Acceptable Complexity Measures of Theorems

Bruno Grenet

Laboratoire de l'Informatique du Parallélisme
École Normale supérieure de Lyon
46, allée d'Italie
69 364 Lyon Cedex 07, France
bruno.grenet@ens-lyon.fr

In 1931 Gödel [1] presented his famous incompleteness theorem in Königsberg, stating that some true mathematical statements are unprovable. Yet, this result gives us no idea about those independent (i.e., true and unprovable) statements, about their frequency, the reason they are unprovable, and so on. In 2005 Calude and Jürgensen [2] proved Chaitin's "heuristic principle" for an appropriate measure: the theorems of a finitely specified theory cannot be significantly more complex than the theory itself [3]. In this paper, we investigate the existence of other measures, different from the original, that satisfy this heuristic principle. Toward this end, we introduce a definition for acceptable complexity measures of theorems.

1. Introduction

In 1931 Gödel [1] presented his famous (first) incompleteness theorem, stating that some true mathematical statements are unprovable. More formally and in modern terms, it can be stated as:

> Every computably enumerable, consistent axiomatic system containing elementary arithmetic is incomplete, that is, there exist true sentences unprovable by the system.

The truth is here defined by the standard model of the theory we consider. Yet, this result gives us no idea about those independent (i.e., true and unprovable) statements, about their frequency, the reason they are unprovable, and so on. Those questions of quantitative results about the independent statements were initially investigated by Chaitin [3], and then by Calude, Jürgensen, and Zimand [4], and Calude and Jürgensen [2]. A state of the art is given in [5]. Those earlier works state that in both topological and probabilistic terms, incompleteness is a widespread phenomenon. Indeed, unprovability appears as the norm for true statements while provability appears to be rare. This interesting result brings two more questions. Which true statements are provable, and why are they provable when others are not?

https://doi.org/10.25088/ComplexSystems.18.4.403

Chaitin [3] proposed an "heuristic principle" to answer the second question: the theorems of a finitely specified theory cannot be significantly more complex than the theory itself. In [2] Chaitin's heuristic principle is proved to be valid for an appropriate measure. This measure is based on the program-size complexity: the complexity $H(s)$ of a binary string s is the length of the shortest program for a self-delimiting Turing machine (to be defined in Section 2) to calculate s [6-9]. We consider the following computable variation of the program-size complexity:

$$\delta(x) = H(x) - |x|.$$

This measure gives us some indication about the reasons certain statements are unprovable. It would be very interesting to have other results in order to understand the incompleteness theorem. Among them would be proving a kind of reverse of the theorem proved in [2]. Their theorem states that there exists a constant N such that any theory that satisfies the hypothesis of Gödel's theorem cannot prove any statements x with $\delta(x) > N$. Another question of interest could be the following: do any independent statements exist with a low δ-complexity?

Those results are some examples of what can be investigated in this domain. Yet, such results seem to be hard to prove with the δ-complexity. The aim of our work is to find other complexities that satisfy this heuristic principle to enable proving the remaining results. Toward this end we introduce the notion of *acceptable complexity measures of theorems* to capture the important properties of δ. After studying the results about δ in [2], we define acceptable complexity measures, study their properties, and try to find other acceptable complexity measures that are different from δ.

This paper is organized as follows. We begin in Section 2 with some notations and useful definitions. In Section 3 we present the results of [2] with some corrections. Section 4 is devoted to the definition of acceptable complexity measures of theorems, and some counter-examples are given in Section 5. Section 5 also contains a proof of the independence of the conditions that are imposed on a complexity measure to be acceptable. In Section 6, we are interested in the possible forms of those acceptable complexity measures.

2. Prerequisites and Notations

Throughout, $\mathbb{N}$ and $\mathbb{Q}$ respectively denote the sets of natural integers and rational numbers. For an integer $i \geq 2$, $\log_i$ is the base i logarithm. We use the notations $\lfloor \alpha \rfloor$ and $\lceil \alpha \rceil$ respectively for the floor and the ceiling of a real α. The cardinality of a set S is denoted by card(S). For every integer $i \geq 2$, we fix an alphabet X_i with i elements, X_i^*

https://doi.org/10.25088/ComplexSystems.18.4.403

being the set of finite strings on X_i, including the empty string λ, and $|w|_i$ being the length of the string $w \in X_i$.

We assume the reader is familiar with how Turing machines process strings [10] and with the basic notions of computability theory [11-13]. We recall that a set is said to be computably enumerable (abbreviated c.e.) if it is the domain of a Turing machine, or equivalently, if it can be algorithmically listed.

The complexity measure we study is the computable variation of the program-size complexity. In order to define it, we first define *self-delimiting Turing machines*, shortly *machines*, that are Turing machines with a prefix-free set as their domain: a set $S \subset X_i^*$ is said to be *prefix-free* if no string of S is a proper extension of another one. In other words, if $x, y \in S$ and if there exists z such that $y = xz$, then $z = \lambda$. We denote by $\mathrm{Prog}_T = \{x \in X_i^* : T \text{ halts on } x\}$ the program set of the Turing machine T. We recall two important results on prefix-free sets. If $S \subset X_i^*$ is a prefix-free set, then Kraft's inequality holds: $\sum_{k=1}^{\infty} r_k \cdot i^{-k} \leq 1$, where $r_k = \{x \in S : |x|_i = k\}$. The second result is called the Kraft-Chaitin theorem and states the following: let $(n_k)_{k \in \mathbb{N}}$ be a computable sequence of non-negative integers such that

$$\sum_{k=1}^{\infty} i^{-n_k} \leq 1,$$

then we can effectively construct a prefix-free sequence of strings $(w_k)_{k \in \mathbb{N}}$ such that for each $k \geq 1$, $|w_k|_i = n_k$.

The *program-size complexity* of a string $x \in X_Q^*$, relative to the machine T, is defined by

$$H_{i,T} = \min\{|y|_i : y \in X_i^* \text{ and } T(y) = x\}.$$

In this definition, we assume that $\min(\emptyset) = \infty$. The invariance theorem ensures the effective existence of a so-called "universal" machine U_i that minimizes the program-size complexity of the strings. For every T, there exists a constant $c > 0$ such that for all $x \in X_i^*$, $H_{i,U_i}(x) \leq H_{i,T}(x) + c$. In the following, we will fix U_i and denote by H_i the complexity H_{i,U_i} relative to U_i.

A *Gödel numbering* for a formal language $L \subseteq X_i^*$ is a computable, one-to-one function $g : L \to X_2^*$. By G_i, or G if there is no possible confusion, we denote the set of all the Gödel numberings for a fixed language. In what follows, we consider theories that satisfy the hypothesis of Gödel's incompleteness theorem, that is, finitely specified, sound, and consistent theories that are strong enough to formalize arithmetic. The first condition means that the set of axioms of the the-

https://doi.org/10.25088/ComplexSystems.18.4.403

ory is c.e., soundness is the property that the theory only proves true sentences, and consistency states that the theory is free of contradictions. We will generally denote such a theory by $\mathcal{F}$ and use $\mathcal{T}$ for the set of theorems that $\mathcal{F}$ proves.

