Feasible Mathematics

Alexandre Harvey-Tremblay (aht@protonmail.ch)

2 January 2018; Last revision 2 January 2018

From algorithmic information theory (and using notions of algorithmic thermodynamics), we introduce *feasible mathematics* as distinct from *universal mathematics*. Feasible mathematics formalizes the intuition that theorems with very long proofs are unprovable within the context of limited computing resources. It is formalized by augmenting the standard construction of Ω with a conjugate-pair that suppresses programs with long runtimes. The domain of the new construction defines feasible mathematics.

Contents

0.1. *Notation*

We will use the following notations: The double vertical lines $|X|$ means the length of the string X. The suffix b, for example in 110_b, refers to the binary notation. The over-line, as in $\overline{|X|}$ indicates the mean value of $|X|$ from a statistical ensemble.

1. Introduction

Some research has been done in the area of feasible numbers. Perhaps the most promising is from Vladimir Yu. Sazonov's paper on feasible numbers (Sazonov(1995)). He suggest that feasible numbers are intuitively the set of numbers F which satisfies $0 \in F$, $F+1 \subseteq F$

and $2^{1000} \notin F$. Then, he goes on to investigate various constructions which would allow the consistent treatment of such a set.

Here, we take a different approach. We recognize that 2^{1000} is a large number but nonetheless, it can be compressed to a short representation. Thus, we accept that theorems featuring this number can be proven even in the context of limited resources. Hence, a more general approach to feasibility is required. Therefore, we propose a method to treat feasibility as limits applicable to complexity of the proof itself.

Mathematical proofs come in various sizes and/or other complexities. By limiting proof based on size and other indicators of complexity, we construct a statistical ensemble able to describe the proof landscape available to a mathematician with limited computational resources. We believe that this representation of mathematical feasibility more accurately describe the intuitive notion of feasibility. After-all, a theorem whose shortest proof requires 2^{1000} bits will surely never be proven in our lifetime, but the number 2^{1000} is easily representable even in simple proofs.

The construction is meta-logically applicable to an arbitrary set of formal axioms. Thus, we introduce a distinction between *feasible mathematics* and *universal mathematics*. Universal mathematics is made feasible when the accessible proof landscape of the mathematician is reduced by computational limits. In this sense, all practical work in mathematics is feasible.

To formalize feasible mathematics, we will consider mathematical proofs as computer programs that are executed on a self-delimiting universal Turing machine.

1.1. *Main problem*

Suppose a research group with access to a supercomputer. Alice has been granted a fixed amount of computing resources to use on the supercomputer. She has further been instructed to run a program q_A. With no prior knowledge of q_A, what is the probability that the program will halt within the allocated resources?

Answering this question will require notions of *algorithmic thermodynamics*.

1.2. *Statistical physics*

Before we introduce *algorithmic thermodynamics*, we will provide a brief recap of statistical physics. In statistical physics, we are interested in the distribution that maximizes entropy

$$S = -k_B \sum_{x \in X} p(x) \ln p(x) \tag{1}$$

, where

Observable	Conjugate	Relation
Energy E	Temperature T	$\beta = 1/(k_b T)$
Volume V	Pressure p	$\gamma = p/(k_b T)$
Number of particles N	Chemical potential μ	$\delta = -\mu/(k_b T)$

Table 1. *Typical observables of statistical mechanics.*

$$S \in \mathbb{R}_{\geq 0} \qquad \text{entropy} \tag{2}$$

$$k_B \approx 1.38 \times 10^{-23} \frac{m^2 kg}{s^2 K} \qquad \text{Boltzmann constant} \tag{3}$$

$$X \qquad \text{ensemble of micro-states} \tag{4}$$

$$x \in X \qquad \text{micro-state} \tag{5}$$

$$p(x) \in \mathbb{R} \cap [0,1] \qquad \text{probability of the system being in micro-state } x \tag{6}$$

subject to the fixed macroscopic observables. The solution for this is the Gibbs ensemble. Taking the observables listed in Table 1 as examples, the partition function becomes

