

(1 + ϵ) 近似 MST を高速に構築する ハイブリッドモデル分散アルゴリズム

平岡 昇真

北村 直暉

泉 泰介

増澤 利光

概要

分散グラフアルゴリズムとは、各計算機を頂点、計算機間の通信リンクを辺と見なして、ネットワーク自身をグラフアルゴリズムの入力と見なして扱い、そのうえでグラフに関する諸問題を解く枠組みである。Augustine ら [1] は、分散グラフアルゴリズムのモデルの一つとして、ハイブリッドモデルを提案した。ハイブリッドモデルでは各計算機は同期して動作を行い、ローカル通信では各頂点は各ラウンドで入力として与えられたグラフにおいて隣接している各頂点と任意の長さのビットのメッセージの送受信が可能である。一方、グローバル通信では ID を知っている任意の頂点に対して $O(\log n)$ ビットの通信を行うことができる。ただし、グローバルネットワークでは各頂点が 1 ラウンドで送受信可能なメッセージ数は $O(\log n)$ 個に制限されている。また各頂点は初期状態では自身と自身と隣接する頂点の ID のみを知っているものとする。Feldmann ら [2] は、ローカル通信において各ラウンドで各隣接頂点に送るメッセージが $O(\log n)$ ビットに制限されているモデルを提案している。本研究では Feldmann 等によって提案されたハイブリッドモデルにおける全域木問題や最小全域木について焦点を当てる。ハイブリッドモデルにおける ST と MST の既存研究においては $O(\log^2 n)$ ラウンドの決定性アルゴリズムが知られている [2]。一方で $O(\log^2 n)$ ラウンドよりも高速なアルゴリズムが構築可能であるかという問題は未解決な問題である。ST に関する研究では入力のグラフをカクタスグラフに制限した場合、 $O(\log n)$ ラウンドのアルゴリズムが知られている。また、グローバル通信において、各ラウンドで送れるメッセージ数とメッセージビットを $O(\log^5 n)$ に拡張したモデルでは $O(\log n)$ ラウンドで解けることが示されている [3]。本稿では [3] の結果と同様にグローバル通信におけるメッセージ数とメッセージビットを拡張したモデルにおける $(1 + \epsilon)$ -近似 MST 問題について考える。具体的にはグローバル通信におけるメッセージ数とメッセージビットを $O(\log_{1+\epsilon} W \log^5 n)$ としたときに、 $O(\log n)$ ラウンドで $(1 + \epsilon)$ -近似 MST を構築する乱択アルゴリズムを提案する。ここで W はグラフにおける辺の重みの種類数を表す。この結果は MST の辺の重みの種類数が定数である時、 $O(\log^5 n)$ ラウンドで MST を解くことが可能であることを示しているため、[3] の結果を一般化したものであるということが出来る。

References

- [1] John Augustine, Kristian Hinnenthal, Fabian Kuhn, Christian Scheideler, and Philipp Schneider. Shortest paths in a hybrid network model. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 1280–1299, 2020.
- [2] Michael Feldmann, Kristian Hinnenthal, and Christian Scheideler. Fast hybrid network algorithms for shortest paths in sparse graphs. In *Proceedings of 24th International Conference on Principles of Distributed Systems (OPODIS)*, pages 31:1–31:16, 2020.
- [3] Thorsten G\"otter, Kristian Hinnenthal, Christian Scheideler, and Julian Werthmann. Time-optimal construction of overlay networks. In *Proceedings of the 2021 ACM Symposium on Principles of Distributed Computing (PODC)*, pages 457–468. ACM, 2021.