# Beyond the Turing Test

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

February, 2000

**Abstract.** The main factor of intelligence is defined 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, which is exclusively defined in computational terms. Due to its absolute and non-anthropomorphic character, it is equally applicable to both humans and non-humans. Moreover, it correlates with classical psychometric tests, thus establishing the first firm connection between information theoretical notions and traditional IQ tests. The Turing Test is compared with the C-test and the combination of the two is questioned. In consequence, the idea of using the Turing Test as a practical test of intelligence should be surpassed, 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 present themselves beyond the Turing Test.

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

## 1. Introduction

Turing realised that only machines with very special self-modifying programs could eventually pass the test he had just devised (Turing, 1950). The ability to learn new functions for which the machine had not explicitly been programmed has since then been recognised as a necessary condition for a machine to be considered intelligent. Accordingly, the relationships between AI, IQ tests, inductive inference, learning, and descriptional complexity soon began to be discovered and, in many cases, formalised.

Many early works in AI dealt with the relation between IQ tests and AI (Simon and Kotovsky, 1963) (Evans, 1963). The association between AI and inductive inference was taken on by Solomonoff (Solomonoff, 1957), leading him to the independent discovery of descriptional complexity (also known nowadays as algorithmic information or Kolmogorov complexity), and the formalisation of inductive inference under this complexity (Solomonoff, 1964)(1978). Since intelligence tests occasionally require the extrapolation of an effective sequence, the connection between inductive inference and IQ tests was frequently alluded to (Gold, 1967)(Blum and Blum, 1975). In particular, the idea of "measuring machine power-intelligence as the scope of the *class* of inferable functions" is suggested for the first time (Blum and Blum, 1975). Nonetheless, it is not until Chaitin's challenge to "*develop formal definitions of intelligence and measures of its various components*"

(Chaitin, 1982), that the proposal of measuring intelligence by the use of descriptional complexity is explicitly solicited.

The problem of extrapolating a sequence (i.e. sequential inductive inference) has been clarified and formalised under descriptional complexity. See e.g. (Solomonoff, 1978), (Zvonkin and Levin, 1970), (Li and Vitányi, 1997). However, the *evaluation* of inductive *abilities* (a notably different problem) has not been successfully addressed to date, at least in the way Chaitin suggested. As expected, many of the technical tools and results that will be used in the following are borrowed from Kolmogorov complexity, but some *new* constructs will be introduced in order to make the evaluation feasible and to make it meaningful.

The motivation for such an evaluation is a first step in the construction of a scientific measure of intelligence, which should be compliant with the following assumed requirements: *non-Boolean, factorial, non-anthropomorphic, computational* and *meaningful*. As will be discussed in later sections, despite the philosophical and enlightening virtues of the TT, when implemented, it has various drawbacks. First, it is difficult to gauge, it is not factorial, it is absolutely anthropomorphic, it is informal and it does not give an objective meaning to the word 'intelligence'. On the contrary, IQ tests provided by psychometrics (Spearman, 1904)(Neisser et al., 1996)(Eysenck, 1979) are non-Boolean and factorial. However, psychometrics has neglected (or failed) to incorporate the last three requirements, which, in fact, are highly related. Psychometrics, as the science of measuring human intelligence, is anthropomorphic by definition. The factorial approach has not provided much insight, and no meaning can be extracted from "intelligence is what is measured by intelligence (IQ) tests". I realise that any definition is arbitrary, but it would be qualitatively better if the exercises that compose the tests were related to or derived from computational concepts.

In this paper, a test for one of the main factors of intelligence (comprehension ability) is introduced, according to the five prerequisites mentioned above. I would like to point out that the test is exclusively based on concepts which are derived from the notion of the Turing machine.

The paper is organised as follows. The next section introduces some necessary tools which will be used in the rest of the paper. It will also provide a description of 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. 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 for the construction of a comprehension test (C-test). Section 5 presents the results of applying the C-test to humans and compares it with psychometrical tests. Its applicability to AI is discussed. Section 6 studies the measurement of other factors (knowledge applicability, contextualisation, knowledge con-

struction) under the same conditions that the C-test was devised with. The TT is re-examined in section 7 and compared with the C-Test. The final section concludes with the claim of a new science of intelligence that would make it feasible to answer many new and fascinating questions which lie ahead.

## 2. Preliminaries and Technical Problems

Let us choose any finite alphabet $\Sigma$ composed of symbols (if not specified, $\Sigma = \{0,1\}$). A string or object is any element from $\Sigma^*$, with $\cdot$ being the composition operator, which is usually omitted. We define a sample space $S_\Sigma$ consisting of all finite strings and infinite sequences over $\Sigma$, i.e. $S_\Sigma = \Sigma^* \cup \Sigma^\infty$. By $\langle a, b \rangle$ we denote a standard recursive bijective encoding of $a$ and $b$, such that there is a one-to-one correspondence between $\langle a, b \rangle$ and each pair $(a, b)$. Note that this usually takes more bits than $a \cdot b$. The empty string is denoted by $\varepsilon$. 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 $y$ composed of $l$ symbols, we denote the symbols from position $n$ to position $m$ by $y_{n..m}$ if $1 \leq n \leq m \leq l$. Otherwise it is undefined. With $y_{..m}$, $y_{n..}$, and $y_k$, we denote $y_{1..m}$, $y_{n..l(y)}$, and $y_{k..k}$, respectively. If $y$ has infinite length, $y_{n..}$ denotes the infinite sequence $y_{n..\infty}$. Given any string $x$, we denote by $x_{-d} = x_{1..l(x)-d}$ the prefix of $x$ with length $l(x) - d$, i.e. the string $x$ without its last $d$ elements.

