Augmenting Forests to Meet Odd Diameter Requirements ⋆

Toshimasa Ishii [a,*], Shigeyuki Yamamoto [b], Hiroshi Nagamochi [c]

[a] *Department of Information and Computer Sciences, Toyohashi University of Technology, Aichi 441-8580, Japan*

[b] *I FOR COM Co.,Ltd. Kanagawa 220-0207, Japan*

[c] *Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan*

Abstract

Given a graph $G = (V, E)$ and an integer $D \geq 1$, we consider the problem of augmenting G by the smallest number of new edges so that the diameter becomes at most D. It is known that no constant approximation algorithms to this problem with an arbitrary graph G can be obtained unless $P = NP$. For a forest G and an odd $D \geq 3$, it was open whether the problem is approximable within a constant factor. In this paper, we give the first constant factor approximation algorithm to the problem with a forest G and an odd D; our algorithm delivers an 8-approximate solution in $O(|V|^3)$ time. We also show that a 4-approximate solution to the problem with a forest G and an odd D can be obtained in linear time if the augmented graph is additionally required to be biconnected.

Key words: undirected graph, graph augmentation problem, diameter, forest, approximation algorithm

⋆ An extended abstract of this paper will be presented at 14th International Symp. on Algorithms and Computation (ISAAC'2003), Kyoto, December 2003.

* Corresponding author.

Email addresses: `ishii@ics.tut.ac.jp` (Toshimasa Ishii), `nag@amp.i.kyoto-u.ac.jp` (Hiroshi Nagamochi).

1 Introduction

In communication networks, some transfer delay occurs when we send a message from one node to another node. The least number of links through which the message has to be transmitted is considered as one measurement of such a transfer delay. Therefore, it is desirable that a network has a small diameter, which is defined as the maximum distance between every two nodes in the network. In [1,6,7,9,13,15], the problems of constructing a graph with a small diameter by adding new edges to an initial graph have been studied, as one of the network design problems such as airplane flights scheduling [7].

Given an undirected graph $G = (V, E)$ and a nonnegative integer D, the *augmentation problem with diameter requirements* (for short, *APD*) is to augment G by adding the smallest number of new edges that reduces the diameter to at most D. Note that the case of $D = 1$ is trivial, because only the complete graph can have diameter one. In general, Schoone et al. [15] have shown that APD is NP-hard for any fixed $D \geq 3$. Moreover, it has been shown that there is no constant approximation algorithm to APD unless P=NP, by a reduction from DOMINATING SET due to Li et al. [13] for $D \geq 4$, and by a reduction from SET COVER due to Dodis and Khanna [8] for $D \in \{2, 3\}$. The same results have been shown by Chepoi and Vaxes [6]. Let $OPT_A(G, D)$ denote the optimal value to APD with a graph G and an integer D. Alon et al. [1] have shown that $OPT_A(G, 2) = n - \Delta - 1$ and $OPT_A(G, 3) \geq n - O(\Delta^3)$ hold for any graph G with the maximum degree Δ and a sufficiently large number $n = |V|$ of vertices and that $OPT_A(G, D) \leq n/\lfloor D/2 \rfloor$ holds for any connected graph G. Also for APD with some restricted classes of graphs, several problems have been studied. Erdős et al. [9] have investigated upper and lower bounds on the optimal value to APD in the case where a given graph and an augmented graph are restricted to be triangle-free. Alon et al. [1] have proved that $OPT_A(C_n, D) = n/(2\lfloor D/2 \rfloor - 1) - O(1)$ holds for any cycle C_n of n vertices. Recently Chepoi and Vaxes [6] have presented a 2-approximation algorithm to APD with a forest and an even integer D. They have also proved that their algorithm can be applied to a wider class of graphs G satisfying the following conditions (i) and (ii). (i) G is a Helly graph (see [2] for the definition). (ii) There exists a polynomial time algorithm to the k-DOMINATING SET with G, the problem of finding a smallest set X of vertices such that the distance from each vertex to some vertex in X is at most k. Forests and dually chordal graphs (see [3] for the definition) are included in such a class of graphs. However, it was left open whether APD with an odd diameter is approximable by a constant factor or not, even if G is a forest, while it is also left open whether APD with a forest is NP-hard or not.

As a related problem, we consider APD with an additional requirement that the resulting augmented graph is a biconnected graph, i.e., it has at least two vertex-disjoint paths between every two vertices. This problem is called the *biconnectivity augmentation problem with diameter requirements* (for short, *BAPD*). In communication networks, graph connectivity can be considered as a fundamental measure of its robustness. Eswaran and Tarjan [10] have shown that the problem of augmenting an initial graph up to biconnectivity can be solved in linear time. For graph connectivities augmentation problems, many problems and algorithms have been studied (see [11,14] for surveys). Chepoi and Vaxes [6] have proved that BAPD is NP-hard even if G is a tree.

Definition 1 *Let us call a solution* (a, b)-approximate solution *if the number of edges in the solution is at most b surplus edges over a times the optimal, and an algorithm that delivers such a solution an* (a, b)-approximation algorithm. □

Chepoi and Vaxes have also given a 3-approximation algorithm for an even integer D [6], and a $(7, 3)$-approximation (resp., $(9, 4)$-approximation) algorithm for an odd $D \geq 5$ (resp., $D = 3$) [5], in the case where G is a forest.

In this paper, we consider designing an approximation algorithm to APD and BAPD, in the case where an initial graph G is a forest and D is an odd integer. We partly follow Chepoi and Vaxes' approaches [6] to obtain a 2-approximate solution to APD with forest G and the even integer $D+1$, which is a relaxation of the original problem instance (G, D). Unfortunately, it is not difficult to see that $OPT_A(G, D)/OPT_A(G, D + 1)$ cannot be bounded from above by any constant even in the case of trees. We establish a new lower bound on the optimal value to APD with an odd D. With the 2-approximate solution to the even $D + 1$ and the new lower bound on $OPT_A(G, D)$, we prove that an 8-approximate solution to APD can be constructed in $O(|V|^3)$ time. For BAPD, we propose an $O(|V|)$ time $(4, 2)$-approximation (resp., $(6, 3)$-approximation) algorithm for an odd $D \geq 5$ (resp., $D = 3$).

The paper is organized as follows. In Section 2, we state our main results that APD with a forest and an odd D is 8-approximable and BAPD with a forest and an odd $D \geq 5$ (resp., $D = 3$) is (4,2)-approximable (resp., (6,3)-approximable), after introducing some basic notation. In Section 3, we propose an 8-approximation algorithm, named ODD-APD(G, D), to APD with a forest G and an odd D, after reviewing Chepoi and Vaxes' algorithm [6] to APD with a forest and an even D. In Section 4, we give an approximation algorithm to BAPD with a forest G and an odd D. In Section 5, we give concluding remarks.

2 Preliminaries

Let $G = (V, E)$ stand for an undirected simple graph with a set V of *vertices* and a set E of *edges*. An edge with end vertices u and v is denoted by (u, v). We denote $|V|$ by n and $|E|$ by m. A singleton set $\{x\}$ may be simply written as x, and " $\subset$ " implies proper inclusion while " $\subseteq$ " means " $\subset$ " or " $=$ ". In $G = (V, E)$, its vertex set V and edge set E may be denoted by $V(G)$ and $E(G)$, respectively. For a subset $V' \subseteq V$ in G, $G[V']$ denotes the subgraph induced by V'. For an edge set E', we denote by $V[E']$ the set of all end vertices of edges in E'. For a vertex set $X \subset V$ in a graph G, we denote by $N_G(X)$ the set of vertices in $V - X$ adjacent to some vertex $v \in X$.

