未決定文字列における欠如単語の検索の困難さ

Dominik Köppl [1] and Jannik Olbrich [2]

[1]: 山梨大学, [2]: University of Ulm

山梨県若手研究者奨励事業費補助金 2291 による助成研究

$\widetilde{T} = \mathtt{A}\begin{Bmatrix}\mathtt{A}\\ \mathtt{C}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{G}\\ \mathtt{T}\end{Bmatrix}$, 欠如単語: AAA, CCC, CCA, . . .

第 200 回アルゴリズム研究発表会 [15]

欠如単語

定義 (欠如単語 = absent word)

- T: 文字列, Σ: アルファベット
- T の欠如単語 X: 文字列 T 中に部分文字列として出現しない文字列
- ただし X の長さは 1 以上

例

- $T = \mathtt{AACA}$, $\Sigma = \{\mathtt{A}, \mathtt{C}\}$
- MAWs: $\mathtt{AAA}$, $\mathtt{CAA}$, $\mathtt{CAC}$, $\mathtt{CC}$ (最短の MAW)

定義 (最小欠如単語 = MAW (minimum absent word))

- X : 欠如単語
- X は最小欠如単語 :$\Leftrightarrow$ X のすべての真の部分文字列が T 中に出現

未決定文字列

定義

- 任意の位置に複数の代替文字を持つ
- 位置を記号と呼ぶ

- $\widetilde{T}$: 未決定文字列
- $\widetilde{T}[i] \subset \Sigma$

例

IUPAC 表記法は DNA や RNA 配列における未決定の核酸を表現、例：

- R は A または G
- N は任意の核酸

$$\widetilde{T} = \mathtt{A}\begin{Bmatrix}\mathtt{A}\\ \mathtt{C}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{G}\\ \mathtt{T}\end{Bmatrix}$$

$$\mathcal{L}(\widetilde{T}) = \{\mathtt{AACG}, \mathtt{AACT}, \mathtt{ACCG}, \mathtt{ACCT}\}$$

- $\widetilde{T}$: 未決定文字列の例
- $\mathcal{L}(\widetilde{T})$: $\widetilde{T}$ の言語
- $\mathcal{L}(\widetilde{T})$ は $\widetilde{T}$ で表現されている文字列の集合

未決定文字列の MAW

定義 (欠如単語)

- $\widetilde{T}$: 未決定文字列
- $\widetilde{T}$ の欠如単語 X: すべての $\mathcal{L}(\widetilde{T})$ の文字列中に部分文字列として出現しない文字列
- ただし X の長さは 1 以上

$$\widetilde{T} = \mathtt{A}\begin{Bmatrix}\mathtt{A}\\\mathtt{C}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{G}\\\mathtt{T}\end{Bmatrix}$$

$$\mathcal{L}(\widetilde{T}) = \{\mathtt{AACG}, \mathtt{AACT}, \mathtt{ACCG}, \mathtt{ACCT}\}$$

MAW: AAA, CCC, CCA, ...

定義 (MAW)

- X : 欠如単語
- X は最小欠如単語 :⇔ X のすべての真の部分文字列が任意の $\mathcal{L}(\widetilde{T})$ の文字列中に出現

計算量

- 単純な文字列： $\mathcal{O}(n\sigma)$ 時間 Crochemore+'16
- 単純な文字列： $\mathcal{O}(n+M)$ 時間 Fujishige+'16
- 文字列の集合： $\mathcal{O}(n+M)$ 時間 Okabe+'23,
  ただし n: 要素の文字列の長さの合計
- 未決定文字列： NP 完全 (一つの MAW の検索だけ)

ただし

- $M \leq \sigma n$: MAW の個数 Crochemore+'16
- ※ $|\widetilde{T}| = n$ としても、 $|\mathcal{L}(\widetilde{T})| = \Omega(2^n)$ になる可能性がある

例：

$$\widetilde{T} = \begin{Bmatrix} \mathtt{a} \\ \mathtt{b} \end{Bmatrix} \begin{Bmatrix} \mathtt{a} \\ \mathtt{b} \end{Bmatrix} \begin{Bmatrix} \mathtt{a} \\ \mathtt{b} \end{Bmatrix} \begin{Bmatrix} \mathtt{a} \\ \mathtt{b} \end{Bmatrix} \cdots$$

問題 (k-欠如単語問題)

- 入力： 未決定文字列 $\widetilde{T}$ と整数 k
- 出力： 長さが最大 k の欠如単語が $\widetilde{T}$ に存在するかどうか

