全脳アーキテクチャ勉強会

アナログ計算と計算可能性

平成29年5月9日

河村彰星（東京大学総合文化研究科）
http://www.graco.c.u-tokyo.ac.jp/~kawamura/

計算の理論

計算 によって何ができるか・できないか

決められた手順に従った情報処理（アルゴリズム）
計算のしくみや　使える時間や空間（記憶領域）の量に制約

計算機の歴史とともに発展

1930　計算可能性　K. Gödel, A. Church, A. Turing, S. Kleene, E. Post
50
70　計算量（時間・空間の制限）　多項式時間
M. Blum, J. Hartmanis
S. Cook, L. Levin, R. Karp
90
2016

現実の計算機の仕組に特化した議論ではなく
むしろ情報処理・知能の一般的な限界を理解したい

定義

チューリング機械は次のものにより指定される

- 有限個の**状態**の集合 Q ... ている
- 有 ...
  こ ...
- **選** ...

初め ...

テー ... て

次の ...

状 ...

離散的な動作

「状態○で記号△を読むと次の時刻には
状態と読取り位置を●に変えて
記号▲を書込む」という形の規則

記号・状態・規則は有限個

- 「止」なら　停止する
- (q',σ',d) なら
  状態を q' にし σ' を書込み d の向きに一歩進む

$Q_{\text{受理}}$ の状態で停止したら機械は x を**受理**したという

何故「多項式以内か否か」が重要か

■入力が大きくなると手間が大違い

入力長 $n =$	10	30	50	100	1000	1万	100万	1億
$\sqrt{n}$	1秒以内	1秒以内	1秒以内	1秒以内	1秒以内	1秒以内	1秒以内	1秒以内
n	1秒以内	1秒以内	1秒以内	1秒以内	1秒以内	1秒以内	1秒	2分
n^2	1秒以内	1秒以内	1秒以内	1秒以内	1秒	2分	12日	3百年
n^3	1秒以内	1秒以内	1秒以内	1秒	17分	12日	3万年	3百億年
2^n	1秒以内	18分	36年	4京年				
$n!$	4秒	8百京年						

