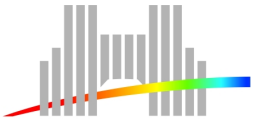


École Normale Supérieure de Lyon, France
Fundamental Computer Science Master, First Year

Acceptable Complexity Measures of Theorems

Bruno Grenet[†]
Supervisor: Cristian S. Calude[‡]

The University of Auckland, New Zealand
Te Whare Wānanga o Tāmaki Makaurau, Aotearoa

June - August 2008

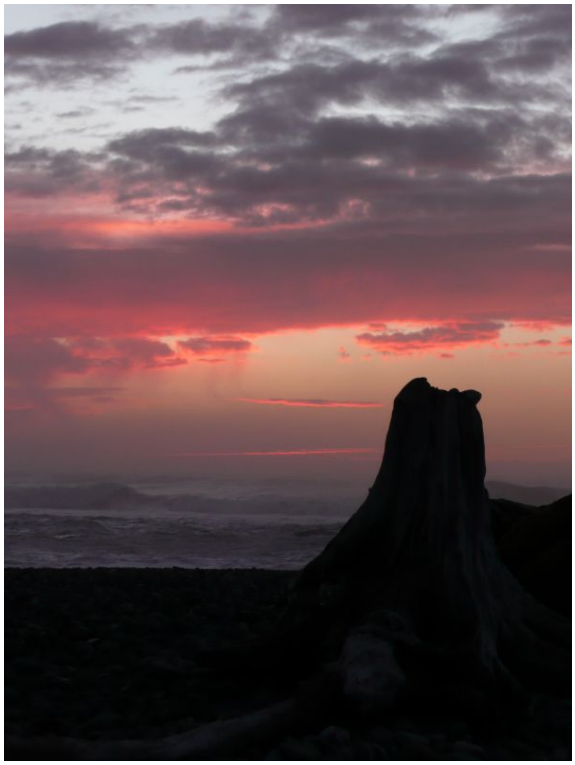
[†] http://perso.ens-lyon.fr/bruno.grenet/
[‡] http://www.cs.auckland.ac.nz/~cristian/

Abstract

In 1930, Gödel [7] presented in Königsberg his famous Incompleteness Theorem, stating that some true mathematical statements are unprovable. Yet, this result gives us no idea about those *independent* (that is, true and unprovable) statements, about their frequency, the reason they are unprovable, and so on. Calude and Jürgensen [4] proved in 2005 Chaitin's "heuristic principle" for an appropriate measure: *the theorems of a finitely-specified theory cannot be significantly more complex than the theory itself* (see [5]). In this work, we investigate the existence of other measures, different from the original one, which satisfy this "heuristic principle". At this end, we introduce the definition of *acceptable complexity measure of theorems*.

Résumé

En 1930, Gödel [7] présente à Königsberg son célèbre Théorème d'Incomplétude, spécifiant que certaines affirmations mathématiques sont indémontrables. Cependant, ce résultat ne nous donne aucune indication à propos de ces affirmations *indépendantes* (c'est-à-dire vraies mais indémontrables), sur leur fréquence, les raisons de leur indémontrabilité, *etc.* Calude and Jürgensen [4] ont prouvé en 2005 le « principe heuristique » de Chaitin pour une mesure de complexité appropriée : *les théorèmes d'une théorie finiment axiomatisable ne peuvent être significativement plus complexes que la théorie elle-même* (*cf* [5]). Dans ce rapport, nous étudions l'existence d'autres mesures, différentes de la mesure originale utilisée dans [4], qui satisfassent ce « principe heuristique ». A cette fin, nous introduisons la définition de *mesure acceptable de complexité des théorèmes*.



