Distributed Dominating Set Approximations beyond Planar Graphs*

Saeed Akhoondian Amiri[1] Stefan Schmid[2,1] Sebastian Siebertz[3,1]

[1] TU Berlin, Germany [2] Aalborg University, Denmark
[3] University of Warsaw, Poland

Abstract. The Minimum Dominating Set (MDS) problem is one of the most fundamental and challenging problems in distributed computing. While it is well-known that minimum dominating sets cannot be approximated locally on general graphs, over the last years, there has been much progress on computing local approximations on *sparse graphs*, and in particular planar graphs.

In this paper we study distributed and deterministic MDS approximation algorithms for graph classes beyond planar graphs. In particular, we show that existing approximation bounds for planar graphs can be lifted to bounded genus graphs, and present (1) a local constant-time, constant-factor MDS approximation algorithm and (2) a local $\mathcal{O}(\log^* n)$-time approximation scheme. Our main technical contribution is a new analysis of a slightly modified variant of an existing algorithm by Lenzen et al. Interestingly, unlike existing proofs for planar graphs, our analysis does not rely on direct topological arguments.

Keywords: Distributed Algorithms, LOCAL Model, Sparse Graphs, Minimum Dominating Sets

1 Introduction

This paper attends to the Minimum Dominating Set (MDS) problem, arguably one of the most intensively studied graph theoretic problems in computer science in general, as well as in distributed computing.

* Preliminary versions of this paper appeared as [2] and [3]. The research of Saeed Akhoondian Amiri has been supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (ERC consolidator grant DISTRUCT, agreement No 648527). Stefan Schmid is supported by Aalborg University's talent management programme. Sebastian Siebertz is partially supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (ERC consolidator grant DISTRUCT, agreement No 648527) and by the National Science Centre of Poland via POLONEZ grant agreement UMO-2015/19/P/ST6/03998, which has received funding from the European Union's Horizon 2020 research and innovation programme (Marie Skłodowska-Curie grant agreement No. 665778).

A dominating set D in a graph G is a set of vertices such that every vertex of G either lies in D or is adjacent to a vertex in D. Finding a minimum dominating set is NP-complete [13], even on planar graphs of maximum degree 3 (cf. [GT2] in [9]), however, the problem admits a PTAS on planar graphs [4], minor closed classes of graphs with local bounded tree-width [8] and graphs with excluded minors [10], and most generally, on every graph class with polynomial expansion [11].

In this paper, we study the *distributed* time complexity of finding dominating sets, in the classic *LOCAL model* of distributed computing [19]. It is known that finding small dominating sets locally is hard: Kuhn et al. [15] show that in r rounds the MDS problem on an n-vertex graphs of maximum degree Δ can only be approximated within factor $\Omega(n^{c/r^2})$ and $\Omega(\Delta^{c'/r})$, where c and c' are constants. This implies that, in general, to achieve a constant approximation ratio, every distributed algorithm requires at least $\Omega(\sqrt{\log n})$ and $\Omega(\log \Delta)$ communication rounds. The currently best results for general graphs are by Kuhn et al. [14] who present a $(1+\epsilon)\ln\Delta$-approximation in $\mathcal{O}(\log(n)/\epsilon)$ rounds for any $\epsilon > 0$, and by Barenboim et al. [5] who present a deterministic $\mathcal{O}((\log n)^{k-1})$-time algorithm that provides an $\mathcal{O}(n^{1/k})$-approximation, for any integer parameter $k \geq 2$.

For sparse graphs, the situation is more promising. For graphs of arboricity a, Lenzen and Wattenhofer [18] present a forest decomposition algorithm achieving a factor $\mathcal{O}(a^2)$ approximation in randomized time $\mathcal{O}(\log n)$, and a deterministic $\mathcal{O}(a \log \Delta)$ approximation algorithm requiring time $\mathcal{O}(\log \Delta)$ rounds. Graphs of bounded arboricity include all graphs which exclude a fixed graph as a (topological) minor and in particular, all planar graphs and any class of bounded genus. Amiri et al. [1] provide a deterministic $\mathcal{O}(\log n)$ constant factor approximation algorithm on classes of bounded expansion, a result which generalizes all known results on excluded minor classes. Czygrinow et al. [6] show that for any given $\delta > 0$, $(1+\delta)$-approximations of a maximum independent set, a maximum matching, and a minimum dominating set, can be computed in $\mathcal{O}(\log^* n)$ rounds in planar graphs, which is asymptotically optimal [17]. Lenzen et al. [16] proposed a constant factor approximation on planar graphs that can be computed locally in a constant number of communication rounds. A finer analysis of Wawrzyniak [24] showed that the algorithm of Lenzen et al. in fact computes a 52-approximation of a minimum dominating set with small message size. Wawrzyniak [23] also showed that message sizes of $\mathcal{O}(\log n)$ suffice to give a (slightly worse) constant factor approximation on planar graphs. In terms of lower bounds, Hilke et al. [12] show that there is no deterministic local algorithm (constant-time distributed graph algorithm) that finds a $(7-\epsilon)$-approximation of a minimum dominating set on planar graphs, for any positive constant ϵ.

1.1 Our Contributions

The first and main contribution of this paper is a deterministic and local constant factor approximation for MDS on graphs that we call *locally embeddable graphs*. A locally embeddable graph G excludes the complete bipartite graph $K_{3,t}$, for

$t \geq 3$, as a depth-1 minor, that is, as a minor obtained by star contractions, and furthermore satisfies that all depth-1 minors of G have constant edge density. The most prominent locally embeddable graph classes are classes of bounded genus. Concretely, our result implies that MDS can be $\mathcal{O}(g)$-approximated locally and deterministically on graphs of (both orientable or non-orientable) genus g. This result generalizes existing constant factor approximation results for planar graphs to a significantly larger graph family.

Our second main contribution is a $\mathcal{O}(\log^* n)$-time local and deterministic MDS approximation scheme which for any $\delta > 0$ can turn any constant factor approximation of an MDS into a $(1+\delta)$-approximate MDS on any class of graphs of sub-logarithmic expansion. Our methods are based on earlier work of Czygrinow et al. [6]. In combination with our constant-factor approximation on locally embeddable graphs, we obtain $(1+\delta)$-approximations in $\mathcal{O}_\delta(\log^* n)$ communication rounds on locally embeddable graphs, and in particular on bounded genus graphs. In combination with Lenzen and Wattenhofer's result [18] on graphs of bounded arboricity, we obtain $(1+\delta)$-approximations in $\mathcal{O}(\log n)$ randomized time on graphs of sub-logarithmic expansion, in particular, for all graphs which exclude a fixed minor. In combination with Amiri et al.'s result [1] on graphs of bounded expansion, we obtain $(1+\delta)$-approximations in $\mathcal{O}(\log n)$ deterministic time on graphs of sub-logarithmic expansion.

We observe that the methods of Czygrinow et al. [6] for maximum weighted independent set and maximum matching extend to graphs of sub-logarithmic expansion, however, we focus on the dominating set problem for the sake of a consistent presentation.

1.2 Novelty

Our main technical contribution is a new analysis of a sligthly modified variant of the elegant algorithm by Lenzen et al. [16] for planar graphs. As we will show, with a slight modification, the algorithm also works on locally embeddable graphs, however, the analysis needs to be changed significantly. Prior works by Lenzen et al. [16] and Wawrzyniak [24] heavily depend on topological properties of planar graphs. For example, their analyses exploit the fact that each cycle in a planar graph defines an "inside" and an "outside" region, without any edges connecting the two; this facilitates a simplified accounting and comparison to the optimal solution. In the case of locally embeddable graphs, such global, topological properties do not exist. In contrast, in this paper we leverage the inherent local properties of our low-density graphs, which opens a new door to approach the problem.