3. The Function δ_g

In this section we present the function δ_g and discuss some results. The function was defined in [2] and almost all of the results come from that paper. Hence, complete proofs of the results can be found in [2]. Yet, there was a mistake in that paper, and we need to modify the definition of δ_g and adapt the proofs with the new definition. The transformations are essentially cosmetic in almost all of the proofs so we give only sketches of them. For Theorem 3, there are a bit more than details to change, so a complete proof is provided. Furthermore, we formally prove an assertion used in the proof of Theorem 3.

We first define, for every integer $i \geq 2$, the function δ_i by

$$\delta_i(x) = H_i(x) - |x|_i.$$

Now, in order to ensure that the complexity studied is not dependent on the way we write the theorems, we define the δ-complexity induced by a Gödel numbering g by

$$\delta_g(x) = H_2(g(x)) - \lceil \log_2(i) \cdot |x|_i \rceil,$$

where g is a Gödel numbering with a domain in X_i^*.

The definition in [2] was $\delta_g(x) = H_2(g(x)) - \lceil \log_2 i \rceil \cdot |x|_i$.

The first result comes in fact from [8], and the theorem presented here is one of its direct corollaries.

Theorem 1. [2, Corollary 4.3] For every $t \geq 0$, the set $\{x \in X_i^* : \delta_i(x) \leq t\}$ is infinite.

Proof. Following [8, Theorem 5.31], for every $t \geq 0$, the set $C_{i,t} = \{x \in X_i^* : \delta_i(x) > -t\}$ is immune (a set is said to be *immune* when it is infinite and contains no infinite c.e. subset). Hence, as $\text{Complex}_{i,t} = \{x \in X_i^* : \delta_i(x) > t\}$ is an infinite subset of an immune set, it is immune itself. Because the set in the statement is the complement of the immune set $\text{Complex}_{i,t}$, it is not computable, and in particular infinite. □

Theorem 2 states that the definitions via a Gödel numbering or without this device are not far from each other. It allows us to work with the function δ_i instead of δ_g and thus simplify the proof due to

https://doi.org/10.25088/ComplexSystems.18.4.403

the elimination of some technical details. Nevertheless, those details are present in our proof of Theorem 2.

Theorem 2. [2, Theorem 4.4] Let $A \subseteq X_i^*$ be c.e. and $g : A \to B^*$ be a Gödel numbering. Then, there effectively exists a constant c (depending upon U_i, U_2, and g) such that for all $u \in A$ we have

$$\left| H_2(g(u)) - \log_2(i) \cdot H_i(u) \right| \leq c. \tag{1}$$

Proof. We will in fact prove the existence of two constants c_1 and c_2 such that on one hand

$$H_2(g(u)) \leq \log_2(i) \cdot H_i(u) + c_1 \tag{2}$$

and on the other hand

$$\log_2(i) \cdot H_i(u) \leq H_2(g(u)) + c_2. \tag{3}$$

For each string $w \in \mathrm{Prog}_{U_i}$, we define $n_w = \lceil \log_2(i) \cdot |w|_i \rceil$. This integer verifies that

$$\sum_{w \in \mathrm{Prog}_{U_i}} 2^{-n_w} = \sum_{w \in \mathrm{Prog}_{U_i}} 2^{-\lceil \log_2(i) \cdot |w|_i \rceil} \leq \sum_{w \in \mathrm{Prog}_{U_i}} i^{-|w|_i} \leq 1,$$

because Prog_{U_i} is prefix-free. This inequality shows that the sequence (n_w) satisfies the conditions of the Kraft-Chaitin theorem. Consequently, we can construct, for every $w \in \mathrm{Prog}_{U_i}$, a binary string s_w of length n_w such that the set $\left\{ s_w : w \in \mathrm{Prog}_{U_i} \right\}$ is c.e. and prefix-free. Accordingly, we can construct a machine M whose domain is this set, such that for every $w \in \mathrm{Prog}_{U_i}$,

$$M(s_w) = g\left(U_i(w)\right).$$

If we denote, for a string $x \in X_i^*$, x^* the lexicographically first string of length $H_i(x)$ such that $U_i(x^*) = x$, we now have $M\left(s_{w^*}\right) = g\left(U_i(w^*)\right) = g(w)$, and hence

$$H_M(g(w)) \leq \left| s_{w^*} \right|_2 = \lceil \log_2(i) \cdot |w^*|_i \rceil = \lceil \log_2(i) \cdot H_i(w) \rceil \leq \log_2(i) \cdot H_i(w) + 1.$$

By the invariance theorem, we get the constant c_1 such that equation (2) holds true.

We now prove the existence of c_2 such that equation (3) holds true. The proof is quite similar. For each string $w \in \mathrm{Prog}_{U_2}$, we define

https://doi.org/10.25088/ComplexSystems.18.4.403

$m_w = \lceil \log_i(2) \cdot |w|_2 \rceil$. As for the n_w, the integers m_w satisfy

$$\sum_{w \in \mathrm{Prog}_{U_2}} i^{-m_w} \leq \sum_{w \in \mathrm{Prog}_{U_2}} 2^{-|w|_2} \leq 1.$$

We can also apply the Kraft-Chaitin theorem to effectively construct, for every $w \in \mathrm{Prog}_{U_2}$, a string $t_w \in X_i^*$ of length m_w such that the set $\left\{t_w : w \in \mathrm{Prog}_{U_2}\right\}$ is c.e. and prefix-free. As g is a Gödel numbering and hence one-to-one, we can construct a machine D whose domain is the previous set such that $D(t_w) = u$ if $U_2(w) = g(u)$. Now, if $U_2(w) = g(u)$, then

$$H_D(u) \leq \lceil \log_i(2) \cdot |w|_2 \rceil \leq \\ \log_i(2) \cdot |w|_2 + 1 \leq \log_i(2) \cdot H_2(g(u)) + d.$$

So we apply the invariance theorem to get a constant d' such that $\log_2(i) \cdot H_i(u) \leq \log_2(i) \cdot H_D(u) + d'$, hence

$$\log_2(i) \cdot H_i(u) \leq H_2(g(u)) + d + d'.$$

The constant $c_2 = d + d'$ satisfies equation (3). □

In [2], equation (1) was given as $\left|\delta_g(u) - \lceil \log_2 i \rceil \cdot \delta_i(u)\right| \leq d$. Theorem 2 gives a similar result for δ, hence $\left|\delta_g(u) - \log_2(i) \cdot \delta_i(u)\right| \leq c + 1$, where c is the constant of the theorem. In the proof, we supposed that $A = X_i^*$ but it is still valid with a proper subset of X_i^*.

Corollary 1 is important for the generalization of δ_g that is presented in Section 4. It is the same kind of result as Theorem 2, but applied to two Gödel numberings.

