Beyond the Turing Test

Jose Hernandez-Orallo
Dep. de Sistemes Informàtics i Computació, Universitat Politècnica de València Camí de Vera, s/n, E-46022 València (Spain) Email: jorallo@dsic.upv.es

May 15, 1999

Abstract. We define the main factor of intelligence as the ability to comprehend, formalising this ability with the help of new constructs based on descriptional complexity. The result is a comprehension test, or C-test, exclusively defined in terms of universal descriptional machines (e.g universal Turing machines). Despite the absolute and non-anthropomorphic character of the test it is equally applicable to both humans and machines. Moreover, it correlates with classical psychometric tests, thus establishing the first firm connection between information theoretic notions and traditional IQ tests. The Turing Test is compared with the C-test and their joint combination is discussed. As a result, the idea of the Turing Test as a practical test of intelligence should be left behind, and substituted by computational and factorial tests of different cognitive abilities, a much more useful approach for artificial intelligence progress and for many other intriguing questions that are presented beyond the Turing Test.

Keywords: Measurement of Intelligence, Descriptional Complexity, Turing Test, Comprehension, Psychometrics, AI's Anthropomorphism.

1. Introduction

"*The original question, 'Can machines think?' I believe to be too meaningless to deserve discussion*" (Turing 1950). Turing's intention was to leave behind that question and motivate the endeavour for making machines intelligent. Fifty years of philosophical discussions have shown that his goal has not been fulfilled, mainly because the Turing Test (TT) has been usually misunderstood as a test for intelligence and not simply as an imaginary test of humanity (Fostel 1993), a philosophical exercise contrived to stop the sterile debate that the advent of the first computers were generating.

Turing cannot be reproached for these misunderstandings. The TT "*was never intended as a test for intelligence, at least not by Turing himself, and it should not be criticized for failing something that it was not supposed to do*" (Larsson 1993). In fact, there is no better way to state that intelligence can be reasonably judged by observation than Turing's imitation game, and that there is no reason to deny the intelligence of non-human beings.

Nonetheless, the overuse of the TT as a referent or even a test of AI progress has had very negative consequences. AI has striven to imitate human's behaviour in many tasks, under the slogan "*Artificial intelligence is that thing that if made by humans would require intelligence*", which has promoted the view that "*human intelligence subsumes machine intelligence*" (Bradford and Wollowski 1995). Finally, the most negative result of the

© 1999 *Kluwer Academic Publishers. Printed in the Netherlands.*

misinterpretation of the TT jointly with his celebrity is that there has not been the necessary effort for designing new alternative intelligence tests. The TT has even eclipsed such well-reputed proposals as Simon's early works on the relation between IQ tests and AI (Simon and Kotovsky 1963), on some heuristic approaches to solve analogy problems from IQ tests (Evans 1963), Chaitin's suggestion "*develop formal definitions of intelligence and measures of its various components*" (Chaitin 1982). Even Johnson's call "*Needed: A New Test of Intelligence*" (Johnson 1992) has been responded by formalisations of the TT (Bradford and Wollowski 1995), once again.

From a practical point of view, it is vain to revive the little informative question "*Can Machines Think?*" (Epstein 1992) which has generated many words but few clues about how to progress in AI or how to understand intelligence. As any other discipline, AI requires an effective measure of its major issue, a gradual and detailed measure of intelligence. Concretely, a scientific measure of intelligence should comply with some requirements:

- **Non-Boolean** : intelligence is not an absolute attribute. From Darwin's "mental continuity" to infant psychology, there is an unquestionable certainty that intelligence is a gradual aptitude. Any gradation of the TT as a function of the time of the test or the score of the judges shows the inappropriateness of the TT to measure intelligence in a gradual way, i.e., to give a continuous value of intelligence. The reason is obvious: the TT is a test of humanity (Fostel 1993), and the idea of being more or less human makes no sense.
- **Factorial** : intelligence is multi-dimensional. It is quite unbelievable that there is a concrete ability that would be optimal for every context or world. Nonetheless, it is conceivable that a concrete ability would be almost optimal for most contexts, and intelligence has sometimes been defined as "second-best in everything". This may justify that a wide context as everyday life, which includes many other contexts, can distinguish a special kind of ability, the g factor, the primitive concept found in everyday life.
- **Non-anthropomorphic** : Maybe the major problem of AI is that its reference of intelligent behaviour is human intelligence. Recently, AI researchers have paid attention to other 'intelligences': ants, rats, etc. in order to scale up the problem. However, the reference or the goal is always human intelligence. Nowadays the reason is not only pride and anthropocentrism but necessity, because "*there is not yet a solid definition of intelligence that doesn't depend on relating it to human intelligence*" (McCarthy 1998).
- **Computationally based** : According to the Church-Turing thesis, there is no reason to think that intelligent systems cannot be implemented in current computers. The only question is whether they have the necessary speed and memory for it. I share the opinion that "*computers of 30 years ago were fast enough if only we knew how to program them*" (McCarthy 1998), and the AI problem is just to discover what makes a program intelligent, or, in other words, "*What kind of Information Processing is Intelligence?*" (Chandrasekaran 1990). Consequently, it is extremely significant to be able to state the problem computationally, namely, to give the specification of the problem

in computational terms, in order to solve the problem with AI means, which are exclusively machines and programs.

- **Meaningful** : intelligence is not an ethereal ability. It cannot be defined as "*intelligence is what is measured by intelligence tests*". Intelligence must be expressed from its original meaning, the ability to comprehend, and this ability is what should be measured.

Psychometrics has developed measurements of intelligence according to the first two requirements. Since Spearman founded the field (Spearman 1904), psychometrics has been more and more characterised by the scientific method: systematic experimentation and statistical rigour. "*Despite the many shortcomings of an IQ score, no other measure has been found to be related to so many other behaviors of theoretical or practical significance*" (Zigler and Seitz 1982). However, psychometrics has neglected, or failed, to incorporate the last three requirements, which, in fact, are highly related. Psychometrics is anthropomorphic by definition since it is the science of measuring human intelligence. Although there have been adaptations and essays with chimpanzees, dolphins and other animals, the reference is always the Homo Sapiens Sapiens. The frequently demanded theoretical foundation of psychometrics depends on the change of the point of reference, closely connected to the last three requirements.

On the contrary, Computational Learning Theory is non-anthropomorphic and computational. The question "What is to learn?" has been assimilated to the formal notion of identification in the limit (Gold 1967). After some discouraging results on the learnability of very simple languages, the complexity of learning has been studied for other paradigms, mainly PAC learning (Valiant 1984) and Query Learning (Angluin 1988).

However, these theoretical results have not been used to develop rigorous measurements of learning ability. Contrarily, some contests and comparisons have been held for practical systems, but as collections of arbitrary examples extracted from the literature, without many justifications of the theoretical complexity of each of them. The reason is that it is difficult to establish the complexity of an instance, when the results of computational learning theory apply to classes of concepts, and most results are asymptotical. Actually, the paradigm of identification in the limit is not applicable for finite instances, because they can be identified by themselves.