The *length of a path* P is defined by the number of edges in P and is denoted as $|P|$ (i.e., $|P| = |E(P)|$). For two vertices $u, v \in V$ in $G = (V, E)$, the *distance between* u *and* v is defined as the length of a path between u and v with the shortest length, and it is denoted by $d_G(u, v)$. The *diameter of a graph* G, denoted by $diam(G)$, is defined as the maximum among distances between all pairs of vertices in G. For a vertex $u \in V$ in a graph G and an integer k, let $N_G^k(u)$ denote the set of vertices v with $d_G(u, v) = k$. A set $B_G(u, k) = \bigcup_{k' \le k} N_G^{k'}(u)$ of vertices is called the *ball centered at* u *of radius* k. $B_G(u, k)$ may be simply called a k*-ball* (*with a center* u). For a subset $V' \subseteq V$ of vertices and a family $\mathcal{B}$ of balls, we say that $\mathcal{B}$ *covers* V' if every vertex in V' is contained in some ball in $\mathcal{B}$.

A *forest* is a graph with no cycle. For a forest $G = (V, E)$, a vertex $v \in V$ with degree 1 or 0 is called a *leaf*, and we denote the set of all leaves in G by $L(G)$. For a set X of vertices in a forest G, two vertices x_1 and x_2 in V are called *adjacent with respect to* X if the path between x_1 and x_2 does not contain any vertex in $X - \{x_1, x_2\}$ (note that a path between two vertices is uniquely determined in a forest). A graph $G = (V, E)$ with $|V| \ge k + 1$ is called k*-vertex-connected*, if the deletion of any vertex set X with $|X| \le k - 1$ leaves a connected graph. The *vertex-connectivity* of G, denoted by $\kappa(G)$, is defined as the largest integer k for which G is k-vertex-connected.

In this paper, we consider the following two problems.

Problem 2 *Augmentation Problem with Diameter Requirements* (APD)
Input: *A graph* $G = (V, E)$ *and a nonnegative integer* D.
Output: *A set* E^* *of edges with the minimum cardinality such that* $diam(G^*)$ $\le D$ *holds for* $G^* = (V, E \cup E^*)$. □

Problem 3 *Biconnectivity Augmentation Problem with Diameter Requirements (BAPD)*
Input: *A graph* $G = (V, E)$ *with* $|V| \ge 3$ *and a nonnegative integer* D.
Output: *A set* E^* *of edges with the minimum cardinality such that* $diam(G^*)$

$\leq D$ *and* $\kappa(G^*) \geq 2$ *hold for* $G^* = (V, E \cup E^*)$. □

Let $OPT_A(G,D)$ and $OPT_B(G,D)$ denote the optimal value to APD and BAPD with G and an integer D, respectively. For these problems, we show the following two theorems.

Theorem 4 *If G is a forest and D is an odd integer, then an 8-approximate feasible solution to APD can be found in $O(n^3)$ time.* □

Theorem 5 *Let G be a forest. Then a $(4,2)$-approximate ($(6,3)$-approximate) feasible solution to BAPD can be found in $O(n)$ time if D is an odd ≥ 5 (resp., $D = 3$ holds).* □

3 APD with a forest

In this section, let $G = (V, E)$ be a forest and $D = 2R + 1$ be an odd integer with $R \geq 1$. We show that APD is 8-approximable in $O(n^3)$ time in the case where G is a forest and D is an odd integer.

Let $P_{u,v}$ denote a path between two vertices u and v in G (note that $P_{u,v}$ is uniquely determined if G is a forest). We first find a 2-approximate solution E_1 to APD with the forest G and the even $D' = D + 1$ by Chepoi and Vaxes' algorithm [6]. Note that $|E_1|/2$ is a lower bound on $OPT_A(G,D)$ since we have $OPT_A(G,D) \geq OPT_A(G,D+1) \geq |E_1|/2$. We then construct an 8-approximate solution to APD with G and D based on the edge set E_1. In this section, we first review Chepoi and Vaxes' algorithm [6] in Section 3.1, analyze properties of solutions by their algorithm, derive another lower bound on $OPT_A(G,D)$, and finally propose an 8-approximation algorithm based on these analyses in Section 3.2.

3.1 Even diameters

The following algorithm EVEN-APD is a 2-approximation algorithm to APD with a forest G and an even D' by Chepoi and Vaxes [6].

Algorithm EVEN-APD(G, D')

Input: A forest $G = (V, E)$ and an even integer $D' \geq 2$.

Output: A new edge set E_1 with $diam((V, E \cup E_1)) \leq D'$ and $|E_1| \leq 2OPT_A(G, D')$.

Step 1: Let $R = \frac{D'-2}{2}$. Choose a center $c^* \in V$ for the $(R+1)$-ball and a set C_1 of centers for R-balls so that the family of these $|C_1|+1$ balls covers V in G and the number $|C_1|+1$ of centers is minimized. Halt after outputting the set $E_1 = \{(c^*, c) \mid c \in C_1\}$ of new edges. □

For the completeness of the paper, we give a sketch of the proof for the correctness of algorithm EVEN-APD.

Theorem 6 [6] *The edge set E_1 obtained by algorithm EVEN-APD(G, D') satisfies $|E_1| \leq 2OPT_A(G, D')$, and can be found in $O(nm)$ time.*

Proof sketch: Let E^* be an optimal solution to APD with G and $D' = 2R+2$, and $W = V[E^*]$. Let $X = V - (\bigcup_{w \in W} B_G(w, R))$; X is the set of vertices which are not contained in $B_G(w, R)$ for any $w \in W$. Then we can prove from $diam((V, E \cup E^*)) \leq 2R+2$ that $d_G(x, y) \leq 2R+2$ holds for every two vertices x and y in X and that there is a vertex $v^* \in V$ with $X \subseteq B_G(v^*, R+1)$. Moreover, the edge set $E' = \{(v^*, w) \mid w \in W\}$ satisfies $diam((V, E \cup E')) \leq 2R+2$ and $|E'| \leq |W| \leq 2|E^*|$. From the construction of E_1, we have $|E_1| \leq |E'| \leq 2|E^*| = 2OPT_A(G, D')$. □

Let $H_1 = (V, E \cup E_1)$ and $C_2 = N_G(c^*)$. The following lemma holds from the construction of E_1.

Lemma 7 *(i) $|C_1| = |E_1| \leq 2OPT_A(G, 2R+2) \leq 2OPT_A(G, 2R+1)$ holds.*
(ii) The family of R-balls with centers in $C_1 \cup C_2$ covers V if $R \geq 1$.
(iii) In H_1, every vertex $v \in V$ satisfies $d_{H_1}(c^*, v) \leq R+1$.
(iv) Every two vertices $u_1, u_2 \in V$ with $d_{H_1}(u_1, u_2) > 2R+1$ satisfy $d_{H_1}(u_1, u_2) = 2R+2$ and $d_{H_1}(c^*, u_1) = d_{H_1}(c^*, u_2) = R+1$; such a vertex u_i satisfies $d_G(u_i, c) = R$ for some $c \in C_1 \cup C_2$. □

