Moje zainteresowania

Artur Jeż

18 X 2012

1. Układy równań nad zbiorami liczb naturalnych
2. Kompresja, równania w słowach, ...

Równania nad zbiorami liczb naturalnych

$$\varphi_j(X_1, \dots, X_n) = \psi_j(X_1, \dots, X_n) \quad j = 1, \dots, m$$

- $X_i \subseteq \mathbb{N}$
- operacje: $\cup$, $\cap$, $+$

Równania nad zbiorami liczb naturalnych

$$\varphi_j(X_1, \ldots, X_n) = \psi_j(X_1, \ldots, X_n) \quad j = 1, \ldots, m$$

- $X_i \subseteq \mathbb{N}$
- operacje: $\cup$, $\cap$, $+$

$$X + Y = \{x + y \mid x \in X, y \in Y\}$$

Równania nad zbiorami liczb naturalnych

$$\varphi_j(X_1, \dots, X_n) = \psi_j(X_1, \dots, X_n) \quad j = 1, \dots, m$$

- $X_i \subseteq \mathbb{N}$
- operacje: $\cup$, $\cap$, $+$

$$X + Y = \{x + y \mid x \in X, y \in Y\}$$

Przykład

- $X = \{0\} \cup (X + \{2\})$ $\quad X =$ zbiór liczb parzystych

Równania nad zbiorami liczb naturalnych

$$\varphi_j(X_1, \ldots, X_n) = \psi_j(X_1, \ldots, X_n) \quad j = 1, \ldots, m$$

- $X_i \subseteq \mathbb{N}$
- operacje: $\cup$, $\cap$, $+$

$$X + Y = \{x + y \mid x \in X, y \in Y\}$$

Przykład

- $X = \{0\} \cup (X + \{2\})$ $\quad X =$ zbiór liczb parzystych
- $X + \{1\} = (X + X) \cup \{2\}$ $\quad$ wiele rozwiązań, w tym $\{1\}$ oraz $\{1, 2, 3, \ldots\}$

Równania nad zbiorami liczb naturalnych

$$\varphi_j(X_1, \ldots, X_n) = \psi_j(X_1, \ldots, X_n) \quad j = 1, \ldots, m$$

- $X_i \subseteq \mathbb{N}$
- operacje: $\cup$, $\cap$, $+$

$$X + Y = \{x + y \mid x \in X, y \in Y\}$$

Przykład

- $X = \{0\} \cup (X + \{2\})$ $\quad X =$ zbiór liczb parzystych
- $X + \{1\} = (X + X) \cup \{2\}$ $\quad$ wiele rozwiązań, w tym $\{1\}$ oraz $\{1, 2, 3, \ldots\}$

Cel

- Charakteryzacja rozwiązań.

Układy równań języków formalnych

$$\begin{cases} \varphi_1(X_1,\ldots,X_n) & = & \psi_1(X_1,\ldots,X_n) \\ & \vdots & \\ \varphi_m(X_1,\ldots,X_n) & = & \psi_m(X_1,\ldots,X_n) \end{cases}$$

- X_i: podzbiór Σ^*.
- φ_i: zmienne, stałe, operacje na językach

$$L \cdot L' = \{ww' \mid w \in L, w' \in L'\}$$

Układy równań języków formalnych

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

- X_i: podzbiór Σ^*.
- φ_i: zmienne, stałe, operacje na językach

$$L \cdot L' = \{ww' \mid w \in L, w' \in L'\}$$

- prosty
- łatwo przyciąć do potrzeb
- unifikuje gramatyki i automaty

Układy równań w postaci nieuwikłanej: gramatyki

$$\begin{cases} X_1 &= \varphi_1(X_1, \ldots, X_n) \\ &\vdots \\ X_n &= \varphi_n(X_1, \ldots, X_n) \end{cases}$$

- Ginsburg i Rice $(\cup, \cdot)$: gramatyki bezkontekstowe

Układy równań w postaci nieuwikłanej: gramatyki

$$\begin{cases} X_1 & = & \varphi_1(X_1, \ldots, X_n) \\ & \vdots & \\ X_n & = & \varphi_n(X_1, \ldots, X_n) \end{cases}$$

- Ginsburg i Rice ($\cup, \cdot$): gramatyki bezkontekstowe
  - rozwiązanie $(S_1, \ldots, S_n)$ jest najmniejsze: $S_i \subseteq S_i'$, dla każdego rozwiązania $(S_1', \ldots, S_n')$