$$Z = \sum_{x \in X} e^{-\beta E(x) - \gamma V(x) - \delta N(x)} \tag{7}$$

, where

$$Z \in \mathbb{R}_{>0} \qquad \text{normalization constant} \tag{8}$$

The probability of occupation of a micro-state is;

$$p(x) = \frac{1}{Z} e^{-\beta E(x) - \gamma V(x) - \delta N(x)} \tag{9}$$

The average values and their variance for the observables are;

$$\overline{E} = \sum_{x \in X} p(x) E(x) \qquad \overline{E} = \frac{-\partial \ln Z}{\partial \beta} \qquad \overline{(\Delta E)^2} = \frac{\partial^2 \ln Z}{\partial \beta^2} \tag{10}$$

$$\overline{V} = \sum_{x \in X} p(x) V(x) \qquad \overline{V} = \frac{-\partial \ln Z}{\partial \gamma} \qquad \overline{(\Delta V)^2} = \frac{\partial^2 \ln Z}{\partial \gamma^2} \tag{11}$$

$$\overline{N} = \sum_{x \in X} p(x) N(x) \qquad \overline{N} = \frac{-\partial \ln Z}{\partial \delta} \qquad \overline{(\Delta N)^2} = \frac{\partial^2 \ln Z}{\partial \delta^2} \tag{12}$$

The laws of thermodynamics can be recovered by taking the following derivatives

$$\left.\frac{\partial S}{\partial \overline{E}}\right|_{V,N} = \frac{1}{T} \qquad \left.\frac{\partial S}{\partial \overline{V}}\right|_{E,N} = \frac{p}{T} \qquad \left.\frac{\partial S}{\partial \overline{N}}\right|_{E,V} = -\frac{\mu}{T} \tag{13}$$

which can be summarized as

$$d\overline{E} = TdS - pd\overline{V} + \mu d\overline{N} \tag{14}$$

This is known as the equation of state of the thermodynamic system. The entropy can be recovered from the partition function. It is given by

$$S = k_B \left(\ln Z + \beta\overline{E} + \gamma\overline{V} + \delta\overline{N}\right) \tag{15}$$

1.3. *Algorithmic thermodynamics*

Many authors ((Bennett et al.(1998)Bennett, Gacs, Li, Vitanyi, and Zurek; Chaitin(1975); Fredkin and Toffoli(1982); Kolmogorov(1965); Zvonkin and Levin(1970); Solomonoff(1964); Szilard(1964); Tadaki(2002); Tadaki(2008))) have discussed the similarity between physical entropy $S = -k_B \sum p_i \ln p_i$ and the entropy in information theory $S = -\sum p_i \log_2 p_i$. Furthermore, the similarity between the halting probability Ω and the Gibbs ensemble of statistical physics has also been studied ((Li and Vitanyi(2008); Calude and Stay(2006); Baez and Stay(2012); Tadaki(2002))). Tadaki suggests to augment Ω with a multiplication constant D which acts as a decompression term on Ω.

Chaitin construction $\qquad\qquad$ Tadaki ensemble $\qquad$ (16)

$$\Omega = \sum_{q \in \text{halts}} 2^{-|q|} \qquad \Rightarrow \qquad \Omega_D = \sum_{q \in \text{halts}} 2^{-D|q|} \tag{17}$$

With this change, the Gibbs ensemble compares to the Tadaki ensemble as follows;

Gibbs ensemble $\qquad\qquad$ Tadaki ensemble

$$Z = \sum_{x \in X} e^{-\beta E(x)} \qquad\qquad \Omega_D = \sum_{q \in \text{halts}} 2^{-D|q|} \tag{18}$$

Interpreted as a Gibbs ensemble, the Tadaki construction forms a statistical ensemble where each program corresponds to one of its micro-state. It maximizes the entropy subject to constraints on its observables. The Tadaki ensemble admits a single observable; the prefix code length $|q|$. As a result, it describes the partition function of a system which maximizes the entropy subject to the constraint that the average length of the codes is a constant $\overline{|q|}$;

$$\overline{|q|} = \sum_{q \in \text{halts}} |q| 2^{-|q|} \qquad \text{from 10} \tag{19}$$

, where

$$\overline{|q|} \in \mathbb{R} \qquad \text{Average prefix code length of } \Omega_D \tag{20}$$

In this interpretation, the Tadaki ensemble will have an entropy which corresponds to the choice of prefix-free codes available to encode the programs.

$$S = k_B \left(\ln \Omega + D\overline{|q|} \ln 2 \right) \qquad \text{from 15} \tag{21}$$

where the constant $\ln 2$ comes from the base 2 of the halting probability function instead of base e of the Gibbs ensemble.

