On Matchings and b-Edge Dominating Sets: A 2-Approximation Algorithm for the 3-Edge Dominating Set Problem

Toshihiro Fujito⋆

Department of Computer Science and Engineering
Toyohashi University of Technology
Toyohashi 441-8580 Japan
fujito@cs.tut.ac.jp

Abstract. We consider a multiple domination version of the edge dominating set problem, called the *b-EDS* problem, where an edge set $D \subseteq E$ of minimum cardinality is sought in a given graph $G = (V, E)$ with a demand vector $b \in \mathbb{Z}^E$ such that each edge $e \in E$ is required to be dominated by $b(e)$ edges of D. When a solution D is not allowed to be a multi-set, it is called the *simple* b-EDS problem. We present 2-approximation algorithms for the simple b-EDS problem for the cases of $\max_{e \in E} b(e) = 2$ and $\max_{e \in E} b(e) = 3$. The best approximation guarantee previously known for these problems is 8/3 due to Berger et al. [2] who showed the same guarantee to hold even for the minimum cost case and for arbitrarily large b. Our algorithms are designed based on an LP relaxation of the b-EDS problem and locally optimal matchings, and the optimum of b-EDS is related to either the size of such a matching or to the optimal LP value.

1 Introduction

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* (abbreviated to *eds*) if the edges in it collectively dominate all the edges in a graph. The *edge dominating set problem (EDS)* asks to find an eds of minimum cardinality (cardinality case) or of minimum total cost (cost case). It was shown by Yannakakis and Gavril that, although EDS has important applications in areas such as telephone switching networking, it is NP-complete even when graphs are planar or bipartite of maximum degree 3 [12]. The classes of graphs for which its NP-completeness holds were later refined and extended by Horton and Kilakos to planar bipartite graphs, line and total graphs, perfect claw-free graphs, and planar cubic graphs [7], although EDS admits a PTAS (polynomial time approximation scheme) for planar [1] or λ-precision unit disk graphs [8]. Meanwhile, some polynomially solvable special cases have been also discovered for trees [9], claw-free chordal graphs, locally connected claw-free graphs, the line graphs of total graphs, the line

⋆ 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.

R Ravi and I.L. Gørtz (Eds.): SWAT 2014, LNCS 8503, pp. 206–216, 2014.

graphs of chordal graphs [7], bipartite permutation graphs, cotriangulated graphs [11], and so on.

There are various variants of the basic EDS problem, and the most general one among them was introduced by Berger et al. [2] in the form of the *capacitated b-edge dominating set* problem (b, c)-EDS, where an instance consists of a graph $G = (V, E)$, a demand vector $b \in \mathbb{Z}_+^E$, a capacity vector $c \in \mathbb{Z}_+^E$ and a cost vector $w \in \mathbb{Q}_+^E$. A set D of edges in G is called a (b, c)-eds if each $e \in E$ is adjacent to at least $b(e)$ edges in D, where we allow D to contain at most $c(e)$ multiple copies of edge e. The problem asks to find a minimum cost (b, c)-eds. The (b, c)-EDS problem generalizes the EDS problem in much the same way that the set multicover problem generalizes the set cover problem. In the special case when all the capacities c are set to $+\infty$, we call the resulting problem the *uncapacitated* b-EDS problem and its feasible solutions *uncapacitated* b-eds's, whereas it is called the *simple* b-EDS problem when $c(e) = 1$ for all $e \in E$. If $b(e)$'s are set to a same value for all $e \in E$, it is called *uniform* (b, c)-EDS.

Let $b_{\max}$ denote $\max_{e \in E} b_e$. We mainly focus on the simple b-EDS problem (i.e., $(b, 1)$-EDS), and b-EDS (or b-eds) in what follows usually means the simple one unless otherwise stated explicitly. It should be noted, however, that this does not impose serious restrictions in problem solving, as long as $b_{\max}$ is bounded by some constant, since general (b, c)-EDS can be reduced to $(b, 1)$-EDS by introducing $\min\{b_{\max}, c(e)\} - 1$ many copies of e, each of them parallel to e with $b = 0$, for all the edges $e \in E$.

1.1 Previous Work

It was shown by Yannakakis and Gavril that the minimum EDS can be efficiently approximated to within a factor of 2 by computing any maximal matching [12]. They used the theorem of Harary [6] to lower bound the cardinality of a minimum eds by that of a smallest maximal matching. More recently, a 2.1-approximation algorithm first [4], and then 2-approximation algorithms [5,10] have been successfully obtained for the cost case of EDS problem via polyhedral approaches.

