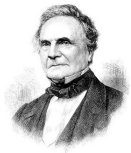
第三日
時間と空間の制限

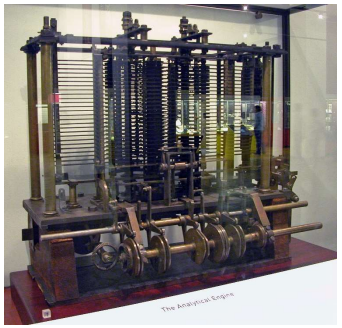
再

まとめ　第二日　機械の万能性と限界

- 機械は有限の文字列（算譜）で記述できる
- 入力された算譜を実行する計算ができる（EVAL は認識可能）
- しかし EVAL は判定可能ではない（対角線論法）
- 様々な言語の判定不能性が帰着により示される



As soon as an Analytic Engine exists, it will necessarily guide the future course of the science. Whenever any result is sought by its aid, the question will arise—By what course of calculation can these results be arrived at by the machine in the shortest time?
— Charles Babbage (1864)



解析機関（Analytic Engine）

解析機関は、実現すれば必ずや科学の発展を方向づけるものとなろう。この機械を用いて結果を得ようとするとき重要となる問がある――それは、**いかなる手順で計算すれば最も短時間で結果に辿り着くか**である。

「解析機関」『ウィキペディア日本語版』https://ja.wikipedia.org/w/index.php?title=%E8%A7%A3%E6%9E%90%E6%A9%9F%E9%96%A2&oldid=82653674）

計算量の考え方（原則）　1950～70年代

- チューリング機械での計算にかかる時間（遷移の回数）や空間（訪れる欄の数）を考える
  現実にかかる時間や空間をよく表している
- それが入力の長さに応じてどう変るか函数として表す
  「長さ n の入力なら必ず時間（や空間）が $T(n)$ 以内で済む」ような函数 T が計算量の尺度
- その函数の増大の速さに着目
  特に n の多項式以内か否かが重要

定義

機械 M が**多項式時間**であるとは　或る多項式 p が存在し
任意の長さ n の任意の入力に対する M の計算が
時間 $p(n)$ 以内に終る（遷移 $p(n)$ 回以下で停止する）ことをいう
言語 A を認識する多項式時間の機械 M が存在するとき
A は**多項式時間認識可能**であるという

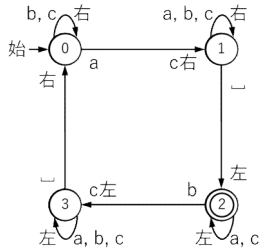
n の多項式以内である例

$$n \quad n\sqrt{n} \quad n^2 \log n \quad 5n^3 + 4n$$

n の多項式以内でない例

$$2^n \quad 2^{2^n} \quad 1.05^n \quad n! \quad n^{\log n}$$

例　第一日のこの機械は
時間 $(n+1)^2$ 以内に停止するので
MOREA は多項式時間認識可能



定義

機械 M が**多項式時間**であるとは　或る多項式 p が存在し
任意の長さ n の任意の入力に対する M の計算が
時間 $p(n)$ 以内に終る（遷移 $p(n)$ 回以下で停止する）ことをいう
言語 A を認識する多項式時間の機械 M が存在するとき
A は**多項式時間認識可能**であるという

「認識可能」の定義では $x \notin A$ のときは停止しないことを許していた
それに合せて「多項式時間認識」の定義は　長さ n の入力 x について
- $x \in A$ のとき M は x を時間 $p(n)$ 以内に受理する
- $x \notin A$ のとき M は x を時間 $p(n)$ 以内に受理しない

とすべきでは？

そう定義しても「多項式時間認識可能」の意味は変りません

受理せずに時間 $p(n)$ が経過したら不受理と確定できるので

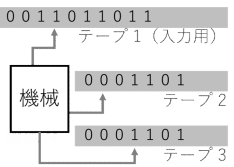
多項式時間では「認識可能」と「判定可能」の違いはない
（多項式時間認識可能な言語は補集合も多項式時間認識可能）

