lives' we are interested in or the agent may face. One solution for this is the so-called Darwin-Wallace distribution for environments, as introduced in [29]. This distribution is conceived for measuring (social) intelligence, but other distributions could be used for other abilities. These distributions could then be used for the calculation of potential or acquisition speed.

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: $min\{t : \exists w \ \Pi(\alpha, \phi, t, t + s, w) \geq c\}$. If this minimum value does not exist, the optimistic speed is infinite.*

Definition 16 has many possible alternative formulations, meaning different things, such as $argmax_w \Pi(\alpha, \phi, t, t, w)$ 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 [56] (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

[8]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 more infeasible).

would possibly require fewer bits of information to identify which 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 supports the study of perfect training sequences for machines using Levin's optimal search [41] (or any other learning machine), as Solomonoff did [56], 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 [29], 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 the last section, 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?". 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 analyse 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 counterintuive.

The usefulness of potential abilities can even be wider if we are able to *measure* them. And we mean measuring them by the observation of their behaviour, using tests or related mechanisms, and not by analysing the DNA or the code of the system.

Apparently, the evaluation of potential abilities will be generally more difficult than the evaluation of 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 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 [54, sec. 6], bad interfaces, etc. It is not then the problem referred to by other uses of the term potential in psychology [45][59][43], as discusssed 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 the set of all possible agents, 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 potential of a distribution of agents κ is:*

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

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 when she is 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 two general approaches for measuring potential intelligence:

1. An analysis of the evolution of an ability for a similar individual for a similar distribution of environments can give information about saturation points and when they will be reached. For instance, from the rightmost plot of Figure 1 (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 a distribution of environments, sampling is necessary.

2. Another 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 distribution of environments of interest.

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 distributions of environments which include rewards and interfaces (or these are kept constant), because otherwise the results would not be extrapolable (for a related discussion see, e.g., [54, sec. 6][30]). 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][27, 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[9].

6 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, was a potential universality-holder[10]. 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[63, 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 [57] studied several stages in the process towards the technical 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.

There is also no real difference between an artificial life system running and a fantastic world running on a human's mind, such as figuring out a different world by imagination or after reading a novel. 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 computable function. Minds and machines are just the substrate to execute them. Since natural

[9]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 $M_R(100101) = 01101001$.

[10]In fact, Conway's game of life [23] 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 Turing machine.

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 [63, 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' [47][50].

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. Also, it implies that the machine can halt (i.e., die). However, other more restrictive views of the universality property are more compatible (or even intrinsic) with intelligence, such as temporary universality (by temporary 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 potential, and this and other concepts should be very clear before we want to extend them to AI artefacts.

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 [24]) and some training (optimal training sequences [52]). 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 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. In other words, machines should be designed to be potentially intelligent rather than intelligent.

References

[1] 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.

[2] G. J. Chaitin. On the length of programs for computing finite sequences. *Journal of the Association for Computing Machinery*, 13:547–569, 1966.

[3] G. J. Chaitin. A theory of program size formally identical to information theory. *Journal of the ACM (JACM)*, 22(3):329–340, 1975.

[4] J. W. Comley and D. L. Dowe. General Bayesian networks and asymmetric languages. In *Proc. Hawaii International Conference on Statistics and Related Fields*, 5-8 June 2003.

[5] J. W. Comley and D. L. Dowe. Minimum message length and generalized Bayesian nets with asymmetric languages. In P. Grünwald, M. A. Pitt, and I. J. Myung, editors, *Advances in Minimum Description Length: Theory and Applications (MDL Handbook)*, pages 265–294. M.I.T. Press, April 2005. Chapter 11, ISBN 0-262-07262-9. Final camera-ready copy submitted in October 2003. [Originally submitted with title: "Minimum Message Length, MDL and Generalised Bayesian Networks with Asymmetric Languages".].

[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. Minimum Message Length and statistically consistent invariant (objective?) Bayesian probabilistic inference - from (medical) "evidence". *Social Epistemology*, 22(4):433 – 460, October - December 2008.

[8] 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.

[9] D. L. Dowe, L. Allison, T. I. Dix, L. Hunter, C. S. Wallace, and T. Edgoose. Circular clustering by minimum message length of protein dihedral angles. Technical report CS 95/237, Dept Computer Science, Monash University, Melbourne, Australia, 1995.