3.2 Odd diameters

In this section, we propose an algorithm, named ODD-APD(G, D), for constructing a solution to APD with an odd diameter $D = 2R+1$. This algorithm consists of the following three steps. In the first step, we compute a center c^*, a set C_1 of centers and the set $C_2 = N_G(c^*)$ in Lemma 7, and augment G by the new edge set $E_1 = \{(c^*, v) \mid v \in C_1\}$. In $H_1 = (V, E \cup E_1)$, there may be a vertex $u \in V$ such that $d_{H_1}(u, u') > 2R+1$ for some other vertex u'. We call such a vertex u *distant.* By Lemma 7(iv), we see that $d_{H_1}(u, u') = 2R+2$ holds. Thus, to make the diameter at most $D = 2R+1$, it suffices to decrease by at least one the distance between those vertices in the second and third steps. In the second step, we compute a set C_3 by choosing at most $2|C_1|$ vertices from

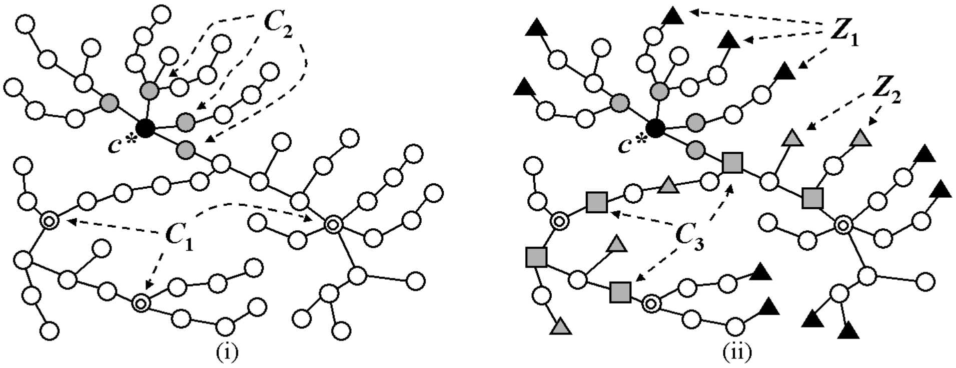

Fig. 1. (i) Illustration of a tree G, a vertex c^* and sets C_1 and C_2 of vertices obtained by applying Step 1 of algorithm ODD-APD$(G, 7)$ to G, where each vertex in C_1 (resp., C_2) is drawn by a double circle (resp., a shaded circle). (ii) Illustration of distant vertices and a set C_3 of vertices, where each distant vertex in Z_1 (resp., Z_2) is drawn by a black (resp., shaded) triangle, and each vertex in C_3 is drawn by a shaded square.

$N_G(C_1 \cup C_2)$ and augment H_1 by the new edge set $E_2 = \{(c^*, v) \mid v \in C_3\}$, by which $H_2 = (V, E \cup E_1 \cup E_2)$ satisfies $d_{H_2}(c^*, v) \leq R$ for some distant vertex v in H_1. In the third step, we augment H_2 by a new edge set E_3 with $|E_3| \leq 2OPT_A(G, D)$ so that in $H_3 = (V, E \cup E_1 \cup E_2 \cup E_3)$, every distant vertex u in H_1 now satisfies $d_{H_3}(u, u') \leq 2R + 1$ for all vertices $u' \in V$.

More precisely, algorithm ODD-APD(G, D) is described as follows, where E_3 in Step 3 is constructed based on our new lower bound on $OPT_A(G, D)$, and how to choose such E_3 will be described after verifying Step 2.

Algorithm ODD-APD(G, D)

Input: A forest $G = (V, E)$ and an odd integer $D \geq 3$.

Output: An set E^* of edges with $diam((V, E \cup E^*)) \leq D$ and $|E^*| \leq 8OPT_A(G, D)$.

Step 1: Let $R = \frac{D-1}{2}$. Compute a center c^*, a set C_1 of centers, the set $C_2 = N_G(c^*)$ and the edge set E_1 that satisfy Lemma 7.

Step 2: Regard each component G^ℓ of G as a rooted tree by choosing its root c^ℓ as c^* if $c^* \in V(G^\ell)$ and as an arbitrary vertex in $C_1 \cap V(G^\ell)$ otherwise. For each vertex $c \in C_1$, let $Q_c = N_G(\{c, p\}) \cap V(P_{c,p})$ for the nearest ancestor $p \in C_1 \cup C_2$ of c in the rooted tree G^ℓ containing c. Let $C_3 = \bigcup_{c \in C_1} Q_c$, $E_2 = \{(c^*, v) \mid v \in C_3\}$, and $H_2 = (V, E \cup E_1 \cup E_2)$.

Step 3: Compute a set E_3 of edges such that $|E_3| \leq 2OPT_A(G, D)$ and $d_{(V, E \cup E_3)}(u, v) \leq 2R + 1$ holds for every two vertices $u, v \in V$ with $d_{H_2}(u, v) >$

$2R+1$. Halt after outputting $E^* = E_1 \cup E_2 \cup E_3$. $\square$

We first prove the correctness of algorithm ODD-APD(G, D) under the assumption that Step 3 works correctly, by Lemmas 8–10. These lemmas are derived almost directly from the structure of edge sets $E_1 \cup E_2$ obtained in Steps 1 and 2. After that, we will describe the procedure of Step 3 for the details and show its correctness.

Let Z denote the set of all distant vertices in $H_1 = (V, E \cup E_1)$ (see Fig. 1 where $D = 7$). Let Z_1 be the set of distant vertices $u \in Z$ such that u has only one vertex in $C_1 \cup C_2$ that is adjacent to u with respect to $C_1 \cup C_2$ in G (recall the definition of "adjacent with respect to a vertex set in a forest" given in Section 2). Let $Z_2 = Z - Z_1$. For Q_c obtained in Step 2, we have $|Q_c| \leq 2$ for each $c \in C_1$ and $C_3 \subseteq \bigcup_{c \in C_1 \cup C_2} N_G(c)$, and hence we see that the following properties hold.

Lemma 8 *(i) $|C_3| \leq 2|C_1|$ holds. (ii) For two vertices $c_i, c_j \in C_1 \cup C_2$ adjacent with respect to $C_1 \cup C_2$, assume that the path P_{c_i,c_j} satisfies $|P_{c_i,c_j}| \geq 2$. Then we have $|P_{c_i,c_j}| = 2$ and $c^* \in V(P_{c_i,c_j})$ if $\{c_i, c_j\} \subseteq C_2$ holds, and we have $V(P_{c_i,c_j}) \cap N_G(\{c_i, c_j\}) \subseteq C_3$ otherwise.* $\square$

Lemma 9 *Every distant vertex $u \in Z_2$ satisfies $d_{H_2}(u, u') \leq 2R+1$ for any vertex $u' \in V$.*

PROOF. It suffices to show that every distant vertex $u \in Z_2$ satisfies $d_{H_2}(c^*, u) \leq R$ by Lemma 7(iii). Let $c \in C_1 \cup C_2$ be a vertex with $d_G(c, u) = R$ (such c exists by Lemma 7(iv)). From the definition of Z_2, it is not difficult to see that $(V(P_{c,u}) \cap V(P_{c,c'})) - \{c, c'\} \neq \emptyset$ holds in G for some vertex $c' \in C_1 \cup C_2$. This implies that $|P_{c,c'}| \geq 2$ and $\{c, c'\} - C_2 \neq \emptyset$ hold. Hence, by Lemma 8(ii), the vertex v with $V(P_{c,c'}) \cap N_G(c) = \{v\}$ satisfies $v \in C_3$. Thus $v \in V(P_{c,u})$ and $(c^*, v) \in E_2$ indicate that $d_{(V, E \cup E_2)}(u, c^*) \leq d_G(u, v) + 1 = R$ holds. $\square$

By Lemmas 7(i) and 8(i), we have $|E_1 \cup E_2| \leq 3|C_1| \leq 6OPT_A(G, D)$. Therefore, the following lemma holds.