- k-欠如単語問題は判定問題
- 判定問題の答えは Yes・No

例

$$\widetilde{T} = \mathtt{A}\begin{Bmatrix}\mathtt{A}\\\mathtt{C}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{G}\\\mathtt{T}\end{Bmatrix}$$

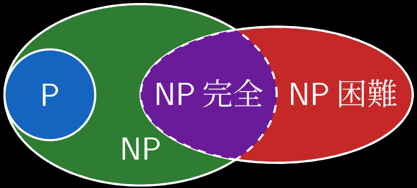
$$\mathcal{L}(\widetilde{T}) = \{\mathtt{AACG}, \mathtt{AACT}, \mathtt{ACCG}, \mathtt{ACCT}\}$$

- $k = |\widetilde{T}| + 1$: いつでも Yes
- $k = 4$: Yes: $\mathtt{ACCA}$
- $k = 3$: Yes: $\mathtt{AAA}$
- $k = 2$: Yes: $\mathtt{GC}$
- $k = 1$: No

問題 (k-欠如単語問題)

- 入力： 未決定文字列 $\tilde{T}$ と整数 k
- 出力： 長さが最大 k の欠如単語が $\tilde{T}$ に存在するかどうか

- k-欠如単語問題は判定問題
- 判定問題の答えは Yes・No



方針

- 欠如単語問題 $\in$ NP を証明
- 欠如単語問題 $\in$ NP 困難 を証明

$\Rightarrow$ 欠如単語問題 $\in$ NP 完全

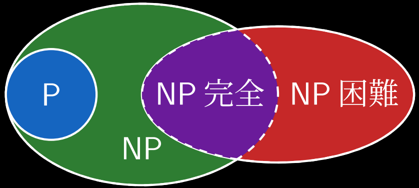
NP 問題

定義 (クラス NP)

- 問題 M は以下の条件を満たすと、NP の要素となる
- 入力に対する正しい出力が Yes であるとき、それを多項式時間で検証するための証拠が存在

問題 (証明検証)

- X: 文字列, $\tilde{T}$: 未決定文字列
- X は $\tilde{T}$ の *MAW* の一つかどうか

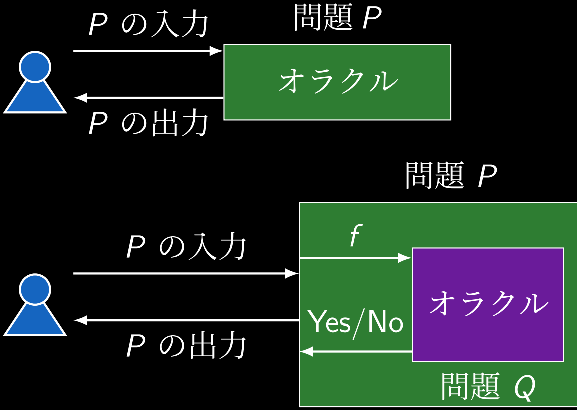


- X が $\tilde{T}$ に出現するかどうか、多項式時間で判断 Abrahamson'87
- X のすべての部分文字列の個数は $O(n^2)$
- 証明検証問題を多項式時間で解ける

$\Rightarrow$ 欠如単語問題 は NP 問題

定義 (Karp の帰着方法)

- 入力：判定問題 P と Q、多項式時間アルゴリズム f
- $f(P$ の入力$) = Q$ の入力
- P の入力 S が P の Yes 入力 $\Leftrightarrow$ $f(S)$ は Q の Yes 入力
- その際、 P が Q に多項式時間多対一帰着可能であると言う



補題

- P は NP 困難
- P が Q に多項式時間多対一帰着可能

$\Rightarrow$ Q も NP 困難

- 欠如単語問題 は NP 困難 だと証明したい $\Rightarrow Q =$ 欠如単語問題
- $P =$ 3-SAT とする

定義 (3-SAT 判定問題)

入力:

- 連言標準形（CNF: conjunctive normal form）の式 F
- F は一連の節 C_i を連言（AND）で結合
- 各節 C_i は 3 つのリテラルの選言である
- n 個の変数 $x_1, \ldots, x_n$ が順序付け

