Theoretical Computer Science 410 (2009) 5072–5081

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Distance paired-domination problems on subclasses of chordal graphs[☆]

Lei Chen [a], Changhong Lu [b,*], Zhenbing Zeng [a]

[a] Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, Shanghai, 200062, China

[b] Department of Mathematics, East China Normal University, Shanghai, 200241, China

A R T I C L E I N F O

Article history:
Received 16 February 2009
Received in revised form 22 July 2009
Accepted 6 August 2009
Communicated by D.-Z. Du

Keywords:
Algorithm
Paired-domination
k-distance domination
Interval graph
Block graphs
Trees

A B S T R A C T

Let $G = (V, E)$ be a graph without isolated vertices. For a positive integer k, a set $S \subseteq V$ is a k-distance paired-dominating set if each vertex in $V - S$ is within distance k of a vertex in S and the subgraph induced by S contains a perfect matching. In this paper, we present two linear time algorithms to find a minimum cardinality k-distance paired-dominating set in interval graphs and block graphs, which are two subclasses of chordal graphs. In addition, we present a characterization of trees with unique minimum k-distance paired-dominating set.

1. Introduction

Let $G = (V, E)$ be a simple graph without isolated vertices. The distance between u and v, denoted by $d_G(u, v)$, is the minimum length of a u-v path in G. For a vertex $v \in V$ and a positive integer k, the *k-neighborhood* of v in G is defined as $N_k(G, v) = \{u \in V \mid d_G(u, v) = k\}$. When $k = 1$, it is the neighborhood of v and simply denoted by $N(G, v)$. The set $N^k(G, v) = \bigcup_{i=1}^{k} N_k(G, v) = \{u \in V \mid 1 \leq d_G(u, v) \leq k\}$ is called the *open total k-neighborhood* of v in G and the set $N^k[G, v] = N^k(G, v) \cup \{v\}$ is called the *closed total k-neighborhood* of v. For $S \subseteq V$, $N^k(G, S) = \bigcup_{v \in S} N^k(G, v)$ and $N^k[G, S] = N^k(G, S) \cup S$. If G is clear from the content, these notations are also denoted by $d(u, v)$, $N_k(v)$, $N(v)$, $N^k(v)$, $N^k[v]$, $N^k(S)$ and $N^k[S]$, respectively. For $S \subseteq V$, the subgraph of G induced by the vertices in S is denoted by $G[S]$. A *matching* in a graph G is a set of pairwise nonadjacent edges in G. For a matching M in G, a vertex v is *unsaturated by M* if v is not incident to any edge of M. Otherwise, we say that v is *saturated by M*. A *perfect matching M* in G is a matching such that G has no unsaturated vertex by M. For a set $S \subseteq V$ and a vertex $v \in S$, the set $P_k(v, S) = N_k(v) - N^k[S - \{v\}]$ is called the *private k-neighborhood of v with regard to S* and a vertex $u \in P_k(v, S)$ is called a *private k-neighbor of v with regard to S*. Some other notations and terminology not introduced in here can be found in [16].

Domination and its variations in graphs have been extensively studied [2,7,8]. A set $S \subseteq V$ is a *dominating set* for a graph $G = (V, E)$ if every vertex in $V - S$ is adjacent to a vertex in S. A set $S \subseteq V$ is a *paired-dominating set* of G if S is a dominating set of G and the induced subgraph $G[S]$ has a perfect matching. The paired-domination was introduced by Haynes and Slater [9]. There are many results on this problem [3–5,10,12,13,15].

For a positive integer k, a set $S \subseteq V$ is a *k-distance paired-dominating set* if each vertex in $V - S$ is within distance k of a vertex in S and $G[S]$ has a perfect matching. Let M be a perfect matching of $G[S]$. If $e = uv \in M$, we say that u and v are

☆ Supported in part by National Natural Science Foundation of China (Nos. 60673048 and 10471044) and National Basic Research Program of China (2003CB318003) and Shanghai leading Academic Discipline Project (No. B407).

* Corresponding author. Tel.: +86 21 54863276.
E-mail address: chlu@math.ecnu.edu.cn (C. Lu).

0304-3975/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2009.08.005

paired in S or u (v) is the *paired vertex* of v (u). We say that $v \in V$ is dominated by u, if $u \in S$ and $d(u, v) \le k$. The *k-distance paired-domination problem* is to determine the *k-distance paired-domination number*, which is the minimum cardinality of a k-distance paired-dominating set for a graph G. The k-distance paired-domination problem was introduced by Raczek as a generalization of paired-domination [14]. We can view a paired-dominating set as a k-distance paired-dominating set with $k = 1$. In [14], Raczek proved that k-distance paired-domination problem is *NP*-complete even restricted to bipartite graphs.

A graph is *chordal* if every cycle of length at least four has a chord. Chordal graphs are raised in the theory of perfect graphs, see [6]. It contains trees, split graphs, interval graphs, block graphs, directed path graphs, undirected path graphs …as subclasses. The subclasses of chordal graphs are of most interesting in the study of many graphs optimization problem [2]. In [4], it was proved that paired-domination problem is *NP*-complete even restricted to split graphs, whose vertex set are the disjoint union of a clique C and a stable set S. For $k \ge 2$, it is easy to point out that k-distance paired-domination problem is *NP*-complete for chordal graph by transforming the paired-domination problem to it as follows. Let G be a chordal graph. We construct the new graph G^* by attaching a path of length $k - 1$ to every vertex of G. Then, G has a paired-dominating set of size at most l if and only if G^* has a k-distance paired-dominating set of size at most l. However, for split graphs, any two vertices in clique C can form a minimum k-distance paired-dominating set. Hence, the k-distance paired-domination number is two for any nontrivial split graph when $k \ge 2$. Meanwhile, a split graph can be partitioned into a clique C and a stable set S in polynomial time [11].

Based on the above discussion, we focus on the k-distance paired-domination problem on other subclasses of chordal graphs in this paper. We provide two linear algorithms to find the minimum k-distance paired-dominating set in interval and block graphs. The algorithms presented in this paper generalize the algorithms in [4]. In Section 2, a linear algorithm will be given for this problem in interval graphs. In Section 3, we will present a linear algorithm for this problem in block graphs. In Section 4, we give a characterization of trees with the unique minimum k-distance paired-dominating set.

2. *k*-distance paired-domination in interval graphs

An *interval representation* of a graph is a family of intervals assigned to the vertices so that vertices are adjacent if and only if the corresponding intervals intersect. A graph having such a representation is an *interval graph*. Booth and Lueker [1] gave an $O(|V(G)| + |E(G)|)$-time algorithm for recognizing an interval graph and constructing an interval representation using *PQ*-tree.