Układy równań w postaci nieuwikłanej: gramatyki

$$\begin{cases} X_1 &= \varphi_1(X_1, \ldots, X_n) \\ &\vdots \\ X_n &= \varphi_n(X_1, \ldots, X_n) \end{cases}$$

- Ginsburg i Rice ($\cup$, $\cdot$): gramatyki bezkontekstowe
  - rozwiązanie $(S_1, \ldots, S_n)$ jest najmniejsze: $S_i \subseteq S'_i$, dla każdego rozwiązania $(S'_1, \ldots, S'_n)$
- rozszerzone przez Okhotina ($\cap$, $\cup$ i $\cdot$): gramatyki koniunkcyjne

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\Sigma = \{a\}$.

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\Sigma = \{a\}$.
- nieuwikłane $\{\cup, \cdot\}$: tylko języki regularne

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\Sigma = \{a\}$.
- nieuwikłane $\{\cup, \cdot\}$: tylko języki regularne
- $\{\cdot, {}^c\}$: przykład języka nieregularnego [Leiss 1994]

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\Sigma = \{a\}$.
- nieuwikłane $\{\cup, \cdot\}$: tylko języki regularne
- $\{\cdot, {}^c\}$: przykład języka nieregularnego [Leiss 1994]
- nieuwikłane $\{\cup, \cap, \cdot\}$: ?

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\Sigma = \{a\}$.
- nieuwikłane $\{\cup, \cdot\}$: tylko języki regularne
- $\{\cdot, {}^c\}$: przykład języka nieregularnego [Leiss 1994]
- nieuwikłane $\{\cup, \cap, \cdot\}$: ?

- jedyna informacja: długość $a^n \longleftrightarrow$ liczba n

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\mathbb{N}$
- nieuwikłane $\{\cup, \cdot\}$: okresowe
- $\{\cdot, {}^c\}$: nieokresowe [Leiss 1994]
- nieuwikłane $\{\cup, \cap, +\}$: ?

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\mathbb{N}$
- nieuwikłane $\{\cup, \cdot\}$: okresowe
- $\{\cdot, {}^c\}$: nieokresowe [Leiss 1994]
- nieuwikłane $\{\cup, \cap, +\}$: ?

- wyrażenia arytmetyczne (wyrażenie regularne)
- obwody arytmetyczne
- automaty
- gramatyki

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\mathbb{N}$
- nieuwikłane $\{\cup, \cdot\}$: okresowe
- $\{\cdot, {}^c\}$: nieokresowe [Leiss 1994]
- nieuwikłane $\{\cup, \cap, +\}$: ?

- wyrażenia arytmetyczne (wyrażenie regularne)
- obwody arytmetyczne
- automaty
- gramatyki
- układy równań

Języki formalne i zbiory liczb naturalnych

Prosty przypadek

- $\mathbb{N}$
- nieuwikłane $\{\cup, \cdot\}$: okresowe
- $\{\cdot, {}^c\}$: nieokresowe [Leiss 1994]
- nieuwikłane $\{\cup, \cap, +\}$: ?

- wyrażenia arytmetyczne (wyrażenie regularne)
- obwody arytmetyczne
- automaty
- gramatyki
- układy równań

$$L_X \cdot L_Y \implies X + Y = \{x + y \mid x \in X, y \in Y\}$$

Przykład

$$
\begin{aligned}
X_1 &= (X_2 + X_2 \cap X_1 + X_3) \cup \{1\}\\
X_2 &= (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\}\\
X_3 &= (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\}\\
X_{12} &= X_3 + X_3 \cap X_1 + X_2
\end{aligned}
$$

Przykład

$$X_1 = (X_2 + X_2 \cap X_1 + X_3) \cup \{1\}$$
$$X_2 = (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\}$$
$$X_3 = (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\}$$
$$X_{12} = X_3 + X_3 \cap X_1 + X_2$$

- Liczby w zapisie k-pozycyjnym: ciągi cyfr z $\Sigma_k = \{0, 1, \dots, k-1\}$.
- $(a_1 \dots a_\ell)_k$: liczba zapisana jako $a_1 \dots a_\ell$ w notacji k-pozycyjnej
- zbiór liczb $\longleftrightarrow$ język formalny nad Σ_k

