Brief Announcement: Beeping a Maximal Independent Set Fast

Stephan Holzer⋆ and Nancy Lynch⋆

Massachusetts Institute of Technology (MIT), Cambridge, USA
holzer@mit.edu, lynch@csail.mit.edu

Abstract. We adapt a recent algorithm by Ghaffari for computing a Maximal Independent Set in the LOCAL model, so that it works in the significantly weaker BEEP network model. For networks with maximum degree Δ, our algorithm terminates locally within time $O((\log \Delta + \log(1/\varepsilon)) \cdot \log(1/\varepsilon))$, with probability at least $1 - \varepsilon$. Moreover, the algorithm terminates globally within time $O(\log^2 \Delta) + 2^{O(\sqrt{\log \log n})}$ with high probability in n, the number of nodes in the network. The key idea of the modification is to replace explicit messages about transmission probabilities with estimates based on the number of received messages.

1 Introduction

Computing a Maximal Independent Set (MIS) is a widely studied problem in distributed computing theory. One of the weakest models of communication in which this problem has been studied is the BEEP model, e.g., [1,3]. For that model, we obtain:

Theorem 1 (Global termination complexity). *Our algorithm computes an MIS within $O(\log^2 \Delta) + 2^{O(\sqrt{\log \log n})}$ rounds w.h.p., where n is the number of nodes in the network and $\Delta \leq n$ is the maximum degree of any node.*

This improves over the state-of-the-art algorithm [3] for a large range of values of the parameter Δ. We obtain this bound by translating Ghaffari's algorithm [2] for the LOCAL model into the BEEP model. We adapt the proof of Theorem 1.2 of [2], which infers a global bound from a local bound stated in Theorem 1.1 of [2]. In particular we state a local bound in Theorem 2 and use Theorem 2 to replace Theorem 1.1 of [2] in the proof of Theorem 1.2 of [2]. We argue that this replacement works also for the BEEP model.

Theorem 2 (Local termination complexity). *In our algorithm, for each node v, the probability that v decides whether it is in the MIS within $O((\log \Delta + \log(1/\varepsilon)) \cdot \log(1/\varepsilon))$ rounds is at least $1 - \varepsilon$.*

Note that this local bound is only a factor of $O(\log(1/\varepsilon))$ larger than the $O(\log \Delta + \log(1/\varepsilon))$ bound for the algorithm in the LOCAL model [2]. The key idea in the proof and algorithm of Theorem 2 is that, instead of maintaining full information about its neighbors' states, a node keeps a single binary estimate for the aggregate state of its entire neighborhood.

⋆ Supported by: AFOSR Contract Number FA9550-13-1-0042, NSF Award 0939370-CCF, NSF Award CCF-1217506, and NSF Award CCF-AF-1461559.

2 Models and Definitions

LOCAL *and* BEEP *Models:* In both models, the network is abstracted as an undirected graph $G = (V, E)$ where $|V| = n$. All nodes wake up simultaneously. Communication occurs in synchronous rounds. In the LOCAL model (e.g., [2]), each node knows its graph neighbors. Nodes communicate reliably, where in each round nodes can exchange an arbitrary amount of information with their immediate graph neighbors. On the other hand, in the BEEP model (e.g., [1,3]), nodes do not know their neighbors. Nodes communicate reliably and a node can choose to either beep or listen. If a node v listens in slot[1] t it can only distinguish between silence (no neighbor beeps in slot t) or the presence of one or more beeps (at least one neighbor beeps in in slot t).

Graph-related Definitions: A set of vertices $I \subseteq V$ is an *independent set* of G if no two nodes in I are neighbors in G. An independent set $I \subseteq V$ is a *maximal independent set (MIS)* of G if, for all $v \in V \setminus I$, the set $I \cup \{v\}$ is not independent. An event occurs *with high probability* (w.h.p.), if it occurs with probability at least $1 - n^{-c}$ for some constant $c \geq 1$.

3 Algorithms

The MIS algorithm of [2] runs for $R := \beta(\log \Delta + \log(2/\varepsilon)) = O(\log \Delta + \log(1/\varepsilon))$ rounds, where $\beta = 1300$. In each round t, each node v has a *desire-level* $p_t(v)$ for joining the MIS, which initially is set to $p_0(v) = 1/2$. Ghaffari [2] calls the total sum of the desire-levels of neighbors of v its *effective-degree* $d_t(v)$, i.e., $d_t(v) = \sum_{u \in N(v)} p_t(u)$. The desire-levels change over time depending on whether or not $d_t(v) \geq 2$. In each round, node v gets *marked* with probability $p_t(v)$. If v is marked, and no neighbor of v is marked, v joins the MIS and gets removed along with its neighbors. Using the power of the LOCAL model, in each round t, nodes exchange exact values of $p_t(u)$ with all their neighbors.

