Dominik Wawszczak numer indeksu: 440014

numer grupy: 1

Zadanie 1

Rozpocznijmy od udowodnienia, że dany w zadaniu problem jest w klasie NP. Istotnie, weryfikację czy konkretna funkcja f spełnia wszystkie krotki z \mathcal{A} można zrealizować w czasie wielomianowym, zwyczajnie sprawdzając dla każdej krotki, czy spełniona jest co najmniej jedna z odpowiednich nierówności.

Pozostaje pokazać, że problem ten jest NP-trudny. W tym celu wskażemy wielomianową redukcję problemu 3-CNF-SAT do problemu z treści zadania. Weźmy więc dowolną instancję problemu 3-CNF-SAT. Niech m będzie liczbą zmiennych w tej instancji. Przyjmijmy n=2m+1. Zmienną x_i będziemy utożsamiać z liczbą 2i-1, a zaprzeczenie zmiennej $\neg x_i$ z liczbą 2i. Liczbę 2m+1 będziemy traktować jako sztuczny indykator tego czy zmienna jest prawdziwa, czy fałszywa. Konkretnie, jeśli f(2i-1) > f(2m+1), to zmienna x_i jest prawdziwa, a jeśli f(2i-1) < f(2m+1), to zmienna ta jest fałszywa.

Rozpocznijmy z pustym zbiorem krotek \mathcal{A} , do którego będziemy dodawać kolejne krotki. Na początek dla każdej pary $(i,j) \in \{1,\ldots,n\}^2$ takiej, że i < j, wrzućmy krotkę (i,j,j,i), gwarantując tym samym, że jeśli istnieje odpowiednia funkcja f, to jest ona iniekcją.

Kolejnym krokiem będzie zagwarantowanie, że dla każdego $i \in \{1, ..., m\}$ zachodzi dokładnie jeden z dwóch przypadków: f(2i) < f(2m+1) < f(2i-1), czyli $x_i = \top$, albo f(2i-1) < f(2m+1) < f(2i), co oznacza, że $x_i = \bot$. Niech $a,b,c \in \{1,...,n\}$ będą parami różne. Zauważmy, że wrzucając do \mathcal{A} krotkę (c,b,b,a) sprawiamy, że nie może zachodzić f(a) < f(b) < f(c), przy czym na każdym z pozostałych 5 możliwych posortowań f(a), f(b) i f(c) krotka ta jest spełniona. Możemy w ten sposób "zabronić" wszystkich kolejności f(2i-1), f(2i) oraz f(2m+1), z wyjątkiem dwóch wymienionych wyżej.

Bez straty ogólności możemy założyć, że żadna klauzula nie zawiera jednocześnie zmiennej i jej zaprzeczenia, gdyż takie klauzule są zawsze spełnione. Klauzulę z instancji problemu 3-CNF-SAT będziemy rozpatrywać jako zbiór liczb odpowiadających zawartym w niej literałom (powtórzenia możemy oczywiście zignorować), zgodnie z numeracją opisaną w drugim akapicie. Przykładowo, klauzulę $(x_4 \vee \neg x_6 \vee x_9)$ będziemy rozumieć jako zbiór $\{7,12,17\}$, a klauzulę $(x_5 \vee \neg x_2 \vee \neg x_2)$ jako $\{4,9\}$. Dla każdego zbioru jednoelementowego $\{a\}$ dodamy do $\mathcal A$ krotkę (2m+1,a,2m+1,a). Dla każdego zbioru dwuelementowego $\{a,b\}$ dodamy do $\mathcal A$ krotkę (2m+1,a,2m+1,b). Wreszcie dla każdego zbioru trójelementowego $\{a,b,c\}$ dodamy do $\mathcal A$ krotkę $(2m+1,a,b\oplus 1,c)$, zakładając bez straty ogólności a < b < c, gdzie \oplus jest operatorem bitowym xor.

Pozostaje wykazać równoważność:

istnieje funkcja $f: \{1, ..., n\} \to \mathbb{Q}$ spełniająca wszystkie krotki z $\mathcal{A} \iff$ formuła z instancji problemu 3-CNF-SAT jest spełnialna.

Zaczniemy od wykazania implikacji "w lewo". Weźmy więc wartościowanie spełniające formułę, po czym dla każdego $i=1,\ldots,m$ przypiszmy f(2i-1)=i=f(2i)+m+1, jeżeli $x_i=\top$, albo f(2i-1)=i-m-1=f(2i)-m-1, w przeciwnym wypadku. f(2m+1) zdefiniujemy jako 0. Łatwo zauważyć, że funkcja f spełnia wszystkie krotki z akapitów 3 i 4. Dla klauzul, którym odpowiadają zbiory jedno lub dwuelementowe również nietrudno zauważyć, że funkcja f spełnia odpowiadające im krotki. Pozostają klauzule, którym odpowiadają zbiory trójelementowe

 $\{a,b,c\}$. Przypuśćmy nie wprost, że $0>f(a) \land f(b\oplus 1)>f(c)$. Skoro b< c i klauzula ta nie zawiera jednocześnie zmiennej i jej zaprzeczenia, to $\left\lfloor \frac{b+1}{2} \right\rfloor < \left\lfloor \frac{c+1}{2} \right\rfloor$. Ponieważ f(a)<0, to odpowiadający jej literał jest fałszem, zatem f(b)>0 lub f(c)>0. Jeżeli f(b)>0, mamy:

$$f(b \oplus 1) = \left| \frac{b+1}{2} \right| - m - 1 < \left| \frac{c+1}{2} \right| - m - 1 \leqslant f(c),$$

czyli sprzeczność. W drugim przypadku natomiast zachodzi:

$$f(c) = \left| \frac{c+1}{2} \right| > \left| \frac{b+1}{2} \right| \geqslant f(b \oplus 1),$$

co również daje sprzeczność.