[10] D. L. Dowe, L. Allison, T. I. Dix, L. Hunter, C. S. Wallace, and T. Edgoose. Circular clustering of protein dihedral angles by minimum message length. In *Pacific Symposium on Biocomputing '96*, pages 242–255. World Scientific, Jan 1996.

[11] 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.

[12] 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.

[13] 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*, September 1997.

[14] D. L. Dowe and J. Hernández-Orallo. IQ tests are not for machines, yet. *Intelligence*, 40(2):77 – 81, 2012.

[15] D. L. Dowe, J. J. Oliver, L. Allison, C. S. Wallace, and T. I. Dix. Learning rules for protein secondary structure prediction. In *Dept. Research Conference, Comp. Sci. Dept., University of WA, ISBN 0 86422 195 9*, 1992. Also available as TR 163, Dept. of Computer Science, Monash University, Clayton, Victoria 3168, Australia.

[16] D. L. Dowe, J. J. Oliver, L. Allison, C. S. Wallace, and T. I. Dix. A decision graph explanation of protein secondary structure prediction. In *Proceedings of the 26th Hawaii International Conference on System Sciences (HICSS-26), Biotechnology Computing Track*, volume 1, pages 669–678, 1993.

[17] D. L. Dowe, J. J. Oliver, and C. S. Wallace. MML estimation of the parameters of the spherical Fisher distribution. In *Algorithmic Learning Theory, 7th International Workshop, ALT '96, Sydney, Australia, October 1996, Proceedings*, volume 1160 of *Lecture Notes in Artificial Intelligence*, pages 213–227. Springer, October 1996.

[18] D. L. Dowe and K. Prank. Complexity and information-theoretic approaches to biology, conference stream introduction. In *Proc. 3rd Pacific Symposium on Biocomputing (PSB-98)*, pages 559–560, Hawaii, USA, Jan 1998.

[19] D. L. Dowe and K. Prank. Information-theoretic approaches to biology, conference stream introduction. In *Pacific Symposium on Biocomputing*, pages 252–253, Hawaii, USA, January 1999.

[20] D. L. Dowe and C. S. Wallace. Kolmogorov complexity, minimum message length and inverse learning. In W Robb, editor, *Proceedings of the Fourteenth Biennial Australian Statistical Conference (ASC-14)*, page 144, Broadbeach, Gold Coast, Queensland, Australia, July 1998.

[21] T. Edgoose, L. Allison, and D. L. Dowe. An MML classification of protein structure that knows about angles and sequence. In *Pacific Symposium on Biocomputing '98*, pages 585–596. World Scientific, Jan 1998.

[22] L. J. Fitzgibbon, D. L. Dowe, and F. Vahid. Minimum message length autoregressive model order selection. In *Proc. Int. Conf. on Intelligent Sensors and Information Processing*, pages 439–444, Chennai, India, Jan 2004.

[23] M. Gardner. Mathematical games: The fantastic combinations of John Conway's new solitaire game "life". *Scientific American*, 223(4):120–123, 1970.

[24] 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)*, 2009.

[25] J. Hernández-Orallo. Beyond the Turing Test. *J. Logic, Language & Information*, 9(4):447–466, 2000.

[26] 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.

[27] J. Hernández-Orallo and D. L. Dowe. Measuring universal intelligence: Towards an anytime intelligence test. *Artificial Intelligence*, 174(18):1508 – 1539, 2010.

[28] J. Hernández-Orallo and D. L. Dowe. Mammals, machines and mind games. Who's the smartest?. *The Conversation, http://theconversation.edu.au/*, April 2011.

[29] 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.

[30] 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.

[31] 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.

[32] 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.

[33] 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.

[34] 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, 2010.

[35] 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.

[36] 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.

[37] A. N. Kolmogorov. Three approaches to the quantitative definition of information. *Problems of Information Transmission*, 1:4–7, 1965.

[38] S. Legg. *Machine Super Intelligence*. Department of Informatics, University of Lugano, June 2008.

[39] S. Legg and M. Hutter. Universal intelligence: A definition of machine intelligence. *Minds and Machines*, 17(4):391–444, 2007.

[40] S. Legg and J. Veness. An Approximation of the Universal Intelligence Measure. In *Proceedings of Solomonoff 85th memorial conference*. Springer, 2012.

[41] L. A. Levin. Universal sequential search problems. *Problems of Information Transmission*, 9(3):265–266, 1973.

[42] M. Li and P. Vitányi. *An introduction to Kolmogorov complexity and its applications (3rd ed.)*. Springer-Verlag, 2008.

[43] 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.

[44] 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.