The complexity of an object can be measured in many ways, one of which is its degree of randomness, which turns out to be essentially equal to its shortest description (Kolmogorov, 1965). Descriptional Complexity, now commonly referred to as 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 (Li and Vitányi, 1997)(Gammerman and Vovk, 1999).

However, the algorithmic prefix version of descriptional complexity, usually denoted by $K(x)$ is unsuitable for our purposes because it is not monotone on prefixes, i.e. $K(xy)$ can be less than $K(x)$. $K(x)$ is problematic for prediction in the continuous case (see e.g. Vitány and Li 1997), but, more importantly, the use of $2^{-K(x)}$ as a probability prior would imply that the probability of the sequence $0^n$ to be followed by a 0 is greater if $n = n_0 = 10^{10}$ than if $n = n_1 = 141568756142169$, which is quite counterintuitive because $n_0 < n_1$. To avoid these problems, we shall work with monotone machines.

There are slightly different definitions of monotone machines (Solomonoff, 1964)(Levin, 1973)(Schnorr, 1973). We follow (Li and Vitányi, 1997):

*Definition 1.* A **Monotone Machine** $\beta$ is a Turing machine with a one-way read-only input tape, some work tapes, and a one-way write-only output tape. The input tape contains a one-way infinite sequence of 0's and 1's and

initially the input head scans the leftmost bit. The output tape is written one symbol in $\Sigma$ at a time, and the output is defined as the finite binary sequence on the output tape if the machine halts, and the possibly infinite sequence appearing on the output tape in a never-ending process if the machine does not halt at all. For a (possibly infinite) sequence $x$, we write $\beta(p) = x$ if $\beta$ outputs $x$ after reading $p$ and no more. (Machine $\beta$ either halts or computes forever without reading additional input).

*Definition 2.* The **Monotone Complexity** of an object $x$ given $y$ on $\beta$, with $\beta$ being a monotone machine, is defined as:

$$Km_\beta(x|y) = \min_p\{l(p) : \beta(\langle p, y\rangle) = x\omega,\ \omega \in S_\Sigma\}$$

The monotone complexity of an object $x$ is denoted by $Km_\beta(x) = Km_\beta(x|\varepsilon)$.

There is an additively optimal monotone machine $U$ such that there exists an independent constant $c$ such that for any other monotone machine $\beta$ and for all $x$ $Km_U(x) < Km_\beta(x) + c$. If we select this machine $U$ as a reference machine, the subscript can be dropped, thus assuming only constant errors. Kolmogorov Complexity and the monotone variant $Km$ also constitute an absolute and objective criterion of complexity, and they are independent (up to a constant term) of the descriptional mechanism $\beta$ due to the invariance theorem. The relationship between monotone complexity and other variants of Kolmogorov complexity is of logarithmic additive terms (see e.g. Li and Vitányi, 1997).

Occam's razor, which states that "given two alternative explanations, choose the simplest one", was formalised under descriptional complexity by (Solomonoff, 1964), approximated by Rissanen in 1978 under the name "Minimum Description Length" principle (MDL), finally re-formulated in its current one part code (Rissanen, 1996)(Barron et al., 1998).

In (Vitányi and Li, 1997) it is shown that under some reasonable assumptions on the $\mu - probability$ of correctly extrapolating a sequence, *a fixed-length y extrapolation from x maximises* $\mu(y|x)$ *iff it minimizes* $Km(xy) - Km(x)$. In other words, this difference, which is always positive since $Km$ is monotone, states that the shorter that the description of $xy$ wrt. $x$ is (i.e., the less novel $y$ is), the better the prediction is.

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 why such an intriguing proposal has not yet been implemented:

1. $K(x)$ and $Km(x)$ are not computable. If a compression test is constructed, how do we know whether the subject's answer is a hit?
2. There can be different alternative plausible descriptions. In other words, there may exist a $y'$ such that $Km(xy') = Km(xy) + c$ with $c$ being a very small constant.

3. Despite the invariance theorem, the constant involved is relevant, and there is no reason to prefer one descriptional system over another.
4. The test would finally measure the ability of compression, but, as will be argued in the following section, this differs slightly with the ability of comprehension, the main factor of intelligence that is to be measured.

The first problem can be solved by incorporating time into the definition of $Km$. The most appropriate way to weight the space and time execution of a program, the formula $LT_\beta(p_x) = l(p_x) + \log \tau_\beta(p_x)$, where $\tau$ is the number of steps the machine has taken until $x$ is printed, was introduced by Levin in the seventies[1] (see e.g. Levin, 1973). The corresponding complexity, denoted by $Kt$ (see e.g. Li and Vitányi, 1997) is a very practical alternative to $K$, because as well as avoiding intractable descriptions, it is computable. Moreover, it better accounts for the idea of simplicity, and Occam's razor should be better formalised under this variant.

Let us parametrise the definition of $\tau$ in the following way: $\tau_\beta(p)[..n]$ 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. Consider also $\tau_\beta(p)[n..m] = \tau_\beta(p)[..m] - \tau_\beta(p)[..n-1]$. In the same way, $LT_\beta(p)[n..m] = l(p) + log\, \tau_\beta(p)[n..m]$ and $LT_\beta(p)[..n] = l(p) + log\, \tau_\beta(p)[..n]$:

Then, the next variant comes directly and is a parametrisation of $Kt$:

*Definition 3.* The **$k$-Projectable Length-Time Complexity** of an object $x$ given $y$ on a monotone machine $\beta$ is defined as:

$$Ktm^k_\beta(x|y) =$$
$$\min_p\{LT_\beta(\langle p, y\rangle)[..l(x)] - l(y) : \exists\omega \in S_\Sigma\, l(\omega) \geq k \text{ s.t. } \beta(\langle p, y\rangle) = x\omega)\}$$

Since $LT_\beta(\langle p, y\rangle)$ takes the length of $y$ into consideration, this must be corrected by the term $-l(y)$. It is trivial to see that $Ktm^0$ is the corresponding monotone notion to $Kt$. Definition 3 will serve as a starting point for facing the other three unsolved problems (2,3,4). In fact, we first require distinguishing what comprehending is (problem 4), which is addressed in the following section, and in Section 4 we shall address how to measure the comprehension ability (problems 2 and 3).

