Problem k-minimalnego drzewa rozpinającego

Gabriel Budziński
254609

May 24, 2023

1 Wprowadzenie

Weźmy graf nieskierowany $G = (V, E)$ o n wierzchołkach $w \in V$, nieujemnych kosztach c_e krawędzi $e \in E$ oraz liczbę $k \in \mathbb{N}$. Problem k-minimalnego drzewa rozpinającego (*ang.* kMST - k-minimal spanning tree, MSkT - minimal spanning k-tree) polega na poszukiwaniu drzewa w G o minimalnym koszcie, w które wchodzi co najmniej k wierzchołków G. Problem ten jest NP-trudny nawet dla V należących do płaszczyzny Euklidejskiej. Problem ten jest silnie związany z innym, występującym we wcześniejszych latach w literaturze [1] - minimum weight k-cardinality tree, którego rozwiązaniem jest znalezienie w grafie G poddrzewa o k krawędziach.

2 k-cardinality tree

2.1 Opis problemu

Weźmy graf $G = (V, E)$ ze zbiorem wierzchołków V i krawędzi E. Moce zbiorów V i E to odpowiednio $n = |V|$ oraz $m = |E|$. Dla każdej krawędzi $e \in E$ dana jest waga $w(e) \in \mathbb{R}$, a waga zbioru $E' \subseteq E$ jest definiowana jako $\sum_{e \in E'} w(e)$.

Drzewem w G jest podgraf $T = (V(T), E(T))$ taki, że T nie zawiera cykli i jest spójny. Będziemy używać notacji $w(T)$ opisując $w(E(T))$. Moc $|T|$ zbioru T jest mocą $E(T)$. Dla zadanego k, gdzie $1 \leqslant k \leqslant n - 1$ k-cardinality tree jest drzewem T o mocy $|T| = k$. Jeśli $k = n - 1$ to T jest drzewem rozpinającym G. Zadane jest znalezienie takiego T, że $w(T) = \min_{T' \subseteq G} w(T')$. Dla $k = n - 1$ takim T jest minimalne drzewo rozpinające, które można znaleźć w czasie wielomianowym algorytmem zachłannym (Kruskal [2], Prim [3]). Dla ustalonego k problem jest różnież rozwiązywalny przez wyliczenie wszystkich możliwych drzew. Jeśli mamy wagi zadane są dla wierzchołków, to możemy rozpatrywać graf krawędziowy do zadanego grafu G.

2.2 Zastosowania w praktyce

Powyższy problem pojawia się w najmie pól naftowych [4]. Rząd ma następującą regułę "50%" obejmującą morskie pola naftowe: jeśli firma najęła pole naftowe ma ona ustaloną liczbę lat, dajmy na to 5, aby eksploatować to pole. Po upływie tego czasu firma ma obowiązek zwrócić co najmniej 50% najętego pola. Ponadto, oddawana część pola musi być spójna. Oczywistym celem z punktu widzenia firmy jest zwrot częsci o najmniejszej wartości (i zachowanie części o wartości największej). W pracy [4] pola naftowe mają postać prostokąta podzielonego na mniejsze kwadraty. Firma, która najmuje pole ma 5 lat na zebranie informacji o wartości w_i każdego z podkwadratów. Część pola, którą firma odda odpowiada podzbiorowi co najmniej 50% podkwadratów, który jest spójny i ma najmniejszą całkowitą wartość wszystkich w_i. Aby zamodelować spójność weźmy graf liniowy do oczekiwanego, który jest grafem kratowym (patrz 1). Spójny podzbiór kwadratów w prostokącie odpowiada spójnemu podgrafowi G. Ponieważ wagi kwadratów odpowiadają wagom wierzchołków G, to rozwiązując problem k-CARD TREE w grafie wierzchołkowym do G, gdzie $k \geqslant \frac{n}{2}$ mamy optymalną część pola do zwrotu.