In implementing this algorithm in the BEEP model, we do not require that a node v learn the exact values of $p_t(u)$ for all neighbors u in order to compute $d_t(v)$. Instead, we allow node v to decide, based on how many beeps v receives within a certain number of rounds, whether $d_t(v)$ is more likely to be larger than 1 or smaller than 3. To make this decision, we define an *interval* to consist of $I := 1000(\ln(1500) + \ln(2/\varepsilon))$ consecutive slots. We use one interval in the BEEP model to emulate each round of the algorithm [2] in the LOCAL model. During part of an interval, the algorithm computes the ratio of the number of beeps received ($b_t(v)$) to the total number of slots in which v listened during the interval ($c_t(v)$). Node v decides to update its desire-level:

$$p_{t+1}(v) = \begin{cases} p_t(v)/2, & \text{if } b_t(v)/c_t(v) > \frac{5}{6} \\ \min\{2p_t(v), 1/2\}, & \text{if } b_t(v)/c_t(v) \leq \frac{5}{6} \end{cases} .$$

Thus, we replace the condition $d_t(v) \geq 2$ in the algorithm of [2] by the condition $b_t(v)/c_t(v) > \frac{5}{6}$.

[1] To disambiguate, we refer to the rounds of the BEEP model as *slots*.

4 Local Analysis of our MIS Algorithm

We demonstrate that for each node v, the accuracy of deciding whether v's effective degree is high or low is good enough to translate the algorithm of [2] into the Beep model, i.e., our algorithm does not require v to learn exact desire-values of its neighbors. We say node v is a *good node* in an interval t, if at the end of the interval the following three conditions are satisfied: (i) $c_t(v) > I/3$, (ii) if $b_t(v)/c_t(v) > \frac{5}{6}$, then $d_t(v) \geq 1$, and (iii) if $b_t(v)/c_t(v) \leq \frac{5}{6}$, then $d_t(v) \leq 3$.

Lemma 1. *A good node v always (i) draws correct conclusions about whether its effective degree is high or low, and (ii) adjusts its desire-values in the same way as in the algorithm of [2].*

Lemma 1 allows us to modify the analysis of [2] to obtain statements about good nodes. To argue that this applies to large parts of the graphs, we show that most nodes are good:

Lemma 2. *For any interval, the probability that a node v is good is at least $1 - 2e^{-I/1000}$.*

To prove Lemma 2, we use a Chernoff Bound to bound the probability that $b_t(v)/c_t(v)$ reflects whether the effective degree is high or low based on the condition $b_t(v)/c_t(v) > \frac{5}{6}$ rather than $d_t(v) \geq 2$.

We define two kinds of *golden intervals* for a node v, by analogy with the definition of *golden rounds* in [2]: Interval t is a golden interval of type-1, if $b_t(v)/c_t(v) \leq \frac{5}{6}$ and $p_t(v) = 1/2$. Interval t is a golden interval of type-2, if $b_t(v)/c_t(v) > \frac{5}{6}$ and at least $d_t(v)/10$ of $d_t(v)$ is contributed by neighbors u with $d_t(u) \leq 3$. Using Lemma 1 and Lemma 2 we show:

Lemma 3. *In each type-1 (resp., type-2) golden interval, with probability at least $1/1000$, v joins the MIS (resp., one of v's neighbors joins the MIS). If $R/13$ intervals are golden, then the probability that v has not decided whether it is in the MIS during the first R intervals is at most $\varepsilon/2$.*

Lemma 4. *By the end of interval R, with probability at least $1 - 1500e^{-I/1000}$, either v has joined, or has a neighbor in, the (computed) MIS, or at least one of its golden interval types appeared at least $R/13$ times.*

We prove Lemma 4 by adapting ideas of [2]. Theorem 2 follows from combining Lemma 3 and Lemma 4.

References

1. Yehuda Afek, Noga Alon, Ziv Bar-Joseph, Alejandro Cornejo, Bernhard Haeupler, and Fabian Kuhn. Beeping a Maximal Independent Set. *Distributed Computing*, 26(4):195–208, 2013.
2. Mohsen Ghaffari. An improved distributed algorithm for Maximal Independent Set. In *Proceedings of the 2015 ACM-SIAM Symposium on Discrete Algorithms*, pages 270–277, 2016.
3. Alex Scott, Peter Jeavons, and Lei Xu. Feedback from nature: an optimal distributed algorithm for Maximal Independent Set selection. In *Proceedings of the 2013 ACM Symposium on Principles of Distributed Computing*, pages 147–156, 2013.