Vol.2016-AL-157 No.7
2016/3/6

Fast and Simple Local Algorithms for 2-Edge Dominating Sets and 3-Total Vertex Covers

Toshihiro Fujito[1,a)] Daichi Suzuki[1,b)]

Abstract: A local algorithm is a deterministic (i.e., non-randomized) distributed algorithm in an anonymous port-numbered network running in a constant number of synchronous rounds, and this work studies the approximation performance of such algorithms. The problems treated are b-edge dominating set (b-EDS) that is a multiple domination version of the edge dominating set (EDS) problem, and t-total vertex cover (t-TVC) that is a variant of the vertex cover problem with a clustering property. After observing that EDS and 2-TVC are approximable within 4 and 3, respectively, using a single run of the local algorithm for finding a maximal matching in a bicolored graph, it will be seen that running the maximal matching local algorithm for bicolored graph twice, 2-EDS and 3-TVC can be approximated within factors 2 and 3, respectively.

1. Introduction

In the era of big data, it is almost mandatory to compute solutions an order of magnitude faster than ever before, and sublinear or constant time algorithms are urgently wanted in various areas of computation. It is fortunate meanwhile that the high computation power has become relatively easily accessible nowadays, and it is typically provided by computer networks of large scale. Distributed algorithms of high efficiency can be regarded as lying at the crossing of these demands and supplies, and this paper focuses on such algorithms running in constant time.

A *local algorithm* is a distributed algorithm, under the message-passing model of computation, that runs in a constant number of synchronous communication rounds (An excellent survey on local algorithms can be found in [26]). Here, the same computer network, called *communication graph* $G = (V, E)$, is both the input and the system for solving the problem. Each node of the communication graph is a computational entity having an unlimited computing power. The computation proceeds in rounds, in each of which each node can send and receive messages of unbounded length to and from all of its neighboring nodes (although the algorithms to be presented use only messages of $O(1)$ length). There are some variants in the communication graph models, and we assume a very weak one among them throughout the paper. It is assumed that a *port numbering* is assigned in G, which means that the edges incident to a node $u \in V$ are uniquely labeled and u can use those labels to choose which neighbors of u it sends messages to and receives from, for all the nodes u in G. While no other information such as *unique identifiers* are available to any nodes, it is also assumed in this paper that G is a graph of bounded degree, and there is a constant Δ such that any node in G has at most Δ neighbors. In this case, every node in G is initially given Δ as the only local input, and must produce the local output of its own, by running some algorithm common to all the other nodes in G. The computing power of distributed algorithms of this sort can be said to be severely limited, and independent sets or matchings, for instance, that can be computed in cycles are empty (vertex or edge) sets only [19]. Nevertheless, some nontrivial results, both positive and negative, are getting accumulated in recent years and the lists of those results can be found in [26].

The main problem treated in the paper is a graph covering problem called *edge dominating set*. In an undirected graph an edge is said to *dominate* itself and all the edges adjacent to it, and a set of edges is an *edge dominating set* (henceforth an *eds*) if the edges in it collectively dominate all the edges in a graph. The *edge dominating set* problem (henceforth *EDS*) asks to find an eds of minimum cardinality. The EDS problem is one of classic NP-complete graph problems, and it was proven to be so even if graphs are planar or bipartite of maximum degree 3 by Yannakakis and Gavril [28]. While the problem was later shown to remain NP-hard under various classes of restricted graphs [17], some polynomially solvable special cases have been also discovered [17], [21], [24]. Computing the minimum size edge dominating set is equivalent to that of the *minimum maximal matching*, and simply computing any maximal matching is a 2-approximation algorithm for them. Whereas EDS is known to admit a PTAS (polynomial time approximation scheme) for some special cases [4], [18], no better approximation algorithm has been found in the general case, and some nontrivial approximation lower bounds have been obtained (under some likely complexity hypothesis) [8], [9], [23]. The parameterized complexity of EDS has also been extensively studied [7], [9], [10], [13], [27].

[1] Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi 441–8580 Japan
[a)] fujito@cs.tut.ac.jp
[b)] suzuki@algo.cs.tut.ac.jp

The *b-edge dominating set* problem (henceforth *b-EDS*) is a multi-domination version of EDS, and it is a natural extension of EDS such as the (multi)set multicover and multi-dominating set problems. Here, each edge e of an input graph is associated with an integer $b(e)$, and a solution is required to dominate each e $b(e)$ times (and hence, the ordinary EDS corresponds to the case when $b(e) \equiv 1, \forall e \in E$). Typically two versions of b-EDS can be considered, depending on the types of feasible solutions, where a solution can be an edge multiset (called an *mb-eds* henceforth) in one case, and it has to be an ordinary edge set (called an *sb-eds* henceforth) in the other. The former is named *multiple b-edge dominating set* (henceforth *mb-EDS*) and the latter *simple b-edge dominating set* (henceforth *sb-EDS*). Whereas 8/3-approximation is known possible for the most general type of b-EDS [5], mb-EDS was shown approximable within 2 in linear time [6], and sb-EDS within 2 when $b(e) \leq 3, \forall e \in E$ [14]. The b-EDS problem treated in this paper is 2-EDS, that is the case when $b(e) \equiv 2, \forall e \in E$.