## 3. Formalising Comprehension

To comprehend is to understand the inner mechanism of a given evidence by constructing a plausible model of it. In some way, comprehension is stricter than inductive learning in terms of justification, because comprehension usually requires that the subject be able to explain the concept to others. In the case of infinite concepts, this explanation is only possible if the subject

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

has a finite description of the concept. Consequently, comprehension could be understood in terms of identification. However, if a concept is finite, like most *everyday* concepts, both notions diverge significantly. A short finite concept can be easily identified by its extensional description, which has no insight and which has surely not identified any mechanism or pattern from it, if the evidence ever had one. This is an age-old question 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 requires 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 at least it has a chance of having discovered the right one.

There is a fundamental feature that determines this difference, known as the comprehension requirement, namely that *the thing being defined cannot appear in the definition*, which is also one of the four laws of definition, according to methodology (Bochenski, 1965). Ancient and modern teachers have used it whenever they ask their pupils whether they have comprehended a concept. This is one of the oldest evaluation criteria ever used and a premise with which the pioneer of the psychometric approach, Binet, designed his first tests to avoid "rote learning".

At first sight, Kolmogorov complexity seems sufficient to distinguish extensional descriptions from intensional ones. However, the ideal 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 gives the void hypothesis plus the data itself as a set of exceptions. This most extensional description gives no 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 is highly compressed while another part is quoted as an exception.

*Example 1.* Given the sequence $1^n$, where up to $m$ bits have been set to 0 by using pattern $p$. The MDL principle will give a sequence of $1^n$ plus the exceptions as the most plausible hypothesis, and will predict 1, because it minimizes $Km(xy) - Km(x)$. The zeros will be considered as noise (not explained) until the cost of quoting the exceptions exceeds the cost of $p$. In fact, it is easy to see that this problem only happens for relatively short sequences, because, if there is a pattern, there is always a value of $n$ from which the use the pattern begins to be simpler than to quote the exceptions. One may argue, that for short sequences, exceptions would simply be allowed or not, depending on the purpose of the inductive technique: prediction or explanation, respectively. However, we will see that this is not easy.

The MDL principle *avoids* this problem by finding a compromise between the length of the hypothesis plus the length of exceptions, since *"it is difficult to find a valid mathematical way to force a sensible division of the*

*information at hand in a meaningful part and a meaningless part"* (Vitányi and Li, 1997). Koppel introduced the notion of sophistication with the goal of distinguishing the structural part of an object (Koppel, 1987) from its data (or non-compressible part of it). However, it can 'disguise' a general effective interpreter as fictitious pattern and leave a great amount of real pattern as data. Thus, a different approach is required to distinguish whether a description has exceptions (partially or totally extensional) or whether is composed exclusively of pattern (it is all structure or fully intensional).

One positive result of this paper is that it is possible to distinguish pattern from data, at least to the extent of discerning the part of a description which is used for all the data *to the limit* (the structure). Let us introduce the necessary constructs for this mathematisation:

*Definition 4.* A description $p'$ is $m$-**equivalent in the limit** to a description $p$ for a monotone machine $\phi$ iff $\exists n \in \mathbb{N}, n > 0$ and $\exists z \in \mathbb{Z}$ such that $\phi(p')_{n+z..n+z+m} = \phi(p)_{n..n+m}$

Note that if $l(\phi(p)_{n..}) = s < m$ then the subscript is not well-defined and the descriptions are not $m$-equivalent (they would be $s$-equivalent). In what follows, $m$ is the 'match', $n$ is the 'shift' and $z$ is the 'phase'. Informally, two descriptions are $m$-equivalent in the limit if there is an 'alignment' point from which their predictions match at least $m$ symbols.

*Definition 5.* A description $p$ is an $m$-**fully Projectable Description** of $x$ given $y$ on a monotone machine $\phi$ iff $\neg\exists p'$ with $\phi(\langle p', y\rangle)_{..m'} \neq \phi(\langle p, y\rangle)_{..m'}$ such that $\langle p', y\rangle$ is $m$-equivalent in the limit to $\langle p, y\rangle$ with shift $n$ and phase $z$ and $n + z \leq l(x)$ and $LT(\langle p', y\rangle)[n + z..n + z + m'] < LT(\langle p, y\rangle)[n..n + m']$ with $m' = l(x) + m$.

The concept is more insightful for $m = \infty$. In this case, we would read that an $\infty$-fully projectable description of $x$ cannot have a simpler description which is different form $p$ and which at the same time is $\infty$-equivalent in the limit that. Note that $LT$ is used instead of $l$ and only applied to the first chunk of length $m'$ where $p'$ and $p$ begin to be equivalent. The reason for introducing a parameter $m$ in both definitions is because it is impossible to effectively know whether two sequences are equivalent up to the infinite. In practice, the highest goal we can aim for is that $m$ be the greatest number possible.

*Example 2.* Given the evidence "3, 12, 21, 30, 102, 111, 120" (properly codified into a binary sequence) we can consider several projectable descriptions. For instance, $D_1$ = "3, 12, 21, 30, 102, 111, 120 and 1 forever" is not fully projectable 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 projectable

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 amount to 3 ordered" is fully projectable. Similarly, the description $D_4$ = "repeat 3, 12, 21, 30, 102, 111, 120 forever" is fully projectable (unless there is a way to compress the rote pattern). Finally, the following description is also fully projectable $D_5$ = "the $y$ values of a polynomial $y = P(x)$" where $P$ is a polynomial such that $P(1) = 3$, $P(2) = 12$, ..., $P(7) = 120$.