Among the approximation results obtained in [2] those relevant to ours are summarized in the following list (note: their results hold even for the cost cases of (b, c)-EDS problems):

- The (b, c)-EDS problem can be approximated within a factor of $8/3$.
- The uniform and uncapacitated b-EDS problem can be approximated within a factor of 2.1 if $b = 1$ or a factor of 2 if $b \geq 2$.
- The integrality gap of the LP relaxation they used for (b, c)-EDS, much more complex one than ours with additional valid inequalities, is at most $8/3$ and it is tight even when $b(e) \in \{0, 1\}$, $\forall e \in E$.

A linear-time 2-approximation algorithm for uncapacitated b-EDS was obtained by Berger and Parekh [3].

1.2 Our Work

One of the main subjects studied in the current paper is the *approximate* min-max relationships between simple b-EDS and locally optimal matchings. A most well-known example of such is perhaps the one between the vertex cover number and the size of a maximal matching. Let $\tau(G)$ denote the vertex cover number of G (i.e., the cardinality of any smallest vertex cover for G) and M be any maximal matching in G. Then, $|M| \le \tau(G) \le 2|M|$.

Let us simplify matters in the following discussion by restricting ourselves to the case of uniform and simple b-EDS, and let $\gamma_b(G)$ denote the cardinality of any smallest $(b,1)$-eds for $G=(V,E)$ where $b(e) \equiv b$, $\forall e \in E$. To introduce lower bounds on $\gamma_b(G)$, we start with the following integer program, the most natural IP formulation for simple b-EDS:

$$\min\{x(E) \mid x(\delta(e)) \ge b(e) \text{ and } x_e \in \{0,1\}, \forall e \in E\},$$

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 $0 \le x_e \le 1$ would result in the following LP:

$$\min\{x(E) \mid x(\delta(e)) \ge b(e) \text{ and } 0 \le x_e \le 1, \forall e \in E\}.$$

Relaxing the LP above further by dropping the upper bound constraint on each x_e, we obtain an LP and its dual in the following forms:

$$\begin{array}{lll} \text{LP: (P)} & \min z_P(x) = x(E) & \\ \text{subject to:} & x(\delta(e)) \ge b(e), & \forall e \in E \\ & x_e \ge 0, & \forall e \in E \end{array} \qquad \begin{array}{lll} \text{LP: (D)} & \max z_D(y) = \sum_{e\in E} b(e) y_e & \\ \text{subject to:} & y(\delta(e)) \le 1, & \forall e \in E \\ & y_e \ge 0, & \forall e \in E \end{array}$$

Notice here that (P) coincides with the LP relaxation of uncapacitated b-EDS rather than simple one.

For any matching M in $G=(V,E)$, let $y^M \in \mathbb{R}^E$ be a vector of dual variables such that

$$y_e^M = \begin{cases} \frac{1}{2} & \text{if } e \in M \\ 0 & \text{otherwise} \end{cases}$$

As $\delta(e)$ contains at most two edges of M for any $e \in E$, y^M is always feasible for (D), and its value $z_D(y)$ equals to $b|M|/2$. So, $b|M|/2$ can serve as a lower bound on $\gamma_b(G)$ for any matching M in G.

Following the way locally optimal solutions are often termed in the local search optimization, we say that a matching M in G is *k-opt* if, for any matching N in G larger than M, $|M \setminus N| \ge k$ (So, a *maximal* matching is 1-opt). As for upper bounds on $\gamma_b(G)$, $|M_1|$ provides itself as such a bound on $\gamma_1(G)$ for any 1-opt matching M_1. It is also the case, as will be shown later (Corollary 1), that $\gamma_2(G) \le 2|M_2|$ for any 2-opt matching M_2 in G. So it would be extremely pleasing if the following min-max relationships hold for all $b \in \mathbb{N}$:

$$\frac{b|M_b|}{2} \le \gamma_b(G) \le b|M_b|,$$

where M_b is any b-opt matching in G. It is, however, too good to be true, and it will be shown (in Section 4) that $\gamma_b(G)$ cannot be bounded above by $b|M_b|$ in general when $b \ge 3$.

Nevertheless, $\gamma_3(G)$ can be related to a stronger bound as follows. Letting dual(G) denote the optimal value of LP:(D) above for graph G, it will be seen (in Corollary 2) that the following min-max relation to hold:

$$\mathrm{dual}(G) \leq \gamma_3(G) \leq 2 \cdot \mathrm{dual}(G)$$

for any G (recall that dual(G) is the optimal value of the LP relaxation for *uncapacitated* b-EDS).

These upper bounds are obtained algorithmically; our algorithms, building solutions upon b-opt matchings, approximate the (*unweighted*) simple b-EDS problems within a factor of 2, where b is not assumed to be uniform, for $b_{\max} = 2$ and for $b_{\max} = 3$. Unlike the polyhedral approaches explored in [2], ours is more graph theoretic and our algorithms are purely combinatorial.