Next, we introduce a labeling method to find a minimum k-distance paired-dominating set in an interval graph. Let $G = (V, E)$ be an interval graph and its interval representation is I. For every vertex $u_i \in V$, I_i is the corresponding interval, and let a_i (b_i, respectively) denote the left endpoint (right endpoint, respectively) of interval I_i. We order the vertices of G by $u_1, u_2, \ldots, u_n$ in increasing order of their left endpoints. It is obvious that if $u_iu_j \in E$ with $j < i$, then $u_ju_k \in E$ for every $j + 1 \le k \le i$. Let $V_i = \{u_j \in V \mid j \le i\}$. If G is a connected interval graph, it is easy to know that $G[V_i]$ is also connected. In this paper, we only consider connected interval graphs.

Let $F(u_i) = u_j$ for $2 \le i \le n$, where $j = \min\{a \mid u_au_i \in E \text{ and } a < i\}$. In particular, we assume that $F(u_1) = u_1$. We define the notation $F^l(u)$ as follows:

$$F^l(u) = \begin{cases} F(u) & \text{if } l = 1; \\ F(F^{l-1}(u)) & \text{if } l \ge 2; \end{cases}$$

Let $w(u_i) = u_j$ for $1 \le i \le n$, where $j = \max\{a \mid d(u_a, u_i) > k \text{ and } a < i\}$. In particular, if $w(u_i)$ does not exist, we assume that $w(u_i) = u_0$ ($u_0 \notin V$). For convenience, we use kPD_i to denote a minimum k-distance paired-dominating set of $G[V_i]$.

Lemma 1. *Let G be an interval graph with vertex ordering $u_1, u_2, \ldots, u_n$ by the increasing order of their left endpoints. If $F^k(u_i) \ne u_i$ and $F^{k+1}(u_i) = F^k(u_i)$, then $\{u_1, u_2\}$ is a kPD_i.*

Proof. $F^{k+1}(u_i) = F^k(u_i)$ implies that $F^k(u_i) = u_1$. Let l be the minimum index such that $F^l(u_i) = u_1$. Assume that $u_{i_a} = F^a(u_i)$ for $1 \le a \le l$. Then $1 = i_l < i_{l-1} < \cdots < i_1 < i_0 = i$. For any vertex u_b $(\neq u_1)$ in V_i, there exists an integer c such that $1 \le c \le l$ and $i_c < b \le i_{c-1}$. $u_{i_c}u_{i_{c-1}} \in E$ implies that $u_{i_c}u_b \in E$. Thus $d(u_1, u_b) \le d(u_1, u_{i_c}) + 1 \le l - c + 1 \le l \le k$. As $u_1u_2 \in E$, $\{u_1, u_2\}$ is a kPD_i. □

Lemma 2. *Suppose G is an interval graph with vertex ordering $u_1, u_2, \ldots, u_n$ by the increasing order of their left endpoints. Then $|kPD_{i+1}| \ge |kPD_i|$ for $2 \le i \le n - 1$.*

Proof. If there is a kPD_{i+1} such that it does not contain u_{i+1}, we claim that $d_{G[V_i]}(u_a, u_b) \le d_{G[V_{i+1}]}(u_a, u_b)$ for any two vertices $u_a, u_b \in V_i$. Suppose to the contrary that there exist two vertices $u_a, u_b \in V_i$ with $d_{G[V_i]}(u_a, u_b) > d_{G[V_{i+1}]}(u_a, u_b)$. Then any shortest u_a-u_b path contains u_{i+1}. Let $P:\ u_a = u_{i_1}, u_{i_2}, \ldots, u_{i+1}, \ldots, u_b$ be a shortest u_a-u_b path with $a < b$. There exists an integer l such that $i_j < b$ and $i_{l+1} > b$ for $j = 1, 2, \ldots, l$. Hence, $u_{i_l}u_b \in E$ and the path $u_a = u_{i_1}, u_{i_2}, \ldots, u_{i_l}, u_b$ is a shorter u_a-u_b path than P. It is a contradiction. So, a kPD_{i+1} is also a k-distance paired-dominating set of $G[V_i]$, and hence $|kPD_{i+1}| \ge |kPD_i|$. Therefore, we assume that S is a kPD_{i+1} and $u_{i+1} \in S$.

Case 1. *u_{i+1} is paired with u_i in S.*
If $N_{G[V_i]}(u_i) \subseteq S$, then $S - \{u_i, u_{i+1}\}$ is a k-distance paired-dominating set of $G[V_i]$. So $|kPD_{i+1}| = |S| > |S| - 2 \ge |kPD_i|$. If

there is a vertex $w \in N_{G[V_i]}(u_i) - S$, then $S - \{u_{i+1}\} \cup \{w\}$ is also a k-distance paired-dominating set of $G[V_i]$. It follows that $|kPD_{i+1}| \geq |kPD_i|$.

Case 2. *u_{i+1} is paired with some vertex u_a in S ($1 \leq a < i$).*
If $u_i \notin S$, then $S - \{u_{i+1}\} \cup \{u_i\}$ is a k-distance paired-dominating set of $G[V_i]$. If $u_i \in S$ and u_i is paired with u_b, then $u_a u_b \in E$ and $S - \{u_i, u_{i+1}\}$ is a k-distance paired-dominating set of $G[V_i]$. For any case, it follows that $|kPD_{i+1}| \geq |kPD_i|$. □

Lemma 3. *Suppose G is an interval graph with vertex ordering $u_1, u_2, \ldots, u_n$ by the increasing order of their left endpoints. Let $F^k(u_i) = u_a$ and $F(u_a) = u_b$ with $b < a < i$. Then,*

(1) *$kPD_l \cup \{u_b, u_a\}$ is a kPD_i if $w(u_b) = u_l$ with $l \geq 2$;*
(2) *$\{u_1, u_2, u_b, u_a\}$ is a kPD_i if $w(u_b) = u_1$;*
(3) *$\{u_b, u_a\}$ is a kPD_i if $w(u_b) = u_0$.*

