Asymptotically optimal approach to a given diameter Undirected MST problem on random input data.

E. Gimadi, A. Shevyakov, and E. Shin

IM SO RAN, NGU, Novosibirsk, Russia

OPCS 2019, Novosibirsk, Academgorodok,
26–30 Aug, 2019

SUMMARY

In consideration:

• On asymptotically optimal approach to solvability hard discrete optimization problems
• Statenent of the Given-Diameter MST problem.
• An approximation algorithm $\mathcal{A}$ for solving the problem.
• A probabilistic analysis of Algorithm
• Sufficient conditions of asymptotic solvability the Given-Diameter MST problem.

Осн. фактором, опред-м реализуемость алгоритмов, явл. размерность (длина записи входа) задачи, которая в 50-70 г.г. прошлого века ассоциировалась с понятием "проклятия размерности" (curse of dimensionality) — экспоненциальным ростом времени решения задачи при увеличении длины записи входных данных (Ричард Беллман, 1961 г.). В противовес этому в рамках асимптотически точного (asymptotically optimal) подхода размерность задачи является нашим другом и союзником.

К н/вр. определилось немало успешных примеров реализации подхода к решению таких задач дискретной оптимизации, как задачи маршрутизации, многоиндексные задачи о назначениях, задачи кластеризации, задачи размещения, покрытия, экстремальные задачи на графах и сетях и т.п.

Введение

Труднорешаемость этих задач обуславливает актуальность разработки эффективных алгоритмов приближенного решения с гарантированными оценками таких показателей качества их работы как —

временная сложность,

точность,

надежность срабатывания

———————————————————————————-

Книга: Гимади Э.Х., Хачай М.Ю. «Экстремальные задачи на множествах перестановок» / – Екатеринбург: “Изд-во Учебно-методический центр УПИ”, 2016. – 210 с.

Performance guarantees and asymptotical optimality

$\varepsilon_A(n)$ – оценка относ. погрешности алгоритма A

на детерм. входах определяется неравенством:

$$\frac{|W_A(I) - OPT(I)|}{OPT(I)} \leq \varepsilon_A(n).$$

$\delta_A(n)$ – вероятность несрабатывания алгоритма A

на случ. входах определяется вероятностным нер-ом:

$$\mathbb{P}\left\{\frac{|W_A(I) - OPT(I)|}{OPT(I)} > \varepsilon_A(n)\right\} \leq \delta_A(n).$$

Асимптотическая точность алгоритма

Прибл. алгоритм A асимпт. точен, если при $n \to \infty$

$$\varepsilon_A(n) \to 0, \quad \delta_A(n) \to 0.$$

Ю.И. Журавлев: почти всегда...

Рис.: На конференции по распознаванию образов. Будва 2013

Первый пример асимпт. точного подхода:

Гимади–Перепелица (1969)

Алгоритм ИБГ труд-ти $\mathcal{O}(n^2)$ для ЗК со случ. дискретной ф.р. эл-в матрицы (c_{ij}) расстояний

$$p(k) = \mathbb{P}\{c_{ij} = k\}, \ \ 1 \le k \le K_n,$$

асимпт. точен при

$$\sum_{k=1}^{K_n} \frac{1}{p(1) + \ldots + p(k)} = o(n).$$

Виталий Афанасьевич Перепелица

В случае равном. распр. ИБГ асимпт. точен,

если разброс эл-в матрицы ограничен величиной $o\left(n/\log n\right)$

Теорема Петрова vs нер-ва Чебышева

Первые рез-ты по обоснованию асимптотической точности были получены с использованием нер-ва Чебышева. Позже более продуктивной оказалась

Теорема Петрова:

Пусть $X_1, \dots, X_n$ — н.сл.в. и сущ-т. полож. константы T и $h_1, \dots, h_n$ такие, что

$$\mathbb{E}e^{tX_j} \le e^{\frac{1}{2}h_j t^2}, \ (j = \overline{1, n}, \ 0 \le t \le T).$$