Przykład

$$\begin{aligned} X_1 &= (X_2 + X_2 \cap X_1 + X_3) \cup \{1\} \\ X_2 &= (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\} \\ X_3 &= (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\} \\ X_{12} &= X_3 + X_3 \cap X_1 + X_2 \end{aligned} \qquad ((10^*)_4, (20^*)_4, (30^*)_4, (120^*)_4)$$

- Liczby w zapisie k-pozycyjnym: ciągi cyfr z $\Sigma_k = \{0, 1, \ldots, k-1\}$.
- $(a_1 \ldots a_\ell)_k$: liczba zapisana jako $a_1 \ldots a_\ell$ w notacji k-pozycyjnej
- zbiór liczb $\longleftrightarrow$ język formalny nad Σ_k

Przykład

$$X_1 = (X_2 + X_2 \cap X_1 + X_3) \cup \{1\}$$
$$X_2 = (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\}$$
$$X_3 = (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\}$$
$$X_{12} = X_3 + X_3 \cap X_1 + X_2$$

$$((10^*)_4, (20^*)_4, (30^*)_4, (120^*)_4)$$

- Liczby w zapisie k-pozycyjnym: ciągi cyfr z $\Sigma_k = \{0, 1, \dots, k-1\}$.
- $(a_1 \dots a_\ell)_k$: liczba zapisana jako $a_1 \dots a_\ell$ w notacji k-pozycyjnej
- zbiór liczb $\longleftrightarrow$ język formalny nad Σ_k

- $X_2 + X_2 = (20^*)_4 + (20^*)_4 = (10^+)_4 \cup (20^*20^*)_4$

Przykład

$$X_1 = (X_2 + X_2 \cap X_1 + X_3) \cup \{1\}$$
$$X_2 = (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\}$$
$$X_3 = (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\}$$
$$X_{12} = X_3 + X_3 \cap X_1 + X_2$$

$$((10^*)_4, (20^*)_4, (30^*)_4, (120^*)_4)$$

- Liczby w zapisie k-pozycyjnym: ciągi cyfr z $\Sigma_k = \{0, 1, \ldots, k-1\}$.
- $(a_1 \ldots a_\ell)_k$: liczba zapisana jako $a_1 \ldots a_\ell$ w notacji k-pozycyjnej
- zbiór liczb $\longleftrightarrow$ język formalny nad Σ_k

- $X_2 + X_2 = (20^*)_4 + (20^*)_4 = (10^+)_4 \cup (20^*20^*)_4$
- $X_1 + X_3 = (10^*)_4 + (30^*)_4 =$

Przykład

$$X_1 = (X_2 + X_2 \cap X_1 + X_3) \cup \{1\}$$
$$X_2 = (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\}$$
$$X_3 = (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\}$$
$$X_{12} = X_3 + X_3 \cap X_1 + X_2$$

$$((10^*)_4, (20^*)_4, (30^*)_4, (120^*)_4)$$

- Liczby w zapisie k-pozycyjnym: ciągi cyfr z $\Sigma_k = \{0, 1, \ldots, k-1\}$.
- $(a_1 \ldots a_\ell)_k$: liczba zapisana jako $a_1 \ldots a_\ell$ w notacji k-pozycyjnej
- zbiór liczb $\longleftrightarrow$ język formalny nad Σ_k

- $X_2 + X_2 = (20^*)_4 + (20^*)_4 = (10^+)_4 \cup (20^* 20^*)_4$
- $X_1 + X_3 = (10^*)_4 + (30^*)_4 = (10^+)_4 \cup$

Przykład

$$X_1 = (X_2 + X_2 \cap X_1 + X_3) \cup \{1\}$$
$$X_2 = (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\}$$
$$X_3 = (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\}$$
$$X_{12} = X_3 + X_3 \cap X_1 + X_2$$

$$((10^*)_4, (20^*)_4, (30^*)_4, (120^*)_4)$$

- Liczby w zapisie k-pozycyjnym: ciągi cyfr z $\Sigma_k = \{0, 1, \ldots, k-1\}$.
- $(a_1 \ldots a_\ell)_k$: liczba zapisana jako $a_1 \ldots a_\ell$ w notacji k-pozycyjnej
- zbiór liczb $\longleftrightarrow$ język formalny nad Σ_k