チャーチとチューリングの定立（続）

「現実的な手間で計算できる」 ＝ （チューリング機械で）多項式時間認識可能

チューリング機械の定義の細部は
計算法が多項式時間であるかないかに
影響を与えない

（「$0(n^3)$ であるかないか」など細かい量には
それなりに影響がある）



今後は一々チューリング機械を考えず　計算手順が解り易いように説明する

時間＝「基本的な操作の回数」と大雑把に考えてよい
- 四則演算やビットの操作など
- 機械語での命令数

なぜ多項式時間か否かが重要か

■ 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				
10^n	17分	3京年						
$n!$	3.6秒	800京年						

多項式時間

指数時間

1秒に100万回の処理ができるとしたときにかかる時間

実際には　多項式時間の中での速さの差
（n^2 と n^3 の違いなど）も勿論重要なのですが
この講義では「多項式時間であるかないか」という
より大きな違いに着目します

なぜ多項式時間か否かが重要か

■ 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 PRIME 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 の多項式時間で判定する方法が存在［AKS04］

問題 HAMILTON 与えられたグラフがハミルトン閉路をもつか判定
（全頂点を一度づつ通る辿り方）

頂点の並べ方（$n!$ 個ある）を全部調べれば判る
n の多項式時間で判定する方法があるかどうかは判らない（多分なさそう）

［AKS05］ M. Agrawal, N. Kayal and N. Saxena. PRIMES is in P. *Annals of Mathematics*, 160: 781–793, 2004.

定義

機械 M が**多項式空間**であるとは　或る多項式 p が存在し
任意の長さ n の任意の入力に対する M の計算が
空間 $p(n)$ 以内である（テープ上で初めの欄の左右 $p(n)$ 個の範囲に留まる）ことをいう
言語 A を認識する多項式空間の機械 M が存在するとき
A は**多項式空間認識可能**であるという

定理

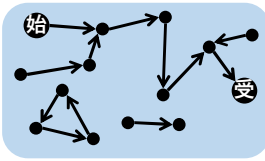
多項式時間認識可能 $\overset{①}{\Longrightarrow}$ 多項式空間認識可能 $\overset{②}{\Longrightarrow}$ 指数時間認識可能

多項式 q が存在して
入力長 n のとき $2^{q(n)}$ 時間以内

① 空間を 1 使う（新たな欄に移動する）にも
時間 1 かかるので

② 多項式空間機械 M に対して多項式 q が存在し
長さ n の入力に対する M の状況として
あり得るものは $2^{q(n)}$ 個以内

受理するのであれば時間 $\leq 2^{q(n)}$ で受理する



状況間の遷移関係

問題 PRIME 入力 正の整数 X（を十進表示したもの）
答 X は素数か

問題 HAMILTON 入力 グラフ G
答 G にハミルトン閉路はあるか

※ PRIME は本当は多項式時間認識可能であることが今では判っているが

定理

PRIME や HAMILTON は多項式空間認識可能

n 桁の割り算ひとつひとつは容易（n の多項式時間）

入力
48581583386419 ÷
n 桁

2 = 24290791693209 あまり 1
3 = 16193861128806 あまり 1
⋮
6306457 = 7703467 あまり 0

48581583386418

指数的に多くの手間がかかるが前の行を覚えておく必要はない（同じ場所に上書きしてよい）

問題 PRIME 入力 正の整数 X（を十進表示したもの）
答 X は素数か

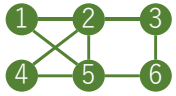
問題 HAMILTON 入力 グラフ G
答 G にハミルトン閉路はあるか

※ PRIME は本当は多項式時間認識可能であることが今では判っているが

定理

PRIME や HAMILTON は多項式空間認識可能

入力



n 頂点

辺がちゃんと繋がった経路になっているか確かめるのは容易

1 2 3 4 5 6　ダメ（3と4の間や6と1の間が繋がっていないので）
1 2 3 4 6 5　ダメ（3と4の間や4と6の間が繋がっていないので）
1 2 3 5 4 6
⋮
6 5 4 3 2 1
} 経路の候補 $n!$ 個

