Theory Comput Syst (2015) 56:330–346
DOI 10.1007/s00224-014-9549-5

# New Results on Polynomial Inapproximability and Fixed Parameter Approximability of EDGE DOMINATING SET

**Bruno Escoffier · Jérôme Monnot · Vangelis Th. Paschos · Mingyu Xiao**

Published online: 13 June 2014

**Abstract** An edge dominating set in a graph $G = (V, E)$ is a subset $S$ of edges such that each edge in $E - S$ is adjacent to at least one edge in $S$. The EDGE DOMINATING SET problem, to find an edge dominating set of minimum size, is a basic and important NP-hard problem that has been extensively studied in approximation algorithms and parameterized complexity. In this paper, we present improved hardness results and parameterized approximation algorithms for EDGE DOMINATING SET. More precisely, we first show that it is NP-hard to approximate EDGE DOMINATING

Research partially supported by the French Agency for Research under the DEFIS program TODO, ANR-09-EMER-010, the National Natural Science Foundation of China under the Grant 61370071 and Fundamental Research Funds for the Central Universities under the Grant ZYGX2012J069. An extended abstract of the paper appears in the proceedings of IPEC'12, LNCS, volume 7535

B. Escoffier (✉)
Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, 75005, Paris, France
e-mail: bruno.escoffier@lip6.fr

B. Escoffier
CNRS, UMR 7606, LIP6, 75005 Paris, France

J. Monnot · V. Th. Paschos
PSL Research University, Université Paris-Dauphine, LAMSADE, CNRS UMR 7243, Paris, France
e-mail: monnot@lamsade.dauphine.fr

V. Th. Paschos
Institut Universitaire de France, Paris, France
e-mail: paschos@lamsade.dauphine.fr

M. Xiao
School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China
e-mail: myxiao@gmail.com

SET in polynomial time within a factor better than 1.18. Next, we give a parameterized approximation schema (with respect to the standard parameter) for the problem and, finally, we develop an $O^*(1.821^\tau)$-time exact algorithm where $\tau$ is the size of a minimum vertex cover of $G$.

**Keywords** Edge dominating set · Parameterized complexity · Approximation algorithms

## 1 Introduction

As one of the basic problems in Garey and Johnson's work on NP-completeness [15], EDGE DOMINATING SET has received high attention in history. It is NP-hard even in planar or bipartite graphs of maximum degree 3 [24]. Due to its theoretical and practical interests, many algorithms have been developed in order to tackle it. There is a simple 2-approximation algorithm for EDGE DOMINATING SET in unweighted graphs. It is not hard to verify that any maximum matching in the graph is an edge dominating set of size at most double of the minimum size. Carr et al. [6] proved a $(2 + \frac{1}{10})$-approximation algorithm for WEIGHTED EDGE DOMINATING SET (the generalization of EDGE DOMINATING SET where weights are assigned to the edges of the input graph and the objective becomes to determine a minimum total-weight edge dominating set), the ratio of which was later improved to 2 by Fujito and Nagamochi [14]. Improved results have also been obtained in sparse graphs [5] and in dense graphs [19]. However, providing an approximation algorithm with ratio (strictly) smaller than 2, or proving that such algorithm does not exist (under some likely complexity hypothesis) still remains as an open problem. Chlebik and Chlebikova [8] proved that it is NP-hard to approximate it within any factor better than $\frac{7}{6}$. Assuming the unique game conjecture (UGC), [19] showed some inapproximability results on dense instances, a corollary of which is that for every $\varepsilon > 0$ EDGE DOMINATING SET is inapproximable within ratio $3/2 - \varepsilon$ (under UGC).

In terms of parameterized complexity, EDGE DOMINATING SET, with parameter $k$ being the size of the solution, is fixed-parameter tractable (FPT). Fernau [12] gave an $O^*(2.6181^k)$-time algorithm that has been subsequently improved by Fomin *et al.* [13] downto $O^*(2.4181^k)$ and by Binkele-Raible and Fernau [1] downto $O^*(2.3819^k)$. Currently, the best result is the $O^*(2.3147^k)$-time algorithm by Xiao *et al.* [21]. When the graph is restricted to be of maximum degree 3, the result can be further improved to $O^*(2.1479^k)$ [22]. There is also a long list of contributions to exact algorithms for EDGE DOMINATING SET, such as the $O^*(1.4423^{|V|})$-time algorithm by Raman *et al.* [18], the $O^*(1.4082^{|V|})$-time algorithm by Fomin et al. [13], the $O^*(1.3226^{|V|})$-time algorithm by Rooij and Bodlaender [20], and finally the $O^*(1.3160^{|V|})$-time algorithm by Xiao and Nagamochi [23].

In this paper, we study parameterized approximation for EDGE DOMINATING SET. A parameterized approximation algorithm is a technique combining parameterization and approximation for getting approximation algorithms with fixed-parameter running time. In this way, we may be able to achieve approximation ratios unachievable (or yet unachieved) in polynomial time via fixed-parameter running times that are smaller than the running times of exact algorithms. We may also be able to use this technique to handle W[1]-hard problems which unlikely have fixed-parameter tractable algorithms. The interested reader can be referred to [4, 10, 17] for more about this issue. Let the parameter $k$ be the size of the solution to our problem. In the FPT framework, we want to design algorithms with running time $f(k)|I|^{O(1)}$ that decide whether there is a solution of size at most $k$ or not, where $f$ is a computable function. In approximation algorithms, we are interested in designing polynomial-time algorithms to find a solution of size $g(k)$, where $g$ is a computable function. In parameterized approximation, we wish to design algorithms with running time $f(k)|I|^{O(1)}$ that either find an approximate solution of size $g(k)$ or report that there is no solution of size $k$. Clearly, any fixed-parameter tractable problem allows parameterized approximation algorithms for any computable function $g$. However, this may not hold for W[1]-hard problems. For example, the dominating set problem (find a set $S$ of $k$ vertices in graph $G = (E, V)$ such that each vertex in $V - S$ is adjacent to at least one vertex in $S$) does not allow parameterized approximation algorithms for $g(k)$ of the form $k + c$ with fixed constant $c$ [10]. For EDGE DOMINATING SET, we are interested in designing parameterized approximation algorithms, which produce edge dominating sets of size at most $(1+\varepsilon)k$ (or assert that there is no solution of size $k$) in $f(k, \varepsilon)|I|^{O(1)}$ time for some computable function $f$. Of course, the goal is to find such an algorithm for a function $f$ which is smaller than the $O^*(2.3147^k)$-time (exact) FPT algorithm by Xiao et al. [21]. This issue has already been considered for other FPT problems, in particular for the MIN VERTEX COVER problem. In [2, 3] several parameterized approximation algorithms running faster than (exact) FPT algorithms and achieving ratios better than the ratio 2 (achievable in polynomial time) are given. Note that [3] asks as open question if similar results can be achieved for EDGE DOMINATING SET.

