Złożoność komunikacyjna

Krzysztof Nozderko

19 października 2004 roku

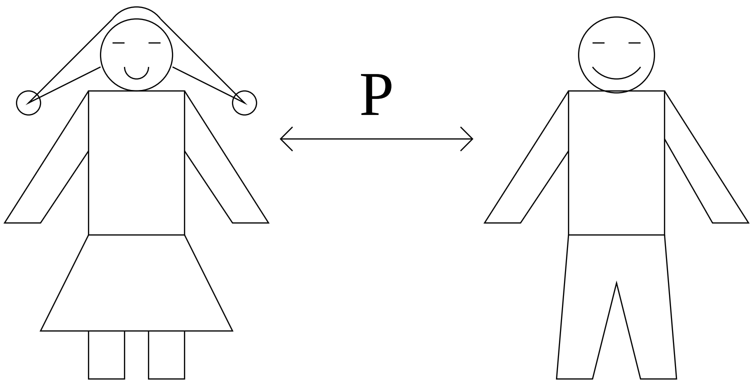
System ma wykonać jakieś zadanie, na podstawie danych rozproszonych po różnych częściach systemu (nazywanych procesorami, częściami lub graczami).

W celu wykonania zadania gracze muszą się ze sobą komunikować.

Przyjmujemy następujące założenia:

- są tylko 2 części systemu
- każda część systemu otrzymuje pewną ustaloną część informacji wejściowej
- jedyny zasób to komunikacja
- zadanie = obliczenie pewnej funkcji na podstawie danych wejściowych

X, Y, Z – skończone zbiory.
f: $X \times Y \to Z$ – dowolna funkcja
Alicja i Bob chcą wspólnie obliczyć f(x,y).
Alicja zna tylko $x \in X$.
Bob zna tylko $y \in \mathrm{Y}$.
Alicja i Bob komunikują się wg pewnego ustalonego protokołu $\mathcal{P}$, który zależy tylko od f.



Alicja ma x

Bob ma y

Protokół $\mathcal{P}$ specyfikuje:

- czyja jest kolej na mówienie – zależy to tylko od dotychczasowej rozmowy
- co mówi uczestnik – to zależy tylko od dotychczasowej rozmowy, oraz danych wejściowych mówcy
- kiedy obliczenie się kończy
- obliczoną odpowiedź f(x,y)

Definicja 1 ***Protokół $\mathcal{P}$ dla funkcji f: $X \times Y \rightarrow Z$ to drzewo binarne, w którym każdy węzeł v jest etykietowany albo przez***

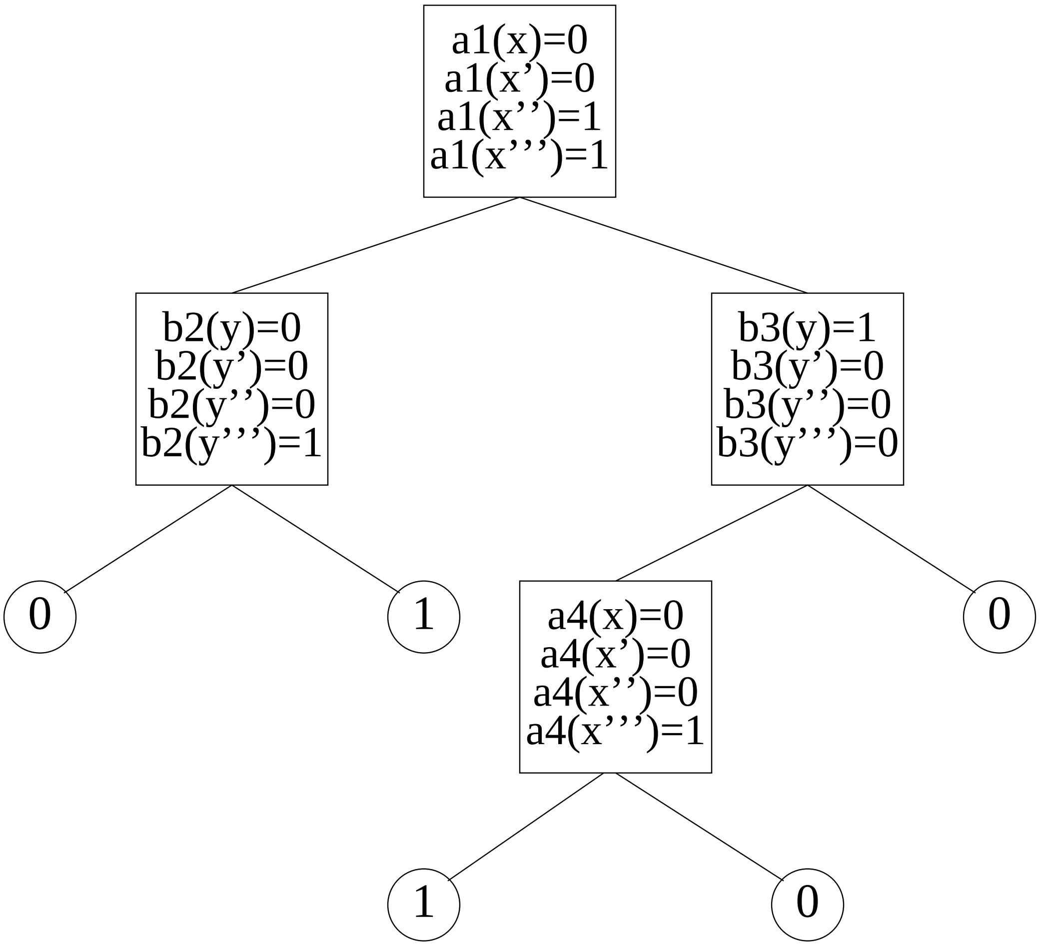
- ***funkcję*** $a_v : X \rightarrow \{0,1\}$
- ***albo przez*** $b_v : Y \rightarrow \{0,1\}$