Lemma 10 *Assume that such an edge set E_3 in Step 3 can be found. Then the edge set E^* obtained by algorithm ODD-APD(G, D) satisfies $diam((V, E \cup E^*)) \leq D$ and $|E^*| \leq 8OPT_A(G, D)$.* $\square$

In the rest of this section, we show how to find E_3 in Step 3 of algorithm ODD-APD(G, D). For this, we establish the following lemmas. In order to specify distant vertices in Z_1, we divide G into subgraphs G_i corresponding to $c_i \in C_1 \cup C_2$ in the following manner. Let G_i be the component containing

$c_i \in C_1 \cup C_2$ in G' which is obtained from G by removing the set $C_3 \cup \{c^*\}$ of vertices and the set $\{(c, c') \in E \mid c, c' \in C_1 \cup C_2\}$ of edges (see Fig. 2(i)).

Lemma 11 *(i) $V(G_i) \cap (C_1 \cup C_2) = \{c_i\}$ holds.*
(ii) A vertex $u \in V - (C_1 \cup C_2)$ is contained in G_i if and only if c_i is the only vertex in $C_1 \cup C_2$ that is adjacent to u with respect to $C_1 \cup C_2$.
(iii) Every vertex $v \in V(G_i)$ satisfies $d_G(c_i, v) \le R$. Moreover, each vertex $v \in V(G_i)$ with $d_G(c_i, v) = R$ is a leaf also in G.
(iv) Every distant vertex $v \in Z_1$ is contained in some G_i, and it is a leaf in G with $d_G(c_i, v) = R$.

PROOF. (i) Let c_i and c_j be two vertices in $C_1 \cup C_2$ adjacent to each other with respect to $C_1 \cup C_2$. If $d_G(c_i, c_j) = 1$ (resp., $d_G(c_i, c_j) \ge 2$) holds, then c_i and c_j are contained in distinct components in G' by $E(G') \cap \{(c_i, c_j) \in E \mid c_i, c_j \in C_1 \cup C_2\} = \emptyset$ (resp., by Lemma 8 (ii) and $(C_3 \cup \{c^*\}) \cap V(G') = \emptyset$).

(ii) It is not difficult to see that the property follows from the construction of G_i.

(iii) By (ii), $d_G(c_i, v) \le R$ holds since v must be contained in the R-ball with the center c_i. Moreover, we see that if $d_G(c_i, v) = R$ holds, then v is a leaf in G.

(iv) Assume that v is a distant vertex in Z_1. From (ii), the definition of Z_1, and the construction of G_i, v is contained in some G_i having $c_i \in C_1 \cup C_2$. Lemma 7(iv) and the above statement (iii) say that $d_G(c_i, v) = R$ holds and v is a leaf in G. □

Since each distant vertex in Z_1 is a leaf in G by Lemma 11(iv), we call a vertex $v \in Z_1$ a *distant leaf*. Let G_i^j, $j = 1, 2, \ldots, g_i$ be the component in $G_i - c_i$ such that $V(G_i^j)$ contains a distant leaf, where g_i denotes the number of such subgraphs G_i^j in G_i.

Lemma 12 *Let G_i^j be a component in $G_i - c_i$ such that $V(G_i^j)$ contains a distant leaf, and $v \in V(G_i^j)$ be a distant leaf in G_i. For any vertex $x \in V - V(G_i^j) - \{c_i\}$, we have $d_G(v, x) > R$.*

PROOF. From the definition of G_i^j, each vertex $x \in V(G_i) - V(G_i^j) - \{c_i\}$ satisfies $d_G(v, x) > R$. Assume that $x \in V - V(G_i)$ holds. Lemma 11(ii) implies that the path $P_{x,v}$ always contains the vertex c_i. Since $d_G(c_i, v) = R$ holds by Lemma 11(iv), we have $d_G(v, x) > R$. □

Let $\mathcal{G} = \{G_1, G_2, \ldots, G_t\}$ be the family of subgraphs G_i which have distant leaves, where $g_1 \geq g_2 \geq \cdots \geq g_t$ holds. Let $V(G_i^j) \cap N_G(c_i) = \{a_i^j\}$ for each $G_i \in \mathcal{G}$ (note that $|V(G_i^j) \cap N_G(c_i)| = 1$ holds from the construction of G_i^j). We here establish a new lower bound on $OPT_A(G, 2R+1)$. Let E' be an arbitrary feasible solution to APD. If no edge in E' is incident to any vertex in $V(G_i^k) \cup V(G_j^\ell)$ for some G_i^k and G_j^ℓ, $i \neq j$, then $(c_i, c_j) \in E$ or $(c_i, c_j) \in E'$ must hold. It follows since otherwise Lemma 12 implies that $d_{(V, E \cup E')}(x, y) > D$ would hold for two distant leaves $x \in V(G_i^k)$ and $y \in V(G_j^\ell)$. Hence for the family $\mathcal{G}'$ of G_i such that some G_i^k has no edge in E' incident to it, the set of the corresponding centers c_i induces a complete graph in $(V, E \cup E')$; E' includes all edges of such a complete graph other than those in E. Moreover, for each $G_i \in \mathcal{G} - \mathcal{G}'$, any G_i^k has some edge in E' incident to it; $\bigcup_{G_i \in \mathcal{G} - \mathcal{G}'} \bigcup_k V(G_i^k)$ has at least $\lceil \sum_{G_i \in \mathcal{G} - \mathcal{G}'} g_i/2 \rceil$ edges in E' incident to it. Thus, intuitively, E' contains the edges connecting any two corresponding centers c_i for some family $\mathcal{G}'$ of G_i and at least $\lceil \sum_{G_i \in \mathcal{G} - \mathcal{G}'} g_i/2 \rceil$ edges incident to some G_i^k with $G_i \in \mathcal{G} - \mathcal{G}'$. Let

$$f_1(i,j) = |\{(c_i, c_j)\} - E| + \left\lceil \frac{1}{2} \sum_{\ell \in \{1,2,\ldots,t\} - \{i,j\}} g_\ell \right\rceil, 1 \leq i < j \leq t,$$

$$f_2(i,j,k) = |\{(c_i, c_j), (c_j, c_k), (c_k, c_i)\} - E| + \left\lceil \frac{1}{2} \sum_{\ell \in \{1,2,\ldots,t\} - \{i,j,k\}} g_\ell \right\rceil,$$

$$1 \leq i < j < k \leq t,$$

$$f_3(r) = \frac{r^2}{2} - \frac{3r}{2} + 1 + \left\lceil \frac{1}{2} \sum_{r+1 \leq \ell \leq t} g_\ell \right\rceil, 1 \leq r \leq t.$$

Note that in the case of $\mathcal{G}' = \{G_i, G_j\}$ (resp., $\mathcal{G}' = \{G_i, G_j, G_k\}$, resp., $\mathcal{G}' = \{G_1, G_2, \ldots, G_r\}$), the first term of $f_1(i,j)$ (resp., $f_2(i,j,k)$, resp., $f_3(r)$) indicates the number of edges connecting c_i, c_j (resp., any two vertices in $\{c_i, c_j, c_k\}$, resp., any two vertices in $\{c_1, c_2, \ldots, c_r\}$) to be included in E' and its second term indicates the number of edges incident to $\bigcup_{G_i \in \mathcal{G} - \mathcal{G}'} V(G_i)$ to be included in E'. We can prove that $OPT_A(G, 2R+1) \geq \min\{\min_{1 \leq i < j \leq t} f_1(i,j)$, $\min_{1 \leq i < j < k \leq t} f_2(i,j,k), \min_{r \in \{1,2,\ldots,t\} - \{2,3\}} f_3(r)\}$.