Although $D_4$ and $D_5$ may seem counterintuitive, it should be realised that a fully projectable description simply 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 projectable) 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 6.* The $m$-**Explanatory Complexity** of an object $x$ given $y$ on a monotone machine $\beta$ is defined as:

$$Et_\beta(x|y) = \min_p\{LT_\beta(\langle p, y\rangle)[..l(x)] - l(y)$$

such that $p$ is an $m$-fully projectable description of $x$ given $y$ $\}$

The string $y$, which we have carried along, represents the context or previous knowledge where the explanation must be applied.

In the same way as is done with $Km$ and the MDL principle, we can denote with $SED(x|y)$ the Simplest (in $LT$ terms) $\infty$-Explanatory Description for $x$ given $y$, i.e. the first simplest fully projectable description (in lexicographic order) 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[2]. However, most everyday data is not compressible and it is still comprehended.

Another approach is the idea of reinforcement or cross-validation. For instance, if we remove the last element of the previous series, i.e. "3, 12, 21, 30, 102, 111", it is not very likely that $D_4$ and $D_5$ be produced; however $D_3$ can still be generated. In general,

*Definition 7.* **Stability**. A string $x$ is *s-stable on the right* given $y$ in the descriptional system $\beta$ iff $\forall i, 1 \leq i \leq s : SED_\beta(x_{-i}|y) \equiv SED_\beta(x|y)$.

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

[2] 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).

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 projectable, stability (like reinforcement or cross-validation) is a methodological criterion which can be used to avoid this phenomenon. Both conditions (fully-projectableness and stability) are necessary.

*Example 3.* Consider the sequence $x = 1^i01^j$ with $i$ and $j$ being random and independent. Imagine that the description $p =$ "print 1 forever except from position $i+1$" is selected as the shortest description for $x$ and stability is defined in terms of MDL instead of SED. Under these conditions $p$ would be $j$-stable. On the contrary, this does not happen with SED because $p$ cannot be fully projectable, since there exists a different description $p' =$ "print 1 forever" which is simpler and equivalent to $p$ from position $i + 1$.

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 innately find more complex patterns than the minimal description? (Marcus et al., 1999) This search for more informative and explanatory hypotheses instead of the shortest 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 these hypotheses (Harman, 1965). This informativeness or investment in the hypotheses was advocated by Popper for the scientific method (Popper, 1962), and as we have seen, it is equally applicable for cognition. Even if we make the very strong assumption of Occam's razor, i.e., things in nature are not unnecessarily complex, the previous rationale is justified by the fact that, just as every incompressible string has compressible substrings, *most* compressible strings have incompressible substrings, because there comes a point where the string is so short that it is not worth compressing. 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 in the end will not turn out to be part of the whole description of the whole evidence. This rationale leads to the next theorem:

*Theorem 1.* For every monotone machine $\beta$, there exists a constant $c$ which depends exclusively on $\beta$ such that for every string $x$ of length $n$ with $SED(x) = s$ and $l(s) = m$ such that $m < n$, and for every partition $x = yz$ such that $l(y) < m - c$, then $SED(y)$ is not equivalent in the limit with $s$.

*Proof.* Consider any string $x$ and $SED(x) = s$ with $l(s) = m$ such that $m < n$. Take any prefix $y$ such that $l(y) < m - c$. Consider the 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 $s$. Jointly, $LT_\beta(p_y)[..l(x)] = l(p(y)) + \log(k' \cdot l(x)) < m - c + c' + \log k' + \log l(x) \leq m - c' - \log k' + c' + \log k' + \log l(x) =$

$m + \log l(x) \leq LT_\beta(s)[..l(x)]$ since $l(s) = m$ and $x$ cannot be printed in less than $l(x)$ steps. Obviously, $LT_\beta(SED(y))[..l(x)] \leq LT_\beta(p_y)[..l(x)]$ because $SED$ is the simplest explanation in $LT$ terms. From here, we finally have that $SED(y)$ is simpler (in $LT$ terms) than $s$. Consequently, $s$ and $SED(y)$ cannot be equivalent in the limit because $s = SED(x)$ is fully projectable and, by definition, there cannot exist a description with less $LT$ equivalent in the limit. □

Although the result is still worse for the MDL principle, as shown in Example 1, the theorem seems to also discredit SED. However, if we demand stability this does not happen, because $p_y$ would not be stable. The idea of stability or cross-validation is then supported by the previous theorem. In fact, it is an innate *aesthetic* preference in the explanations that human beings generate. Why does answer 23 seem better to the series "3,7,11,15,19, ..." than answer 3? Why is 23 the 'correct' solution in IQ tests? In Hofstadter's 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).

Despite the fact that hardly any definition can completely grasp the intuitive notion that generates it, the arguments provided in this section allow us to state that $SED$ descriptions which are stable formalise the notion that comprehension has taken place. The following section is devoted to ensuring that the descriptions get the same meaning from a sequence. It also discusses how to measure the difficulty of an instance.

## 4. Testing Comprehension Ability

Theoretically, there are two ways to know whether a system's operation is compliant with certain requirements: by inspecting its code (or program) or by testing its behaviour. In general, for complex systems, as has been finally recognised in software engineering, verification must be experimental in practice, by means of sets of tests. However, it is an open and difficult problem to devise a *complete* specification of intelligence, mainly because it depends on a consensus on the *abilities* that an intelligent system should have. Nonetheless, it is currently possible to distinguish certain abilities that are fundamental for intelligence. A verification of intelligence behaviour should begin with these fundamental traits and gradually add more diverse (factorial) exercises in order to make the test set more robust. Traditionally, comprehending is recognised as the most important trait of intelligence, and we have formalised it in a computational framework. This allows for the construction of exercises for a test which are selected theoretically rather than experimentally. This does not mean that they are necessarily more representative, but at least we know exactly what is 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 which is independent to the mean ability of the subjects or society who have taken the test before.