The EDS problem itself has some interesting applications, especially in view of its close relation to minimum maximal matchings, such as telephone switching networking as described in [28], and b-EDS plays an important role in any application of EDS when the fault tolerance and/or robustness need to be taken into account. Another aspect of an eds is that it induces a *vertex cover* where a vertex cover $C \subseteq V$ is a set of nodes such that every edge in G is incident to some node in C; namely, an edge set $D \subseteq E$ is an eds for a graph $G = (V, E)$ if and only if the set of endnodes of the edges in D, denoted $V(D)$, is a vertex cover for G. It is perhaps worth pointing out here that the vertex set $V(D)$ thus induced from an eds D is not a mere vertex cover but with a *clustering* property. A vertex set $C \subseteq V$ is said to be a *t-total vertex cover* ($t \geq 1$), henceforth a *t-tvc*, for a connected graph G if it is a vertex cover for G such that each connected component of the subgraph of G induced by C has at least t nodes. Hence, if C is a t-tvc, C is a vertex cover and each member of C belongs to a "cluster" containing at least t members of C. The problem of computing a minimum t-tvc is named *t-TVC* (thus, 1-TVC is the ordinary vertex cover problem). It was introduced in [12], [20], and was further studied in [11]. Having such clustering properties could be desirable or required in some applications, and variants with such properties enforced are considered in other combinatorial optimization problems as well, such as *r-gatherings* [1]. It is known that the t-TVC problem is NP-hard, not approximable within $10\sqrt{5} - 21 - \epsilon$ (unless P=NP), and approximable within 2, for each $t \geq 1$ [12].

1.1 Previous Work and Ours

Not so many works are known for EDS in the area of distributed algorithms, and it could be partially due to the fact that, at least under the model of local algorithms considered (i.e., deterministic distributed algorithms in anonymous port-numbered networks running in a constant number of rounds), the case is in a sense settled. It was shown by Suomela that EDS can be approximated within $4 - 2/\Delta$ in $O(\Delta^2)$ rounds, and the matching lower bound for approximation ratios was obtained at the same time [25]. Moreover, the same lower bound was shown to hold even if each node is provided with a unique identifier [15]. The vertex cover problem is known to be approximable within 2 by a local algorithm [2], [3], but nothing is known about the t-TVC problem for $t \geq 2$.

This work is mainly concerned with local algorithms for approximating the 2-EDS problem. It will be shown that, after observing in passing that EDS is approximable within 4 in only 2Δ rounds, m2-EDS is within 2 in the same running time. We then present a local algorithm for s2-EDS, designed by extending that for m2-EDS. Interestingly, approximation becomes easier in either version of 2-EDS than in EDS, and s2-EDS will be shown approximable within 2 in $4\Delta + 2$ rounds. Local algorithms for 2-TVC and 3-TVC are considered as well. It follows from the way vertex covers are constructed by the 3-approximation algorithm of Polishchuk and Suomela [22] that 2-TVC can be approximated within 3 in $2\Delta + 1$ rounds. It will be seen that 3-TVC can be approximated equally well, within the same factor of 3. A 3-tvc is obtained from an s2-eds computed by the previous algorithm, and it will be shown to become no larger than thrice the minimum vertex cover size despite the fact that the s2-eds used is, as constructed by extending an eds, in general larger than the eds used to 3-approximate 2-TVC.

2. Preliminaries

For an edge set $F \subseteq E$ in a graph $G = (V, E)$, $V(F)$ denotes the set of nodes induced by the edges in F (i.e., the set of all the endnodes of the edges in F). For a node set $S \subseteq V$ let $\delta(S)$ denote the set of edges incident to a node in S. When S is an edge set, we let $\delta(S) = \delta(\cup_{e \in S} e)$ where edge e is a set of two nodes; then, $\delta(S)$ also denotes the set of edges dominated by S. When S is a singleton set $\{s\}$, $\delta(\{s\})$ is abbreviated to $\delta(s)$. For a node set $U \subseteq V$, $N(U)$ denotes the set of neighboring nodes of those in U (i.e., $N(U) = \{v \in V \mid \{u, v\} \in E \text{ for some } u \in U\}$), and $N(u)$ means $N(\{u\})$.

An edge set in G is a *simple* 2*-matching* if at most two edges in it are incident to any node in G.