Lemma 13 *$OPT_A(G, 2R+1) \geq f(G)$ holds, where*

$$f(G) = \min\{\min_{1 \leq i < j \leq t} f_1(i,j), \min_{1 \leq i < j < k \leq t} f_2(i,j,k), \min_{r \in \{1,2,\ldots,t\} - \{2,3\}} f_3(r)\}. \quad (1)$$

PROOF. Let E' be an arbitrary feasible solution to APD with a forest G and $D = 2R + 1$. Let $\mathcal{G}' \subseteq \mathcal{G}$ be the family of subgraphs G_i such that some G_i^j satisfies $V(G_i^j) \cap V[E'] = \emptyset$.

Claim 14 *$E \cup E'$ contains the edge (c_i, c_j) for every pair of graphs $G_i, G_j \in \mathcal{G}'$.*

PROOF. Let $v_i \in V(G_i^h)$ and $v_j \in V(G_j^k)$ be two distant leaves with $V(G_i^h) \cap V[E'] = \emptyset = V(G_j^k) \cap V[E']$, and $H' = (V, E \cup E')$. Then the edge (c_i, c_j) must be included in $E \cup E'$, since otherwise Lemma 12 implies that $d_{H'}(v_i, v_j) > 2R + 1$ would hold, contradicting the feasibility of E'. □

This claim implies that the set $\{c_i \mid G_i \in \mathcal{G}'\}$ of vertices induces a complete graph in H'; the edge set $E'_1 = \{(c_i, c_j) \notin E \mid \{G_i, G_j\} \subseteq \mathcal{G}'\} \subseteq E'$ holds.

Moreover, every G_i^j in $G_i \in \mathcal{G} - \mathcal{G}'$ has some edge in E' incident to $V(G_i^j)$. Let E'_2 be the set of all edges in E' incident to some vertex in $\bigcup_{G_i \in \mathcal{G} - \mathcal{G}'} \bigcup_j V(G_i^j)$. Since one edge can contribute to two distinct $V(G_i^j)$ and $V(G_k^\ell)$, we have $|E'_2| \geq \lceil \sum_{G_i \in \mathcal{G} - \mathcal{G}'} g_i/2 \rceil$. Note that $E'_1 \cap E'_2 = \emptyset$ holds since E'_1 consists of edges connecting two distinct vertices in $\{c_i \mid G_i \in \mathcal{G}\}$ and each edge in E'_2 is incident to some vertex in $V - \{c_i \mid G_i \in \mathcal{G}\}$. Hence we have $|E'| \geq |E'_1| + |E'_2|$. Let $f'(\mathcal{G}') = |E'_1| + |E'_2|$ and $f''(r) = \min\{f'(\mathcal{G}') \mid |\mathcal{G}'| = r\}$. From the definition of f_1 and f_2, we have $f''(2) = \min_{1 \leq i < j \leq t} f_1(i, j)$ and $f''(3) = \min_{1 \leq i < j < k \leq t} f_2(i, j, k)$.

Assume that $r = |\mathcal{G}'| \geq 1$ holds. Since G is a forest, the graph induced by $\{c_i \mid G_i \in \mathcal{G}'\}$ has at most $r - 1$ edges in E, which implies that we have $|E'_1| \geq r(r-1)/2 - (r-1) = (r-1)(r-2)/2$. Moreover, we can see that $|E'_2| \geq \lceil \sum_{i=r+1}^{t} g_i/2 \rceil$ holds since $g_1 \geq g_2 \geq \cdots \geq g_t$ holds. Thus, $f'(\mathcal{G}') \geq f''(r) \geq f_3(r)$ holds.

Now note that each $G_i \in \mathcal{G}$ satisfies $g_i \geq 1$. Hence, we can see that $f''(0) \geq f''(1)$ holds, since $|E'_1| = 0$ satisfies in both cases of $|\mathcal{G}'| = 0$ and $|\mathcal{G}'| = 1$. Consequently, $|E'| \geq \min_{0 \leq r \leq t} f''(r) \geq \min\{f''(2), f''(3), \min_{r \in \{1,2,\ldots,t\} - \{2,3\}} f_3(r)\}$ holds, which proves the lemma. □

The following lemma shows that we can find an edge set E_3 with $|E_3| \leq 2f(G)$ and $d_{(V, E \cup E_3)}(u, v) \leq 2R+1$ for every two distant leaves u and v (see Fig. 2(ii)).

Lemma 15 *For an edge set E_3 chosen according to the following conditions (i)–(iii), we have $|E_3| \leq 2f(G)$ and $d_{(V, E \cup E_3)}(u, v) \leq 2R + 1$ for every two distant leaves u and v.*

(i) If $f(G) = f_1(i, j)$ holds for some $1 \leq i < j \leq t$, then let $E_3 = \{(c_i, c_j) - E\} \cup \{(c_i, a_\ell^h) \mid i \neq \ell \neq j,\ 1 \leq h \leq g_\ell\}$.

(ii) If $f(G) = f_2(i, j, k)$ holds for some $1 \leq i < j < k \leq t$, then let $E_3 = (\{(c_i, c_j), (c_j, c_k), (c_k, c_i)\} - E) \cup \{(c_i, a_\ell^h) \mid \ell \notin \{i, j, k\},\ 1 \leq h \leq g_\ell\}$.

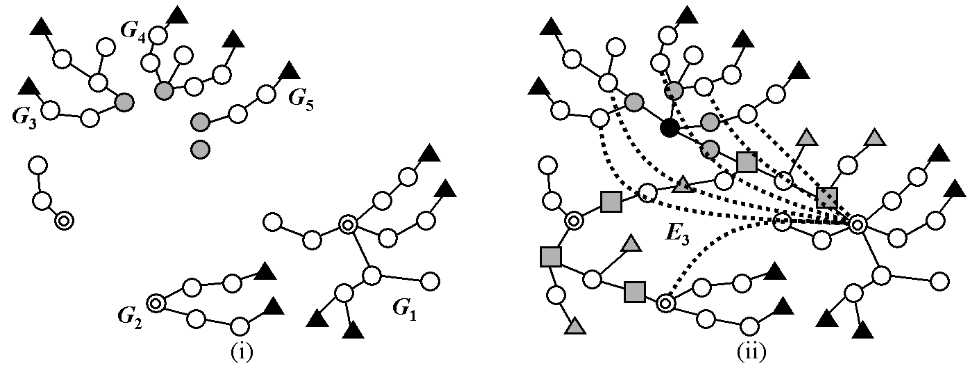

Fig. 2. (i) Illustration of components in G' having a center in $C_1 \cup C_2$, and $\mathcal{G} = \{G_1, G_2, \ldots, G_5\}$, which are obtained from G in Fig. 1, where $g_1 = 3$, $g_2 = g_3 = g_4 = 2$, and $g_5 = 1$ hold. (ii) Illustration of a set E_3 of broken edges based on $f(G)$, where $f(G) = f_1(1, 2) = 4$ holds.

(iii) If $f(G) = f_3(r)$ holds for $r = 1$ or some $4 \le r \le t$, then let $E_3 = \{(c_i, c_j) \notin E \mid 1 \le i < j \le r\} \cup \{(c_1, a^h_\ell) \mid r+1 \le \ell \le t,\ 1 \le h \le g_\ell\}$.

