STR 2020冬

文字列方程式 について

稲永 俊介

文字列方程式

- 定数記号集合 Σ と変数記号集合 V について，$(L, R) \in (\Sigma \cup \mathrm{V})^* \times (\Sigma \cup \mathrm{V})^*$ を文字列方程式と呼ぶ（通常 $L = R$ と表記する）．

例）$\Sigma = \{\mathrm{a}, \mathrm{b}\}$, $\mathrm{V} = \{u, x, y, z\}$ とする．文字列方程式

$$x\mathrm{a}uz\mathrm{a}u = yz\mathrm{b}x\mathrm{aab}y$$

は次の解 f (morphism) を持つ．

$\mathrm{f}(u) = \mathrm{bab}$, $\mathrm{f}(x) = \mathrm{abb}$, $\mathrm{f}(y) = \mathrm{ab}$, $\mathrm{f}(z) = \mathrm{ba}$ について

$$\mathrm{f}(x\mathrm{a}uz\mathrm{a}u) = \mathrm{f}(yz\mathrm{b}x\mathrm{aab}y) = \mathrm{abbababbaabab}$$

※ 一般に文字列方程式の解は複数存在しうる

文字列方程式の計算量の上界

アルゴリズム	計算量
Makanin 1977	Decidable
Jaffar 1990, Schulz 1990	4-NEXPTIME 非決定性チューリング機械で $2^{2^{2^{2^{p(n)}}}}$ 時間
Koscielski & Pacholski 1996	3-NEXPTIME 非決定性チューリング機械で $2^{2^{2^{p(n)}}}$ 時間
Plandowski & Rytter 1998	2-NEXPTIME 非決定性チューリング機械で $2^{2^{p(n)}}$ 時間
Gutierrez 1998	EXPSPACE 決定性チューリング機械で $2^{p(n)}$ 領域
Plandowski 1999	NEXPTIME 非決定性チューリング機械で $2^{p(n)}$ 時間
Plandowski 2004	PSPACE 決定性チューリング機械で $p(n)$ 領域
Jez 2016	PSPACE 決定性チューリング機械で $p(n)$ 領域

この世で最も複雑なアルゴリズムの1つ

n: 文字列多項式の記述長, $p(n)$: n の任意の多項式

文字列方程式の計算量の上界

アルゴリズム	計算量	
Makanin 1977	Decidable	Makanin
Jaffar 1990, Schulz 1990	4-NEXPTIME 非決定性チューリング機械で $2^{2^{2^{2^{p(n)}}}}$ 時間	
Koscielski & Pacholski 1996	3-NEXPTIME 非決定性チューリング機械で $2^{2^{2^{p(n)}}}$ 時間	
Plandowski & Rytter 1998	2-NEXPTIME 非決定性チューリング機械で $2^{2^{p(n)}}$ 時間	LZ分解
Gutierrez 1998	EXPSPACE 決定性チューリング機械で $2^{p(n)}$ 領域	Makanin
Plandowski 1999	NEXPTIME 非決定性チューリング機械で $2^{p(n)}$ 時間	LZ分解
Plandowski 2004	PSPACE 決定性チューリング機械で $p(n)$ 領域	
Jez 2016	PSPACE 決定性チューリング機械で $p(n)$ 領域	Recompression

n: 文字列多項式の記述長, $p(n)$: n の任意の多項式

文字列方程式の極小解の長さ

論文	極小解の長さ の上界
Makanin 1977	$2^{2^{2^{p(n)}}}$
Plandowski 1999	$2^{2^{p(n)}}$

n: 文字列多項式の記述長

知られている下界は $2^{p(n)}$

未解決問題（予想）とその帰結

予想１

記述長 n の文字列方程式の
極小解の長さの上界は $2^{p(n)}$ である.

予想２

文字列方程式の判定問題は NP に属する.

※文字列方程式は NP 困難 [Angluin, 1980]

定理１ [Plandowski & Rytter, 1998]

予想１ ➔予想２

系１

予想２ ➔ 文字列方程式は NP 完全

定理１の概要

定理１ [Plandowski & Rytter, 1998]

文字列方程式の極小解の長さ M の上界が $2^{p(n)}$ ならば，文字列方程式の判定問題は NP に属する.