Обозначим: $H = \sum\limits_{j=1}^{n} h_j$. Тогда

$$\mathbb{P}\left\{\sum_{j=1}^{n} X_j > x\right\} \le \begin{cases} \exp\{-x^2/2H\} \text{ при } 0 \le x < HT, \\ \exp\{-Tx/2\} \text{ при } x \ge HT, \end{cases}$$

Примеры труднорешаемых задач с реализациями асимптотически точного подхода к их решению

• TSP и m-PSP (одного и нескольких коммивояжеров).
• Многоиндексная аксиальная задача о назначениях.
• Трехиндексная планарная m-слойная задача о назначениях.
• Задача отыскания покрытия полного взвеш. графа заданным числом несмежных циклов.
• Задачи упаковки в контейнеры и в полосу
• Задача отыскания связного остовного подграфа с максим. весом ребер в полном неориентированном графе с заданными степенями вершин.
• Задачи маршрутизации транспортных средств (VRP)
• Задача отыскания в графе минимального остовного дерева с ограниченным снизу (или сверху) диаметром.

Минимальное остовное дерево (MST)

The Minimum Spanning Tree Problem (MSTP)

is a one of the classic discrete optimization problems. Given weighted graph $G = (V, E)$, MSTP is to find a spanning tree of a minimal total weight.

The polynomial solvability of MSTP:

was shown in the classic algorithms by Boruvka (1926), Kruskal (1956) and Prim (1957).
These algorithms have complexity $\mathcal{O}(n^2)$ and $\mathcal{O}(M \log n)$, where $M = |E|$ and $n = |V|$.

Random MST

Expectation

Expectation of a weight MST on a random graph can be unexpectedly small.
For example, on a complete graph with weights of edges from class $\mathrm{UNI}(0;1)$, the weight of a MST w.h.p. (with high probability) is close to the constant 2,02... [Frieze:1985].
Similar results [Angel at al: 2011, Cooper and Frieze: 2016].

Diameter-Bounded generalization of MST

The diameter of a tree is the maximum number of edges within the tree connecting a pair of vertices.

Diameter-Bounded MST

Given a graph G and a number $d = d_n$, the goal is to find in the graph G a spanning tree T_n of minimal total weight having its diameter bounded
1) either above to given number d (d-BAMST), or
2) bounded from below to given number d (d-BBMST)

In the case 1

the problem is NP-hard for any diameter between 4 and $n - 1$, even for the edge weights equal to 1 or 2 [GJ].

In the case 2

the problem is NP-hard, because its particular case for $d = n - 1$ is the problem HAMILTONIAN PATH [GJ].

Given-Diameter MST

In current report another modification of MST problem is studied.

d-MST

We consider a given-diameter minimum spanning tree problem (d-MST) on the complete graph G_n. We introduce a polynomial-time algorithm to solve this problem and provide conditions for this algorithm to be asymptotically optimal.

A probabilistic analysis

is performed under conditions that edges weights of given graph are $UNI(a_n; b_n)$-entries (i.e. identically independent distributed random variables).

First, we describe the algorithm $\mathcal{A}'$ in the case of a directed graph, and then the algorithm $\mathcal{A}$ in the case of an undirected graph.

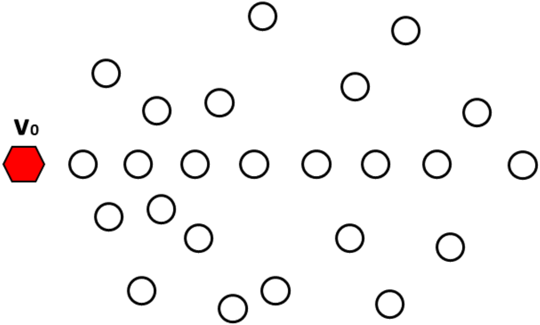
Algorithm $\mathcal{A}'$ for finding d-MST on directed graphs

Stage 1

From arbitrary vertex $v_0 \in V$ build a path

$$P(d) = (v_0, v_1, \ldots, v_d),$$

where $v_{k+1} \notin P(k)$ is closest to the v_k, $0 \leq k < d$.