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15

Table 1: Pole naftowe

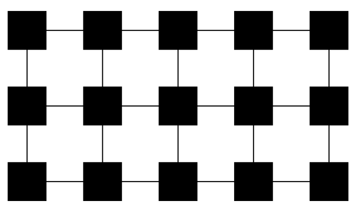

Figure 1: Graf G

2.3 Złożoność problemu k-CARD TREE

Pokażemy, że problem drzewa Steinera, który jest powszechnie znanym z bycia silnie NP-trudnym [5], może być zredukowany do k-CARD TREE. W uprzednim problemie zadany jest podzbiór $S \subseteq V$ i próbujemy znaleźć drzewo Steinera o minimalnej wadze, czyli drzewo T w G takie, że $S \subseteq V(T)$ oraz $w(T)$ jest najmniejsza. Dowód redukcji znajduje się w pracy [1].

2.4 Najnowsze metody poszukiwania rozwiązań

2.4.1 Przeszukiwanie sąsiedztwa

W pracy [6] podano wariant metody *Variable neighbourhood search* (VNS [7]) przygotowany do problemu k-CARD TREE. Przestrzeń rozwiązań $\mathcal{S}_k$, będąca zbiorem wszystkich podrzew T grafu G o dokładnie k krawędziach. Moc tej przestrzeni to $|\mathcal{S}_k| = \binom{n}{k+1} \cdot O(k^k)$. Odległość między dwoma drzewami T_1 oraz T_2 zdefiniowano jako:

- $\rho(T_1, T_2) = |E_{T_1} \backslash E_{T_2}| = |E_{T_2} \backslash E_{T_1}|$
- $\eta(T_1, T_2) = |V_{T_1} \backslash V_{T_2}| = |V_{T_2} \backslash V_{T_1}|$

Wprowadzona przez Uroševića, Brimberga i Mladenovića metoda *Variable neighbourhood decomposition search* (VDNS [8]) rozszerza VNS na dwupoziomowy VNS po dekompozycji problemu.

Powyższa metoda porównana została ze standardowym VNS oraz dwoma implementacjami tabu search (TS-1 [9], TS-2 [10]). Testy przeprowadzone zostały dla $n = |V|$ z przedziału [500,5000]. Na podstawie eksperymentów wykazano lepsze od konkurencji wyniki algorytmu VNDS, zarówno w postaci rozwiązań bliżej optimum, jak i krótszego czasu wykonania.

2.4.2 Branch-and-bound

W pracy [11] rozszerzającej podejścia *branch-and-bound* i *branch-and-cut* odpowiednio z prac [12] oraz [13] rozważany jest ukorzeniona wersja problemu k-CARD TREE, zwana dalej RKCTP, gdzie zadany jest wierzchołek $r \in V$, a rozwiązanie musi zawierać r.

Celem zapronowanej w pracy metody jest rozwiązanie k-CARD TREE (zwanego tu KCTP) jako sekwencji $n-k+1$ nierosnących podproblemów RKCTP. Metoda bazowana jest na następującym pomyśle: mając zadany wierzchołek $v \in V$, weźmy $G' = G - v$. KCTP w G (którego rozwiązanie przedstawmy jako KCTP(G)) zawiera wierzchołek v lub nie. Jeśli KCTP(G) zawiera v, to problem zredukowany jest od RKCTP zakorzenionego w v (którego optymalne rozwiązanie przedstawmy jako RKCTP(G, v)). W przeciwnym przypadku v nie należy do optymalnego rozwiązania KCTP w G i może być odrzycony. Ta obserwacja prowadzi do następującego wniosku:

$$KCTP(G) = min\{RKCTP(G, v), KCTP(G')\}$$

Problem KCTP został więc sprowadzony do sekwencji podproblemów RKCTP, które będą rozwiązywane metodą *branch-and-bound*. Waskie ograniczenia na optymalne rozwiązanie RKCTP z pracy [14] oraz prosta i wydajna strategia rozgałęzień są kluczowe do minimalizacji kosztów przeszukiwania. Aby rozwiązać RKCTP metodą *branch-and-bound* definiujemy podproblem P na zbiorach krawędzi (S_1, S_0). Zbiór S_1 jest złożony z krawędzi drzewa zakorzenionego w r, zwanych *ustalonymi krawędziami*. Zbiór S_0 jest zbiorem *zakazanych krawędzi*. S_1 jest rozwiązaniem P jeśli ma dokładnie k krawędzi i nie ma żadnych krawędzi z S_0. W podproblemie P, jeśli S_1 jest rozwiązaniem to P jest podproblemem terminalnym w drzewie przeszukiwań. W przeciwnym przypadku, jeśli $E\backslash(S_1 \cup S_0)$ jest niepusty to wybieramy krawędź e z tego zbioru taką, że $S_1 \cup \{e\}$ też jest drzewem oraz definiujemy dwa nowe podproblemy: krawędź e dodajemy do zbiorów S_1 albo S_0. Poprzez te dwie operacje podczas kroku rozgałęziania tworzone są dwa nowe podproblemy RKCTP takie, że optymalne osiągalne rozwiązanie P istnieje tylko w jednej z gałęzi. W tej metodzie rozważaną krawędzią e jest krawędź z $E\backslash(S_1 \cup S_0)$ o minimalnej wadze. Ponadto strategią wyboru gałęzi jest *depth-first-search*.

Podczas testów metodę porównano z dwoma innymi dającymi rozwiązanie KCTP lub jego ograniczenie dolne: k pierwszych kroków algorytmu Kruskala [2] oraz ogranicznie Kataoka'i [14]. Zaproponowana w pracy metoda działała lepiej od obydwu wymienionych, częściej zwracając rozwiązanie optymalne oraz dając lepsze ograniczenia dolne.

3 *k*-spanning tree

3.1 Heurystyki

3.2 Algorytm hybrydowy: tabu search i kolonia mrówek

3.3 *k*-spanning tree na okręgach

References

[1] M. Fischetti, H. W. Hamacher, K. Jornsten, and F. Maffioli, *Weighted k-cardinality trees: Complexity and polyhedral structure*. PhD thesis, Universität Kaiserslautern, 1992.

[2] J. B. Kruskal, "On the shortest spanning subtree of a graph and the traveling salesman problem," *Proceedings of the American Mathematical Society*, vol. 7, 1956.

[3] R. Prim, "Shortest connection networks and some generalizations," *Bell System Technical Journal*, vol. 36, 1957.

[4] H. W. Hamacher and K. Jörnsten, "Optimal relinquishment according to the norwegian petroleum law: A combinatorial optimization approach," *Energy, Natural Resources and Environmental Economics*, 1993.

[5] M. R. Garey and D. S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-Completeness*. Freeman and Company, 1979.

[6] D. Urošević, J. Brimberg, and N. Mladenović, "Variable neighborhood decomposition search for the edge weighted k-cardinality tree problem," *Computers and operational research*, vol. 31, 2004.

[7] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers and operational research*, vol. 24, 1997.

[8] P.Hansen, N. Mladenović, and D. Perez-Britos, "Variable neighborhood decomposition search," *Journal of Heuristics*, 2001.

[9] N. Mladenović and D. Urošević, "Variable neighborhood search for the k-cardinality tree," *Metaheuristics: Computer Decision-Making*, 2001.

[10] J. Kurt and L. Arne, "Tabu search for weighted k-carfinality trees," *Asia-Pacific Journal of Operational Research*, vol. 14, 1997.

[11] L. Simonetti, F.Protti, Y. Frota, and C. de Souza, "New branch-and-bound algorithms for k-cardinality tree problems," *Electronic notes in descrete mathematics*, vol. 37, 2011.

[12] F. Quintao, A. Cunha, and G. Mateus, "Integer programming formulations for the k-cardinality tree problem," *Electronic Notes in Discrete Mathematics*, vol. 30, 2008.

[13] M. Chimani, M. Kandyba, and I. L. nad P. Mutzel, "Obtaining optimal k-cardinality trees fast," *ACM Journal of Experimental Algorithmics*, vol. 14, 2009.

[14] S. Kataoka, N. Araki, and T. Yamada, "Upper and lower bounding procedures for the minimum rooted k-subtree problem," *European Journal of Operational Research*, vol. 122, no. 3, 2000.