Proof. (1) It is obvious that $kPD_l \cup \{u_b, u_a\}$ is a k-distance paired-dominating set of $G[V_i]$. Next, we show that $|kPD_i| \geq |kPD_l| + 2$. Let S be a kPD_i. As $F^k(u_i) = u_a$, $d(u_j, u_i) > k$ for every vertex u_j with $j < a$. S contains some vertex u_c with $a \leq c \leq i$. Assume that $u_{i_1} \in S$ is the last vertex, which dominates u_i, in the vertex ordering and u_{i_1} is paired with u_{i_2} in S. It is obvious that $i_2 \geq b$. Let $w(u_{i_1}) = u_c$, $w(u_{i_2}) = u_d$ and $l' = \min\{c, d\}$. As $d(u_l, u_b) > k$, $d(u_l, u_{i_1}) > k$ and $d(u_l, u_{i_2}) > k$. Hence $l' \geq l \geq 2$. Let u_e be the last vertex of $S - \{u_{i_1}, u_{i_2}\}$ in the vertex ordering. If $e \geq l'$, then $S - \{u_{i_1}, u_{i_2}\}$ is a k-distance paired-dominating set of $G[V_e]$. So $|kPD_i| - 2 \geq |kPD_e|$. Since $e \geq l' \geq l$, by Lemma 2, $|kPD_e| \geq |kPD_l|$. Then $|kPD_i| \geq |kPD_l| + 2$. If $e < l'$, then $S - \{u_{i_1}, u_{i_2}\}$ is a k-distance paired-domination set of $G[V_{l'}]$. As $l' \geq l$, by Lemma 2, $|kPD_{l'}| \geq |kPD_l|$. Therefore, $|kPD_i| \geq |kPD'_l| + 2 \geq |kPD_l| + 2$.

(2) Since $w(u_b) = u_1$ and $u_1 u_2 \in E$, it is obvious that $b \geq 3$. Thus, $\{u_1, u_2, u_b, u_a\}$ is a kPD_i.

(3) It is obvious that $\{u_b, u_a\}$ is a kPD_i. □

Based on the above lemmas, we have the following algorithm for k-distance paired-domination problem in interval graphs.

Algorithm *k*-MPDI. Find a minimum k-distance paired-dominating set of an interval graph.
Input An interval graph $G = (V, E)$ with a vertex ordering $u_1, u_2, \ldots, u_n$ ordered by the increasing order of left endpoints, in which each vertex u_i has a label $D(u_i) = 0$.
Output A minimum k-distance paired-dominating set kPD of G.
Method

```
kPD = ∅;
For i = n to 1 do
    If (D(u_i) = 0) then
        If (F^k(u_i) ≠ u_i and F^{k+1}(u_i) ≠ F^k(u_i)) then
            kPD = kPD ∪ {F^k(u_i), F^{k+1}(u_i)};
            D(u) = 1 for every vertex u ∈ N^k[F^k(u_i)] ∪ N^k[F^{k+1}(u_i)];
        else if (F^k(u_i) ≠ u_i) then
            kPD = kPD ∪ {u_1, u_2};
            D(u) = 1 for every vertex u ∈ N^k[u_1];
        else
            kPD = kPD ∪ {u_1, u_2};
            D(u_i) = 1;
        endif
    endif
endfor
```

Theorem 4. *Given a vertex ordering ordered by the increasing order of left endpoints, the algorithm k-MPDI can produce a minimum k-distance paired-dominating set of an interval graph G in $O(m + n)$, where $m = |E(G)|$ and $n = |V(G)|$.*

Proof. The construction and correctness of the algorithm k-MPDI are based on Lemmas 1–3. Since each vertex and edge are used in a constant number, hence the algorithm k-MPDI can finish in $O(m + n)$, where $m = |E(G)|$ and $n = |V(G)|$. □

3. *k*-distance paired-domination in block graphs

In a graph $G = (V, E)$ with $|V| = n$ and $|E| = m$, a vertex x is a *cut-vertex* if there are more connected components in $G - x$ than that in G. A *block* of G is a maximal connected subgraph of G without a cut-vertex. If G itself is connected and has no cut-vertex, then G is a block. It is obvious that the intersection of any two blocks contains at most one vertex, and a vertex is a cut-vertex if and only if it is the intersection of two or more blocks. An *end block* is a block with only one cut-vertex. A *block graph* is a connected graph whose blocks are complete graphs. If every block is K_2, then it is a tree. Every block graph not isomorphic to complete graph has at least two end blocks. For technical reasons, we say that a complete graph has an end block and any vertex is a cut-vertex.

Let G be a block graph with $|V| = n$ and $|E| = m$. For a vertex $v \in V(G)$ and a block B, *the distance of v and B*, denoted by $d_G(v, B)$, is defined as the maximum of $d_G(u, v)$ for $u \in V(B)$.We say a block B is *farthest from v* if $d_G(v, B)$ is maximum over all blocks. Note that B is an end block if B is farthest from v. Our algorithm works on a vertex ordering. In order to obtain this vertex ordering, we first define a vertex ordering connected operation. Let $S = x_1, x_2, \ldots, x_s$ be a vertex ordering and $T = u_1, u_2, \ldots, u_t$ be another vertex ordering. We use $S + T$ to denote a new vertex ordering $x_1, x_2, \ldots, x_s, u_1, u_2, \ldots, u_t$. Let v be a cut-vertex of G. Beginning with a block farthest from v and working recursively inward, we can find a vertex order $v_1, v_2, \ldots, v_n$ as follows.

Procedure VOB
$S = \emptyset$; (S is a vertex ordering.)
Let v be a cut-vertex of G;
While ($G \neq \emptyset$) do
  If (G is a complete graph) then
    Let $V(G) = \{u_1, u_2, \ldots, u_a\}$;
    $S = S + u_1, u_2, \ldots, u_a$;
    $G = G - \{u_1, u_2, \ldots, u_a\}$;
  else
    Let B be an end block farthest from v with $V(B) = \{u_1, u_2, \ldots, u_b, x\}$, where x is the cut-vertex in B;
    $S = S + u_1, u_2, \ldots, u_b$;
    $G = G - \{u_1, u_2, \ldots, u_b\}$;
  endif
enddo
Output S.

Let $v_1, v_2, \ldots, v_n$ be the vertex ordering obtained by procedure VOB. In order to describe the algorithm, we need some notations. For a vertex v_i with $1 \leq i < n$, we define *the father of* v_i as $F(v_i) = v_j$ such that $j = \max\{a \mid v_i v_a \in E,\ a > i\}$. Moreover, v_i is called a *child of* v_j and let $C(v_j) = \{v_i \mid F(v_i) = v_j\}$ be the *child set of* v_j. In special, we define $F(v_n) = v_n$. Obviously, if v_j is the father of some vertex in G, then v_j is a cut-vertex. In addition, we define the l-ancestor of v_i as follows:

$$F^l(v_i) = \begin{cases} F(v_i) & \text{if } l = 1; \\ F(F^{l-1}(v_i)) & \text{if } l \geq 2. \end{cases}$$

The l-child set of v_i, denoted by $C^l(v_i)$, is defined as $C^l(v_i) = \{v_j \mid F^l(v_j) = v_i\}$. In fact, $C(v_i) = C^1(v_i)$. In special, $C^0(v_i) = v_i$. For convenience, let *kPD* denote a minimum k-distance paired-dominating set of G.