3. A Local Algorithm for EDS, m2-EDS, and 2-TVC

For a graph $G = (V, E)$ let $G_D = (V_L \cup V_R, E_D)$ denote the *bipartite double cover* of G, where $V_L = \{u_L \mid u \in V\}, V_R = \{u_R \mid u \in V\}$, and $E_D = \{\{u_L, v_R\}, \{u_R, v_L\} \mid \{u, v\} \in E\}$. Thus, there exist exactly two edges, $\{u_L, v_R\}$ and $\{u_R, v_L\}$, in G_D corresponding to any edge $\{u, v\}$ in G. Let $p : E_D \to E$ denote the function mapping each of $\{u_L, v_R\}$ and $\{u_R, v_L\}$ to $\{u, v\}$.

For any maximal matching $M_D \subseteq E_D$ computed in G_D, let $\tilde{M}$ denote the mapping of M_D into E; that is, $\tilde{M} = \{p(e) \mid e \in M_D\}$. The *multiplicity* of an edge $e \in \tilde{M}$ is the number of edges in M_D corresponding to e, and it is defined by the function $m : \tilde{M} \to \mathbb{N}$ such that $m(e) \stackrel{\text{def}}{=} |\ p^{-1}(e) \cap M_D\ |$. Clearly, $m(e) \in \{1, 2\}$ for all $e \in \tilde{M}$.

It is rather straightforward to verify that 1) $\tilde{M} \subseteq E$ is a simple 2-matching in G, and 2) $V(\tilde{M}) \subseteq V$ is a vertex cover for G [26]. While $\tilde{M}$ is not necessarily a maximal simple 2-matching in G, this means that $\tilde{M}$ is an edge dominating set for G as well, and we can say more:

Lemma 1. *For any maximal matching M_D in the bipartite double cover G_D of G,*
(1) $\tilde{M} \subseteq E$ is an eds and $(\tilde{M}, m)$ is an m2-eds for G, and
(2) $\sum_{e' \in \delta(e) \cap \tilde{M}} m(e') \leq 4$ for any $e \in E$,
where $\tilde{M} = p(M_D)$ and $m(e) = |\, p^{-1}(e) \cap M_D \,|$.

Proof. (1) Clearly, each of $\{u_L, v_R\}$ and $\{u_R, v_L\}$ is dominated by M_D in G_D for any edge $\{u, v\}$ of G as M_D is a maximal matching in G_D. Moreover, any edge dominating $\{u_L, v_R\}$ cannot simultaneously dominate $\{u_R, v_L\}$ in G_D, and vice versa, from the way G_D is constructed, for any $\{u, v\} \in E$. Therefore, there exist two different edges in M_D dominating $\{u_L, v_R\}$ and $\{u_R, v_L\}$ in G_D, and both of them appear in $(\tilde{M}, m)$, either as two edges or as a single edge with multiplicity of 2, and hence, $(\tilde{M}, m)$ is an m2-eds for G.
(2) Observe that $\sum_{e' \in \delta(e) \cap \tilde{M}} m(e')$ denote the number of edges in M_D dominating either $\{u_L, v_R\}$ or $\{u_R, v_L\}$ for $e = \{u, v\} \in E$. Any edge in G_D is dominated by at most two edges of the matching M_D, and hence, the number of edges in M_D dominating either $\{u_L, v_R\}$ or $\{u_R, v_L\}$ is at most 4 for any $\{u, v\} \in E$.

□ □

An eds $\tilde{M}$ can be computed by the following technique which has been often used in designing local algorithms for various graph problems.
(1) A key component of this technique is a simple local algorithm of Hańćkowiak et al. for computing a maximal matching in a bounded-degree bipartite graph G, with color classes L and R, where each node of G is informed of which color class of G it belongs to by the local input [16]. Port numberings are assumed but unique node identifies are not. The algorithm repeatedly performs the following steps for $i = 1, \cdots, \Delta$:
  (a) Any unmatched left node (in L) sends a proposal to its ith neighbor.
  (b) If any unmatched right node (in R) receives a proposal, it accepts the proposal, becomes matched, and informs the proposal sender of its acceptance. In case more than one proposal arrives simultaneously, it accepts the one received from a neighbor with the smallest port number.
  (c) If an unmatched left node (in L) receives a reply of acceptance from its ith neighbor, it becomes matched and halts (Otherwise, it goes on by returning to Step (1a)).
As Steps (1a) and (1c) can be executed in a single round, a maximal matching in a bipartite graph with the local inputs of color classes can be computed in 2Δ rounds.
(2) Observe now that, by simulating the algorithm above on G for the problem of finding a maximal matching in a bipartite graph, one can compute a maximal matching M_D in the bipartite double cover G_D; each node u of G simulates the behavior of both of its copies, the left node u_L and the right node u_R, both inheriting the port numbering of the original node u.
Once M_D is computed, $\tilde{M}$ is available almost immediately as $\{u, v\} \in \tilde{M}$ iff $\{u_L, v_R\}$ or $\{u_R, v_L\} \in M_D$. The multiplicity of each $e \in \tilde{M}$ is easy to compute as well. For each $u \in V$ matched by $\tilde{M}$, check if both of u_L and u_R are matched by M_D, and if so, check if their mates are the same ($m(e) = 2$ in this case) or not ($m(e) = 1$ in this case).