A second interesting technique developed in this paper is based on *preprocessing*: we show that the constants involved in the approximation can be further improved by a local preprocessing step.

Another interesting feature of our modified algorithm is that it is *first-order definable*. More precisely, there is a first order formula $\varphi(x)$ with one free variable, such that in every planar graph G the set $D = \{v \in V(G) : G \models \varphi(v)\}$ corresponds exactly to the computed dominating set. In particular, the algorithm

can be modified such that it does not rely on any *maximum* operations, such as finding the neighbor of maximal degree. The advent of sub-microprocessor devices, such as biological cellular networks or networks of nano-devices, has recently motivated the design of very simple, "stone-age" distributed algorithms [7], and we believe that our work nicely complements the finite-state machine model assumed in related work, and opens an interesting field for future research.

1.3 Organization

The remainder of this paper is organized as follows. We introduce some preliminaries in Section 2. The constant-factor constant-time approximation result is presented in Section 3, and the $\mathcal{O}(\log^* n)$-time approximation scheme is presented in Section 4. We conclude in Section 5.

2 Preliminaries

Graphs. We consider finite, undirected, simple graphs. Given a graph G, we write $V(G)$ for its vertices and $E(G)$ for its edges. Two vertices $u, v \in V(G)$ are adjacent or neighbors in G if $\{u, v\} \in E(G)$. The degree $d_G(v)$ of a vertex $v \in V(G)$ is its number of neighbors in G. We write $N(v)$ for the set of neighbors and $N[v]$ for the closed neighborhood $N(v) \cup \{v\}$ of v. We let $N^1[v] := N[v]$ and $N^{i+1}[v] := N[N^i[v]]$ for $i > 1$. A graph G has radius r if there is a *central vertex* $v \in V(G)$ such that $N^r[v] = V(G)$. If $E' \subseteq E$, we write $N_{E'}(v)$ for the set $\{u \in V(G) : \{u, v\} \in E'\}$. For $A \subseteq V(G)$, we write $N[A]$ for $\bigcup_{v \in A} N[v]$. The *arboricity* of G is the minimum number of forests into which its edges can be partitioned. The edge density of G is $\epsilon(G) = |E(G)|/|V(G)|$. It is well known that the arboricity of a graph is within factor 2 of its degeneracy, that is, $\max_{H \subseteq G} |E(H)|/|V(H)|$. For $A \subseteq V(G)$, the graph $G[A]$ induced by A is the graph with vertex set A and edge set $\{\{u, v\} \in E(G) : u, v \in A\}$. For $B \subseteq V(G)$ we write $G - B$ for the graph $G[V(G) \setminus B]$. A graph H is a subgraph of a graph G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$.

Bounded depth minors and local embeddable graphs. A graph H is a minor of a graph G, written $H \preceq G$, if there is a set $\{G_v : v \in V(H)\}$ of pairwise disjoint connected subgraphs $G_v \subseteq G$ such that if $\{u, v\} \in E(H)$, then there is an edge between a vertex of G_u and a vertex of G_v. We say that G_v is contracted to the vertex v. If $G_1, \ldots, G_k \subseteq V(G)$ are pairwise disjoint connected subgraphs of G, then we write $G/G_1/\ldots/G_k$ for the minor obtained by contracting the subgraphs G_i. Note that it does not matter in what order we contract the subgraphs.

A star is a connected graph G such that at most one vertex of G, called the center of the star, has degree greater than one. H is a *depth-1 minor* of G if H is obtained from a subgraph of G by star contractions, that is, if there is a set $\{G_v : v \in V(H)\}$ of pairwise disjoint stars $G_v \subseteq G$ such that if $\{u, v\} \in E(H)$, then there is an edge between a vertex of G_u and a vertex of G_v.

More generally, H is a *depth-r minor* of G, written $H \preceq_r G$, if there is a set $\{G_v : v \in V(H)\}$ of pairwise disjoint connected subgraphs $G_v \subseteq G$ of radius at most r such that if $\{u, v\} \in E(H)$, then there is an edge between a vertex of G_u and a vertex of G_v.

We write $K_{t,3}$ for the complete bipartite graph with partitions of size t and 3, respectively. A graph G is a *locally embeddable graph* if it excludes $K_{3,t}$ as a depth-1 minor for some $t \geq 3$ and if $|E(H)|/|V(H)| \leq c$ for some constant c and all depth-1 minors H of G.

More generally, we write $\nabla_r(G)$ for $\max_{H \preceq_r G} |E(H)|/|V(H)|$. A class $\mathcal{C}$ of graphs has *bounded expansion* if there is a function $f : \mathbb{N} \to \mathbb{N}$ such that $\nabla_r(G) \leq f(r)$ for all graphs $G \in \mathcal{C}$. This is equivalent to demanding that the arboricity of each depth-r minor of G is functionally bounded by r. The class $\mathcal{C}$ has *sub-logarithmic expansion* if the bounding function $f(r) \in o(\log r)$. Note that if every graph $G \in \mathcal{C}$ excludes a fixed minor, then $\mathcal{C}$ has constant expansion, hence classes of sub-logarithmic expansion generalize proper minor closed classes of graphs.

Dominating sets. Let G be a graph. A set $M \subseteq V(G)$ *dominates* G if all vertices of G lie either in M or are adjacent to a vertex of D, that is, if $N[M] = V(G)$. A minimum dominating set M is a dominating set of minimum cardinality (among all dominating sets). The size of a minimum dominating set of G is denoted $\gamma(G)$.

f-Approximation. Let $f : \mathbb{N} \to \mathbb{R}^+$. Given an n-vertex graph G and a set D of G, we say that D is an f-approximation for the dominating set problem if D is a dominating set of G and $|D| \leq f(n) \cdot \gamma(G)$. An algorithm computes an f-approximation for the dominating set problem on a class $\mathcal{C}$ of graphs if for all $G \in \mathcal{C}$ it computes a set D which is an f-approximation for the dominating set problem. If f maps every number to a fixed constant c, we speak of a constant factor approximation.

Bounded genus graphs. The (orientable, resp. non-orientable) genus of a graph is the minimal number ℓ such that the graph can be embedded on an (orientable, resp. non-orientable) surface of genus ℓ. We write $g(G)$ for the orientable genus of G and $\tilde{g}(G)$ for the non-orientable genus of G. Every connected planar graph has orientable genus 0 and non-orientable genus 1. In general, for connected G, we have $\tilde{g}(G) \leq 2g(G) + 1$. On the other hand, there is no bound for $g(G)$ in terms of $\tilde{g}(G)$. As all our results apply to both variants, for ease of presentation, and as usual in the literature, we will simply speak of the genus of a graph in the following. We do not make explicit use of any topological arguments and hence refer to [20] for more background on graphs on surfaces. We will use the following facts about bounded genus graphs.

Graphs of genus g are closed under taking subgraphs and edge contraction.

Lemma 1. *If $H \preceq G$, then $g(H) \leq g(G)$ and $\tilde{g}(H) \leq \tilde{g}(G)$.*

One of the arguments we will use is based on the fact that bounded genus graphs exclude large bipartite graphs as minors. The lemma follows immediately

from Lemma 1 and from the fact that $g(K_{m,n}) = \left\lceil \frac{(m-2)(n-2)}{4} \right\rceil$ and $\tilde{g}(K_{m,n}) = \left\lceil \frac{(m-2)(n-2)}{2} \right\rceil$ (see e.g. Theorem 4.4.7 in [20]).