補題１ [Plandowski & Rytter, 1998]

文字列方程式の極小解の LZ 分解のサイズは $O(n^2 \log^2 M \, (\log n + \log\log M))$ で抑えられる.

n: 文字列多項式の記述長，M: 極小解の長さ

定理１の概要

定理１ [Plandowski & Rytter, 1998]

文字列方程式の極小解の長さ M の上界が $2^{p(n)}$ ならば，文字列方程式の判定問題は NP に属する.

1. 極小解の候補の LZ 分解が与えられたとする.
2. 1. の LZ 分解を文法に変換する.
3. 2. で求めた文法を方程式の左右に代入した式を２つの文法とみなして，等価性を判定する.

➔ 補題1より，LZ分解のサイズは $O(\mathrm{poly}(n)\mathrm{polylog}(M))$
2. 3 は LZ分解のサイズの多項式時間.

補題 1 (再掲)

補題１ [Plandowski & Rytter, 1998]

文字列方程式の極小解の LZ 分解のサイズは
$O(n^2 \log^2 M \, (\log n + \mathrm{loglog}\, M))$ で抑えられる.

n: 文字列多項式の記述長, M: 極小解の長さ

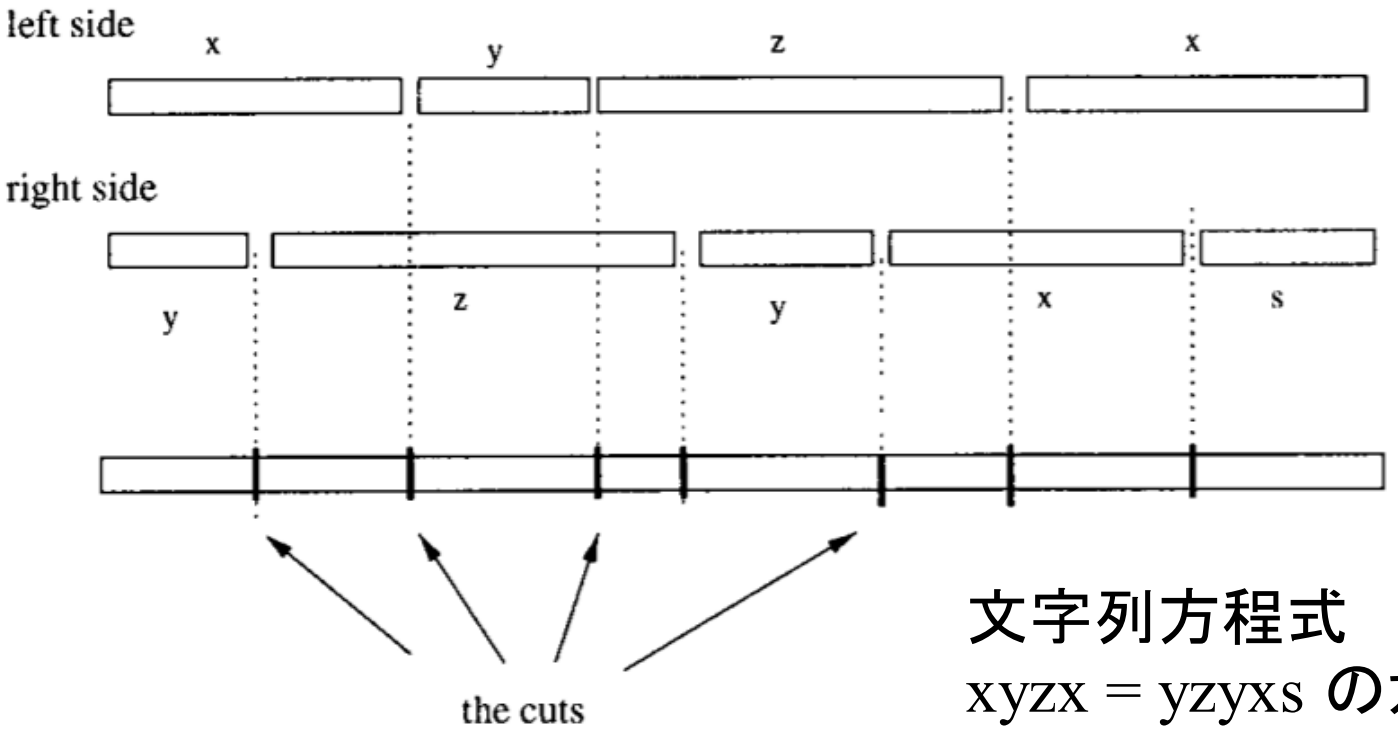
次に示す補題2を用いて，補題１を証明している.