The remaining parts of this paper are organized as follows. In Section 2, we give an improved hardness result for EDGE DOMINATING SET by showing that it is not $5\sqrt{5} - 10 + \varepsilon < 1.18$ approximable in polynomial time unless P=NP. In Sections 3 and 4 we tackle parameterized approximation algorithms, answering positively to the open question in [3]. More precisely, in Section 3, we first give a simple algorithm to present the basic ideas, and then improve this algorithm in Section 4. We conclude the article in Section 5 by devising a parameterized algorithm for EDGE DOMINATING SET where the parameter is the vertex cover number of the graph.

## 2 An Improved Polynomial-Time Lower Bound

In this section, we give some new hardness results for EDGE DOMINATING SET, which are based on a reduction preserving approximation from the famous MIN VERTEX COVER problem (find a minimum subset $S$ of vertices in a graph such that each edge has at least one endpoint in $S$) to EDGE DOMINATING SET.

Before, recall some existing results between MIN VERTEX COVER and EDGE DOMINATING SET. The first two are rather folklore: there exist two simple approximation preserving reductions between MIN VERTEX COVER and EDGE DOMINATING SET transforming a polynomial-time $\rho$-approximation algorithm for one of them into a polynomial-time $2\rho$-approximation algorithm for the other one. Let $G = (V, E)$ be a simple graph and let $M^* \subseteq E$ and $C^* \subseteq V$ be a minimum edge dominating set and a minimum vertex cover of $G$, respectively. We will use $\tau = |C^*|$ to denote the size of a minimum vertex cover of $G$. Since, it is well known that $M^*$ can be supposed to be a maximal matching, we get $\tau = |C^*| \geqslant |M^*|$. Also $V(M^*)$, the set of endpoints of $M^*$, forms a vertex cover of $G$ and then $2|M^*| \geqslant \tau$. Thus:

$$\tau \geqslant |M^*| \geqslant \frac{\tau}{2}. \tag{1}$$

Now, from any $\rho$-approximation algorithm for MIN VERTEX COVER given by $V'$, we can polynomially find an edge dominating set $E'$ by taking at most one arbitrary edge incident to each vertex of $V'$. Thus, using (1) we get: $|E'| \leqslant |V'| \leqslant \rho \times \tau \leqslant 2\rho|M^*|$. Conversely, from any $\rho$-approximation algorithm for EDGE DOMINATING SET given by $M'$, we can construct a vertex cover $V' = V(M')$ of $G$ by taking the endpoints of $M'$. Hence, using (1) we deduce: $|V'| = 2|M'| \leqslant 2\rho|M^*| \leqslant 2\rho \times \tau$.

In Theorem 1 just below, we improve the expansion $2\rho$ of the reduction to $2\rho - 1$. Dealing with weighted versions of these two problems, it is proved in [6] that weighted MIN VERTEX COVER can be approximated as well as weighted EDGE DOMINATING SET.

**Theorem 1** *For any $\rho \geqslant 1$, if there is a polynomial-time $\rho$-approximation algorithm for* EDGE DOMINATING SET, *then there exists a polynomial-time $(2\rho - 1)$-approximation algorithm for* MIN VERTEX COVER.

*Proof* We will show that for each instance $G = (V, E)$ of MIN VERTEX COVER, we can construct at most $|V|$ instances $G_i = (V_i, E_i)$ (where $|V_i| \leqslant 3|V|$) of EDGE DOMINATING SET such that a $(2\rho - 1)$-approximation solution to $G$ can be found in polynomial time based on a $\rho$-approximation solution to each $G_i$. For each positive integer $1 \leqslant i \leqslant |V|$, the graph $G_i = (V_i, E_i)$ is a graph constructed from $G$ in the following way: $V_i = V \cup \{a_j, a'_j : j \in \{1, \dots, i\}\}$ and $E_i = E \cup F_i \cup H_i$, where $F_i = \{(a_j, a'_j) : j \in \{1, \dots, i\}\}$ and $H_i = \{(v, a_j) : v \in V, j \in \{1, \dots, i\}\}$. Informally, $G_i$ contains a copy of $G$, an induced matching $F_i$ and a complete bipartite graph between the vertices of $G$ and the left part of the induced matching $F_i$. In Fig. 1 an illustration of the construction of $G_i$ for $i = 2$ is given.

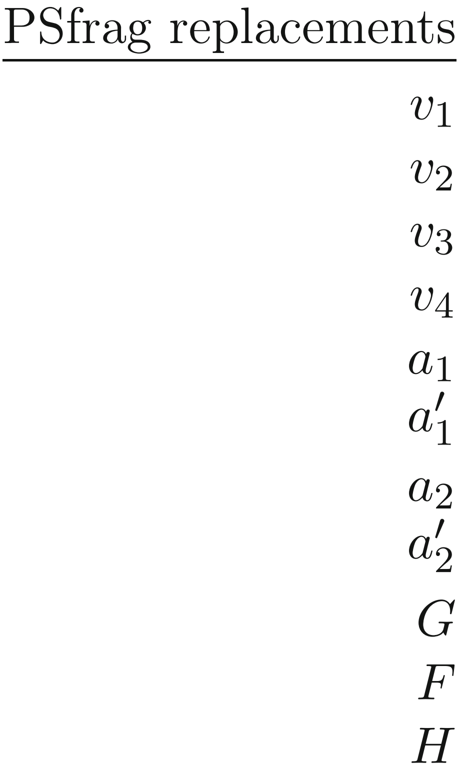
We first show that a $\rho$-approximation solution to $G_\tau$ implies a $(2\rho - 1)$-approximation solution to $G$, where $\tau$ is the size of a minimum vertex cover of $G$.

Let $M^*_\tau$ and $C^*_\tau$ be a minimum edge dominating set and a minimum vertex cover of $G_\tau$ respectively. Since $G_\tau$ contains $\tau$ independent edges $F_\tau$, we know that $|M^*_\tau| \geqslant \tau$. On the other hand, a perfect matching between $C^*$ and $\{a_1, \cdots, a_\tau\}$ is an edge dominating set of size $\tau$ of $G_\tau$. We so have:

$$|M^*_\tau| = \tau. \tag{2}$$

Let $M_\tau$ be a $\rho$-approximation edge dominating set of $G_\tau$ and $U_\tau = V(M_\tau) \cap V(G)$. We can see that $U_\tau$ is a vertex cover of $G$. Since $M_\tau$ is an edge dominating set of $G_\tau$

**Fig. 1** An illustration of the construction of $G_i$ for $i = 2$



and $F_\tau$ is a set of $\tau$ independent edges in $G_\tau$, we know that $V(M_\tau)$ contains at least $\tau$ vertices in $V(F_\tau)$. Therefore, we have:

$$|U_\tau| \leqslant 2|M_\tau| - \tau. \tag{3}$$

By combining the fact that $|M_\tau| \leqslant \rho|M_\tau^*|$ together with (2) and (3), we get:

$$|U_\tau| \leqslant 2|M_\tau| - \tau \leqslant 2\rho|M_\tau^*| - \tau = (2\rho - 1)\tau. \tag{4}$$

Therefore, $U_\tau$ is a vertex cover of size at most $(2\rho - 1)\tau$ of $G$.