In our algorithm, two labels on each vertex, denoted by $(D(w), L(w))$, are used:

$$D(w) = \begin{cases} 0 & \text{if } w \text{ is not dominated;} \\ 1 & \text{if } w \text{ is dominated.} \end{cases}$$

$$L(w) = \begin{cases} 0 & \text{if } w \text{ is not put into kPD;} \\ 1 & \text{if } w \text{ is put into kPD, but it has no paired vertex in kPD;} \\ 2 & \text{if } w \text{ is put into kPD, and it has a paired vertex in kPD.} \end{cases}$$

Now, we are ready to present the algorithm to determine a minimum k-distance paired-dominating set in block graphs.

Algorithm *k*-MPDB. Find a minimum k-distance paired-dominating set of a block graph.
Input A block graph $G = (V, E)$ with a vertex ordering $v_1, v_2, \ldots, v_n$ ($n \geq 2$) obtained by Procedure VOB. Each vertex v_i has labels $(D(v_i), L(v_i)) = (0, 0)$.
Output A minimum k-distance paired-dominating set *kPD* of G.
Method
  For $i = 1$ to $n - 1$ do
    If ($D(v_i) = 0$) then
      Let $A(v_i) = \{w \in N_k(v_i) - \{F^k(v_i)\} \mid G[\{u \mid u \in C(w),\ L(u) = 1\}]$ *has no perfect matching*$\}$;
      If ($A(v_i) = \emptyset$) then
        $L(F^k(v_i)) = 1$;
        $D(u) = 1$ for every vertex $u \in N^k[F^k(v_i)]$;   (*)
      endif
    endif
    If ($D(v_i) = 1$) then
      Let $C_1(v_i) = \{w \mid w \in C^k(v_i)$ *and* $L(w) = 1\}$;
      $L(w) = 2$ for every vertex $w \in C_1(v_i)$;
      Let M be a maximum matching in $G[C_1(v_i)]$ and $C_2(v_i)$ be the vertex set of saturated vertices by M in $C_1(v_i)$;
      If ($C_1(v_i) - C_2(v_i) \neq \emptyset$), then

Let $\{v_{i_1}, v_{i_2}, \ldots, v_{i_l}\}$ be the subset of $C^{k-1}(v_i)$ such that $G[C(v_{i_j}) \cap C_1(v_i)]$ has no perfect matching for $1 \leq j \leq l$;
For $j = 1$ to l do
$L(v_{i_j}) = 2$;
$D(u) = 1$ for every vertex $u \in N^k[v_{i_j}]$; (**)
Let w be a vertex in $C(v_{i_j}) \cap C_1(v_i) - C_2(v_i)$;
For every vertex $v \in C(v_{i_j}) \cap C_1(v_i) - C_2(v_i) - \{w\}$,
$L(v') = 2$ for some vertex $v' \in C(v)$ with $L(v') = 0$;
endfor
endif
endif
endfor
If $(D(v_n) = 0)$ then
$L(v_n) = 2$;
$L(w) = 2$ for some vertex $w \in C(v_n)$ with $L(w) = 0$;
$D(v_n) = 1$; (***)
else
Let $C_1(v_n) = \{w \mid w \in N^k[v_n]\ \textit{and}\ L(w) = 1\}$;
$L(w) = 2$ for every vertex $w \in C_1(v_n)$;
Let M be a maximum matching in $G[C_1(v_n)]$ and $C_2(v_n)$ be the vertex set of saturated vertices by M in $C_1(v_n)$;
If $(C_1(v_n) - C_2(v_n) \neq \emptyset)$, then
For every vertex $w \in C_1(v_n) - C_2(v_n)$
If $(L(F(w)) \neq 2\ \textit{and}\ w \neq v_n)$ then
$L(F(w)) = 2$;
$D(u) = 1$ for every vertex $u \in N^k[F(w)]$; (****)
else
$L(v') = 2$ for some vertex $v' \in C(w)$ with $L(v') = 0$;
endif
endif
endif
Output $kPD = \{v \mid L(v) = 2\}$
end

Next, we will prove the correctness of the algorithm k-MPDB. For a given block graph with order at least two, when the algorithm k-MPDB terminates, any vertex has changed its labels. In detail, for the considering vertex v_i $(\neq v_n)$ with $D(v_i) = 0$ and $A(v_i) = \emptyset$, it changed its label $D(v_i) = 1$ in the line indicated (*) of the algorithm k-MPDB. For the considering vertex v_i $(\neq v_n)$ with $D(v_i) = 0$ and $A(v_i) \neq \emptyset$, it changed its label $D(v_i) = 1$ in the line indicated (**) or (****) of the algorithm k-MPDB. For v_n with $D(v_n) = 0$, it changed its label $D(v_n) = 1$ in the line indicated (***) of the algorithm k-MPDB. Hence, when the algorithm k-MPDB terminates, $D(v) = 1$ for every vertex $v \in V$ and $L(u) = 2$ for every vertex $u \in kPD$. Moreover, $G[kPD]$ contains a perfect matching. Thus kPD is a k-distance paired-dominating set of G. It suffices to prove that kPD is also a minimum k-distance paired-dominating set of G.

Let $S_i = \{v \mid L(v) = 1$ or 2, when v_i is the considering vertex in some step of the algorithm k-MPDB$\}$ and $S'_i = \{v \mid L(v) = 2$, when v_i is the considering vertex in some step of the algorithm k-MPDB$\}$ for $i = 1, 2, \ldots, n$. In particular, $S_{n+1} = S'_{n+1} = \{v \mid L(v) = 2$, when the algorithm k-MPDB terminates$\}$. We use the induction on i to prove that for every $1 \leq i \leq n+1$, there is a minimum k-distance paired-dominating set S such that $S_i \subseteq S$ and $G[S'_i]$ has a perfect matching. Obviously, $S_1 = S'_1 = \emptyset$ and it is true for $i = 1$. Assume that there is a minimum k-distance paired-dominating set S in G such that $S_i \subseteq S$ and $G[S'_i]$ has a perfect matching for $1 \leq i \leq n$. We show that S_{i+1} and S'_{i+1} also hold. The following lemmas will help us to prove the fact.