直感的には，極小解は繰り返し構造を多く含むことを補題２で示している.

補題2

補題2 [Plandowski & Rytter, 1998]

文字列方程式 E の任意の極小解の任意の部分文字列は E の少なくとも1つのカットを含む，または触れている．



文字列方程式
xyzx = yzyxs のカット

文字列方程式と圧縮

- 文字列方程式の極小解は繰り返しを多く含む
  ➔ LZ分解を用いると，指数的に縮む.
- また Jez は，Recompression と呼ばれる
  LZ とは大きく異なる圧縮方法を用いて，
  文字列方程式を解くアルゴリズムを提案している.

- また，文字列方程式をデータ圧縮に応用する
  研究も一部で始まっている（詳細不明）.

Generalized Word Equations:
A New Approach to Data Compression
M. Kutwin, W. Plandowski, A. Zaroda, DCC 2019: 585

1変数の文字列方程式

- 変数の種類数を1つに限定し，
  両辺における変数の出現回数は任意とする．

例） $xx\mathrm{baababa} = \mathrm{ababa}x\mathrm{ab}x$

この文字列方程式は
解 $x = \mathrm{ababaababa}$ を持つ．

1変数の文字列方程式

- 変数の種類数を1つに限定すると,
  極小解の長さは方程式の長さ未満であることが知られている.

論文	1変数文字列方程式の極小解の長さの上界
Khmelevskii 1971	cn
Obono et al. 1994	$4n$ (証明なし)
Baba et al. 2003	$n-1$

n: 文字列多項式の記述長

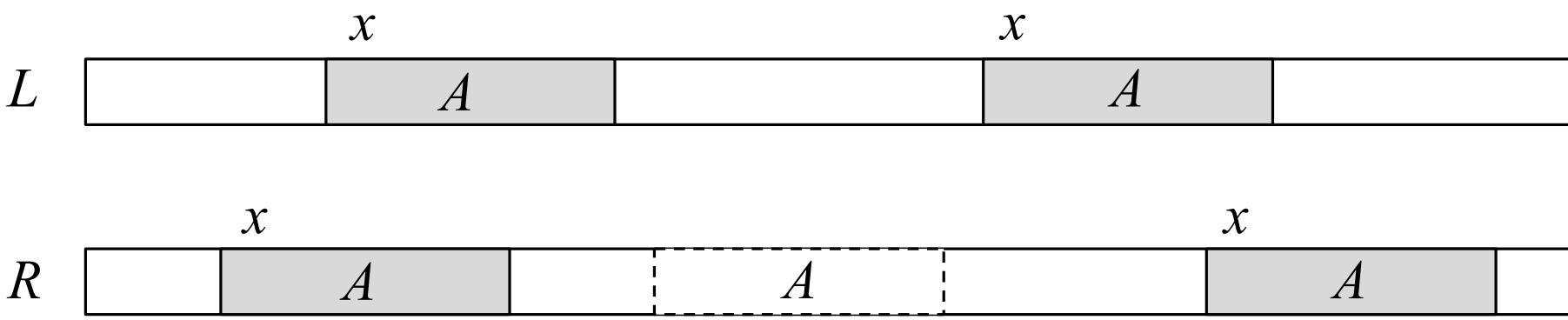
1変数の文字列方程式の解の長さ

- $Y \in (\Sigma \cup \{x\})^*$ 中の変数 x の出現回数を $\#_x(Y)$ と書く.
- $L = R$ を任意の1変数 文字列方程式とする.
  $\#_x(L) \neq \#_x(R)$ のとき，解は一意に決まり，
  その長さは n 未満である [Obono et al. 1994].
- よって，以降は $\#_x(L) = \#_x(R)$ の場合を考える.

1変数の文字列方程式の解の長さ

観察 [Baba et al. 2003]

A を1変数文字列方程式 $L = R$ の解とする.
L と R それぞれの k 番目の x の出現位置の差 d_k が $|A|$ 以下であるとき, A は d_k を周期に持つ.