Corollary 1. [2, Corollary 4.5] Let $A \subseteq X_i^*$ be c.e. and $g, g' : A \to B^*$ be two Gödel numberings. Then, there effectively exists a constant c (dependent upon U_2, g, and g') such that for all $u \in A$ we have

$$|H_2(g(u)) - H_2(g'(u))| \leq c. \tag{4}$$

In order to have a complete formal proof of Theorem 3, we need to bound the complexity of the set of theorems $\mathcal{T}$ that a theory $\mathcal{F}$ proves. Such is the aim of Lemma 1.

Lemma 1. Let $\mathcal{F}$ be a finitely specified, arithmetically sound (i.e., each proven sentence is true), consistent theory strong enough to formalize arithmetic, and denote by $\mathcal{T}$ its set of theorems written in the alphabet X_i. Then for every $x \in \mathcal{T}$,

$$\frac{1}{2} \cdot |x|_i + O(1) \leq H_i(x) \leq |x|_i + O(1).$$

https://doi.org/10.25088/ComplexSystems.18.4.403

Proof. For the upper bound, it is sufficient to give a way to describe those theorems using descriptions not greater than their lengths, and which ensure that the computer used is self-delimiting. We first note that a theorem in $\mathcal{T}$ is a special well-formed formula. The bound we give is valid for the set of all the well-formed formulas. Consider the following program C: on its input x, C tests if x is a well-formed formula. C outputs x if it is well-formed, or enters an infinite loop if it is not.

This program has to be modified because its domain is not prefix-free. The idea here is to append a marker at the end of the input that appears only at the end of the words. In that way, if x is a prefix of y, then the end marker has to appear in y. As the marker can only appear at the end of y, then $x = y$ to ensure that the domain is prefix-free. We now have to define an end marker. It is sufficient to take an ill-formed formula. More precisely, we need a formula y such that for every well-formed formula x, $x\,y$ is ill-formed, and for every $z \in X_i^*$, $x\,y\,z$ is also ill-formed. For instance, we can take $y = ++$, where the symbol + is interpreted as the addition of natural numbers. There are in all formal systems plenty of possibilities for this y (another choice could be (+, for instance, or any ill-formed formula with parentheses around). In the following, y represents such a fixed ill-formed formula.

The new machine C works as follows: on an input z, C checks if $z = x\,y$ with a certain x. If the case arises, C checks if x is a well-formed formula, and if it is then outputs x. In all other cases, C diverges. Now we have a new machine C whose domain is prefix-free, such that $H_C(x) \leq |x|_i + |y|_i$. By the invariance theorem, we get a constant c such that $H_i(x) \leq |x|_i + c$.

We now prove the lower bound; that is, that the complexity of a theorem has to be greater than one half of its length, up to a constant. The idea is the following: If we consider a sentence x of the set of theorems $\mathcal{T}$, then it may contain some variables that cannot be compressed. More precisely, because we can work with many variables, it is not possible for each variable to be represented by a word that has a small complexity. To formalize the idea, we have to define what the variables are in our formal language. Consider that the variables are created as follows. A variable is denoted by a special character, say v, indicating that it is a variable, and then a binary number is written to identify each variable. This number is called the *identifier* of the variable. In the following, we denote by v_n the variable identified by the integer n.

Now, we have to consider the formulas defined by

$$\varphi(m, n) \equiv \exists\, v_m \,\exists\, v_n\, (v_m = v_n).$$

https://doi.org/10.25088/ComplexSystems.18.4.403

We suppose that m and n are random strings, that is, $H_i(m) \geq |m|_i + O(1)$ and $H_i(n) \geq |n|_i + O(1)$. Furthermore, we suppose that $H(m, n) \geq |m|_i + |n|_i + O(1)$, in other words that m and n together are random. We can suppose that such words do exist. Then

$$H_i(\varphi(m, n)) \geq H_i(m) + H_i(n) + O(1) \geq$$
$$|m|_i + |n|_i + O(1) \geq \frac{1}{2} \cdot |\varphi(m, n)|_i + O(1).$$

Thus, we obtained the lower bound. □

Improving the bounds in Lemma 1 seems to be difficult. A preliminary work would be to define exactly what is accepted as a formal language.

Theorem 3 is the formal version of Chaitin's heuristic principle. The substance of the proof comes from previous results.

Theorem 3. [2, Theorem 4.6] Consider a finitely specified, arithmetically sound (i.e., each proven sentence is true), consistent theory that is strong enough to formalize arithmetic, and denote by $\mathcal{T}$ its set of theorems written in the alphabet X_i. Let g be a Gödel numbering for $\mathcal{T}$. Then, there exists a constant N, which depends upon U_i, U_2, and $\mathcal{T}$ such that $\mathcal{T}$ contains no x with $\delta_g(x) > N$.

Proof. By Lemma 1, for every $x \in \mathcal{T}$, $\delta_i(x) \leq c$. Using Theorem 2, there exists a constant N such that for every $x \in \mathcal{T}$, $\delta_g(x) \leq N$. □

The δ_g measure is also useful for proving a probabilistic result about independent statements. Indeed, we can prove that the probability of a true statement of length n to be provable tends to zero when n tends to infinity.

Proposition 1. [2, Proposition 5.1] Let $N > 0$ be a fixed integer, $\mathcal{T} \subset X_i^*$ be c.e., and $g : \mathcal{T} \to B^*$ be a Gödel numbering. Then,

$$\lim_{n\to\infty} i^{-n} \cdot \text{card}\left\{x \in X_i^* : |x|_i = n,\, \delta_g(x) \leq N\right\} = 0. \qquad (5)$$

We do not give a proof of Proposition 1 because it is essentially technical and can be found in [2]. In Section 5, the proof of Proposition 5 uses the same arguments and differs only by details. Now we can express the probabilistic result about independent statements.

Theorem 4. [2, Theorem 5.2] Consider a consistent, sound, finitely specified theory strong enough to formalize arithmetic. The probability that a true sentence of length n is provable in the theory tends to zero when n tends to infinity.

The proof of Theorem 4 can be found in [2, p. 11].

https://doi.org/10.25088/ComplexSystems.18.4.403

4. Acceptable Complexity Measures

The function δ_g is our model for building the notion of acceptable complexity measures of theorems. Toward this end, we first define what a *builder* is, and then the properties it has to verify in order to be called "acceptable". An acceptable complexity measure of theorems will then be a complexity measure built via an acceptable builder.

Definition 1. For a computable function $\hat{\rho}_i : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$, we define the *complexity measure builder* ρ by

$$\begin{aligned} \rho : G &\to \left[X_i^* \to \mathbb{Q}\right] \\ g &\mapsto \left[u \mapsto \hat{\rho}_i\left(H_2\left(g\left(u\right)\right), \left|u\right|_i\right)\right]. \end{aligned}$$

The function $\hat{\rho}_i$ is called the *witness* of the builder. In the following, we will use $\rho_g(u)$ instead of $\rho(g)(u)$.