Lemma 2. *If $g(G) = g$, then G excludes $K_{4g+3,3}$ as a minor and if $\tilde{g}(G) = \tilde{g}$, then G excludes $K_{2\tilde{g}+3,3}$ as a minor.*

Graphs of bounded genus do not contain many disjoint copies of minor models of $K_{3,3}$: this is a simple consequence of the fact that the orientable genus of a connected graph is equal to the sum of the genera of its blocks (maximal connected subgraphs without a cut-vertex) and a similar statement holds for the non-orientable genus, see Theorem 4.4.2 and Theorem 4.4.3 in [20].

Lemma 3. *A graph G contains at most $\max\{g(G), 2\tilde{g}(G)\}$ disjoint copies of minor models of $K_{3,3}$.*

Finally, note that graphs of bounded genus have small edge density.

Lemma 4. *For every graph G it holds that $|E(G)| \leq 3 \cdot |V(G)| + 6g(G) - 6$ and $|E(G)| \leq 3 \cdot |V(G)| + 3\tilde{g}(G) - 3$.*

Lemma 5. *Let $\mathcal{G}$ be a class of graphs of genus at most g. Then the degeneracy and edge density of every graph $G \in \mathcal{G}$ is in $\mathcal{O}(\sqrt{g})$.*

Proof. We prove the lemma for degeneracy of G, the claim for edge density follows. We prove the lemma for graphs with orientable genus g; an analogous argument works for graphs of non-orientable genus g. Let $G \in \mathcal{G}$ with genus at most g, and suppose the degeneracy of G is c. We prove that $c \in O(\sqrt{g})$. Let us denote by v, e the number of vertices and edges of G, respectively. By Lemma 4 we have $e \leq 3 \cdot v + 6g - 6$. On the other hand, by definition of the degeneracy, every vertex in G has degree at least c, so $\frac{c \cdot v}{2} \leq 3v + 6g - 6 \Rightarrow c \leq \frac{12g-12}{v} + 6$ (1). To find the maximum value of c for a fixed genus, we must minimise v. A complete graph on v vertices has genus at most $v^2/12$ [20], therefore by plugging it into (1), we obtain that $c \leq \sqrt{12g} + 6$.

Distributed complexity. We consider the standard *LOCAL* model of distributed computing [19], see also [22] for a recent survey. A distributed system is modeled as a graph G. At each vertex $v \in V(G)$ there is an independent agent/host/processor with a unique identifier $id(v)$. Initially, each agent has no knowledge about the network, but only knows its own identifier. Information about other agents can be obtained through message passing, i.e., through repeated interactions with neighboring vertices, which happens in synchronous communication rounds. In each round the following operations are performed:

(1) Each vertex performs a local computation (based on information obtained in previous rounds).
(2) Each vertex v sends one message to each of its neighbors.
(3) Each vertex v receives one message from each of its neighbors.

The *distributed complexity* of the algorithm is defined as the number of communication rounds until all agents terminate. We call a distributed algorithm r-local, if its output depends only on the r-neighborhoods $N^r[v]$ of its vertices.

Algorithm 1 Dominating Set Approximation Algorithm for Planar Graphs

1: Input: Planar graph G
2: $(*$ *Phase 1* $*)$
3: $D \leftarrow \emptyset$
4: **for** $v \in V$ (in parallel) **do**
5: **if** there does not exist a set $A \subseteq V(G) \setminus \{v\}$ such that $N(v) \subseteq N[A]$ and $|A| \leq 6$ **then**
6: $D \leftarrow D \cup \{v\}$
7: **end if**
8: **end for**
9: $(*$ *Phase 2* $*)$
10: $D' \leftarrow \emptyset$
11: **for** $v \in V$ (in parallel) **do**
12: $d_{G-D}(v) \leftarrow |N[v] \setminus N[D]|$
13: **if** $v \in V \setminus N[D]$ **then**
14: $\Delta_{G-D}(v) \leftarrow \max_{w \in N[v]} d_{G-D}(w)$
15: choose any $dom(v)$ from $N[v]$ with $d_{G-D}(dom(v)) = \Delta_{G-D}(v)$
16: $D' \leftarrow D' \cup \{dom(v)\}$
17: **end if**
18: **end if**
19: **end for**
20: **return** $D \cup D'$

3 A Constant Local MDS Approximation

Let us start by revisiting the MDS approximation algorithm for planar graphs by Lenzen et al. [16], see Algorithm 3. The algorithm works in two phases. In the first phase, it adds all vertices whose (open) neighborhood cannot be dominated by a small number of vertices to a set D. It has been shown in [16] that the set D is small in planar graphs. In the second phase, it defines a dominator function dom which maps every vertex v that is not dominated yet by D to its dominator. The dominator $dom(v)$ of v is chosen arbitrary among those vertices of $N[v]$ which dominate the maximal number of vertices not dominated yet.

We now propose the following small change to the algorithm. As additional input, we require an integer c which bounds the edge density of depth-1 minors of G and we replace the condition $|A| \leq 6$ in Line 5 by the condition $|A| \leq 2c$. In the rest of this section, we show that the modified algorithm computes a constant factor approximation on any locally embeddable class of graphs. Note that the algorithm does not have to compute the edge density of G, which is not possible in a local manner. Rather, we leverage Lemma 4 which upper bounds $\epsilon(G)$ for any fixed class of bounded genus graphs: this upper bound can be used as an input to the local algorithm.

We first show that the set D computed in Phase 1 of the algorithm is small. The following lemma is a straight-forward generalization of Lemma 6.3 of [16], which in fact uses no topological arguments at all.

Lemma 6. *Let G be a graph and let M be a minimum dominating set of G. Assume that for some constant c all depth-1 minors H of G satisfy $|E(H)|/|V(H)| \leq c$. Let*

$$D := \{v \in V(G) : \text{ there is no set } A \subseteq V(G) \setminus \{v\} \text{ such that } N(v) \subseteq N[A] \text{ and } |A| \leq 2c\}.$$

Then $|D| \leq (c+1) \cdot |M|$.

Proof. Let H be the graph with $V(H) = M \cup N[D \setminus M]$ and where $E(H)$ is a minimal subset of $E(G[V(H)])$ such that all edges with at least one endpoint in $D \setminus M$ are contained in $E(H)$ and such that M is a dominating set in H. By this minimality condition, every vertex $v \in V(H) \setminus (D \cup M)$ has exactly one neighbor $m \in M$, no two vertices of $V(H) \setminus (M \cup D)$ are adjacent, and no two vertices of M are adjacent.