Moreover, the philosophical problem of any measurement of learning ability is the same as the philosophical problem of prediction: given any finite set of examples, there are infinite many concepts which are consistent with them and diverge in their predictions. This is exactly the 'subjectivity objection' of IQ tests: there may be controversy about the *correct* answer.

In the following we adopt an information-theoretic approach to solve these problems in order to give a definition and a measure of intelligence according to the five prerequisites mentioned above, exclusively based on concepts derived from the notion of Turing machine. Despite the title of this paper, the result is a veritable tribute to Alan Turing on the 50th anniversary of his celebrated paper.

The paper is organised as follows. The following section introduces the necessary definitions and tools for the rest of the paper, as well as some technical difficulties which are solved in the subsequent sections. In particular, section 3 formalises the initially vague notion of comprehension in information-theoretical terms and relates it with the *ontology* that only intelligent systems are able to construct. Once settled on a computational setting for comprehension, Section 4 deals with its measurement by solving the 'subjectivity objection' under the notion of unquestionability, and by ordering the difficulty of instances. This allows the construction of a comprehension test (C-test). Section 5 presents the results of applying it to humans and compares it with psychometrical tests. The applicability to AI is discussed. Section 6 studies the measurement of other factors (knowledge applicability, contextualisation, knowledge construction) under the same conditions that the C-test has been devised with.

After the previous results and auspices, the TT is re-examined in section 7 and reduced to its original philosophical and even metaphorical character. Compared with the C-Test, the significance of the TT is recognised, as well as the acute deficiencies of its misinterpretation and incarnations, like the Loebner Prize. The final section concludes with the proposal of a radical change of paradigm; a science of intelligence that would make it feasible to answer many new and fascinating questions.

2. Preliminaries and Technical Problems

We choose any finite alphabet Σ composed of symbols (if not specified, $\Sigma = \{0, 1\}$). A string or object is any element from Σ^*, being $\cdot$ the composition operator, usually omitted or represented by $\langle a, b\rangle = a \cdot b$. The empty string or empty object is denoted by ε. The term $l(x)$ denotes the length or size of x in bits and $log\, n$ will always denote the binary logarithm of n. For every string y, we denote with $y_{n..m}$, being $n \leq m$, the symbols from position n to position m. For every natural number n, $y_{n..n} = \varepsilon$. With $y_{..m}$, $y_{n..}$, and y_k we denote $y_{0..m}$, $y_{n..l(y)}$, and $y_{k..k+1}$, respectively. A string x is a substring of y iff there exist two strings z, w such that $y = zxw$, or, what is equivalent, $y_{n..m} = x$ with $n = l(z)$ and $m = l(z) + l(x)$. A string x is a prefix of y iff exists a string z such that $y = xz$, or, what is equivalent, $x = y_{0..m}$ being $m = l(x)$. Given any string x, we denote by $x_{-d} = x_{0..l(x)-d}$ the prefix of x with length $l(x) - d$, i.e. the string x without its last d elements.

Under the Church-Turing thesis, any *description* of any object of reality can be converted into a description in a universal computational machine (e.g. a universal Turing machine), so our idea of 'object' is, in this sense, universal. Given a universal descriptional machine ϕ, and a program p, we denote by $\phi(p)$ the result of executing p on ϕ. Two descriptions p and p' are extensionally equivalent (denoted $p \equiv p'$) iff $\phi(p) = \phi(p')$. It is easy to see that any description has infinite equivalent descriptions.

Definition 1. A k-**projectible description** for a string x is a program p on a descriptional mechanism ϕ such that $\exists y \in \Sigma^* : \phi(p) = y$, and $\exists w\, l(w) = k : y = xw$ (i.e. $x = y_{0..l(x)}$). w is known as the *prediction* of p.

The compression ratio of a program p is given by: $CR_\phi(p) = l(\phi(p))/l(p)$. The compression ratio of an infinite projectible description is always infinite. The relative compression ratio of a projectible description p for a finite string x is defined as $CR_\phi(p|x) = l(x)/l(p)$

The complexity of an object can be measured in many ways, one of them being its degree of randomness (Kolmogorov 1965), which turns out to be equal to the shortest description of it. Descriptional Complexity, Algorithmic Complexity or Kolmogorov Complexity was independently introduced by Solomonoff, Kolmogorov and Chaitin to formalise this idea, and it has been gradually recognised as a key issue in statistics, computer science, AI, epistemology and cognitive science (see e.g. Li and Vitányi 1997).

Definition 2. The **Kolmogorov Complexity** of an object x given y on a descriptional mechanism (or bias) β is defined as:

$$K_\beta(x|y) = min\{l(p) : \phi_\beta(\langle p, y\rangle) = x)\}$$

where p denotes any "prefix-free" β-program, and $\phi_\beta(\langle p, y\rangle)$ denotes the result of executing p using input y.

The complexity of an object x is denoted by $K_\beta(x) = K_\beta(x|\varepsilon)$. It can be seen elsewhere (e.g. Li and Vitányi 1997) that Kolmogorov Complexity is an absolute and objective criterion of complexity, and it is independent (up to a constant term) of the descriptional mechanism β. In other words, there is an invariance theorem that states that any universal machine can emulate another. For this reason many properties are proven just in an asymptotic way and β is usually omitted. Throughout the paper we will use the relation $<^+$ as the asymptotical extension of $<$, namely $a <^+ b$ iff there exists an independent positive constant k such that $a < b + k$. We will denote x^* as the first (in lexicographical order) program for x such that $K(x) = l(x^*)$. [1]

$K(x)$ also represents the idea of simplicity, deeply involved in the philosophy of inductive reasoning and learning theory. It is not surprising that learning and recognition were soon re-understood under this context. Solomonoff

[1] It is obvious to see that $\forall x \in \Sigma^*\ K(x|x) <^+ 0$ because there is always a program of constant size of the form "*print the input*". It is also easy to see that $\forall x \in \Sigma^*\, K(x) <^+ l(x)$ because there is always a program of size less than $l(x)$ plus a constant value of the form "*print x*". In the case that $K(x) \geq l(x)$ (i.e. $l(x^*) \geq l(x)$) we say that x is random. It is obvious that $K(x|x^*) <^+ 0$. However, since $K(\cdot|\cdot)$ is not computable, it is shown elsewhere (e.g. Li and Vitányi 1997) that the other way is just $K(x^*|x) <^+ log\, l(x)$.

proposed the view of unsupervised learning as information compression (Solomonoff 1964) and Watanabe considered "pattern recognition as information compression" (Watanabe 1972). The classical Occam's razor of the scientific method: "given two alternative explanations, choose the simplest one" was formalised by Rissanen in 1978 under the name "Minimum Description Length" (MDL) Principle, finally re-formulated in its current one part code (Rissanen 1996).