Now we define three properties that a builder has to verify in order to be acceptable. We recall that $\mathcal{F}$ denotes a theory that satisfies the hypothesis of Gödel's incompleteness theorem, and $\mathcal{T}$ its set of theorems.

Definition 2. A builder ρ is said to be *acceptable* if for every g, the measure ρ_g verifies the three following conditions:

1. For every theory $\mathcal{F}$, there exists an integer $N_{\mathcal{F}}$ such that if $\mathcal{F} \vdash x$, then $\rho_g(x) < N_{\mathcal{F}}$.
2. For every integer N,

$$\lim_{n\to\infty} i^{-n} \cdot \operatorname{card}\left\{x \in X_i^* : |x|_i = n \text{ and } \rho_g(x) \leq N\right\} = 0.$$

3. For every Gödel numbering g', there exists a constant c such that for every string $u \in X_i^*$, $\left|\rho_g(u) - \rho_{g'}(u)\right| \leq c$.

Condition 1 is simply the formal version of Chaitin's heuristic principle. Condition 2 corresponds to Proposition 1 and eliminates trivial measures. Finally, condition 3 ensures the independence on the way the theorems are written. In other words, conditions 1, 2, and 3 ensure that an acceptable complexity measure satisfies Theorem 3, Proposition 1, and Corollary 1, respectively.

Proposition 2 will be useful in the following. It is a weaker version of condition 1 and is used to prove that a measure is not acceptable, and more precisely that it does not satisfy this first property.

https://doi.org/10.25088/ComplexSystems.18.4.403

Proposition 2. Let ρ_g be an acceptable complexity measure. Then there exists an integer N such that for every integer $M \geq N$, the set

$$\left\{x \in X_i^* : \rho_g(x) \leq M\right\} \tag{6}$$

is infinite.

Proof. We consider a theory $\mathcal{F}$ and the integer $N_{\mathcal{F}}$ given by condition 1 in Definition 2. Clearly, $\mathcal{F}$ can prove an infinite number of theorems, such as "$n = n$" for all integer n. All of them have by condition 1 a complexity bounded by $N_{\mathcal{F}}$. If $\mathcal{T}$ is the set of theorems that $\mathcal{F}$ proves, then

$$\mathcal{T} \subset \left\{x \in X_i^* : \rho_g(x) \leq N_{\mathcal{F}}\right\}.$$

As $\mathcal{T}$ is infinite, so is the set in the proposition, and it remains true for every $M \geq N_{\mathcal{F}}$. □

We now prove that the δ_g complexity is an acceptable complexity measure. This result is natural because the notion of an acceptable complexity measure was built to generalize δ_g.

Proposition 3. The function δ_g is an acceptable complexity measure.

Proof. The δ_g function we defined plays the role of ρ_g. We have to provide an acceptable builder. We define

$$\hat{\delta}_i(x, y) = x - \left\lceil \log_2(i) \cdot y \right\rceil$$

to play the role of $\hat{\rho}_i$. Then $\delta_g(x) = \hat{\delta}_i\left(H_2(g(x)), |x|_i\right)$.

In fact, the properties of δ_g proved in [2] are exactly what we need here. One can easily check that condition 1 is ensured by Theorem 3, condition 2 by Proposition 1, and condition 3 by Corollary 1. □

The goal of defining an acceptable builder and an acceptable measure is to study complexities other than δ_g. Example 1 proves that the program-size complexity is not acceptable. This result, even though it is plain, is very important. Indeed, it justifies the need to define other complexity measures.

Example 1. A first natural complexity to study is the program-size complexity. There is no difficulty in verifying that H is a complexity measure. Formally, we have to define $\hat{\rho}_i(x, y) = x$ such that $H_2(g(x)) = \hat{\rho}_i(x, |x|_i)$. We study the properties of the builder $g \mapsto [x \mapsto H_2(g(x))]$. Here is how it behaves with the three conditions from Definition 2.

https://doi.org/10.25088/ComplexSystems.18.4.403

1. Condition 1 cannot be verified. Indeed, we note that

   $$\text{card}\left\{x \in X_i^* : H_2\big(g(x)\big) \leq N\right\} \leq \text{card}\left\{y \in X_2^* : H_2\big(y\big) \leq N\right\} \leq 2^N.$$

   If the condition was verified, the set of theorems $\mathcal{T}$ proved by $\mathcal{F}$ would be bounded by 2^N, which is a contradiction.

2. On the contrary, condition 2 is obviously verified. Indeed, as

   $$\text{card}\left\{x \in X_i^* : H_2\big(g\,(x)\big) \leq N\right\} \leq 2^N,$$

   $$\left\{x \in X_i^* : |x|_i = n \,\text{and}\, H_2\big(g\,(x)\big) \leq N\right\} = \emptyset \text{ for large enough } n.$$

3. Condition 3 corresponds exactly to Corollary 3, and is verified.

As the program-size complexity cannot be used, we try to find other ways to better reflect the intrinsic complexity. That is why we use the length of the strings to alter the complexity. It seems natural that the longest strings are also the most difficult to describe (we have to be very careful with this statement because it is not really true). In Section 5, we give two more examples of unacceptable builders.

5. Independence of the Three Conditions

The aim of this section is to prove that the conditions 1, 2, and 3 in Definition 2 are independent from each other. Toward this end, we give two new examples of unacceptable builders, each of which exactly satisfy two conditions in Definition 2. Furthermore, they give us an idea of the ingredients needed to build an acceptable complexity builder. In particular, they show us that a builder shall neither be too small nor too big.

Example 2. Let $\hat{\rho}_i^1$ be the function defined by $\hat{\rho}_i^1\,(x, y) = x\,/\,y$ if $y \neq 0$ and 0 otherwise. It defines a builder ρ^1 and for every Gödel numbering g, we can define ρ_g^1 by

$$\rho_g^1\,(x) = \begin{cases} \dfrac{H_2\,(g\,(x))}{|x|_i}, & \text{if}\, x \neq \lambda, \\ 0, & \text{otherwise.} \end{cases}$$

We show later that ρ^1 is too small of a complexity measure. In fact, it is even bounded. In order to avoid this problem, we define ρ^2 by dividing the program-size complexity by the logarithm of the length.

https://doi.org/10.25088/ComplexSystems.18.4.403

Example 3. We consider $\hat{\rho}_i^2$ defined by

$$\hat{\rho}_i^2(x, y) = \begin{cases} \dfrac{x}{\lceil \log_i y \rceil}, & \text{if } y > 1, \\ 0, & \text{otherwise.} \end{cases}$$

The corresponding builder applied with a Gödel numbering g defines the function

$$\rho_g^2(x) = \begin{cases} \dfrac{H_2(g(x))}{\lceil \log_i |x|_i \rceil}, & \text{if } |x|_i > 1, \\ 0, & \text{otherwise.} \end{cases}$$

In order to make the proofs easier, we introduce a new function for each builder that has been defined. Those functions make no use of Gödel numberings and are the equivalents of δ_i for ρ^1 and ρ^2. They can help us in the proofs because we prove first that they are equal to the complexity measures, up to a constant. For ρ^1, we define ρ_i^1 by $\rho_i^1(x) = H_i(x) / |x|_i$ if $x \neq \lambda$ and 0 otherwise. And similarly, for ρ^2, we define $\rho_i^2(x) = H_i(x) / \lceil \log_i |x|_i \rceil$ if $|x|_i > 1$ and 0 otherwise.

