12 Problem niesprzeczności układów o ustalonej liczbie równań

12.1 Algorytmy pseudowielomianowe i problemy silnie NP-zupełne

Będziemy rozważać jeszcze jedno pojęcie rozmiaru danych. Niech A będzie macierzą, wektorem lub jakimś ukladem liczb całkowitych o elementach $a_1, \ldots, a_k$. Definiujemy pseudorozmiar danych $\langle A \rangle^*$ przyjmując, że

$$\langle A \rangle^* = \langle A \rangle + \max\{| a_j | : j = 1, \ldots, k\}.$$

Zauważmy, że zdefiniowana wartość spełnia nierówność

$$\langle A \rangle \leq \langle A \rangle^* \leq 2 \cdot 2^{\langle A \rangle}$$

oraz, że $\langle A \rangle^*$ jest obliczalne w czasie wielomianowym (w czasie $w(\langle A \rangle)$ dla pewnego wielomianu w).

Algorytm $\mathcal{A}$ jest pseudowielomianowy, jeżeli istnieje taki wielomian w, że dla dowolnych danych A algorytm $\mathcal{A}$ po uruchomieniu z daną A wykonuje najwyżej $w(\langle A \rangle^*)$ operacji. Pokażemy dalej, że istnieje pseudowielomianowy algorytm rozwiązujący problem niesprzeczności układu m liniowych równań diofantycznych.

Problem L jest silnie NP-zupełny, jeżeli istnieje wielomian w taki, że problem

$$\{A \in L : \langle A \rangle^* \leq w(\langle A \rangle)\}$$

jest NP-zupełny.

Pojęcie silnej NP-zupełności ma własności podobne do własności NP-zupełności. Podstawową własność problemów silnie NP-zupełnych wyraża następujący lemat:

Lemat 12.1 *Jeżeli dla pewnego problemu silnie NP-zupełnego istnieje algorytm pseudowielomianowy, to P=NP.*

Dowód. Przypuśćmy, że problem L jest silnie NP-zupełny, a w jest wielomianem takim, że problem $L' = \{A \in L : \langle A \rangle^* \leq w(\langle A \rangle)\}$ jest NP-zupełny. Niech $\mathcal{A}$ będzie pseudowielomianowym algorytmem rozwiązującym problem L'. Zdefiniujemy wielomianowy algorytm rozwiązujący problem L'.

Problem L_1 jest silnie redukowalny do problemu L_2, jeżeli dla problemu L_2 jest zdefiniowane pojęcie pseudorozmiaru danych oraz jeżeli istnieje taka funkcja F przekształcająca dane dla problemu L_1 w dane dla problemu L_2, obliczana w czasie wielomianowym i taki wielomian w, że

$$X \in L_1 \Leftrightarrow F(X) \in L_2$$

oraz

$$\langle F(X) \rangle^* \leq w(\langle F(X) \rangle)$$

dla dowolnych danych X dla problemu L_1.

Lemat 12.2 *Jeżeli L_2 jest problemem z klasy* **NP**, *a problem L_1 jest* **NP**-*zupełny oraz silnie redukowalne do L_2, to problem L_2 jest silnie* **NP**-*zupełny.*

Dowód. Niech w będzie wielomianem, a F funkcją o własnościach podanych w definicji silnej redukowalności i wynikających z tego, że L_1 jest silnie redukowalny do L_2. Będziemy rozważać problem

$$L_3 = \{A \in L_2 : \langle A \rangle^* \leq w(\langle A \rangle)\}.$$

Zauważmy, że funkcja F także redukuje problem L_1 do L_3. Ponieważ sprawdzenie dodatkowe warunku z definicji L_3 (poza należeniem do L_2) wymaga obliczenia pseudorozmiaru danych i wartości pewnego wielomianu i wobec tego może być wykonane w czasie wielomianowym, więc problem L_3 należy do klasy **NP** i w konsekwencji jest **NP**-zupełny. To oznacza, że problem L_2 jest silnie **NP**-zupełny. □

12.2 Silna NP-trudność problemu 11.4

Przypuśćmy, że rozważamy układ równań liniowych $A\vec{x} = \vec{b}$, którego wszystkie współczynniki i wyrazy wolne przyjmują wartości 0 lub 1. Zauważmy, że rozmiar $\langle A, \vec{b} \rangle$ takiego układu i jego pseudorozmiar $\langle A, \vec{b} \rangle^*$ różnią się najwyżej o 1 i zachodzi nierówność

$$\langle A, \vec{b} \rangle^* \leq \langle A, \vec{b} \rangle + 1.$$

Twierdzenie 12.3 *Problem istnienia selektora redukuje się wielomianowo do problemu niesprzeczności układów $A\vec{x} = \vec{b}$ diofantycznych równań liniowych spełniających warunek $\langle A, \vec{b} \rangle^* \leq \langle A, \vec{b} \rangle + 1$. Tak więc problem 11.4 jest silnie* **NP***-trudny.*