According to the MDL principle, given any sequence x, the optimal model in ϕ for it is x^*. If x^* is projectible, i.e. it allows the prediction of the subsequent symbols of the sequence x, then $\phi(x^*)_{n+1}$ will be the most plausible prediction according to Occam's razor. In order to ensure that x^* is projectible we need to introduce a projectible variant of K:

Definition 3. The k-**Projectible Kolmogorov Complexity** of an object x given y on a descriptional mechanism (or bias) β is defined as: $K'_\beta(x|y) = min\{\, l(p) : \exists w \in \Sigma^* \; l(w) = k$ such that $\phi_\beta(\langle p, y\rangle) = xw)\}$ where p denotes any "prefix-free" β-program.

The literature has used Kolmogorov Complexity and not its projectible variant for prediction due to the following theorem:

Theorem 1. For every string x, $K'(x) <^+ K(x)$.

Proof. Every non-projectible program p can be transformed into a projectible program p' = "execute p and then print 1 forever". Let us denote by c the length of this extra coding of "and then print 1 forever". Hence, there exists a constant $k = c + 1$ such that $l(p') < l(p) + k$, i.e. $l(p') <^+ l(p)$. This can be extended to the definitions of K' and K, thus the theorem is proven. □

The contrary relationship ($K(x) <^+ K'(x)$) does not hold. Consider the string x = "1,2,3, ..., n". The projectible program p' = "print the natural numbers, ordered" has constant size, say $l(p') = c$. On the contrary, the non-projectible program p = "print the first n natural numbers, ordered" is, in the general case, not smaller than $c' + log\, n$.

Thus, the ideal MDL principle is represented by the first (in lexicographical order) projectible description for x, denoted by x^+, such that $K'(x) = l(x^+)$. From here, a compression/prediction test based on Chaitin's proposal (Chaitin 1982) seems to be easily applicable. However, there are many technical reasons that explain that such an intriguing proposal has not been addressed yet[2]:

1. $K(x)$ is not computable, so x^+ cannot be effectively computed. If a compression test is constructed, how do we know whether the subject's answer is a hit?
2. There are different equally alternative plausible descriptions: x^+ is just the first one in lexicographic order of all the shortest descriptions.

[2] At least to the author's knowledge and as Chaitin himself has recognised (Chaitin 1998, personal communication).

3. Despite the invariance theorem that states that x^+ depends on ϕ only up to a constant, this constant is relevant if $l(x)$ is small, and there is no reason to prefer one descriptional system over another.
4. The test intends to measure the ability of compression, but this does not match exactly[3] with the ability of comprehension, i.e., intelligence.

The first problem can be solved by incorporating time into the definition of K. The most appropriate way[4] to weigh space and time of a program, the formula $LT_\beta(p_x) = l(p_x) + log\, Cost_\beta(p_x)$, was introduced by Levin in the seventies (see e.g. Levin 1973). Then the next variant comes directly:

Definition 4. The **Levin's Length-Time Complexity** of an object x given y on a descriptional mechanism β:

$$Kt_\beta(x|y) = min\{LT_\beta(p|y) : \phi_\beta(\langle p, y\rangle) = x)\}$$

where $LT_\beta(p|y) = l(p) + log\, Cost_\beta(\langle p, y\rangle)$

This is a very practical alternative of Kolmogorov Complexity, because as well as avoiding intractable descriptions, it is computable. Moreover, it accounts better for the idea of simplicity, and Occam's razor should be better formalised under this variant.

To extend LT-Complexity to projectible descriptions, we must measure $Cost_\beta(p)$ in an asymptotical way. Consider a machine ϕ such that the output tape cannot be rectified. $Cost_\beta(p)[..n]$ is defined as the time or machine steps such that the first n symbols of the definite output are placed at the beginning of the output tape and $Cost_\beta(p)[n..m] = Cost_\beta(p)[..m] - Cost_\beta(p)[..n]$. From here we only need $LT_\beta(p_x)[n..m] = l(p_x) + log\, Cost_\beta(p_x)[n..m]$ and $LT_\beta(p_x)[..n] = l(p_x) + log\, Cost_\beta(p_x)[..n]$ for the following definition:

Definition 5. The **k-Projectible Length-Time Complexity** of an object x given y on a descriptional mechanism β is defined as: $Kt'_\beta(x|y) = min\{LT_\beta(x|y)[..l(x)] : \exists w \in \Sigma^*\ l(w) = k \text{ such that } \phi_\beta(\langle p, y\rangle) = xw)\}$

This definition will serve as a start point to face the other three unsolved problems (2,3,4). In fact, they require first to distinguish what is to comprehend (problem 4), which is addressed in the following section, and later on to solve how to measure it (problems 2 and 3).

3. Formalising Comprehension

To comprehend is to understand either the inner mechanism or the plausible model of some evidence. In some way, comprehension is stricter than learning in terms of justification, because comprehension usually entails

[3] "*I just see how Kolmogorov Complexity and Intelligence could be well related but I don't think it would be 'exactly' so.*" (Hofstadter 1997, personal communication).

[4] Intuitively, every algorithm must invest some effort either in time or demanding/essaying new information, in a relation which approximates the function LT.

that the subject is able to explain the concept to others. In the case of infinite concepts, this explanation is only possible if the subject has a finite description of the concept. Consequently, comprehension can be understood in terms of identification. However, if a concept is finite, like most *everyday* concepts, both notions diverge significantly. A finite concept can be easily identified by the extensional description of the concept, which has no insight and surely has not identified any mechanism or pattern from it, if the evidence ever had one. This question is old in logic, where comprehension means the connotation of a term, opposed to its denotation or extension. Hence, an *extensional* description (by enumeration) has no connotation and consequently entails no comprehension at all. On the contrary, an *intensional* description (by comprehension) may have not discovered the right meaning or *real* mechanism of the evidence, but still has a chance of discovering it.

There is a fundamental feature that determines this difference: "*The Defined Thing CANNOT Appear in the Definition*", which is known as **Comprehension Requirement**. It is also one of the four laws of definition, according to methodology (Bochenski 1965). It is frequently used as a criterion by teachers when asking their pupils whether they have comprehended a concept. In fact, the pioneer of the psychometric approach, Binet, designed his first tests to avoid this "rote learning".

Traditional use in mathematics also distinguishes informally an extensional definition from an intensional definition (or by comprehension). For infinite sets, frequent in mathematics, every definition must be intensional (or by comprehension). Nonetheless, for finite sets there is still no formal difference between an intensional description and an extensional one.

At first sight, Kolmogorov or Descriptional Complexity seems sufficient to distinguish extensional descriptions from intensional ones. However, the MDL principle, which chooses the shortest description for a given concept x, does not ensure that the description is intensional. In the vast majority of cases, the data is not compressible, and the MDL principle will give the data itself, being the most extensional description, which does not give any hint about the comprehension of that data. Even in the rare cases where the data is compressible, a short description does not ensure that all the data is described intensionally; there could be a part that could be highly compressed and another part that could be quoted as an exception.

The question is then more conspicuous: is there any way to distinguish *pattern* from exceptions, program from data?

