Rozwiązanie zadania F 877.
Atom w stanie podstawowym pochłonie foton (wiązki laserowej) o energii odpowiadającej stanowi wzbudzonemu, a następnie, średnio po czasie τ, wyemituje go w losowym kierunku. W każdym akcie absorpcji foton przekazuje atomowi swój pęd $p = h/\lambda$, w wyniku czego atom zmniejsza swą prędkość (pęd fotonu jest przeciwnie skierowany do pędu atomu) o $\Delta u = h/(M\lambda)$. Proces powtórzy się po wyemitowaniu pochłoniętego fotonu, tj. po czasie τ. Ze względu na losowy kierunek emisji średni pęd emitowanego fotonu wynosi zero. Atomy poruszają się więc ze średnim opóźnieniem $a = \Delta u/\tau$, a droga przebyta do zatrzymania wynosi

$$s = u_0^2/(2a) = u_0^2 \tau M\lambda/(2h) \approx 11\ \text{cm}.$$

Opisaną tu metodę „chłodzenia" atomów zaproponowali T. W. Hänsch (nagroda Nobla 2005) i A. L. Schawlow (Nagroda Nobla 1981) w pracy w *Optics Communications* **15**, 68 (1975) skąd zaczerpnięte zostały także dane liczbowe zadania.

Informatyczny kącik olimpijski (80): Podział grafu

Jednym z najtrudniejszych zadań, z którym mierzyli się uczestnicy światowych finałów konkursu ACM ICPC w roku 2014, było zadanie pt. *Metal Processing Plant*. W zadaniu mamy dany n-wierzchołkowy ($n \leqslant 200$) nieskierowany graf pełny. Dla każdej krawędzi uv grafu znamy jej wagę $d(u,v)$. Wagą podzbioru wierzchołków S nazwiemy maksymalną wagę krawędzi po wszystkich parach wierzchołków z tego podzbioru, tzn. $d(S) = \max_{u,v\in S} d(u,v)$. Należy znaleźć podział wierzchołków grafu na dwa podzbiory A, B, których sumaryczna waga $d(A) + d(B)$ jest jak najmniejsza.

Załóżmy na początek, że zdecydowaliśmy, iż podzielimy graf na dwa podzbiory o wagach nie większych niż ustalone liczby d_A i d_B, i chcemy stwierdzić, czy da się to zrobić. Innymi słowy, chcemy znaleźć dwa podzbiory A i B, że $d(A) \leqslant d_A$ i $d(B) \leqslant d_B$. Dla każdego wierzchołka u wprowadźmy zmienną logiczną x_u, która oznaczać będzie, czy wierzchołek u należy do podzbioru B. Skonstruujemy teraz listę zdań logicznych, które zakodują ograniczenia na wartości zmiennych.

Każda z krawędzi uv łączy dwa wierzchołki z podzbioru A (wtedy $d(u,v) \leqslant d_A$), dwa wierzchołki z podzbioru B (wtedy $d(u,v) \leqslant d_B$) lub wierzchołki należące do różnych podzbiorów (i wtedy nie mamy ograniczenia na wagę tej krawędzi). Załóżmy bez straty ogólności, że $d_A \leqslant d_B$.

(1) Jeśli $d(u,v) \leqslant d_A$, to wierzchołki u, v mogą znajdować się gdziekolwiek.
(2) Jeśli $d_A < d(u,v) \leqslant d_B$, to co najmniej jeden z wierzchołków u, v musi znajdować się w podzbiorze B, czyli musi być spełniona alternatywa $x_u \vee x_v$.
(3) Jeśli $d_B < d(u,v)$, to wierzchołki u, v muszą znajdować się w różnych podzbiorach, czyli muszą być spełnione obie alternatywy $x_u \vee x_v$ oraz $\neg x_u \vee \neg x_v$.

Każda z alternatyw zawiera dwie zmienne (lub ich negacje), zatem do znalezienia takich wartości zmiennych, które spełniają wszystkie alternatywy, możemy użyć algorytmu 2-SAT. Algorytm ten działa w czasie liniowym od liczby zmiennych i alternatyw, w naszym przypadku będzie to zatem czas $O(n^2)$.

Jesteśmy już gotowi zaproponować pierwsze rozwiązanie naszego zadania. Ponieważ waga podzbioru jest równa wadze jednej z krawędzi grafu, więc mamy $O(n^2)$ możliwości wyboru każdej z liczb d_A i d_B. Sprawdzenie dla wszystkich tych wyborów, czy istnieje podział, i wybranie podziału dla najmniejszej sumy $d_A + d_B$ daje rozwiązanie działające w czasie $O(n^6)$. Jednak wybory te nie są zupełnie niezależne. Zauważmy, że jeśli dla pewnej pary d_A, d_B podział istnieje, to istnieje też dla pary o większych wartościach. Zatem dla ustalonej wartości d_B, którą wybierzemy na $O(n^2)$ sposobów, możemy znaleźć najmniejszą wartość d_A, używając wyszukiwania binarnego w czasie $O(\log n)$. To da nam rozwiązanie działające w czasie $O(n^4 \log n)$.

Okazuje się, że możemy istotnie zmniejszyć liczbę kandydatów na wartość d_B. Powiemy, że krawędź uv jest *wewnętrzna*, jeśli oba wierzchołki u, v znajdują się w tym samym podzbiorze. Załóżmy, że znaleźliśmy maksymalne drzewo rozpinające grafu. Każda krawędź uv nienależąca do tego drzewa wyznacza cykl (w którym pozostałe krawędzie należą do drzewa). Jeśli cykl ten ma nieparzystą długość, to co najmniej jedna z jego krawędzi musi być wewnętrzna. Z kolei jeśli cykl ten ma parzystą długość i zawiera jakąś krawędź wewnętrzną, to musi zawierać co najmniej dwie takie krawędzie (a więc co najmniej jedną krawędź drzewową).

Załóżmy, że $d_B = d(u,v)$ nie jest równe wadze żadnej krawędzi z maksymalnego drzewa rozpinającego, w szczególności krawędź wewnętrzna uv nie należy do drzewa. Jeśli ta krawędź wyznacza cykl długości parzystej, to na tym cyklu znajduje się jeszcze jedna krawędź wewnętrzna xy. Ale wtedy $d_B \geqslant d(x,y) > d(u,v)$, co daje sprzeczność. Zatem krawędź uv wyznacza cykl długości nieparzystej. Rozważmy krawędź niedrzewową xy wyznaczającą nieparzysty cykl, taką że waga $d(x,y)$ jest maksymalna; zatem $d(x,y) \geqslant d(u,v)$. Ponieważ $d(x,y)$ jest minimalną wagą na tym cyklu i cykl ten zawiera krawędź wewnętrzną, to $d_B \geqslant d(x,y)$. Tak więc w tym przypadku $d(u,v) = d(x,y)$.

Z tego wynika, że mamy $O(n)$ kandydatów na wartość d_B. Są to wagi wszystkich krawędzi należących do drzewa oraz maksimum z wag krawędzi niedrzewowych tworzących cykle o nieparzystej długości. Drzewo rozpinające można znaleźć algorytmem Kruskala w czasie $O(n \log n)$. Jeśli wykonamy dwukolorowanie tego drzewa w czasie $O(n)$, to będziemy mogli dla każdej niedrzewowej krawędzi odpowiadać w czasie stałym, czy tworzy ona cykl nieparzystej długości.

Ostatecznie dostajemy algorytm działający w czasie $O(n^3 \log n)$.

Tomasz IDZIASZEK