Lemma 5. *Let v_i $(\neq v_n)$ be the considering vertex with $D(v_i) = 0$ in some step of the algorithm. If $A(v_i) = \emptyset$, then there is a minimum k-distance paired-dominating set S in G such that $S_i \cup \{F^k(v_i)\} \subseteq S$ and $G[S'_{i+1}]$ has a perfect matching.*

Proof. Since $D(v_i) = 0$, $v_i \neq v_n$ and $A(v_i) = \emptyset$, $L(F^k(v_i)) = 1$ in the next step of the algorithm. Thus $S_{i+1} = S_i \cup \{F^k(v_i)\}$ and $S'_{i+1} = S'_i$. By inductive hypothesis, there is a minimum k-distance paired-dominating set S in G such that $S_i \subseteq S$ and $G[S'_i]$ has a perfect matching. Since $S'_{i+1} = S'_i$, the second requirement holds. Next, we prove that $F^k(v_i) \in S$.

Suppose to the contrary that $F^k(v_i) \notin S$. Since S is a minimum k-distance paired-dominating set of G, v_i is dominated by some vertex in S. Let $v \in S$ be the last vertex, which dominates v_i, in the vertex ordering obtained by Procedure VOB. Assume that its paired vertex in S is v'. Since $D(v_i) = 0$ and v is paired with v' in S, it follows that $v \notin S_i$ and $v' \notin S'_i$. If $v' \notin S_i$, let $I = \{u \mid F^l(u) = F^k(v_i)\ \textit{for some}\ l \geq 1\}$. According to Procedure VOB, every vertex in $I - \{v_1, v_2, \ldots, v_{i-1}\}$ within distance k of $F^k(v_i)$. Thus each vertex in $(N^k[v] \cup N^k[v']) \cap (I - \{v_1, v_2, \ldots, v_{i-1}\})$ is within distance k of $F^k(v_i)$. On the other hand, each vertex in $\{v_1, v_2, \ldots, v_{i-1}\}$ either has been dominated by some vertex in S_i or will be dominated by the father of some vertex in $S_i - S'_i$ (see Lemma 7). Therefore $S - \{v, v'\} \cup \{F^k(v_i), w\}$ is also a minimum k-distance paired-dominating set, where w is a neighbor of $F^k(v_i)$. If $v' \in S_i$, then $d(v', v_i) = k + 1$ and v is the father of v'. As $A(v_i) = \emptyset$, the induced subgraph of $B = \{w \mid w \in C(v) \text{ and } L(w) = 1\}$ has a perfect matching. There is a vertex $w' \in B$ such that its paired vertex, say w'', is not in B. Since each vertex in $N^k[v] - \{v_1, v_2, \ldots, v_{i-1}\}$ is within distance k of $F^k(v_i)$, so $S - \{v, w''\} \cup \{F^k(v_i), w\}$ is also a minimum k-distance paired-dominating set, where w is a neighbor of $F^k(v_i)$. So we proved the lemma. □

From Lemma 5, when we consider the vertex v_i ($\neq v_n$) in G such that $D(v_i) = 0$ and $A(v_i) = \emptyset$, $F^k(v_i)$ will be put into *kPD*. However, we cannot determine its paired vertex at once, so let $L(F^k(v_i)) = 1$ and $D(u) = 1$ for every vertex $u \in N^k[F^k(v_i)]$. For the case $A(v_i) \neq \emptyset$, Lemma 7 implies that $A(v_i) \subseteq S$.

The next two lemmas will process the case $D(v_i) = 1$ and $v_i \neq v_n$, when v_i is considered in the algorithm.

Lemma 6. *Let v_i ($\neq v_n$) be the considering vertex with $D(v_i) = 1$ in some step of the algorithm. If $C_1(v_i) - C_2(v_i) = \emptyset$, then there is a minimum k-distance paired-dominating set S such that $S_{i+1} \subseteq S$ and $G[C_1(v_i)]$ has a perfect matching.*

Proof. In the algorithm k-MPDB, set $L(w) = 2$ for every vertex $w \in C_1(v_i)$. It is obvious that $S_{i+1} = S_i$ and $S'_{i+1} = S'_i \cup C_1(v_i)$. Let S be a minimum k-distance paired-dominating set of G such that $S_i \subseteq S$ and $G[S'_i]$ has a perfect matching. $C_1(v_i) - C_2(v_i) = \emptyset$ implies that $G[C_1(v_i)]$ has a perfect matching. □

From Lemma 6, when v_i ($\neq v_n$) is the considering vertex in the algorithm such that $D(v_i) = 1$ and $G[C_1(v_i)]$ has a perfect matching, it is enough to set $L(w) = 2$ for each vertex $w \in C_1(v_i)$.

Lemma 7. *Let v_i ($\neq v_n$) be the considering vertex with $D(v_i) = 1$ in some step of the algorithm. Assume that $C_1(v_i) - C_2(v_i) \neq \emptyset$ and $V_p = \{v_{i_1}, v_{i_2}, \ldots, v_{i_l}\}$ is a subset of $C^{k-1}(v_i)$ such that $G[C(v_{i_j}) \cap C_1(v_i)]$ has no perfect matching for $1 \le j \le l$. Let w_j ($1 \le j \le l$) be a vertex in $C(v_{i_j}) \cap C_1(v_i) - C_2(v_i)$. For each vertex $v \in C(v_{i_j}) \cap C_1(v_i) - C_2(v_i) - \{w_j\}$, take a vertex $v' \in C(v)$ with $L(v') = 0$ into C_p. Then there is a minimum k-distance paired-dominating set S of G such that $S_i \cup V_p \cup C_p \subseteq S$ and $G[C_1(v_i) \cup V_p \cup C_p]$ has a perfect matching.*

Proof. In the algorithm k-MPDB, set $L(w) = 2$ for every vertex $w \in C_1(v_i) \cup V_p \cup C_p$. It is obvious that $S_{i+1} = S_i \cup V_p \cup C_p$ and $S'_{i+1} = S'_i \cup C_1(v_i) \cup V_p \cup C_p$. Let S be a minimum k-distance paired-dominating set of G such that $S_i \subseteq S$ and $G[S'_i]$ has a perfect matching. By the selection of V_p and C_p, $G[C_1(v_i) \cup V_p \cup C_p]$ has a perfect matching. Hence, it suffices to prove that $V_p \cup C_p \subseteq S$.