多項式時間

指数時間

1秒に1000000回の処理が
できるとしたときにかかる時間

■指数個（以上）の組合せから何かを探す場面は多い（組合せ爆発）

問題 与えられた正整数（十進法で n 桁）が素数か判定

それ未満の数（10^n 個ほどある）で割り切れるか全部調べれば判る

それよりも劇的に速く　n の多項式時間で解く方法が見つかった

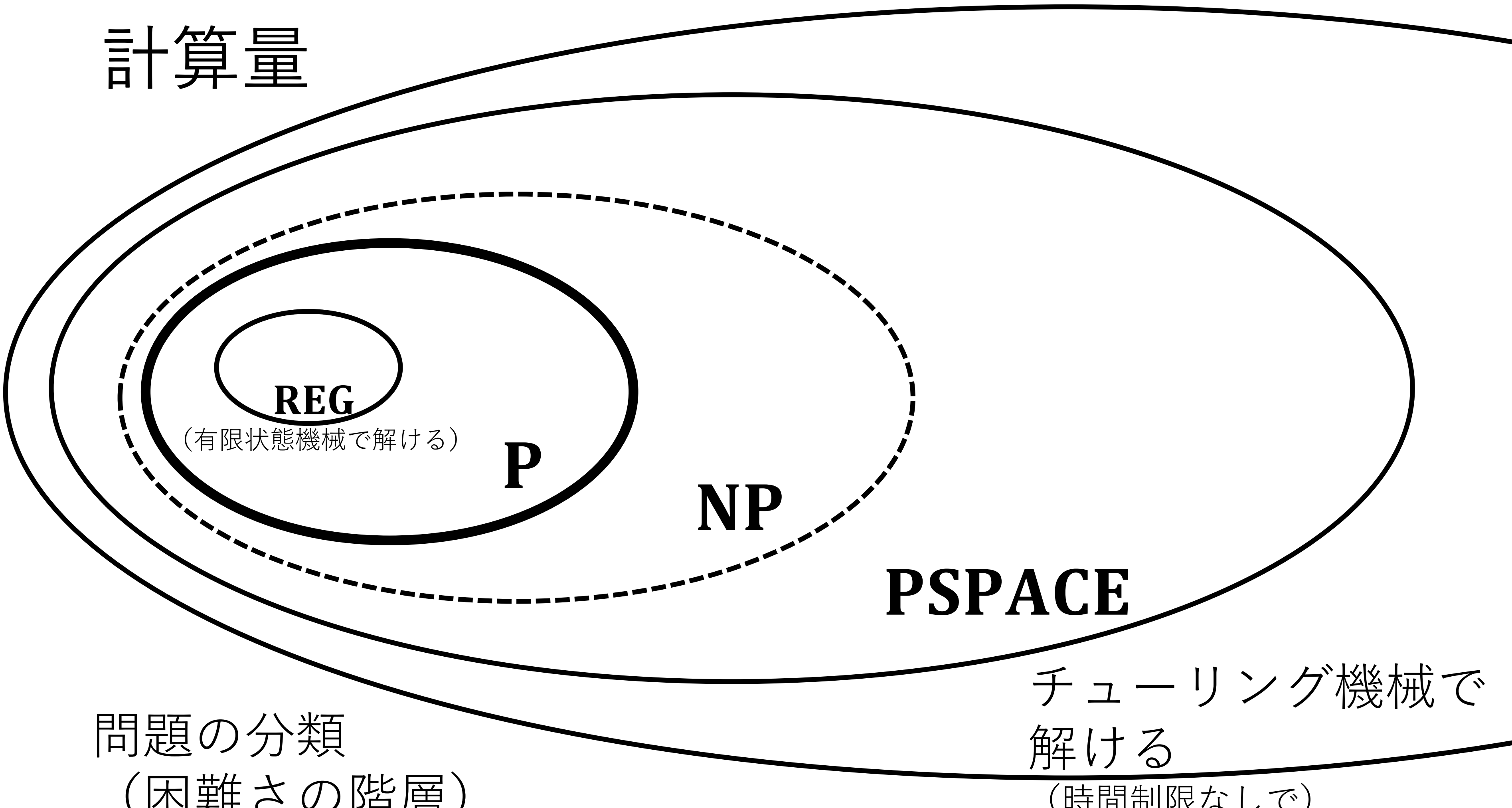
AKS素数判定法（2002）

問題 与えられたグラフ（頂点 n 個）がハミルトン閉路をもつか判定

頂点の並べ方（$n!$ 個ある）を全部調べれば判る

n の多項式時間で解く方法があるかどうかは判っていない

計算量



問題の分類
（困難さの階層）

未解決 $\mathbf{PSPACE} \overset{?}{=} \mathbf{P}$
（異なると予想する人が多い）

チャーチ・チューリングのテーゼ

数学的にハッキリした主張

ボンヤリした主張

チューリング機械で解ける ⇔ 「実際に計算できる」

- 現実の「計算」に出て来そうな手順は
確かにチューリング機械で実現できる（経験上）
- 他の色々なやり方で定義された計算可能性の概念とも一致
一般再帰函数、ラムダ計算、…

チューリングの論文

Turing, A.M. (1936). "On Computable Numbers, with an Application to the Entscheidungsproblem". *Proceedings of the London Mathematical Society* 2, **42**: 230–265.

あらゆる処理手順がこの機械の動きとして書けそうな理由を第9節で論じている

9. *The extent of the computable numbers.*

No attempt has yet been made to show that the "computable" numbers include all numbers which would naturally be regarded as computable. All arguments which can be given are bound to be, fundamentally, appeals to intuition, and for this reason rather unsatisf The real question at issue is "What are the possibl carried out in computing a number?"

チューリング機械が
計算という行為全般を
捉えていると思える理由

The arguments which I shall use are of three kinds.

(*a*) A direct appeal to intuition.

(*b*) A proof of the equivalence of two definitions (in case the new definition has a greater intuitive appeal).

(*c*) Giving examples of large classes of numbers which are computable.

Once it is granted that computable numbers are all "computable", several other propositions of the same character follow. In particular, it follows that, if there is a general process for determining whether a formula of the Hilbert function calculus is provable, then the determination can be carried out by a machine.

I. [Type (a)]. This argument is only an elaboration of the ideas of § 1.

Computing is normally done by writing certain symbols on paper. We may suppose this paper is divided into squares like a child's arithmetic book. In elementary arithmetic the two-dimensional character of the paper is sometimes used. But such a use is always avoidable, and I think that it will be agreed that the two-dimensional character of p [illegible] of computation. I assume then that the computatio [illegible] one-dimensional paper, *i.e.* on a tape divided into squa [illegible] I shall also suppose that the number of symbols which may be printed is finite. If we were to allow an infinity of symbols, then there would be symbols differing to an arbitrarily small extent†. The effect of this restriction of the number of symbols is not very serio [illegible] o use sequences of symbols in the place of singl [illegible] c numeral such as

記号は有限個と
考えてよかろう

無限個使ったら区別
できなくなるから

---

† If we regard a symbol as literally printed on a square we may suppose that the square is $0 \leqslant x \leqslant 1$, $0 \leqslant y \leqslant 1$. The symbol is defined as a set of points in this square, viz. the set occupied by printer's ink. If these sets are restricted to be measurable, we can define the "distance" between two symbols as the cost of transforming one symbol into the other if the cost of moving unit area of printer's ink unit distance is unity, and there is an infinite supply of ink at $x = 2$, $y = 0$. With this topology the symbols form a conditionally compact space.

17 or 999999999999999 is normally treated as a single symbol. Similarly in any European language words are treated as single symbols (Chinese, however, attempts to have an enumerable infinity of symbols). The differences from our point of view betw is that the compound symbols, if they at one glance. This is in accordance with e ence. We cannot tell at a glance whether 9999999999999999 and 999999999999999 are the same.

大きな数値を一記号と考えると
やはり一瞬では区別できない

The behaviour of the computer at any moment is determined by the symbols which he is observing, and his "state of mind" at that moment. We may suppose that there is a bound B to the number of symbols or squares which the co moment. If he wishes to observe more, he mu ons. We will also suppose that the number of sta taken into account is finite. The reasons for this are of the same character as those which restrict the number of symbols. If we admitted an infinity of states of mind, some of them will be "arbitrarily close" and will be confused. Again, the restriction is not one which seriously affects computation, since the use of more complicated states of mind can be a ls on the tape.

「計算者」の状態も
有限としてよかろう

Let us imagine the operat r to be split up into "simple operations" wh is not easy to imagine them further divided. Every such operation consists of some change