We construct a depth-1 minor $\tilde{H}$ of H by contracting the star subgraphs G_m induced by $N_H[m] \setminus D$ for $m \in M \setminus D$ to a single vertex v_m. Note that all vertices of $N[D \setminus M] \setminus D$ disappear into some star G_m, hence $\tilde{H}$ has exactly $|D \setminus M| + |M|$ vertices. Let $w \in D \setminus M$. As $N_G(w)$ cannot be covered by less than $(2c+1)$ elements from $V(G) \setminus \{w\}$ (by definition of D), w also has at least $(2c+1)$ neighbors in $\tilde{H}$. On the other hand, $\tilde{H}$ has at most $c \cdot |V(\tilde{H})|$ edges, and also the subgraph $\tilde{H}[D \setminus M]$ has at most $c \cdot |D \setminus M|$ edges. Hence

$$\begin{aligned}
&(2c+1) \cdot |D \setminus M| - c \cdot |D \setminus M| \\
&\leq \sum_{w \in D \setminus M} d_{\tilde{H}}(w) - |E(\tilde{H}[D \setminus M])| \\
&\leq |E(\tilde{H})| \\
&\leq c \cdot |V(\tilde{H})| \\
&= c \cdot (|D \setminus M| + |M|),
\end{aligned}$$

and hence $|D \setminus M| \leq c \cdot |M|$, which implies the claim. □

Assumption 1 *For the rest of this section, we assume that G satisfies that all depth-1 minors H of G satisfy $|E(H)|/|V(H)| \leq c$ for some constant c, and we fix M and D as in Lemma 6. Note that in particular, in the class of graphs of genus at most g, c belongs to $\mathcal{O}(\sqrt{g})$. We furthermore assume that for some $t \geq 3$, G excludes $K_{t,3}$ as depth-1 minor.*

Let us write R for the set $V(G) \setminus N[D]$ of vertices which are not dominated by D. The algorithm defines a dominator function $dom : R \to N[R] \subseteq V(G) \setminus D$. The set D' computed by the algorithm is the image $dom(R)$, which is a dominating set of vertices in R. As R contains the vertices which are not dominated by D, $D' \cup D$ is a dominating set of G. This simple observation proves that the algorithm correctly computes a dominating set of G. Our aim is to find a bound on $|dom(R)|$.

We fix an ordering of M as $m_1, \ldots, m_{|M|}$ such that the vertices of $M \cap D$ are first (minimal) in the ordering and inductively define a minimal set $E' \subseteq E(G)$ such that M is a dominating set with respect to E'. For this, we add all edges $\{m_1, v\} \in E(G)$ with $v \in N(m_1) \setminus M$ to E'. We then continue inductively by adding for $i > 1$ all edges $\{m_i, v\} \in E(G)$ with $v \in N(m_i) \setminus (M \cup N_{E'}(\{m_1, \ldots, m_{i-1}\}))$.

For $m \in M$, let G_m be the star subgraph of G with center m and all vertices v with $\{m, v\} \in E'$. Let H be the depth-1 minor of G which is obtained by contracting all stars G_m for $m \in M$. This construction is visualized in Figure 1. In the figure, solid lines represent edges from E', lines from $E(G) \setminus E'$ are dashed. We want to count the *endpoints* of directed edges, which represent the dominator function *dom*.

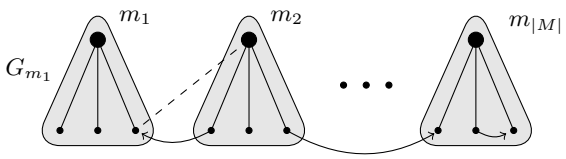

Fig. 1. The graphs G_m. Solid lines represent edges from E', directed edges represent the dominator function *dom*.

In the following, we call a directed edge which represents the function *dom* a *dom-edge*. We did not draw *dom*-edges that either start or end in M. When counting $|dom(R)|$, we may simply add a term $2|M|$ to estimate the number of endpoints of those edges. We also did not draw a *dom*-edge starting in G_{m_1}. In the figure, we assume that the vertex m_1 belongs to $M \cap D$. Hence every vertex v from $N[m_1]$ is dominated by a vertex from D and the function is thus not defined on such v. However, the vertices of $N(m_1)$ may still serve as dominators, as shown in the figure.

H has $|M|$ vertices and by our assumption on the density of depth-1 minors of G, it has at most $c|M|$ edges.

Our analysis proceeds as follows. We distinguish between two types of *dom*-edges, namely those which go from one star to another star and those which start and end in the same star. By the star contraction, all edges which go from one star to another star are represented by a single edge in H. We show in Lemma 7 that each edge in H does not represent many such *dom*-edges with distinct endpoints. As H has at most $c|M|$ edges, we will end up with a number of such edges that is linear in $|M|$. On the other hand, all edges which start and end in the same star completely disappear in H. In Lemma 10 we show that these star contractions "absorb" only few such edges with distinct endpoints.

We first show that an edge in H represents only few *dom*-edges with distinct endpoints. For each $m \in M \setminus D$, we fix a set $C_m \subseteq V(G) \setminus \{m\}$ of size at most $2c$

which dominates $N_{E'}(m)$, note that existence of C_m follows from the definition of the set D. Recall that we assume that G excludes $K_{t,3}$ as depth-1 minor.

Lemma 7. *Let* $1 \leq i < j \leq |M|$. *Let* $N_i := N_{E'}(m_i)$ *and* $N_j := N_{E'}(m_j)$.

1. *If* $m_j \in M \setminus D$, *then*

$$|\{u \in N_j : \text{ there is } v \in N_i \text{ with } \{u, v\} \in E(G)\}| \leq 2ct.$$

2. *If* $m_i \in M \setminus D$ *(and hence* $m_j \in M \setminus D$*), then*

$$|\{u \in N_i : \text{ there is } v \in N_j \text{ with } \{u, v\} \in E(G)\}| \leq 4ct.$$

Proof. By definition of E', it holds that $m_i \notin C_{m_j}$. Let $c \in C_{m_j}$ be arbitrary. Then there are at most $t-1$ distinct vertices $u_1, \ldots, u_{t-1} \in (N_j \cap N(c))$ such that there are $v_1, \ldots, v_{t-1} \in N_i$ (possibly not distinct) with $\{u_k, v_k\} \in E(G)$ for all k, $1 \leq k \leq t-1$. Otherwise, we can contract the star with center m_i and branch vertices $N(m_i) \setminus \{c\}$ and thereby find $K_{t,3}$ as depth-1 minor, a contradiction. See Figure 2 for an illustration in the case of an excluded $K_{3,3}$. Possibly, $c \in N_j$ and it is connected to a vertex of N_i, hence we have at most t vertices in $N_j \cap N[c]$ with a connection to N_i. As $|C_{m_j}| \leq 2c$, we conclude the first item.

Regarding the second item, let $c \in C_{m_i}$ be arbitrary. If $c \neq m_j$, we conclude just as above, that there are at most $t-1$ distinct vertices $u_1, \ldots, u_{t-1} \in (N_i \cap N(c))$ such that there are $v_1, \ldots, v_{t-1} \in N_j$ (possibly not distinct) with $\{u_k, v_k\} \in E(G)$ for all k, $1 \leq k \leq t-1$ and hence at most t vertices in $N_i \cap N[c]$ with a connection to N_j. Now assume $c = m_j$. Let $c' \in C_{m_j}$. There are at most $t-1$ distinct vertices $u_1, \ldots, u_{t-1} \in (N_i \cap N_E(m_j))$ such that there are vertices $v_1, \ldots, v_{t-1} \in N_j \cap N(c)$ (possibly not distinct) with $\{u_k, v_k\} \in E(G)$ for all k, $1 \leq k \leq t-1$. Again, considering the possibility that $c' \in N_i$, there are at most t vertices in $N_i \cap N_E(m_j)$ with a connection to $N_j \cap N(c)$. As $|C_{m_j}| \leq 2c$, we conclude that in total there are at most $2ct$ vertices in $N_i \cap N_E(m_j)$ with a connection to N_j. In total, there are hence at most $(2c-1)t + 2ct \leq 4ct$ vertices of the described form.