Gillepsie Beach, South Island

1 Introduction

In 1931, Gödel [7] presented in Königsberg his famous *(first) Incompleteness Theorem*, stating that some true mathematical statements are unprovable. More formally and in modern terms, it states the following:

> 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* (that is, true and unprovable) statements, about their frequency, the reason they are unprovable, and so on. Those questions of quantitative results about the independent statements have been investigated by Chaitin [5] in a first time, and then by Calude, Jürgensen and Zimand [2] and Calude and Jürgensen [4]. A state of the art is given in [3]. Those results 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 other ones are unprovable?

Chaitin [5] 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.* It was proven [4] that Chaitin's "heuristic principle" is valid for a 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 the next section) to calculate s (see [8, 6, 1, 9]). We consider the following computable variation of the program-size complexity:

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

This measure gives us some indications about the reasons of unprovability of certain statements. It would be very interesting to have other results in order to understand the Incompleteness Theorem. Among them, one can try to prove a kind of reverse of the theorem Calude and Jürgensen proved. Their theorem states that there exists a constant N such that any theory which satisfies the hypothesis of Gödel's Theorem cannot prove any statements x with $\delta(x) > N$. Another question of interest could be the following: Does there exist any independent statements with a low δ-complexity?

Those results are only 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 which satisfy this "heuristic principle" in order to be able to prove the remaining results. At this end, we introduce the notion of *acceptable complexity measure of theorems* which captures the important properties of δ. After studying the results of [4] about δ, we define the acceptable complexity measures. We study their properties, and try to find some other acceptable complexity measures, different from δ.

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

2 Prerequisites and notations

In the sequel, $\mathbb{N}$ 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 $\mathrm{card}(S)$. For every integer $i \geq 2$, we fix an alphabet X_i with i elements, X_i^* being the set of finite strings on X_i, including the empty string λ, and $|w|_i$ the length of the string $w \in X_i$.

We assume the reader is familiar with Turing machines processing strings [13] and with the basic notions of computability theory (see, for example [12, 11, 10]). We recall that a set is said computably enumerable (abbreviated c.e.) if it is the domain of a Turing machine, or equivalently if it can be algorithmically listed.

The complexity measures we study are *computable variation* of the *program-size complexity*. In order to define it, we define the *self-delimiting Turing machines*, shortly *machines*, which are Turing machines the domain of which is a prefix-free set. A set $S \subset X_i^*$ is said *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 $PROG_T = \{x \in X_i^* : T \text{ halts on } x\}$ 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 $\min(\varnothing) = \infty$. The Invariance Theorem ensures the effective existence of a so-called *universal* machine U_i which minimize 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 sequel, 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 numbering for a fixed language. In what follows, we consider theories which satisfy the hypothesis of Gödel Incompleteness Theorem, that is finitely-specified, sound and consistent theories strong enough to formalize arithmetic. The first condition means that the set of axioms of the theory is c.e.; soundness is the property that the theory only proves true sentences; consistency states that the theory is free of contradictions. We will generally denote by $\mathcal{F}$ such a theory, and by $\mathcal{T}$ the set of theorems that $\mathcal{F}$ proves.

3 The function δ_g

We present in this section the function δ_g and some results about it. It was defined in [4] and almost all the results come from this paper. Hence, complete proofs of the results can be found in it. Yet, there was a mistake in the paper, and we need to modify a bit the definition of δ_g. We have to adapt the proofs with the new definition. The transformations are essentially cosmetic in almost all the proofs so we give only sketches of them. For Theorem 3.2, there are a bit more than details to change, so we provide a complete proof of this result. Furthermore, we formally prove an assertion used in the proof of Theorem 3.5.

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 we study is not dependent on the way we write the theorems, we define the δ-complexity *induced by a Gödel numbering* g by[1]

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

where g is a Gödel numbering the domain of which is in X_i^*.

The first result comes in fact from [1], and the theorem we present here is one of its direct corollaries.