Koppel introduced the notion of sophistication with the goal of distinguishing the structural part of an object (Koppel 1988) from its data (or non-compressible part of it). Logical Depth, as defined by Bennett (Bennett 1988), is shown to be equivalent to sophistication up to a constant (Koppel 1987). Both, however, can 'disguise' a general effective interpreter as fictitious pattern and leave a great amount of real pattern as data.

It is then required a different approach to distinguish whether a description has exceptions (partially or totally extensional) or is composed exclusively of pattern (it is all structure or totally intensional). The idea is to compare the part which is used for all the data *to the limit* (the structure), with the part which is only used in some portion of the data (the exception).

Definition 6. A description p' is (n, k)-**equivalent in the limit** to a description p iff $\exists n \in \mathbb{N}, n > 0$ and $\exists k \in \mathbb{Z}$ such that $\phi(p')_{n+k..} = \phi(p)_{n..}$

Informally, two descriptions are equivalent in the limit if there is a point from which their predictions always match. If both descriptions are k-projectible with k finite they are always equivalent in the limit. If only one of both is ∞-projectible then they cannot be equivalent in the limit. Hence, the definition applies when both descriptions are ∞-projectible descriptions.

Definition 7. A description p is a **Fully Projectible Description** of x given y iff $\langle p, y \rangle$ is an ∞-projectible description of x and $\neg\exists p'$ s.t. $\langle p', y \rangle \not\equiv \langle p, y \rangle$ but (n, k)-equivalent in the limit to $\langle p, y \rangle$ and $LT(p'|y)[n + k..n + k + l(x)] < LT(p|y)[n..n + l(x)]$.

The first condition that p' is not extensionally equivalent to p ($p' \not\equiv p$) is to avoid that given two or more equivalent descriptions, only the shortest one would be projectible. The second condition measures that this p' is simpler than p. Note that LT (and only applied to the first chunk of length $l(x)$ where p' and p begin to be equivalent) is used instead of l.

Example 1. Given the evidence "3, 12, 21, 30, 102, 111, 120", we can consider several projectible descriptions. For instance, D_1 = "3, 12, 21, 30, 102, 111, 120 and 1 forever" is not fully projectible because there exists a shorter description "1 forever" which is equivalent in the limit. In the same way, D_2 = "Start with number 3. The following three numbers are obtained by adding 9 to the preceding one. Continue with number 102. The following numbers are obtained by adding 9 to the preceding one" is not fully projectible because there exists a shorter description "Start with number 3. The following numbers are obtained by adding 9 to the preceding one" which is equivalent in the limit. On the contrary, the description D_3 = "numbers whose digits in decimal representation amounts to 3 ordered" is fully projectible. Similarly, the description D_4 = "repeat 3, 12, 21, 30, 102, 111, 120 for ever" is fully projectible. Finally, the following description is also fully projectible D_5 = "the y values of a polynomial $y = P(x)$" where P is a polynomial such that $P(1) = 3$, $P(2) = 12$, ..., $P(7) = 120$.

D_4 and D_5 may seem counterintuitive but it should be realised that a fully projectible description just formalises the idea of explanation (and not yet the comprehension requirement): it describes the evidence, it accounts for all of it (there are no exceptions because it is fully projectible) and it can be related (explained) to others (because of the use of LT, descriptions which are extremely time consuming are avoided). Hence, D_4, whether we like it or not, is an *explanation* for the evidence.

For the moment, we can define a new variant of descriptional complexity:

Definition 8. The **Explanatory Complexity** of an object x given y on a descriptional mechanism β is defined as:

$$Et_\beta(x|y) = min\{LT_\beta(p|y)[..l(x)] \text{ s.t. } \langle p, y\rangle \text{ is fully projectible } \}$$

The string y, which we have supposed empty in the previous example, represents the context or previous knowledge where the explanation must be applied. In the same way it is done with K and the MDL principle, we can denote with $SED(x|y)$ the Shortest (in LT terms) Explanatory Description for x given y, i.e. the first shorstest fully projectible (in lexicographic order) description for x given y. Logically, $l(SED(x|y)) = Et(x|y)$.

However, we still have that for most strings, $SED(x)$ will be just the *rote* description "repeat x forever" which does not follow the comprehension requirement. A first idea to avoid this phenomenon is to force the description to be shorter than the data and to say that the data has no comprehensive explanation if this is not the case[5]. However, most of everyday data is not compressible and it is still comprehended.

Another approach is the idea of reinforcement or cross-validation (Hernández-Orallo 1999a). For instance, if we remove the last element of the previous series, i.e. "3, 12, 21, 30, 102, 111", it is not much expectable that D'_4 and D'_5 would be produced but D'_3 can still be generated. In general,

Definition 9. **Stability**. A string x is *m-stable on the right* in the descriptional system β iff $\forall d, 1 \leq d \leq m : SED_\beta(x_{-d}) \equiv SED_\beta(x)$.

In other words, a string x is *m-stable on the right* if taking m elements from the right, it still has the *same* best explanation. These m elements, if given a posteriori, are considered reinforcement or confirmation, and, if given a priori, are considered redundancy or hints to help to find the explanation.

Consequently, although rote learning can be trickily used to make an extensional description fully projectible, stability (like reinforcement or cross-validation) is a methodological criterion to avoid this phenomenon. This finally gives sufficient characterisation to state that a description entails that the learner who has generated it has comprehended the data.

There is still another reason to support the previous notion of comprehension as an ontological principle. Why must we avoid rote learning? Why must we anticipate? Why do children find more complex patterns? (Marcus et al. 1999) Why are we genetically programmed to open any black box we are presented? This search for more informative hypotheses instead of the easiest ones may lead to fantasy, but this is not dangerous provided that the system can interact with the world in order to refute some of them.

This informativeness or investment in the hypotheses was advocated by Popper for the scientific method, and as we have seen, it is equally applicable

[5] A different approach is the notion of exception, studied and formalised in (Hernández-Orallo and Minaya-Collado 1998) and (Hernández-Orallo and García-Varea 1998).

for cognition. Even if we make the very strong assumption of Occam's razor, i.e., things in nature are not complex unnecessarily, the previous rationale is justified by the fact that, just as every incompressible string has compressible substrings, *most* compressible strings have incompressible substrings, because the shorter the less worthy that is to compress. If the evidence is presented incrementally, it is better to invest in more informative or general hypotheses instead of finding the optimal one for each chunk, which will finally turn out not to be part of the whole description of the whole evidence. This rationale leads to the next theorem:

Theorem 2. For every descriptional mechanism β, there exists a constant c which depends exclusively on β such that for every string x of length n with $SED(x) = s$ and $l(s) = m$ s.t. $m < n$, and any partition $x = yz$, $l(y) < m - c$, then $SED(y)$ is not equivalent in the limit with s.