With respect to the notion of unquestionability, psychometrics has striven to show that it is not absurd to talk about the 'correct' solution, at least if by 'correct' we mean the prediction of the simplest comprehensive answer. Its rationale is that if the great majority matches some solution it is *because there are not alternative explanations of similar complexity*, and, consequently, it is the most plausible one. However, this assertion is made from a very subjective and informal point of view.

At first glance, it seems that given some data $x$ of length $n$, we can still modify any explanation $p$ with the addendum "Execute $p$ but print a '1' every $m$ symbols that are printed beginning from $n+1$". This alternative explanation would be comprehensive for the data but would differ from $p$ in the limit. It would only be a little longer and this would depend on the descriptional machine used.[3] To avoid these problems in an implementation of a test, the following constructions are sufficient:

*Definition 8.* **Plausibility**. A fully projectable description $p$ for a string $x$ given $y$ is $(c,d)$-*plausible on the right* in a monotone machine $\beta$ iff $\forall i, 0 \leq i \leq d : LT_\beta(SED_\beta(x_{-i}|y))[..l(x_{-i})] + c > LT_\beta(p|y)[..l(x_{-i})]$.

Intuitively, a description is $(c,d)$-plausible if it is at most $c$ bits longer (in LT terms) than the best explanation for $x$, and this holds even if we remove up to $d$ elements from the right of $x$.

*Definition 9.* **Unquestionability**. A fully projectable description $p$ for $x$ is $(c,d)$-*unquestionable* in a monotone machine $\beta$ iff it is $(c,d)$-*plausible* and there does not exist another $(c,d)$-*plausible* description $p'$ for $x$.

This is a more restrictive condition as $c$ and $d$ are greater. In order to still obtain some unquestionable descriptions we must make the strings larger. However, as we shall see below, if $c$ and $d$ 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 $(c,d)$-unquestionable.

The second question was to 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 to the *complexity* of the simplest (in

[3] This is a very difficult problem which can be addressed by recognising the addendum as a non-reinforced part (an exception). This has been done in (Hernández-Orallo, 2000) for universal (constructive) representations, but it is not easy to extend the framework to any universal representations (Hernández-Orallo and Minaya-Collado, 1998).

*LT* terms) explanation (i.e. *Et*) and the *explicitness* of the description wrt. the data. To do this, we adapt the definition of potential (Li and Vitányi, 1997) and the notion of k-compressibility to the corresponding notion of comprehensibility:

*Definition 10.* A string $x$ is $k$-**incomprehensible** given $y$, denoted by $incomp(x|y)$, in a descriptional system $\beta$ iff $k$ is the least positive integer number such that: $Ktm_\beta(SED_\beta(x|y)|\langle x,y\rangle) \leq k \cdot \log l(x)$.

The use of the factor $\log l(x)$ is to compensate the fact that $x$ must be printed and, therefore, for all $x$ we have $Et(x) \geq log\, l(x)$. E.g., consider a string $x$ of length 256 and $y = \varepsilon$, with $Et(x) = 50$; its comprehensibility is $k$= 7.

Definition 10 measures the difficulty of finding $SED(x|y)$ from $x$ and $y$, because descriptions of the form "repeat $x$ forever" for $x^n$ have a high absolute $Et$ value (to quote $x$) but low relative complexity (w.r.t. $\langle x,y\rangle$).

Now a generic test of the ability of comprehension can be constructed by generating a series of strings of gradual comprehensibility. Unquestionability is achieved by providing 'redundant' information up to a limit, because, otherwise, the problems would be much too long. However, there must be sufficient support to not distort its difficulty. In other words, when the subject finds the solution, it should be sure that he/she/it has found it. For instance, given the series "a, c, c, a, c, c, c, a, c, c, c, c, a, ..." it seems logical to expect that the series would follow "c, c, c, c, c, a, c, ...", so it is redundant to present more than the necessary symbols, but less would make the answer questionable.

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

*Definition 11.* **C-Test**. Let us select a descriptional system[4] $\beta$ which is sufficiently expressive and impartial, and which is composed of an alphabet of symbols $\Omega_\beta$ and a set of operations $\Theta_\beta$. These operations manipulate these symbols, and they have a corresponding *cost* (or length). We provide (or programme) the alphabet, operations and cost to $S$. Depending on the expected intelligence of a system we select a sufficiently wide range $1..K$ of difficulty. For each $k = 1..K$, we randomly choose $p$ sequences $x^{k,p}$ which are *k-incomprehensible*, (*c*,*d*)-*plausible*, (*c*,*d*)-*unquestionable* and *s-stable* with $s \geq r$, $r$ being the number of redundant symbols (or hints) of each exercise. The questions are the $K \cdot p$ sequences without their $s-r$ elements ($x^{k,p}_{-(s+r)}$). We give them to $S$ and we ask for the following element according to the best explanation that is able to construct with $\Omega_\beta$ and $\Theta_\beta$. We leave $S$ a

[4] From now, we shall deal with any type of descriptional system since any machine can be 'wrapped' into a monotone machine or, alternatively, monotone complexity can be computed on non-monotone machines by providing the length of the input string in an additional input tape.

fixed time $t$ and we record its answers: $guess(S, x^k_{-s+r+1})$. The result of this test of comprehensibility (or C-test) is measured as:

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

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

In an informal way, the test measures the ability of finding the best explanation for sequences of increasing comprehensibility in a fixed time.[5] The relevance of the time given and the weighting $e$ of the difficult exercises is still an open question. I would be in favor of either including (logarithmically) the time in the resulting value $I(S)$ or fixing a high time and penalising wrong results with a negative value of $hit[a, b]$ (blank answers with a zero value). However, this would create problems if you would ever want to measure two different things: the intelligence of a subject and the speed of the subject.