Theorem 3.1 ([4, Corollary 4.3])**.** *For every $t \geq 0$, the set $\{x \in X_i^* : \delta_i(x) \leq t\}$ is infinite.*

Proof. Following [1, Theorem 5.31], for every $t \geq 0$, the set $C_{i,t} = \{x \in X_i^* : \delta_i(x) > -t\}$ is immune[2]. 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. The set in the statement being the complement of the immune set $\text{Complex}_{i,t}$, it is not computable, and in particular infinite. □

The next theorem 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 to simplify the proofs thanks to the elimination of some technical details. Nevertheless, those details are present in the following proof.

Theorem 3.2 ([4, 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*

$$|H_2(g(u)) - \log_2(i) \cdot H_i(u)| \leq c. \tag{3.1}$$

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

$$H_2(g(u)) \leq \log_2(i) \cdot H_i(u) + c_1 \tag{3.2}$$

and on the other hand

$$\log_2(i) \cdot H_i(u) \leq H_2(g(u)) + c_2. \tag{3.3}$$

For each string $w \in PROG_{U_i}$, we define $n_w = \lceil \log_2(i) \cdot |w|_i \rceil$. This integers verify the following:

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

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 $w \in PROG_{U_i}$, a binary string s_w of length n_w and such that the set $\{s_w : w \in PROG_{U_i}\}$ is c.e. and prefix-free. Accordingly, we can construct a machine M whose domain is this set, and such that for every $w \in PROG_{U_i}$,

$$M(s_w) = g(U_i(w)).$$

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 $M(s_{w^*}) = g(U_i(w^*)) = g(w)$, and hence

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

[1] The definition in [4] was $\delta_g(x) = H_2(g(x)) - \lceil \log_2 i \rceil \cdot |x|_i$.

[2] A set is said immune when it is infinite and contains no infinite c.e. subset.

By the Invariance Theorem, we get the constant c_1 such that (3.2) holds true.

We now prove the existence of c_2 such that (3.3) holds true. The proof is quite similar. For each string $w \in PROG_{U_2}$, we define $m_w = \lceil \log_i(2) \cdot |w|_2 \rceil$. As for the n_w, the integers m_w satisfy

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

We can also apply the Kraft-Chaitin Theorem to effectively construct, for every $w \in PROG_{U_2}$, a string $t_w \in X_i^*$ of length m_w and such that the set $\{t_w : w \in PROG_{U_2}\}$ 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 and 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 (3.3).

□

Comment. In [4], the equation (3.1) was $|\delta_g(u) - \lceil \log_2 i \rceil \cdot \delta_i(u)| \leq d$. Theorem 3.2 gives a similar result for δ, hence $|\delta_g(u) - \log_2(i) \cdot \delta_i(u)| \leq c + 1$, 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^*.

The next corollary will be important for the generalization of δ_g we will do in the next section. It is the same kind of result as above, but applied to two Gödel numberings.

Corollary 3.3 ([4, 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 (depending upon U_2, g and g') such that for all $u \in A$ we have:*

$$\left|H_2(g(u)) - H_2(g'(u))\right| \leq c. \tag{3.4}$$

In order to have a complete formal proof of Theorem 3.5, we need to bound the complexity of the set $\mathcal{T}$ of theorems that a theory $\mathcal{F}$ proves. It is the aim of the following lemma.

Lemma 3.4. *Let $\mathcal{F}$ be a finitely-specified, arithmetically sound (i.e. each arithmetical 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 $x \in \mathcal{T}$,*

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

Proof. We begin proving that the complexity of a theorem has to be greater than a 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 which cannot be compressed. To formalize the idea, we have to define in a formal way what the variables in our formal language are. We 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-written number identifying each variable. This number is called the identifier of the variable. In order to prevent any ambiguity, we can add another special symbol at the end of the identifier, and it can be the same character as at the beginning, v. In the sequel, we denote by v_n the variable the identifier of which is the integer n.

Now, we have to consider the formulae defined by

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

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

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

Thus, we obtained the lower bound.

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 we use 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 formulae. We consider the following program C: on its input x, C tests if x is a well-formed formula. It outputs it if the case arises, and enters in an infinite loop else.