1変数の文字列方程式の解の長さ

補題3 [Baba et al. 2003]

p を 文字列方程式 $L = R$ の解 A の周期の1つとする.
このとき $|A| \geq \max_{1 \leq k \leq m} d_k + p - 1$ ならば,
A の接頭辞 $A[1..|A| - p]$ もまた $L = R$ の解である.

ただし $m = \#_x(L) = \#_x(R)$

- 前述の観察と，周期性補題を用いて証明できる.

1変数の文字列方程式の解の長さ

補題4 [Baba et al. 2003]

$\#_x(L) = \#_x(R)$ を満たす文字列方程式 $L = R$ の極小解の長さは高々

$$\max_{1 \le k \le m} d_k + \min_{1 \le k \le m, d_k \neq 0} d_k - 2$$

- $|A| \geq \max_{1 \le k \le m} d_k + \min_{1 \le k \le m, d_k \neq 0} d_k - 1$ を満たす極小解 A が存在すると仮定する.
  観察より, A は周期 $p \leq \min_{1 \le k \le m, d_k \neq 0} d_k$ を持つ.
  補題3より, $A[1..|A| - p]$ もこの文字列方程式の解となるが, これは A の極小性に反する.

1変数の文字列方程式を解くアルゴリズム

アルゴリズム	計算時間
Charatonik & Pacholski 1991	$O(n^6)$
Obono et al. 1994	$O(n \log n)$
Dabrowski & Plandowski 2011	$O(n + \#_x \log n)$
Jez 2016	$O(n)$

n: 文字列多項式の記述長, $\#_x$: 変数 x の出現回数

各アルゴリズムの概要 1

1変数文字列方程式 E: $A_0 x A_1 x \ldots x A_r = x B_1 x \ldots x B_r$

- Obono et al. 1994: $O(n \log n)$
  - 解の周期性を利用.
  - u, v を原始的 (primitive) な文字列とし, $|uv| \leq |A_0 B_1|$ を満たすとする.
  - $(uv)^k u$ が文字列方程式 E の解であるようなすべての k を $O(n)$ 時間で求められる.
  - u と v の組の候補は $O(\log n)$ 個.

各アルゴリズムの概要2

1変数文字列方程式 E: $A_0 x A_1 x \ldots x A_r = x B_1 x \ldots x B_r$

- Dabrowski & Plandowski 2011: $O(n + \#_x \log n)$
  - Obono et al. のアルゴリズムの改良版.
  - 文字列方程式の定数部分の文字列集合 $S = \{A_0, A_1, \ldots, A_r, B_1, \ldots, B_r\}$ を前処理して, 解の検証を高速化する.
  - S に対する Aho-Corasik オートマトンを使って, S の Prefix Table を $O(n)$ 時間で構築.
  - 解1つあたり, $O(\#_x)$ 時間で検証可能.

各アルゴリズムの概要3

1変数文字列方程式 E: $A_0 x A_1 x \ldots x A_r = x B_1 x \ldots x B_r$

- Jez 2016: $O(n)$
  - 前述の2つのアルゴリズムとは異なり,
    解の組合せ的性質は使わない.
  - Recompression のアルゴリズムによって,
    方程式と解の候補の中の2グラムを
    新しい文字に置き換えていく.
  - $O(n + \#_x \log n)$ 時間の手法を得たのち,
    色々工夫して $O(n)$ 時間にする.
  - 論文は50ページ...

２変数の文字列方程式

論文	２変数文字列方程式の解の長さ
Ilie & Plandowski 2000	$\|x\| \leq 2n$ $\|y\| \leq 2n^2$

アルゴリズム	計算時間
Charatonik & Pacholski 1991	$O(n^{100})$
Ilie & Plandowski 2000	$O(n^6)$
Dabrowski & Plandowski 2004	$O(n^5)$

n: 文字列多項式の記述長

これから何をやるべきか？やれそうか？

- Recompression を使わずに1変数文字列方程式を $O(n)$ 時間で解くアルゴリズム？
- 2変数文字列方程式を $o(n^5)$ 時間で解くアルゴリズム？

- 3変数の文字列方程式の解の長さは $O(n^3)$？
- 究極的には，予想1：
  「文字列方程式の極小解の長さの上界は $2^{p(n)}$」が示せると素晴らしい.