Proof. Consider any string x and $SED(x) = s$ with $l(s) = m$ s.t. $m < n$. Take any prefix y such that $l(y) < m - c$. It has a fully projectible description $p_y =$ "print y for ever" with $l(p_y) = l(y) + c' < m - c + c'$, this constant c' being the space which is required for coding "print .. for ever". Since the computational cost of p_y is linear, say $k' \cdot l(x)$, it is sufficient to choose $c \geq c' + log\, k'$ to ensure that the description p_y is shorter than s, and $LT_\beta(p_y)[..l(x)] < LT_\beta(s)[..l(x)]$ because $log\, k' \cdot l(x) = log\, k' + log\, l(x)$. Moreover, p_y and s cannot be equivalent in the limit because s is fully projectible and, by definition, there does not exist a description with less LT equivalent in the limit. □

It is clear that the idea of stability or cross-validation is supported by the previous theorem. In fact, it is an innate *aesthetic* preference in the explanations that human beings generate. Why is it more pleasant the answer 23 to the series "3,7,11,15,19, ..." than the answer 3? In Hofstadter words, "*it would be nice if we could define intelligence in some other way than "that which gets the same meaning out of a sequence of symbols as we do"*" (Hofstadter 1979). Theorem 2 simply states why we do in that way.

As a result of this section, stable objects give SED descriptions where comprehension has taken place, i.e., comprehensive descriptions. The following section is devoted to ensure that the descriptions would give the same meaning out of a sequence, and how to measure their complexity.

4. Testing Comprehesion Ability

Theoretically, there are two ways to know whether a system's operation is compliant with some requirements: by inspecting its code (or program) or by testing its behaviour. In general, for complex systems, as it has been finally recognised in software engineering, verification must be experimental, by means of sets of tests. It is an open and hard problem to devise a *complete* specification of intelligence, mainly because it depends on a consensus on the *abilities* that an intelligent system must have. However, it is possible to distinguish some abilities that are fundamental for intelligence. A verification of intelligence behaviour should begin with these fundamental traits, and gradually add more diverse test cases in order to make the test set more robust. Comprehending is the most important trait of intelligence, and we have formalised it in a computational framework. This allows the construction of exercises for a test which are not selected experimentally but theoretically, so, finally, we know what is to be measured, quite unlike psychometrics.

However, if we intend to measure comprehensibility there are still two questions to solve. First, we must design unquestionable exercises, in order to avoid the 'subjectivity objection' of IQ tests. Secondly, we require an absolute referent of comprehension difficulty in order to give a non-Boolean score independent to the mean ability of the subjects or society who have made the test before.

Psychometrics has striven to show that it is not absurd to talk about the 'correct' solution. Its rationale is that if the great majority matches with some solution is *because there are not alternative solutions of similar complexity*, and, consequently, it is the most plausible. However, this assertion is made from a very subjective and informal point of view.

Let us make formal and objective this idea. At first sight it seems that stability avoids this but, if we restrict to stable descriptions, we can still modify any explanation p with the addendum "Execute p but print a '1' every hundred symbols that are printed" which would be comprehensive for the data but would differ from p in the limit, and would be only a little longer. For this reason, we must introduce the notion of plausibility:

Definition 10. **Plausibility**. A fully projectible description p for a string x is (c, m)-*plausible on the right* in the descriptional system β iff $\forall d, 0 \leq d \leq m : LT_\beta(SED_\beta(x_{-d}))[..l(x_{-d})] + c > LT_\beta(p)[..l(x_{-d})]$.

Intuitively, a description is (c, m)-plausible if it is at most c bits longer (in LT terms) then the best explanation for x and this holds even if we remove up to m elements from the right of x. From here, we can face unquestionability in the following way:

Definition 11. **Unquestionability**. A fully projectible description p for x is (c, m)-*unquestionable* in the descriptional system β iff it is (c, m)-*plausible* and there does not exist another (c, m)-*plausible* description p' for x.

This is a more restrictive condition as c and m are greater. In order to still obtain some unquestionable descriptions we must make the strings larger. However, as we will see later, if c and m are tuned conveniently for a concrete descriptional mechanism, the tests can still be composed of short strings x such that their $SED_\beta(x)$ is (c, m)-unquestionable.[6]

[6] This restriction to unquestionable descriptions not only preserves the goal of the test but even strengthens it, in ontological terms. It has been frequently argued in philosophy of

Once we are able to obtain strings whose SED_β is (c, m)-unquestionable, we should ascertain the difficulty of each problem, in order to be able to give a test set of exercises of different comprehensibility. The idea is to relate this difficulty with explanatory complexity (Et) and the explicitness of the description wrt. the data:

Definition 12. A string x is k-**incomprehensible** given y, denoted by $incomp(x|y)$, in a descriptional system β iff k is the least positive integer number such that: $Et_\beta(x|y) \cdot G(SED(x|y)|\langle x, y\rangle) \leq k \cdot log\, l(x)$.

The use of the factor $logl(x)$ is to compensate the fact that x must be printed and, therefore, for all x $Et(x) \geq log\ l(x)$. Consequently, for all $x, k \geq 1$. As an example, consider a string x of length 256, with $Et(x) = 50$. The comprehensibility of x is $k = 7$.

The function G corrects the degree of explicitness of the description wrt. the data and it is defined as follows (Hernández-Orallo 1999b):

Definition 13. The **information gain** of an object x wrt. an object y in a descriptional system β is given by: $G_\beta(x|y) = Kt_\beta(x|y)/Kt_\beta(x)$.

Definition 12 finally measures the real difficulty of finding $SED(x)$ from x, because descriptions of the form "repeat x for ever" which have Et high (to quote x) are corrected by G, but the length of x is still important.

Now we are prepared to construct a generic test of the ability of comprehension by generating a series of strings of gradual comprehensibility. However, as we have said, it is important that the answer is unquestionable, because if not, the answer would be an arbitrary choice from the examiner. A way is to provide redundant information to make the answer unquestionable, with some limitations, obviously, because if not, the problems would be much too long. For instance, given the series "a, c, c, a, c, c, c, a, c, c, c, c, a, ..." it seems logical to expect that it would follow "c, c, c, c, c, a, c, ...", so it is redundant to present more than the necessary symbols.

The measurement that is to be presented below requires the collaboration of the subject, which must employ all its resources to perform the test. It is not necessary that the subject understands the aim of the test but at least it should be programmed to do it.

We can finally obtain the degree of intelligence of a given system as the value which results from applying the following test:

Definition 14. **C-Test**. Let us select a descriptional system β sufficiently expressive and impartial, composed of an alphabet of symbols Ω_β and a set of

science and induction that the plausibility and unquestionability of a theory or explanation not only depends on the intrinsic characteristics of the explanation but also on the ability of finding alternative explanations. In this sense we can see intelligence as the most important means to augment the plausibility and confidence of explanations, and, consequently, the ontology of an 'intelligent' system.

operations Θ_β to manipulate these symbols, and their corresponding *cost* (or length). We provide (or programme) to S the alphabet, operations and cost. Depending on the expected intelligence of a system we select a sufficiently wide range $1..K$ of difficulty. For each $k = 1..K$ we choose randomly p sequences $x^{k,p}$, being *k-incomprehensible*, *c-plausible*, *(c,m)-unquestionable* and *d-stable* with $d \geq r$, r being the number of redundant symbols (or hints) of each exercise. The questions are the $K \cdot p$ sequences without their $d - r$ elements ($x^{k,p}_{-(d+r)}$). We give them to S and we ask for the following element according to the best explanation that is able to construct with Ω_β and Θ_β. We leave S a fixed time t and we record its answers: $guess(S, x^k_{-d+r+1})$. The result of this test of comprehensibility or (C-test) is measured as:

$$I(S) = \sum_{k=1..K} k^e \cdot \sum_{i=1..p} hit[x^{k,j}_{-d+r+1}, guess(S, x^{k,j}_{-d+r+1})]$$

the function *hit* is usually measured as $hit(a,b) = 1$ if $a = b$ and 0 otherwise (negative values can be used to penalise errors). The value e is simply for weighting the difficult questions ($e = 0$ means that all have the same value).