John C. Baez and Mike Stay(Baez and Stay(2012)) take the analogy further by suggesting an interpretation of algorithmic information theory based on thermodynamics, where the characteristics of programs are considered to be observables. Starting from Gregory Chaitin's Ω number, the Chaitin construction

$$\Omega = \sum_{q \in \text{halts}} 2^{-|q|} \tag{22}$$

is extended with algorithmic observables to obtain

Gibbs ensemble	Baez-Stay ensemble	(23)
$Z = \sum_{x \in X} e^{-\beta E(x) \gamma V(x) - \mu N(x)}$	$\Omega' = \sum_{q \in \text{halts}} 2^{-\beta E(q) - \gamma V(q) - \delta N(q)}$	(24)

Noting the similarity between the Gibbs ensemble of statistical physics (7) and (24), these authors suggest an interpretation where E is the expected value of the logarithm of the program's runtime, V is the expected value of the length of the program and N is the expected value of the program's output. Furthermore, they interpret the conjugate variables as (quoted verbatim from their paper);

"

1 $T = 1/\beta$ is the *algorithmic temperature* (analogous to temperature). Roughly speaking, this counts how many times you must double the runtime in order to double the number of programs in the ensemble while holding their mean length and output fixed.

2 $p = \gamma/\beta$ is the *algorithmic pressure* (analogous to pressure). This measures the tradeoff between runtime and length. Roughly speaking, it counts how much you need to decrease the mean length to increase the mean log runtime by a specified amount, while holding the number of programs in the ensemble and their mean output fixed.

3 $\mu = -\delta/\beta$ is the *algorithmic potential* (analogous to chemical potential). Roughly speaking, this counts how much the mean log runtime increases when you increase the mean output while holding the number of programs in the ensemble and their mean length fixed.

"

–John C. Baez and Mike Stay

From equation (24), they derive analogues of Maxwell's relations and consider thermodynamic cycles, such as the Carnot cycle or Stoddard cycle. For this, they introduce the concepts of *algorithmic heat* and *algorithmic work*.

Other authors have suggested other alternative mappings(Li and Vitanyi(2008); Tadaki(2008)).

2. Derivation of feasible mathematics

(a) We start from the standard Chaitin construction applicable to a self-delimiting universal Turing machine (Chaitin(1975)).

$$\Omega = \sum_{q \in \text{halts}} 2^{-|q|} \tag{25}$$

, where

$$\Omega \in (\mathbb{R} \cap [0,1]) \quad \text{numerical value of the sum} \tag{26}$$
$$q \in \Sigma_b \quad \text{binary program (encoded as a prefix-free code)} \tag{27}$$
$$|q| : \Sigma_b \to \mathbb{N} \quad \text{length of the program's code} \tag{28}$$

(b) We augment Ω with a multiplication constant D; we obtain the Tadaki ensemble (Tadaki(2002)).

$$\Omega_D = \sum_{q \in \text{halts}} 2^{-D|q|} \tag{29}$$

, where

$$\Omega_D \in (\mathbb{R} \cap [0,1]) \quad \text{numerical value of the sum} \tag{30}$$
$$D \in \mathbb{R} \quad \text{Conjugate to program length} \tag{31}$$

(c) With this addition, Ω_D has the same mathematical structure as a Gibbs ensemble of statistical physics (Tadaki(2008); Baez and Stay(2012)).

$$\begin{array}{ll} \text{Gibbs ensemble} & \text{Tadaki ensemble} \\ G = \sum_{x \in X} e^{-\beta E(x)} & \Omega_D = \sum_{q \in \text{halts}} 2^{-D|q|} \end{array} \tag{32}$$