Wartością protokołu $\mathcal{P}$ dla danych wejściowych (x,y) jest etykieta liścia, do którego się dojdzie startując od korzenia, idąc wg następujących reguł:

- ***z węzła wewnętrznego v etykietowanego przez*** a_v ***idziemy***
  - ***w lewo, jeśli*** $a_v(x) = 0$
  - ***w prawo, jeśli*** $a_v(x) = 1$
- ***z węzła wewnętrznego v etykietowanego przez*** b_v ***idziemy***
  - ***w lewo, jeśli*** $b_v(y) = 0$
  - ***w prawo, jeśli*** $b_v(y) = 1$

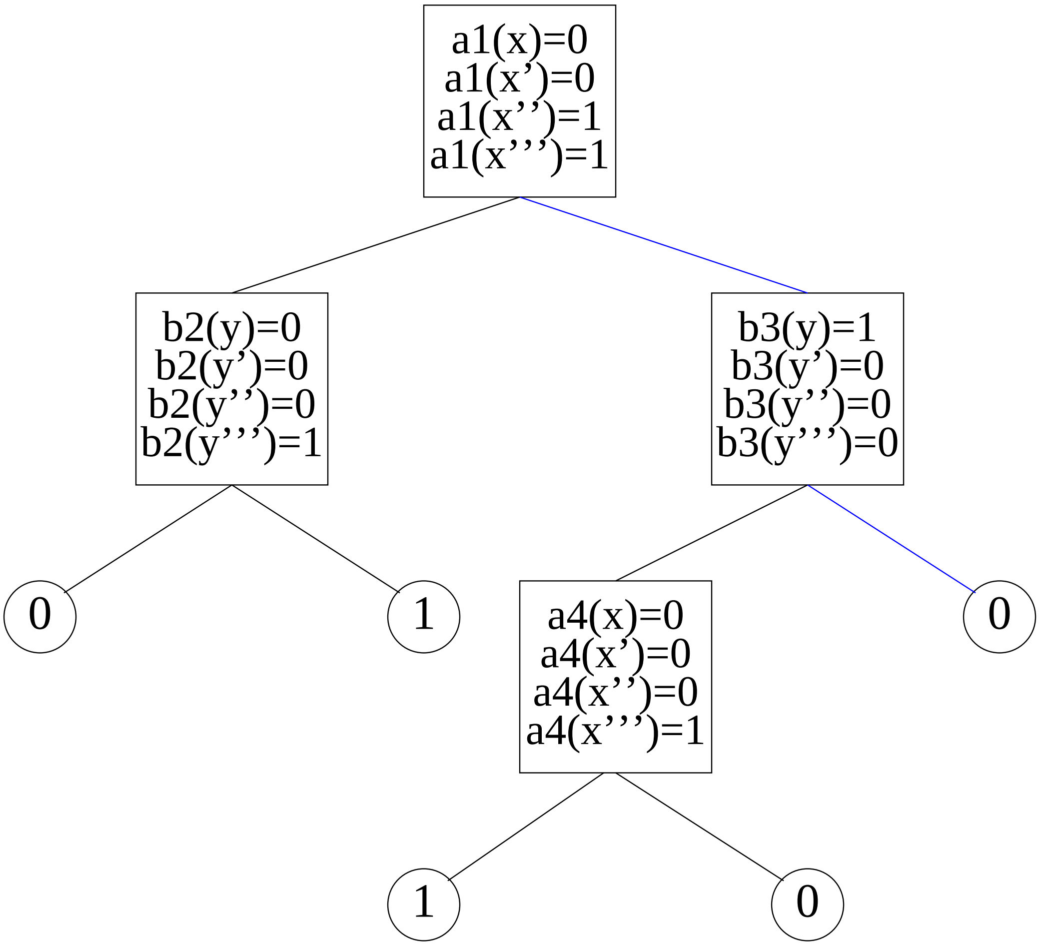
Koszt protokołu $\mathcal{P}$ dla danych wejściowych (x,y) jest długością ścieżki (od korzenia do liścia) jaką dostaniemy dla (x,y). Koszt protokołu $\mathcal{P}$ to wysokość drzewa.

Przykład drzewa protokołu dla $X = \{x, x', x'', x'''\}$, $Y = \{y, y', y'', y'''\}$.