Mapping M_D to $\tilde{M}$ and setting the multiplicity of each edge in $\tilde{M}$ require no additional communication round, and hence, both $\tilde{M}$ and the multiset $(\tilde{M}, m)$ can be computed in 2Δ rounds.

To analyze the quality of an m2-eds $(\tilde{M}, m)$ computed by the algorithm above, let us consider an integer program formulation of the mb-EDS problem:

$$\min \{x(E) \mid x(\delta(e)) \geq b(e) \text{ and } x_e \in \mathbb{Z}_+, \forall e \in E\},$$

where $x(F) = \sum_{e \in F} x_e$ for $F \subseteq E$, and $\delta(e) = \{e\} \cup \{e' \in E \mid e' \text{ is adjacent to } e\}$ for $e \in E$. Replacing the integrality constraints by linear constraints $0 \leq x_e$, we obtain an LP and its dual LP in the following forms:

$$\begin{aligned}
\text{LP: } (\mathrm{P_{eds}}) \quad & \min z_P(x) = x(E) \\
\text{subject to: } \quad & x(\delta(e)) \geq b(e), \quad \forall e \in E \\
& x_e \geq 0, \quad \forall e \in E \\
\text{LP: } (\mathrm{D_{eds}}) \quad & \max z_D(y) = \sum_{e \in E} b(e) y_e \\
\text{subject to: } \quad & y(\delta(e)) \leq 1, \quad \forall e \in E \\
& y_e \geq 0, \quad \forall e \in E
\end{aligned}$$

Let $\tilde{y} \in \mathbb{R}^E$ denote a vector of dual variables for the multiset $(\tilde{M}, m)$ such that

$$\tilde{y}_e = \begin{cases} m(e)/4 & \text{if } e \in \tilde{M} \\ 0 & \text{otherwise} \end{cases}$$

We are ready to show the performance of the algorithm above for approximating EDS and m2-EDS problems:

Theorem 2. *The local algorithm given above computes a 4-approximation to EDS and a 2-approximation to m2-EDS, in 2Δ rounds.*

Proof. By Lemma 1.2 $\tilde{y}$ is dual feasible in LP:$(\mathrm{D_{eds}})$. In case of EDS, $b(e) \equiv 1, \forall e \in E$, and hence, its objective value is

$$z_D(\tilde{y}) = \sum_{e \in \tilde{M}} y_e = \sum_{e \in \tilde{M}} (m(e)/4) \geq \left|\tilde{M}\right|/4,$$

whereas it is

$$z_D(\tilde{y}) = \sum_{e \in \tilde{M}} 2y_e = \sum_{e \in \tilde{M}} (m(e)/2) = \left(\sum_{e \in \tilde{M}} m(e)\right)/2$$

in case of m2-EDS where $b(e) \equiv 2, \forall e \in E$. Therefore, the optimum of EDS is lower bounded by $|\tilde{M}|/4$ and that of m2-EDS by $\left(\sum_{e \in \tilde{M}} m(e)\right)/2$. □ □

Clearly, the vertex set $V(\tilde{M})$ is a 2-tvc for G, and it can be computed by each node checking if it is matched by $\tilde{M}$ after $\tilde{M}$ is computed. It is then exactly the algorithm of Polishchuk and Suomela [22], who showed that $V(\tilde{M})$ is no larger than thrice the minimum vertex cover size, and hence,

Corollary 3. *The* 2*-TVC problem can be approximated within* 3 *in* 2Δ *rounds.*

Remark: Better algorithms are known for the vertex cover problem [2], [3] as stated in Sect. 1, but their outputs are not necessarily 2-tvc's.

4. A Local Algorithm for s2-EDS and 3-TVC

As was seen already, $\tilde{M} \subseteq E$ computed by the algorithm of Sect. 3 is a simple 2-matching as well as an eds for $G = (V, E)$. It is not necessarily a maximal simple 2-matching, and even if it is so, it doesn't have to be a simple 2-eds.