This program has to be modified a bit as its domain is not prefix-free. The idea here is to add at the end of the input an ill-formed formula. More precisely, we need a formula y such that for every well-formed formula x, xy is ill-formed, and for every $z \in X_i^*$, xyz 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.

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

Comment. Improving the bounds in this lemma seems to be hard. A preliminary work should be to define exactly what we accept as a formal language.

The next theorem is the formal version of Chaitin's "heuristic principle". The very substance of the proof comes from previous results.

Theorem 3.5 ([4, Theorem 4.6])**.** *Consider a finitely-specified, arithmetically sound (i.e. each arithmetical 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. 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 3.4, for every $x \in \mathcal{T}$, $\delta_i(x) \leq c$. Using Theorem 3.2, 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 to prove a probabilistic result about independent statements. Indeed, we can prove that the probability a true statement of length n is provable tends to zero when n tends to infinity while the probability a statement is true remains always strictly positive.

Proposition 3.6 ([4, 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,*

$$\lim_{n \to \infty} i^{-n} \cdot card\{x \in X_i^* : |x|_i = n, \delta_g(x) \leq N\} = 0. \tag{3.5}$$

We do not give a proof of this proposition because it is essentially technical. It can be found in [4]. In Section 5, the proof of Proposition 5.6 uses the same arguments and differs from this one only by details. Now, we can express the probabilistic result about independent statements. The proof of this result can be found in [4, p. 11].

Theorem 3.7 ([4, 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, while the probability that a sentence of length n is true is strictly positive.*

4 Acceptable complexity measures

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

Definition 4.1. For a computable function $\hat{\rho}_i : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$, we define the *complexity measure builder* ρ by

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

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

Now, we define three properties that a builder has to verify to be *acceptable*. We recall that $\mathcal{F}$ denotes a theory which satisfy the hypothesis of Gödel Incompleteness Theorem, and $\mathcal{T}$ its set of theorems.

Definition 4.2. A builder ρ is said *acceptable* if for every g, the measure ρ_g verifies the three following conditions:

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

(ii) For every integer N,

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

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

The first property is simply the formal version of Chaitin's "heuristic principle". The second one corresponds to Proposition 3.6 and eliminate trivial measures. Finally, (iii) ensures the independence on the way the theorems are written. In other words, the properties (i), (ii) and (iii) ensure that an acceptable complexity measure satisfy Theorem 3.5, Proposition 3.6 and Corollary 3.3 respectively.

The following proposition will be useful in the sequel. It is a weaker version of the property (i) which is used to prove that a measure is not acceptable, and more precisely that it does not satisfy this first property.

Proposition 4.3. *Let ρ_g be an acceptable complexity measure. Then there exists an integer N such that for every integer $M \geq N$, the set*

$$\{x \in X_i^* : \rho_g(x) \leq M\} \tag{4.1}$$

is infinite.

Proof. We consider a theory $\mathcal{F}$ and the integer $N_{\mathcal{F}}$ given by the property (i) in Definition 4.2. Clearly, $\mathcal{F}$ can prove an infinity of theorems, such as "$n = n$" for all integer n. All of them have by property (i) a complexity bounded by $N_{\mathcal{F}}$. If $\mathcal{T}$ is the set of theorem that $\mathcal{F}$ proves, then

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

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 as the notion of acceptable complexity measure was built to generalize δ_g.

Proposition 4.4. *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. Let define

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

which plays the role of $\hat{\rho}_i$. Then $\delta_g(x) = \hat{\delta}_i(H_2(g(x)), |x|_i)$.

In fact, the properties of δ_g proved in [4] are exactly what we need here. One can easily check that(i) is ensured by Theorem 3.5, (ii) by Proposition 3.6 and (iii) by Corollary 3.3. □

The goal of defining an acceptable builder and an acceptable measure is to study other complexities than δ_g. The following example 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 4.5. 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 $\hat{\rho}_i(x, y) = x$ and such that $H_2(g(x)) = \hat{\rho}_i(x, |x|_i)$. We study the properties of the builder $g \mapsto [x \mapsto H_2(g(x))]$. Let us see how it behaves with the three properties of Definition 4.2.