Rysunek 1: Drzewo protokołu

Obliczamy f(x”,y) = 0. (skrajnie prawa ścieżka w drzewie)



Rysunek 2: Obliczamy f(x”,y)

Definicja 2 ***Dla funkcji f: X × Y → Z, (deterministyczna) złożoność komunikacyjna funkcji f to***

$$D(f) = \min_{\mathcal{P}} koszt(\mathcal{P})$$

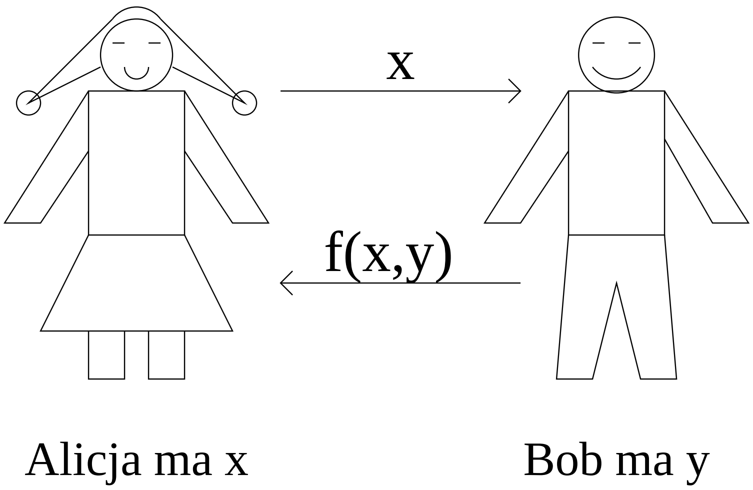
gdzie $\mathcal{P}$ ***oznacza protokół obliczający f.***

Fakt 1 ***Dla każdej funkcji f: X × Y → Z,***

$$D(f) \leqslant \log_2 |X| + \log_2 |Z|$$

Dowód

Złożoność $\log_2 |X| + \log_2 |Z|$ można otrzymać gdy Alicja prześle Bobowi x, Bob obliczy odpowiedź z = f(x,y) i odeśle jej z.



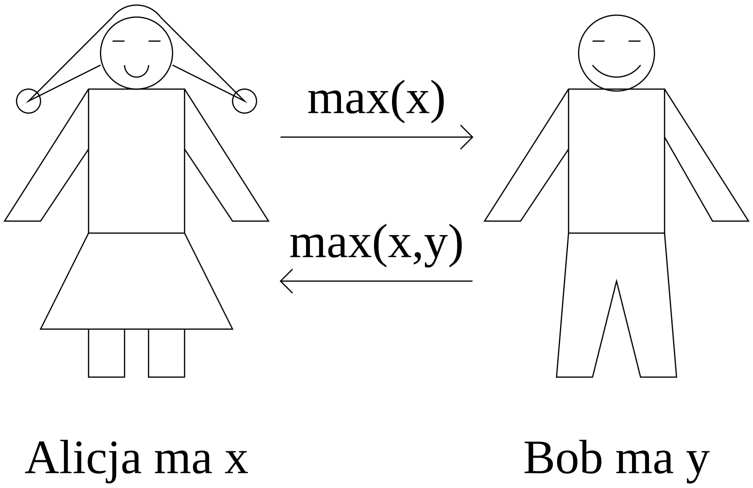
□

Przykład

Alicja ma x $\subseteq$ {1, . . . , n}

Bob ma y $\subseteq$ {1, . . . , n}

Obliczają Max(x,y) = max(x $\cup$ y)



Rysunek 3: Algorytm obliczania Max(x,y)

Więc D(Max) $\leqslant$ 2log n

Przykład

Alicja ma x $\subseteq \{1, \ldots, n\}$
Bob ma y $\subseteq \{1, \ldots, n\}$
Obliczają Med(x,y) = mediana(x $\cup$ y)

Algorytm o złożoności $\leqslant n + \log_2 n$:

1. Alicja przesyła Bobowi x (n bitowy string).

2. Bob oblicza Med(x,y) i przesyła Alicji wynik ($\log_2 n$ bitów).

Więc D(Med) $\leqslant n + \log_2 n$

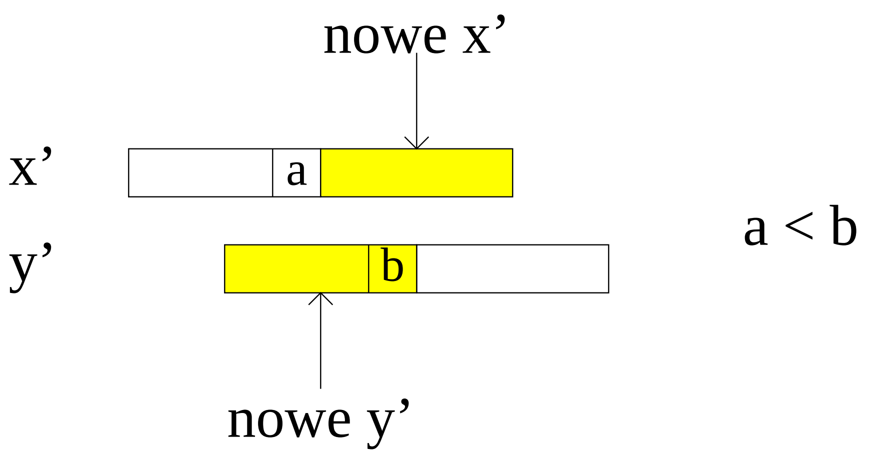
Algorytm o złożoności $O(\log^2 n)$:

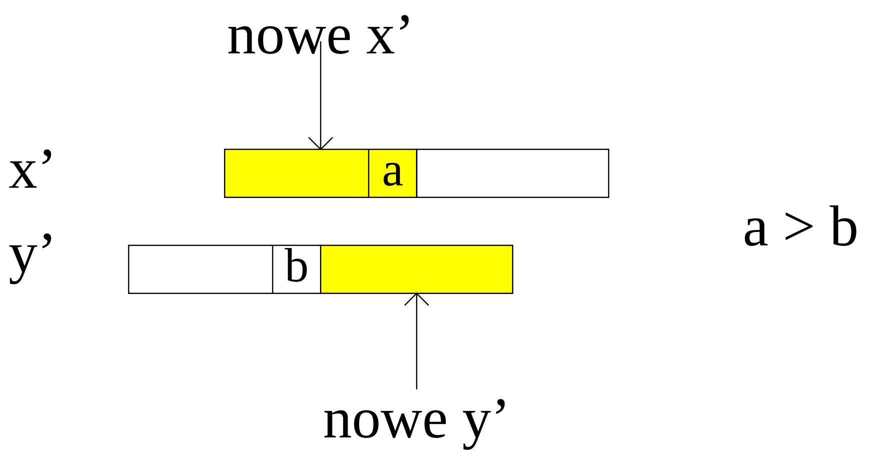
- przeszukujemy binarnie przedział [1,n]
- w każdym kroku:
  - Alicja i Bob mają przedział [i, j], wewnątrz którego leży mediana
  - $k := (i + j) / 2$
  - Alicja przesyła Bobowi ile elementów $> k$ oraz $< k$ jest w x ($O(\log n)$ bitów)
  - Bob decyduje w której połówce jest mediana i komunikuje to Alicji (1 bit)

$\mathcal{P}$ wykonuje $O(\log n)$ kroków.
Krok - $O(\log n)$ bitów.
$D(Med) = O(\log^2 n)$

Inny algorytm $O(\log^2 n)$:

- Alicja i Bob uzupełniają swe zbiory 1-kami i n-ami tak, by $|x| = |y| = 2^k$, oraz by mediana się nie zmieniła
- W każdym kroku pętli:
  - Alicja ma $x' \subseteq x$, w którym może być mediana (początkowo $x' = x$)
  - Bob ma $y' \subseteq y$, w którym może być mediana (początkowo $y' = y$)
  - a = mediana(x'), b = mediana(y')
  - Alicja i Bob wymieniają sią wartościami a i b ($O(\log n)$ bitów)
  - jeśli $a = b$, to jest to mediana
  - jeśli $a < b$, to
    x' := górna połowa(x'),
    y' := dolna połowa(y')
  - jeśli $a > b$, to
    x' := dolna połowa(x'),
    y' := górna połowa(y')
  - $|x'|=|y'|=1 \Rightarrow$ mediana to $\min(x',y')$





W każdym kroku rozmiar x' i y' jest zmniejszany 2-krotnie $\Rightarrow$ O(log n) kroków

1 krok = O(log n) bitów

D(Med) = $O(\log^2 n)$

Algorytm działający w czasie O(log n).
Poprzedni + heurystyka:

- a i b są przesyłane bit po bicie począwszy od najbardziej znaczącego bitu
- jeśli $a_i = b_i$, oraz $a_j \neq b_j$ dla $i < j$ to:
  - na podstawie j-tego bitu określamy które z a i b jest większe
  - w przyszłości przesyłane mediany będą się zgadzać na bitach wcześniejszych niż j-ty, więc ich już nie będziemy przesyłać
- ale jeśli mediana którą chcemy przesyłać, nie leży wewnątrz przedziału wyznaczonego przez mediany w poprzednim kroku, to osoba której mediana jest poza zakresem udaje, że jej medianą jest poprzednia mediana jego przeciwnika żeby się o tym poinformować potrzeba O(1) komunikacji

Kombinatoryczny punkt spojrzenia na protokoły:
dzielimy przestrzeń $X \times Y$ na zbiory, tak by zbiór reprezentował komunikację jaka będzie miała miejsca dla dowolnego elementu ze zbioru
cel: dowodzenie dolnych ograniczeń złożoności protokołów

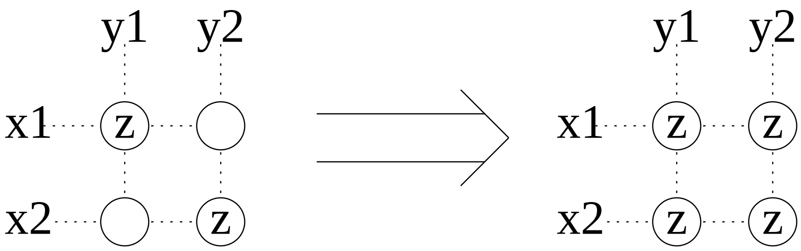
Definicja 3 *Niech $\mathcal{P}$ będzie protokołem, a v wierzchołkiem w drzewie protokołu.* R_v *jest zbiorem danych wejściowych (x,y), dla których idąc po drzewie dojdziemy do v.*

Fakt 2 *Jeśli L to zbiór liści protokółu $\mathcal{P}$, to $\{R_l\}_{l \in L}$ jest podziałem $X \times Y$.*

Definicja 4 *Kombinatoryczny prostokąt (lub krótko prostokąt) w $X \times Y$,*
to $R \subseteq X \times Y$ taki,
że $R = A \times B$ dla jakiegoś $A \subseteq X$ i jakiegoś $B \subseteq Y$.