一度に読み書きするのは
一字だけとしてよかろう

計算ってデジタル・離散的なものだけなの？

「強いチャーチ・チューリングのテーゼ」

チューリング機械で解ける ⇔ 「実際に計算できる」

どんな物理現象を使っても？

計算効率についても？

NO → チューリング機械ではできない計算ができる → すごい！

？？？

YES → どんな物理現象もチューリング機械で高速に計算（予測）できる → すごい！

アナログとデジタル

実数値を表現するために

アナログ　物理的な量をそのまま用いる

デジタル　何らかの離散値に切り捨て

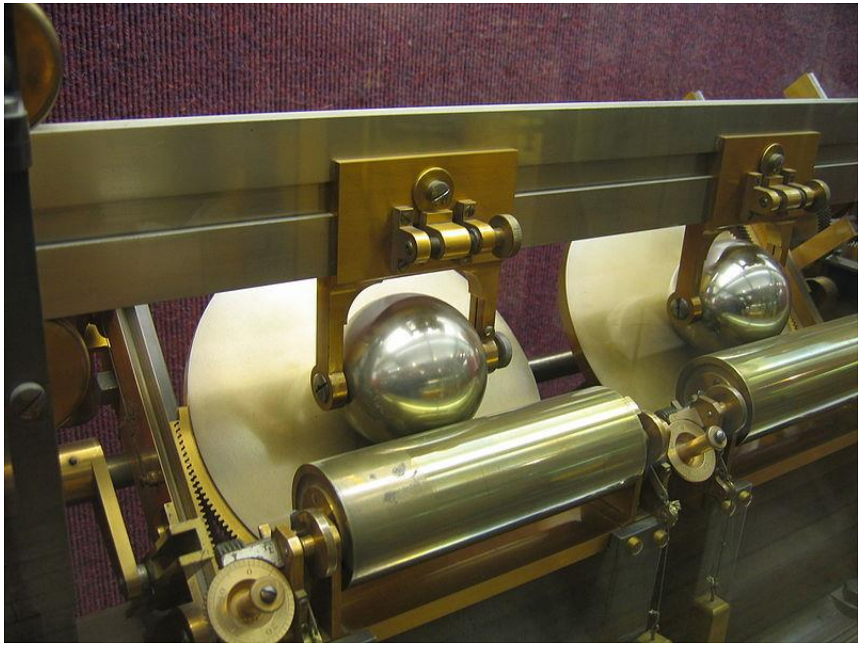
時刻⇔針の位置

デジタル表現の特徴

予め決められた離散的な値しか表せない
保存・複製・誤り訂正などが容易

「アナログのままで計算すればよいのでは？」

アナログ計算機

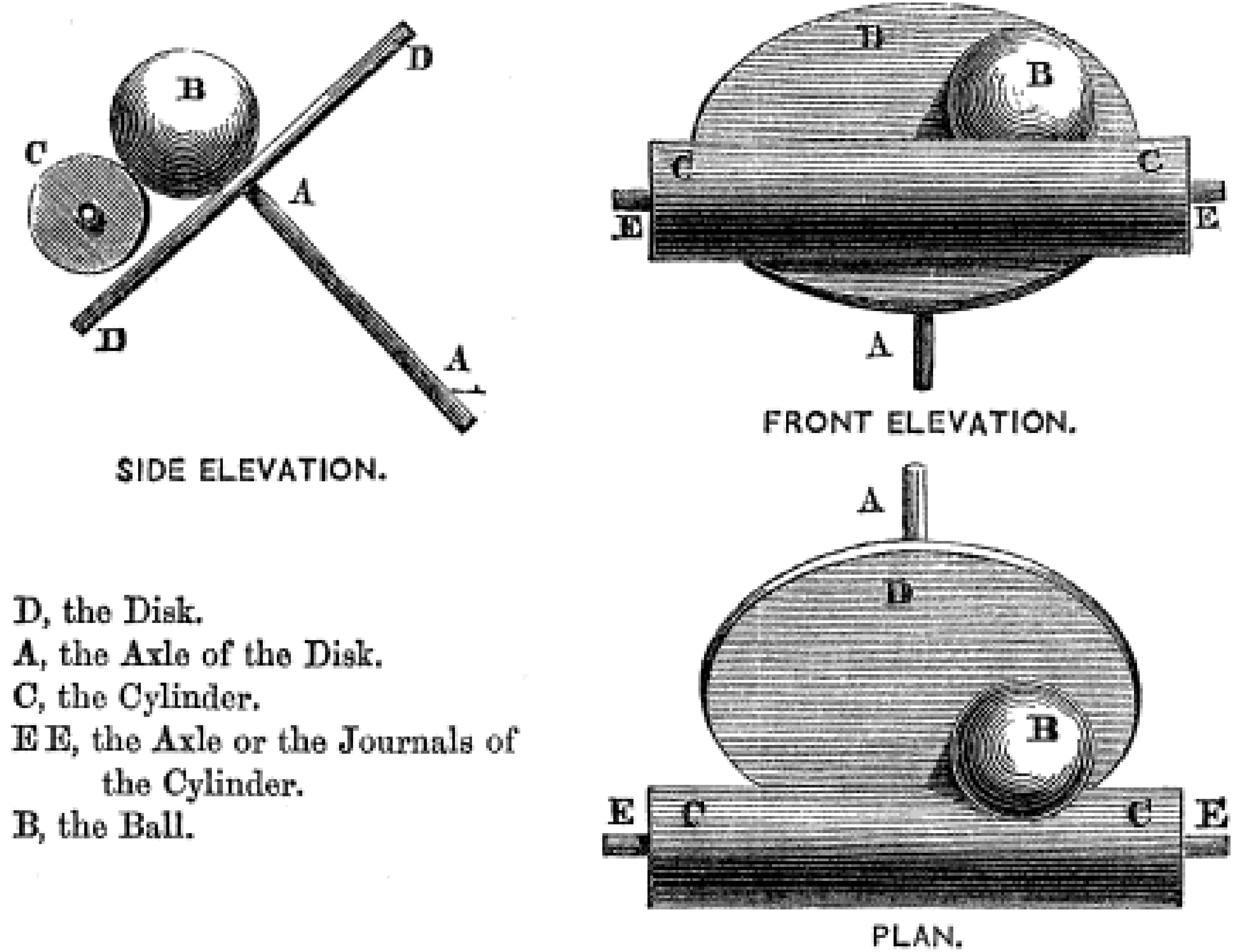


Creative Commons Attribution 3.0 Unported
https://commons.wikimedia.org/wiki/File:Harmonic_analyser_disc_and_sphere.jpg

微分解析機
（Differential Analyzer）

微分方程式を解くアナログ計算機
加算機・乗算器・積分器などを適当に接続
1870年代にジェームズ＆ウィリアム・トムソンが構想
1930年頃に実現　1950年代まで使われる

