Spis treści

5 Uliniowienie wielu sekwencji

5.6 Klasa MAX SNP

Zaczniemy od twierdzenia Fagina charkteryzującego klasę NP. Ustalamy pewną sygnaturę pierwszego rzędu (tylko symbole relacji). *Formuła drugiego rzędu* φ wygląda podobnie do formuły pierwszego rzędu, z wyjątkiem tego, że φ może zawierać symbole relacji nie należące do sygnatury (symbole te nazwiemy *zmiennymi drugiego rzędu*). Formuła φ może też zawierać kwantyfikatory wiążące zmienne drugiego rzędu. Zależność φ od zmiennych wolnych (pierwszego i drugiego rzędu) występujących w φ możemy zapisać następująco: $\varphi(\vec{x}, \vec{S})$, gdzie $\vec{x}$ jest ciągiem zmiennych pierwszego rzędu, a $\vec{S}$ jest ciągiem zmiennych drugiego rzędu.

Twierdzenie 5.6.1 (O postaci normalnej) *Każda formuła φ drugiego rzędu jest równoważna we wszystkich strukturach relacyjnych formule postaci*

$$Q_1S_1 \dots Q_nS_nQ_{n+1}x_1 \dots Q_{n+m}x_m\varphi'(x_1, \dots, x_m, S_1, \dots, S_n),$$

gdzie $Q_i \in \{\forall, \exists\}$, dla $i = 1, \dots, n+m$, $x_1, \dots, x_m$ są zmiennymi pierwszego rzędu, $S_1, \dots, S_n$ są zmiennymi drugiego rzędu, a formuła φ' nie zawiera kwantyfikatorów.

Każdą formułę postaci

$$\exists S_1 \dots \exists S_n Q_1x_1 \dots Q_mx_m\varphi, \tag{5.1}$$

gdzie φ nie zawiera kwantyfikatorów będziemy nazywać formułą Σ_1^1.

Twierdzenie 5.6.2 (Fagin, 1974) *Ustalmy nietrywialną sygnaturę relacyjną (np. zawierającą symbol realcji 2-argumentowej). Zbiór kodów skończonych struktur nad w/w sygnaturą należy do klasy NP $\iff$ istnieje Σ_1^1 formuła, której modelami skończonymi są dokładnie w/w struktury.*

Przykład 5.6.1 (SAT)

$$\exists T \forall c \exists x [(P(c,x) \wedge T(x)) \vee (N(c,x) \wedge \neg T(x))].$$

Intuicja: $P(c,x)$ oznacza, że klauzula c zawiera pozytywne wystąpienie zmiennej x. Natomiast $N(c,x)$ oznacza, że klauzula c zawiera negatywne wystąpienie zmiennej x. □

Uwaga 1: W formule (5.1) można przyjąć, że $n = 1$, czyli że mamy do czynienia z postacią

$$\exists S Q_1 x_1 \ldots Q_m x_m \varphi.$$

Wystarczy wziąć S o arności równej sumie arności zmiennych $S_1, \ldots, S_n$ i zamiast np. $S_1(z_1, \ldots, z_k)$ piszemy $S(z_1, \ldots, z_k, *, \ldots, *)$, gdzie $*$ jest dowolną ustaloną stałą.

Uwaga 2: W formule (5.1) można przyjąć postać

$$\exists S \forall \vec{x} \exists \vec{y} \varphi(\vec{x}, \vec{y}, S), \tag{5.2}$$

czyli, że mamy do czynienia tylko z jedną alternacją kwantyfikatorów pierwszego rzędu. Dowód tego faktu wynika z obserwacji, że jeśli mamy alternację kwantyfikatorów $\forall \vec{x} \exists \vec{y} \psi$, to zamiast niej możemy równoważnie napisać

$$\exists R [\forall \vec{x} \exists \vec{y} R(\vec{x}, z) \wedge \forall \vec{x} \forall \vec{y} (R(\vec{x}, \vec{y}) \longrightarrow \psi)],$$

w ten sposób dostajemy o jedną alternację mniej.

Zatem można przyjąć, że każdy problem w NP można przedstawić formułą postaci (5.2). Następująca definicja pochodzi od Papadimitriou i Yannakakisa (1991). Problem jest *silnie* NP (strict NP), gdy da się go przedstawić formułą postaci

$$\exists S \forall \vec{x} \varphi(\vec{x}, S).$$

Klasę problemów silnie NP będziemy oznaczać przez SNP.