出力：論理式全体の値を真にするような真偽値 $x_1, \ldots, x_n$ があるかどうか

3-SAT 判定問題は NP 困難 Karp'72

例：

- 変数 x_1, x_2, x_3, x_4
- $C_1 = x_1 \vee \neg x_2 \vee x_3$
- $C_2 = \neg x_1 \vee x_2 \vee \neg x_4$
- $C_3 = x_2 \vee x_3 \vee \neg x_4$
- $F = C_1 \wedge C_2 \wedge C_3$

出力： Yes:

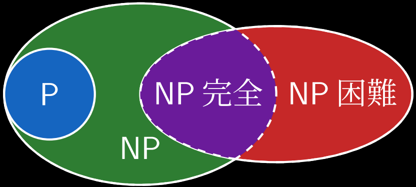
- $x_1 = x_2 =$ 真

$\Rightarrow F =$ 真

方針

- 線形量を取る未決定文字列 $\tilde{S}$ で F の反対を表現
- つまり、C_i を満たさない真偽を $\tilde{S}$ の長さ n の部分文字列で表現
- 長さ $n-1$ のすべての可能な部分文字列を $\tilde{S}$ で列挙

$\Rightarrow$ F が充足する割り当てを持つ $\Leftrightarrow$ $\tilde{S}$ の最短の MAW の長さは n



$\widetilde{T}_j$ の定義

任意節 $C_j = (\ell_a \vee \ell_b \vee \ell_c)$ に対して：

- ℓ_i が変数 x_i のリテラル
- $v_i = \begin{cases} 0 & \text{ただし } \ell_i = x_i \\ 1 & \text{ただし } \ell_i \neq x_i \end{cases}$
- 以下の未決定文字列 $\widetilde{T}_j$ を定義

$$\widetilde{T}_j = \left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\}^{a-1} \{ v_a \} \left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\}^{b-a-1} \{ v_b \} \left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\}^{c-b-1} \{ v_c \} \left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\}^{n-c}.$$

例

- 変数: x_1, x_2, x_3, x_4
- $C_j = (x_1 \vee \neg x_2 \vee x_3)$
- $\widetilde{T}_j = \{0\}\{1\}\{0\}\left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\}$

$\widetilde{S}$ の定義

$$\widetilde{S} = \widetilde{T_1}\,\{\$\}\ldots\{\$\}\,\widetilde{T_m}\,\{\$\}\begin{Bmatrix}0\\1\\\$\end{Bmatrix}^{n-1}\{\$\}\begin{Bmatrix}0\\1\end{Bmatrix}^{n-1}.$$

- $\widetilde{T}_j$ は n の記号を持つ

$\Rightarrow$ $\widetilde{S}$ は $\mathcal{O}(nm)$ の記号を持つ

- $|\widetilde{S}[i]| \leq 3 \Rightarrow \widetilde{S}$ は F を $\mathcal{O}(nm)$ 領域で表現

例

$$F = (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee \neg x_3 \vee x_4),$$

$$\widetilde{S} = \{0\}\{1\}\{0\}\begin{Bmatrix}0\\1\end{Bmatrix}\{\$\}\begin{Bmatrix}0\\1\end{Bmatrix}\{0\}\{1\}\{0\}\{\$\}\begin{Bmatrix}0\\1\\\$\end{Bmatrix}^{3}\{\$\}\begin{Bmatrix}0\\1\end{Bmatrix}^{3}.$$

例の MAW

$$F = (x_1 \lor \neg x_2 \lor x_3) \land (x_2 \lor \neg x_3 \lor x_4),$$

$$\widetilde{S} = \{0\}\{1\}\{0\}\begin{Bmatrix}0\\1\end{Bmatrix}\{\$\}\begin{Bmatrix}0\\1\end{Bmatrix}\{0\}\{1\}\{0\}\{\$\}\begin{Bmatrix}0\\1\\\$\end{Bmatrix}^3\{\$\}\begin{Bmatrix}0\\1\end{Bmatrix}^3.$$

- 1000 は $(x_1 = 1, x_2 = x_3 = x_4 = 0)$ を表現
- 長さ 3 のすべての文字列が $\widetilde{S}$ に出現
- 1000 は MAW (特に、最短の MAW)

帰着定理

$$\widetilde{S} = \widetilde{T_1}\,\{\,\$\,\}\ldots\{\,\$\,\}\,\widetilde{T_m}\,\{\,\$\,\}\begin{Bmatrix}0\\1\\\$\end{Bmatrix}^{n-1}\{\,\$\,\}\begin{Bmatrix}0\\1\end{Bmatrix}^{n-1}, \Sigma = \{0, 1, \$\}$$