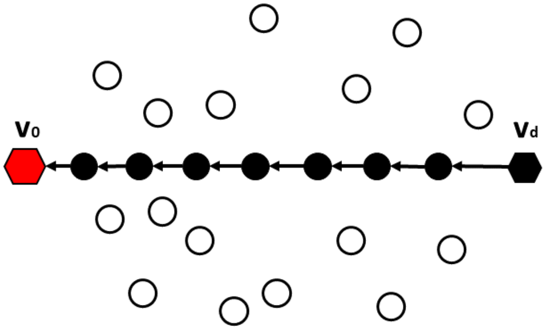
Algorithm $\mathcal{A}'$ for finding d-MST on directed graphs

Stage 1

From arbitrary vertex $v_0 \in V$ build a path

$$P(d) = (v_0, v_1, \ldots, v_d),$$

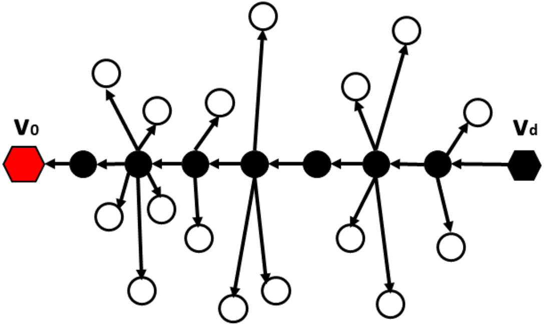
where $v_{k+1} \notin P(k)$ is closest to the v_k, $0 \leq k < d$.



Algorithm $\mathcal{A}'$ for finding d-MST on directed graphs

Stage 2

Let $P = P(d)$, $V' = V \setminus P$.
Every vertex $v' \in V'$ is connected by the shortest possible edge with a vertex $v(v') \in P \setminus \{v_0, v_d\}$.
By E' we denote the set of edges $(v', v(v'))$, $v' \in V'$.



Algorithm $\mathcal{A}'$ on directed graphs

Approximate solution $T_{\mathcal{A}'}$

As a result we obtainan approximate solution of the problem: the spanning tree $T_{\mathcal{A}'}$ with a diameter which equal to $d = d_n$, since when connecting any vertex from V' to path $P \setminus \{v_0, v_d\}$ during the Stage 2, the diameter does not change.

The weight of $T_{\mathcal{A}'}$

The weight of the resulting spanning tree $T_{\mathcal{A}'}$ is equal to

$$W_{\mathcal{A}'} = W(P) + W(E'),$$

where

$$W(P) = \sum_{k=1}^{d} c(v_{k-1}, v_k),$$

$$W(E') = \sum_{e \in E'} c_e.$$

Algorithm analysis

Time complexity

$$O(n^2)$$

Since Stage 1 is performed in time $O((n-d)^2)$.
On Stage 2 it takes about $d(n-d)$ comparison operations.

Probabilistic analysis

It is assumed that weights of graph edges are i.r.v. η from the class $UNI(a_n, b_n)$, namely, uniformly distributed on a set

$$(a_n, b_n),\ 0 < a_n \leq b_n < \infty.$$

Two ranges of parameter d

We perform analysis for two cases of values of the parameter d:

$$\text{Case 1: } \ln n \leq d < n\theta \quad \text{and} \quad \text{Case 2: } n\theta \leq d < n,$$

$$\text{where} \quad \theta = \frac{1}{e} - 1 \approx 0,63.$$

Algorithm analysis

min over k variables

Put r.i.v. $\eta_k = \min$ over k variables from the class $UNI(a_n, b_n)$;
$\xi_k = \min$ over k variables from the class $UNI(0, 1)$.