積分器の原理の例　$\int f\,\mathrm{d}x$

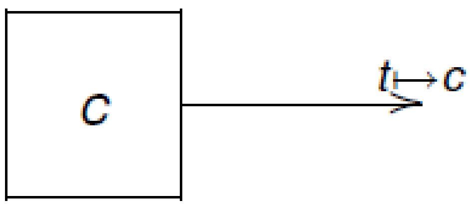


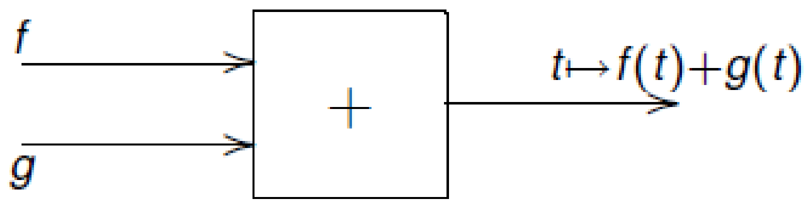
J・トムソンの論文（1876）より

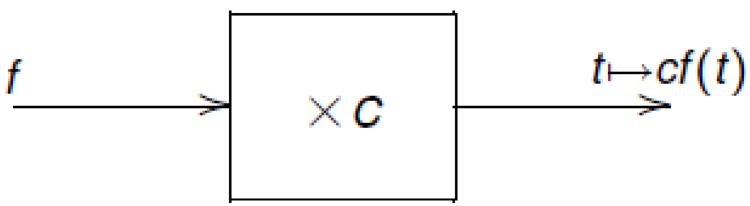
汎用アナログ計算機（GPAC）

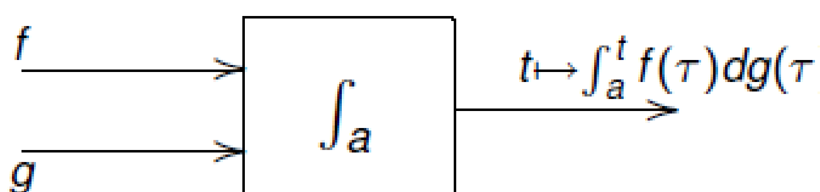
微分解析機の数理モデル（シャノン 1941）

次の素子から成る有向グラフ（フィードバックあり）

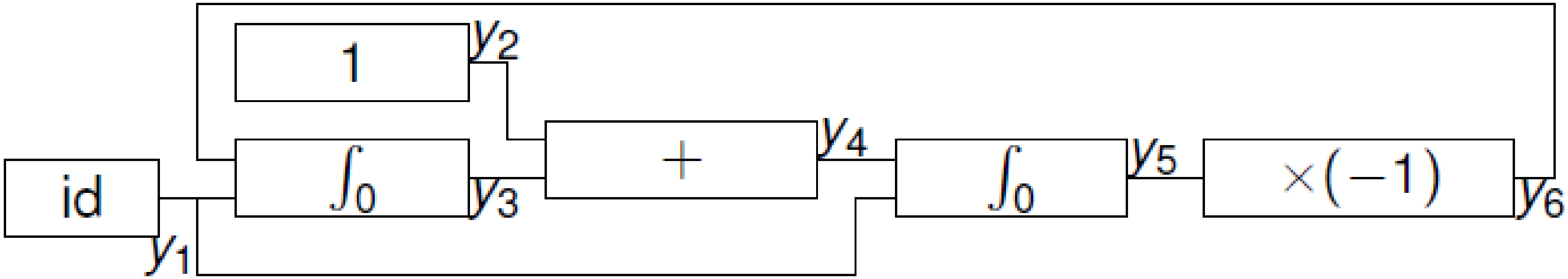
- 定数：


- 加算器：


- 定数倍器：


- 積分器：


例



$$\begin{cases} y_1(t) = t \\ y_2(t) = 1 \\ y_3(t) = \int_0^t y_6(\tau)\, dy_1(\tau) \\ y_4(t) = y_2(t) + y_3(t) \\ y_5(t) = \int_0^t y_4(\tau)\, dy_1(\tau) \\ y_6(t) = -y_5(t) \end{cases} \quad \therefore \quad \begin{cases} y_1(t) = t \\ y_2(t) = 1 \\ y_3(t) = \cos t - 1 \\ y_4(t) = \cos t \\ y_5(t) = \sin t \\ y_6(t) = -\sin t \end{cases}$$

数値への操作

「四則演算・ビット演算は単位時間」？

定理［PS76, S79］

整数の「加算」「乗算」「指定ビットの取得」を許すと多項式時間でも **PSPACE** の問題がすべて解ける

$x_1 = 10$ （二進数）

$x_2 = x_1 * x_1$ 　100

$x_3 = x_2 * x_2$ 　10000

$x_4 = x_3 * x_3$ 　100000000

$\vdots$

膨大な桁数を並列に利用して計算ができる（詳細略）

一演算を単位時間というのは「現実的」ではない

正確な演算ができるとして有理数の除算と下位ビットとしても同じ

[PS76] V. R. Pratt and L. J. Stockmeyer. On the computational complexity of ordinary differential equations. *J. Comput. Syst. Sci.* 12, 198-221, 1976.
[S79] A. Schönhage. On the power of random access machines. In *Proc. ICALP*, 520–529, 1979.