As observed in the proof of Lemma 1.1, there exist two different edges, say e_1 and e_2, in M_D dominating $\{u_L, v_R\}$ and $\{u_R, v_L\}$ in G_D, for any $\{u, v\} \in E$. When M_D is mapped to $\tilde{M}$, however, these two might become one resulting in a single domination of $\{u, v\}$ in G. More precisely, when $\{u_L, v_R\}$ (or $\{u_R, v_L\}$) is dominated by these two edges e_1 and e_2 in G_D, $\tilde{M}$ dominates $\{u, v\}$ twice as $p(e_1) \neq p(e_2)$. Therefore, $\{u, v\}$ is dominated only once by $\tilde{M}$ if and only if e_1 (e_2, respectively) is the only edge of M_D dominating $\{u_L, v_R\}$ ($\{u_R, v_L\}$, respectively) in G_D and $p(e_1) = p(e_2)$. Formally, let $\tilde{M}$ be divided into $\tilde{M}_1$ and $\tilde{M}_2$ such that $\tilde{M}_2 = \{e \in \tilde{M} \mid p^{-1}(e) \subseteq M_D\}$ and $\tilde{M}_1 = \tilde{M} \setminus \tilde{M}_2 = \{e \in \tilde{M} \mid |p^{-1}(e) \cap M_D| = 1\}$. We can then restate the above argument as follows:

Lemma 4. *An edge e is dominated only once by $\tilde{M}$ in G iff e is dominated only by a single edge of $\tilde{M}_2$ in G.*

It thus suffices to dominate those edges specified in Lemma 4, on top of $\tilde{M}$, to construct an s2-eds. For this purpose, let $V_2 = V(\tilde{M}_2)$ and then, the set of edges subject to additional dominations is exactly $\tilde{M}_2 \cup E_2$, where $E_2 = \{\{u, v\} \in E \mid u \in V_2, v \notin V(\tilde{M})\}$.

To describe the algorithm for dominating those edges in $\tilde{M}_2 \cup E_2$, consider the bipartite graph $G_B = (V_2 \cup F, E_2)$ that we can find once $\tilde{M}$ is computed by the algorithm of Sect. 3, where $F = N(V_2) \setminus V(\tilde{M})$, the set of nodes in the neighborhood of V_2 and unmatched by $\tilde{M}$.

(1) Compute a simple 2-matching $\tilde{M} \subseteq E$ by running the algorithm of Sect. 3 on $G = (V, E)$.
(2) Compute a maximal matching M_B in G_B with color classes V_2 and F. To do so, we once again use the local algorithm of Hańćkowiak et al. [16]. Each node of G knows if it belongs to $V_2 = V(\tilde{M}_2)$ immediately after $\tilde{M}$ is computed in Step 1, and any node unmatched by $\tilde{M}$ can know if it belongs to F by checking if any of its neighbors belongs to $V(\tilde{M}_2)$ using one additional round.

Clearly, any edge in E_2 is dominated by M_B. On the other hand, there are three cases for $e = \{u, v\} \in \tilde{M}_2$ to consider: 1) $\{u, v\} \subseteq V(M_B)$ (i.e., both u and v matched by M_B), 2) $|\{u, v\} \cap V(M_B)| = 1$ (i.e., only one of them matched), and 3) $\{u, v\} \cap V(M_B) = \emptyset$ (i.e., neither matched). For each $\{u, v\} \in \tilde{M}_2$, u and v can check which is the case, by exchanging messages between them in one round. In cases 1) or 2) $\{u, v\}$ is successfully dominated twice by $\tilde{M} \cup M_B$, whereas it is still dominated only once (by $\{u, v\}$ itself) otherwise. So, we need to pick one additional edge to dominate $\{u, v\}$ in case 3), but picking exactly one edge among those incident to either u or v requires the symmetry breaking in general, and it is hard to do in an anonymous network. Therefore, instead of trying to do so, we let each of u and v to add one edge incident to it, other than $\{u, v\}$, to M_B while dropping $\{u, v\}$ from $\tilde{M}_2$.

(3) For any $\{u, v\} \in \tilde{M}_2$, if $\{u, v\} \cap V(M_B) = \emptyset$, each of u and v picks an edge incident to it other than $\{u, v\}$, and adds it to $\tilde{M}_2$ while dropping $\{u, v\}$ from $\tilde{M}_2$. In case when u or v cannot pick any edge other than $\{u, v\}$, then keep it in $\tilde{M}_2$.

Let $\tilde{M}_2' \subseteq E$ denote the edge set resulting from modifying $\tilde{M}_2$ in Step 3 above, and $\tilde{M}_2$ the original subset of $\tilde{M}$. It is then clear at this point that every edge in $\tilde{M}_2 \cup E_2$ is dominated twice by $\tilde{M}_2' \cup M_B$, and hence, the output $\tilde{M}_1 \cup \tilde{M}_2' \cup M_B$ of the algorithm is a valid s2-eds for G, which is computed in $4\Delta + 2$ rounds in total.