PROOF. First we show $|E_3| \le 2f(G)$. The cases of (i) and (ii) are clear from the definition of f_1 and f_2, respectively. The case of (iii) follows since $|\{(c_i, c_j) \notin E \mid 1 \le i < j \le r\}| \le r(r-1)/2 \le 2(r^2/2 - 3r/2 + 1)$ holds by $r = 1$ or $r \ge 4$.

Let $C' = \{c_i, c_j\}$ and $c' = c_i$ (resp., $C' = \{c_i, c_j, c_k\}$ and $c' = c_i$, resp., $C' = \{c_1, \ldots, c_r\}$ and $c' = c_1$) in the case of (i) (resp., (ii), resp., (iii)). Let $H'_3 = (V, E \cup E_3)$. Then for each distant leaf $v \in V(G^\ell_h)$, we have $d_{H'_3}(c', v) \le d_{H'_3}(a^\ell_h, v) + 1 = R$ if $c_h \notin C'$ holds, and we have $d_{H'_3}(c', v) \le R + 1$ if $c_h \in C'$ holds. Moreover, for every two distant leaves $v' \in V(G_h)$ and $v'' \in V(G_k)$ with $\{c_h, c_k\} \subseteq C'$, we have $d_{H'_3}(v', v'') \le 2R + 1$ by $(c_h, c_k) \in E_3 \cup E$. Therefore we see that $d_{H'_3}(u, v) \le 2R + 1$ holds for every pair of distant leaves u and v. □

The procedure of Step 3 is described as follows.

Step 3: Let G' be the forest obtained from G by removing the set $C_3 \cup \{c^*\}$ of vertices and the set $\{(c, c') \in E \mid c, c' \in C_1 \cup C_2\}$ of edges. Let G_i, $i = 1, 2, \ldots, t$ be the component containing $c_i \in C_1 \cup C_2$ and $N^R_{G_i}(c_i) \ne \emptyset$. Compute a lower bound $f(G)$ on $OPT_A(G, D)$ based on $\mathcal{G} = \{G_1, \ldots, G_t\}$. Let E_3 be the set of edges as defined in Lemma 15. Halt after outputting $E^* = E_1 \cup E_2 \cup E_3$. □

Lemma 16 *Algorithm ODD-APD(G, D) can be implemented to run in $O(n^3)$ time.*

PROOF. An edge set E_1 can be found in $O(n^2)$ time by Theorem 6. In

Step 2, it is not difficult to see that a vertex set C_3 can be found in linear time by the depth first search [16] from c^ℓ in each G^ℓ. So E_2 can be found in $O(n)$ time. In Step 3, computing $f(G)$ takes $O(n^3)$ time. In total, algorithm ODD-APD(G, D) can be implemented to run in $O(n^3)$ time. □

Summarizing the argument given so far, Theorem 4 is now established.

4 BAPD with a forest

In this section, we propose an algorithm, named ODD-BAPD(G, D), which delivers a $(4, 2)$-approximate (resp., $(6, 3)$-approximate) solution to BAPD with a forest G and an odd D in $O(n)$ time if $D \geq 5$ (resp., $D = 3$) holds (recall the definition of (a, b)-approximations in Definition 1).

Let $G = (V, E)$ be a forest with $|V| \geq 3$ and $I(G)$ be the set of isolated vertices $v \in V$ with $N_G(v) = \emptyset$. The algorithm ODD-BAPD(G, D) consists of three steps. In the first step, we compute a set C_1 of centers such that $L(G) \subseteq C_1$ holds, a family of R-balls with centers in C_1 covers V, and the number $|C_1|$ of centers is minimized (recall the definition of $L(G)$ given in Section 2). Pick up two centers $c_1, c_2 \in C_1$ and let $E_1 = \{(c_1, c_2)\} \cup \{(c_1, c) \mid c \in C_1 - I(G) - \{c_1, c_2\}\} \cup \{\{(c_1, c), (c_2, c)\} \mid c \in I(G) - \{c_1, c_2\}\}$ (note that $|C_1| \geq |L(G)| \geq 2$ holds, because a forest G with $|V| \geq 3$ satisfies $|L(G)| \geq 2$). Note that for the biconnectivity, each leaf (resp., isolated vertex) in G must be incident to at least one added edge (resp., at least two added edges). Since each leaf in G is incident to E_1, no distant vertex is a leaf in G. In the second step, we add a set E_2 of new edges with $|E_2| \leq |C_1|$ for $D \geq 5$ (resp., $|E_2| \leq 2|C_1|$ for $D = 3$) in order to reduce the distance between every two distant vertices to at most D, by a slightly modified procedure from Step 2 in algorithm ODD-APD(G, D). In the last step, we replace some edges in $E_1 \cup E_2$ in order to attain the biconnectivity. More precise description of the algorithm is given as follows.

Algorithm ODD-BAPD(G, D)

Input: A forest $G = (V, E)$ and an odd integer $D \geq 3$.

Output: An edge set E^* with $diam((V, E \cup E^*)) \leq D$, $\kappa((V, E \cup E^*)) \geq 2$, and $|E^*| \leq 4OPT_A(G, D) + 2$ for $D \geq 5$ (resp., $|E^*| \leq 6OPT_A(G, D) + 3$ for $D = 3$).

Step 1: Let $R = \frac{D-1}{2}$. Compute a set C_1 of centers with the minimum cardinality such that $L(G) \subseteq C_1$ holds and the family of R-balls with centers in

C_1 covers V. Choose two distinct vertices $c_1, c_2 \in C_1$. Let $E_1 = \{(c_1, c_2)\} \cup \{(c_1, c) \mid c \in C_1 - I(G) - \{c_1, c_2\}\} \cup \{\{(c_1, c), (c_2, c)\} \mid c \in I(G) - \{c_1, c_2\}\}$.

Step 2: Regard each component G^ℓ of G as a rooted tree by choosing its root c^ℓ as c_1 if $c_1 \in V(G^\ell)$ and as an arbitrary vertex in $C_1 \cap V(G^\ell)$ otherwise. For each vertex $c_i \in C_1$, let $p(c_i) \in C_1$ be the nearest ancestor of c_i in G^ℓ and let a_i, b_i be vertices in $V(P_{c_i, p(c_i)})$ with $d_G(c_i, a_i) = R$ and $d_G(c_i, b_i) = R + 1$ if $d_G(c_i, p(c_i)) \geq R + 1$ holds, and $a_i = b_i = \emptyset$ otherwise. Let $Q_i = \{a_i\}$ (resp., $Q_i = \{a_i, b_i\}$) for each vertex $c_i \in C_1 - \{c_1\}$ if $R \geq 2$ (resp., $R = 1$) holds. Let $E_2 = \{(c_1, v) \mid v \in \bigcup_{c_i \in C_1 - \{c_1\}} Q_i\}$.

Step 3: For each component G'_i of G with $\{c_1, c_2\} \cap V(G'_i) = \emptyset$ and $|V(G'_i)| \geq 2$, pick up arbitrarily one edge $(c, c_1) \in E_1$ with $c \in V(G'_i)$ and replace the edge (c, c_1) with a new edge (c, c_2). In the component G' of G with $c_1 \in V(G')$, for the set L_1 of leaves $v \in L(G') - \{c_1\}$ with $c_2 \notin V(P_{v, c_1})$, replace each edge $(c_1, v) \in E_1$ with $v \in L_1$ with a new edge (c_2, v). Let E'_1 be the edge set obtained from E_1 by this procedure. Halt after outputting $E^* = E'_1 \cup E_2$. □

We prove the correctness of algorithm ODD-BAPD(G, D) by the following lemmas. The next lemma shows $|E_1| \leq |C_1| + |I(G)| - 1 \leq 2OPT_B(G, 2R + 1) + 1$.