(i) This first property cannot be verified. Indeed, we note that

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

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

(ii) This property is on the contrary obviously verified. Indeed, as $\operatorname{card}\{x \in X_i^* : H_2(g(x)) \leq N\} \leq 2^N$, $\{x \in X_i^* : |x|_i = n \text{ and } H_2(g(x)) \leq N\} = \varnothing$ for large enough n.

(iii) This property corresponds exactly to Corollary 3.3, and is verified.

As the program-size complexity cannot be used there, we try to find other complexities which 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[3]. In the next section, we will give two other examples of builder which are not acceptable.

[3] One has to be very careful with this statement which is not really true.

5 Independence of the three conditions

The aim of this section is to prove that the conditions (i), (ii) and (iii) in Definition 4.2 are independent from each other. At this end, we give two new examples of unacceptable builders. Each of those unacceptable builders exactly satisfy two conditions in Definition 4.2. Furthermore, they give us a first 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 5.1. Let $\hat{\rho}_i^1$ be the function defined by $\hat{\rho}_i^1(x,y) = x/y$ if $y \neq 0$ and 0 else. It defines a builder ρ^1 and for every Gödel numbering g, we can define ρ_g^1 by

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

We will see in the sequel that ρ^1 is a too small complexity. 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.

Example 5.2. We consider $\hat{\rho}_i^2$ defined by

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

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

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

In order to make the proofs easier, we introduce a new function for each already defined builders. Those functions make no use of Gödel numberings. They 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 up to a constant equal to the complexity measures. For ρ^1, we define ρ_i^1 be by $\rho_i^1(x) = H_i(x)/|x|_i$ if $x \neq \lambda$ and 0 else. 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 else.

Lemma 5.3. *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*

$$\left| \rho_g^j(u) - \log_2(i) \cdot \rho_i^j(u) \right| \leq c, \tag{5.1}$$

$j = 1, 2$.

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 sequel, we suppose that $|u|_i > 0$ (for $j = 1$) or $|u|_i > 1$ (for $j = 2$).

Theorem 3.2 states that

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

We now just have to divide the whole inequality by $|u|_i \geq 1$ to obtain (5.1) 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 the above lemma, we can note that both of the measures satisfy (iii).

Proposition 5.4. *The function ρ_g^1 verifies condition (i) in Definition 4.2, but does not verify (ii).*

Lemma 5.5. *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 [1, 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 $x \neq \lambda$,

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

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

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

Furthermore, Lemma 5.3 states that for every x, we have

$$\begin{aligned} \rho_g^1(x) &\leq c + \log_2(i) \cdot \rho_i^1(x) \\ &\leq c + \log_2(i) \cdot (1 + \alpha + \beta). \end{aligned}$$

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

Proof of Proposition 5.4. The property (i) is obvious since Lemma 5.5 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 $\left\{x \in X_i^* : |x|_i = n \text{ and } \rho_g^1(x) \leq N\right\}$ is the set X_i^n. Hence the limit of (ii) is 1 instead of 0.

□

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

Proposition 5.6. *The function ρ_g^2 verifies condition (ii) in Definition 4.2, but does not verify (i).*

Proof. We begin with the proof of (ii) for ρ^2. Theorem 5.3 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

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

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

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

We note that

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

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 Stolz-Cesàro Theorem to ensure that

$$\lim_{n\to\infty} \sum_{k=1}^{N\cdot\lceil \log_i n\rceil} i^{-N\cdot\lceil \log_i n\rceil} \cdot r_k = 0. \tag{5.2}$$

On the other hand,

$$\lim_{n\to\infty} i^{N\cdot\lceil \log_i n\rceil - n} = 0. \tag{5.3}$$

We just have to combine (5.2) and (5.3) to obtain (ii).