Lemma 2. Let $A \subseteq X_i^*$ be c.e. and $g : A \to B^*$ be a Gödel numbering. Then, there effectively exists a constant c (dependent upon U_i, U_2, and g) such that for all $u \in A$, we have

$$\left| \rho_g^j(u) - \log_2(i) \cdot \rho_i^j(u) \right| \leq c, \tag{7}$$

$j = 1, 2$.

Proof. We first note that this difference is null for $u = \lambda$ in the case $j = 1$, and for $|u|_i \leq 1$ in the case $j = 2$. In the following, we suppose that $|u|_i > 0$ (for $j = 1$) or $|u|_i > 1$ (for $j = 2$).

Theorem 2 states that

$$\left| H_2(g(u)) - \log_2(i) \cdot H_i(u) \right| \leq c.$$

We now just have to divide the whole inequality by $|u|_i \geq 1$ to obtain equation (7) with $j = 1$ and by $\lceil \log_i |u|_i \rceil$, which is not less than one but for finitely many u to obtain the result with $j = 2$. □

This result allows us to work with much easier forms of the complexity functions. We now study the properties that ρ_g^1 and ρ_g^2 satisfy. As a corollary of Lemma 2, we can note that both of the measures satisfy condition 3.

https://doi.org/10.25088/ComplexSystems.18.4.403

Proposition 4. The function ρ_g^1 verifies condition 1 in Definition 2, but does not verify condition 2.

Lemma 3. There exists a constant M such that for all $x \in X_i^*$, $\rho_g^1(x) \leq M$.

Proof. The result is plain for $x = \lambda$. We now suppose that $|x|_i > 0$. In view of [8, Theorem 3.22], there exist two constants α and β such that for all $x \in X_i^*$,

$$H_i(x) \leq |x|_i + \alpha \cdot \log_i |x|_i + \beta,$$

so, for $x \neq \lambda$,

$$\rho_i^1(x) \leq 1 + \alpha \cdot \frac{\log_i |x|_i}{|x|_i} + \beta \cdot \frac{1}{|x|_i} \cdot$$

Because $\log_i(|x|_i) / |x|_i \leq 1$ for every $x \neq \lambda$, then

$$\rho_i^1(x) \leq 1 + \alpha + \beta.$$

Furthermore, Lemma 2 states that for every x, we have

$$\rho_g^1(x) \leq c + \log_2(i) \cdot \rho_i^1(x) \leq c + \log_2(i) \cdot (1 + \alpha + \beta).$$

Accordingly, $M = \lceil c + \log_2(i) \cdot (1 + \alpha + \beta) \rceil$ satisfies the statement of the lemma. □

Proof. [Proof of Proposition 4] Condition 1 is obvious since Lemma 3 tells us that the bound is valid for every sentence x, not only provable ones. On the contrary, the fact that ρ_g^1 is bounded by M implies that for $N \geq M$, the set $\{x \in X_i^* : |x|_i = n \text{ and } \rho_g^1(x) \leq N\}$ is the set X_i^n. Hence the limit of condition 2 is 1 instead of 0. □

This proof shows us that an acceptable complexity measure cannot be too small (ρ^1 is even bounded). We now show, thanks to the complexity measure ρ^2, that an acceptable complexity measure cannot be too big, either.

Proposition 5. The function ρ_g^2 verifies condition 2 in Definition 2, but does not verify condition 1.

https://doi.org/10.25088/ComplexSystems.18.4.403

Proof. We begin with the proof of condition 2 for ρ^2. Theorem 2 allows us to consider ρ_i^2 instead of ρ_g^2, with a new constant $\lceil (N+c) / \log_2(i) \rceil$. Indeed, it states that $\rho_g^2(x) \geq \log_2(i) \cdot \rho_i^2(x) - c$, and consequently

$$\left\{x \in X_i^n : \rho_g^2(x) \leq N\right\} \subseteq \left\{x \in X_i^n : \rho_i^2 \leq \left\lceil \frac{N+c}{\log_2(i)} \right\rceil\right\}.$$

In order to avoid too many notations, we still denote this constant by N.

First, we note that

$$\left\{x \in X_i^n : \rho_i^2(x) \leq N\right\} = \left\{x \in X_i^n : \exists\ y \in X_i^{\leq N \cdot \lceil \log_i n \rceil},\ U_i(y) = x\right\}.$$

Translating in terms of cardinals, we obtain

$$\begin{aligned}&\operatorname{card}\left\{x \in X_i^n : \rho_i^2(x) \leq N\right\} \leq \\ &\quad \operatorname{card}\left\{x \in X_i^n : \exists\ y \in X_i^{\leq N \cdot \lceil \log_i n \rceil},\ U_i(y) = x\right\} \leq \\ &\quad \operatorname{card}\left\{y \in X_i^{\leq N \cdot \lceil \log_i n \rceil} : \left| U_i(y) \right| = n\right\} \leq \\ &\quad \operatorname{card}\left\{y \in X_i^{\leq N \cdot \lceil \log_i n \rceil} : U_i(y) \text{ halts.}\right\} \leq \\ &\quad \sum_{k=1}^{N \cdot \lceil \log_i n \rceil} \underbrace{\operatorname{card}\left\{y \in X_i^k : U_i(y) \text{ halts.}\right\}}_{r_k}\end{aligned}$$

We extend these inequalities to the limit when n tends to infinity:

$$\begin{aligned}&\lim_{n \to \infty} i^{-n} \cdot \operatorname{card}\left\{x \in X_i^n : \rho_g^2(x) \leq N\right\} \leq \lim_{n \to \infty} \sum_{k=1}^{N \cdot \lceil \log_i n \rceil} i^{-n} \cdot r_k \leq \\ &\quad \lim_{n \to \infty} i^{N \cdot \lceil \log_i n \rceil - n} \cdot \sum_{k=1}^{N \cdot \lceil \log_i n \rceil} i^{-N \cdot \lceil \log_i n \rceil} \cdot r_k.\end{aligned}$$

We note that

$$\lim_{n \to \infty} \sum_{k=1}^{N \cdot \lceil \log_i n \rceil} i^{-N \cdot \lceil \log_i n \rceil} \cdot r_k = \lim_{m \to \infty} \sum_{k=1}^{m} i^{-m} \cdot r_k.$$

https://doi.org/10.25088/ComplexSystems.18.4.403

Now,

$$\lim_{m\to\infty} \frac{\sum_{k=1}^{m+1} r_k - \sum_{k=1}^{m} r_k}{i^{m+1} - i^m} = \frac{i}{i-1} \cdot \lim_{m\to\infty} i^{-m} \cdot r_m = 0.$$

The last inequality comes from Kraft's inequality:

$$\sum_{m=1}^{\infty} i^{-m} \cdot r_m \leq 1.$$

So we can apply the Stolz-Cesàro theorem to ensure that

$$\lim_{n\to\infty} \sum_{k=1}^{N\cdot\lceil \log_i n\rceil} i^{-N\cdot\lceil \log_i n\rceil} \cdot r_k = 0. \tag{8}$$

On the other hand,

$$i^{N\cdot\lceil \log_i n\rceil - n} = 0. \tag{9}$$

We just have to combine equations (8) and (9) to obtain condition 2.

