

Zadania domowe. Zestaw 3.3

Maciej Poleski

15 stycznia 2013

1

Sklonujmy zbiór V jako V' . Konstruujemy zbiór E' jako: $(u, v) \in E \Rightarrow (u, v') \in E' \wedge (u', v) \in E', u' \in E', v' \in E'$. Gdzie wagi tych krawędzi są równe wadze krawędzi z oryginalnego grafu. Otrzymujemy graf $G' = (V \cup V', E')$. Taki graf ma własność polegającą na tym, że każda ścieżka parzystej długości (krawędziowo) rozpoczynająca się w V kończy się w V . (Przejście z dowolnego wierzchołka dwoma krawędziami w dowolną stronę powoduje że wracamy do tego zbioru wierzchołków z którego wychodziliśmy). Każda ścieżka parzystej długości z grafu wyjściowego jest odwzorowana w nowym grafie (po prostu co drugi wierzchołek ma dodatkowy '). Jednocześnie nie tworzymy żadnych dodatkowych krawędzi. Oznacza to że najkrótsza ścieżka z dowolnego wierzchołka $s \in E$ parzystej długości do dowolnego innego wierzchołka jest najkrótszą ścieżką w nowym grafie kończącą się w dokładnie tym samym wierzchołku (teraz mamy już gwarancję że jest parzystej długości - nieparzystej kończą się w V'). Pozostaje więc uruchomić standardowy algorytm Dijkstry (wersja na kopcu) na nowym grafie. Jako bonus uzyskamy również najkrótsze ścieżki nieparzystej długości (w V'). Nowy graf ma dwa razy więcej wierzchołków i krawędzi więc złożoność pozostaje taka sama.

Warto zauważyć że podczas implementacji nie musimy modyfikować grafu wejściowego - możemy przechowywać na kopcu informacje powiązane z każdym wierzchołkiem o tym czy dotarliśmy do danego wierzchołka ścieżką parzystej długości czy nieparzystej. (Innymi słowy możemy wyobrazić sobie że mamy graf taki jak w tym rozwiązaniu a w rzeczywistości cały czas operować na grafie wejściowym.)

2

Jeżeli istnieje rozwiązanie to istnieje rozwiązanie w którym każda niewiadoma jest niedodatnia. $x_i + a - x_j - a = x_i - x_j$ więc możemy znaleźć największe x_i i dodać do wszystkich niewiadomych $-x_i$. Układ nierówności możemy przedstawić w postaci grafu skierowanego. Dla każdej nierówności $x_i - x_j \leq w$ tworzymy krawędź skierowaną (x_j, x_i) o wadze w . Gwarantuje to że odległość x_i od x_j jest nie większa od w . Jeżeli w takim grafie jest sens mówić o odległościach to możemy obliczyć najkrótsze ścieżki za pomocą algorytmu F-B. Procedurę szukania rozwiązania możemy sobie wyobrazić jak wybieranie dowolnej krawędzi i relaxowanie do oporu (jeżeli istnieje krawędź którą można relaxować - zrób to i powtórz procedurę). Taka procedura nie zakończy się jeżeli istnieje cykl o ujemnej wadze (dlatego całość realizujemy za pomocą F-B). W odniesieniu do układu nierówności relaxowanie krawędzi jest ope-

racją poprawienia wartości niewiadomej tak aby nierówność była spełniona. Gdy nie ma krawędzi do relaxowania - znaczy że wszystkie nierówności są już spełnione. Gdy istnieje cykl o ujemnej wadze zachodzi sytuacja gdy układ jest sprzeczny - nie da się nadać niewiadomym takich wartości aby wszystkie nierówności naraz zostały spełnione. Pozostaje pytanie od czego zacząć relaxowanie (czyli od której nierówności rozpocząć). Pokazałem wyżej że jeżeli istnieje rozwiązanie to istnieje rozwiązanie w którym wartości niewiadomych są niedodatnie. Oznacza to że wstępnie możemy nadać wszystkim niewiadomym wartość 0 i rozpocząć relaxowanie. Jak już wcześniej zaznaczyłem - aby uzyskać odpowiednią złożoność obliczeniową i możliwość wykrycia sytuacji sprzecznej używamy algorytmu Bellmana-Forda.

Czyli rozwiązaniem problemu jest skonstruowanie grafu o n wierzchołkach i m krawędziach zadanych nierównościami i uruchomienie na tym grafie algorytmu Bellmana-Forda. Wstępnie odległość każdego wierzchołka wynosi 0 (dla implementacji algorytmu F-B nie jest istotne który wierzchołek jest startowy - ważne aby jakikolwiek wierzchołek miał zadaną odległość tak aby relaxowanie mogło się odbyć). Odpowiedzią jest fakt istnienia cyklu o ujemnej wadze lub wyznaczone w opisany powyżej sposób odległości.