However, we cannot construct $G_\tau$ in polynomial time directly, since it is NP-hard to compute the size of the minimum vertex cover $\tau$. To handle this problem, we compute $M_i$ and $U_i$ for each $G_i$ with $i \in \{1, \cdots, |V(G)|\}$, and return $U_{i^*}$ such that $|U_{i^*}| \leqslant \min_{i=1}^{|V(G)|}\{|U_i|\}$ and $i^* \in \{1, \cdots, |V(G)|\}$. Hence, by (4), $U_{i^*}$ is a vertex cover of $G$ with size $|U_{i^*}| \leqslant |U_\tau| \leqslant (2\rho - 1)\tau$. □

It is NP-hard to approximate MIN VERTEX COVER within any factor smaller than $10\sqrt{5} - 21$ by a result of Dinur and Safra [9]. By this result and Theorem 1, we get the following corollary.

**Corollary 1** *For any $\varepsilon > 0$,* EDGE DOMINATING SET *is not $(5\sqrt{5} - 10 + \varepsilon)$-approximable in polynomial time unless $P = NP$.*

Note that under UGC, since MIN VERTEX COVER cannot be approximated to within $2 - \varepsilon$ for any $\varepsilon > 0$ [16], we get that for any $\varepsilon > 0$, EDGE DOMINATING SET is not $(3/2 - \varepsilon)$-approximable in polynomial time, which is the same lower bound recently achieved in [19].

## 3 A Simple Parameterized Approximation Schema

In this section, we design a simple parameterized approximation schema for EDGE DOMINATING SET. As mentioned in Introduction, this algorithm contains the basic idea upon which the improved algorithms in Section 4 is built.

### 3.1 Constrained Edge Dominating Set

First of all, we introduce a CONSTRAINED EDGE DOMINATING SET problem and present some properties for it. Given a graph $G = (V, E)$ and a prescribed subset $V_1 \subseteq V$ of non-isolated vertices, an edge dominating set $M$ is called a constrained edge dominating set of $G$, if $V_1 \subseteq V(M)$. In the CONSTRAINED EDGE DOMINATING SET problem, we are asked to find a constrained edge dominating set of minimum size. CONSTRAINED EDGE DOMINATING SET is a natural generalization of EDGE DOMINATING SET where $V_1 = \emptyset$. We show a simple approximation algorithm for CONSTRAINED EDGE DOMINATING SET.

**Lemma 1** *For an instance $(G, V_1)$ of* CONSTRAINED EDGE DOMINATING SET, *let $M_1$ be a maximum matching in the induced graph $G[V_1]$, $M_2$ be a maximum matching in the induced graph $G[V - V_1]$, and $M_3$ be a set of $|V_1 - V(M_1)|$ edges such that each edge in $M_3$ is incident on a different vertex in $V_1 - V(M_1)$. Edge set $M' = M_1 \cup M_2 \cup M_3$ is a constrained edge dominating set with size $|M'| \leqslant (2-\rho_1)\nu$, where $\nu$ is the size of a minimum constrained edge dominating set $M^*$ and $\rho_1\nu$ is the number of edges in $M^*$ with both endpoints in $V_1$.*

*Proof* Let $\alpha_1, \alpha_2$ and $\alpha_3$ be, respectively, the numbers of edges of $M^*$ with both endpoints in $V_1$, one endpoint in $V_1$ and the other one in $V - V_1$, and both endpoints in $V - V_1$. Since $V_1 \subseteq V(M^*)$, we have $2\alpha_1 + \alpha_2 = |V_1| = 2|M_1| + |M_3|$. Since $M_1$ is a maximum matching in $G[V_1]$, $|M_1| \geqslant \alpha_1$. Note finally that for each edge in $M_2$ at least one of its endpoints has to be in $V(M^*)$, and then $\alpha_2 + 2\alpha_3 \geqslant |M_2|$.

From these inequalities, we obtain $2|M_1| + |M_2| + |M_3| \leqslant 2|M^*|$, and then $|M'| \leqslant (2 - \rho_1)\nu$ since $|M_1| \geqslant \alpha_1 = \rho_1\nu$. □

Note that Lemma 1 is a special case of Lemma 3 in the next section (but we prefer to give a proof of both lemmas for readability). Note also that Lemma 1 implies a 2-approximation algorithm for CONSTRAINED EDGE DOMINATING SET, with a better ratio obtained when the set $V_1$ is large.

### 3.2 A Parameterized Approximation Schema for EDGE DOMINATING SET

As already mentioned in introduction, deciding whether a graph contains an edge dominating set of size $k$ can be done in $O^*(2.3147^k)$ time by the parameterized algorithm presented in [21]. Here we design a parameterized approximation algorithm for it. It is based on the following property:

**Property 1** *Suppose that there are a set $V_1$ and an edge dominating set $M$ such that $V_1 \subseteq V(M)$, $|M| \leqslant k$ and $|V_1| = k + \rho' k$. Then the number of edges in $M$ that have both endpoints in $V_1$ is at least $\rho' k$.*

Indeed, if there were $\alpha < \rho' k$ edges in $M$ with both endpoints in $V_1$, then the number of vertices in $V_1$ would be at most $2\alpha + (|M| - \alpha) \leqslant |M| + \alpha < k + \rho' k = |V_1|$, a contradiction. Together with Lemma 1, this means (taking $M = M^*$) that the computed edge set $M'$ is of size at most $(2 - \rho')k$.

Then, our goal is to find such a large set $V_1$. As in several articles devising FPT algorithms for EDGE DOMINATING SET, we can use the fact that $V(M^*)$ for a minimum edge dominating set $M^*$ is a vertex cover of $G$. For each edge in the graph, at least one endpoint of it is in $V(M^*)$. Then, we can use a branching algorithm to construct a set $V_1$ of size up to $k + \rho' k$ such that $V_1$ is part of the vertex set of a minimum edge dominating set $V(M^*)$ in $G$. We iteratively select an edge $(a, b)$ in the current graph and branch into two branches by including either $a$ or $b$ into $V_1$ and delete it from the graph until the size of $V_1$ is $k + \rho' k$ or the remaining graph has no edge. This process produces at most $2^{k+\rho' k}$ vertex sets $V_1$ of size at most $k + \rho' k$ in $O^*(2^{k+\rho' k})$ time and at least one of them is contained in $V(M^*)$. For each of the vertex sets $V_1$, we use the algorithm in Lemma 1 to compute $M'$ and return a smallest one. Using Property 1, the returned edge set is an edge dominating set of size at most $(2 - \rho')k$ if $|M^*| \leqslant k$ (note that if in a leaf of the search tree we have a set $V_1 \subseteq V(M^*)$ with $|V_1| < k + \rho' k$, this means that the remaining graph is empty and the output solution is then optimal by Lemma 1). By taking $\rho' = 1 - \rho$, we deduce the following result.

**Lemma 2** *For any $\rho > 0$, there exists a $(1 + \rho)$-approximation algorithm to $k$-EDGE DOMINATING SET running in $O^*(2^{(2-\rho)k})$ time for* $0 \leqslant \rho \leqslant 1$.