Przykład 5.6.2 (3SAT)
Sygnatura zawiera cztery predykaty 3-argumentowe: C_0, C_1, C_2, C_3. Interpretacja $C_i(x_1, x_2, x_3)$ jest taka, że istnieje klauzula zawierająca zmienne $x_1, \ldots, x_i$ negatywnie oraz zmienne $x_{i+1}, \ldots, x_3$ pozytywnie. Wówczas formuła

$$\begin{aligned} \exists T \forall x_1 \forall x_2 \forall x_3 [(C_0(x_1, x_2, x_3) \longrightarrow (T(x_1) \vee T(x_2) \vee T(x_3))) \wedge \ldots \\ \wedge (C_3(x_1, x_2, x_3) \longrightarrow (\neg T(x_1) \vee \neg T(x_2) \vee \neg T(x_3)))] \end{aligned}$$

wyraża 3SAT. □

Niech Π będzie problemem NP reprezentowanym formułą (5.2). Definiujemy *problem optymalizacyjny* $MAX\Pi$ związany z Π, polegający na znalezieniu (dla każdej skończonej struktury) relacji S tak, aby zmaksymalizować liczebność zbioru $\{\vec{x} \mid \exists\vec{y}\varphi(\vec{x},\vec{y},S)\}$. W ten sposób otrzymujemy klasę

$$MAXNP = \{MAX\Pi \mid \Pi \in NP\}.$$

W podobny sposób definiujemy klasę

$$MAXSNP = \{MAX\Pi \mid \Pi \in SNP\}.$$

Przykład 5.6.3 Poniżej podajemy przykłady problemów z MAX SNP.
(a) MAX 3SAT.
(b) MAX 2SAT.
(c) MAX CUT: dany graf, podzielić zbiór wierzchołków na S oraz $-S$ tak, aby zmaksymalizować liczbę krawędzi pomiędzy S i $-S$. Formuła opisująca ten problem to

$$\exists S\forall x\forall y(E(x,y) \longrightarrow (S(x) \land \neg S(y)) \lor (\neg S(x) \land S(y)),$$

gdzie $E(x,y)$ oznacza, że w grafie istnieje krawędź łącząca wierzchołki x oraz y. $\square$

Jak wyglądają podstawowe składniki dowolnego problemu optymalizacyjnego Π? Z każdą *instancją* I problemu Π związany jest zbiór *rozwiązań* $SOL(I)$ oraz *funkcja kosztu* $c : SOL(I) \longrightarrow \mathbb{R}^{\geq 0}$, która każdemu rozwiązaniu $\xi \in SOL(I)$ przyporządkowuje koszt $c(\xi)$. Ponadto mamy *optimum* dla I, wartość $OPT(I) \in \mathbb{R}^{\geq 0}$. Algorytm A rozwiązujący problem optymalizacyjny Π znajduje dla każdej instancji $I \in \Pi$ rozwiązanie $A(I) \in SOL(I)$. Powiemy, że *błąd* algorytmu A jest ε, gdy dla każdej instancji $I \in \Pi$

$$\left|\frac{c(A(I))}{OPT(I)} - 1\right| \leq \varepsilon.$$

Podamy teraz definicję pojęcia *L-redukcji.* Dane dwa problemy optymalizacyjne Π, Π'. Powiemy, że Π L-redukuje się do Π' ($\Pi \leq_L \Pi'$), gdy istnieją stałe $\alpha, \beta > 0$ oraz funkcje obliczalne w czasie wielomianowym $f : \Pi \longrightarrow \Pi'$ oraz dla $I \in \Pi$, $g_I : SOL(f(I)) \longrightarrow SOL(I)$, takie że dla dowolnej instancji I problemy Π mamy

- $OPT(f(I)) \leq \alpha \cdot OPT(I)$.
- Dla $\xi \in SOL(f(I))$, mamy
$$|c(g_I(\xi)) - OPT(I)| \leq \beta \cdot |c'(\xi) - OPT(f(I))|.$$

Lemat 5.6.3
(i) *Złożenie L-redukcji jest L-redukcją.*
(ii) *Jeśli* $\Pi \leq_L \Pi'$ *oraz* Π' *ma algorytm działający w czasie wielomianowym o błędzie* ε, *to dla* Π *istnieje algorytm wielomianowy o błędzie* $\alpha \cdot \beta \cdot \varepsilon$.

Twierdzenie 5.6.4 (Papadimitriou, Yannakakis'91)
(i) *Dla każdego problemu* $\Pi \in MAXSNP$ *istnieje* ε *oraz algorytm działający w czasie wielomianowym, który aproksymuje* Π *z błędem* ε.
(ii) *Problemy MAX 3SAT, MAX 2SAT, MAX CUT są zupełne w klasie MAX SNP ze względu na L-redukcje.*

Powiemy, że problem optymalizacyjny Π ma *wielomianowy schemat aproksymacji (PTAS)*, gdy istnieje algorytm $A(-,-)$, taki że dla każdego ε, $A(-,\varepsilon)$ aproksymuje Π w czasie wielomianowym, z błędem ε. *Uwaga:* czas działania $A(-,\varepsilon)$ może wykładniczo zależeć od ε. Mamy następujący natychmiastowy wniosek z Lematu 5.6.3 (ii).