実数の表現

整数など　→文字列で表示（零型）

実数など　→文字列から文字列への函数で表現（一型）

例：

$\boxed{\text{実数 } t}$

n ↑　　↓ t の 2^{-n} 近似

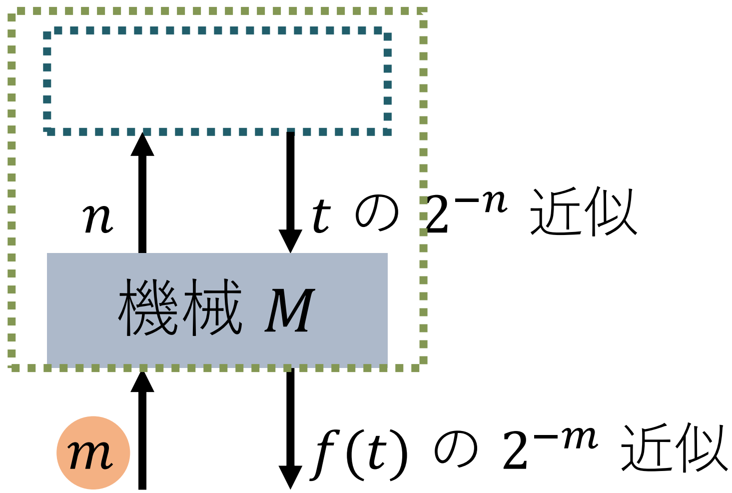
（距離 2^{-n} 以内にある有理数）

我々が扱いたいのは「二型」の計算

実函数の計算

「 t が任意の正確さでわかるとき
$f(t)$ を任意の正確さで求められるか？ 」

計算可能な
函数は連続



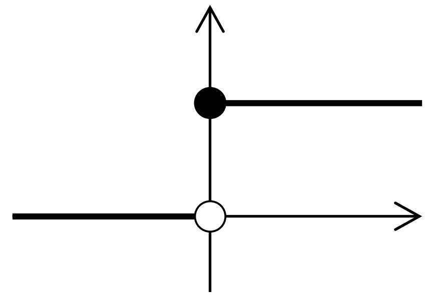
函数 $f\colon [0,1] \to \mathbf{R}$ を計算する機械 M

このように定義される（デジタルな意味での）
「多項式時間計算可能」を以下 **P** で表す
空間 **PSPACE**

m についての

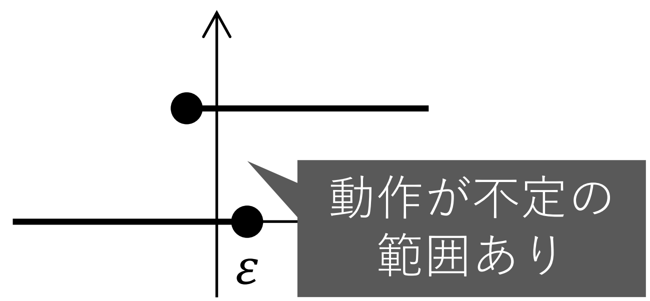
計算と連続性

$x \geq 0$? を判定する関数は連続でない（ので計算不能）



計算可能な関数は連続

多値関数による判定は計算可能



定理［BH98］

このような多値関数を比較命令として用いた計算はチューリング機械でも効率的に模倣できる

[BH98] V. Brattka and P. Hertling. Feasible real random access machines. *J. Complexity* 162, 490–526, 1998.

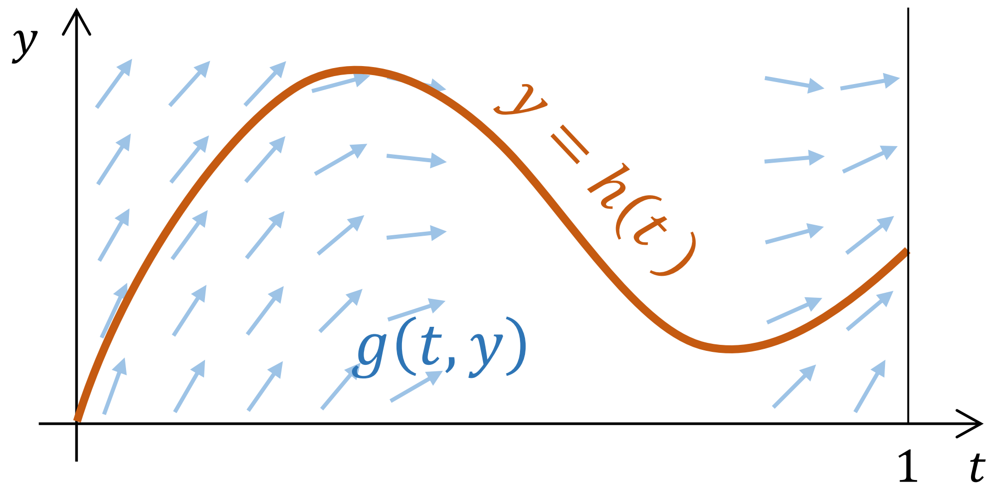
計算と連続性

∀ε∃δ　入力のδ近似から
出力のε近似が

連続	決る
計算可能	計算できる
多項式時間計算可能など	たやすく計算できる

微分方程式

単純な
$g\colon [0,1] \times \mathbf{R}^d \to \mathbf{R}$
から　複雑な
$h\colon [0,1] \to \mathbf{R}^d$
が生じ得るか？