- $X_2 + X_2 = (20^*)_4 + (20^*)_4 = (10^+)_4 \cup (20^*20^*)_4$
- $X_1 + X_3 = (10^*)_4 + (30^*)_4 = (10^+)_4 \cup (10^*30^*)_4 \cup (30^*10^*)_4$

Przykład

$$X_1 = (X_2 + X_2 \cap X_1 + X_3) \cup \{1\}$$
$$X_2 = (X_{12} + X_2 \cap X_1 + X_1) \cup \{2\}$$
$$X_3 = (X_{12} + X_{12} \cap X_1 + X_2) \cup \{3\}$$
$$X_{12} = X_3 + X_3 \cap X_1 + X_2$$

$$((10^*)_4, (20^*)_4, (30^*)_4, (120^*)_4)$$

- Liczby w zapisie k-pozycyjnym: ciągi cyfr z $\Sigma_k = \{0, 1, \ldots, k-1\}$.
- $(a_1 \ldots a_\ell)_k$: liczba zapisana jako $a_1 \ldots a_\ell$ w notacji k-pozycyjnej
- zbiór liczb $\longleftrightarrow$ język formalny nad Σ_k

- $X_2 + X_2 = (20^*)_4 + (20^*)_4 = (10^+)_4 \cup (20^*20^*)_4$
- $X_1 + X_3 = (10^*)_4 + (30^*)_4 = (10^+)_4 \cup (10^*30^*)_4 \cup (30^*10^*)_4$
- $(X_2 + X_2) \cap (X_1 + X_3) = (10^+)_4$

Uogólnienie przykładu

Twierdzenie (Równania nieuwikłane)

Dla automatu kratowego M istnieje nieuwikłany system z $\{\cup, \cap, +\}$ *o najmniejszym rozwiązaniu* $(L(M))_k$.

Uogólnienie przykładu

Twierdzenie (Równania nieuwikłane)

Dla automatu kratowego M istnieje nieuwikłany system z $\{\cup, \cap, +\}$ o najmniejszym rozwiązaniu $(L(M))_k$.

- Zamknięte na $\cap$, $\cup$ i dopełnienie
- zawiera wiele języków, np. gramatyki liniowe, kodowawnie obliczeń maszyn Turinga itp.

Uogólnienie przykładu

Twierdzenie (Równania nieuwikłane)

Dla automatu kratowego M istnieje nieuwikłany system z $\{\cup, \cap, +\}$ o najmniejszym rozwiązaniu $(L(M))_k$.

- Zamknięte na $\cap$, $\cup$ i dopełnienie
- zawiera wiele języków, np. gramatyki liniowe, kodowawnie obliczeń maszyn Turinga itp.

Twierdzenie (Równania ogólne)

$S \subseteq \mathbb{N}$ jest jedynym (najmniejszym, największym) rozwiązaniem układu równań w postaci uwikłanej, z operacjami $\{\cup, +\}$ lub $\{\cap, +\}$

wtedy i tylko wtedy

S jest rekurencyjny (rek. przeliczalny, ko-rek. przeliczalny)

Algorytmy dla skompresowanych danych

- Coraz wiecej danych jest przechowywana (transmitowana) w skompresowanej formie
- Chcemy na nich wciąż operować

Algorytmy dla skompresowanych danych

- Coraz wiecej danych jest przechowywana (transmitowana) w skompresowanej formie
- Chcemy na nich wciąż operować

Zapotrzebowania na algorytmy działające na skompresowanej formie. Czas działania.

Algorytmy dla skompresowanych danych

- Coraz wiecej danych jest przechowywana (transmitowana) w skompresowanej formie
- Chcemy na nich wciąż operować

Zapotrzebowania na algorytmy działające na skompresowanej formie. Czas działania.

Lokalna rekompresja

- Nowa technika: rekompresja.
- Liczne zastosowania.
- Przykład: równania w słowach.

Równania w słowach

Definicja

Dla równania $U = V$, gdzie $U, V \in (\Sigma \cup \mathcal{X})^*$. Czy istnieje wartościowania $S : \mathcal{X} \mapsto \Sigma^*$ spełniające to równanie?

Równania w słowach

Definicja

Dla równania $U = V$, gdzie $U, V \in (\Sigma \cup \mathcal{X})^*$. Czy istnieje wartościowania $S : \mathcal{X} \mapsto \Sigma^*$ spełniające to równanie?