In an informal way, "*the test measures the ability of finding the best explanation for sequences of increasing comprehensibility in a fixed time*".[7]

5. Measurement of Pretended Intelligent Systems

The preceding test is applicable to any system whose degree of intelligence is questioned. Selecting appropriately the descriptional system and the rest of parameters of the test, it can be used for humans, animals, computers, extraterrestrial beings and any collection of the preceding working jointly.

Although Definition 14 evaluates a single ability, there are still many ways to realise a specific test. In (Hernández-Orallo and Minaya-Collado 1998) the test was implemented by using an abstract machine quite similar to a state machine. From here, a variety of strings of different *comprehensibility* in that machine were generated. Although the set of k-potent numbers of length at most n can be computed in polynomial time in n (see a proof in (Li and Vitányi 1997)), the cost of $O(n^k)$, forces to use some heuristics for this. In the same way, G was approximated. Finally, a sieve was applied for obtaining only *c-plausible*, *(c,m)-unquestionable* and *d-stable* sequences.

The same work presents the results of applying the test to 65 subjects from species Homo Sapiens Sapiens aged between 14 and 32 years, jointly

[7] One relevant feature of the test is that, although the subject is supposed to be a particular universal descriptional system ϕ_s with a particular background knowledge (life experience) B_s, it is given a descriptional system β over it, which highly minimises the influence of the difference between the computations performed by ϕ_s and other subject ϕ_t, i.e. the difference between $Et_s(x|\langle B_s, \beta\rangle)$ and $Et_t(x|\langle B_t, \beta\rangle)$. This makes it possible for the notions of plausibility and unquestionability to be similar for both subjects.

with a classic test of intelligence, the *European IQ Test*. The correlation between both tests was 0.77. This value only justifies a further more exhaustive study over greater groups and several variations derived from Definition 14. Another remarkable experimental result shown in Fig. 1 is that the relation between hit ratio and k-uncomprehensibility is straight, which suggests that comprehensibility really estimates the difficulty of each string.

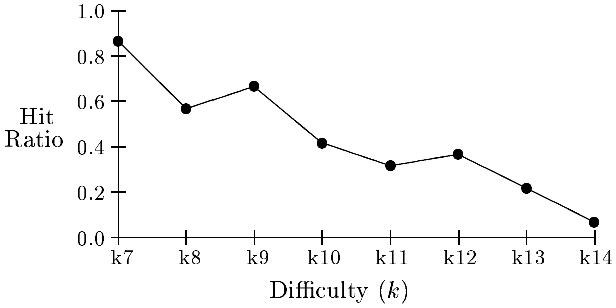

Figure 1

Logically, it is not expectable (for the moment) that contrasted and widely used IQ tests would be substituted by these C-tests. Nonetheless, this could be a startpoint towards a theoretical foundation of psychometrics *free* from the Homo Sapiens as a reference.

However, it is not human intelligence but non-human what is urgent to be measured. A formal declaration of what is expected from an intelligent system should allow two important things: to derive more intelligent systems from a more concrete specification and, secondly, to evaluate them. Definition 14 provides a first step for both things, a detailed scale for measuring the progress (in one intelligence factor) of generic systems in AI. As any other field of science, a great advance in a discipline happens when one of its topics can be measured in an effective and justified way. AI, as a science, requires measurements of intelligence and its different factors.

Modern AI systems are much more functional than systems from the sixties or the seventies. They solve problems in an automated way that before required human intervention. However, these complex problems are solved because a methodical solution is found by the system's designers, not because current systems are more intelligent than preceding ones. We share the opinion that "*it is time to begin to distinguish between general, intelligent programs and the special performance systems*" (Nilsson 1995).

This initial aim of being more general is nowadays still represented by two subfields of AI: automated reasoning and machine learning. Automated theorem provers are able to solve complex problems from different fields of mathematics. The great advance of the last two decades is mainly caused by the existence of sets of problems to compare different systems. Even these sets have evolved and grown to huge and complete libraries of theorem proving problems, like TPTP (Suttner and Sutcliffe 1996). Machine learning is also taking a more experimental character and different systems (from different paradigms) are evaluated according to classical problems in the literature. However, as we said in the introduction, there is no theoretical (nor empirical) measurement about the complexity of the problems which compose these test sets. This complexity could be obtained by adapting comprehensibility to several representational languages.

6. Factorisation

During this century, psychometrics have striven for differentiating between background knowledge (either evolutionary-acquired or life-acquired) and 'liquid intelligence' (or individual adaptability). Accordingly, exercises from IQ tests are strictly selected to avoid the influence of background knowledge to be foolproof to 'idiots savants'. However, there are many knowledge-independent abilities (or factors) to measure. Some factors usually found in psychological tests are 'verbal ability', 'visual ability', 'calculation / deductive ability', etc.

The C-test measures one factor, which could empirically be identified with the g factor or liquid intelligence. There are more independent factors which could be measured by using extensions of the framework presented in the previous section. For instance, knowledge applicability, contextualisation and knowledge construction ability can be measured in the following way:

- Knowledge Applicability (or 'crystallized intelligence'): we provide a background knowledge B and we give a set of sequences x_i such that $incomp(x_i|B) = incomp(x_i) - u$ but still $SED(x_i|B) = SED(x_i)$ and are unquestionable with or without B. We can compare the difference of performance between cases with B and without B. This test would actually measure the application of the background knowledge depending on two parameters: the complexity of B (i.e. $Kt(B)$) and the necessity or usefulness of B, measured by u.
- Contextualisation: it is measured in a similar way as knowledge applicability but providing different contexts $B_1, B_2, ..., B_T$ with different sequences $x_{i,t}$ such that $incomp(x_{i,t}|B_t) = incomp(x_{i,t}) - u$. This multiplicity of background knowledge (a new parameter T) differentiates this factor from the previous one. Analogy tests generally resemble this type of exercises, as it was shown in (Hernández-Orallo and Minaya-Collado 1998) where a definition (and measurement) of analogy was discussed.
- Knowledge Construction: we provide a set of sequences x_i such that exists a common knowledge or context B and a constant u such that for $incomp(x_i|B) \leq incomp(x_i) - u$. A significant increase of performance must take place between the first sequence and the later sequences. The parameters are the same as the first case, the complexity of B and the constant u. This learning from precedents has also been studied in AI (see e.g. Winston 1982).