Lemma 17 *$|C_1| + |I(G)| \leq 2OPT_B(G, 2R + 1) + 2$ holds.*

PROOF. Let E^* be an optimal solution to BAPD with G and $D = 2R + 1$, $W = V[E^*]$, and $H^* = (V, E \cup E^*)$. Since $\kappa(H^*) \geq 2$ holds, we have $L(G) \subseteq W$ and at least two edges in E^* are incident to each $v \in I(G)$. Thus, $|E^*| \geq (|W| + |I(G)|)/2$ holds.

Now we consider the family $\mathcal{B}^*$ of R-balls with centers in C_1 in G. Let $X = V - (\bigcup_{B \in \mathcal{B}^*} B)$; X is the set of vertices which are not contained in any R-ball $B \in \mathcal{B}^*$. The set X satisfies the following property.

Claim 18 *Every two vertices $x, y \in X$ satisfy $d_G(x, y) \leq 2R + 1$.*

PROOF. Assume by contradiction that some two vertices $x, y \in X$ satisfy $d_G(x, y) > 2R + 1$. Since E^* is feasible to BAPD, we have $d_{H^*}(x, y) \leq 2R + 1$. Hence, any path $P^*_{x,y}$ between x and y in H^* with $|P^*_{x,y}| \leq 2R + 1$ contains at least one edge in E^*. For $P^*_{x,y}$, let $v_x \in W \cap V(P^*_{x,y})$ and $v_y \in W \cap V(P^*_{x,y})$ denote the vertex in $W \cap V(P^*_{x,y})$ nearest to x and y, respectively. Since neither x nor y is covered by $\mathcal{B}^*$, we have $d_G(x, v_x) \geq R + 1$ and $d_G(y, v_y) \geq R + 1$. Thus we have $|P^*_{x,y}| \geq 2R + 2$, a contradiction. □

By this claim and the properties that G is a forest and the path between each pair of vertices is unique, we can see that the set X is covered by the family of two R-balls with two centers a and b for some edge $(a, b) \in E$. Therefore, $\mathcal{B}^* \cup \{B_G(a, R), B_G(b, R)\}$ covers V, and we have $|W| + 2 \geq |C_1|$ from the minimality of $|C_1|$. Hence, we have $|E^*| \geq (|W| + |I(G)|)/2 \geq (|C_1| + |I(G)| - 2)/2$, from which $|C_1| + |I(G)| \leq 2OPT_B(G, 2R + 1) + 2$ holds. □

From the construction of C_1, $H_1 = (V, E \cup E_1)$ satisfies $d_{H_1}(c_1, v) \leq R + 1$ for every vertex $v \in V$. Hence any distant vertex $v \in V$ satisfies $d_{H_1}(v, c_1) = R+1$. Moreover, by $L(G) \subseteq C_1$, any distant vertex $v \in V$ satisfies $v \notin L(G)$.

Lemma 19 *For each distant vertex $v \in V$, we have $d_{H_1}(v, c_1) = R + 1$ and there exists a path $P_{c_h,p(c_h)}$ in G for some vertex $c_h \in C_1 - \{c_1\}$ with $v \in V(P_{c_h,p(c_h)})$.*

PROOF. The first statement was proven above. Let v be a distant vertex. By $v \notin L(G)$ and $L(G) \subseteq C_1$, there is a path P_{c_i,c_j} in G for some vertices $c_i, c_j \in C_1$ with $v \in V(P_{c_i,c_j})$. Thus, the second statement holds. □

The following lemma shows that $diam(H_2) \leq D$ holds for $H_2 = (V, E \cup E_1 \cup E_2)$.

Lemma 20 *H_2 satisfies $d_{H_2}(c_1, v) \leq R$ for all vertices $v \in V$.*

PROOF. Let $v \in V$ be a distant vertex. By Lemma 19, $d_G(v, c_i) = R$ holds for some $c_i \in C_1 - \{c_1\}$ and $d_G(v, c) \geq R$ holds for every $c \in C_1$. Again by Lemma 19, we can see that there exists a vertex $c_h \in C_1 - \{c_1\}$ such that $v \in V(P_{c_h,p(c_h)})$. If $c_h = c_i$ holds, then we have $d_{H_2}(v, c_1) = 1 \leq R$ because $d_G(c_h, p(c_h)) = d_G(c_h, v) + d_G(v, p(c_h)) \geq 2R \geq R + 1$ implies $v = a_h$.

We consider the case of $p(c_h) = c_i$. Note that there exists a vertex $c_k \in C_1$ with $p(c_k) = p(c_h) = c_i$ and $R \leq d_G(c_k, v) \leq R+1$, since each vertex in $N_G(v)$ is contained in some ball with a center in C_1. Also note that $d_G(c_k, c_i) \geq R+1$ holds. The case of $d_G(c_k, v) = R$ indicates that $v = a_k$ holds by the above arguments, and hence the lemma is proven. Assume that $d_G(c_k, v) = R + 1$ holds. If $R = 1$ holds, then we have $v = b_k$, from which $d_{H_2}(v, c_1) = 1 \leq R$ holds. If $R \geq 2$ holds, then the vertex $v' \in V(P_{c_k,c_i}) \cap N_G(v)$ with $d_G(c_k, v') = R$ satisfies $v' = a_k$, from which $d_{H_2}(v, c_1) = 2 \leq R$ holds. □

The following lemma proves the correctness of algorithm ODD-BAPD(G, D).

Lemma 21 *A set $E'_1 \cup E_2$ of edges is feasible to BAPD with a forest G and $D = 2R+1$ and satisfies $|E'_1 \cup E_2| \leq 4OPT_B(G, D) + 2$ (resp., $|E'_1 \cup E_2| \leq 6OPT_B(G, D) + 3$) if $R \geq 2$ (resp., $R = 1$) holds.*

PROOF. From the construction of $|E_1|$, we have $|E_1| \leq |C_1| - 1 + |I(G)|$. From the construction of $|E_2|$, we have $|E_2| \leq |C_1| - 1$ if $R \geq 2$ holds, and $|E_2| \leq 2(|C_1| - 1)$ if $R = 1$ holds. By Lemma 17 and $|E_1| = |E'_1|$, we have $|E'_1 \cup E_2| \leq 4OPT_B(G, D) + 2$ (resp., $6OPT_B(G, D) + 3$) if $R \geq 2$ (resp., $R = 1$) holds.

Let $H'_2 = (V, E \cup E'_1 \cup E_2)$. By Lemma 20 and the construction of E'_1, we can see that each vertex $v \in V$ satisfies $d_{H'_2}(c_1, v) \leq R+1$ and every vertex $v \in V$ with $d_{H'_2}(c_1, v) = R+1$ satisfies $d_{H'_2}(c_2, v) = R$. Thus $d_{H'_2}(u, v) \leq 2R+1$ holds for every pair of vertices $u, v \in V$.

For proving that $\kappa(H'_2) \geq 2$ holds, it suffices to show that each leaf in $L(G)$ belongs to a cycle containing the edge (c_1, c_2) in H'_2 (note that $(c_1, c_2) \in E \cup E'_1 \cup E_2$ holds). The case of $L(G) - \{c_1, c_2\} = \emptyset$ is clear, since G is a tree with only two leaves c_1 and c_2 by $|V| \geq 3$ and we have $(c_1, c_2) \in E \cup E'_1 \cup E_2$. Let $v \in L(G) - \{c_1, c_2\}$ be a leaf in G. The case of $v \in I(G)$ is also clear by $\{(v, c_1), (v, c_2)\} \subseteq E'_1$.