[45] A. R. Mahrer. Potential intelligence: a learning theory approach to description and clinical implication. *The Journal of General Psychology*, 59(1):59–71, 1958.

[46] 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/.

[47] 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.

[48] D. R. Powell, L. Allison, T. I. Dix, and D. L. Dowe. Alignment of low information sequences. In *Australian Computer Science Theory Symposium (CATS '98)*, pages 215–230. Springer Verlag, 2-3 February 1998.

[49] D. R. Powell, D. L. Dowe, L. Allison, and T. I. Dix. Discovering simple DNA sequences by compression. In *Pacific Symposium on Biocomputing '98*, pages 597–608. World Scientific, Jan 1998.

[50] 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.

[51] 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.

[52] R. J. Solomonoff. Training sequences for mechanized induction. *Self-organizing systems, eds., M. Yovit, G. Jacobi, and G. Goldsteins*, 7:425–434, 1962.

[53] R. J. Solomonoff. A formal theory of inductive inference. *Information and Control*, 7:1–22, 224–254, 1964.

[54] 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.

[55] R. J. Solomonoff. Complexity-based induction systems: comparisons and convergence theorems. *Information Theory, IEEE Transactions on*, 24(4):422–432, 1978.

[56] R. J. Solomonoff. Perfect training sequences and the costs of corruption - a progress report on induction inference research. *Oxbridge Research*, 1984.

[57] R. J. Solomonoff. The Time Scale of Artificial Intelligence: Reflections on Social Effects. *Human Systems Management*, 5:149–153, 1985.

[58] P. J. Tan and D. L. Dowe. Decision forests with oblique decision trees. In *Lecture Notes in Artificial Intelligence (LNAI) 4293 (Springer), Proc. 5th Mexican International Conf. Artificial Intelligence*, pages 593–603, Apizaco, Mexico, Nov. 2006.

[59] T. R. Thorp and A. R. Mahrer. Predicting potential intelligence. *Journal of Clinical Psychology*, 15(3):286–288, 1959.

[60] A. M. Turing. Computing machinery and intelligence. *Mind*, 59:433–460, 1950.

[61] Gerhard Visser and D. L. Dowe. Minimum message length clustering of spatially-correlated data with varying inter-class penalties. In *Proc. 6th IEEE International Conf. on Computer and Information Science (ICIS) 2007*, pages 17–22, July 2007.

[62] C. S. Wallace. Intrinsic classification of spatially correlated data. *Computer Journal*, 41(8):602–611, 1998.

[63] C. S. Wallace. *Statistical and Inductive Inference by Minimum Message Length*. Springer-Verlag, 2005.

[64] C. S. Wallace and D. M. Boulton. An information measure for classification. *Computer Journal*, 11:185–194, 1968.

[65] C. S. Wallace and D. M. Boulton. An invariant Bayes method for point estimation. *Classification Society Bulletin*, 3(3):11–34, 1975.

[66] C. S. Wallace and D. L. Dowe. MML estimation of the von Mises concentration parameter. Technical Report 93/193, Dept. of Computer Science, Monash University, Clayton 3168, Australia, December 1993.

[67] C. S. Wallace and D. L. Dowe. Intrinsic classification by MML - the Snob program. In *Proc. 7th Australian Joint Conf. on Artificial Intelligence*, pages 37–44. World Scientific, November 1994.

[68] C. S. Wallace and D. L. Dowe. Minimum message length and Kolmogorov complexity. *Computer Journal*, 42(4):270–283, 1999.

[69] C. S. Wallace and D. L. Dowe. Refinements of MDL and MML coding. *Computer Journal*, 42(4):330–337, 1999.

[70] C. S. Wallace and D. L. Dowe. MML clustering of multi-state, Poisson, von Mises circular and Gaussian distributions. *Statistics and Computing*, 10:73–83, January 2000.

[71] C. S. Wallace and P. R. Freeman. Estimation and inference by compact coding. *Journal of the Royal Statistical Society series B*, 49(3):240–252, 1987. See also Discussion on pp252-265.

[72] F. Woergoetter and B. Porr. Reinforcement learning. *Scholarpedia*, 3(3):1448, 2008.

[73] J. D. Zakis, I. Cosic, and D. L. Dowe. Classification of protein spectra derived for the Resonant Recognition model using the Minimum Message Length principle. In *Proc. 17th Australian Computer Science Conference (ACSC-17)*, pages 209–216, January 1994. Christchurch, NZ.

[74] 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.