$$h(0) = 0, \qquad h'(t) = g\big(t, h(t)\big)$$

定理［K83, K10］

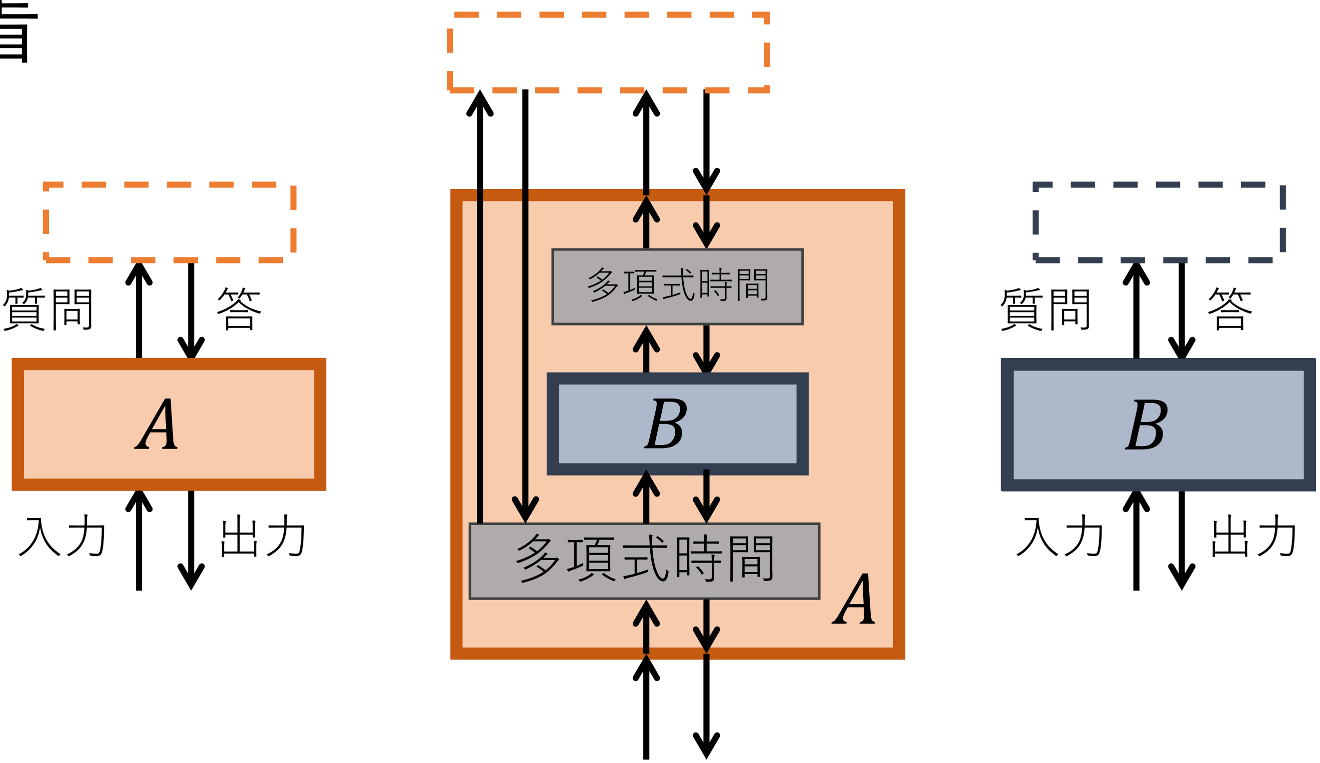
リプシッツ連続な函数 g が **P** のとき
上記の方程式の解 h は
- 常に **PSPACE** である
- **PSPACE 完全**である場合がある

どういう意味か？

PSPACEの中で最難

[K83] K. Ko. On the computational complexity of ordinary differential equations. *Inform. Contr.* 58, 1983.
[K10] A. Kawamura. Lipschitz continuous ordinary differential equations are polynomial-space complete. *Comput. Complexity* 19, 2010.