Knowledge Applicability may also be correlated with deductive abilities and these may also correlate with the idea of congruence or coherence, since it can be measured as constraint satisfaction (Thagard 1989). Other factors are more related with *intentionality* than *intensionality* and general intelligence. These are reactivity, pro-activity, interactivity and the recently elsewhere vindicated emotional abilities, that can be measured adopting notions from Query Learning paradigms (Angluin 1988), possibly formalised using interactive Turing machines.

However, not every factor is meaningful for intelligence. Factors like "playing chess well" are much too specific to be robust to background knowledge. Other factors will result in being highly correlated (experimentally or *theoretically*) to other more distinct factors. The influence of the descriptional mechanism should also be studied for each factor.

In the end, the matter at issue is to refine and extend all the previous ideas in order to make different and founded tests of intelligence, knowing exactly what is measured. This is an urgent and fascinating task for AI.

7. The C-test and The Turing Test

The imitation game was conceived by Turing to dissipate the doubts about possibly non-human intelligent beings. He left no place for human's exclusivity: intelligence can be evaluated by an exclusively behavioural test. Unfortunately, instead of recognising this his most important contribution, the test is still understood as 'a goal' in AI. Nonetheless, this view has been responded by many authors, which criticise that the TT does provide little information about what intelligence is; it is just a test of humanity (Fostel 1993), that, in fact, if applied to human beings, yields many paradoxes. The result of applying it to ourselves is a recursive trap which is unable to answer the question of how intelligent the Homo Sapiens is.

There have been unsuccessful attempts to correct the two main problems of the Turing Test for measuring intelligence: its informal character and its anthropocentrism.[8] There is still a third problem, which is the necessity of several intelligent 'judges' and a 'referent' to make the test. The self-reference question arises again: Who is the first intelligent being to start the game? These and other problems are incarnated in the Loebner Prize, which usually awards the participant who has devised the system more able to cheat the judge, because "*humans are surprisingly bad at distinguishing humans from computers*" (Johnson 1992). Furthermore, there is no way of knowing who is cheating, the system or its designer.

However, if fairly played, the imitation game is a hard examination for any pretended intelligent system. It is extremely difficult to behave like an

[8] In some cases, this has led to disparate proposals, as the so-called formalisation of the TT (Bradford and Wollowski 1995), sustained from the assumption that we are able to solve NP-complete problems in polynomial time. As (McCarthy 1998) clarifies: "*humans often solve problems in NP-complete domains in times much shorter than is guaranteed by the general algorithms, but can't solve them quickly in general*".

average human being of this epoch (it is even difficult for some average human beings). For a non-human-contextualised being, it would be required to comprehend the complex behaviour of human beings of these times, their evolution-acquired traits, their language, their culture, their limitations, etc. It is much easier then to try to cheat the judges. In fact, the judges "*are especially fooled into reading structure into chaos, reading meaning into nonsense. (...) Sensitivity to subtle patterns in our environment is extremely important to our ability to perceive, learn and communicate*" (Shieber 1994).

Curiously, it is precisely this 'lack' of the judges what the C-tests measure. However, the C-tests, as they have been presented, are necessary (at least to obtain a minimum value of $I(S)$) but not sufficient (other important factors should be measured as well). It has been already suggested that both kind of tests (TT and factorial) could be combined in order to give a more accurate test of intelligence: "*it is this posing of puzzles in arbitrary domains that is the hardest part of the Turing Test, and a part that no program has yet passed*" (Shapiro 1992). This idea, however, would ultimately turn the TT into a lightweight and less rigorous version of a factorial C-Test.

In our opinion, the TT should be celebrated as an extremely valuable philosophical exercise about the behavioural character of intelligence. However, in practice, it should be substituted by progressively more accurate computational tests of different cognitive abilities.

8. Conclusions

Turing devised a way for identifying intelligent beings from non-intelligent ones without solving the problem of what intelligence is. In fact, an imitation game is the only way to make sense from such an apparent paradox. However, the approach has shown numerous limitations and troubles which make it useless for AI in practice. With current theoretical and technical tools of computer science, it is difficult to develop non-human intelligence without a computational formalisation of the problem we are trying to solve.

It is high time to address the fundamental problem: what intelligence is. This paper presents an important step in this line. A formalisation of one of the main factors of intelligence, the g factor or liquid intelligence is defined computationally. This definition has been used to develop an intelligence test, very different from the TT and in compliance with classical IQ tests. Like the latter it distinguishes acquired knowledge from liquid intelligence. More importantly, the C-Test, unlike the TT and IQ tests, is not anthropomorphic. The factor is defined as the ability of finding comprehensive explanations, and thus is meaningful. This makes it philosophically acceptable: intelligence is what allows us to comprehend the world.

Sooner or later we will need to face the fact that computers will be closer and closer to human intelligence. Once arrived to this hallmark of

AI, *the misunderstood goal of the TT*, it will be indispensable to have an objective measure of intelligence, in order to solve the incipient technical and ethical problems that could be derived from here. The paradigm presented in this paper, since it is theoretically justified, allows the projection of the measurement of intelligence beyond human intelligence.

There are many more interesting questions beyond the TT. How many independent computational factors does human intelligence have? How intelligent does the Homo Sapiens result in the end? Which main factors make a chimpanzee significantly different from us? How intelligent might machines be with the current computational power? Psychometrics, Anthropology, Zoology and AI have only partially dealt with some of these problems. All these questions require new theoretical tools for a radical change of paradigm: a science of intelligence grounded in theoretical computer science and information theory.

To construct this science, Turing's call is now more compelling then ever: "*We can only see a short distance ahead, but we can see plenty there that needs to be done*".

Acknowledgements

I am much obliged to the main inspirers of this work, G. Chaitin and D. Hofstadter, for their encouraging comments, especially during 1997, when the main ideas were taking shape. Since then, the work has matured with the help of numerous collaborations and suggestions: R. Beneyto and J.M. Lorente from the Dep. of Logic and Phil. of Science of the U. of Valencia, P. Thagard from the Dep. of Phil. of the U. of Waterloo, O. Pellicer from the Dep. of Psychobiology of the U. of Valencia and N.T. Crook from the Oxford Brookes U. Finally, I am especially grateful for the comments from K. Araque, R. Barreiro, E. Fueyo, I. García, E. Hernández, N. Minaya and I. Soto on several drafts of this paper.

References

D. Angluin. Queries and concept learning. *Machine Learning*, 2(4):319–342, 1988.

Editorial. Artificial Stupidity. *The Economist*, 324, no. 7770, August 1, p.14, 1992.

C. H. Bennett. Logical depth and physical complexity. In (Herken 1988), pp. 227-258.

J. M. Bochenski. *The Methods of Contemporary Thought*. Dordrecht, D. Reidel 1965.

P. G. Bradford and M. Wollowski. A Formalization of the Turing Test (The Turing Test as an Interactive Proof System). *SIGART Bulletin*, 6(4), p. 10, 1995.