When $\rho = 0$, Lemma 2 implies that $k$-EDGE DOMINATING SET can be solved in $O^*(4^k)$ time, which is far away from the current best parameterized algorithm of running time $O^*(2.3147^k)$. To reduce the gap, we will improve the running time bound of our parameterized approximation schema in the next section.

## 4 Improved Parameterized Approximation Schemata

In the algorithm presented in Section 3.2, in order to search $V_1$ we may need to branch on each edge. One way to reduce the running time is to reduce the number of branchings in the algorithm. This approach has been used for (exact) FPT algorithms to obtain improved running times. We will use some of these improved branchings, but we need to combine them with approximability. We first deal with these approximation properties in Section 4.1 and then present the improved parameterized approximation algorithm in Section 4.2.

### 4.1 More Approximation Algorithms for CONSTRAINED EDGE DOMINATING SET

Given a graph $G = (V, E)$. We consider a partition $(V_1, V_2, V_3)$ of the vertex set $V$ such that:

- Each connected component of the induced graph $G[V_2]$ is a clique; and
- There is no edge between a vertex in $V_2$ and a vertex in $V_3$.

Once the set $V_1$ is given, we can find in linear time the set of connected components of $G[V - V_1]$ which are cliques and which constitute $V_2$. Let us now give more properties of our problems based on this partition.

We consider an instance $(G = (V, E), V_1)$ of CONSTRAINED EDGE DOMINATING SET. Let $M^*$ be a minimum constrained edge dominating set of $(G = (V, E), V_1)$ and $\nu = |M^*|$.

We denote by $\alpha_1$ (resp., $\alpha_2$, $\alpha_3$) the number of edges in $M^*$ with both endpoints in $V_1 \cup V_2$ (resp., with one endpoint in $V_1$ and one in $V_3$, both endpoints in $V_3$). This partitions the edge set $E$ into three sets, hence, $\nu = \alpha_1 + \alpha_2 + \alpha_3$.

Moreover, since the connected components of $G[V_2]$ are cliques and $V(M^*)$ is a vertex cover of $G$, we know that $V(M^*)$ contains at least $|C_i| - 1$ vertices in each clique $C_i$ of $G[V_2]$. Assume that there are $p$ cliques $C_1, \cdots, C_p$ in $G[V_2]$ among which $q$ cliques $Q_1, \cdots, Q_q$ are such that $V(Q_i) \subseteq V(M^*)$. Then $V(M^*) \cap V_2 = |V_2| - p + q$. In other words, we have:

$$2\alpha_1 + \alpha_2 = |V(M^*) \cap (V_1 \cup V_2)| = |V_1| + |V_2| - p + q. \tag{5}$$

We are ready now to specify an approximation algorithm for CONSTRAINED EDGE DOMINATING SET (Algorithm **ApproxPoly1** in Fig. 2), which is a generalization of the algorithm in Lemma 1.

**Lemma 3** *Edge set $M$ =**ApproxPoly1**$(G)$ is a constrained edge dominating set of $(G, V_1)$ with size $|M| \leqslant (2 - \rho_1)\nu$, where $\rho_1\nu = \alpha_1$ is the number of edges in $M^*$ with both endpoints in $V_1 \cup V_2$.*

*Proof* We first show that $M$ is a constrained edge dominating set. It is easy to see that $M' = M_1 \cup M_2 \cup M_3$ is an edge dominating set and each vertex in $V_1$ will appear in at least one edge in $M_1 \cup M_3$. When we remove an edge $e = (u, c'_i) \in M'_1$ with $c'_i \in V'_2$ from $M'$, we know by Step 3 of the algorithm that every other neighbor $w$ of $u$ is saturated by $M_1$. Hence, the edge $(u, w)$ incident to $u$ is still dominated and then $M$ is still an edge dominating set. Furthermore vertex $u$ is not in $V_1$ and we know that $M$ is a constrained edge dominating set.

Now we prove the claim on the size of $M$. By the construction, we know that the size of the maximum matching in $G[V_1 \cup V_2 \cup V'_2]$ is at least $\alpha_1 + p - q$ (take the $\alpha_1$

```
Input: A graph $G = (V = V_1 \cup V_2 \cup V_3, E)$ with the above partition of $V$.
Output: An edge dominating set $M$ such that $V_1 \subseteq V(M)$.

1. Add a vertex $c'_i$ to each clique $C_i$ in $G[V_2]$, to create a clique of size $|C_i| + 1$.
   Let $V'_2 = \{c'_1, \cdots, c'_p\}$ be the set of added vertices.
2. Compute a maximum matching $M_1$ in $G[V_1 \cup V_2 \cup V'_2]$.
3. While there is an edge $e = (u, c'_i)$ in $M_1$ with $c'_i \in V'_2$ and there exists a
   neighbor $w$ of $u$ not saturated by $M_1$, replace $e$ with $(u, w)$ in $M_1$.
4. Let $M'_1$ be the set of edges in $M_1$ with an endpoint in $V'_2$.
5. Compute a maximum matching $M_2$ in $G[V_3]$.
6. For each unsaturated vertex in $V_1$, select an arbitrary edge incident on it.
   Let $M_3$ be the set of such edges.
7. Output $M = M_1 \cup M_2 \cup M_3 - M'_1$.
```

**Fig. 2** Algorithm **ApproxPoly1**

edges of $M^*$ in $V_1 \cup V_2$, and add the $p - q$ edges between an unsaturated vertex in a clique of $G[V_2]$ and the corresponding vertex $c'_i$). Then, we get $|M_1| \geqslant \alpha_1 + p - q$.

Since each edge in $M_1 - M'_1$ contains two vertices in $V_1 \cup V_2$, each edge in $M_3 \cup M'_1$ contains one vertex in $V_1 \cup V_2$, and all of these vertices are different, it holds that

$$2|M_1| + |M_3| - |M'_1| = 2(|M_1| - |M'_1|) + (|M_3| + |M'_1|) \leqslant |V_1| + |V_2|.$$

Therefore:

$$|M_1| + |M_3| - |M'_1| \leqslant |V_1| + |V_2| - p + q - \alpha_1 = 2\alpha_1 + \alpha_2 - \alpha_1 = \alpha_1 + \alpha_2. \quad (by(5)) \tag{6}$$

Note that there are at most $\alpha_2 + 2\alpha_3$ different vertices in $V(M^*) \cap V_3$. Then:

$$|M_2| \leqslant \alpha_2 + 2\alpha_3. \tag{7}$$

Summing (6) and (7), we get $|M| = |M_1| + |M_2| + |M_3| - |M'_1| \leqslant \alpha_1 + 2\alpha_2 + 2\alpha_3 = (2 - \rho_1)\nu$. □

Note that Lemma 1 is a special case of Lemma 3 where the vertex set $V_2$ is an empty set. Lemma 3 shows that we do not need to branch on each clique component in $G[V - V_1]$ in order to search the vertex set of a constrained edge dominating set.