The set $C_1(v_i) - C_2(v_i)$ is an independent set since M is a maximum matching in $G[C_1(v_i)]$. As the label of every vertex in $C_1(v_i)$ is (1, 1), each vertex in C_p can always be found. Then $|C_p| = |C_1(v_i)| - |C_2(v_i)| - l$. In addition, $C_p \cap S = \emptyset$. Let $C'_p = \{x \mid x \in S - C_1(v_i) \text{ is paired with some vertex in } C_1(v_i)\}$. Then $|C'_p| \ge |C_1(v_i)| - |C_2(v_i)|$. If $V_p \cap S = \emptyset$, then $S - C'_p \cup V_p \cup C_p$ is also a minimum k-distance paired-dominating set of G. If $V_p \cap S \neq \emptyset$, let $V'_p = V_p \cap S$. If $V'_p \subseteq C'_p$, then $S - C'_p \cup V_p \cup C_p$ is also a minimum k-distance paired-dominating set of G. If $V'_p \not\subseteq C'_p$, let $V''_p = V'_p - C'_p$. Without loss of generality, we assume that $V''_p = \{v_{i_1}, v_{i_2}, \ldots, v_{i_a}\}$, where $1 \le a \le l$. Let w'_j ($1 \le j \le a$) be a vertex in $C(w_j)$ with $L(w'_j) = 0$. Since the label of w_j is (1, 1), w'_j can always be found. Then $S - C'_p \cup C_p \cup V_p \cup \{w'_1, \ldots, w'_a\}$ is also a minimum k-distance paired-dominating set of G. Up to now, we proved that $V_p \cup C_p \subseteq S$. □

From Lemma 7, when $v_i (\neq v_n)$ is the considering vertex in the algorithm such that $D(v_i) = 1$ and $G[C_1(v_i)]$ has no perfect matching, we must put their fathers or children of some vertices in $C_1(v_i)$ into *kPD*. At the same time, set $D(u) = 1$ for each vertex in $N^k[v_{i_j}]$. The next lemmas will process the case $v_i = v_n$.

Lemma 8. *Let v_n be the considering vertex in the last step of algorithm k-MPDB. If $D(v_n) = 0$, then there is a minimum k-distance paired-dominating set S of G such that $S_n \cup \{v_n, w\} \subseteq S$, where $w \in C(v_n)$ and $L(w) = 0$.*

Proof. By the inductive hypothesis, there is a minimum k-distance paired-dominating set S of G such that $S_n \subseteq S$ and $G[S'_n]$ has a perfect matching. As $D(v_n) = 0$, by the algorithm k-MPDB, we know that $S_n = S'_n$. In the algorithm k-MPDB, set $L(v_n) = 2$ and $L(w) = 2$, where $w \in C(v_n)$ and $L(w) = 0$. As the label of v_n is (0, 0), w can always be found. So, $S_{n+1} = S_n \cup \{v_n, w\}$ and $S'_{n+1} = S'_n \cup \{v_n, w\}$. It suffices to prove that $\{v_n, w\} \subseteq S$.

If $v_n \in S$, $S - \{w'\} \cup \{w\}$ is also a minimum k-distance paired-dominating set of G, where w' is the paired vertex of v_n in S. If $v_n \notin S$, then v_n must be dominated by a vertex in S, say v, and its paired vertex is v'. Since $S_n = S'_n \subseteq S$, each vertex in $\{v_1, v_2, \ldots, v_{n-1}\}$ is dominated by some vertex in S_n and $v, v' \notin S_n$, thus $S - \{v, v'\} \cup \{v_n, w\}$ is also a minimum k-distance paired-dominating set of G. □

From Lemma 8, when v_n is the considering vertex in the algorithm such that $D(v_n) = 0$, then v_n and its child w with $L(w) = 0$ will be put into *kPD*.

Lemma 9. *Let v_n be a considering vertex in the last step of algorithm k-MPDB. If $D(v_n) = 1$ and $C_1(v_n) - C_2(v_n) = \emptyset$, then there is a minimum k-distance paired-dominating set S such that $S_{n+1} \subseteq S$ and $G[C_1(v_n)]$ has a perfect matching.*

Proof. By the inductive hypothesis, there is a minimum k-distance paired-dominating set S of G such that $S_n \subseteq S$ and $G[S'_n]$ has a perfect matching. If $D(v_n) = 1$, then v_n has already been dominated by some vertex in $N^k(v_n)$. If $C_1(v_n) - C_2(v_n) = \emptyset$, then $S_{n+1} = S_n$ and $S'_{n+1} = S'_n \cup C_1(v_n)$. Since $C_1(v_n) - C_2(v_n) = \emptyset$, $G[C_1(v_n)]$ has a perfect matching. □

From Lemma 9, when v_n is the considering vertex in the algorithm with $D(v_n) = 1$ and $C_1(v_n) - C_2(v_n) = \emptyset$, it is enough to set $L(w) = 2$ for every vertex $w \in C_1(v_n)$.

Lemma 10. *Let v_n be a considering vertex in the last step of algorithm k-MPDB. If $D(v_n) = 1$ and $C_1(v_n) - C_2(v_n) \neq \emptyset$, let $V_p = \{x \mid C(x) \cap (C_1(v_n) - C_2(v_n)) \neq \emptyset\}$ and M be a maximum matching in $G[C_1(v_n) \cup V_p]$. For each vertex $v \in C_1(v_n) - V(M)$, take one of its children, say v', with $L(v') = 0$ into C_p. Then there is a minimum k-distance paired-dominating set S of G such that $V_p \cup C_p \subseteq S$ and $G[C_1(v_n) \cup V_p \cup C_p]$ has a perfect matching.*

Proof. By the inductive hypothesis, there is a minimum k-distance paired-dominating set S of G such that $S_n \subseteq S$ and $G[S'_n]$ has a perfect matching. If $D(v_n) = 1$ and $C_1(v_n) - C_2(v_n) \neq \emptyset$, then $S_{n+1} = S_n \cup V_p \cup C_p$ and $S'_{n+1} = S'_n \cup C_1(v_n) \cup V_p \cup C_p$. By the selection of V_p and C_p, it is obvious that $G[C_1(v_n) \cup V_p \cup C_p]$ has a perfect matching. Hence, it suffices to prove that $V_p \cup C_p \subseteq S$.

Since $C_1(v_n) \subseteq S_n \subseteq S$, let $C'_p = \{x \mid x \in S - C_1(v_n)$ is paired with some vertex in $C_1(v_n)\}$, then $|C'_p| \geq |C_1(v_n)| - |C_2(v_n)|$ as $C_1(v_n) - C_2(v_n)$ is an independent set. If $V_p \cap S = \emptyset$, then $S - C'_p \cup V_p \cup C_p$ is also a minimum k-distance paired-dominating set. If $V_p \cap S \neq \emptyset$, let $V'_p = V_p \cap S$. If $V'_p \subseteq C'_p$, then $S - C'_p \cup V_p \cup C_p$ is also a minimum k-distance paired-dominating set. If $V'_p \not\subseteq C'_p$, let $V''_p = V'_p - C'_p = \{x_1, x_2, \ldots, x_a\}$. Suppose that $y_i \in C_1(v_n)$ with $x_i y_i \in M$ and y'_i is a child of y_i with $L(y'_i) = 0$. As the label of y_i is $(1, 1)$, y'_i can always be found. Then $S' = S - C'_p \cup V_p \cup C_p \cup \{y'_1, y'_2 \ldots, y'_a\}$ is also a minimum k-distance paired-dominating set. □

From Lemma 10, when v_n is the considering vertex in the algorithm with $D(v_n) = 1$ and $C_1(v_n) - C_2(v_n) \neq \emptyset$, then put the father or a child of some vertex in $C_1(v_n) - C_2(v_n)$ into *kPD*.