G. J. Chaitin. Gödel's Theorem and Information. *Int. J. Theo. Phys.*, 21, 941-54, 1982.

B. Chandrasekaran. What kind of Information Processing is Intelligence?. In D. Partridge and Y. Wilks (eds.) *Foundations of AI: A Source Book*. Cambridge U.P., 1990.

E. Dietrich. Programs in the Search for Intelligent Machines: The Mistaken Foundations of AI. In D. Partridge and Y. Wilks (eds.) *Foundations of AI: A Source Book*. Cambridge Univ. Press, 1990.

R. Epstein. Can Machines Think?. *AI Magazine*, 12(2), 80-95, 1992.

T. G. Evans. A Heuristic Program to Solve Geometric Analogy Problems. In. M. Minsky (ed.) *Semantic Information Processing*, MIT Press, 1968, PhD thesis, MIT, 1963.

H. J. Eysenck. *The Structure and Measurement of Intelligence*. Springer-Verlag 1979.

G. Fostel. The Turing Test is For the Birds. *SIGART Bulletin*, 4(1), 7-8, 1993.
E. M. Gold. Language Identification in the Limit. *Inform and Control*, 10, 447-474, 1967.
G. Harman. The inference to the best explanation. *Philos. Review*,74: 88-95, 1965.
S. Harnad. The Turing Test Is Not a Trick: Turing Indistinguishability Is A Scientific Criterion. *SIGART Bulletin*, 3(4), 9-10, 1992.
R. Herken. *The universal Turing machine: a half-century survey.* Oxford University Press, 1988, 2nd Edition 1994.
J. Hernández-Orallo. Constructive Reinforcement Learning. *Intl. Journal of Intelligent Systems*, to appear.
J. Hernández-Orallo. Unified Information Gain Measures for Inference Processes. *Collegium Logicum - Annals of the Kurt-Gödel-Society* V.4, Springer, in press 1999.
J. Hernández-Orallo and I. García-Varea. Explanatory and Creative Alternatives to the MDL principle. In S. Rini, G. Poletti (eds.) *Proc. of Intl. Conf. on Model Based Reasoning* (MBR'98), Pavia 1998. Also to appear in *Foundations of Science*.
J. Hernández-Orallo and N. Minaya-Collado. A Formal Definition of Intelligence Based on an Intensional Variant of Algorithmic Complexity In *Proc. of the Intl. Symp. of Engin. of Intelligent Systems* (EIS'98), ICSC Press, pp. 146-163, 1998.
D. R. Hofstadter. *Gödel, Escher, Bach.* Basic Books, 1979.
W. L. Johnson. Needed: A New Test of Intelligence. *SIGART Bulletin*, 3(4), 7-9 Ed., 1992.
A. N. Kolmogorov Three Approaches to the Quantitative Definition of Information. *Problems Inform. Transmission*, 1(1):1-7, 1965.
M. Koppel Complexity, Depth, and Sophistication. *Complex Systems*, 1, 1087-1091, 1987.
M. Koppel Structure. In (Herken 1988), pp. 435-452.
J. E. Larsson. The Turing Test Misunderstood. *SIGART Bulletin*, 4(4), p. 10, 1993.
L. A. Levin. Universal search problems. *Problems Inform. Transmission*, 9:265-266, 1973.
M. Li and P. Vitányi *An Introduction to Kolmogorov Complexity and its Applications.* 2nd Ed. Springer-Verlag 1997.
G. F. Marcus, S. Vijayan, S. Bandi Rao and P. M. Vishton Rule Learning by Seven-Month-Old Infants. *Science*, pp. 77-80, January 1998.
J. McCarthy What's is AI. `http://www-formal.stanford.edu/jmc/whatisai.html`, 1998.
P. J. R. Millican and A. Clark (eds.) *Machines and Thought. The Legacy of Alan Turing, Vol. I.* Clarendon Press, Oxford, 1996.
U. Neisser, G. Boodoo, T. J. Bouchard, A. W. Boykin, N. Brody, S. J. Ceci, D. F. Halpem, J. C. Lochlin, R. Perloff, R. J. Sternberg, S. Urbina. Intelligence: Knowns and Unknowns. *American Psychologist*, 51, 77-101, 1996.
N. J. Nilsson. Eye on the Prize. *AI Magazine*, July 1995.
B. Preston. AI, anthropocentrism, and the evolution of "intelligence". Minds and Machines 1, 259-277, 1991.
J. Rissanen. Fisher information and stochastic complexity. IEEE Trans. IT, 42(1), 1996.
R. C. Schank. What is AI, Anyway?. *AI Magazine*, 8, 59-65, 1987 and in R.J. Sternberg and D.K. Detterman, What is Intelligence? contemporary viewpoints on its nature and definition, Norwood, NJ. : Ablex, 1986.
S. C. Shapiro. The Turing Test and The Economist. *SIGART Bulletin*, 3(4), 10-11, 1992.
S. M. Shieber. Lessons from a Restricted Turing Test. *Comm. of the ACM*, 37(6), 1994.
H. Simon and K. Kotovsky. Human acquisition of concepts for sequential patterns. *Psych. Review* 70, 534-46, 1963.
R. J. Solomonoff. A formal theory of inductive inference. *Inf. Control*, 7, 1-22, March, 224-254, June 1964.
C. Spearman. 'General Intelligence' objectively determined and measured. *Amer. J. of Psych.*, 15, 201-293, 1904.

R. J. Sternberg. *Intelligence, Information Processing, and Analogical Reasoning.* John Wiley & Sons 1977.

T. Stonier. *Beyond Information. The Natural History of Intelligence.* Springer 1992.

C. B. Suttner and G. Sutchliffe. *The TPTP Problem Library.* Tech. Univ. Munich, 1996

P. Thagard. Explanatory coherence. *Behavioural and Brain Sciences*, 12(3), 435-502, 1989.

A. M. Turing. On computable numbers with an application to the Entscheidungsproblem. *Proc. London Math. Soc.*, series 2, 42:230-65, 1936. Correction, Ibid, 43:544-6, 1937.

A. M. Turing. Computing Machinery and Intelligence. *Mind*, 59, 433-460, 1950.

L. Valiant. A theory of the learnable. *Comm. of the ACM*, 27(11), 1134-1142, 1984.

S. Watanabe. Pattern Recognition as Information Compression. In S. Watanabe (ed.) *Frontiers of Pattern Recognition*, New York: Academic Press, 1972.

T. Winograd. Thinking Machines: Can There Be? Are We?. In T. Winograd and F. Flores (eds.) *Understanding Computers and Cognition*, Norwood, 1986.

P. H. Winston. Learning New Principles from Precedents and Exercises. *Artificial Intelligence*, 19(3), 1982.

E. Zigler and V. Seitz. Thinking Machines: Can There Be? Are We?. In B. B. Wolman, *Handbook of Human Intelligence Handbook of intelligence : theories, measurements, and applications.* New York, Wiley 1982.