Na koniec pozostaje wykazać implikację "w prawo". Niech więc f będzie funkcją spełniającą każdą klauzulę z \mathcal{A} . Bez straty ogólności możemy założyć, że f(2m+1)=0, po prostu odejmując stałą f(2m+1) od każdej wartości funkcji, co nie wpłynie na żadną nierówność. Zgodnie z tym, co opisano w drugim akapicie, przyjmiemy $x_i = \top$, jeśli f(2i-1) > 0, albo $x_i = \bot$, gdy f(2i-1) < 0. Od razu widać, że formuły zawierające jeden lub dwa literały są spełnione przy takim wartościowaniu. Żeby formuła trójelementowa, której odpowiada zbiór $\{a,b,c\}$ była niespełniona, musiałoby zachodzić f(a), f(b), f(c) < 0. Skoro f(b) < 0, to $f(b \oplus 1) > 0$, czyli nie zachodzi żadna z nierówności f(2m+1) < f(a) i $f(b \oplus 1) < f(c)$, co oznacza sprzeczność, gdyż istnieje krotka w \mathcal{A} , której funkcja f nie spełnia.

Wykazana równoważność kończy rozwiązanie zadania.

Zadanie 2

Rozpocznijmy od napisania programu liniowego do problemu z treści zadania:

$$\forall u \in V(G) \quad 0 \leqslant x_u \leqslant 1,$$

$$\forall x_u + x_v \geqslant 1,$$

$$\forall x_u + x_v \geqslant 1,$$

$$\forall x_v \in V(G) \quad \sum_{v \in N[u]} x_v \leqslant y,$$

$$\min : u$$

Weźmy rozwiązanie tego programu w liczbach rzeczywistych, które oczywiście możemy uzyskać w czasie wielomianowym. Wówczas zachodzi $y \leq \mathrm{OPT}(G)$, ponieważ biorąc rozwiązanie programu w liczbach całkowitych otrzymamy optymalne rozwiązanie problemu. Niech

$$X = \left\{ u : x_u \geqslant \frac{1}{2} \right\}.$$

Wtedy X jest pokryciem wierzchołkowym, ponieważ gdyby dla pewnej krawędzi $uv \in E(G)$ zachodziło $x_u < \frac{1}{2}$ i $x_v < \frac{1}{2}$, to mielibyśmy $x_u + x_v < 1$.

Zauważmy, że dla każdego $u \in V(G)$ spełnione jest

$$|X \cap N[u]| \leqslant 2 \cdot \sum_{v \in N[u]} x_v \leqslant 2 \cdot y \leqslant 2 \cdot \text{OPT}(G).$$
 (*)

Wynika stąd, że wskazany algorytm jest algorytmem 2-aproksymacyjnym tego problemu, co kończy rozwiązanie pierwszej części zadania.

Żeby gęstość pokrycia wierzchołkowego znaleziona przez nasz algorytm wyniosła więcej niż $2 \cdot \mathrm{OPT}(G) - 1$, musi istnieć wierzchołek $u \in V(G)$, dla którego w każdej nierówności ciągu nierówności * zachodzi równość. W szczególności oznacza to, że dla pewnego $u \in V(G)$ w nierówności

$$|X \cap N[u]| \leqslant 2 \cdot \sum_{v \in N[u]} x_v$$

zachodzi równość.

Nierówność ta bierze się z dodania stronami nierówności postaci $1 \leq 2x_v$, dla $v \in X \cap N[u]$, oraz $0 \leq 2x_v$, dla $v \in N[u] \setminus X$. We wszystkich tych nierównościach musi więc zachodzić równość, zatem

$$\bigvee_{v \in N[u]} x_v \in \left\{0, \frac{1}{2}\right\}.$$

Jeżeli dla pewnego $v \in N[u] \setminus \{u\}$ byłoby $x_v = 0$, to mielibyśmy $x_u + x_v \leqslant \frac{1}{2} < 1$, czyli sprzeczność, toteż $x_v = \frac{1}{2}$, dla każdego $v \in N[u] \setminus \{u\}$.

Jeśli $x_u=0$ oraz u nie jest wierzchołkiem izolowanym, to biorąc dowolnego sąsiada v otrzymujemy $x_u+x_v=\frac{1}{2}<1$, czyli również sprzeczność, z czego wynika, że $x_u=\frac{1}{2}$. O wierzchołkach izolowanych natomiast możemy założyć, że nie ma ich w naszym pokryciu, ponieważ i tak są one zbedne.

Z powyższego wynika, że jeżeli gęstość pokrycia wierzchołkowego jest większa niż $2 \cdot \mathrm{OPT}(G) - 1$, to istnieje wierzchołek w tym pokryciu, którego wszyscy sąsiedzi również w nim są. Możemy więc go zwyczajnie usunąć z tego pokrycia i powtarzać tę procedurę tak długo, aż taki wierzchołek istnieje. Na koniec zostaniemy z pokryciem o gęstości co najwyżej $2 \cdot \mathrm{OPT}(G) - 1$, co kończy rozwiązanie zadanie.