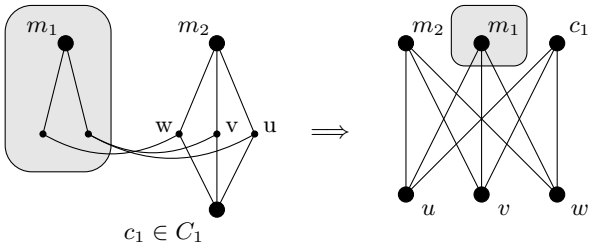

Fig. 2. Visualisation of the proof of Lemma 7 in the case of excluded $K_{3,3}$

We write Y for the set of all vertices $\{u \in N_{E'}(m_i) : m_i \notin D$ and there is $v \in N_{E'}(m_j)$, $j \neq i$ and $\{u, v\} \in E(G)\}$.

Corollary 1. $|Y| \leq 6c^2t|M|$.

Proof. Each of the $c|M|$ many edges in H represents edges between N_i and N_j, where N_i and N_j are defined as above. By the previous lemma, if $i < j$, there are at most $2ct$ vertices in $N_i \cap Y$ and at most $4ct$ vertices in $N_j \cap Y$, hence in total, each edge accounts for at most $6ct$ vertices in Y.

We continue to count the edges which are inside the stars. First, we show that every vertex has small degree inside its own star.

Lemma 8. *Let $m \in M \setminus D$ and let $v \in N_{E'}(m) \setminus C_m$. Then*

$$|\{u \in N_{E'}(m) : \{u, v\} \in E(G)\}| \leq 2c(t-1).$$

Proof. Let $c \in C_m$. By the same argument as in Lemma 7, there are at most $t-1$ distinct vertices $u_1, \ldots, u_{t-1} \in (N_{E'}(m) \cap N(c))$ such that $\{u_k, v\} \in E(G)$ for all k, $1 \leq k \leq t-1$.

Let $C := \bigcup_{m \in M \setminus D} C_m$. We show that there are only few vertices which are highly connected to $M \cup C$. Let $Z := \{u \in N_{E'}(M \setminus D) : |N(u) \cap (M \cup C)| > 4c\}$.

Lemma 9.

$$|Z| < |M \cup C|.$$

Proof. Assume that $|Z| > |M \cup C|$. Then the subgraph induced by $Z \cup M \cup C$ has more than $\frac{1}{2}4c|Z|$ edges and $|Z \cup M \cup C|$ vertices. Hence its edge density is larger than $2c|Z|/(|Z \cup M \cup C|) > 2c|Z|/(2|Z|) = c$, contradicting our assumption on the edge density of depth-1 minors of G (which includes its subgraphs).

Finally, we consider the image of the *dom*-function inside the stars.

Lemma 10.

$$\left| \bigcup_{m \in M \setminus D} \{u \in N_{E'}(m) : dom(u) \in (N_{E'}(m) \setminus (Y \cup Z))\} \right| \leq (2(t-1)+4)c|M|.$$

Proof. Fix some $m \in M \setminus D$ and some $u \in N_{E'}(m)$ with $dom(u) \in N_{E'}(m) \setminus (Y \cup Z)$. Because $dom(u) \notin Y$, $dom(u)$ is not connected to a vertex of a different star, except possibly for vertices from M. Because $dom(u) \notin Z$, it is however connected to at most $4c$ vertices from $M \cup C$. Hence it is connected to at most $4c$ vertices from different stars. According to Lemma 8, $dom(u)$ is connected to at most $2c(t-1)$ vertices from the same star. Hence the degree of $dom(u)$ is at most $4c + 2c(t-1)$. Because u preferred to choose $dom(u) \in N_{E'}(m)$ over m as its dominator, we conclude that m has at most $4c + 2c(t-1)$ E'-neighbors. Hence, in total there can be at most $(2(t-1)+4)c|M|$ such vertices. □

We are now ready to put together the numbers.

Lemma 11. *If all depth-1 minors H of G have edge density at most c and G excludes $K_{t,3}$ as depth-1 minor, then the modified algorithm computes a $6c^2t + (2t+5)c + 4$ approximation for the minimum dominating set problem on G.*

Proof. The set D has size at most $(c+1)|M|$ according to Lemma 6. Since M is a dominating set also with respect to the edges E', it suffices to determine $|\{dom(u) : u \in (N_{E'}[M \setminus D] \setminus N[D])\}|$. According to Corollary 1, the set $Y = \{u \in N_{E'}(m_i) :$ there is $v \in N_{E'}(m_j)$, $i \neq j$ and $\{u,v\} \in E(G)\}$ has size at most $6c^2t|M|$. In particular, there are at most so many vertices $dom(u) \in N_{E'}(m_i)$ with $u \in N_{E'}(m_j)$ for $i \neq j$. Clearly, $|dom(R) \cap M| \leq |M|$ and $|dom(M)| \leq |M|$. Together, this bounds the number of endpoints of dom-edges that go from one star to another star. According to Lemma 9, there are only few vertices which are highly connected to $M \cup C$, that is, the set $Z = \{u \in N_{E'}(M \setminus D) : |N(u) \cap (M \cup C)| > 4c\}$ satisfies $|Z| < |M \cup C|$. We have $|C| \leq 2c|M|$, as each C_m has size at most $2c$. It remains to count the image of dom inside the stars which do not point to Y or Z. According to Lemma 10, this image has size at most $(2(t-1)+4)c|M|$. In total, we hence find a set of size

$$(c+1)|M| + 6c^2t|M| + 2|M| + (2c+1)|M| + (2(t-1)+4)c|M| \leq (6c^2t + (2t+5)c + 4)|M|.$$

Our theorem for bounded genus graphs is now a corollary of Lemmas 2, 5 and 11.

Theorem 2. *Let $\mathcal{C}$ be a class of graphs of orientable genus at most g (non-orientable genus at most $\tilde{g}$ resp.). The modified algorithm computes an $\mathcal{O}(g\sqrt{g})$-approximation ($\mathcal{O}(\tilde{g})$-approximation resp.) for the dominating set in $\mathcal{O}(g)$ ($\mathcal{O}(\tilde{g})$ resp.) communication rounds.*

For the special case of planar graphs, our analysis shows that the algorithm computes a 199-approximation. This is not much worse than Lenzen et al.'s original analysis (130), however, off by a factor of almost 4 from Wawrzyniak's [24] improved analysis (52).

3.1 Improving the Approximation Factor with Preprocessing

We now show the approximation factors related to the genus g, derived in the previous section, can be improved to linear time, using a local preprocessing step.

Given a graph G and a vertex $v \in V(G)$. Let $K = \{K_1, \ldots, K_j\}$ denote the set of minimal subgraphs of G containing v such that for all $1 \leq i \leq j$, $K_{3,3}$ is a depth-1 minor of K_i. Let $K_h \in K$ be the one with lexicographically smallest identifiers in K, we say K_h is the v-canonical subgraph of G and we denote it by K_v. If $K = \emptyset$ we set $K_v := \emptyset$.