Wniosek 5.6.5 *Jeśli jakiś problem zupełny w klasie MAX SNP ma wielomianowy schet aproksymacji, to każdy problem w MAX SNP ma wielomianowy schemat aproksymacji.*

Następujące ważne twierdzenie mówi, że zapewne bardzo trudno będzie znaleźć wielomianowe schematy aproksymacji dla problemów zupełnych w klasie MAX SNP.

Twierdzenie 5.6.6 (Arora, Lund, Motwani, Sudan, Szegedy'92)
Jeśli MAX SNP-trudny problem ma wielomianowy schemat aproksymacji, to P=NP.

Głównym narzędziem w dowodzie Twierdzenia 5.6.6 są tzw. *probabilistycznie sprawdzalne dowody (PCP)*. Poniżej wyjaśnimy to pojęcie i przytoczymy podstawowy wynik dotyczący PCP. Definicja pochodzi od Arory i Safry (1992). Niech $y \in \{0,1\}^*$ będzie dowolnym słowem. Przez $M^y(-,-)$ będziemy oznaczać probabilistyczną maszynę Turinga z wyrocznią y, działającą w czasie wielomianowym. Niech $f, g : \mathbb{N} \longrightarrow \mathbb{N}$ będą dowolnymi funkcjami. Powiemy, że język L ma probabilistycznie sprawdzalny dowód z parametrami $f(n), g(n)$ (oznaczamy $L \in PCP(f(n), g(n))$), gdy istnieje probabistyczna maszyna Turinga $M^y(-,-)$ z wyrocznią y, działająca w czasie wielomianowym i taka, że:

- Na wejściu dostaje słowo x oraz losowe słowo r o długości $|r| = O(f(|x|))$.
- Generuje zbiór zapytań $Q(r,x) = \{q_1, \ldots, q_m\}$, gdzie $m = O(g(|x|))$.

- Czyta bity $y_{q_1}, \ldots, y_{q_m}$ z wyroczni y (y jest nazywane *dowodem*).
- Wykonuje obliczenie deterministyczne w czasie wielomianowym, używając jako danych: r, x oraz $y_{q_1}, \ldots, y_{q_m}$. Wynik jest binarny: $M^y(r,x) \in \{0,1\}$.
- *Warunki akceptowalności:* istnieje $\delta > 0$ taka, że dla każdego słowa x,
  - Jeśli $x \in L$, to istnieje wyrocznia y taka, że dla każdego losowego ciągu r, $M^y(r,x) = 1$.
  - Jeśli $x \notin L$, to dla każdej wyroczni y, prawdopodobieństwo wylosowania takiego r, że $M^y(r,x) = 0$ jest co najmniej δ.

Zauważmy, że oczywiście mamy $NP = PCP(0, poly(n))$, czyli problemy z NP są rozwiązywane bez losowania ciągu r, ale a dostępem do wielomianowej długości dowodu. Mamy następujące ciekawe twierdzenie. Jest to tzw. twierdzenie PCP.

Twierdzenie 5.6.7 (Arora, Lund, Motwani, Sudan, Szegedy'92)

$$NP = PCP(\log n, 1).$$

Powyższe twierdzenie powstawało w dwóch etapach. W pierwszej wersji pokazano (Arora, Safra'92), że $NP = PCP(\log n, \log n)$.

Twierdzenie 5.6.8 (Wang, Jiang'94)
Następujący problem jest MAX SNP trudny: dana funkcja s (podobieństwo na literach) oraz drzewo filogenetyczne w postaci gwiazdy, znaleźć optymalne filogenetyczne uliniowienie.

Dowód Twierdzenia 5.6.8 jest poprzez technicznie skomplikowaną L-redukcję z problemu MAX CUT-B (problem MAX CUT dla grafów o rzędzie wszystkich wierzchołków ograniczonym przez stałą B. Główną słabością dowodu (i samego twierdzenia) jest to, że w redukcji dobiera się funkcję s (w zależności od instancji MAX CUT-B) oraz, że s nie spełnia warunku trójkąta. Ciekawym problemem jest pytanie czy można dowieść MAX SNP trudności filogenetycznego uliniowienia dla ustalonej funkcji s spełniającej warunek trójkąta (np. dla odległości edycyjnej).