, where

$$\left(2^{-D|q|}\right) \quad \text{micro-state representing a program} \tag{33}$$
$$D \in \mathbb{R} \quad \text{algorithmic decompression} \tag{34}$$

(d) Following (Baez and Stay(2012)), we interpret the Tadaki ensemble within the context of algorithmic thermodynamics. We can introduce a probability distribution for Ω and Ω_D that maximizes the entropy of the system.

$$\begin{array}{ll} \text{Halting probability} & \text{Halting probability with fixable } \overline{|q|} \\ p(q) = \frac{1}{\Omega} 2^{-|q|} & p(q, D) = \frac{1}{\Omega_D} 2^{-D|q|} \end{array} \tag{35}$$

In the case of Ω_D, D is a Lagrange multiplier and $p(q, D)$ is the probability measure that maximizes the entropy subject to the constraint that the average program length is $\overline{|q|}$.

$$\overline{|q|} = \sum_{q \in \text{halts}} p(x)|q| \tag{36}$$

, where

$$\overline{|q|} \in \mathbb{R}_{\geq 0} \qquad \text{average program length} \tag{37}$$

(e) Finally, to obtain feasible mathematics, we introduce into Ω_D the observable $t(q)$ the runtime of program q and pair it with its conjugate W. We obtain the construction Z.

$$Z = \sum_{q \in \text{halts}} 2^{-Wt(q) - D|q|} \tag{38}$$

$$\tag{39}$$

, where

$$Z \in \mathbb{R}_{\geq 0} \qquad \text{numerical value of the sum} \tag{40}$$

$$t(q) : q \to \mathbb{N} \qquad \text{number of iterations required for } q \text{ to halt} \tag{41}$$

$$W \in \mathbb{R} \qquad \text{conjugate to } t(q) \text{ in units of } (iterations)^{-1} \tag{42}$$

$$|q| : q \to \mathbb{N} \qquad \text{number of bits of program } q \tag{43}$$

$$D \in \mathbb{R} \qquad \text{conjugate to } |q| \text{ in units of } (bits)^{-1} \tag{44}$$

The corresponding probability measure is:

$$p(q, W, D) = \frac{1}{Z} 2^{-Wt(q) - D|q|} \tag{45}$$

It maximizes the entropy subject to the following constraints:

$$\overline{|q|} = \sum_{q \in \text{halts}} p(q, W, D)|q| \qquad \text{fixed average program length } \overline{|q|} \tag{46}$$

$$\bar{t} = \sum_{p \in \text{halts}} p(q, W, D)t(q) \qquad \text{fixed average program runtime } \bar{t} \tag{47}$$

Let us now study this equation in more detail in the following section.

3. Results

We interpret the supercomputing research group as taking a similar role to the role taken by the various baths in thermodynamics (heat bath, particle bath). For example, in thermodynamics we would say that a system which can exchange energy with its environment is in a heat bath. Its temperature will be constant but its total energy would fluctuate as it is exchanged with the bath. By analogy, in feasible mathematics, we would imagine that a computation occurs in a supercomputer which schedule priority, assigns memory, etc. so has to maintain various computing resources fixed during the calculation. This replaces the role of the baths.

To make this more precise, let us define what we mean by fixed resources.

3.1. *Fixed resources*

Each Lagrange multiplier of the partition function Z is a computing resource fixed by the supercomputer. In the provided definition of Z, there are two such constants : W and D. They can be interpreted as follows:

name	constant	
Halting-power	$\mathcal{P} = \frac{1}{W}$	(48)
Halting-density	$\mathcal{F} = \frac{D}{W}$	(49)

— The halting-power counts how much the runtime must be doubled in order to double the entropy of the ensemble while holding the mean length fixed.
— The halting-density counts how much the average length must be decreased to increase the average runtime by a specified amount, while holding the entropy in the ensemble fixed.

3.2. *Alternative formulations*

There exists alternative constructions of Z such that other resources are fixed by the supercomputer.