It remains to analyze the performance of this algorithm, and it will be based again on the the dual LP:($\mathrm{D_{eds}}$) of the LP relaxation for m2-EDS. Recall the vector $\tilde{y} \in \mathbb{R}^E$ of dual variables defined for the multiset $(\tilde{M}, m)$ in Sect. 3, and we also use it here as defined in terms of $\tilde{M}_1$ and $\tilde{M}_2$ such that

$$\tilde{y}_e = \begin{cases} 1/4 & \text{if } e \in \tilde{M}_1 \\ 1/2 & \text{if } e \in \tilde{M}_2 \\ 0 & \text{otherwise} \end{cases}$$

By the same reasoning as the one used for an m2-eds $(\tilde{M}, m)$, it can be seen that $\tilde{y}$ is dual feasible in LP:($\mathrm{D_{eds}}$), and moreover, the solution size $|\tilde{M}_1 \cup \tilde{M}_2' \cup M_B|$ would be bounded above by twice the objective value of $\tilde{y}$, which is $z_D(\tilde{y}) = \sum_{e \in \tilde{M}} 2y_e$, if it is the case that $|\tilde{M}_2' \cup M_B| \leq 2|\tilde{M}_2|$. Among the three cases considered earlier for $e = \{u, v\} \in \tilde{M}_2$, two edges of $\tilde{M}_2' \cup M_B$ can be distinctively associated with e in cases 2) and 3). In case of 1), however, where both u and v are matched by M_B, three edges (two from M_B and $e \in \tilde{M}_2'$) must be balanced with e. To deal with such a case, $\tilde{y}$ is modified as follows. Suppose that both u and v are matched by M_B for $\{u, v\} \in \tilde{M}_2$, and let e_1 and e_2 denote those two edges in M_B matching u and v, respectively. Replace $e = \{u, v\}$ in $\tilde{M}_2$ by these two edges e_1 and e_2, and do this operation for every $e \in \tilde{M}_2$ corresponding to case 1). Each of these operations can be seen to be an augmentation of the matching $\tilde{M}_2$ along an alternating path of length 3, and hence, the resulting edge set $\tilde{M}_2''$ remains as a matching in G_B. Moreover, no edge in M_B touches a node in $V(\tilde{M}_1)$, and therefore, when $\tilde{y}$ is altered to $\tilde{y}'$ such that

$$\tilde{y}_e' = \begin{cases} 1/4 & \text{if } e \in \tilde{M}_1 \\ 1/2 & \text{if } e \in \tilde{M}_2'' \\ 0 & \text{otherwise} \end{cases}$$

it remains dual feasible in LP:($\mathrm{D_{eds}}$). Since each edge $e \in \tilde{M}_2$ corresponding to case 1) is replaced by e_1 and e_2 in $\tilde{M}_2''$ distinctively, each with the dual value of $1/2$, those three edges associated with e, namely e_1 and e_2 in M_B and e itself, can be accounted for by the values of y_{e_1} and y_{e_2}, in bounding the solution size within a factor 2 of the optimum; or in other words,

$$|\tilde{M}_1 \cup \tilde{M}_2' \cup M_B| \leq 2z_D(\tilde{y}').$$

We may thus conclude:

Theorem 5. *The local algorithm given above computes a* 2*-approximation to s*2*-EDS in* $4\Delta + 2$ *rounds.*

Let us turn our attention to the 3-TVC problem. Each component of the subgraph of G induced by any s2-eds S for G contains at least two edges, and hence, $V(S)$ is always a 3-tvc for G. Therefore, attaching the following step, which requires no additional round of communication, to the above algorithm at the end enables it to compute a 3-tvc for G:

(4) For each $u \in V$ check if any edge incident to it belongs to the previous output of $\tilde{M}_1 \cup \tilde{M}'_2 \cup M_B$. Set the local output of u as "yes, I'm in a solution" if it does, and "no, I'm not in a solution" otherwise.

So the output of this algorithm is $V(\tilde{M}_1 \cup \tilde{M}'_2 \cup M_B)$, and it remains to estimate its size. To do so, consider the following LP relaxation of the vertex cover problem and its dual LP:

$$
\begin{aligned}
\text{LP: } (\mathrm{P_{vc}}) \quad & \min \sum_{v \in V} x_v \\
\text{subject to: } \quad & x_u + x_v \geq 1, \quad \forall \{u, v\} \in E \\
& x_v \geq 0, \quad \forall v \in V \\
\text{LP: } (\mathrm{D_{vc}}) \quad & \max \sum_{e \in E} y_e \\
\text{subject to: } \quad & y(\delta(v)) \leq 1, \quad \forall v \in V \\
& y_e \geq 0, \quad \forall e \in E
\end{aligned}
$$

where $y(F) = \sum_{e \in F} y_e$ for $F \subseteq E$.

Recall now the feasible solution $\tilde{y}' \in \mathbb{R}^E$ of LP:($\mathrm{D_{eds}}$) used in lower bounding the size of a minimum s2-eds, and observe that $\tilde{y}'(\delta(v)) \leq 1/2$ for all $v \in V$. It then means that $2\tilde{y}'$ is feasible in LP:($\mathrm{D_{vc}}$).