2 Preliminaries

In this paper only graphs with no loops are considered. For an edge set $F \subseteq E$, $V[F]$ denotes the set of vertices induced by the edges in F (i.e., the set of all the endvertices of the edges of F). For a vertex set $S \subseteq V$ let $\delta(S)$ denote the set of edges incident to a vertex 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 vertices; 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 vertex set $U \subseteq V$, $N(U)$ denotes the set of neighboring vertices 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 $N(\{u\})$. The degree of a vertex u is denoted by $d(u)$. When $\delta(S), N(U)$, and $d(u)$ are considered only within a subgraph H of G (or when restricted to within a vertex subset or edge subset T), they are denoted by $\delta_H(S), N_H(U)$, and $d_H(u)$ (or $\delta_T(S), N_T(U)$, and $d_T(u)$), respectively.

When an edge e is dominated by up to $b(e)$ edges, it is said to be *fully dominated*.

3 A 2-Opt Algorithm for 2-EDS

Here a 2-approximation algorithm for the simple 2-EDS problem is presented. The algorithm is quite simple: Compute a 2-opt matching M_2 so that no augmenting path of length 3 or shorter occurs. Then, for each matched edge $e \in M_2$, if one of its endvertices is a neighbor of an exposed vertex via edge e', add e' besides e itself to a solution set while, if neither has, add any edge adjacent to e.

Divide the edge set E of an instance graph G according to demands into E_1 and E_2, where $E_i = \{e \in E \mid b(e) = i\}$. Let $G_i = (V_i, E_i)$ denote the subgraph of G induced by E_i.

1. Compute a 2-opt matching M_2 in $G_2 = (V_2, E_2)$; so no augmenting paths of length 3 or shorter occurs.
2. $D_2 \leftarrow M_2$.

Let $X \subseteq V_2$ denote the set of vertices in G_2 exposed by M_2, and consider $N_{G_2}(u)$ and $N_{G_2}(v)$ for each $e = \{u, v\} \in M_2$. If both of them contain vertices exposed by M_2, they must be same and unique, i.e., $N_{G_2}(u) \cap X = N_{G_2}(v) \cap X = \{x\}$ for some $x \in X$; otherwise, an augmenting path of length 3 having e in the middle is found. One edge adjacent to e is added to D_2, and which one to add is determined according to which of $N_{G_2}(u)$ and $N_{G_2}(v)$ contains an exposed vertex.

3. For each $e = \{u, v\} \in M_2$,
   (a) if $N_{G_2}(u)$ contains an exposed vertex $x \in X$, then add edge $\{u, x\}$ into D_2,
   (b) else if $N_{G_2}(v)$ contains an exposed vertex, then add any edge in $\delta_G(v)$ (other than e) into D_2,
   (c) else add any edge in $\delta_G(e)$ (other than e) into D_2.

Note: If it is only to dominate twice the edges in $\delta_{G_2}(u)$, we may choose any one of them in Step 3(a). It could be the case, however, that both of $N_{G_2}(u)$ and $N_{G_2}(v)$ contain the exposed vertex x as a unique exposed vertex in common, and it is then necessary to choose $\{u, x\}$ in this step to fully dominate $\{v, x\}$.

By this time all the edges in E_2 are fully dominated. It remains only to dominate those in E_1 that are not yet dominated even once.

4. Set $E'_1 \leftarrow E_1 \setminus \delta_G(D_2)$.
5. Compute a 1-opt matching M_1 in $G[E'_1]$ and output $D_2 \cup M_1$.

Theorem 1. *The* 2*-opt algorithm given above is a* 2*-approximation algorithm for the* $(b, 1)$*-EDS problem when* $b_{\max} = 2$.

Proof. Consider an edge $e \in E_2$. It becomes fully dominated, if $e \in M_2$, when an edge adjacent to e is added to D_2 in step 3. For $e \notin M_2$, if both of its endvertices are matched by M_2, it is made dominated fully by M_2 (in step 2). If $e = \{u, x\} \notin M_2$ is incident to an exposed vertex x, another unmatched edge incident to either u or x must be chosen into D_2 in step 3. Therefore, all the edges in E_2 become fully dominated after step 3. As not-yet-dominated edges in E_1 are taken care of in step 5 when a maximal matching M_1 is entirely chosen into a solution, the algorithm computes a simple 2-eds for G.

The performance analysis of this algorithm is omitted here as it can be subsumed by the one for the 3-opt algorithm for 3-EDS presented in Section 5. □

In case when $b(e)$ is uniformly equal to 2 in the above, $G_2 = G$ and $|D_2| \leq 2|M_2|$. Thus,

Corollary 1. *For any* 2*-opt matching* M_2 *in* G*,* $\gamma_2(G) \leq 2\,|M_2|$.