A. Markow Początek badań.

- Podejrzenie: nierozstrzygalne ($\geq_{rec}$ 10. pr. Hilberta).

Równania w słowach

Definicja

Dla równania $U = V$, gdzie $U, V \in (\Sigma \cup \mathcal{X})^*$. Czy istnieje wartościowania $S : \mathcal{X} \mapsto \Sigma^*$ spełniające to równanie?

A. Markow Początek badań.

- Podejrzenie: nierozstrzygalne ($\geq_{rec}$ 10. pr. Hilberta).

G. Makanin '77 Spełnialność równań słów jest rozstrzygalna.

- System przepisywania.
- Długie, skomplikowane, duża złożoność.
- poprawiona przez 20 lat (Gutiérrez EXPSPACE '98)

Równania w słowach

Definicja

Dla równania $U = V$, gdzie $U, V \in (\Sigma \cup \mathcal{X})^*$. Czy istnieje wartościowania $S : \mathcal{X} \mapsto \Sigma^*$ spełniające to równanie?

A. Markow Początek badań.

- Podejrzenie: nierozstrzygalne ($\geq_{rec}$ 10. pr. Hilberta).

G. Makanin '77 Spełnialność równań słów jest rozstrzygalna.

- System przepisywania.
- Długie, skomplikowane, duża złożoność.
- poprawiona przez 20 lat (Gutiérrez EXPSPACE '98)

W. Plandowski & W. Rytter '98 kompresja, algorytm poly$(n, \log N)$

W. Plandowski '99 Podwójnie wykładnicze oszacowanie na N

W. Plandowski '99 Algorytm w PSPACE

- system przepisywania, nietrywialna własność stopu,
- dowód skomplikowany, lecz rozsądnie krótki.

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$a\ a\ a\ b\ a\ b\ c\ a\ b\ a\ b\ b\ a\ b\ c\ b\ a$$

$$a\ a\ a\ b\ a\ b\ c\ a\ b\ a\ b\ b\ a\ b\ c\ b\ a$$

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$\color{red}{a\ a\ a}\ b\ a\ b\ c\ a\ b\ a\ b\ b\ a\ b\ c\ b\ a$$

$$\color{red}{a\ a\ a}\ b\ a\ b\ c\ a\ b\ a\ b\ b\ a\ b\ c\ b\ a$$

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$a_3 \quad b\ a\ b\ c\ a\ b\ a\ b\ b\ a\ b\ c\ b\ a$$

$$a_3 \quad b\ a\ b\ c\ a\ b\ a\ b\ b\ a\ b\ c\ b\ a$$

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$a_3 \; b \; a \; b \; c \; a \; b \; a \; b_2 \; a \; b \; c \; b \; a$$

$$a_3 \; b \; a \; b \; c \; a \; b \; a \; b_2 \; a \; b \; c \; b \; a$$

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$a_3 \quad b \quad d \quad c \quad d \quad a \quad b_2 \quad d \quad c \; b \; a$$

$$a_3 \quad b \quad d \quad c \quad d \quad a \quad b_2 \quad d \quad c \; b \; a$$

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$a_3 \quad b \quad \color{green}{d} \quad c \quad \color{green}{d} \quad a \quad \color{blue}{b_2} \quad \color{green}{d} \quad c \quad \color{red}{e}$$

$$a_3 \quad b \quad \color{green}{d} \quad c \quad \color{green}{d} \quad a \quad \color{blue}{b_2} \quad \color{green}{d} \quad c \quad \color{red}{e}$$

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$a_3 \quad b \quad d \quad c \quad d \quad a \quad b_2 \quad d \quad c \quad e$$

$$a_3 \quad b \quad d \quad c \quad d \quad a \quad b_2 \quad d \quad c \quad e$$

Inne podejście

Równość słów

Jak sprawdzić równość słów?

$$a_3 \quad b \quad d \quad c \quad d \quad a \quad b_2 \quad d \quad c \quad e$$

$$a_3 \quad b \quad d \quad c \quad d \quad a \quad b_2 \quad d \quad c \quad e$$

Powtarzamy!

Idea

Dla obu słów

- zastępujemy pary
- zastępujemy maksymalne powtórzenia litery

Każda litera jest zastąpiona: długość spada o połowę w iteracji.