帰着

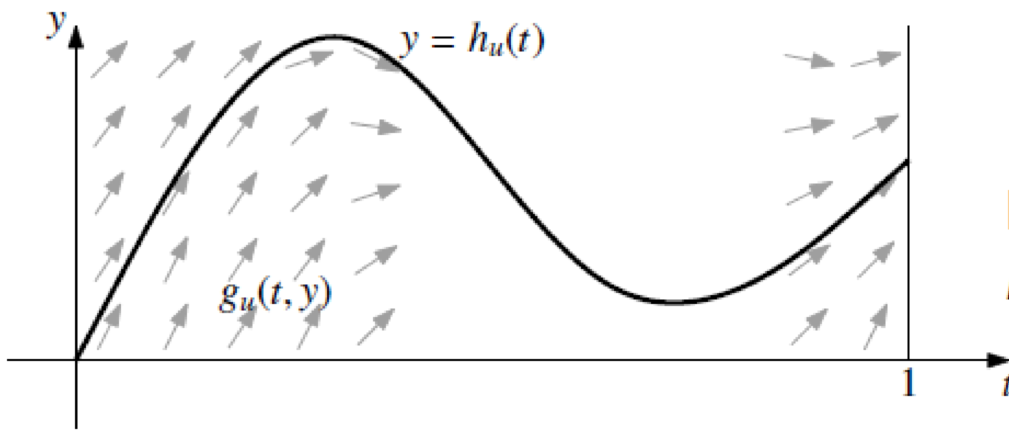


$$A \leq B$$

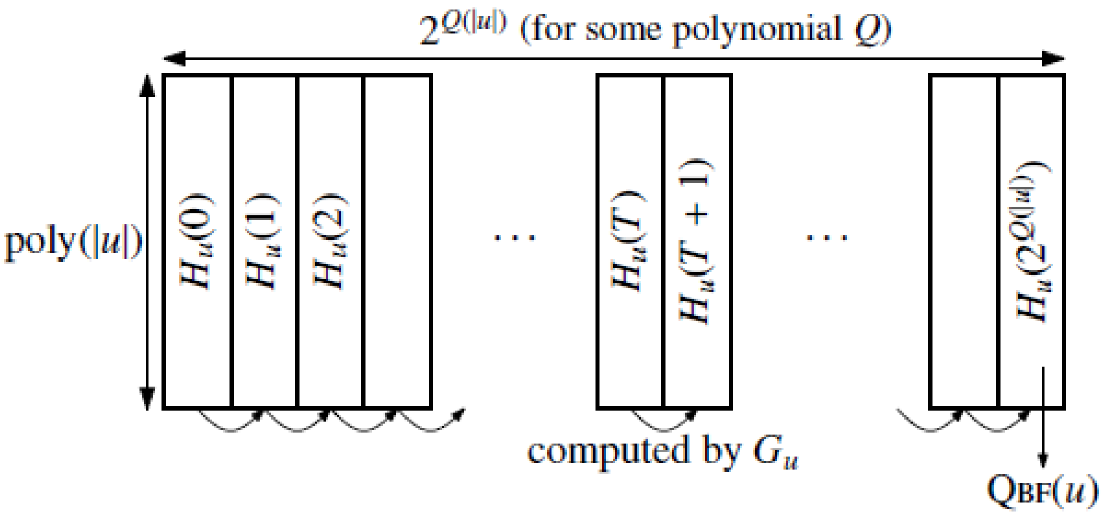
「A は B に多項式時間で帰着可能」

h が **PSPACE** 完全とは　**PSPACE** の任意の問題が h に帰着

微分方程式の計算量（PSPACE完全）

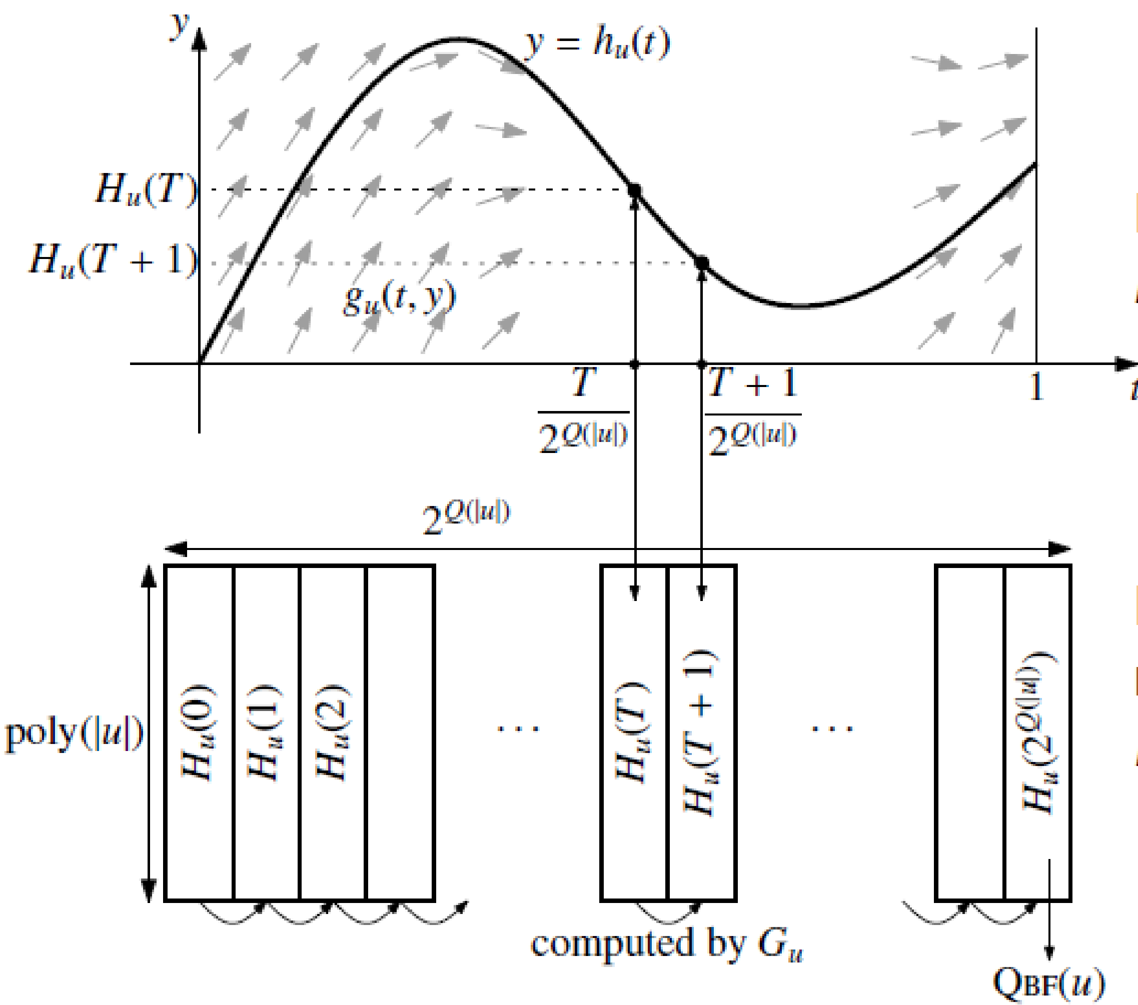


Differential equation:
$h'_u(t) = g_u(t, h(t))$



Description of a **PSPACE** machine on input u:
$H_u(T+1) = G_u(T, H_u(T))$

微分方程式の計算量（PSPACE完全）



Differential equation:
$h'_u(t) = g_u(t, h(t))$

Description of a **PSPACE** machine on input *u*:
$H_u(T+1) = G_u(T, H_u(T))$

滑らかな動きのみで
PSPACE計算を
模倣できる！

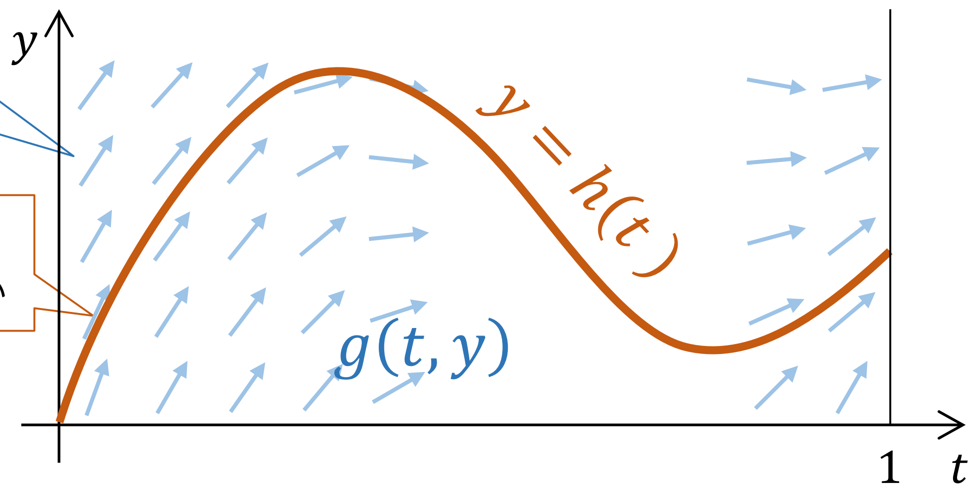
入出力は何か？

先述のPSPACE完全性は「方程式を
無制限なやりとりで指定できる」ことを前提としていた

与えられた (t, y) から
$g(t, y)$ を求めるのは易しいが

与えられた t から
$h(t)$ を求めるのは難しい

ような場合が存在すると判った



しかし実際に「作れる」物理系は
・方程式自体はいっぺんに指定できる比較的単純なもの
・初期値は自由に設定できる
というものであろう

…とは一体
何なのか？

「単純な」微分方程式とは何だろうか？
それに限定すればすべてが多項式時間に収まるのか？

定理［Pour-El 1974, Rubel 1987, Graça 2004］

適切な仮定を置くと　GPACで作れる函数は結局
g が**多項式**の場合の解 h となっているような函数のみ
このとき h は **P**

逆に「**P** の函数は皆 GPAC で近似できる」という結果もある［Bournezら 2007～13］

定理［MM93など］

g が解析的函数のとき h は **P**

$$f(z) = \sum_{j \in \mathbf{N}} a_j z^j$$

「自然界にあるのは解析函数だけ」？

[MM93] N. T. Müller and B. Moiske. Solving initial value problems in polynomial time. *Proc.* JAIIO-PANEL 283–293, 1993.

「単純な」微分方程式とは何だろうか？
それに限定すればすべてが多項式時間に収まるのか？

定理［Pour-El 1974, Rubel 1987, Graça 2004］

適切な仮定を置くと　GPACで作れる函数は結局
g が**多項式**の場合の解 h となっているような函数のみ
このとき h

逆に「**P** の函数は　007〜13］

定理［MM9

g が解析的函

互いに引力を及ぼし合う三点の運動

（衝突など特異点もあり得る）

$$\ddot{x}_i = -\sum_{j\neq i} m_j \cdot \frac{x_i - x_j}{\left|x_i - x_j\right|^3}$$

定理［KTZ17］

三体問題の或る特殊な場合について
与えられた初期値から指定時刻での値を求める問題は
初期値を単位立方体から一様に選ぶとき平均計算量は **P**

[KTZ17] A. Kawamura, H. Thies, and M. Ziegler. Average-case complexity for the n-body problem. *Proc. CCA*, 2017.