Fakt 3 ***$R \subseteq X \times Y$ jest prostokątem wtw***

$$(x_1, y_1) \in R \wedge (x_2, y_2) \in R \Rightarrow (x_1, y_2) \in R.$$



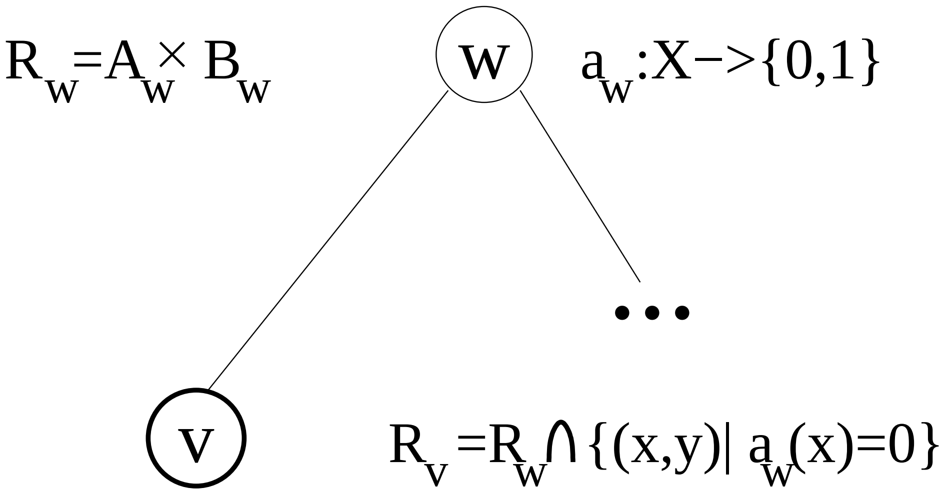
Rysunek 4: Własność prostokąta

Fakt 4 ***Dla każdego protokółu $\mathcal{P}$ i węzła v, R_v jest prostokątem.***

Dowód

Indukcja po głębokości węzła v:

1. **$\mathrm{R}_{korzeń} = \mathbf{X} \times \mathbf{Y}$, więc $\mathrm{R}_{korzeń}$ jest prostokątem.**
2. **krok indukcyjny:**



$$\mathbf{R}_v = \mathbf{A}_w \times \mathbf{B}_w \cap \{(\mathbf{x},\mathbf{y})|\ \mathbf{a}_w(\mathbf{x}) = 0\}$$
$$\mathbf{R}_v = (\mathbf{A}_w \cap \{\mathbf{x}|\ \mathbf{a}_w(\mathbf{x}) = 0\}) \times \mathbf{B}_w$$

Definicja 5 *Podzbiór $R \subseteq X \times Y$ jest zwany f-monochromatycznym (lub krótko monochromatycznym) jeżeli wartość funkcji f dla elementów tego zbioru jest taka sama.*

Przykład monochromatycznego prostokąta $X = \{0{,}1\}^3$, $Y = \{0{,}1\}^3$, $R = A \times B$.

B

A

	000	001	010	011	100	101	110	111
000	0	1	1	0	1	0	0	0
001	1	0	0	0	0	0	0	1
010	1	0	0	0	1	0	0	0
011	0	0	1	0	0	0	0	1
100	1	0	0	0	1	0	0	1
101	1	1	1	0	0	0	1	1
110	0	0	0	0	1	0	0	0
111	0	1	1	0	1	1	0	1

Rysunek 5: 0-monochromatyczny prostokąt

Lemat 5 ***Każdy protokół $\mathcal{P}$ dla funkcji f indukuje podział zbioru $X \times Y$ na f-monochromatyczne prostokąty. Prostokątów jest tyle ile liści w $\mathcal{P}$.***

Przykład

	y	y'	y''	y'''
x	0	0	0	1
x'	0	0	0	1
x''	0	0	0	0
x'''	0	1	1	1

Rysunek 6: Przykładowy podział f

Wniosek 6 ***Jeżeli każdy podział zbioru $X \times Y$ na f-monochromatyczne prostokąty jest utworzony przez co najmniej t prostokątów, to D(f) $\geqslant$ log$_2$ t***

Dowód

t prostokątów = t liści,
drzewo mające t liści ma wysokość $\geqslant \log_2 t$
$\square$

Techniki szacowania złożoności z dołu:

- kombinatoryczne
  - zbiór oszukujący (ang. fooling set)
    zbiór t.że żadne jego 2 elementy nie mogą być w tym samym monochromatycznym prostokącie
    duży zbiór oszukujący $\Rightarrow$ wiele prostokątów $\Rightarrow$ wiele liści $\Rightarrow$ wysokie drzewo $\Rightarrow$ duża złożoność
  - wielkość prostokąta
    małe prostokąty $\Rightarrow$ jest ich wiele $\Rightarrow$ wiele liści $\Rightarrow$ wysokie drzewo $\Rightarrow$ duża złożoność
- algebraiczne

Definicja 6 ***Niech f: X × Y → {0, 1}. Zbiór S ⊂ X × Y nazywany jest oszukującym (ang. fooling set) jeśli ∃ z ∈ {0, 1}, t. że***

- $\forall\ (x,y) \in S,\ f(x,y) = z$
- $\forall\ (x_1,y_1)\ \forall\ (x_2,y_2),\ (x_1,y_1) \neq (x_2,y_2) \Rightarrow f(x_1,y_2) \neq z$ ***lub*** $f(x_2,y_1) \neq z$

Czyli zbiór oszukujący jest:

- monochromatyczny
- bardzo nieprostokątny

Lemat 7 ***Jeżeli f ma zbiór oszukujący S rozmiaru t, to D(f)*** $\geqslant \log_2 t$.

Dowód

- Żaden monochromatyczny prostokąt nie może zawierać $>$ 1 element z S
(przypuśćmy przeciwnie:
R zawiera (x_1,y_1) oraz (x_2,y_2)
więc R zawiera też (x_1,y_2) oraz (x_2,y_1)
ale f(x_1,y_2) $\neq$ z lub f(x_2,y_1) $\neq$ z
więc R nie jest monochromatyczny)
- więc musi być przynajmniej t prostokątów, aby pokryć S
- teza wynika z wcześniejszego lematu

Przykład

Alicja ma x $\in \{0, 1\}^n$

Bob ma y $\in \{0, 1\}^n$

Alicja i Bob chcą obliczać funkcję

EQ(x,y) = [x = y]

Rozwiązanie

S = $\{(\alpha,\alpha) \mid \alpha \in \{0, 1\}^n\}$

S jest zbiorem oszukującym, bo

- $\forall\alpha$ **EQ(α,α) = 1**
- $\forall\alpha \neq \beta$ **EQ(α,β) = 0**

|S| = 2^n, więc D(EQ) $\geqslant$ n

licząc też 0-prostokąty: D(EQ) $\geqslant$ n + 1

a ponieważ D(EQ) $\leqslant$ n + 1

to D(EQ) = n + 1

Przykład

Alicja ma x $\in$ {1, . . . , n}
Bob ma y $\in$ {1, . . . , n}

$$DISJ(x, y) = \begin{cases} 1 & jeśli\ x \cap y = \phi \\ 0 & wpp \end{cases}$$

Obliczają funkcję rozłączności DISJ(x,y)

Rozwiązanie

S = {(A,$\overline{A}$) | A $\subseteq$ {1, . . . , n}}
S to zbiór oszukujący, bo:

- **$\forall$ A, DISJ(A, $\overline{A}$) = 1**
- **$\forall$ A $\neq$ B, DISJ(A, $\overline{B}$) = 0 lub DISJ($\overline{A}$, B) = 0**

  (jeśli a $\in$ A i a $\notin$ B to a $\in$ $\overline{B}$,
  więc DISJ(A, $\overline{B}$) = 0
  a jeśli b $\notin$ A i b $\in$ B to b $\in$ $\overline{A}$,
  więc DISJ($\overline{A}$, B) = 0)

$|S| = 2^n$, więc $D(DISJ) \geqslant n$
licząc także 0-prostokąty $D(DISJ) \geqslant n+1$
a ponieważ $D(DISJ) \leqslant n + 1$
to $D(DISJ) = n + 1$

Fakt 8 ***Niech*** μ ***będzie rozkładem prawdopodobieństwa w*** $\boldsymbol{X} \times \boldsymbol{Y}$***.***
Jeśli każdy f-monochromatyczny prostokąt $\boldsymbol{R}$ ***ma miarę*** $\mu(\boldsymbol{R}) \leqslant \delta$***, to*** $\boldsymbol{D(f)} \geqslant \log_2 1/\delta$

Technika zbioru oszukującego = szczególny przypadek techniki mierzenia

S = fooling set, |S| = t.

$$\mu(x,y) = \begin{cases} 1/t & \textbf{dla } (x,y) \in S \\ 0 & \textbf{wpp} \end{cases}$$

dowolny monochromatyczny prostokąt R, może zawierać co najwyżej 1 element z S, więc μ(R) $\leqslant$ 1/t, co pociąga D(f) $\geqslant \log_2 t$