Idea

Dla obu słów

- zastępujemy pary
- zastępujemy maksymalne powtórzenia litery

Każda litera jest zastąpiona: długość spada o połowę w iteracji.

while $U \notin \Sigma$ i $V \notin \Sigma$ **do**
  Litery $\leftarrow$ litery występujace w $U = V$
  for $a \in$ Litery **do**
    zastąp maksymalne bloki a^ℓ przez a_ℓ (świeża litera)
  Pary $\leftarrow$ pary liter z $U = V$
  for $ab \in$ Pary **do**
    zastąp pary ab przez c (świeża litera)

Idea

Dla obu słów

- zastępujemy pary
- zastępujemy maksymalne powtórzenia litery

Każda litera jest zastąpiona: długość spada o połowę w iteracji.

while $U \notin \Sigma$ i $V \notin \Sigma$ **do**
    Litery $\leftarrow$ litery występujace w $U = V$
    for $a \in$ Litery **do**
        zastąp maksymalne bloki a^ℓ przez a_ℓ (świeża litera)
    Pary $\leftarrow$ pary liter z $U = V$
    for $ab \in$ Pary **do**
        zastąp pary ab przez c (świeża litera)

Jak to zrobić dla równań?

Przykład

Przykład roboczy

$XbaYa = ba^3bab^2a^2$ ma rozwiązanie $S(X) \leftarrow ba^3$, $S(Y) \leftarrow b^2a$

Przykład

Przykład roboczy

$XbaYa = ba^3bab^2a^2$ ma rozwiązanie $S(X) \leftarrow ba^3$, $S(Y) \leftarrow b^2a$

Zastępujemy ba przez c.

$$XbaYa = baabaabbaa \qquad \text{dla } S(X) \leftarrow baaa \; S(Y) \leftarrow bba$$
$$XcY = caacabca \qquad \text{dla } S(X) \leftarrow caa \; S(Y) \leftarrow abc$$

Przykład

Przykład roboczy

$XbaYa = ba^3bab^2a^2$ ma rozwiązanie $S(X) \leftarrow ba^3$, $S(Y) \leftarrow b^2a$

Zastępujemy ba przez c.

$$XbaYa = baabaabbaa \qquad \text{dla } S(X) \leftarrow baaa \; S(Y) \leftarrow bba$$
$$XcY = caacabca \qquad \text{dla } S(X) \leftarrow caa \; S(Y) \leftarrow abc$$

A co jeśli zastępujemy ab przez d?

$$XbaYa = baaabaabbaa \qquad \text{dla } S(X) \leftarrow baaa \; S(Y) \leftarrow bba$$

Przykład

Przykład roboczy

$XbaYa = ba^3bab^2a^2$ ma rozwiązanie $S(X) \leftarrow ba^3$, $S(Y) \leftarrow b^2a$

Zastępujemy ba przez c.

$$XbaYa = baabaabbaa \qquad \text{dla } S(X) \leftarrow baaa\ S(Y) \leftarrow bba$$
$$XcY = caacabca \qquad \text{dla } S(X) \leftarrow caa\ S(Y) \leftarrow abc$$

A co jeśli zastępujemy ab przez d?

$$XbaYa = baaabaabbaa \qquad \text{dla } S(X) \leftarrow baaa\ S(Y) \leftarrow bba$$

Jest problem z tymi „parami na brzegu”. Poprawimy to!

Typy par

Definicja (Typy par)

Wystąpienie ab jest

- jawne jeśli pochodzi z U lub z V;
- ukryte jeśli pochodzi z $S(X)$;
- kłopotliwe w przeciwnym przypadku.

Para jest kłopotliwa jeśli ma kłopotliwe wystąpienie, w p.p. jest niekłopotliwa.

$$XbaYb \leftarrow baaababbab \qquad \text{dla } S(X) \leftarrow baaa \; S(Y) \leftarrow bba$$

Kompresja par niekłopotliwych

KompresjaParNieKł

1: $c \in \Sigma$ nowa litera

2: zastąp wszystkie ab w $U = V$ przez c

Kompresja par niekłopotliwych

KompresjaParNieKł

1: $c \in \Sigma$ nowa litera

2: zastąp wszystkie ab w $U = V$ przez c

Lemat

To działa.