4 b-Opt Matchings and $\gamma_b(G)$

4.1 Case of 3-EDS

This subsection shows that the ratio of $\gamma_b(G)$ to $b\,|M_b|$ for a b-opt matching M_b in G can be larger than 1 even for $b = 3$.

Let $P_4 = \{e_{i,1}e_{i,2}e_{i,3}e_{i,4} \mid 1 \leq i \leq k\}$ be a collection of simple paths of length 4, starting and ending at the common vertices u_1 and u_2 respectively, and being mutually vertex disjoint except at these two vertices. Construct a graph G by attaching two edges $e_{0,1}$ and $e_{0,2}$ at u_1 and u_2, respectively, but disjointly at the other endvertices of $e_{0,1}$ and $e_{0,2}$ from other vertices in G. Let M be a matching in G such that $M = \{e_{0,1}, e_{0,2}, e_{i,3} \mid 1 \le i \le k\}$. Then, M is a maximum matching in G with $|M| = k+2$ as there exists no augmenting path w.r.t. M. Meanwhile, $\gamma_3(G) = 4k$ since all the edges in all the paths of P_4 must be used to constitute a 3-eds for G, and hence, $\frac{\gamma_3(G)}{3|M|} = \frac{4k}{3(k+2)} > 1$ for $k > 6$ even if M is a maximum matching in G.

4.2 Case of b-EDS

This subsection shows that the ratio of $\gamma_b(G)$ to $b\,|M_b|$ for a b-opt matching M_b in G can be arbitrarily large as b grows.

Let S_i be a star graph centered at vertex s_i with $b/2$ edges, for $1 \le i \le b/2$ ("small" stars). Let L_i also be a star graph centered at vertex l_i with $(b/2)^2$ edges, for $1 \le i \le b/2$ ("large" stars). Construct a bipartite graph G with all the center vertices in S_i's and L_i's on one side, and a set U of $(b/2)^2$ vertices on the other side, by attaching leaves of S_i's and L_i's at the vertices in U. Each leaf of L_i is attached to a distinct vertex of U for $1 \le i \le b/2$. There are $(b/2)^2$ leaves of S_i's in total, and they are also attached to the vertices of U distinctively.

Observe now that $d(s_i) = b/2, d(l_i) = (b/2)^2$ for $1 \le i \le b/2$, $d(u) = b/2 + 1$ for $u \in U$, and $|\delta(e)| = b/2 + b/2 = b$ for any edge e in $\delta(s_i)$. Therefore, any b-eds for $G = (V, E)$ must contain all of $\delta(e)$'s for all $e \in \delta(s_i)$, covering all the edges of G, and meaning that $\gamma_b(G) = |E| = (b/2)^2(b/2+1)$. On the other hand, there exist $b/2+b/2 = b$ vertices on the other side of U, and hence, $|M| \le b$ for any matching M in G. It thus follows that

$$\frac{\gamma_b(G)}{b\,|M|} \ge \frac{(b/2)^2(b/2+1)}{b^2} = \frac{b+2}{8}$$

even if M is a maximum matching in G.

5 A 3-Opt Algorithm for 3-EDS

Here a 2-approximation algorithm for the simple 3-EDS problem is presented. In the beginning the algorithm dominates all the edges with demands of 3 using a 3-opt matching M_3. As was observed in the previous section, however, it is not good enough to choose the M_3-edges along with some edges adjacent to them. Moreover, as we treat the case when edges with demands of 2 or less are allowed to coexist, a part of the solution dominating those demand-3 edges may interfere with another part dominating those with smaller demands, and it makes the task of designing an algorithm more complicated than otherwise.

Divide the edge set E of an instance graph G according to demands into E_1, E_2, and E_3, where $E_i = \{e \in E \mid b(e) = i\}$. Let $G_i = (V_i, E_i)$ denote the subgraph of G induced by E_i.

1. Compute a 3-opt matching M_3 in $G_3 = (V_3, E_3)$; so no augmenting paths of length 5 or shorter occurs.
2. $D_3 \leftarrow M_3$.

Note: At this point an edge in M_3 is dominated once, and one in $E_3 \setminus M_3$ is also dominated once if it is incident to an exposed vertex but otherwise, it is dominated twice by $D_3 = M_3$.

We need to exercise special care in handling those edges incident to the vertices exposed by M_3, and a bipartite subgraph of G_3 induced by those edges is constructed for that purpose as follows; this will be the main body of algorithmic operations and analysis provided later.