Now, it remains to prove that (i) is not verified. At this end, we suppose that (i) holds. We note $\mathcal{T}$ the set of theorems that $\mathcal{F}$ proves. Note first that

$$\begin{aligned}
\operatorname{card}\left\{x \in X_i^* :\ |x|_i = n \text{ and } H_2(g(x)) \leq N \cdot \lceil \log_i n\rceil\right\} &\leq \operatorname{card}\left\{y \in B^* : H_2(y) \leq N \cdot \lceil \log_i n\rceil\right\} \\
&\leq 2^{N\cdot\lceil \log_i n\rceil} \\
&\leq 2^{N\cdot(\log_i n + 1)} \\
&\leq 2^N \cdot n^{N\cdot\log_i 2}.
\end{aligned} \tag{5.4}$$

So, if (i) holds for all $x \in \mathcal{T}$, we have

$$\operatorname{card}\{x \in \mathcal{T} : |x| = n\} \leq \alpha n^{\beta N}, \tag{5.5}$$

for every integer n, where α and β come from (5.4).

But we now consider the set of formulae

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

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

As all those formulae belong to the predicate logic, 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 arbitrary large n, $2^{\mathcal{O}(n)}$ formulae of length n which belong to $\mathcal{T}$. That contradicts (5.5), and so, (i) is false. □

We can now prove that (i), (ii) and (iii) in Definition 4.2 are independent from each other. As we know, with δ_g, that there exists an acceptable complexity builder, it is sufficient to prove that for each of the three conditions, there exists a builder which does not satisfy it while it satisfies both other ones.

Theorem 5.7. *Each condition in Definition 4.2 is independent from others.*

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

In fact, our proof here does not exactly follow the scheme we gave. It is still unknown if all the complexity measure builders satisfy (iii), or if there exist some of them not satisfying it. Thus, the proof is built as follows. We prove that either all complexity builders satisfy (iii), or there exists at least one complexity builder satisfying (i) and (ii) without satisfying (iii). We also give the exact question the answer of which would make the choice between the both 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 to know if $H_2(g(x)) = H_2(g'(x))$ 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 $x \in X_i^*$, $\rho_g(x) = \hat{\rho}_i(H_2(g(x)), |x|_i) = \hat{\rho}_i(H_2(g'(x)), |x|_i) = \rho_{g'}(x)$. Consequently

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

and the builder ρ satisfy (iii).

We suppose now that the second case holds, that means that there exist infinitely many strings $x \in X_i^*$ such that $H_2(g(x)) \neq H_2(g'(x))$. We consider the acceptable complexity measure δ_g. We 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 $\hat{\rho}_i = \hat{\delta}_i^2$. Let us consider the behaviour of this function with the three properties:

(i) As 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 plain that $\rho_g(x) \leq {N_{\mathcal{F}}}^2$. So (i) is verified.

(ii) For an integer $N \geq 1$, if $\rho_g(x) \leq N$, then $\delta_g(x) \leq N$ too. So we have the following:

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

Consequently,

$$\begin{aligned} &\lim_{n\to\infty} i^{-n} \cdot \operatorname{card}\{x \in X^*i : |x|_i = n \text{ and } \rho_g(x) \leq N\} \\ &\quad\leq \lim_{n\to\infty} i^{-n} \cdot \operatorname{card}\{x \in X_i^* : |x|_i = n \text{ and } \delta_g(x) \leq N\} = 0. \end{aligned}$$

So (ii) is also verified.