Algorithm 2 Dominating Set Approximation for Graphs of Genus $\leq g$

Input: Graph G of genus at most g
Run Phase 1 of Algorithm 3
($*$ *Preprocessing Phase* $*$)
for $v \in V - D$ (in parallel) **do**
    compute K_v **in** $G - D$ **(see Lemma 12)**
for $i = 1..g$ **do**
  for $v \in V - D$ (in parallel) **do**
    if $K_v \neq \emptyset$ **then**
      chosen : $=$ **true**
      for all $u \in N^{12}(v)$
        if $K_u \cap K_v \neq \emptyset$ **and** $u < v$ **then chosen :=** **false**
      if (chosen $=$ **true) then** $D := D \cup V(K_v)$
Run Phase 2 of Algorithm 3

Lemma 12. *Given a graph G and a vertex $v \in V(G)$. The v-canonical subgraph K_v of v can be computed locally in at most 6 communication rounds. Furthermore, K_v has at most 24 vertices.*

Proof. The proof is constructive. As $K_{3,3}$ has diameter 2, every minimal subgraph of G containing $K_{3,3}$ as a depth-1 minor has diameter at most 6 (every edge may have to be replaced by a path of length 3). Therefore, it suffices to consider the subgraph $H = G[N^6(v)]$ and find the lexicographically minimal subgraph which contains $K_{3,3}$ as depth-1 minor in H which includes v as a vertex. If this is the case, we output it as K_v; otherwise we output the empty set. Furthermore, $K_{3,3}$ has 9 edges and hence a minimal subgraph containing it as depth-1 minor has at most 24 vertices (again, every edge is subdivided at most twice and $2 \cdot 9 + 6 = 24$).

To improve the approximation factor, we propose the following modified algorithm, see Algorithm 2.

Theorem 3. *Algorithm 2 provides a $24g + \mathcal{O}(1)$ MDS approximation for graphs of genus at most g, and requires $12g + \mathcal{O}(1)$ communication rounds.*

Proof. The resulting vertex set is clearly a legal dominating set. Moreover, as Phase 1 is unchanged, we do not have to recalculate the size of D.

In the preprocessing phase, if for two vertices $u \neq v$ we choose both K_u, K_v, then they must be disjoint: Since the diameter of any depth-1 minor of $K_{3,3}$ is at most 6, if two such canonical subgraphs intersect, the distance between u, v can be at most 12. Hence, each vertex v can decide in its 12-neighbourhood whether its K_v is the smallest among all choices. On the other hand, by Lemma 3, there are at most g disjoint such models. So in the *preprocessing* phase, we can remove at most g disjoint subgraphs K_v (and add their vertices to the dominating set) and thereby select at most $24g$ extra vertices for the dominating set. Once the *preprocessing* phase is finished, the remaining graph is locally embeddable.

In order to compute the size of the set in the third phase, we can use the analysis of Lemma 11 for $t = 3$, which together with the first phase and preprocessing phase, results in a $24g + \mathcal{O}(1)$-approximation guarantee.

To count the number of communication rounds, note that the only change happens in the second phase. In that phase, in each iteration, we need 12 communication rounds to compute the 12-neighbourhood. Therefore, the number of communication rounds is $12g + \mathcal{O}(1)$. □

This significantly improves the approximation upper bound of Theorem 2: namely from $4(6c^2 + 2c)g + \mathcal{O}(1)$ (which, using $c \leq 3.01$ for sufficiently large graphs, can be as high as $241g + \mathcal{O}(1)$), to $24g + \mathcal{O}(1)$, at the price of $12g$ extra communication rounds.

3.2 A Logical Perspective

Interestingly, as we will elaborate in the following, a small modification of Algorithm 3 can be interpreted both from a distributed computing perspective, namely as a local algorithm of constant distributed time complexity, as well as from a logical perspective.

First order logic has atomic formulas of the form $x = y, x < y$ and $E(x, y)$, where x and y are first-order variables. The set of first order formulas is closed under Boolean combinations and existential and universal quantification over the vertices of a graph. To define the semantics, we inductively define a satisfaction relation $\models$, where for a graph G, a formula $\phi(x_1, \ldots, x_k)$ and vertices $v_1, \ldots, v_k \in V(G)$, $G \models \phi(v_1, \ldots, v_k)$ means that G satisfies ϕ if the free variables $x_1, \ldots, x_k$ are interpreted as $v_1, \ldots, v_k$, respectively. The free variables of a formula are those not in the scope of a quantifier, and we write $\phi(x, \ldots, x_k)$ to indicate that the free variables of the formula ϕ are among $x_1, \ldots, x_k$. For $\phi(x_1, x_2) = x_1 < x_2$, we have $G \models \phi(v_1, v_2)$ if $v_1 < v_2$ with respect to the ordering $<$ of $V(G)$ and for $\phi(x_1, x_2) = E(x_1, x_2)$ we have $G \models \phi(v_1, v_2)$ if $\{v_1, v_2\} \in E(G)$. The meaning of the equality symbol, the Boolean connectives, and the quantifiers is as expected.

A first-order formula $\phi(x)$ with one free variable naturally defines the set $\phi(G) = \{v \in V(G) : G \models \phi(v)\}$. We say that a formula ϕ defines an f-approximation to the dominating set problem on a class $\mathcal{C}$ of graphs, if $\phi(G)$ is an f-approximation of a minimum dominating set for every graph $G \in \mathcal{C}$.

Observe that first-order logic is not able to count, in particular, no fixed formula can determine a neighbor of maximum degree in Line 14 of the algorithm. Observe however, that the only place in our analysis that refers to the dominator function *dom* explicitly is Lemma 10. The proof of the lemma in fact shows that we do not have to choose a vertex of maximal residual degree, but that it suffices to choose a neighbour of degree greater than $4c+2c(t-1)$ if such a vertex exists, or any vertex, otherwise. For every fixed class of bounded genus, this number is a constant. We use the binary predicate $<$ to make a unique choice of a dominator in this case.

Then we define D by the following formula

$$\varphi_D(x) = \neg(\exists x_1 \dots \exists x_{2c} \forall y \left(E(x,y) \to \bigvee_{1 \le i \le 2c} E(y,x_i) \right)$$

and D' by

$$\psi_{D'}(x) = \exists y \Big(E(x,y) \wedge \forall z \big(\varphi_D(z) \to \neg E(y,z) \big) \wedge \xi_{\max}(x,y) \Big),$$

where $\xi_{\max}(x,y)$ states that x is the maximum (residual) degree neighbour of y up to threshold $4c + 2c(t-1)$. We can express this cumbersome formula with $4c + 2c(t-1)$ quantifiers. Note that the formulas φ_D and $\psi_{D'}$ are different in spirit. While φ_D directly describes a property of vertices which causes them to be included in the dominating set, in the formula $\psi_{D'}(x)$ we state the existence of an element which is not yet dominated by D and which elects x as a dominator.

4 $(1+\delta)$-Approximations

In this section we show how to extend techniques developed by Czygrinow et al. [6] for planar graphs to graphs of sub-logarithmic expansion. These graphs are very general classes of sparse graphs, including planar graphs and all classes that exclude a fixed minor. We focus on the dominating set problem, however, the approximations for the maximum weight independent set problem and maximum matching problem proposed by Czygrinow et al. can be extended in a similar way.

Our notation in this section closely follows that of Czygrinow et al. [6]. In particular, we will work with vertex and edge weighted graphs, that is, every graph G is additionally equipped with two weight functions $\omega : V(G) \to \mathbb{R}^+$ and $\bar{\omega} : E(G) \to \mathbb{R}^+$. If $H \subseteq G$ is a subgraph of G, then we write $\omega(H)$ for $\sum_{v \in V(H)} \omega(v)$ and $\bar{\omega}(H)$ for $\sum_{e \in E(H)} \bar{\omega}(e)$. If H is a minor of G then it naturally inherits a weight function ω_H from G as follows. If $u, v \in V(H)$ with corresponding branch sets $H_u, H_v \subseteq V(G)$ in G, then $\omega(u) = \sum_{w \in V(H_v)} \omega(w)$ and if $\{u,v\} \in E(H)$, then $\bar{\omega}_H(\{u,v\}) = \sum_{e \in E(G), e \cap V(H_u) \neq \emptyset, e \cap V(H_v) \neq \emptyset} \bar{\omega}(e)$.