- Let $X \subseteq V_3$ be the set of vertices in G_3 exposed by M_3. Notice that X is an independent set in G_3 since M_3 does not allow an augmenting path of length 1 to exist.
- Let $A \subseteq V_3$ be the set of neighboring vertices of X in G_3, i.e., $A = N_{G_3}(X)$.
- Let $B = (X \cup A, E_B)$ denote the bipartite subgraph graph of G_3, consisting of the vertex partition (X, A), and the set E_B of E_3-edges lying between them.
- Let $M' \subseteq M_3$ be the set of matched edges having an endvertex in A, i.e., $M' = \{e \in M_3 \mid e \cap A \neq \emptyset\}$.
- Let $M_c = \{e \in M' \mid e \subseteq A\}$. Then, each edge in $M' \setminus M_c$ has exactly one of its endvertices in A; denote it by $a(e)$ and the other endvertex of e by $\bar{a}(e)$, for each $e \in M' \setminus M_c$.
- Divide $M' \setminus M_c$ further, according to the G-degree of $a(e)$, into $M_s = \{e \in M' \setminus M_c \mid d_G(a(e)) = 2\}$, and $M_d = \{e \in M' \setminus M_c \mid d_G(a(e)) \geq 3\}$.
- Accordingly divide A into $A_c = \{\text{both endvertices of } e \mid e \in M_c\}$, $A_s = \{a(e) \mid e \in M_s\}$, and $A_d = \{a(e) \mid e \in M_d\}$, and E_B into $E_c = \delta_B(A_c), E_s = \delta_B(A_s)$, and $E_d = \delta_B(A_d)$.

Clearly, $d_B(a) \geq 1$ for all $a \in A$. Observe that $d_B(a(e)) = 1$ for all $a(e) \in A_s$ since only two edges are incident to $a(e)$ in G and they are $\{x, a(e)\}$ for some $x \in X$ and $\{a(e), \bar{a}(e)\} \in M_s$ where $\bar{a}(e) \notin A$. It is also the case that $d_B(a) = 1$ for all $a \in A_c$: Consider a pair of vertices, a_1, a_2, in A_c such that they are the endvertices of one edge in M_c. Then, they must be adjacent to a unique and same vertex in X as otherwise, an augmenting path of length 3 would result.

Consider the subgraph $B_d = (X_d \cup A_d, E_d)$ of B induced by E_d, where $X_d = N_B(A_d) \subseteq X$.

3. Compute a *maximal* edge subset N of E_d in B_d such that $|\delta_B(x) \cap N| \leq 2$ for each $x \in X_d$ and $|\delta_B(a) \cap N| \leq 1$ at the same time for each $a \in A_d$.
4. Set $D_3 \leftarrow D_3 \cup E_c \cup E_s \cup N$.

At this point, every edge in E_c (and those in M_c) is fully dominated by $E_c \cup M_c \subseteq D_3$, and there could be such edges also in $E_s \cup E_d$. Let $\tilde{E}_s$ and $\tilde{E}_d$ denote the subsets of E_s and E_d, respectively, consisting of edges (of E_s and E_d) not-yet fully dominated by $M_3 \cup E_c \cup E_s \cup N$. As each edge in E_s is dominated at least twice by $M_3 \cup E_s$, if it is adjacent to any other in $E_c \cup E_s \cup N$, it must be fully dominated. Therefore, $\tilde{E}_s$ forms a matching in G, and no edge in $\tilde{E}_s$ is adjacent to any in $N \cup E_c$.

In the next two steps, all the edges in $E_s \cup M_s$ will be made fully dominated.

5. For $e \in \tilde{E}_s$, add one edge from E incident to the exposed endvertex of e occurring in X, into D_3; such an edge must exist as otherwise, e cannot be fully dominated.
6. For each $e \in M_s$, add any edge in $\delta_G(\bar{a}(e))$ (such an edge must exist in $\delta_G(\bar{a}(e))$ as, otherwise, e cannot be fully dominated).

Suppose, for some $a \in A_d$, $\delta_B(a) \cap N = \emptyset$. Then, every edge in $\delta_B(a)$ must be fully dominated by $N \cup M_d$; if $e \in \delta_B(a)$ is not, it can be added to N contradicting the maximality of N. Therefore, every edge in $\tilde{E}_d$ must be either in N or adjacent to an N-edge, implying that it is dominated at least twice by $N \cup M_3$.

In the next two steps, the algorithm adds edges to D_3 so that all the edges in $E_d \cup M_d$ become fully dominated. Let $M_N \subseteq M_d$ denote the set of M_d-edges adjacent to an edge in N.