Kompresja par niekłopotliwych

KompresjaParNieKł

1: $c \in \Sigma$ nowa litera
2: zastąp wszystkie ab w $U = V$ przez c

Lemat

To działa.

dowód

- jawne przekształcamy
- ukryte same się przekształcą
- kłopotliwych nie ma

Kompresja par kłopotliwych

Czemu ab jest kłopotliwa?

Jest X t. że aX jest w $U = V$ oraz $S(X) = bw$ lub (symetrycznie).

Kompresja par kłopotliwych

Czemu *ab* jest kłopotliwa?

Jest X t. że aX jest w $U = V$ oraz $S(X) = bw$ lub (symetrycznie).

- dla aX: zastępujemy X przez bX
  (zmieniamy $S(X) = bw$ na $S(X) = w$)

Kompresja par kłopotliwych

Czemu *ab* jest kłopotliwa?

Jest X t. że aX jest w $U = V$ oraz $S(X) = bw$ lub (symetrycznie).

- dla aX: zastępujemy X przez bX
  (zmieniamy $S(X) = bw$ na $S(X) = w$)
- dla Xb: zastępujemy X przez Xa
  (zmieniamy $S(X) = wa$ na $S(X) = w$)

Kompresja par kłopotliwych

Czemu *ab* jest kłopotliwa?

Jest X t. że aX jest w $U = V$ oraz $S(X) = bw$ lub (symetrycznie).

- dla aX: zastępujemy X przez bX
  (zmieniamy $S(X) = bw$ na $S(X) = w$)
- dla Xb: zastępujemy X przez Xa
  (zmieniamy $S(X) = wa$ na $S(X) = w$)

Lemat

Dla wszystkich zmiennych: ab przestaje być kłopotliwe.

A niekłopotliwe umiemy kompresować.

KompresjaPar

KompresjaPar(a, b)

for X t. że aX jest w $U = V$ i $S(X) = bw$ **do**
  zastąp X przez bX
for X t. że Xb jest w $U = V$ and $S(X) = wa$ **do**
  zastąp X przez Xa
KompresjaParNieKł(a,b)

KompresjaPar

KompresjaPar(a, b)

for X t. że aX jest w $U = V$ i $S(X) = bw$ **do**
  zastąp X przez bX
for X t. że Xb jest w $U = V$ and $S(X) = wa$ **do**
  zastąp X przez Xa
KompresjaParNieKł(a,b)

Lemat

KompresjaPar(a, b) *kompresuje parę ab.*

KompresjaPar

KompresjaPar(a, b)

for X t. że aX jest w $U = V$ i $S(X) = bw$ **do**
  zastąp X przez bX
for X t. że Xb jest w $U = V$ and $S(X) = wa$ **do**
  zastąp X przez Xa
KompresjaParNieKł(a,b)

Lemat

KompresjaPar(a, b) *kompresuje parę ab.*

Używamy w algorytmie.

KompresjaPar

KompresjaPar(a, b)

for X t. że aX jest w $U = V$ i $S(X) = bw$ **do**
  zastąp X przez bX
for X t. że Xb jest w $U = V$ and $S(X) = wa$ **do**
  zastąp X przez Xa
KompresjaParNieKł(a,b)

Lemat

KompresjaPar(a, b) *kompresuje parę ab.*

Używamy w algorytmie.

Powtórzenia: podobnie.

Analiza

Twierdzenie (Skracanie rozwiązań)

Niech ab będą dwoma kolejnymi literami w $U = V$ *lub w* $S(U) = S(V)$*. Przynajmniej jedna z liter a, b jest skompresowana.*

Analiza

Twierdzenie (Skracanie rozwiązań)

Niech ab będą dwoma kolejnymi literami w $U = V$ lub w $S(U) = S(V)$. Przynajmniej jedna z liter a, b jest skompresowana.

Wniosek

Algorytm ma $\mathcal{O}(\log N)$ faz.

Analiza

Twierdzenie (Skracanie rozwiązań)

Niech ab będą dwoma kolejnymi literami w $U = V$ lub w $S(U) = S(V)$. Przynajmniej jedna z liter a, b jest skompresowana.

Wniosek

Algorytm ma $\mathcal{O}(\log N)$ faz.

Wniosek

Równanie ma długość $\mathcal{O}(n^2)$.

- w jednej fazie wprowadzamy $\mathcal{O}(n^2)$ nowych liter do równania.
- w fazie równanie skraca się o stały czynnik.