Jeśli

- μ ma rozkład jednostajny,

- każdy prostokąt byłby mniejszy od k,

to zachodzi
D(f) $\geqslant$ log|X| + log |Y| - log k

Przykład

Alicja ma x $\in \{0, 1\}^n$

Bob ma y $\in \{0, 1\}^n$

$$IP(x, y) = \sum_{i=1}^{n} x_i y_i \ (mod\ 2)$$

IP = iloczyn wektorowy modulo 2

Rozwiązanie

- **0-monochromatyczny prostokąt zawiera $\leqslant 2^n$ elementów**
  (R = A $\times$ B - 0-monochromatyczny
  A' = span(A), B' = span(B)
  R'= A' $\times$ B' - 0-monochromatyczny
  A' $\perp$ B' więc dim(A') + dim(B') $\leqslant$ n
  wielkość R $\leqslant$ |A'| |B'| $\leqslant 2^n$)
- **elementów (x,y) t. że IP(x,y) = 0, jest $2^{2n}/2 = 2^{2n-1}$**
- **μ - rozkład jednostajny na zerach**

Z powyższych wynika, że

μ(R) $\leqslant 2^n \ / \ 2^{2n-1} = 2^{-(n-1)}$

więc D(IP) $\geqslant$ n - 1

Algebraiczny sposób - Rank Lower Bound

z każdą funkcją f: $X \times Y \to \{0, 1\}$ będziemy związywać macierz M_f t., że $M_f(x,y) = f(x,y)$

Do szacowania złożoności wykorzystamy naszą wiedzę o macierzach.

Definicja 7 *rank(f) to liniowy rząd macierzy* $\boldsymbol{M_f}$

Lemat 9 ***Dla każdej funkcji f: X $\times$ Y $\rightarrow$ {0, 1},***

$$D(f) \geqslant \log_2 rank(f)$$

Dowód

L_1 - zbiór liści etykietowanych 1

dla każdego $l \in L_1$, określam macierz M_l:

$$M_l(x,y) = \begin{cases} 1 & dla\ (x,y) \in R_l \\ 0 & wpp \end{cases}$$

f(x,y)=0 $\Rightarrow$ M_l(x,y) = 0

f(x,y)=1 $\Rightarrow$ M_l(x,y) = 1

$$M_f = \sum_{l \in L_1} M_l$$

$$rank(M_f) \leqslant \Sigma_{l \in L_1} rank(M_l)\ \ (w\text{ł}a\text{ś}ciwo\text{ść}\ rz\text{ę}du)$$

rank(M_l) = 1, więc rank(M_f) $\leqslant |L_1| \leqslant |L|$

Więc $\mathcal{P}$ musi mieć $\geqslant$ rank(f) liści

Lemat 10 ***Dla dowolnej funkcji***
f: $\boldsymbol{X \times Y \rightarrow \{0, 1\}}$***,***
$\boldsymbol{D(f) \geqslant log_2(2rank(f) - 1)}$

Dowód
Podobnie jak poprzednio, ale tym razem zamieniam 0 i 1 rolami (tzn. rozpatruje funkcję not(f))
$M_{not(f)} = J - M_f$,
J = macierz samych 1, rank(1) = 1, więc zliczając 0-prostokąty i 1-prostokąty
$D(f) \geqslant \log_2(2rank(f) - 1)$

Przykład

M_{EQ} odpowiadająca funkcji EQ to macierz identycznościowa.

$\mathrm{rank}(\mathrm{M}_{EQ}) = 2^n$, więc $\mathrm{D(EQ)} \geqslant n$

Przykład

M_{IP} to macierz odpowiadająca funkcji IP.

$\mathrm{N} = (\mathrm{M}_{IP})^2$

$\mathrm{N}(y,y') = \Sigma_{z\in\{0,1\}^n} <y,z><z,y'>$

$\mathrm{N}(y,y')$ = ilość elementów z,

dla których $<y,z> = <z,y'> = 1$

$\mathrm{rank}(\mathrm{N}) = 2^n - 1$

$\mathrm{rank}(\mathrm{M}_{IP}) = \mathrm{rank}(\mathrm{N})$, bo

$\mathrm{rank}(AB) \leqslant \min(\mathrm{rank}(A), \mathrm{rank}(B))$

więc $\mathrm{D(IP)} \geqslant n$

Randomizacja

- Alicja i Bob mogą rzucać monetą
  Alicja ma dostęp do losowego ciągu r_A
  Bob ma dostęp do losowego ciągu r_B
  r_A, r_B - niezależne, wybrane wg pewnego rozkładu prawdopodobieństwa
- w drzewie $\mathcal{P}$ węzły Alicji etykietowane są przez funkcje $a_v(x, r_A)$, a Boba przez $b_v(y, r_B)$
- komunikacja dla (x,y) przestaje być ustalona, staje się zmienną losową

Podejścia

- konserwatywne (Las Vegas) - protokoły zawsze poprawne
- liberalne (Monte Carlo) - protokoły poprawne z dużym prawdopodobieństwem

Definicja 8 ***Niech $\mathcal{P}$ będzie randomizowanym protokołem.***
Poniższe prawdopodobieństwa są oparte o r_A i r_B.