Dowód. Aby dowieść to twierdzenie wystarczy zauważyć, że w dowodzie twierdzenia 11.14 konstruowaliśmy układy równań o współczynnikach i wyrazach wolnych równych 0 lub 1, i powtórzyć dowód tego twierdzenia. □

12.3 Wnioski z twierdzenia Papadimitriou

Jedno z oszacowań rozwiązań układu diofantycznych równań liniowych znalazł Papadimitriou dowodząc

Twierdzenie 12.4 (Papadimitriou) *Niech A będzie macierzą o wymiarach $n \times m$ o wyrazach całkowitych i niech $\vec{b} \in R^m$ będzie wektorem o współrzędnych całkowitych. Załóżmy, że elementy A i b mają wartości bezwzględne $\leq a$. Jeżeli układ m równań $A\vec{x} = \vec{b}$ z n niewiadomymi ma rozwiązanie złożone z liczb naturalnych, to ma także rozwiązanie złożone z liczb naturalnych nie przekraczających $(n+1)(ma)^{2m+1}$.*

Twierdzenie Papadimitriou z dowodem jest zamieszczone w tych materiałach także jako twierdzenie 15.11.

Wniosek 12.5 *Problem niesprzeczności układu liniowych równań diofantycznych (problem 11.4) należy do klasy NP.*

Dowód. O tym, że problem niesprzeczności układu liniowych równań diofantycznych należy do klasy NP świadczy następujący algorytm:

Algorytm 12.1 niedeterministyczny algorytm rozwiązujący problem 11.4
Dane: układ równań $A\vec{x} = \vec{b}$.

1) ustalamy liczbę m równań i liczbę n zmiennych danego układu,
2) obliczamy liczbę a, która jest największą liczbą spośród wartości bezwzględnych elementów macierzy A i wyrazów wektora $\vec{b}$,
3) obliczamy liczbę $B = (n+1)(ma)^{2m+1}$,
4) niedeterministycznie zgadujemy wektor $\vec{x} = (x_1, \ldots, x_n)$ o współrzędnych $x_i \leq B$ dla $i = 1, \ldots, n$,
5) obliczamy wektor $A\vec{x}$ i porównujemy go z wektorem $\vec{b}$; jeżeli są równe, to przekazujemy, że dany układ ma rozwiązanie, a w przeciwnym razie – że jest sprzeczny.

Nietrudno zauważyć, że liczbę operacji wykonywanych przez podany algorytm szacuje pewien wielomian zależący od n i m. Największymi liczbami, które obliczamy podczas wykonywania tego algorytmu są współrzędne wektora $A\vec{x}$. Co do wartości bezwzględnej są to liczby $\leq n \cdot a \cdot B$. Rozmiar tych liczb jest $\leq 2 \cdot n \cdot \langle a \rangle \cdot (2m+1)(\langle a \rangle + m)$, a więc daje się oszacować za pomocą wielomianu zależącego od n, m i rozmiaru $\langle a \rangle$ liczby a. Ponieważ liczby n, m i $\langle a \rangle$ są mniejsze od rozmiaru danych, więc podany algorytm jest wielomianowy. □

Wniosek 12.6 *Problem niesprzeczności układu m liniowych równań diofantycznych (problem 11.3) należy do klasy NP.* $\square$

Wniosek 12.7 *Istnieje pseudowielomianowy algorytm rozwiązujący problem 11.3 niesprzeczności układu m liniowych równań diofantycznych.*

Dowód. Rozważmy następujący algorytm posługujący się dwoma listami S i S_p wektorów o m współrzędnych.

Algorytm 12.2 rozwiązujący problem 11.3.
Dane: układ $A\vec{x} = \vec{b}$ złożony z m równań liniowych.