To improve the running time of our parameterized approximation schema, we also need to consider a particular case of the graph where in the partition $(V_1, V_2, V_3)$ each connected component of $G[V_3]$ is a path of length 2.

Let $N$ be the number of these paths in $G[V_3]$. Considering a minimum constrained edge dominating set $M^*$, we denote by:

- $N_1$ the set of paths in $G[V_3]$ such that there is an edge in $M^*$ between a vertex in $V_1$ and the central vertex of the path; set $n_1 = |N_1|$;
- $N_2$ the set of paths in $G[V_3]$ such that there is an edge of the path in $M^*$; set $n_2 = |N_2|$; and

- $N_3$ the set of remaining paths in $G[V_3]$; set $n_3 = |N_3|$.

Observe that some paths of $G[V_3]$ may be counted twice (one with $N_1$ and one with $N_2$, if $M^*$ is not a matching); so, $N \leqslant n_1 + n_2 + n_3$. Note that for each of the $n_3$ remaining paths, $M^*$ has to take two edges (between $V_1$ and the endpoints of the path) to cover the edges of the path. In other words, $\alpha_2 \geqslant 2n_3 + n_1$. Moreover, by definition, $n_2 \leqslant \alpha_3$.

Consider Algorithm **ApproxPoly2** (Fig. 3) on an instance $(G, V_1)$ of CONSTRAINED EDGE DOMINATING SET.

**Lemma 4** *Edge set* $M$ =`ApproxPoly2`$(G)$ *is a constrained edge dominating set of* $(G, V_1)$ *with size* $|M| \leqslant \nu + n_3$.

*Proof* The fact that $M$ is a constrained edge dominating set can be obtained similarly as in the proof of Lemma 3. We only need to prove the claim on the size of $M$.

Let us denote by $\gamma_1$ (resp., $\gamma_2$) the number of edges of $M_1$ that have both endpoints in $V_1 \cup V_2$ (resp., one endpoint in $V_1$, the other one being a central vertex of a path in $V_3$). Then, $|M_1| = \gamma_1 + \gamma_2 + |M_1'|$.

By the construction, we know that the size of the maximum matching in $G[V_1 \cup V_2 \cup V_2' \cup V_3']$ is at least $\alpha_1 + n_1 + p - q$. Then we get: $|M_1| = \gamma_1 + \gamma_2 + |M_1'| \geqslant \alpha_1 + n_1 + p - q$. We also have $2\gamma_1 + \gamma_2 + |M_1'| + |M_3| \leqslant |V_1| + |V_2|$, which is equivalent to $\gamma_1 + |M_3| \leqslant |V_1| + |V_2| - \gamma_1 - \gamma_2 - |M_1'|$. Therefore:

$$\gamma_1+|M_3| \leqslant |V_1|+|V_2|-p+q-\alpha_1-n_1 = 2\alpha_1+\alpha_2-\alpha_1-n_1 = \alpha_1+\alpha_2-n_1 \quad (by (5)) \tag{8}$$

**Input**: A graph $G = (V = V_1 \cup V_2 \cup V_3, E)$, where each component in $G[V_3]$ is a path of length 2.
**Output**: An edge dominating set $M$ such that $V_1 \subseteq V(M)$.

1. Add a vertex $c_i'$ to each clique $C_i$ in $G[V_2]$, to create a clique of size $|C_i| + 1$. Let $V_2' = \{c_1', \cdots, c_p'\}$ be the set of added vertices.
2. Compute a maximum matching $M_1$ in $G[V_1 \cup V_2 \cup V_2' \cup V_3']$, where $V_3'$ is the set of central vertices of paths in $G[V_3]$.
3. While there is an edge $e = (u, c_i')$ in $M_1$ such that $c_i' \in V_2'$ and there exists a neighbor $w$ of $u$ not saturated by $M_1$, then replace $e$ with $(u, w)$ in $M_1$.
4. Let $M_1'$ be the set of edges in $M_1$ with an endpoint in $V_2'$.
5. For each path where the central vertex is not saturated by $M_1$, take one edge in this path.
   Let $M_2$ be this set of edges.
6. For each unsaturated vertex in $V_1$, select an arbitrary edge. Let $M_3$ be the set of such edges.
7. Output $M = M_1 \cup M_2 \cup M_3 - M_1'$.

**Fig. 3** Algorithm `ApproxPoly2`

Note that $\gamma_2 + |M_2| = N$, $N \leqslant n_1 + n_2 + n_3 \leqslant n_1 + \alpha_3 + n_3$ and $\nu = \alpha_1 + \alpha_2 + \alpha_3$. We get:

$$\begin{aligned}
&|M| = |M_1| + |M_2| + |M_3| - |M_1'| = \gamma_1 + \gamma_2 + |M_2| + |M_3| = \gamma_1 + |M_3| + N\\
&\leqslant \alpha_1 + \alpha_2 - n_1 + N \qquad (by\ (8))\\
&\leqslant \nu + n_3.
\end{aligned}$$

that concludes the proof of the lemma. □

### 4.2 An Improved Parameterized Approximation Schema

Now we are able to give the improved parameterized approximation schema ApproxFPT for $k$-EDGE DOMINATING SET as well as $k$-CONSTRAINED EDGE DOMINATING SET. As explained earlier, the principle is to search the vertex set $V_1$ by using some 'good' branchings. Then, in each leaf of our search tree, we will use the approximation algorithms devised in Section 4.1 (either directly, or after some other steps).

We consider a $k$-constrained edge dominating set $(G, V_1)$ with partition $I = (V_1, V_2, V_3)$ of the vertex set. Let $t = |V_1| + |V_2| - p$ (where $p$ is the number of cliques in $G[V_2]$). When $t \geqslant (2-\rho)k$ $(0 \leqslant \rho \leqslant 1)$, there are at least $(1-\rho)k$ edges in any optimal solution $M^*$ with both endpoints in $V_1 \cup V_2$. Therefore, Lemma 3 implies that a $(1+\rho)$-approximation solution to $k$-CONSTRAINED EDGE DOMINATING SET can be found in polynomial time, if $t \geqslant (2-\rho)k$. We will use a branch-and-search method to move vertices from $V_3$ to $V_1 \cup V_2$ and therefore to increase the parameter $t$. Note that for each vertex $v \in V_3$, it is either in $V(M^*)$ or not. For the second case, all neighbors of $v$ should be in $V(M^*)$ since $V(M^*)$ is a vertex cover of the graph. Then, we can branch on $v$ by either moving $v$ into $V_1$ (this means $v \in V(M^*)$) or by moving the neighbor set $N(v)$ of $v$ in $G[V_3]$ into $V_1$ (this means $v \notin V(M^*)$) and moving all newly created clique components in $G[V_3]$ into $V_2$. When $v$ is a vertex of degree $\geqslant 3$ in $G[V_3]$, we can branch with recurrence:

$$C(t) \leqslant C(t+1) + C(t+3), \tag{9}$$

where $C(t)$ is the worst size of the search tree in the algorithm when the current value of $|V_1| + |V_2| - p$ is $t$. When the maximum degree of $G[V_3]$ is at most 2, we may only get:

$$C(t) \leqslant C(t+1) + C(t+2),$$

by branching on a maximum degree vertex. In fact, there are some techniques to branch on a component $H$ in $G[V_3]$ with a recurrence not worse than (9), if $H$ is not a path of length 2 [20, 21, 23].