- ***$\mathcal{P}$ oblicza funkcję f bezbłędnie, jeśli $\forall$ (x,y), Pr[$\mathcal{P}$(x,y) = f(x,y)] = 1***
- ***$\mathcal{P}$ oblicza funkcję f z ϵ-błędem jeśli $\forall$ (x,y), Pr[$\mathcal{P}$(x,y) = f(x,y)] $\geqslant$ 1 - ϵ***
- ***$\mathcal{P}$ oblicza funkcję f z jednostronnym ϵ-błędem,***
  jeśli $\forall$ (x,y) t. że f(x,y) = 0,
  Pr[$\mathcal{P}$(x,y) = 0] = 1,
  i $\forall$ (x,y) t. że f(x,y) = 1,
  Pr[$\mathcal{P}$(x,y) = 1] $\geqslant$ 1 - ϵ,

Dla tych samych danych

- możemy otrzymywać różne wyniki
- komunikacja może mieć różną długość (czas działania średni lub pesymistyczny)

Definicja 9 *Niech $\mathcal{P}$ - randomizowany protokół* ***Pesymistycznym przypadkiem czasu działania*** *randomizowanego protokołu $\mathcal{P}$ dla danych wejściowych (x,y) jest maksymalna liczba bitów komunikacji dla dlowolnych wyborów ciągów losowych r_A i r_B.*
Pesymistycznym kosztem *$\mathcal{P}$ jest maksimum po wszystkich (x,y) pesymistycznego czasu działania $\mathcal{P}$ dla (x,y).*

Średnim przypadkiem czasu działania *randomizowanego protokołu $\mathcal{P}$ dla danych wejściowych (x,y) jest oczekiwana ilość bitów komunikacji (lub równoważnie głębokości liści) po wszystkich wyborach losowych ciągów r_A i r_B.* ***Średnim kosztem*** *$\mathcal{P}$ jest maksimum po wszystkich (x,y) średniego czasu działania $\mathcal{P}$ dla (x,y).*

Definicja 10 ***Niech f: X $\times$ Y $\to$ {0, 1} Definujemy następujące miary złożoności f:***

- $\boldsymbol{R_0(f)} = \min_{\mathcal{P}}$ ***średni koszt randomizowanego*** $\mathcal{P}$ ***obliczającego f***
- ***Dla 0*** $< \epsilon <$ ***1/2,*** $\boldsymbol{R_\epsilon(f)} = \min_{\mathcal{P}}$ ***pesymistyczny koszt randomizowanego*** $\mathcal{P}$ ***obliczającego f z*** ϵ***-błędem. Oznaczamy np.*** $\boldsymbol{R(f)} = \boldsymbol{R_{1/3}(f)}$
- ***Dla 0*** $< \epsilon <$ ***1,*** $\boldsymbol{R^1_\epsilon(f)} = \min_{\mathcal{P}}$ ***pesymistyczny koszt randomizowanego*** $\mathcal{P}$ ***obliczającego f z jednostronnym*** ϵ***-błędem. Oznaczamy np.*** $\boldsymbol{R^1(f)} = \boldsymbol{R^1_{1/2}(f)}$

Dla protokołów:

- bezbłednych - koszt pesymistyczny odpowiada protokołom deterministycznym; jedynym ciekawym kosztem jest więc średni
- popełniających błedy - koszt pesymistyczny wygodniejszy;
  rózni się od średniego tylko stałą w mnożeniu
  - jeśli mamy protokół o złożoności pesymistycznej, koszt średni będzie niegorszy
  - niech $\mathcal{P}$ robi $\epsilon/2$-błąd w średnim czasie t, zatem
    Pr[P robi $> 2t/\epsilon$ kroków] $\leqslant \epsilon$ / 2
    $\mathcal{P}$' = symuluje $\mathcal{P}$ przez $\leqslant 2t/\epsilon$ kroków
    jeśli $\mathcal{P}$' zwraca $\begin{cases} \textit{to co } \mathcal{P} & \textit{jeśli } \mathcal{P} \textit{ się zakończył} \\ 0 & \textit{wpp} \end{cases}$
    $\mathcal{P}$' robi błąd ϵ w pesymistycznym czasie $2t/\epsilon$

Przykład

Alicja ma $a = a_0 a_1 \ldots a_{n-1}$
Bob ma $b = b_0 b_1 \ldots b_{n-1}$
Alicja i Bob chcą obliczać funkcję
EQ(a,b) = [a = b]

1. Alicja i Bob wybierają liczbę pierwszą p t. że $n^2 < p < 2n^2$
2. Alicja konstruuje wielomian $A(x) = a_0 + a_1 x + \ldots + a_{n-1} x^{n-1} (mod\, p)$, a Bob wielomian $B(x) = b_0 + b_1 x + \ldots + b_{n-1} x^{n-1} (mod\, p)$
3. Alicja losuje $t \in \{0, 1, \ldots, p - 1\}$, oraz wysyła Bobowi t oraz A(t)
4. Bob zwraca czy A(t) = B(t)

Analiza poprawności:

- **jeśli a = b, protokół działa poprawnie**
- **jeśli a $\neq$ b, to A(x) = B(x) dla $\leqslant$ n - 1 argumentów, bo A(x) - B(x) ma $\leqslant$ n - 1 miejsc zerowych, więc pr. błędu $\leqslant \frac{n-1}{p} \leqslant \frac{n}{n^2} = \frac{1}{n}$**

Przesyłanych jest O(log p) = O(log n) bitów, więc:

- **$R_{\frac{1}{n}}$(EQ) = O(log n)**
- **$R^1_{1/n}$(NE) = O(log n)**