7. For each $e \in M_N$, if $\delta_B(a(e))$ contains an edge in $\tilde{E}_d$, add one more edge from $\delta_G(a(e)) \setminus N$ into D_3, which must exist as $d_G(a(e)) \geq 3$. If $\delta_B(a(e))$ contains no edge in $\tilde{E}_d$, add any edge in $\delta_G(\bar{a}(e))$ if it exists (if it doesn't, add instead any edge in $\delta_G(a(e))$), into D_3.
8. For each $e \in M_d \setminus M_N$, add two edges into D_3, one from $\delta_G(a(e))$ and another from $\delta_G(\bar{a}(e))$ if it exists (if it doesn't, add instead any edge in $\delta_G(a(e))$).

We also need to take care of the edges in $M_3 \setminus M'$ and those around them, and two adjacent edges are added in a simple way for each of these matched edges.

9. For each $e \in M_3 \setminus M'$, add any edge in E incident to the endvertices of e, one each, into D_3; in case when either of them does not exist take two edges, instead of one, from the other end of e.

Finally, all the remaining edges in $E \setminus E_3$ are taken care of by simply running the 2-opt algorithm for 2-EDS on G after the demands are appropriately adjusted. Let E'_i denote the set of edges with demands of i adjusted right after step 9. Then, $E'_3 = \emptyset$ and $E_3 \subseteq E'_0$ since any edge in E_3 has been fully dominated by now (Lemma 1). Moreover, $E'_2 = E_2 \setminus \delta_{E_2}(D_3)$, $E'_1 \subseteq (E_1 \setminus \delta_{E_1}(D_3)) \cup (E_2 \cap \delta_{E_2}(D_3))$ and $E'_0 \subseteq (E_1 \cap \delta_{E_1}(D_3)) \cup (E_2 \cap \delta_{E_2}(D_3)) \cup E_3$.

10. Run the 2-opt algorithm for 2-EDS on G after setting $b'(e) \leftarrow \max\{0, b(e) - |\delta_G(e) \cap D_3|\}$ for $e \in E$, and compute a 2-eds $D_2 \cup M_1$ for (G, b').
11. Output $D_3 \cup D_2 \cup M_1$.

Lemma 1. *Every edge in E_3 becomes fully dominated after step 9.*

Proof. 1. Consider $e \in M' \cup E_B$. Since $M' \cup E_c \cup E_s \subseteq D_3$, $e \in M_c \cup E_c$ is fully dominated whereas $e \in M_s \cup E_s$ is at least twice dominated by the end of step 4, and if not yet fully dominated, e is made so in steps 5 and 6. Any edge in E_d becomes fully dominated by the end of step 7, while any edge in M_d does by the end of step 8.

2. Any edge $e \in M_3 \setminus M'$ is made fully dominated in step 9.
3. Consider $e \in E_3 \setminus (M_3 \cup E_B)$. Both endvertices of e are matched by M_3 ensuring that e is twice dominated by M_3. Observe now that for any matched vertex $u \in V_3 \setminus X$,

$\delta_G(u)$ contains only one edge in D_3 (namely, the matched edge incident to u) only if $d_G(u) = 1$ or $u = \bar{a}(e)$ for some $e \in M_d$ with an N-edge incident to $a(e)$. For any other matched vertex $u \in V_3 \setminus X$, $\delta_G(u) \setminus M_3$ contains at least one D_3-edge, and hence, if either endvertex of e is such a vertex, e is fully dominated.

The case that $d_G(u) = 1$ at an endvertex of e is excluded since e is unmatched. What remains is the case when $u = \bar{a}(e_1)$ and $v = \bar{a}(e_2)$ for $e = \{u, v\}$ such that both e_1 and e_2 are in M_d and each of $a(e_1)$ and $a(e_2)$ has an N-edge incident to it. Since no augmenting path of length 5 exists in G_3, $\{e, e_1, e_2\}$ together with those two N-edges incident to $a(e_1)$ and $a(e_2)$ must form a blossom (of length 5). There cannot exist another edge in E_B incident to either $a(e_1)$ or $a(e_2)$ as it would imply an augmenting path of length 5. So, each of $a(e_1)$ and $a(e_2)$ has only one incident edge in B, and both of them are N-edges having a common exposed vertex at their endvertices. It means, however, that those N-edges are fully dominated even *before* step 7, and hence, $\delta_B(a(e))$ contains no edge in $\tilde{E}_d$ when step 7 is executed. Therefore, an unmatched edge in $\delta_G(u)$ or $\delta_G(v)$ is added to D_3 in step 7, ensuring e being fully dominated. □

Thus, the correctness of the algorithm above follows from this lemma and the correctness of the 2-opt algorithm for 2-EDS:

Theorem 2. *The* 3*-opt algorithm for the* $(b, 1)$*-EDS problem given above computes a feasible* 3*-eds for* G *when* $b_{\max} = 3$.