4.1 Clustering algorithm

We first generalise the partitioning algorithm provided by Czygrinow et al. to graphs with logarithmic expansion.

Definition 1 (Pseudo-Forest). *A* pseudo-forest *is a directed graph* $\boldsymbol{F}$ *in which every vertex has out-degree at most* 1.

For a directed graph $\boldsymbol{F}$, we write F for the underlying undirected graph of $\boldsymbol{F}$. The following lemma is a straight forward generalization of Fact 1 of [6].

Lemma 13. *Let G be a graph of arboricity a with an edge-weight function $\bar{\omega}$. There is a distributed procedure which in two rounds finds a pseudo-forest $\boldsymbol{F}$ such that F is a subgraph of G with $\bar{\omega}(F) \geq \bar{\omega}(G)/(2a)$.*

It is now straight forward to generalise also Lemma 2 of [6].

Lemma 14 (HeavyStar). *There is a local algorithm which takes an edge weighted n-vertex graph G of arboricity at most a as input and in $\mathcal{O}(\log^* n)$ rounds computes a partition of $V(G)$ into vertex disjoint stars $H_1, \ldots, H_x \subseteq V(G)$ such that $H = G/H_1/\ldots H_x$ has total weight $\bar{\omega}_H(H) \leq (1-1/(8a))\cdot\bar{\omega}(G)$.*

We come to the final clustering algorithm. We fix a function $f(r) \in o(\log r)$ which bounds the expansion of the input graphs G. In particular recall that then the depth-r minors of G have arboricity bounded by $2f(r)$. By iteratively contracting star forests, we obtain minors at exponential depth, as described by the next lemma.

Lemma 15 (Proposition 4.1, statement (4.4) of [21]). *Taking a depth-1 minor of a depth-1 minor for r times gives a depth-$(3^r-1)/2$ minor of G.*

In particular, when iterating the star contraction routine of Algorithm 3, in iteration i we are dealing with a subgraph of arboricity $2f(3^i-1)/2) =: g(i)$ which is sublinear in i. Hence, we may apply Lemma 14 in each iteration. Note that we do not require the arboricity as an input to the algorithm of Lemma 14. Note furthermore, that we have

$$\lim_{i\to\infty} \big(1-1/(8g(i))\big)^i = \lim_{i\to\infty} e^{-i/g(i)} = 0,$$

hence for every $\epsilon > 0$ there is a constant i_0 depending only on ϵ and g (and not on the graph size n) such that $\big(1-1/(8g(i_0))\big)^{i_0} \leq \epsilon$ (we may assume that the function g is monotone).

Algorithm 3 Clustering

1: Input: G with $\nabla_r(G) \leq f(r)$, $\epsilon > 0$ and i_0 with $\big(1-1/(8g(i_0))\big)^{i_0} \leq \epsilon$
2: **for** $i = 1, \ldots, i_0$
3: Call Lemma 14 to find vertex disjoint stars $H_1, \ldots, H_x$ in G
4: $G \leftarrow G/H_1/\ldots/H_x$ with weights modified accordingly
5: **for** $v \in V(G)$:
6: $W_v \leftarrow$ vertices contracted to v
7: **return** $\{W_v : v \in V(G)\}$.

Lemma 16 (Clustering). *Let $c \geq 1$ be a constant and let G be a graph with $\nabla_r(G) \leq f(r)$. If the clustering algorithm gets G and $\epsilon > 0$ as input, then it returns a set of clusters $C_1, \ldots, C_l$ partitioning G, such that, each cluster has*

radius at most $(3^{i_0}-1)/2$ *(where* i_0 *is the number of iterations in the algorithm). Moreover, if we contract each* C_i *to a single vertex to obtain a graph* H*, then* $\bar{\omega}_H(H) \leq \epsilon \cdot \bar{\omega}(G)$.

Proof. As described above, the graph G_i we are dealing with in iteration i has arboricity $2f(3^i-1)/2) = g(i)$ which is sublinear in i. By applying Lemma 14 to G_i, we compute in $\mathcal{O}(\log^* n)$ rounds a partition of $V(G_i)$ into vertex disjoint stars $H_1, \ldots, H_x \subseteq V(G_i)$ such that $H = G_i/H_1/\ldots H_x$ has total weight $\bar{\omega}_H(H) \leq (1-1/(8g(i)) \cdot \bar{\omega}(G_i)$. Note that by Lemma 15, the graph G_i obtained in iteration i is a depth-$((3^i-1)/2$ minor of G. Hence, by induction, after i iterations, the edge weight of the graph G_i is at most $\big(1-1/(8g(i))\big)^i$. As argued above, there exists i_0 such that $\big(1-1/(8g(i_0))\big)^{i_0} \leq \epsilon$, at which time we stop the algorithm.

As in each round we invest at most time $\mathcal{O}(\log^* n)$, in total we invest at most $\mathcal{O}(\log^* n \cdot i_0) = \mathcal{O}_\epsilon(\log^* n)$ time to compute the clustering. Here, we use the notation $\mathcal{O}_\epsilon$ to denote that we are treating all constants depending on ϵ only as constants.

4.2 Approximation for minimum dominating set

Theorem 4. *There exists a deterministic distributed algorithm which for given* $0 < \delta < 1$ *finds in an* n*-vertex graph* G *of sub-logarithmic expansion which is given together with a* c*-approximation of a minimum dominating set a dominating set* D *of size at most* $(1+\delta) \cdot \mathrm{Opt}$*, where* Opt *denotes the size of a minimum dominating set in* G*. The algorithm runs in* $\mathcal{O}_\delta(\log^* n)$ *rounds.*

Corollary 2. *Let* $\mathcal{C}$ *be a class of graphs of sub-logarithmic expansion. Let* A *be an algorithm which provides a* c*-approximation for dominating sets on graph from* $\mathcal{C}$ *in* t *rounds. There is an algorithm which for every* $0 < \delta < 1$ *provides a* $(1+\delta)$*-approximation for dominating sets for every* n*-vertex graph* $G \in \mathcal{C}$ *in* $\mathcal{O}_\delta(t + \log^* n)$ *rounds.*

We have chosen to present this extension of Czygrinow et al. [6] because it suits very well to the results we obtained in the previous section. In particular, Corollary 2 in combination with Theorem 2 gives a a deterministic distributed $(1+\delta)$-approximation algorithm in $\mathcal{O}_\delta(\log^* n)$ rounds for dominating sets on graphs of genus at most g. We can similarly combine the corollary with Lenzen and Wattenhofer's result [18] on graphs of bounded arboricity to obtain $(1+\delta)$-approximations in $\mathcal{O}(\log n)$ randomized time on graphs of sub-logarithmic expansion, in particular, for all graphs which exclude a fixed minor or with the result of Amiri et al. [1] to obtain $(1+\delta)$-approximations in $\mathcal{O}(\log n)$ rounds on graphs of sub-logarithmic expansion.