Weight of $T_{\mathcal{A}'}$

According to $\mathcal{A}'$, the weight of $T_{\mathcal{A}'}$ equal to

$$W_{\mathcal{A}'} = W(P) + W(E') = \sum_{k=n-d}^{n-1} \eta_k + \sum_{v' \in V'} \eta_{d-1} =$$

$$= \sum_{k=n-d}^{n-1} \eta_k + (n-d-1)\eta_{d-1} = (n-1)a_n + (b_n - a_n)W'_{\mathcal{A}'},$$

where $$W'_{\mathcal{A}} = \sum_{k=n-d}^{n-1} \xi_k + (n-d-1)\xi_{d-1}.$$

Algorithm analysis

Fact 1

$$EW'_{\mathcal{A}'} \le \widetilde{EW}'_{\mathcal{A}'} = \ln \frac{n}{n-d} + \frac{n-d-1}{d}.$$

Fact 2

In the case $d < n\theta$

$$\ln \frac{n-1}{n-d} < 1.$$

Fact 3

In the case 1 $(d < n\theta)$ the following inequality holds:

$$EW'_{\mathcal{A}'} \le \widetilde{EW}'_{\mathcal{A}'} = \frac{n-1}{d}.$$

Fact 4

In the case 2 $(n\theta \le d < n)$ the following estimate is correct:

$$EW'_{\mathcal{A}'} \le \widetilde{EW}'_{\mathcal{A}'} = \ln n.$$

Lemma

The Algorithm $\mathcal{A}'$ for solving the d-MST on entries $\mathrm{UNI}(a_n; b_n)$ has the following estimates of the relative error

$$\varepsilon_n = (1 + \lambda_n)\frac{(b_n - a_n)}{(n-1)a_n},$$

and the failure probability

$$\delta_n = \lambda_n \widetilde{\mathrm{EW}}'_{\mathcal{A}'},$$

where $\lambda_n > 0$.

Algorithm analysis

Main Theorem.

Let the diameter $d = d_n$ be defined so that

$$\ln n \le d < n\theta \text{ (Case 1) and } n\theta \le d < n \text{ (Case 2)}.$$

Then Algorithm $\mathcal{A}'$ solves the problem d-MST on entries $\mathrm{UNI}(a_n; b_n)$ with estimates

$$\varepsilon_n = \mathcal{O}\left(\frac{b_n/a_n}{\phi(n)}\right), \quad \delta_n = \frac{1}{n},$$

where

$$\phi(n) = \begin{cases} d, \text{ in Case 1}, \\ n/\ln n, \text{ in Case 2}. \end{cases}$$

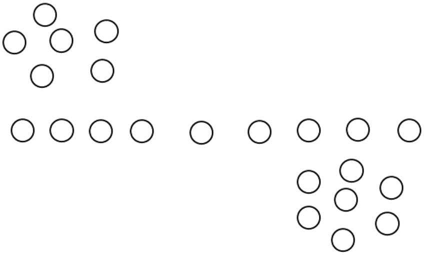
So Algorithm $\mathcal{A}'$ asymptotically optimal, if

$$\frac{b_n}{a_n} = \begin{cases} o(d), & \text{in Case 1}, \\ o\left(\frac{n}{\ln n}\right), & \text{in Case 2}. \end{cases}$$

Algorithm $\mathcal{A}$ for finding d-UMST (in the case of undirected graphs)

Stage 1

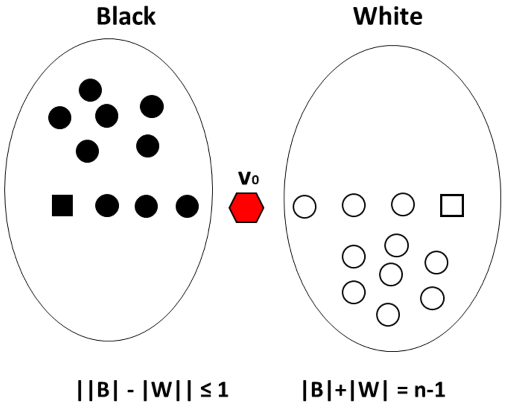
Choose an arbitrary vertex v_0 and divide all other vertices into two sets B and W:



Algorithm $\mathcal{A}$ for finding d-UMST (in the case of undirected graphs)