## 5. Measurement of Pretended Intelligent Systems

The preceding test is applicable to any system whose degree of intelligence is questioned. The test can be used for humans, animals, computers, extraterrestrial beings and any combination of these by appropriately selecting the descriptional system and the rest of parameters of the test.

Although Definition 11 evaluates a single ability, there are still many ways to devise a specific test. An implementation of the test is described in (Hernández-Orallo and Minaya-Collado, 1998). The abstract state machine which was used is not monotone, but this difference is not relevant due to the stability condition. 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 the use of heuristics. In the same way, $m$ fully-projectable descriptions were checked up to a given length limit $m$. Finally, a sieve was applied in order to obtain only ($c$,$d$)-*plausible*, ($c$,$d$)-*unquestionable* and *s-stable* sequences. The creation of the test took several days in all.

The same work presents the results[6] of applying the test to 65 subjects from the species Homo Sapiens Sapiens aged between 14 and 32 years (jointly

[5] One relevant feature of the test is that, although the subject is supposed to have a particular universal descriptional system $\phi_s$ with a particular background knowledge (life experience) $B_s$, it is given a descriptional system $\beta$ over it, which highly minimises the influence of the difference between the computations performed by $\phi_s$ and other subject $\phi_t$, 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.

[6] For more information about the experimental setup, methodology, questionnaire, subjects, times, etc. consult (Hernández-Orallo and Minaya-Collado, 1998). The web page

with a classic test of intelligence, the *European IQ Test*). The correlation between both tests was 0.77. This value does justify a further more exhaustive study on larger groups and several variations derived from Definition 11. Another remarkable experimental result shown in Fig. 1 is that the relation between the hit ratio (the percentage of subjects that gave the right answer) and $k$-incomprehensibility is direct, which suggests that comprehensibility really estimates the difficulty of each string.

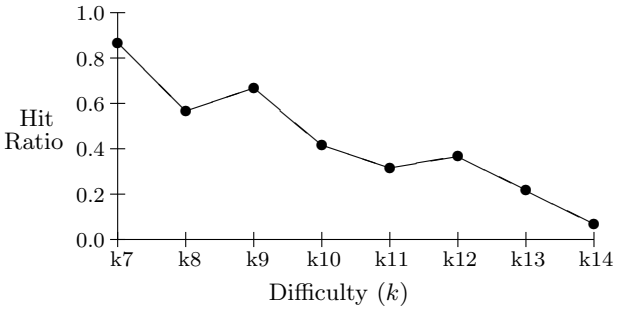


*Figure* 1

Logically, the C-tests cannot be expected to substitute contrasted and widely used IQ tests for the moment. Nonetheless, this could be a starting point towards a theoretical foundation of psychometrics which is *free* from the Homo Sapiens as a reference.

However, it is not human intelligence but non-human intelligence which urgently needs to be measured. A formal declaration of what is expected from an intelligent system should allow for two important things: to derive more intelligent systems from a more concrete specification and, secondly, to evaluate them. Definition 11 provides a first step for both, since a detailed scale for measuring the progress (in one intelligence factor) of generic systems in AI can help to establish the first one. As any other field of science, a great advance in a discipline occurs when one of its topics can be measured in an effective and justified way. Just as aeronautics needs altimeters and speedometers, AI requires measurements of different factors of intelligence.

Nowadays, the initial aim of making general systems is 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 automated deduction over the last two decades can be mainly attributed to the existence of sets of problems for comparing different systems. These sets have evolved and grown to huge and complete libraries of theorem proving problems, such as TPTP (Suttner and Sutcliffe, 1998). Machine learning is also taking a more experimental character and different systems (from different paradigms) are evaluated according to classical (toy) problems in the literature rather than exclusively

`http://www.dsic.upv.es/~jorallo/itests/` includes an up-to-date summary of results and an archive of past and on-going tests.

accepting the results for *classes* of problems which are theoretically expected to occur. In my opinion, experimental test sets should also be automatically generated or at least accompanied by a theoretical measurement about the complexity of each exercise. This complexity could be obtained by adapting the previous notions to several representational languages.

## 6. Factorisation

During the XXth century, psychometrics strove to differentiate 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 in order to be foolproof to 'idiots savants'. Even with this restriction, there are still 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 partially independent factors which could be measured by using extensions of the framework presented in the previous section. For instance, other inductive abilities, such as knowledge applicability, contextualisation and knowledge construction ability, can be measured in the following way:

- Knowledge Applicability (or 'crystallized intelligence'): a background knowledge $B$ and a set of unquestionable (with or without $B$) sequences $x_i$ are provided such that $incomp(x_i|B) = incomp(x_i) - u$ but still $SED(x_i|B) = SED(x_i)$. The difference in performance between cases with $B$ and without $B$ is recorded. This test would actually measure the application of the background knowledge depending on two parameters: the complexity of $B$ and the usefulness of $B$, measured by $u$.
- Contextualisation: it is measured in a way similar to knowledge applicability but different contexts $B_1, B_2, ..., B_T$ are supplied 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 $T$) distinguishes this factor from the previous one.
- Knowledge Construction (or learning from precedents): a set of sequences $x_i$ is provided such that there exists a common knowledge or context $B$ (now not given) and a constant $u$ such that for $incomp(x_i|B) \leq incomp(x_i) - u$. A significant increase in 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 $u$.