Let us now consider $V(\tilde{M}_1)$ and $V(\tilde{M}'_2 \cup M_B)$ separately:

- $\tilde{M}_1$ is a simple 2-matching consisting of paths of length at least 2 and cycles. For every component C of the subgraph induced by $V(\tilde{M}_1)$, let $V(C)$ and $\tilde{M}_1(C)$ denote the sets of nodes and edges in $\tilde{M}_1$ contained in C, respectively. Then, 1) $|\tilde{M}_1(C)| \geq 2$, and 2) $|V(C)| \leq |\tilde{M}_1(C)| + 1$. Therefore, when $|V(C)|$ is compared with the duals assigned on the edges of $\tilde{M}_1(C)$, we have
$$\frac{|V(C)|}{\sum_{e \in \tilde{M}_1(C)} 2\tilde{y}'_e} = \frac{|V(C)|}{|\tilde{M}_1(C)|/2} \leq \frac{2k+2}{k} \leq 3.$$
- The edge set $\tilde{M}'_2$ is obtained from the matching $\tilde{M}_2$ by adding more edges than deleted. It should be noted, however, that $V(\tilde{M}'_2 \cup M_B)$ remains the same as $V(\tilde{M}_2 \cup M_B)$ because M_B is a maximal matching in G_B. Also recall that nonzero duals are assigned, within G_B, only on the edges in $\tilde{M}''_2$. We here do the case analysis as was done earlier depending on the number of edges in M_B incident to u or v for $\{u, v\} \in \tilde{M}_2$.

  Case $\{u, v\} \subseteq V(M_B)$ (i.e., both u and v matched by M_B). In this case both of u and v are matched by two edges of M_B, say e_1 and e_2. It is also the case that both e_1 and e_2 are in $\tilde{M}''_2$ (but $\{u, v\}$ is not). Therefore, the dual value of $2(1/2 + 1/2) = 2$ can be associated with those 4 nodes matched by e_1 and e_2.

  Case $|\{u, v\} \cap V(M_B)| = 1$ (i.e., only one of them matched). There exists just one edge, say e, in M_B incident to either u or v. For those 3 nodes, u, v, and another one matched by e, the dual of $2 \times (1/2) = 1$ on $\{u, v\}$ can be associated.

  Case $\{u, v\} \cap V(M_B) = \emptyset$ (i.e., neither matched). There are only two nodes to account for in this case, namely, u and v, and the dual of 1 on the edge $\{u, v\}$ can be associated.

  In either case the number of nodes is thus bounded by thrice the corresponding dual values.

It follows that the number of nodes in a computed solution is no larger than three times the objective value of dual feasible $2\tilde{y}'$; i.e.,
$$V(\tilde{M}_1 \cup \tilde{M}'_2 \cup M_B) \leq 3 \sum_{e \in E} 2\tilde{y}'_e.$$
Therefore, although the 3-tvc $V(\tilde{M}_1 \cup \tilde{M}'_2 \cup M_B)$ is in general larger than the 2-tvc $V(\tilde{M})$ as the former is constructed by augmenting the latter into a 3-tvc, it is still within the range of 3-approximation of the minimum vertex cover, and hence,

Theorem 6. *The local algorithm given above computes a* 3*-approximation to* 3*-TVC in* $4\Delta + 2$ *rounds.*

Acknowledgments This work is supported in part by the Kayamori Foundation of Informational Science Advancement and a Grant in Aid for Scientific Research of the Ministry of Education, Science, Sports and Culture of Japan.

References

[1] Armon, A.: On Min-max r-gatherings, *Theoret. Comput. Sci.*, Vol. 412, No. 7, pp. 573–582 (2011).

[2] Åstrand, M., Floréen, P., Polishchuk, V., Rybicki, J., Suomela, J. and Uitto, J.: A Local 2-approximation Algorithm for the Vertex Cover Problem, *Proceedings of the 23rd International Conference on Distributed Computing*, DISC'09, pp. 191–205 (2009).

[3] Åstrand, M. and Suomela, J.: Fast Distributed Approximation Algorithms for Vertex Cover and Set Cover in Anonymous Networks, *Proceedings of the Twenty-second Annual ACM Symposium on Parallelism in Algorithms and Architectures*, SPAA '10, pp. 294–302 (2010).

[4] Baker, B.: Approximation algorithms for NP-complete problems on planar graphs, *J. ACM*, Vol. 41, pp. 153–180 (1994).

[5] Berger, A., Fukunaga, T., Nagamochi, H. and Parekh, O.: Approximability of the capacitated b-edge dominating set problem, *Theoret. Comput. Sci.*, Vol. 385, No. 1–3, pp. 202–213 (2007).

[6] Berger, A. and Parekh, O.: Linear Time Algorithms for Generalized Edge Dominating Set Problems, *Algorithmica*, Vol. 50, No. 2, pp. 244–254 (2008).