From the above lemmas, we obtained the following theorem.

Theorem 11. *The algorithm k-MPDB can produce a minimum k-distance paired-dominating set of any block graph with the order at least two in linear time.*

Proof. From Lemmas 5–10, we obtain that there is a minimum k-distance paired-dominating set S of G such that $S_{n+1} \subseteq S$ and $G[S'_{n+1}]$ has a perfect matching. Obviously, the output *kPD* of the algorithm k-MPDB is S_{n+1}, Therefore, *kPD* is a minimum k-distance paired-dominating set of G. Note that we need find a maximum matching M in $G[C_1(v_i)]$ in the algorithm k-MPDB. But $G[C_1(v_i)]$ consists of some disjoint cliques in the block graph. Hence, the maximum matching M in $G[C_1(v_i)]$ can be found in liner time. Therefore, each vertex and edge is used in constant number, the algorithm will be terminated in linear time. □

Since block graphs contain trees, the algorithm k-MPDB can produce a minimum k-distance paired-dominating set in any tree. However, By the speciality of tree, we present a more simple algorithm.

Algorithm k-MPDT. Find a minimum k-distance paired-dominating set of a tree.
Input. A tree $T = (V, E)$ with a vertex ordering $v_1, v_2, \ldots, v_n$ $(n \geq 2)$ such that $d(v_i, v_n) \leq d(v_j, v_n)$ if $i < j$. Each vertex v_i has a label $(D(v_i), L(v_i)) = (0, 0)$.
Output. A minimum k-distance paired-dominating set *kPD* of T.
Method.

```
For i = 1 to n − 1 do
    If (D(v_i) = 0) then
        Let A(v_i) = {w ∈ N_k(v_i) − {F^k(v_i)} | {u | u ∈ C(w), L(u) = 1} ≠ ∅};
        If (A(v_i) = ∅) then
            L(F^k(v_i)) = 1;
            D(u) = 1 for every vertex u ∈ N^k[F^k(v_i)];
        endif
    endif
    If (D(v_i) = 1) then
        Let C_1(v_i) = {w | w ∈ C^k(v_i) and L(w) = 1};
        If (C_1(v_i) ≠ ∅), then
            L(w) = 2 for every vertex w ∈ C_1(v_i);
            Let {v_{i_1}, v_{i_2}, ..., v_{i_l}} be a subset of C^{k−1}(v_i)
            such that C(v_{i_j}) ∩ C_1(v_i) ≠ ∅ for 1 ≤ j ≤ l;
            For j = 1 to l do
                L(v_{i_j}) = 2;
                D(u) = 1 for every vertex u ∈ N^k[v_{i_j}];
                Take a vertex w ∈ C(v_{i_j}) ∩ C_1(v_i), for every vertex v ∈ C(v_{i_j}) ∩ C_1(v_i)−
                {w}, L(v') = 2 for some vertex v' ∈ C(v) with L(v') = 0;
            endfor
        endif
    endif
endfor
```

If ($D(v_n) = 0$) then
  $L(v_n) = 2$;
  $L(w) = 2$ for some vertex $w \in C(v_n)$ with $L(w) = 0$;
  $D(v_n) = 1$;
else
  Let $C_1(v_n) = \{w \mid w \in N^k[v_n]$ *and* $L(w) = 1\}$;
  $L(w) = 2$ for every vertex $w \in C_1(v_n)$;
  Let $C_2(v_n) = \{w \mid w \in C_1(v_n)$ *and* $w \in V(M)$, where M is a maximum matching in $G[C_1(v_n)]\}$.
  If ($C_1(v_n) - C_2(v_n) \neq \emptyset$) then
    For each vertex $w \in C_1(v_i) - C_2(v_i)$,
    If ($L(F(w)) \neq 2$ *and* $w \neq v_n$) then
      $L(F(w)) = 2$;
      $D(u) = 1$ for every vertex $u \in N^k[F(w)]$;
    else
      $L(v') = 2$ for some vertex $v' \in C(w)$ with $L(v') = 0$;
    endif
  endif
endif
Output $kPD = \{v \mid L(v) = 2\}$
end

Corollary 12. *The algorithm k-MPDT can produce a minimum k-distance paired-dominating set of any tree with the order at least two in linear time.*

4. Characterization of trees with the unique minimum *k*-distance paired-dominating sets

Tree is an important subclass of chordal graphs. In [3], authors gave a characterization of trees with the unique minimum paired-dominating set. In this section, we give a characterization of trees with the unique minimum k-distance paired-dominating set.

Theorem 13. *Let $T = (V, E)$ be a tree of order at least three. The set $S \subseteq V$ is the unique minimum k-distance paired-dominating set of T if and only if S is a k-distance paired-dominating set of T such that every vertex in S has a private k-neighbor with regard to S.*

Proof. Suppose that S is the unique minimum k-distance paired-dominating set of T. We want to show that every vertex in S has a private k-neighbor with regard to S. Suppose to the contrary that there is a vertex $u_1 \in S$ with $P_k(u_1, S) = \emptyset$. Let v_1 be the paired vertex of u_1 in S. If $N(v_1) - S \neq \emptyset$, let $w \in N(v_1) - S$, then $S - \{u_1\} \cup \{w\}$ is also a minimum k-distance paired-dominating set of T by $P_k(u_1, S) = \emptyset$. It is a contraction to the unique of S. Hence, we assume that $N(v_1) \subseteq S$. Let $u_1, v_1, u_2, v_2, \dots, u_k, v_k$ be a maximal length vertex sequence such that: (1) u_i, v_i are paired in S for $1 \leq i \leq k$; (2) $v_i u_{i+1} \in E$ for $1 \leq i \leq k-1$; (3) $N(v_i) \subseteq S$ for $1 \leq i \leq k-1$. So, either $N(v_k)$ is a subset of $\{u_1, v_1, u_2, v_2, \dots, u_k\}$ or there is a vertex $w \in N(v_k)$ with $w \notin S$. For the former case, $S - \{u_1, v_k\}$ is a smaller k-distance paired-dominating set of T, a contradiction. For the later case, $S - \{u_1\} \cup \{w\}$ is also a minimum k-distance paired-dominating set of T, a contraction.