It is obvious that these three factors should correlate with the comprehension ability. Other non-inductive factors, especially deductive abilities, are

seemingly easier to measure because there is no problem of unquestionability. It is expected that analogical and abductive abilities[7] can be shown to be closely connected to inductive and deductive abilities both theoretically and experimentally. Inductive abilities, especially knowledge applicability, may also be correlated with deductive abilities (any hypothesis must be checked deductively) and these may also correlate with the idea of congruence or coherence, since it has been shown to be theoretically equivalent to constraint satisfaction (Thagard, 1989).

To show experimental correlations (especially for non-human and/or non-adult subjects), the presentation of the test must change slightly. The exercises should be given one by one and, after each guess, the subject must be given the correct answer (rewards and penalties can be used instead). This has two advantages: there is no need for the subject to understand natural language (or any language) in order to explain the purpose of the test to the subject, and there is no need to tell which factor or purpose is to be measured in each part of the test. There is also one disadvantage, deductive problems should be posed in terms of 'learn to solve' or 'learn to prove' in a way similar to that used by (Solomonoff, 1957) suggested with simple problems of arithmetic. Properly, this problem is not prediction but classifying, i.e. to know which elements could be 'theorems' (class true) in that model. In this sense, it would be interesting to evaluate non-sequential induction, where an unordered set of elements is given as evidence, in the way that Solomonoff has recently formalised (Solomonoff, 1999). In fact, non-sequential induction would be more related to deductive ability while sequential induction would be more related to calculation ability.

Some other factors are more related to *intentionality* than *intensionality* and general intelligence. These are reactivity, pro-activity and interactivity, that could eventually be measured by modifying the C-test. This could be done by adopting notions from Query Learning paradigms (Angluin, 1988) or by using interactive Turing machines. However, not every factor will be 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 then to refine and extend the previous notions in order to make factorial and grounded tests of intelligence, knowing exactly what is measured. This is an urgent and fascinating task for AI.

[7] See an attempt to measure them in (Hernández-Orallo and Minaya-Collado, 1998).

## 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 exclusivism and transcendentalism: intelligence can be evaluated by a solely behavioural test. Unfortunately, instead of recognising this as his most important contribution, the test is still considered 'a goal' in AI. Nonetheless, this view has been responded to by many authors, whose criticism is that the TT provides little information on to what intelligence really is; it is just a test of humanness (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. There is still a third problem, which is the need for several intelligent 'judges' and a 'referent' to implement 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 short-time versions of the TT, such as the Loebner Prize, which usually awards the participant who has devised the system which is better able to cheat the judges. Furthermore, there is no way of knowing who is cheating, the system or its designer.

However, if fairly played (and for long), the imitation game is a hard examination for any intended intelligent system. It is extremely difficult to behave like an average human being of this epoch (it is even difficult for some 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.

On the contrary, 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 already been suggested that both kinds of tests (TT and factorial) could be combined in order to give a more accurate intelligence test, because "*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). The motivation for such a combination is quite the same reason why IQ-tests are used jointly with an interview in post selections and for other evaluation purposes. However, the interview just shows that the questionnaire is incomplete or that the abilities that are measured in the interview are less related to intellect.

In my opinion, the TT should be celebrated as an extremely valuable philosophical exercise about the behavioural character of intelligence. For practical purposes, though, it will be necessary to implement progressively more accurate computational tests of different cognitive abilities.

## 8. Conclusions

Turing devised a way of distinguishing 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, in practice, this approach has numerous limitations and problems which make it useless for application in AI. Experience has shown that 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 a tiny first step along 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, which is very different from the TT and which is 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 to find comprehensive explanations, and thus is meaningful. This makes it philosophically acceptable: intelligence is what allows us to comprehend the world.

Sooner or later we need to face the fact that computers will come closer and closer to human intelligence. Once this milestone of AI has been achieved, it will be absolutely necessary to have an objective measure of intelligence, in order to solve the incipient technical and ethical problems that could be derived from that point. The paradigm presented in this paper allows for the projection of the measurement of intelligence beyond human intelligence.

Once beyond the TT, many more interesting questions present themselves. How many independent computational factors does human intelligence have? How intelligent is the Homo Sapiens? Which factors make a chimpanzee significantly different from the Homo Sapiens? How intelligent can machines be with the current computational power? Psychometrics, Anthropology, Zoology and AI have only partially dealt with some of these questions. Only a science of intelligence which is grounded in theoretical computer science and information theory could answer these questions thoroughly.

## 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 many collaborations and suggestions: K. Araque, R. Barreiro, R. Beneyto, N.T. Crook, E. Fueyo, I. García, E. Hernández, J.M. Lorente, N. Minaya, I. Soto and P. Thagard. Finally, I am especially grateful to the referees of this special issue for their justifiably incisive comments and numerous corrections on an earlier version of this paper.

## References


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

Barron, A., Rissanen, J. and Yu, B., 1998, "The Minimum Description Length Principle in Coding and Modeling", *IEEE Transactions on Information Theory*, **44**(6), 2743–2760.

Bien, Z., Kim Y. T. and Yang, S. H., 1998, "How to Measure the Machine Intelligence Quotient (MIQ): Two Methods and Applications', *World Automation Congress (WAC)*, TSI Press, Albuquerque, NM.

Blum L. and Blum M., 1975, "Towards a Mathematical Theory of Inductive Inference". *Information and Control*, 28:125–155.

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

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

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

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

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

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

Fostel, G., 1993, "The Turing Test is For the Birds", *SIGART Bulletin*, **4**(1), 7–8.

Gammerman, A. and Vovk, V. (eds.), 1999, Special Issue on Kolmogorov Complexity, *The Computer Journal*, **42**(4).