Now, it remains to prove that condition 1 is not verified. Toward this end, we suppose that condition 1 holds. We denote with $\mathcal{T}$ the set of theorems that $\mathcal{F}$ proves. Note first that

$$\begin{aligned} &\text{card}\left\{x \in X_i^* : |x|_i = n \text{ and } H_2(g(x)) \leq N \cdot \lceil \log_i n\rceil\right\} \leq \\ &\quad \text{card}\left\{y \in B^* : H_2(y) \leq N \cdot \lceil \log_i n\rceil\right\} \leq \\ &\quad 2^{N\cdot\lceil \log_i n\rceil} \leq 2^{N\cdot(\log_i n + 1)} \leq 2^N \cdot n^{N\cdot \log_i 2}. \end{aligned} \tag{10}$$

So, if condition 1 holds for all $x \in \mathcal{T}$, we have

$$\text{card}\,\{x \in \mathcal{T} : x \in \mathcal{T}\; |x| = n\} \leq \alpha\, n^{\beta N}, \tag{11}$$

for every integer n, where α and β come from equation (10).

Now consider this set of formulas:

$$\Phi_k = \left\{Q_0\, x_0\, Q_1\, x_1 \ldots Q_k\, x_k \bigwedge_{l=0}^{k} \left(x_l = x_l\right) : Q_l \in \{\forall\, , \exists\}\right\}.$$

Each formula $\varphi \in \Phi_k$ is true, and all formulas have the same length $n_k = O(k)$. Furthermore, $\text{card}\,\Phi_k = 2^k$.

All of those formulas belong to the predicate logic, so all of them are provable in $\mathcal{F}$, that is to say, they belong to $\mathcal{T}$. As we can take k as big as wanted, we can also have n_k as big as wanted.

Now we have, for arbitrarily large n, $2^{O(n)}$ formulas of length n that belong to $\mathcal{T}$. That contradicts equation (11), and so, condition 1 is false. □

https://doi.org/10.25088/ComplexSystems.18.4.403

We can now prove that conditions 1, 2, and 3 in Definition 2 are independent from each other. We already know that an acceptable complexity builder does exist for δ_g. Thus it is sufficient to prove that for each condition a builder exists that does not satisfy it but does satisfy both other conditions.

Theorem 5. Each condition in Definition 2 is independent from the other ones.

Proof. The measure builder ρ^1 is an example measure that satisfies both conditions 1 and 3 but not 2 while ρ^2 does not satisfy 1 but does satisfy 2 and 3. To prove the complete independence of the three conditions, it remains to prove that a complexity measure builder can satisfy both conditions 1 and 2 without satisfying 3.

In fact, our proof does not exactly follow the given scheme. It is still not known if all the complexity measure builders satisfy condition 3, or if some exist that do not satisfy it. Thus, the proof is built as follows. We prove that either all complexity builders satisfy condition 3, or there exists at least one complexity builder satisfying 1 and 2 without satisfying 3. We also give the exact question the answer of which would make the choice between both the possibilities.

Let g and g' be two Gödel numberings from X_i^* to X_2^*, and ρ_g and $\rho_{g'}$ two complexity measures built with the same builder. The question is whether $H_2(g(x)) = H_2(g'(x))$ for all but finitely many $x \in X_i^*$ or if there exists an infinite sequence $(x_n)_{n\in\mathbb{N}}$ such that $H_2(g(x_n)) \neq H_2(g'(x_n))$ for all n. Suppose that the first case holds; then for all but finitely many $x \in X_i^*$,

$$\rho_g(x) = \hat{\rho}_i\big(H_2(g(x)), |x|_i\big) = \hat{\rho}_i\big(H_2(g'(x)), |x|_i\big) = \rho_{g'}(x).$$

Consequently,

$$c = \max\left\{|H_2(g(x)) - H_2(g'(x))| : x \in X_i^*\right\} < \infty,$$

and the builder ρ does satisfy condition 3.

We suppose now that the second case holds, meaning that infinitely many strings exist $x \in X_i^*$ such that $H_2(g(x)) \neq H_2(g'(x))$. We consider the acceptable complexity measure δ_g and define the measure ρ_g by $x \mapsto \delta_g(x)^2$. More formally, if we denote by $\hat{\delta}_i$ the witness of the builder δ, we define the builder ρ via the witness $\hat{\rho}_i = \hat{\delta}_i^2$. We now consider the behavior of this function with the three conditions:

https://doi.org/10.25088/ComplexSystems.18.4.403

1. Because δ_g is acceptable, there exists $N_{\mathcal{F}}$ such that if $\mathcal{F} \vdash x$, then $\delta_g(x) \leq N_{\mathcal{F}}$. Then it is clear that $\rho_g(x) \leq N_{\mathcal{F}}^2$. So condition 1 is verified.
2. For an integer $N \geq 1$, if $\rho_g(x) \leq N$, then we also have $\delta_g(x) \leq N$ and get the following:

   $$\left\{x \in X_i^* : |x|_i = n \text{ and } \rho_g(x) \leq N\right\} \subset \left\{x \in X_i^* : |x|_i = n \text{ and } \delta_g(x) \leq N\right\}.$$

   Consequently,

   $$\lim_{n\to\infty} i^{-n} \cdot \text{card}\left\{x \in X_i^* : |x|_i = n \text{ and } \rho_g(x) \leq N\right\} \leq \lim_{n\to\infty} i^{-n} \cdot \text{card}\left\{x \in X_i^* : |x|_i = n \text{ and } \delta_g(x) \leq N\right\} = 0.$$

   So condition 2 is also verified.
3. We first note that

   $$\rho_g(x) - \rho_{g'}(x) = \delta_g(x)^2 - \delta_{g'}(x)^2 = \left(H_2(g(x)) - \lceil \log_2(i) \cdot |x|_i \rceil\right)^2 - \left(H_2(g'(x)) - \lceil \log_2(i) \cdot |x|_i \rceil\right)^2 = \left(H_2(g(x))^2 - H_2(g'(x))^2\right) - 2 \cdot \lceil \log_2(i) \cdot |x|_i \rceil \left(H_2(g(x)) - H_2(g'(x))\right).$$