For a path $p_1p_2p_3p_4\ldots$ of length at least 3, we can branch on $p_3$ by including it into $V_1$ or including its neighbors $p_2$ and $p_4$ into $V_1$. For the first branch, we will also move a clique component $p_1p_2$ into $V_2$. Then we can get:

$$C(t) \leqslant C(t+2) + C(t+2), \tag{10}$$

which is better than (9).

For a cycle of length at least 5, we branch on an arbitrary vertex in the cycle and then branch on the generated paths in each branch and finally we can get a recurrence not worse than (9). For a cycle $c_1c_2c_3c_4$ of length 4, we can also branch with (10) by including either $\{c_1, c_3\}$ or $\{c_2, c_4\}$ into $V_1$. For the details about the proof of this fact, reader is referred to [20, 21, 23].

It turns out that only for a component of path of length 2 in $G[V_3]$ we cannot branch with a recurrence as good as (9). We will call a branching with recurrence at least as (9) a *good branching*.

The main steps of the improved parameterized approximation schema **ApproxFPT** are listed in Fig. 4.

**Theorem 2** *Let* $\rho^\star \simeq 0.21$ *be such that* $1.466 = 1.619^{(1-\rho^\star)}$. *Then, for any* $\rho$ *with* $0 \leqslant \rho \leqslant 1$, ApproxFPT *is a* $(1+\rho)$*-approximation algorithm running in time* $O^*(2.374^{(1-\rho)k})$ *if* $\rho \leqslant \rho^\star$ *and in time* $O^*(1.466^{(2-\rho)k})$ *if* $\rho \geqslant \rho^\star$.

*Proof* In order to prove the running time claimed, we will prove more generally that in an instance $I$ the algorithm works in time:

- $O^*(1.466^{(2-\rho)k-t(I)})$ if $\rho \geqslant \rho^*$;

```
Input: A graph $G = (V = V_1 \cup V_2 \cup V_3, E)$, an integer $k > 0$ and a real number $0 \leqslant \rho \leqslant 1$.
Output: A $(1+\rho)$-approximation solution $M$ to $k$-CONSTRAINED EDGE DOMINATING SET such that $V_1 \subseteq V(M)$.

1. While $t < (2-\rho)k$ and there is a connected component of $V_2$ which is not a 2-path, do a good branching.
2. If $t \geqslant (2-\rho)k$, compute ApproxPoly1($G$).
3. Elseif $\rho \geqslant 1/2$, compute ApproxPoly2($G$).
4. Elseif $t \geqslant (1-\rho)k$, do
    (a) While $t \leqslant 2(1-\rho)k$ and $V_3 \neq \emptyset$, do branch on a 2-path in $G[V_3]$ by including either its central vertex or its two endpoints into $V_1$;
    (b) Compute ApproxPoly2($G$).
5. Elseif $N \geqslant (1-\rho)k$, then compute ApproxPoly2($G$).
6. Elseif $N \leqslant 2(1-\rho)k/3$, branch into $2^N$ branches by considering the $2^N$ subsets of paths. For each subset $S$, include the central vertex of paths in $S$ into $V_1$, include the two endpoints of the paths not in $S$ into $V_1$, and compute an optimal solution (now $V_3 = \emptyset$).
7. Else consider any subset $S$ of the set of the $N$ paths in $G[V_3]$ with size $|S|$ at most $(1-\rho)k - N$. For each such subset $S$, include the two extremities of the paths in $S$ in $V_1$, and compute ApproxPoly2($G$).
8. If an optimal solution among all the leaves in the search tree is of size at most $(1+\rho)k$, then return it. Else report that there is no solution of size at most $k$.
```

**Fig. 4** Algorithm ApproxFPT

- $O^*(1.466^{(1-\rho)k-t(I)}1.619^{(1-\rho)k})$ if $\rho \leqslant \rho^*$.

Then, the result follows since $t(I) \geqslant 0$ and $1.466 \times 1.619 < 2.374$.

Note also that the positive root of $1 = x^{-1} + x^{-3}$ is $1.4655\ldots < 1.466$ and that of $1 = x^{-1} + x^{-2}$ is $1.6180\ldots < 1.619$.

Consider so the different steps of the Algorithm ApproxFPT:

- Step 1: when branching, in one branch $(1-\rho)k - t(I)$ reduces by at least 1, and in the other branch $(1-\rho)k - t(I)$ reduces by at least 3, so the recurrence is verified.
- Steps 2 and 3: the running time is polynomial, which verifies the claim since when we branch we have $t(I) \leqslant (2-\rho)k$ (the validity for $\rho \leqslant \rho^*$ follows from the fact that in this case $1.466 \leqslant 1.619^{(1-\rho)k}$, hence a polynomial is indeed $O^*(1.466^{-k}1.619^{(1-\rho)k})$).
- Step 4: we directly compute the running time in this node. We build a tree where at each branching $t(I)$ increases by 1 in one branch and by two in the other branch. We stop when $t(I)$ reaches $2(1-\rho)k$ (or before if $V_3$ becomes empty). Then, the number of leaves in this tree is (at most) $1.619^{2(1-\rho)k-t(I)}$. For $\rho \geqslant \rho^*$, $1.619^{1-\rho} \leqslant 1.466$; so, the bound on the running time is valid. For $\rho \leqslant \rho^*$, since $t(I) \geqslant (1-\rho)k$, $1.619^{2(1-\rho)k-t(I)} \leqslant 1.619^{(1-\rho)k}1.466^{(1-\rho)k-t(I)}$. Then, the running time in this node verifies the claim.
- Step 5: the running time is polynomial and verifies the claim.
- Step 6: the running time is $O^*(2^{2(1-\rho)k/3}) = O^*(1.6^{(1-\rho)k})$ which verifies the claim since in this step $t(I) \leqslant (1-\rho)k$.
- Step 7: let $z = \sum_{i=0}^{(1-\rho)K-N}\binom{N}{i}$. The running time is $O^*(z)$. Let $P = (1-\rho)k - N$. Then $z = \sum_{i=0}^{P}\binom{(1-\rho)k-P}{i}$. But $P = (1-\rho)k - N \leqslant (1-\rho)k/3$. Then $z \leqslant 1.619^{(1-\rho)k}$ [21]. This verifies the claim since $t(I) \leqslant (1-\rho)k$.

Let us now prove the approximation ratio claimed. Since Algorithms ApproxPoly1$(G)$ and ApproxPoly2$(G)$ will return a constrained edge dominating set, ApproxFPT$(G, k, \rho)$ will also return a constrained edge dominating set.

In order to prove the ratio claimed, we assume that the size $\nu$ of the optimal solution is not greater than $k$ and consider all the possible cases.