指数的に多くの手間がかかるが前の行を覚えておく必要はない（同じ場所に上書きしてよい）

問題 SR　入力　書換え規則の集合 R と文字列 w
　　答　R による書換えを次々と w に施して ε にできるか

問題 $SR_{\geq}$　入力　書換え規則の集合 R と文字列 w
但し R の各規則 $u \to v$ において $(u$ の長さ$) \geq (v$ の長さ$)$
　　答　R による書換えを次々と w に施して ε にできるか

問題 $SR^1_{\geq}$　入力　書換え規則の集合 R と文字列 w
但し R の各規則 $u \to v$ において $(u$ の長さ$) \geq (v$ の長さ$)$
　　答　R による書換えを次々と w に施すと
可能な書換え方は毎回一通りしかなく　やがて ε に達する

問題 $SR^1_{>}$　入力　書換え規則の集合 R と文字列 w
但し R の各規則 $u \to v$ において $(u$ の長さ$) > (v$ の長さ$)$
　　答　R による書換えを次々と w に施すと
可能な書換え方は毎回一通りしかなく　やがて ε に達する

困難そう（一般的な入力）　容易そう（特殊な入力）

問題 SR　入力　(R, w)
　　答　$w \Rightarrow_R^* \varepsilon$ か

問題 $SR_{\geq}$　SR を各規則が
$(旧長さ) \geq (新長さ)$
の場合に制限したもの

問題 $SR^1_{\geq}$　$SR_{\geq}$ を更に
可能な書換え方が毎回一通りな場合に制限したもの

問題 $SR^1_{>}$　$SR^1_{\geq}$ を更に各規則が
$(旧長さ) > (新長さ)$
の場合に制限したもの

定理（昨日）
SR は認識可能（だが判定可能でない）

？（次頁）

定理
$SR^1_{\geq}$ は多項式空間認識可能

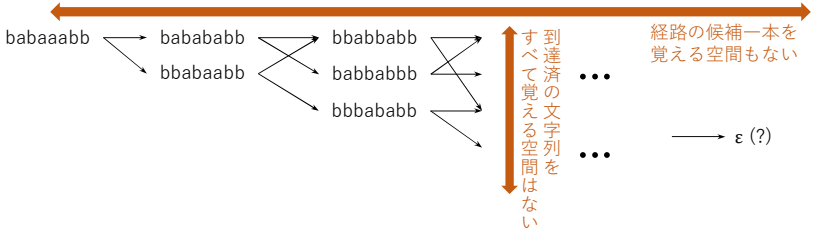
定理
$SR^1_{>}$ は多項式時間認識可能

書換えを実際に順次（機械のテープ上で）行ってみればよい

定理
$SR_{\geq}$ は多項式空間認識可能
※ $\Sigma = \{a, b\}$ の場合を考える（他のときも同様）

問題 $SR_{\geq}$　入力　(R, w) 但し各規則が $(旧長さ) \geq (新長さ)$
　　問　$w \Rightarrow_R^* \varepsilon$ か

入力の長さが n である（w の長さ $< n$）とき
w から書換えにより生じ得る文字列も長さ $< n$ であり　その個数は $< 2^n$
したがって $w \Rightarrow_R^{\leq 2^n} \varepsilon$ かどうか調べればよい
書換え 2^n 回以内で w から ε が得られる という意味
しかし　単純に長さ $\leq 2^n$ の経路すべてを調べることはできない