Action-frequency formulation:

$$Z' = \sum_{q \in \text{halts}} 2^{-\mathcal{A}f(q) - D|q|} \tag{50}$$

The supercomputer must fix

name	constant	
Halting-action	$\mathcal{S} = \frac{1}{\mathcal{A}}$	(51)

— The halting-action counts how much the action must be doubled in order to double the entropy of the ensemble while holding the mean length fixed.

Time-power formulation:

$$Z'' = \sum_{q \in \text{halts}} 2^{-tP(q) - D|q|} \tag{52}$$

The supercomputer must fix

name	constant	
Halting-time	t	(53)

— The halting-time counts how much the time must be doubled in order to double the entropy in the ensemble while holding the mean length fixed.

This last formulation might seem more familiar (as it is easy to imagine that a computation can be stopped after a certain time) but the fixation of the computing resource is less intuitive. To guarantee that the work on each program terminates in time for the cut-off (e.g. there are no partial execution), the supercomputer must adjust the computation power on a per program basis.

3.3. *Relation to* Ω

Theorem 54. $Z \to \Omega_D$ as the amount of available resources is increased arbitrarily.

$$\lim_{\mathcal{P}\to\infty} Z \to \Omega_D \tag{55}$$

Proof.
First, we rewrite Ω_D as:

$$\Omega_D = \sum_{i=1}^{\infty} 2^{-H(q_i)-D|q_i|} \qquad \text{where } H(q_i) := \begin{cases} 0 & q_i \text{ halts} \\ \infty & \text{otherwise} \end{cases} \tag{56}$$

Second, we note that the runtime $t(q_i)$ of a program q_i will be finite if it halts and infinite otherwise.

$$t(q_i) = \begin{cases} t_i \in \mathbb{R}_{\geq 0} & q_i \text{ halts} \\ \infty & \text{otherwise} \end{cases} \tag{57}$$

Then taking the limit of Z,

$$\lim_{\mathcal{P}\to\infty} \frac{1}{\mathcal{P}} t(q_i) = \begin{cases} 0 & q_i \text{ halts} \\ \infty & \text{otherwise} \end{cases} \tag{58}$$

This is the definition of $H(q_i)$. Therefore,

$$\lim_{\mathcal{P}\to\infty} \frac{1}{P} t(q_i) \to H(q_i) \tag{59}$$

Thus,

$$\lim_{\mathcal{P}\to\infty} Z \to \Omega_D \tag{60}$$

□

Theorem 61. Z monotonically converges towards Ω_D as the available resources are increased.

Proof.

Without loss of generality, let us now expand Z explicitly with an example. Assume a system comprised of three micro-states with prefix code-length $|q_1| = 1$, $|q_2| = 2$ and $|q_3| = 3$ and with the *runtimes* $t_1 = 5$, $t_2 = \infty$ and $t_3 = 5$. In this example, q_1 and q_2 halt and q_3 does not. For the purposes of simplicity we can assume that all other programs do not halt. In this case the system is not universal but let us nonetheless use it as a simplified numerical example. The sum Z becomes;

$$Z(W) = 2^{-1-5W} + 2^{-2-\infty W} + 2^{-3-5W} \tag{62}$$

We will now produce a series of numerical calculations with progressively smaller values of W and we will look at the evolution of the error rate $\xi(W) = \Omega - Z(W)$. For this system, $\Omega = 0.101\overline{0}_b$.

W	$Z(W)$	$\xi(W)$	error	
∞	0	Ω	max	(63)
1	$0.000000101..._b$	0.10011011_b	$\approx 2^{-1}$	(64)
0.1	$0.011100010..._b$	$0.00101110..._b$	$\approx 2^{-3}$	(65)
0.01	$0.100110101..._b$	$0.00000010..._b$	$\approx 2^{-6}$	(66)
0.001	$0.011100010..._b$	$0.00000000..._b$	$\approx 2^{-9}$	(67)
$\vdots$	$\vdots$	$\vdots$	$\vdots$	
0	Ω	0	none	(68)

As we can see, increasing the halting-power ($\mathcal{P} = 1/W$) causes the value Z to monotonically converges towards Ω. The error rate decreases as more valid bits of Ω are obtained. □

4. Main results

Theorem 69. An observer knowing n bits of Z will be able to decide at most 2^N programs.