1) ustalamy liczbę n zmiennych danego układu;

2) znajdujemy liczbę a, największą co do wartości bezwzględnej spośród elementów macierzy A i wyrazów wektora $\vec{b}$;

3) obliczamy liczbę $B = (n+1)(ma)^{2m+1}$;

4) zmiennej S przypisujemy listę złożoną z wektora zerowego o m współrzędnych, zmiennej (pomocniczej) S_p – listę pustą, znajdujemy kolumny $\vec{v}_1, \ldots, \vec{v}_n \in R^m$ macierzy A;

5) dla wszystkich i od 1 do n
{lista S zawiera wszystkie kombinacje wektorów $\vec{v}_1, \ldots, \vec{v}_{i-1}$ o naturalnych współczynnikach $\leq B$ i jest listą bez powtórzeń}

    (a) dla wszystkich j od 0 do B
    i dla każdego elementu s listy S

        obliczamy $s_1 = s + j \cdot \vec{v}_i$;

        sprawdzamy, czy s_1 występuje na liście S_p; jeżeli nie występuje, to dopisujemy s_1 do listy S_p;

    (b) listę S_p zapamiętuje w zmiennej S, a następnie powodujemy, że S_p staje się pusta;

6) sprawdzamy, czy $\vec{b}$ znajduje się na liście S; Jeżeli tak, to stwierdzamy, że dany układ ma rozwiązanie, a w przeciwnym razie – że jest sprzeczny.

Poprawność przedstawionego algorytmu jest konsekwencją tego, że informacja podana w komentarzu do punktu 5) jest niezmiennikiem pętli z tego punktu. Jeżeli jest to niezmiennik, to po zakończeniu wykonywania pętli z punktu 5) wartością zmiennej S jest lista zawierająca wszystkie wektory postaci $\sum_{i=1}^{n} x_i\vec{v}_i$, gdzie $x_i \leq B$ dla $i = 1, \ldots, n$. Jeżeli $(\alpha_1, \ldots, \alpha_n)$ jest rozwiązaniem danego równania, to oczywiście $\sum_{i=1}^{n} \alpha_i\vec{v}_i = \vec{b}$.

Przypuśćmy, że dane równanie ma rozwiązanie w liczbach naturalnych. Z twierdzenia Papadimitriou 15.11 wynika, że ma także rozwiązanie naturalne $(\alpha_1, \ldots, \alpha_n)$ takie, że $\alpha_i \leq B$ dla $i = 1, \ldots, n$. Dla tego rozwiązania wektor $\sum_{i=1}^{n} \alpha_i\vec{v}_i$ znajdzie się na liście S. Ponieważ ten wektor jest równy $\vec{b}$, więc w ostatniej fazie algorytm 12.2 znajdzie na liście S wektor $\vec{b}$ i stwierdzi, że dane równanie ma rozwiązanie. Jeżeli natomiast dane równanie nie ma naturalnego rozwiązania, to dla dowolnych liczb naturalnych x_i suma $\sum_{i=1}^{n} x_i\vec{v}_i$ nie jest równa $\vec{b}$ i wektor $\vec{b}$ nie występuje na liście S. Wobec tego, algorytm 12.2 nie znajdzie tego wektora na liście S i stwierdzi, że dane równanie nie ma rozwiązania diofantycznego.

Aby ocenić złożoność algorytmu 12.2 musimy oszacować długość list ze zmiennych S i S_p. Listy te zawsze zawierają wektory postaci $\sum_{i=1}^{k} x_i\vec{v}_i$ dla ustalonego $k \leq n$ i naturalnych współczynników $x_i \leq B$. Współrzędne takich wektorów są liczbami całkowitymi, mają bezwzględne wartości nie przekraczające $k \cdot a \cdot B \leq n \cdot a \cdot B$ i w konsekwencji mogą przyjmować $2 \cdot n \cdot a \cdot B + 1$ różnych wartości. Wektorów o m takich współrzędnych jest $(n \cdot a \cdot B + 1)^m$. W pętli z punktu 5) algorytm najwyżej

$$n \cdot B \cdot (2 \cdot n \cdot a \cdot B + 1)^m \cdot (2 \cdot n \cdot a \cdot B + 1)^m$$

razy wykonuje pewną stałą (zależną od m) liczbę czynności. Pętla z punktu 6) jest wykonywana najwyżej $(n \cdot a \cdot B + 1)^m$ razy. Tak więc złożoność algorytmu 12.2 nie przekracza

$$c \cdot n \cdot (n+1) \cdot (ma)^{2m+1} \cdot (2 \cdot n \cdot a \cdot (n+1) \cdot (ma)^{2m+1} + 1)^{2 \cdot m}$$

i wyraża się wielomianem zależącym od n i a, którego stopień i współczynniki wyrażają się w pewien sposób przez stałą m.

Nietrudno zauważyć, że liczby pojawiające się podczas realizacji algorytmu 12.2 mają wartości bezwzględne $\leq n \cdot a \cdot B$, a ich rozmiar można oszacować za pomocą wielomianu zależącego od n i $\langle a \rangle$ (a także od stałej m). Tak więc algorytm 12.2 korzysta z wielomianowej pamięci.

Ostatecznie, algorytm 12.2 rozwiązuje problem 11.3 i jest pseudowielomianowy. $\square$