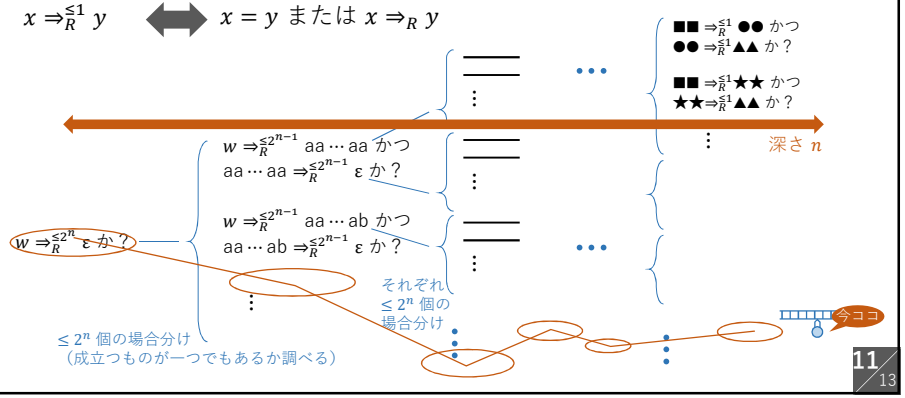
定理
$SR_{\geq}$ は多項式空間認識可能

問題 $SR_{\geq}$　入力　(R, w) 但し各規則が $(旧長さ) \geq (新長さ)$
　　問　$w \Rightarrow_R^* \varepsilon$ か

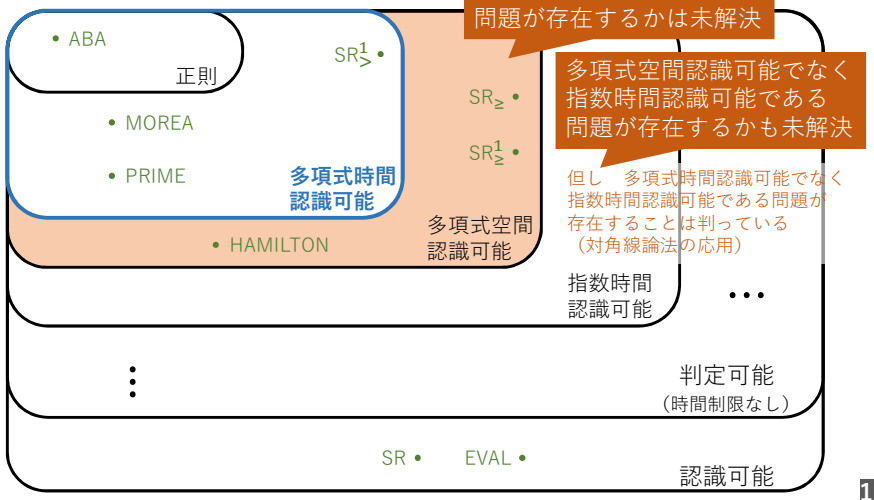
$w \Rightarrow_R^{\leq 2^n} \varepsilon$ かどうか　次の関係を再帰的に用いて調べればよい

$x \Rightarrow_R^{\leq 2^{i+1}} y$ ⟺ 或る $z \in \Sigma^{<n}$ が存在して $x \Rightarrow_R^{\leq 2^i} z$ かつ $z \Rightarrow_R^{\leq 2^i} y$

$x \Rightarrow_R^{\leq 1} y$ ⟺ $x = y$ または $x \Rightarrow_R y$



複雑さの階層　問題の難しさの分類



まとめ　第三日　時間と空間の制限

- 時間計算量は「入力長 n のとき時間 $T(n)$ 以内で計算できる」という形で測る（空間計算量も同様）
- 多項式時間　≒　現実的な手間での計算
- それに比べて多項式空間はいかにも強すぎて非現実的っぽい（指数個の調べ尽しができる）
- しかし「多項式時間認識可能 ≠ 多項式空間認識可能」は未証明（不可能性の証明は難しい）