Gold, E. M., 1967, "Language Identification in the Limit", *Inform & Control*, **10**, 447–474.

Harman, G., 1965, "The inference to the best explanation", *Philos. Review*, **74**, 88–95.

Harnad, S., 1992, "The Turing Test Is Not a Trick: Turing Indistinguishability Is A Scientific Criterion", *SIGART Bulletin*, **3**(4), 9-10.

Herken, R., 1994, *The universal Turing machine: a half-century survey*, Oxford University Press, 1988, 2nd Edition, 1994.

Hernández-Orallo, J., 2000, Constructive Reinforcement Learning, *Intl. J. of Intelligent Systems*, **15**(3), 241–264.

Hernández-Orallo, J. and García-Varea, I., 1998, "Explanatory and Creative Alternatives to the MDL principle", in *Proc. of Intl. Conf. on Model Based Reasoning* (MBR'98), S. Rini, G. Poletti (eds.), Pavia 1998. Also to appear in *Foundations of Science*.

Hernández-Orallo, J. and Minaya-Collado, N., 1998, "A Formal Definition of Intelligence Based on an Intensional Variant of Kolmogorov Complexity" In *Proc. of the Intl. Symp. of Engin. of Intelligent Systems* (EIS'98), ICSC Press, 146–163.

Hofstadter, D. R., 1979, *Gödel, Escher, Bach. An Eternal Golden Braid*, Basic Books.

Johnson, W. L., 1992, "Needed: A New Test of Intelligence", *SIGART Bulletin*, **3**(4), 7–9.

Kolmogorov, A. N., 1965, "Three Approaches to the Quantitative Definition of Information", *Problems Inform. Transmission*, **1**(1):1-7.

Koppel, M., 1987, "Complexity, Depth, and Sophistication", *Complex Systems*, **1**, 1087-91.

Larsson, J. E., 1993, "The Turing Test Misunderstood", *SIGART Bulletin*, **4**(4), 10.

Levin, L. A., 1973, "Universal search problems", *Problems Inform. Transm.*, **9**, 265-6.

Li, M. and Vitányi, P., 1997, *An Introduction to Kolmogorov Complexity and its Applications*, 2nd Ed., Springer-Verlag.

Marcus, G. F., Vijayan, S., Bandi Rao, S. and Vishton, P. M., 1998, "Rule Learning by Seven-Month-Old Infants", *Science*, Jan-1998, 77–80.

Millican, P. J. R. and Clark, A. (eds.), 1996, *Machines and Thought. The Legacy of Alan Turing, Vol. I*, Clarendon Press, Oxford.

Neisser, U., Boodoo, G., Bouchard, T. J., Boykin, A. W., Brody, N., Ceci, S. J. Halpem, D. F., Lochlin, J. C., Perloff, R., Sternberg, R. J. and Urbina, S., 1996, "Intelligence: Knowns and Unknowns", *American Psychologist*, **51**, 77–101.

Popper, K.R., 1962, *Conjectures and Refutations: The Growth of Scientific Knowledge*, Basic Books, New York.

Preston, B., 1991, "AI, anthropocentrism, and the evolution of 'intelligence'", *Minds and Machines*, **1**, 259–277.

Rissanen, J., 1996, "Fisher information and stochastic complexity", *IEEE Trans. Information Theory*, IT-**42**(1).

Shapiro, S. C., 1992, "The Turing Test and The Economist", *SIGART Bltn*, **3**(4), 10–11.

Shieber, S. M., 1994, "Lessons from a Restricted Turing Test", *Comm. of the ACM*, **37**(6).

Schnorr, C. P., 1973, "Process Complexity and Effective Random Tests", *J. Comput. System Sci.*, **7**, 376–388.

Simon H. and Kotovsky, K., 1963, "Human acquisition of concepts for sequential patterns", *Psych. Review*, **70**, 534–46.

Solomonoff, R. J., 1957, "An Inductive Inference Machine", *IRE Convention Record, Section on Information Theory.*

Solomonoff, R. J., 1964, "A formal theory of inductive inference", *Inf. Control*, **7**, 1-22, March, 224–254, June.

Solomonoff, R. J., 1978, "Complexity-based induction sytems: comparisons and convergence theorems", *IEEE Trans. Inform. Theory*, IT-**24**, 422–438.

Solomonoff, R. J., 1997, "The Discovery of Algorithmic Probability", *Journal of Computer and System Sciences*, **55**(1), 73–88.

Solomonoff, R. J., 1999, "Two Kinds of Probabilistic Induction", in the 'Special Issue on Kolmogorov Complexity', *The Computer Journal*, **42**(4), 256–259.

Spearman, C., 1904, "'General Intelligence' objectively determined and measured", *Amer. J. of Psych.*, **15**, 201–293.

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

Sternberg, R. J. and Detterman, D. K., 1986, *What is Intelligence? Contemporary viewpoints on its nature and definition*, Norwood, NJ., Ablex.

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

Suttner, C. B. and Sutcliffe, G., 1998, "The TPTP Problem Library: CNF Release v1.2.1", *Journal of Automated Reasoning*, **21**(2), 177–203.

Thagard, P., 1989, "Explanatory coherence", *Behavioural and Brain Scis.*, **12**(3), 435–502.

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

Turing, A. M., 1936, "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.

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

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

Vitányi, P. and Li, M., 1997, "On Prediction by Data Compression", *Proc. 9th European Conference on Machine Learning*, LNAI, **1224**, 14–30, Springer, Berlin.

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

Zvonkin, A. K. and Levin, L. A., 1970, "The complexity of finite objects and the development of the concepts of information and randomness by means of the Theory of Algorithms", *Russian Math. Surveys*, **25**(6), 83–124.