微分方程式の解の存在と計算量

$$h(0) = 0, \qquad h'(t) = g\big(t, h(t)\big)$$

コーシー・ペアノの定理

任意の g に対し解 h は（原点の近くで）存在

コーシー・リプシッツの定理

リプシッツ連続な g に対し 一意的な解 h が存在

コーシー・コワレフスカヤの定理

解析的な g に対し 解析的な解 h が存在

数学の様々な存在定理を精密化
（どの情報からどの情報が決るのか）

数値計算の問題の難しさを
計算可能性の視点から分類

コーシー・ペアノの定理（の不成立）@計算可能 [PR79]

$g \mapsto h$ は表現法δ_{func}の下で計算不可能

コーシー・リプシッツの定理@PSPACE

$g \mapsto h$は表現法$\delta_{\mathrm{func+L}}$の下で**PSPACE**

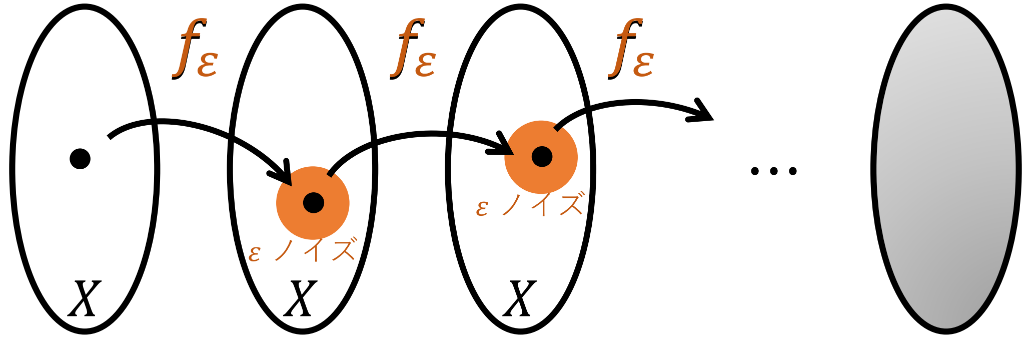
コーシー・コワレフスカヤ@P

$g \mapsto h$は表現δ_{an}の下で**P**

http://cca-net.de/
Computability and Complexity in Analysis

[PR79] M. B. Pour-El and I. Richards. A computable ordinary differential equation which possesses no computable solution. *Ann. Math. Log.* 17, 1979.

ノイズと予測可能性



計算可能距離空間上の
離散力学系 $f: X \to X$

エルゴード
定常分布

定理［GHR11］

f が **P** であっても定常分布は一般には計算不能

∵ チューリング機械を模倣できる

その挙動が予測できたとすると停止問題が解けてしまう

定理［BGR12］

f が **P** であり毎回一定量の**ノイズ**があると　定常分布は

- **PSPACE**
- 更に f も雑音も解析的ならば **P**

[GHR11] S. Galatolo, M. Hoyrup, and C. Rojas. Dynamics and abstract computability: Computing invariant measures. *Discrete and Continuous Dynamical Systems* 29, 193-212, 2011.

[BGR12] M. Braverman, A. Grigo, and C. Rojas. Noise vs computational intractability in dynamics. *Proc. ITCS*, 128-141, 2012.

まとめ

スライド・論文へのリンク
http://www.graco.c.u-tokyo.ac.jp/~kawamura/t/wbai/

計算理論の目標　「何が（短時間で）計算できるのか」

実数を扱う計算では特に
如何なる入出力・表現法・プロトコルで情報をやりとりする話なのか　明確にした上で論ずることが重要

資源や精度の制約をきちんと考えると
古典的計算モデルを大きく超えるわけではなさそう
とはいえ　その「根本的理由」はデジタル計算ほどには明らかでない

計算の限界（「NP完全・PSPACE完全問題は簡単に解けないはず」）は
物理的世界の情報処理をも支配する原理といえる？ [A05]
より精密に理解したい

[A05] S. Aaronson. NP-complete Problems and Physical Reality. *SIGACT News Complexity Theory Column*, 2005.