Proof. We consider a numerical value for Z whose first k bits corresponds to the bits of Ω. We look at two cases: 1) For the first k bits, Z (as with Ω) can decide 2^N programs per bit. 2) For the bits after k, the situation is a bit more complex:

To recover the feasible programs beyond k, an observer can execute programs on a universal Turing machine in dovetail. As they halt, the observer adds their contribution to Z. Once the value of Z is recovered, then all programs taking longer to halt are beyond the feasible bound, regardless of whether they ultimately halt or not.

□

Theorem 70. programs in Z obeys Fermi-Dirac statistics.

Proof. Each micro-state of Z represent a program. Under the assumption that a rational operator will not waste resources by executing the same program twice, then each micro-state are either executed, or not executed during the computation. Thus, as each state is either 'occupied' or 'not-occupied', the statistic is Fermi-Dirac.

□

5. Conclusion

Thus, Z defines the boundary for feasible mathematics.

References

John Baez and Mike Stay. Algorithmic thermodynamics. Mathematical. Structures in Comp. Sci., 22(5):771–787, September 2012. ISSN 0960-1295. . URL `http://dx.doi.org/10.1017/S0960129511000521`.

C. H. Bennett, P. Gacs, Ming Li, P. M. B. Vitanyi, and W. H. Zurek. Information distance. IEEE Transactions on Information Theory, 44(4):1407–1423, July 1998. ISSN 0018-9448. .

Cristian S. Calude and Michael A. Stay. Natural halting probabilities, partial randomness, and zeta functions. Inf. Comput., 204(11):1718–1739, November 2006. ISSN 0890-5401. . URL `http://dx.doi.org/10.1016/j.jc.2006.07.003`.

Gregory J. Chaitin. A theory of program size formally identical to information theory. J. ACM, 22(3):329–340, July 1975. ISSN 0004-5411. . URL `http://doi.acm.org/10.1145/321892.321894`.

Edward Fredkin and Tommaso Toffoli. Conservative logic. International Journal of Theoretical Physics, 21(3):219–253, Apr 1982. ISSN 1572-9575. . URL `https://doi.org/10.1007/BF01857727`.

Andrei Nikolaevich Kolmogorov. Three approaches to the definition of the concept "quantity of information". Problemy peredachi informatsii, 1(1):3–11, 1965.

Ming Li and Paul M.B. Vitanyi. An Introduction to Kolmogorov Complexity and Its Applications. Springer Publishing Company, Incorporated, 3 edition, 2008. ISBN 0387339981, 9780387339986.

Vladimir Yu Sazonov. On feasible numbers. In Logic and computational complexity, pages 30–51. Springer, 1995.

R.J. Solomonoff. A formal theory of inductive inference. part i. Information and Control, 7(1):1 – 22, 1964. ISSN 0019-9958. . URL `http://www.sciencedirect.com/science/article/pii/S0019995864902232`.

Leo Szilard. On the decrease of entropy in a thermodynamic system by the intervention of intelligent beings. Behavioral Science, 9(4):301–310, 1964. ISSN 1099-1743. . URL `http://dx.doi.org/10.1002/bs.3830090402`.

Kohtaro Tadaki. A generalization of chaitin's halting probability omega and halting self-similar sets. Hokkaido Math. J., 31(1):219–253, 02 2002. . URL `http://dx.doi.org/10.14492/hokmj/1350911778`.

Kohtaro Tadaki. A statistical mechanical interpretation of algorithmic information theory. In Local Proceedings of the Computability in Europe 2008 (CiE 2008), pages 425–434. University of Athens, Greece, Jun 2008. URL `http://arxiv.org/abs/0801.4194`.

A K Zvonkin and L A Levin. The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. Russian Mathematical Surveys, 25(6):83, 1970. URL `http://stacks.iop.org/0036-0279/25/i=6/a=R05`.