(iii) We first note that

$$\begin{aligned} \rho_g(x) - \rho_{g'}(x) &= \delta_g(x)^2 - \delta_{g'}(x)^2 \\ &= (H_2(g(x)) - \lceil \log_2(i) \cdot |x|_i \rceil)^2 - (H_2(g'(x)) - \lceil \log_2(i) \cdot |x|_i \rceil)^2 \\ &= (H_2(g(x))^2 - H_2(g'(x))^2) - 2 \cdot \lceil \log_2(i) \cdot |x|_i \rceil \, (H_2(g(x)) - H_2(g'(x))). \end{aligned}$$

We know from Corollary 3.3 that $(H_2(g(x)) - H_2(g'(x)))$ 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 (iii) 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 $H_2(g(x)) > H_2(g'(x)) > M$[4]. Then

$$\begin{aligned} H_2(g(x))^2 - H_2(g'(x))^2 &= (H_2(g(x)) - H_2(g'(x))) \cdot (H_2(g(x)) + H_2(g'(x))) \\ &> 1 \cdot (2 \cdot M) = 2M. \end{aligned}$$

We can also conclude, using an integer $M > N/2$ that this bound cannot exist, that is (iii) 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 will study some conditions a builder, and in particular its witness, has to verify such that the complexity measures it builds are acceptable ones. We restrict 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. Furthermore, it makes the following proofs easier.

Proposition 6.1. *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.*

[4] 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.

Proof. Property (i) in Definition 4.2 remains true with a new constant $\alpha \cdot N + \beta$ instead of N. In the same way,

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

hence Property (ii) is verified. Now, if we consider two Gödel numberings g and g',

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

which proves that Property (iii) is retained. □

We start studying the linear in both variables witnesses. The result we obtain is partial. However, as discussed after Lemma 3.4, this result is not likely to be improved without a complete study of the definition of the formal languages.

Proposition 6.2. *Let f be a function of two variables, linear in both 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 $1/2 \leq \varepsilon \leq 1$, such that*

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

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

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

Proposition 6.1 allows us to fix $\hat{\rho}_i(0, 0) = 0$. Of course, it would be equivalent to consider $\alpha x + \beta y + \gamma xy$, but the chosen version simplifies the notations. Let β' be such that $\beta = \beta' \cdot \log_2(i)$. The proof is done in several steps. We start by showing that one at least of α and γ 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 4.3 is not verified by our complexity measure, and hence neither is Property (i). If $\beta \geq 0$, it is obvious that Property (ii) cannot hold true.

Then, we use the property (i) 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,

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

The only solution is the third one because in order to satisfy (i), this limit has to be infinite. Indeed, if it is finite, we can use the same proof as in Proposition 5.6 to conclude to a contradiction. So we know that $\gamma = 0$, and hence that $\alpha \neq 0$. We can right now say that α and β have the same sign, because the limit cannot be $-\infty$. Using Proposition 6.1, we can assume that $\alpha = 1$. Indeed, $\alpha < 0$ is not possible because of Property (ii).

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

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

Applying Theorem 3.2, we get a constant c such that for every x,

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

We will now use the property (ii) to have other information on β', and hence β. We only know at that stage that $\beta' > 0$. We consider the set

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

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 $H_i(x) \leq |x|_i + \mathcal{O}(\log_i |x|_i)$ (see [1, Theorem 3.22]) to conclude that the above set is X_i^n. And so, property (ii) is not verified, the limit being 1.

Using now the lower bound in Lemma 3.4, 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 $\mathcal{F} \vdash x$,

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

Thus, (i) cannot be verified. □

We study another kind of witnesses. 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 witness of *multiplicative* complexity measures. Indeed, a complexity of the form $H_2(g(x)) \cdot |x|_i$ has no chance to satisfy the desired properties. Unfortunately, such functions never define acceptable measures.

Proposition 6.3. *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 the witness of which is $\hat{\rho}_i$ cannot satisfy at the same time properties (i) and (ii).

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

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

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

$$\operatorname{card}\{x \in \mathcal{T} : |x|_i = n\} \leq 2^{N \cdot f(n)}. \tag{6.1}$$

Now, we give a lower bound to this cardinal. The proof of Proposition 5.6 shows that this cardinal is greater to $2^{\mathcal{O}(n)}$. Accordingly, there exists a constant c such that

$$\operatorname{card}\{x \in \mathcal{T} : |x|_i = n\} \geq 2^{c \cdot n}. \tag{6.2}$$

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

$$f(n) \geq \frac{c}{N} \cdot n. \tag{6.3}$$

We now follow the proof we made to show that ρ_g^1 does not satisfy (ii). We can define

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

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

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

The proof of Lemma 5.3 is still valid here. In the same way, we extend Lemma 5.5 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 an influence on the value of the constant M.