- If $t = |V_1| + |V_2| - p \geqslant (2-\rho)k$, then we have $2\alpha_1 + \alpha_2 \geqslant t \geqslant \alpha_1 + \alpha_2 + \alpha_3 + (1-\rho)k$ and, consequently, $\alpha_1 \geqslant (1-\rho)k$. Using Lemma 3, we get $|\text{ApproxPoly1}(G)| \leqslant (1+\rho)k$.
- Assume $\rho \geq 1/2$. If $\alpha_2 > k$, then the size of the optimal solution is greater than $k$. Else, we have that $|\text{ApproxPoly2}(G)| \leqslant k + n_3 \leqslant k + \alpha_2/2 \leqslant 3k/2 \leqslant (1+\rho)k$.
- In Step 4, we consider the two stop conditions $V_3 = \emptyset$ and $t \geqslant 2(1-\rho)k$ in Step 4(b). For the former, we know from Lemma 4 that ApproxPoly2$(G)$ returns an optimal solution. For the latter, in the current graph we have $k + \alpha_1 \geqslant t > 2(1-\rho)k$, and $2\rho k > k - \alpha_1 \geqslant \alpha_2$ (by $k \geqslant \nu \geq \alpha_1 + \alpha_2$). Then $n_3 \leqslant \alpha_2/2 < \rho k$ and Lemma 4 gives $|\text{ApproxPoly2}(G)| \leqslant k + n_3 \leqslant (1+\rho)k$.

- In Step 5, we have $N \geqslant (1-\rho)k$. But $n_1+n_2+2n_3 \leqslant \nu \leqslant k$, meaning that $n_3 \leqslant k-N \leqslant \rho k$. Again, Lemma 4 gives $|\texttt{ApproxPoly2}(G)| \leqslant k+n_3 \leqslant (1+\rho)k$.
- In Step 6, for each branch $V_3 = \emptyset$ and $\texttt{ApproxPoly2}(G)$ returns the optimal solution under the current condition.
- For the last step, let $G$ be the graph before executing any operations in Step 7 and $n_3$ be the size of $N_3$ in $G$. Let $S_1$ be $N_3$ if $|N_3| \leqslant (1-\rho)k - N$, and $S_1$ be a subset of $N_3$ of size $(1-\rho)k - N$ otherwise. Then, $S_1$ is a subset of size at most $(1-\rho)k - N$ that will be considered in Step 7. We only need to show that in the branch where $S_1$ is considered, we can get a feasible solution within the approximation ratio. Let $G'$ be the graph after removing $S_1$ out of $V_3$ (before running $\texttt{ApproxPoly2}(G)$) in Step 7 and $n'$ be the size of $N_3$ in $G'$. By definition, we have that $n_3 + N \leqslant \nu \leqslant k$, and then $n_3 \leqslant k - N$. Note that $n'_3 \leqslant \max(0, n_3 - ((1-\rho)k - N) \leqslant \rho k$. Therefore, $|\texttt{ApproxPoly2}(G')| \leqslant k + n'_3 \leqslant (1+\rho)k$.

The proof of the lemma is now completed. □

## 5 Parametrization by the Vertex Cover Number

Since the size of any vertex cover in a graph is at least the size of any matching in this graph, any parameterized algorithm for EDGE DOMINATING SET working in $O(f(k)|I|^{O(1)})$ time also works in $O(f(\tau)|I|^{O(1)})$ time, where $\tau$ is the size of the minimum vertex cover of the graph. Hence, it is possible to solve EDGE DOMINATING SET within time $O^*(2.3147^\tau)$ by using the algorithm in [21]. In this section we show that this result can be improved down to $O^*\,(1.821^\tau)$.

To this aim, let us consider the algorithm $\texttt{FPT}_\tau$ presented in Fig. 5, which outputs a minimum edge dominating set in graph $G$. Let $\alpha \simeq 0.2864$ be such that $2.3147^\alpha = \left(\frac{1}{\alpha^\alpha(1-\alpha)^{1-\alpha}}\right)$.

**Theorem 3** *$\texttt{FPT}_\tau(G)$ computes a minimum edge dominate set in $O^*(1.821^\tau)$ time.*

*Proof* We first show that the algorithm returns a minimum edge dominating set. If there exists an edge dominating set of size at most $(1-\alpha)\tau(G)$, it will be found in Step 2 of the algorithm. Suppose now that this is not the case, and let $M'$ be an arbitrary maximal matching. Note that each edge of $M'$ has at least one of its endpoint in $V^*$. In other words, $|V^* \cap V(M')| \geqslant |M'|$, meaning that

$$|V^* \setminus V(M')| \leqslant |V^*| - |M'| \leqslant \alpha\tau.$$

Therefore, the set $V^* \setminus V(M')$ is of size at most $\alpha\tau$ and will be considered in Step 3. We look at the case that $V_1 = V^* \setminus V(M')$. Now $V_2 = V^* \cap V(M')$. Moreover, we have the two following properties:

1. $V_2 \cup S_1$ is a vertex cover of $G$;
2. $V_2 \cup S_1$ is included in $V(M')$.

```
Input: A graph $G = (V, E)$.
Output: A minimum edge dominate set.

1. Compute a minimum vertex cover $V^*$ of $G$ by using the algorithm in [7],
   and let $S^* = V \setminus V^*$.
2. For $k = 1$ to $(1-\alpha)\tau$ determine whether there exists an edge dominating set
   of size at most $k$ by using the algorithm in [21]. If any, output the minimum
   edge dominating set and quit.
3. Otherwise, for each subset $V_1$ of $V^*$ of size at most $\alpha\tau$:
   (a) Let $V_2 = V^* \setminus V_1$, $S_1 = N(V_1) \cap S^*$, and $S_2 = S^* \setminus S_1$;
   (b) Compute a maximum matching $M(V_1)$ in $G[V_2 \cup S_1]$;
   (c) For each vertex in $V_2 \cup S_1$ unsaturated by $M(V_1)$, take one edge incident
       to this vertex. Together with $M(V_1)$, this gives a set $M'(V_1)$ of edges.
4. Output a minimum edge dominating set computed in Step 3 (note that some
   of the edge sets $M'(V_1)$ are not edge dominating sets).
```

**Fig. 5** Algorithm $\texttt{FPT}_\tau$

To see the first property, remark that $S_2$ is an independent set. Moreover, $V_1$ is also an independent set (if there were an edge in $V_1$ then, one of its endpoints should be in $V(M')$, a contradiction with the considered case). Finally, there is no edge between a vertex of $V_1$ and a vertex of $S_2$ by definition of $S_1$.

We already know that $V_2 \subseteq V(M')$. Each vertex $v \in S_1$ is adjacent to a vertex $w \in V_1$. Since $w \notin V(M')$, we know that necessarily $v \in V(M')$. The second property follows.