Stage 1

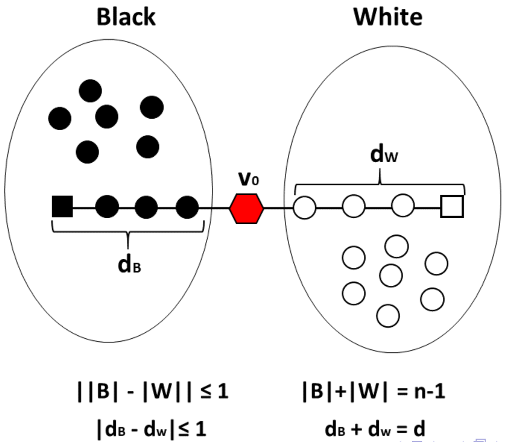
Choose an arbitrary vertex v_0 and divide all other vertices into two sets B and W:



Algorithm $\mathcal{A}$ for finding d-UMST (in the case of undirected graphs)

Stage 2

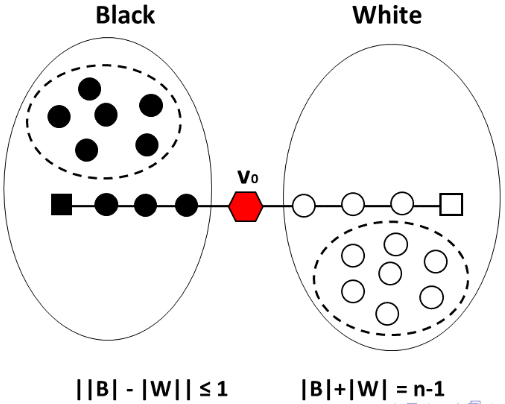
In each set starting at v_0 find a path of a certain length using the approach "go to the nearest unvisited vertex".



Algorithm $\mathcal{A}$ for finding d-UMST (in the case of undirected graphs)

Stage 3

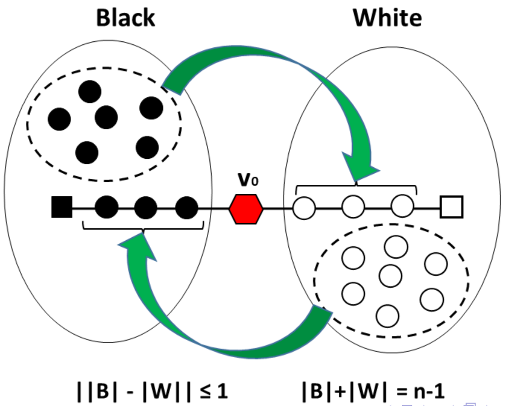
Connect the white remaining vertices to the nearest inner black vertices of the path, and the black remaining vertices to the nearest inner white vertices of the path.



Algorithm $\mathcal{A}$ for finding d-UMST (in the case of undirected graphs)

Stage 3

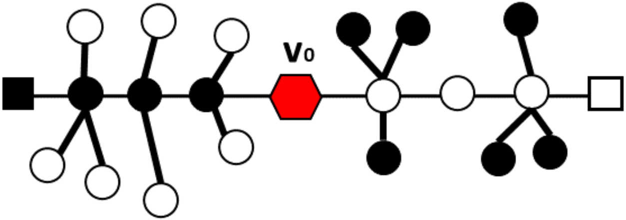
Connect the white remaining vertices to the nearest inner black vertices of the path, and the black remaining vertices to the nearest inner white vertices of the path.



Algorithm $\mathcal{A}$ for finding d-UMST (in the case of undirected graphs)

Stage 3

Connect the white remaining vertices to the nearest inner black vertices of the path, and the black remaining vertices to the



Conclusion

It would be interesting to investigate
(a) the Random d-UMST problem on input data with infinite support like exponential or trunketed-normal distribution;
(b) the problem of finding several edge-disjoined spanning trees with a diameter which is given or bounded.
(c) Conduct a probabilistic analysis of the Algorithm $\mathcal{A}'$ on an undirected graph with the correct account of the dependence of of random objects that occur along the algorithm.

FINISH!

THANK YOU FOR ATTENTION!