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

□

7 Concluding remarks

In this paper, we have studied the δ_g complexity function defined by Calude and Jürgensen [4]. This study has led us to modify a bit the definition of δ_g in order to correct some of the proofs. Then, we have been able to propose a definition of *acceptable complexity measure of theorem* which captures the main properties of δ_g. Studying some complexity measures, we have shown 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.

There remain some open questions. Among them, we can express the following ones:

- Can we improve the bounds of Lemma 3.4? This question could be interesting not only to improve Proposition 6.2 but also for itself: How simple are the well-formed formulae, 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, one has to investigate about the definition of the formal languages. The answer seems to be very dependent on the considered language.
- Do there exist some acceptable complexity measure which are very different from δ_g? The idea here is to find some measures with which we go further on the investigations about the roots of unprovability.
- In view of the proof of Theorem 5.7, 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 Calude and Jürgensen expressed in [4]. The goal of finding new acceptable complexity measures is to have new tools to try to answer their questions, as the existence of independent sentences of small complexity.

Alps, South Island

Acknowledgments

Special thanks are due to my supervisor Cristian S. Calude for his warm hospitality in the University of Auckland. Thanks his perpetual suggestions, corrections and improvements, as well as his encouragements, my stay in Auckland was a very exciting period. I cannot list all the things I learned during three months. *Merci beaucoup Cris !*

Thanks are also due to André Nies for his stimulating comments and ideas. In particular, he gave us the lower bound in Lemma 3.4. I am also grateful to the other members of the Computer Science department for their various kinds of help, and in particular to Robyn and Sithra for their infinite patience.

Several persons in Lyon made this internship possible. I especially thank Jacques Mazoyer for having given the idea to go in Auckland, and all those who made the administrative part easier.

This trip does not come down to the work I did at the University. Thank you to all those who permit me to discover *the land of the long white cloud.*

Tēnā kōrua i ā kōrua manaakitanga mai!

Mount Cook, South Island

References

[1] C. Calude. *Information and Randomness: An Algorithmic Perspective*. Springer-Verlag, Berlin, 1994, second ed., revised and extended, 2002.

[2] C. Calude, H. Jürgensen, and M. Zimand. Is independence an exception? *Appl. Math. Comput.*, 66:63–76, 1994.

[3] C. S. Calude. Incompleteness: A Personal Perspective. *Proc. DCFS'08*, 2008. To appear.

[4] C. S. Calude and H. Jürgensen. Is complexity a source of incompleteness? *Adv. in Appl. Math.*, 35:1–15, 2005.

[5] G. Chaitin. Information-theoretic limitations of formal systems. *J. Assoc. Comput. Mach.*, 21:403–424, 1974.

[6] G. Chaitin. A theory of program size formally identical to information theory. *J. Assoc. Comput. Mach.*, 22:329–340, 1975.

[7] K. Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatsh. Math.*, 38:173–198, 1931.

[8] A. Kolmogorov. Three approaches to the quantitative definition of information. *Int. J. Comput. Math.*, 2:157–168, 1968.

[9] M. Li and P. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Graduate Texts In Computer Science. Springer-Verlag, Berlin, 1993; second ed., 1997.

[10] P. Odifreddi. *Classical Recursion Theory*. North-Holland, Amsterdam, Vol. 1, 1989, Vol. 2, 1999.

[11] C. Papadimitriou. *Computational Complexity*. Addison-Wesley Reading, Mass, 1994.

[12] M. Sipser. *Introduction to the Theory of Computation*. PWS Publishing, Boston, 1997; second ed., 2006.

[13] A. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proc. Lond. Math. Soc.*, 42:230–265, 1936.

Auckland, North Island