From the first property we deduce that the set $M'(V_1)$ is an edge dominating set. Let $m_1 = |M(V_1)|$. We have $|M'(V_1)| = m_1 + (|V_2| + |S_1| - 2m_1)$. But $M'$ has $m'_1 \leqslant m_1$ edges with both endpoints in $V_2 \cup S_1$. By the second property, $|M'| \geqslant m'_1 + (|V_2| + |S_1| - 2m'_1) \geqslant m_1 + (|V_2| + |S_1| - 2m_1) = |M'(V_1)|$, which implies that $M'(V_1)$ is a minimum edge dominating set.

We now analyze the running time of Algorithm $\texttt{FPT}_\tau$. Step 1 can be done in $O^*(1.2738^\tau)$ time [7]. If an edge dominating set has been found in Step 2, then the running time is:

$$O^*\left(2.3147^{(1-\alpha)\tau}\right) = O^*\left(1.821^\tau\right).$$

Otherwise, by Stirling's Formula, we know that the number of subsets of $V^*$ of size at most $\alpha\tau$ is $O^*\left(\left(\frac{1}{\alpha^\alpha(1-\alpha)^{1-\alpha}}\right)^\tau\right) = O^*(1.821^\tau)$. □

## 6 Conclusion

We provide in this article new insights on the approximability of EDGE DOMINATING SET. Our parameterized approximation algorithm first apply some steps of a branching algorithm, and then exploit the specificity of obtained instances to get an approximate solution on them. This is rather different from the notions of *fidelity preserving transformation* recently introduced in [11] where informally the instance is *first* reduced in an approximate way (and then an (exact) FPT algorithm is applied). In particular, our approximation algorithm relies on the branching steps; this is not the case in the approach of [11] and applying this latter approach for EDGE DOMINATING SET is an interesting open question mentioned in [11]. Moreover, our algorithm has complexity $O^*(\gamma_\rho^k)$ for a ratio $\rho$ where $\gamma_1 = 2.374$ (exact algorithm) and $\gamma_2 = 1.466$. Since achieving a ratio 2 is polynomial, one could hope to find approximation algorithms where $\gamma_\rho \to 1$ when $\rho \to 2$, which we leave as open question.

## References

1. Binkele-Raible, D., Fernau, H.: Enumerate and measure: improving parameter budget management. In: Raman, V., Saurabh, S. (eds.): Proc. International Symposium on Parameterized and Exact Computation, IPEC'10, volume 6478 of Lecture Notes in Computer Science, pp. 38–49. Springer-Verlag (2010)
2. Bourgeois, N., Escoffier, B., Paschos, V.Th.: Approximation of MAX INDEPENDENT SET, MIN VERTEX COVER and related problems by moderately exponential algorithms. Discret. Appl. Math. **159**(17), 1954–1970 (2011)
3. Brankovic, L., Fernau, H.: A novel parameterised approximation algorithm for minimum vertex cover. Theor. Comput. Sci. **511**, 85–108 (2013)
4. Cai, L., Huang, X.: Fixed-parameter approximation: conceptual framework and approximability results. In: Bodlaender, H.L., Langston, M.A. (eds.): Proc. International Workshop on Parameterized and Exact Computation, IWPEC'06, volume 4169 of Lecture Notes in Computer Science, pp. 96–108. Springer-Verlag (2006)
5. Cardinal, J., Langerman, S., Levy, E.: Improved approximation bounds for edge dominating set in dense graphs. Theoret. Comput. Sci. **410**(8-10), 949–957 (2009)
6. Carr, R., Fujito, T., Konjevod, G., Parekh, O.: A $(2+\frac{1}{10})$-approximation algorithm for a generalization of the weighted edge-dominating set problem. J. Comb. Optim. **5**, 317–326 (2001)
7. Chen, J., Kanj, I.A., Xia, G.: Improved upper bounds for vertex cover. Theoret. Comput. Sci. **411** (40-42), 3736–3756 (2010)
8. Chlebik, M., Chlebikova, J.: Approximation hardness of edge dominating set problems. J. Comb. Optim. **11**(3), 279–290 (2006)
9. Dinur, I., Safra, M.: The importance of being biased. Proc. STOC'02, 33–42 (2002)
10. Downey, R.G., Fellows, M.R., McCartin, C., Rosamond, F.A.: Parameterized approximation of dominating set problems. Inform. Process. Lett. **109**(1), 68–70 (2008)
11. Fellows, M.R., Kulik, A., Rosamond, F.A., Shachnai, H.: Parameterized approximation via fidelity preserving transformations. In: Czumaj, A., Mehlhorn, K., Pitts, A., Wattenhofer, R. (eds.): Proc. ICALP'12, volume 7391 of Lecture Notes in Computer Science, pp. 351–362. Springer-Verlag (2012)
12. Fernau, H.: Edge dominating set: efficient enumeration-based exact algorithms. In: Bodlaender, H.L., Langston, M.A. (eds.): Proc. International Workshop on Parameterized and Exact Computation, IWPEC'06, volume 4169 of Lecture Notes in Computer Science, pp. 142–153. Springer-Verlag (2006)
13. Fomin, F.V., Gaspers, S., Saurabh, S., Stepanov, A.A.: On two techniques of combining branching and treewidth. Algorithmica **54**(2), 181–207 (2009)
14. Fujito, T., Nagamochi, H.: A 2-approximation algorithm for the minimum weight edge dominating set problem. Discret. Appl. Math. **118**, 199–207 (2002)
15. Garey, M.R., Johnson, D.S.: Computers and intractability. A guide to the theory of NP-completeness. W.H. Freeman, San Francisco (1979)
16. Khot, S., Regev, O.: Vertex cover might be hard to approximate to within $2-\varepsilon$. J. Comput. System Sci. **74**(3), 335–349 (2008)
17. Marx, D.: Parameterized complexity and approximation algorithms. Comput. J. **51**(1), 60–78 (2008)

18. Raman, V., Saurabh, S., Sikdar, S.: Efficient exact algorithms through enumerating maximal independent sets and other techniques. Theory Comput. Syst. **41**(3), 563–587 (2007)
19. Schmied, R., Viehmann, C.: Approximating edge dominating set in dense graphs. Theoret. Comput. Sci. **414**(1), 92–99 (2012)
20. van Rooij, J.M.M., Bodlaender, H.L.: Exact Algorithms for Edge Domination. Algorithmica **64**(4), 535–563 (2012)
21. Xiao, M., Kloks, T., Poon, S.-H.: New parameterized algorithms for the edge dominating set problem. Theor. Comput. Sci. **511**, 147–158 (2013)
22. Xiao, M., Nagamochi, H.: Parameterized edge dominating set in graphs with degree bounded by 3. Theor. Comput. Sci. **508**, 2–15 (2013)
23. Xiao, M., Nagamochi, H.: A refined exact algorithm for edge dominating set. In: Agrawal, M., Barry Cooper, S., Li, A. (eds.): Proc. Theory and Applications of Models of Computation, TAMC'12, volume 7287 of Lecture Notes in Computer Science, pp. 360–372. Springer-Verlag (2012)
24. Yannakakis, M., Gavril, F.: Edge dominating sets in graphs. SIAM J. App. Math. **38**(3), 364–372 (1980)