5.1 Performance Analysis of 3-opt Algorithm for 3-EDS

Recall the dual of our LP relaxation for b-EDS:

$$\begin{aligned} \text{LP: (D)} \quad & \max z_D(y) = \sum_{e \in E} b(e) y_e \\ \text{subject to:} \quad & y(\delta(e)) \le 1, \quad \forall e \in E \\ & y_e \ge 0, \quad \forall e \in E \end{aligned}$$

Suppose $M_3 \subseteq E_3$ is a matching computed in step 1 of the 3-opt algorithm. Recall that $\tilde{E}_s \subseteq E_s$ forms a matching in G, and no edge in $\tilde{E}_s$ is adjacent to any in $N \cup E_c$. Set the value of y_e for $e \in E_3$ as follows:

$$y_e = \begin{cases} \frac{1}{2} & \text{if } e \in M_3 \setminus M_N \\ \frac{1}{4} & \text{if } e \in N \cup M_N \\ \frac{1}{6} & \text{if } e \in \tilde{E}_s \\ 0 & \text{otherwise} \end{cases}$$

Let M_2 and M_1 denote the matchings computed, within the run of the 2-opt algorithm for 2-EDS, in step 10 of the 3-opt algorithm. Recall E'_i, the set of edges with demands of i adjusted right after step 9. Note that $M_2 \subseteq E'_2 = E_2 \setminus \delta_{E_2}(D_3)$ and hence, M_2 contains edges with $b(e) = 2$ only, and no edge in M_2 can be adjacent to any in D_3. The set E'_1 on the other hand may contain E_2-edges e as $b(e)$ could have been lowered to 1 if it is dominated *once* by D_3, and so may $M_1 \subseteq E'_1 \setminus \delta_G(D_2)$.

Set the value of y_e for $e \in E_2 \cup E_1$ as follows:

$$y_e = \begin{cases} \frac{1}{2} & \text{if } e \in M_2 \\ \frac{1}{2} & \text{if } e \in M_1 \cap E_1 \\ \frac{1}{4} & \text{if } e \in M_1 \cap E_2 \\ 0 & \text{otherwise} \end{cases}$$

Lemma 2. *The vector $y \in \mathbb{R}^E$ of dual variables with its values assigned as above is feasible in LP:(D).*

Proof. The dual feasibility of y follows easily if $y(\delta_G(u)) \leq 1/2$ for all $u \in V$. Although this does not hold for all the vertices in G, we will check how large $y(\delta_G(u))$ could be depending on where u is located, and will consider the cases when it exceeds $1/2$ in what follows.

As stated above, $V[D_3] \cap V[M_2] = \emptyset$ and $V[D_2] \cap V[M_1] = \emptyset$, but $V[D_3]$ and $V[M_1]$ are not necessarily disjoint. So, if $u \in V[D_2]$, $y(\delta_G(u)) = y(\delta_{D_2}(u)) = y(\delta_{M_2}(u))$, and hence, $y(\delta_G(u)) \leq 1/2$ in this case.

Suppose $u \in V[D_3] \cap V[M_1]$. Then, the M_1-edge e in $\delta_G(u)$ must come from E_2 and it has to be dominated exactly once by D_3. Consider now for which vertex u of $V[D_3]$ we may have 1) exactly one edge of D_3 is incident to u, 2) the edge in 1) carries a positive dual, and 2) $d_G(u) \geq 2$. It can be verified that such u must be either $\bar{a}(e)$ for $e \in M_N$, or the exposed endvertex of an N edge. In either case the positive dual carried by a D_3-edge is $1/4$, while the one carried by an M_1-edge is also $1/4$; hence, $y(\delta_G(u)) \leq 1/4 + 1/4 = 1/2$ in this case.

If $u \in V[M_1] \setminus V[D_3]$, the M_1-edge contained in $\delta_G(u)$ must come from E_1, and hence, $y(\delta_G(u)) = 1/2$.

What remains is the case when $u \in V[D_3] \setminus V[M_1]$. As observed in passing within the algorithm description, $\tilde{E}_s$ forms a matching and no edge in it is adjacent to any in N. It can be verified from such observations that $\delta_{E_3}(e)$ contains at most two edges with positive dual values for any $e \in D_3$, and those two are either one each from $\tilde{E}_s$ and M_s, one each from M_N and N, or both from N. Among these $y(\delta_{E_3}(u)) = 1/2 + 1/6 > 1/2$ in the first case only, and $y(\delta_{E_3}(u)) \leq 1/2$ in the remaining cases. In the first case, however, $d_G(u) = 2$ and there is no edges incident to u other than those two edges, $e_1 \in \tilde{E}_s$ and $e_2 \in M_s$. Moreover, letting u_1 and u_2 be the other endvertices of e_1 and e_2, respectively, the algorithm adds an edge, with no positive dual, incident to each of u_1 and u_2 into D_3 resulting in $d_{D_3}(u_1) = d_{D_3}(u_2) = 2$. Hence, no more edge can be added to either of u_1 or u_2 in step 10, and each of $y(\delta_G(e_1))$ and $y(\delta_G(e_2))$ remains no larger than 1 in the end.