   We know from Corollary 1 that $\left(H_2(g(x)) - H_2(g'(x))\right)$ is bounded. Thus, we only need to prove that $\left|H_2(g(x))^2 - H_2(g'(x))^2\right|$ is unbounded, and we will be able to conclude that condition 3 is not satisfied by ρ. Suppose that it is bounded by an integer N. As we have supposed that there exist infinitely many $x \in X_i^*$ such that $H_2(g(x)) \neq H_2(g'(x))$, then there exists for every integer M a string x such that $H_2(g(x)) > H_2(g'(x)) > M$. Then

   $$H_2(g(x))^2 - H_2(g'(x))^2 = \left(H_2(g(x)) - H_2(g'(x))\right) \cdot \left(H_2(g(x)) + H_2(g'(x))\right) > 1 \cdot (2 \cdot M) = 2M.$$

   We can impose here without any loss of generality that $H_2(g(x)) > H_2(g'(x))$ because the converse situation would be equivalent. We can also conclude, using an integer $M > N/2$, that this bound cannot exist, meaning that condition 3 is not satisfied. □

6. Form of the Acceptable Complexity Measures

The aim of this section is to give some conditions that a complexity measure has to verify to be acceptable. More precisely, we study some conditions that a builder, and in particular its witness, has to verify such that the complexity measures it builds are acceptable. We restrict

https://doi.org/10.25088/ComplexSystems.18.4.403

our study to particular witnesses, such as linear functions in both variables, or functions defined by

$$\hat{\rho}_i(x, y) = \frac{x}{f(y)}$$

where f is a computable function.

Our first result shows a kind of stability of the acceptable complexity measures and makes the following proofs easier.

Proposition 6. Let ρ_g be an acceptable complexity measure, and $\alpha, \beta \in \mathbb{Q}$ such that $\alpha > 0$. Then $\alpha \cdot \rho_g + \beta$ is also an acceptable complexity measure.

Proof. Condition 1 in Definition 2 remains true with a new constant $\alpha \cdot N + \beta$ instead of N. In the same way,

$$\left\{x \in X_i^* : |x|_i = n \text{ and } \alpha \cdot \rho_g(x) + \beta \leq N\right\} \subseteq \left\{x \in X_i^* : |x|_i = n \text{ and } \rho_g(x) \leq \left\lceil \frac{N - \beta}{\alpha} \right\rceil\right\},$$

hence condition 2 is verified. Now, if we consider two Gödel numberings g and g',

$$\left|\left(\alpha \cdot \rho_g(x) + \beta\right) - \left(\alpha \cdot \rho_{g'}(x) + \beta\right)\right| = \alpha \cdot \left|\rho_g(x) - \rho_{g'}(x)\right| \leq \alpha \cdot c,$$

which proves that condition 3 is retained. □

We start by studying the witnesses that are bilinear functions and obtain a partial result. However, as discussed after Lemma 1, this result is not likely to be improved without a complete study of the definition of the formal languages.

Proposition 7. Let f be a bilinear function of two variables such that $\hat{\rho}_i$ defined by $\hat{\rho}_i(x) = \lfloor f(x) \rfloor$ is computable. If $\hat{\rho}_i$ defines an acceptable complexity measure, then there exist a, b, and ϵ, $a > 0$ and $1/2 \leq \epsilon \leq 1$, such that

$$\hat{\rho}_i(x, y) = \left\lfloor a \cdot \left(x - \epsilon \cdot \log_2(i) \cdot y\right) + b \right\rfloor.$$

Proof. We consider any function that satisfies the hypothesis. Then there exist α, β, and γ such that

$$\hat{\rho}_i(x, y) = \lfloor \alpha x - \beta y + \gamma x y \rfloor.$$

Proposition 6 allows us to fix $\hat{\rho}_i(0, 0) = 0$. Of course, it would be equivalent to consider $\alpha x + \beta y + \gamma x y$, but the chosen version simplifies the notation. Let β' be such that $\beta = \beta' \cdot \log_2(i)$. The proof is done in several steps. We start by showing that at least one of α and γ

https://doi.org/10.25088/ComplexSystems.18.4.403

has to be different from zero, then that $\gamma = 0$. After that, we prove that $\alpha/2 \leq \beta' \leq \alpha$.

Suppose that $\alpha = \gamma = 0$. Then $\rho_g(x) = -\lceil \beta |x|_i \rceil$. If $\beta \leq 0$, then Proposition 2 is not verified by our complexity measure, and hence neither is condition 1. If $\beta \geq 0$, it is obvious that condition 2 cannot hold true.

Then, we use condition 1 and consider the set

$$\{x \in X_i^* : |x|_i = n \text{ and } \rho_g(x) \leq N\} \subseteq \left\{x \in X_i^* : |x|_i = n \text{ and } H_2(g(x)) \leq \left\lceil \frac{\beta n + N + 1}{\gamma n + \alpha} \right\rceil \right\}.$$

Furthermore,

$$\lim_{n \to \infty} \frac{\beta n + N + 1}{\gamma n + \alpha} = \begin{cases} \frac{\beta}{\gamma}, & \text{if } \gamma \neq 0; \\ \frac{N+1}{\alpha}, & \text{if } \gamma = \beta = 0; \\ \pm\infty, & \text{if } \gamma = 0 \text{ and } \beta \neq 0. \end{cases}$$

The only solution is the third one because in order to satisfy condition 1, this limit has to be infinite. Indeed, if it is finite, we can use the same proof as in Proposition 5 and conclude with a contradiction. So we know that $\gamma = 0$, and hence that $\alpha \neq 0$. We can now say that α and β have the same sign, because the limit cannot be $-\infty$. Using Proposition 6, we can assume that $\alpha = 1$. Indeed, $\alpha < 0$ is not possible because of condition 2.

To make the remainder of the proof easier, we define an auxiliary measure as done in Sections 3 and 5 for δ, ρ^1, and ρ^2. Let ρ_i be defined by

$$\rho_i(x) = \lfloor H_i(x) - \beta' \cdot |x|_i \rfloor.$$

Applying Theorem 2, we get a constant c such that for every x,

$$\left| \rho_g(x) - \log_2(i) \cdot \rho_i(x) \right| \leq c.$$

We now use condition 2 to get more information on β', and hence β. We only know that $\beta' > 0$. We consider the set

$$\{x \in X_i^* : |x|_i = n \text{ and } \rho_g(x) \leq N\} \subseteq \{x \in X_i^* : |x|_i = n \text{ and } H_i(x) \leq \beta' \cdot n + N + c + 1\}.$$

If $\beta' > 1$, then for every constant d, if we choose n large enough, we have $\beta' \cdot n > n + d \cdot \log n$. And we can use the inequality $H_i(x) \leq |x|_i + O(\log_i |x|_i)$ (see [8, Theorem 3.22]) to conclude that the

https://doi.org/10.25088/ComplexSystems.18.4.403

given set is X_i^n. And so, condition 3 is not verified, because the limit is 1.