定理

F が充足する割り当てが持つ $\Leftrightarrow$ $\widetilde{S}$ の最短の MAW の長さは n

- 長さが最大 $n-1$ のすべての文字列が $\widetilde{S}$ 中に出現

$\Rightarrow$ 欠如単語は少なくとも長さ n を持つ

- 任意長さ n を持つ \$ を含む文字列が最後の 3 つの記号にも出現

$\Rightarrow$ 長さ n の MAW があれば、\$ を含まない

帰着定理の証明 帰り道 $\Rightarrow$

$$\widetilde{S} = \widetilde{T_1}\,\{\,\$\,\}\ldots\{\,\$\,\}\,\widetilde{T_m}\,\{\,\$\,\}\left\{\begin{matrix}0\\1\\\$\end{matrix}\right\}^{n-1}\{\,\$\,\}\left\{\begin{matrix}0\\1\end{matrix}\right\}^{n-1}, \Sigma = \{0,1,\$\}$$

定理

F が充足する割り当てが持つ $\Leftrightarrow$ $\widetilde{S}$ の最短の MAW の長さは n

証明.

- 仮定：CNF を充足する割り当てがある
- 割り当てをビット配列 $B[1..n]$ で表現する
- i 番目のビット $B[i]$ は x_i の真偽値を表現 (偽 $=0$、真 $=1$)
- B は $\widetilde{T_i}$ に不一致すると、B の割り当てが節が C_i を満たす

$\Rightarrow$ B は $\widetilde{S}$ の MAW

帰着定理の証明 行き道 $\Leftarrow$

$$\widetilde{S} = \widetilde{T_1}\,\{\,\$\,\}\ldots\{\,\$\,\}\,\widetilde{T_m}\,\{\,\$\,\}\left\{\begin{matrix}0\\1\\\$\end{matrix}\right\}^{n-1}\{\,\$\,\}\left\{\begin{matrix}0\\1\end{matrix}\right\}^{n-1}, \Sigma = \{0, 1, \$\}$$

定理

F が充足する割り当てが持つ $\Leftrightarrow$ $\widetilde{S}$ の最短の MAW の長さは n

証明.

- 仮定：CNF を充足できない
- すべての長さ n のビット配列は $\widetilde{S}$ 中に出現
- $\widetilde{S}$ に長さが最大 n の欠如単語はない

$\square$

今後の展望

	単純な文字列	未決定文字列	GD-文字列	ED-文字列	2ED-文字列
MAW	線形	NP 完全			
SUS	線形	NP 完全 ※			
anti-power	N	?	NP 完全		
LPF	線形	N	N	NP 完全	
不一致	線形	N	N	?	NP 完全

- 単純な文字列: $S[i] \in \Sigma$
- 未決定文字列: $\widetilde{S}[i] \subset \Sigma$
- GD-文字列: $\forall i \ \exists k : \widetilde{S}[i] \subset \Sigma^k$
- ED-文字列: $\widetilde{S}[i] \subset \Sigma^*$
- 2ED-文字列: $\widetilde{S}[i] \subset$ ED-文字列

※：２０１番目のアルゴリズム研究会の課題

文字列の一般化の階段

- 単純な文字列: $T = \mathtt{AACG}$
- 未決定文字列: $\widetilde{T} = \mathtt{A}\begin{Bmatrix}\mathtt{A}\\ \mathtt{C}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{G}\\ \mathtt{T}\end{Bmatrix}$
- generalized degenerate (GD)-文字列:
$$\widetilde{T} = \mathtt{A}\begin{Bmatrix}\mathtt{AAC}\\ \mathtt{CAT}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{G}\\ \mathtt{T}\end{Bmatrix}$$
- elastic-degenerate (ED)-文字列:
$$\widetilde{T} = \mathtt{A}\begin{Bmatrix}\epsilon\\ \mathtt{CAT}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{GTCG}\\ \mathtt{T}\end{Bmatrix}$$
- 2ED-文字列: $\widetilde{T} = \begin{Bmatrix} T\begin{Bmatrix}\mathtt{ACT}\\ \mathtt{C}\end{Bmatrix}A \\ A\begin{Bmatrix}\mathtt{T}\\ \mathtt{AAG}\end{Bmatrix}\end{Bmatrix}\begin{Bmatrix}\epsilon\\ \mathtt{CAT}\end{Bmatrix}\mathtt{C}\begin{Bmatrix}\mathtt{GTCG}\\ \mathtt{T}\end{Bmatrix}$