2

Pokrycie zbioru

Problem pokrycia zbioru odegrał w teorii algorytmów aproksymacyjnych podobną rolę jak problem najliczniejszego skojarzenia w teorii algorytmów dokładnych — jego badanie doprowadziło do odkrycia podstawowych technik dziedziny. Problem ten nadaje się też wyjątkowo dobrze do naszych celów. Na jego przykładzie można w prosty sposób wyjaśnić podstawowe techniki projektowania algorytmów aproksymacyjnych. W tym rozdziale omówiłem techniki kombinatoryczne: podejście zachłanne oraz rozwarstwianie. W części II, korzystając z tego samego problemu, wprowadziłem dwie podstawowe techniki oparte na programowaniu liniowym: zaokrąglanie i schemat prymalno–dualny. Ze względu na swoją ogólność problem pokrycia zbioru ma liczne, często zaskakujące zastosowania. W tym rozdziale omówiłem jedno z nich — problem najkrótszego nadsłowa (w rozdziale 7 podałem lepszy algorytm dla tego problemu).

Pierwszą strategią, którą próbuje się zastosować, projektując algorytm aproksymacyjny, jest zapewne strategia zachłanna. Nawet jeśli ta strategia nie działa, dowód tego faktu za pomocą odpowiedniego kontrprzykładu daje doskonały wgląd w strukturę kombinatoryczną rozważanego problemu. Co ciekawe, w wypadku problemu pokrycia zbioru nie możemy liczyć na lepszy algorytm niż zachłanny (patrz rozdz. 29).

Problem 2.1 (pokrycie zbioru). Dla danego n-elementowego uniwersum U rodziny jego podzbiorów $\mathcal{S} = \{S_1, \ldots, S_k\}$ oraz funkcji kosztu $c : \mathcal{S} \to \mathbf{Q}^+$ należy znaleźć najtańszą podrodzinę $\mathcal{S}$ pokrywającą U.

Częstością elementu $e \in U$ będziemy nazywali liczbę zbiorów w $\mathcal{S}$, które zawierają e. Przydatnym parametrem okaże się częstość najczęstszego elementu. Oznaczmy tę wielkość przez f. Algorytmy aproksymacyjne dla problemu pokrycia zbioru osiągają z reguły jeden z dwóch współczynników aproksymacji: $O(\log n)$ lub f. Jasne jest, że w ogólnym przypadku żaden z nich nie jest lepszy od drugiego. Przypadek $f = 2$ jest równoważny problemowi pokrycia wierzchołkowego (patrz zad. 2.7), dla którego skonstruowaliśmy w rozdziale 1 algorytm 2-aproksymacyjny.

2.1. Algorytm zachłanny

Zastosowanie strategii zachłannej do problemu pokrycia zbioru jest bardzo naturalne: wybieramy zbiór, który najtaniej pokrywa nowe elementy, po czym usuwamy pokryte elementy itd. aż wszystkie elementy zostaną usunięte. Niech C będzie zbiorem elementów już pokrytych w danej iteracji algorytmu. *Kosztownością* zbioru S będziemy nazywali średni koszt pokrycia nowego elementu za pomocą tego zbioru, tzn. $c(S)/|S - C|$. *Ceną* elementu jest koszt jego pokrycia. Kiedy wybierany jest kolejny zbiór S, jego koszt jest rozkładany na elementy S; w ten sposób są ustalane ich ceny.

Algorytm 2.2 (algorytm zachłanny dla problemu pokrycia zbioru)

1. $C \leftarrow \emptyset$.
2. Dopóki $C \neq U$, wykonuj:
   znajdź najmniej kosztowny zbiór S,
   niech $\alpha = \frac{c(S)}{|S-C|}$, tzn. kosztowność S,
   wybierz S i dla każdego $e \in S - C$ przyjmij price$(e) = \alpha$,
   $C \leftarrow C \cup S$.
3. Wypisz wybrane zbiory.

Niech $e_1, \ldots, e_n$ będzie ciągiem wszystkich elementów U w kolejności ich pokrywania przez algorytm 2.2 (kolejność elementów pokrywanych w tej samej iteracji rozstrzygamy dowolnie).

Lemat 2.3. *Dla każdego $k \in \{1, \ldots, n\}$, price$(e_k) \leqslant \mathrm{OPT}/(n - k + 1)$.*

Dowód. W dowolnej iteracji algorytmu elementy jeszcze nie pokryte można pokryć zbiorami o łącznym koszcie nie większym niż OPT. Musi więc istnieć zbiór, wcześniej nie użyty, o kosztowności nie większej niż $\mathrm{OPT}/|\overline{C}|$, gdzie $\overline{C} = U - C$. W iteracji, w której pokrywany jest element e_k, zbiór $\overline{C}$ zawiera co najmniej $n-k+1$ elementów. Ponieważ e_k jest pokrywany przez najmniej kosztowny zbiór, to

$$\mathrm{price}(e_k) \leqslant \frac{\mathrm{OPT}}{|\overline{C}|} \leqslant \frac{\mathrm{OPT}}{n-k+1} \qquad \square$$

Z lematu 2.3 wynika bezpośrednio poniższe twierdzenie.

Twierdzenie 2.4. *Algorytm 2.2 jest algorytmem H_n-aproksymacyjnym dla problemu pokrycia zbioru, gdzie $H_n = 1 + \frac{1}{2} + \cdots + \frac{1}{n}$.*

Dowód. Ponieważ koszt każdego wybranego zbioru jest rozkładany na nowe elementy pokrywane przez ten zbiór, łączny koszt pokrycia jest równy $\sum_{k=1}^{n} \mathrm{price}(e_k)$. Z lematu 2.3 wynika, że koszt ten jest nie większy niż $(1 + \frac{1}{2} + \cdots + \frac{1}{n}) \cdot \mathrm{OPT}$. $\square$