Using the lower bound in Lemma 1, we know that for every proven sentence x,

$$H_i(x) \geq \frac{1}{2} \cdot |x|_i.$$

Suppose that $\beta' < 1/2$. Then for every x such that $\mathcal{F} \vdash x$,

$$\rho_i(x) = \left(H_i(x) - \frac{1}{2} \cdot |x|_i\right) + \left(\frac{1}{2} - \beta'\right) \cdot |x|_i \geq \left(\frac{1}{2} - \beta'\right) \cdot |x|_i.$$

Thus, condition 1 cannot be verified. □

We study another kind of witness. Functions defined by

$$\hat{\rho}_i(x, y) = \frac{x}{f(y)}$$

where f is a computable function may be interesting because they are the only reasonable candidates for being witnesses of multiplicative complexity measures. Indeed, a complexity of the form $H_2(g(x)) \cdot |x|_i$ has no chance of satisfying the desired properties. Unfortunately, such functions never define acceptable measures.

Proposition 8. Let f be a computable function, and $\hat{\rho}_i$ defined by

$$\hat{\rho}_i(x, y) = \frac{x}{f(y)}.$$

Then the complexity measure builder with the witness $\hat{\rho}_i$ cannot satisfy conditions 1 and 2 at the same time.

Proof. Suppose that $\rho_g(x) = \hat{\rho}_i\left(H_2(g(x)), |x|_i\right)$ satisfy condition 1. Then consider the set

$$\left\{x \in X^* : |x|_i = n \text{ and } H_2(g(x)) \leq N \cdot f(n)\right\}.$$

Its cardinal is at most $2^{N \cdot f(n)}$. Furthermore, this set contains the set of all sentences in $\mathcal{T}$ with length n. Hence,

$$\text{card}\left\{x \in \mathcal{T} : |x|_i = n\right\} \leq 2^{N \cdot f(n)}. \tag{12}$$

Now we give a lower bound to this cardinal. The proof of Proposition 5 shows that this cardinal is greater than $2^{O(n)}$. Accordingly, there exists a constant c such that

$$\text{card}\left\{x \in \mathcal{T} : |x|_i = n\right\} \geq 2^{c \cdot n}. \tag{13}$$

https://doi.org/10.25088/ComplexSystems.18.4.403

We also obtain that $2^{c \cdot n} \leq 2^{N \cdot f(n)}$ and conclude that

$$f(n) \geq \frac{c}{N} \cdot n. \tag{14}$$

We now follow the proof made earlier to show that ρ_g^1 does not satisfy condition 2. We can define

$$\rho_i(x) = \frac{H_i(x)}{f(|x|_i)},$$

and prove as for ρ^1 and ρ^2 that there exists a constant d such that

$$\left|\rho_g(x) - \log_2(i) \cdot \rho_i(x)\right| \leq d.$$

The proof of Lemma 2 is still valid here. In the same way, we extend Lemma 3 to ρ_g, namely there exists a constant M such that ρ_g is bounded by M. Considering ρ_g instead of ρ_g^1 has just such an influence on the value of the constant M.

Now, we have to note that for $N \geq M$, the set $\left\{x \in X_i^* : |x|_i = n \text{ and } \rho_g(x) \leq N\right\}$ is the set X_i^n to conclude that condition 2 is not verified. □

7. Concluding Remarks

In this paper we studied the δ_g complexity function defined by Calude and Jürgensen in [2]. This study led us to modify the definition of δ_g in order to correct some of the proofs. Then, we have been able to propose a definition for acceptable complexity measures for theorems that capture the main properties of δ_g. After studying some complexity measures, we showed that the conditions of acceptability are quite hard to complete. Yet, the definition seems to be robust enough to allow some investigations to find other natural acceptable complexity measures.

Here are some remaining open questions.

- Can we improve the bounds of Lemma 1? This question is interesting not only for improving Proposition 7 but also for itself. How simple are the well-formed formulas, and in other words, to what extent can we use their great regularities to compress them? Yet, as already discussed, this question needs to be better defined. In particular, the definition of the formal languages has to be investigated. The answer seems to be very dependent on the considered language.

https://doi.org/10.25088/ComplexSystems.18.4.403

- Do there exist some acceptable complexity measures that are very different from δ_g? The idea here is to find some measures that go further in investigating the roots of unprovability.
- In view of the proof of Theorem 5, if we have two Gödel numberings g and g', does the equality $H_2(g(x)) = H_2(g'(x))$ hold for all but finitely many x or are those two quantities infinitely often different from each other?

Those few questions are added to the ones expressed in [2]. The goal of finding new acceptable complexity measures is to have more tools for trying to answer their questions, as the existence of independent sentences of small complexity.

Acknowledgments

Special thanks are due to Cristian Calude without whom this paper would have never existed. His very helpful comments, corrections, and improvements, as well as his hospitality, made my stay in Auckland much nicer than I could have hoped for. Thanks are also due to André Nies for his comments and ideas. In particular, he gave us the lower bound in Lemma 1.

References

[1] K. Gödel, "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I," *Monatshefte für Mathematik*, **38**(1), 1931 pp. 173-198.

[2] C. Calude and H. Jürgensen, "Is Complexity a Source of Incompleteness?" *Advances in Applied Mathematics,* **35**(1), 2005 pp. 1-15.

[3] G. Chaitin, "Information-Theoretic Limitations of Formal Systems," *Journal of the ACM*, **21**(3), 1974 pp. 403-424.

[4] C. Calude, H. Jürgensen, and M. Zimand, "Is Independence an Exception?" *Applied Mathematics and Computation*, **66**(1), 1994 pp. 63-76.

[5] C. Calude, "Incompleteness: A Personal Perspective," *CDMTCS Research Report* **324**, 2008, pp. 14.

[6] A. Kolmogorov, "Three Approaches to the Quantitative Definition of Information," *International Journal of Computer Mathematics*, **2**(1-4), 1968 pp. 157-168.

[7] G. Chaitin, "A Theory of Program Size Formally Identical to Information Theory," *Journal of the ACM*, **22**(3), 1975 pp. 329-340.

[8] C. Calude, *Information and Randomness: An Algorithmic Perspective*, 2nd ed., New York: Springer-Verlag, 2002.

[9] M. Li and P. Vitányi, *An Introduction to Kolmogorov Complexity and Its Applications*, New York: Springer-Verlag, 1997.

https://doi.org/10.25088/ComplexSystems.18.4.403

[10] A. Turing, "On Computable Numbers, With an Application to the Entscheidungsproblem," *Proceedings of the London Mathematical Society*, **42**, 1936 pp. 230-265.

[11] M. Sipser, *Introduction to the Theory of Computation*, 2nd ed., Boston: Thomson Course Technology, 2006.

[12] C. Papadimitriou, *Computational Complexity*, Reading, MA: Addison-Wesley 1994.

[13] P. Odifreddi, *Classical Recursion Theory: The Theory of Functions and Sets of Natural Numbers*, Amsterdam: North-Holland, Vol. 1, 1989, Vol. 2, 1999.

https://doi.org/10.25088/ComplexSystems.18.4.403