Assume that $v \in V(G')$ holds for a component G' of G with $|V(G')| \geq 2$ and $c_1 \notin V(G')$. Let $u_2 \in V(G') \cap C_1$ be a vertex with $(u_2, c_2) \in E'_1$ if $c_2 \notin V(G')$ holds (such u_2 exists from the construction of E'_1), and $u_2 = c_2$ if $c_2 \in V(G')$ holds. Moreover, from $|L(G')| \geq 2$, there exists a vertex $u_1 \in V(G') \cap C_1$ with $(u_1, c_1) \in E'_1$. If $v \neq u_2$ (resp., $v = u_2$) holds, then the cycle $\{(v, c_1), (c_1, c_2), (c_2, u_2)\} \cup E(P_{u_2,v})$ (resp., $\{(v, c_2), (c_2, c_1), (c_1, u_1)\} \cup E(P_{u_1,v})$) proves the claim, where let $(c_2, u_2) = \emptyset$ in the case of $c_2 = u_2$.

Assume that $v \in V(G')$ holds for the component G' of G with $|V(G')| \geq 2$ and $c_1 \in V(G')$. If $v \in L_1$ holds, then the cycle $E(P_{v,c_1}) \cup \{(c_1, c_2), (c_2, v)\}$ proves the claim. If $v \notin L(G') - L_1$ holds, then the cycle $E(P_{v,c_2}) \cup \{(c_2, c_1), (c_1, v)\}$ proves the claim. □

Lemma 22 *Algorithm ODD-BAPD(G, D) can be implemented to run in $O(n)$ time.*

PROOF. By using the algorithm in [4], C_1 can be found in $O(n)$ time. In Step 2, it is not difficult to see that $\bigcup_{c_i \in C_1} Q_i$ can be computed in $O(n)$ time by the depth first search from the vertex c^ℓ in each G^ℓ. In Step 3, L_1 is the set of leaves in components of $G' - c_1$ not containing c_2, where G' denotes the component of G with $c_1 \in V(G')$. Hence E'_1 can be computed in $O(n)$ time. □

Consequently, Lemmas 21 and 22 indicate that Theorem 5 holds.

Finally, we remark that Steps 1 and 2 in algorithm ODD-BAPD(G, D) can find a set E' of edges such that we have $diam((V, E \cup E')) \leq D$ and $|E'|$ is at most two over four times the optimal (resp., at most three over six times the optimal) in the case of $D \geq 5$ (resp., $D = 3$), for any type of problems to which any feasible solution E' satisfies $L(G) \subseteq V[E']$. Since BAPD belongs to such a type of problems, we can prove that BAPD is approximable within some constant without computing $f(G)$ defined in (1). Now it is known in [17] (resp., [12]) that the problem of augmenting an initial graph up to k-edge-connectivity (resp., k-vertex-connectivity) by adding the minimum number of new edges is polynomially solvable (resp., 2-approximable in polynomial time). Therefore, we see that the following property holds.

Corollary 23 *For a forest G, APD with an additional requirement that the resulting augmented graph G' satisfies $\lambda(G') \geq k$ (resp., $\kappa(G') \geq k$) for $k \geq 2$, is $(5, 2)$-approximable (resp., $(6, 2)$-approximable) if D is an odd≥ 5 and $(7, 3)$-approximable (resp., $(8, 3)$-approximable) if $D = 3$ holds, where $\lambda(G)$ denotes the edge-connectivity of G.* □

5 Conclusion

We have shown that APD with a forest G and an odd D is approximable within a constant in polynomial time, by proposing an $O(n^3)$ time 8-approximation algorithm for the problem. For BAPD with a forest G and an odd D, we have shown that a feasible solution E' with $|E'| \leq 4OPT_B(G, D) + 2$ (resp., $|E'| \leq 6OPT_B(G, 3) + 3$) can be found in $O(n)$ time if $D \geq 5$ (resp., $D = 3$) holds. Both algorithms depend on the performance guarantee of Chepoi and Vaxes' algorithm [6] for APD with an even D. Hence, any better approximation algorithm for APD with an even D improves the performance guarantee of approximating APD with an odd D. Actually, it is still open whether APD with a forest is NP-hard or not, and Chepoi and Vaxes [6] conjectured that a solution obtained by their algorithm for APD with a forest and an even D is optimal when $OPT_A(G, D)$ is sufficiently large.

It is also a future work to consider the problem for another class of graphs, while Chepoi and Vaxes' algorithm [6] works for a wider class of graphs containing forests as mentioned in Section 1.

Acknowledgments

We are grateful to the anonymous referee for careful reading and suggestions. This research was partially supported by the Scientific Grant-in-Aid from Ministry of Education, Culture, Sports, Scienceand Technology of Japan.

References

[1] N. Alon, A. Gyárfás, M. Ruszinkó, Decreasing the diameter of bounded degree graphs, J. Graph Theory 35 (2000) 161–172.

[2] C. Berge, Hypergraphs, Elsevier, North-Holland, Amsterdam, 1989.

[3] A. Brandstädt, F. Dragan, V. Chepoi, V. Voloshin, Dually chordal graphs, SIAM J. Discrete Math. 11(3) (1998) 437–455.

[4] R. Chandrasekaran, A. Daughety, Location on tree networks: p-center and n-dispersion problems, Mathematics of Operations Research 6 (1) (1981) 50–57.

[5] V. Chepoi, (personal communication), 2003.

[6] V. Chepoi, Y. Vaxes, Augmenting trees to meet biconnectivity and diameter constraints, Algorithmica 33 (2002) 243–262.

[7] F. R. K. Chung, Diameter of graphs: old problems and new results, Congr Numer 60 (1987) 295–317.

[8] Y. Dodis, S. Khanna, Designing networks with bounded pairwise distance, Proc. 31st Annual ACM Symposium on Theory of Computing (1999) 750–759.

[9] P. Erdős, A. Gyárfás, M. Ruszinkó, How to decrease the diameter of triangle-free graphs, Combinatorica 18 (4) (1998) 493–501.

[10] K. P. Eswaran, R. E. Tarjan, Augmentation problems, SIAM J. Comput. 5 (1976) 653–665.

[11] A. Frank, Connectivity augmentation problems in network design, in Mathematical Programming: State of the Art 1994, J.R. Birge and K.G. Murty (Eds.), The University of Michigan, Ann Arbor, MI (1994) 34–63.

[12] B. Jackson, T. Jordán, A near optimal algorithm for vertex connectivity augmentation, Lecture Notes in Computer Science 1969, Algorithms and Computation (Proc. ISAAC '00) (2000) 312–325.

[13] C. Li, S. McCormick, D. Simchi-Levi, On the minimum-cardinality-bounded-diameter and the bounded-cardinality-minimum-diameter edge addition problems, Operations Research Letters 11 (1992) 303–308.

[14] H. Nagamochi, T. Ibaraki, Graph connectivity and its augmentation: applications of MA orderings, Discrete Applied Mathematics 123 (1) (2002) 447–472.

[15] A. A. Schoone, H. L. Bodlaendar, J. van Leeuwen, Diameter increase caused by edge deletion, J. Graph Theory 11 (3) (1987) 409–427.

[16] R. E. Tarjan, Depth-first search and linear graph algorithms, SIAM J. Comput. 1 (1972) 146–160.

[17] T. Watanabe, A. Nakamura, Edge-connectivity augmentation problems, J. Comput. System Sci. 35 (1987) 96–144.