[7] Binkele-Raible, D. and Fernau, H.: Enumerate and measure: improving parameter budget management, *Proceedings of the International Conference on Parameterized and Exact Computation*, IWPEC'10, pp. 38–49 (2010).

[8] Chlebík, M. and Chlebíková, J.: Approximation hardness of edge dominating set problems, *J. Comb. Optim.*, Vol. 11, No. 3, pp. 279–290 (2006).

[9] Escoffier, B., Monnot, J., Paschos, V. T. and Xiao, M.: New Results on Polynomial Inapproximabilityand Fixed Parameter Approximability of Edge Dominating Set, *Theory Comput. Syst.*, Vol. 56, No. 2, pp. 330–346 (2015).

[10] Fernau, H.: EDGE DOMINATING SET: Efficient Enumeration-based Exact Algorithms, *Proceedings of the Second International Conference on Parameterized and Exact Computation*, IWPEC'06, pp. 142–153 (2006).

[11] Fernau, H., Fomin, F. V., Philip, G. and Saurabh, S.: The Curse of Connectivity: t-Total Vertex (Edge) Cover, *Proceedings of the 16th Annual International Conference on Computing and Combinatorics*, COCOON'10, pp. 34–43 (2010).

[12] Fernau, H. and Manlove, D. F.: Vertex and Edge Covers with Clustering Properties: Complexity and Algorithms, *J. of Discrete Algorithms*, Vol. 7, No. 2, pp. 149–167 (2009).

[13] Fomin, F. V., Gaspers, S., Saurabh, S. and Stepanov, A. A.: On Two Techniques of Combining Branching and Treewidth, *Algorithmica*, Vol. 54, No. 2, pp. 181–207 (2009).

[14] Fujito, T.: On Matchings and b-Edge Dominating Sets: A 2-Approximation Algorithm for the 3-Edge Dominating Set Problem, *Proc. 14th SWAT*, pp. 206–216 (2014).

[15] Göös, M., Hirvonen, J. and Suomela, J.: Lower Bounds for Local Approximation, *J. ACM*, Vol. 60, No. 5, pp. 39:1–39:23 (2013).

[16] Hańćkowiak, M., Karoński, M. and Panconesi, A.: On the Distributed Complexity of Computing Maximal Matchings, *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 219–225 (1998).

[17] Horton, J. and Kilakos, K.: Minimum edge dominating sets, *SIAM J. Discrete Math.*, Vol. 6, No. 3, pp. 375–387 (1993).

[18] Hunt III, H., Marathe, M., Radhakrishnan, V., Ravi, S., Rosenkrantz, D. and Stearns, R.: A unified approach to approximation schemes for NP- and PSPACE-hard problems for geometric graphs, *Proc. 2nd Ann. European Symp. on Algorithms*, pp. 424–435 (1994).

[19] Linial, N.: Locality in distributed graph algorithms, *SIAM J. Comput.*, Vol. 21, No. 1, pp. 193–201 (1992).

[20] Małafiejski, M. and Żyliński, P.: Weakly Cooperative Guards in Grids, *Proceedings of the 2005 International Conference on Computational Science and Its Applications - Volume Part I*, ICCSA'05, pp. 647–656 (2005).

[21] Mitchell, S. and Hedetniemi, S.: Edge domination in trees, *Proc. 8th Southeastern Conf. on Combinatorics, Graph Theory, and Computing*, pp. 489–509 (1977).

[22] Polishchuk, V. and Suomela, J.: A Simple Local 3-approximation Algorithm for Vertex Cover, *Inform. Process. Lett.*, Vol. 109, No. 12, pp. 642–645 (2009).

[23] Schmied, R. and Viehmann, C.: Approximating edge dominating set in dense graphs, *Theoret. Comput. Sci.*, Vol. 414, No. 1, pp. 92 – 99 (2012).

[24] Srinivasan, A., Madhukar, K., Nagavamsi, P., Rangan, C. P. and Chang, M.-S.: Edge domination on bipartite permutation graphs and cotriangulated graphs, *Inform. Process. Lett.*, Vol. 56, pp. 165–171 (1995).

[25] Suomela, J.: Distributed Algorithms for Edge Dominating Sets, *Proceedings of the 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, PODC '10, pp. 365–374 (2010).

[26] Suomela, J.: Survey of local algorithms, *ACM Comput. Surv.*, Vol. 45, No. 2, pp. 24–40 (2013).

[27] Xiao, M., Kloks, T. and Poon, S.-H.: New Parameterized Algorithms for the Edge Dominating Set Problem, *Theoret. Comput. Sci.*, Vol. 511, pp. 147–158 (2013).

[28] Yannakakis, M. and Gavril, F.: Edge dominating sets in graphs, *SIAM J. Appl. Math.*, Vol. 38, No. 3, pp. 364–372 (1980).