Therefore, we may conclude that $y(\delta_G(e)) \leq 1$ for all $e \in E$. □

Lemma 3. *For $y \in \mathbb{R}^E$ of dual variables with its values assigned as above, the 3-opt algorithm computes an output of size no larger than twice the objective value of y in LP:(D), i.e.,*

$$|D_3 \cup D_2 \cup M_1| \leq 2z_D(y) = 2\sum_{e \in E} b(e)y_e.$$

Proof. The term in the objective function of LP:(D) corresponding to y_e is $b(e)y_e$. So if at most $2b(e)y_e$ edges are used per e in dominating all the edges, the claimed inequality holds. For $e \in M_2 \cup (M_1 \cap E_1)$, y_e is set to $1/2$, and 2 edges per $e \in M_2$ and 1 edge per $e \in M_1 \cap E_1$ are used. On the other hand, 1 edge per $e \in M_1 \cap E_2$ is used with $y_e = 1/4$, and it suffices because $2b(e)y_e = 2 \cdot 2 \cdot (1/4) = 1$.

As for D_3, 3 edges are used per $e \in M_3$ where $y_e = 1/2$ if $e \in M_3 \setminus M_N$ but $y_e = 1/4$ if $e \in M_N$. For each $e_1 \in M_N$, however, there exists a mate $e_2 \in N$ of its own, carrying $y_{e_2} = 1/4$, and hence, together with e_2, e_1 can pay $1/2$ that is sufficient for 3 edges.

Besides e and two edges adjacent to e per $e \in M_3$, D_3 uses one more edge per $e' \in \tilde{E}_s$, and it can be paid for by $y_{e'}$ as $2b(e')y_{e'} = 2 \cdot 3 \cdot (1/6) = 1$. □

It follows immediately from these preceding two lemmas that the 3-opt algorithm computes a feasible eds of size no larger than twice the optimum:

Theorem 3. *The* 3*-opt algorithm is a* 2*-approximation algorithm for the* $(b, 1)$*-EDS problem when* $b_{\max} = 3$.

Corollary 2. $\gamma_3(G) \leq 2 \cdot dual(G)$.

References

1. Baker, B.S.: Approximation algorithms for NP-complete problems on planar graphs. J. ACM 41, 153–180 (1994)
2. Berger, A., Fukunaga, T., Nagamochi, H., Parekh, O.: Approximability of the capacitated b-edge dominating set problem. Theoret. Comput. Sci. 385(1-3), 202–213 (2007)
3. Berger, A., Parekh, O.: Linear time algorithms for generalized edge dominating set problems. Algorithmica 50(2), 244–254 (2008)
4. 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. Journal of Combinatorial Optimization 5(3), 317–326 (2001)
5. Fujito, T., Nagamochi, H.: A 2-approximation algorithm for the minimum weight edge dominating set problem. Discrete Appl. Math. 118, 199–207 (2002)
6. Harary, F.: Graph Theory. Addison-Wesley, Reading (1969)
7. Horton, J.D., Kilakos, K.: Minimum edge dominating sets. SIAM J. Discrete Math. 6(3), 375–387 (1993)
8. Hunt III, H.B., Marathe, M.V., Radhakrishnan, V., Ravi, S.S., Rosenkrantz, D.J., Stearns, R.E.: A unified approach to approximation schemes for NP- and PSPACE-hard problems for geometric graphs. In: van Leeuwen, J. (ed.) ESA 1994. LNCS, vol. 855, pp. 424–435. Springer, Heidelberg (1994)
9. Mitchell, S., Hedetniemi, S.: Edge domination in trees. In: Proc. 8th Southeastern Conf. on Combinatorics, Graph Theory, and Computing, pp. 489–509 (1977)
10. Parekh, O.: Edge dominating and hypomatchable sets. In: Proc. 13th SODA, pp. 287–291 (2002)
11. Srinivasan, A., Madhukar, K., Nagavamsi, P., Pandu Rangan, C., Chang, M.-S.: Edge domination on bipartite permutation graphs and cotriangulated graphs. Inform. Process. Lett. 56, 165–171 (1995)
12. Yannakakis, M., Gavril, F.: Edge dominating sets in graphs. SIAM J. Appl. Math. 38(3), 364–372 (1980)