Proof (of Theorem 4). Let G be the given input graph and let D be a dominating set of D with $|D| \leq c \cdot \mathrm{Opt}$, say $D = \{w_1, \ldots, w_k\}$. Associate each vertex $v \in V(G) \setminus D$ with one of its dominator, say with the one of minimum identifier

to obtain a partition $(W_1, \ldots, W_k)$ of G into clusters of radius 1. Then the graph $H = G/W_1/\ldots/W_k$ is a depth-1 minor of G with k vertices and and most $\nabla_1(G) \cdot k$ edges. Define an edge weight function on $E(H)$ by assigning unit weight to each edge. Set $\epsilon = \delta/(2c\nabla_1(G))$. Apply Lemma 16 to find a partition $(V_1, \ldots, V_l)$ of $V(H)$ such that the weight between different clusters is at most $\epsilon \cdot |E(H)|$. By uncontracting the partitions V_i and W_i, we obtain a partition $(U_1, \ldots, U_l)$ of $V(G)$, where each U_i has constant radius. Find an optimal dominating set S_i n each subgraph $G[U_i]$ and return the union $S = \bigcup_{1 \leq i \leq l} S_i$ of these dominating sets. Note that instead of solving the dominating set optimally on each $G[U_i]$, which may be considered an abuse of the local model by some, we may compute a sufficiently good approximation of an optimal dominating set. For this, we may use the PTAS [11] for dominating sets on graphs of polynomial expansion.

Since the U_i form a partition of $V(G)$, it is clear that S is a dominating set of G. Denote by S^* a dominating set of cardinality Opt. Let S' be obtained from S^* by adding for each U_i all vertices $w \in U_i$ which have a neighbor in a different cluster U_j. Then $S' \cap U_i$ is a dominating set of $G[U_i]$. Furthermore, we have

$$|S'| \leq |S^*| + 2\epsilon|E(H)| \leq \text{Opt} + 2c\epsilon\nabla_1(G) \cdot \text{Opt} = (1+\delta)\text{Opt}.$$

Observe that the local solutions S_i cannot be worse than the solution $S' \cap U_i$, hence

$$|S| = \sum_{1 \leq i \leq l} |S_i| \leq \sum_{1 \leq i \leq l} |S' \cap U_i| = |S'| \leq (1+\delta)\text{Opt}.$$

This finishes the proof of the theorem.

5 Conclusion

This paper presented the first constant round, constant factor local MDS approximation algorithm for locally embeddable graphs, a class of graphs which is much more general than planar graphs. Moreover, we have shown how our result can also be used to derive a $\mathcal{O}(\log^* n)$-time distributed approximation scheme for bounded genus graphs.

Our proofs are purely combinatorial and avoid any topological arguments. For the family of bounded genus graphs, topological arguments helped to improve the obtained approximation ratio in a preprocessing step. We believe that this result constitutes a major step forward in the quest for understanding for which graph families such local approximations exist. Besides the result itself, we believe that our analysis introduces several new techniques which may be useful also for the design and analysis of local algorithms for more general graphs, and also problems beyond MDS.

Moreover, this paper established an interesting connection between distributed computing and logic, by presenting a local approximation algorithm which is first-order logic definable. This also provides an interesting new perspective on the recently introduced notion of stone-age distributed computing [7]: distributed algorithms making minimal assumptions on the power of a node.

It remains open whether the local constant approximation result can be generalized to sparse graphs beyond bounded genus graphs. Also, it will be interesting to extend our study of first-order definable approximations.

References

1. Amiri, S. A., de Mendez, P. O., Rabinovich, R., and Siebertz, S. Distributed domination on graph classes of bounded expansion. *arXiv preprint arXiv:1702.02848* (2017).
2. Amiri, S. A., and Schmid, S. Brief announcement: A log-time local mds approximation scheme for bounded genus graphs.
3. Amiri, S. A., Schmid, S., and Siebertz, S. A local constant factor mds approximation for bounded genus graphs. In *Proc. ACM PODC* (2016).
4. Baker, B. S. Approximation algorithms for np-complete problems on planar graphs. *J. ACM 41*, 1 (Jan. 1994), 153–180.
5. Barenboim, L., Elkin, M., and Gavoille, C. A fast network-decomposition algorithm and its applications to constant-time distributed computation. In *Proc. SIROCCO*. 2014, pp. 209–223.
6. Czygrinow, A., Hańćkowiak, M., and Wawrzyniak, W. Fast distributed approximations in planar graphs. In *Proc. 22nd International Symposium on Distributed Computing (DISC)* (2008), pp. 78–92.
7. Emek, Y., and Wattenhofer, R. Stone age distributed computing. In *Proc. ACM Symposium on Principles of Distributed Computing (PODC)* (2013), pp. 137–146.
8. Eppstein, D. Diameter and treewidth in minor-closed graph families. *Algorithmica 27*, 3-4 (2000), 275–291.
9. Garey, M. R., and Johnson, D. S. Computers and intractability: a guide to the theory of NP-completeness. *WH Free. Co., San Fr* (1979).
10. Grohe, M. Local tree-width, excluded minors, and approximation algorithms. *Combinatorica 23*, 4 (2003), 613–632.
11. Har-Peled, S., and Quanrud, K. Approximation algorithms for polynomial-expansion and low-density graphs. In *Algorithms-ESA 2015*. Springer, 2015, pp. 717–728.
12. Hilke, M., Lenzen, C., and Suomela, J. Brief announcement: local approximability of minimum dominating set on planar graphs. In *Proc. ACM Symposium on Principles of Distributed Computing (PODC)* (2014), pp. 344–346.
13. Karp, R. M. Reducibility among combinatorial problems. In *Complexity of computer computations*. Springer, 1972, pp. 85–103.
14. Kuhn, F., Moscibroda, T., and Wattenhofer, R. Local computation: Lower and upper bounds. *arXiv preprint arXiv:1011.5470* (2010).
15. Kuhn, F., Moscibroda, T., and Wattenhofer, R. Local computation: Lower and upper bounds. *J. ACM 63*, 2 (Mar. 2016), 17:1–17:44.
16. Lenzen, C., Pignolet, Y. A., and Wattenhofer, R. Distributed minimum dominating set approximations in restricted families of graphs. *Distributed Computing 26*, 2 (2013), 119–137.
17. Lenzen, C., and Wattenhofer, R. Leveraging linial's locality limit. In *Distributed Computing*, vol. 5218 of *Lecture Notes in Computer Science*. 2008, pp. 394–407.

18. Lenzen, C., and Wattenhofer, R. Minimum dominating set approximation in graphs of bounded arboricity. In *Distributed Computing*, vol. 6343. 2010, pp. 510–524.
19. Linial, N. Locality in distributed graph algorithms. *SIAM J. Comput. 21*, 1 (Feb. 1992), 193–201.
20. Mohar, B., and Thomassen, C. *Graphs on Surfaces*. Johns Hopkins University Press, 2001.
21. Nešetřil, J., and De Mendez, P. O. *Sparsity: graphs, structures, and algorithms*, vol. 28. Springer Science & Business Media, 2012.
22. Suomela, J. Survey of local algorithms. *ACM Comput. Surv. 45*, 2 (Mar. 2013), 24:1–24:40.
23. Wawrzyniak, W. Brief announcement: a local approximation algorithm for mds problem in anonymous planar networks. In *Proceedings of the 2013 ACM symposium on Principles of distributed computing* (2013), ACM, pp. 406–408.
24. Wawrzyniak, W. A strengthened analysis of a local algorithm for the minimum dominating set problem in planar graphs. *Proc. Information Processing Letters (IPL) 114*, 3 (2014), 94 – 98.