For converse, let S be a k-distance paired-dominating set of T such that every vertex in S has a private k-neighbor with regard to S. We want to show that S is the unique minimum k-distance paired-dominating set of T. We prove this by induction on $n(T)$, the order of T. Since S has at least two vertices and every vertex in S has a private k-neighbor with regard to S, the longest path in T has length at least $2k+1$ and hence $n(T) \geq 2k+2$. Let $P:\ v_0, v_1, \dots, v_l$ be a longest path in T with $l \geq 2k+1$. If $l = 2k+1$, it is obvious that $\{v_k, v_{k+1}\}$ is the unique minimum k-distance paired-dominating set of T. This implies that the basis step $n(T) = 2k+2$ holds. Let T be a tree with $n(T) \geq 2k+3$. Assume now that the assertion holds for smaller value of $n(T)$ and $l \geq 2k+2$. We may further assume that T is rooted at v_l.

Fact 1. *v_k and v_{k+1} are paired in S.*

Proof. Since S is a k-distance paired-dominating set, v_0 is dominated by some vertex in S. Without loss of generality, we assume that $v \in S$ is the nearest vertex from v_0 and $v' \in S$ is the paired vertex of v. Obviously, v is a vertex in T_{v_k}, where T_{v_k} is the subtree rooted at v_k.

If $v \notin \{v_0, v_1, \dots, v_k\}$, then v' is a child of v due to the choice of v. In this case, v' has no private k-neighbor with regard to S, a contradiction. Thus $v \in \{v_0, v_1, \dots, v_k\}$. If $v = v_b$ with $b \in \{0, 1, \dots, k-1\}$, then either v or v' has no private k-neighbor with regard to S. It follows that $v = v_k$. If v' is a child of v_k, then v' has no private k-neighbor with regard to S. Hence $v' = v_{k+1}$. $\square$

Let v_c be the first vertex in $\{v_{k+2}, v_{k+3}, \dots, v_{2k+2}\}$ such that $V(T_{v_c}) - (N^k[v_{k+1}] \cup N^k[v_k]) \neq \emptyset$.

Fact 2. $V(T_{v_{c-1}}) \cap S = \{v_k, v_{k+1}\}$.

Proof. Suppose to the contrary that $v \in S \cap V(T_{v_{c-1}})$ is other than v_k, v_{k+1}. Let v' be its paired vertex in S. Since v_k and v_{k+1} are paired in S (By Fact 1), then either v or v' has no private k-neighbor with regard to S. It is a contradiction. □

Let $T' = T - T_{v_{c-1}}$ and $S' = S - \{v_k, v_{k+1}\}$. By Fact 2, $S' \subseteq V(T')$.

Fact 3. *S' is a k-distance paired-dominating set of T'.*

Proof. Let $N = V(T') \cap N^k[v_{k+1}]$. If $N = \emptyset$, it obvious that S' is a k-distance paired-dominating set of T'. Hence, we assume that $N \neq \emptyset$ and $x \in N$. Note that N contains those vertices near to v_c. We will show that x is dominated by some vertex in S'.

Choose a vertex w in $V(T_{v_c}) - V(T_{v_{c-1}})$ farthest to v_{k+1}. It is obvious that w is a leaf in T'. Since $V(T_{v_c}) - (N^k[v_{k+1}] \cup N^k[v_k]) \neq \emptyset$, w cannot be dominated by v_{k+1}. Assume that $u \in S'$ is a vertex nearest to w and $u' \in S'$ is the paired vertex of u. w is dominated by u, as w cannot be dominated by v_{k+1}.

If $u, u' \in V(T_{v_c})$, by the choice of w and u, w is a private k-neighbor of u with regard to S. Furthermore, u' is on the u-v_c path, for otherwise u' has no private k-neighbor with regard to S. As P is a longest path in T, $d(w, v_c) \leq d(v_0, v_c) = c \leq 2k + 2$. Then, $d(u', v_c) \leq d(v_{k+1}, v_c) \leq k$. Assume that v'_c is the neighbor of v_c on the v_c-u' path. If $x \in T_{v'_c}$, then either $d(u', x) \leq k$ or $d(u, x) \leq k$. If $x \notin V(T_{v'_c})$, then $d(u', x) = d(u', v_c) + d(v_c, x) \leq d(v_{k+1}, v_c) + d(v_c, x) = d(v_{k+1}, x) \leq k$. Thus x is dominated by u or u'. If one of u and u' is v_c, then it is obvious that x is dominated by v_c. If $u, u' \notin V(T_{v_c})$, with the similar argument of the first case, x is dominated by u or u'. Therefore, in any case, S' is a k-distance paired-dominating set of T'. □

By Fact 3, S' is a k-distance paired-dominating set of T'. Furthermore, every vertex in S' has a private k-neighbor with regard to S'. By inductive hypothesis, S' is the unique minimum k-distance paired-dominating set of T'. Let y be a private k-neighbor of v_{k+1} with regard to S. As S' is a k-distance paired-dominating set of T', $y \in T_{v_{c-1}}$. Let S_1 be any minimum k-distance paired-dominating set of T. We claim that $2 \leq |S_1 \cap V(T_{v_{c-1}})| \leq 4$.

Proof of the Claim. Since v_0 is dominated by some vertex in S_1, $S_1 \cap V(T_{v_{c-1}})$ have at least two vertices. Suppose to the contrary that $S_1 \cap V(T_{v_{c-1}})$ have more than four vertices. If $v_c \in S_1$ and its paired vertex is not in $T_{v_{c-1}}$, then $(S_1 - V(T_{v_{c-1}})) \cup \{v_k, v_{k+1}\}$ is a smaller k-distance paired-dominating set of T, a contradiction. If $v_c \in S_1$ and its paired vertex is v_{c-1}, then $(S_1 - V(T_{v_{c-1}})) \cup \{v_k, v_{k+1}, v_{c-1}\}$ is a smaller k-distance paired-dominating set of T, a contradiction. So we assume that $v_c \notin S_1$. There is a neighbor v'_c of v_c with $v'_c \notin S_1$, for otherwise $(S_1 - V(T_{v_{c-1}})) \cup \{v_k, v_{k+1}\}$ is a smaller k-distance paired-dominating set of T. As $S_1 \cap V(T_{v_{c-1}})$ have more than four vertices, $(S_1 - V(T_{v_{c-1}})) \cup \{v_k, v_{k+1}, v_c, v'_c\}$ is a smaller k-distance paired-dominating set of T. It is also a contradiction. □

Case 1 $|S_1 \cap V(T_{v_{c-1}})| = 2$.

Since v_0 is dominated by S_1, $|S_1 \cap V(T_{v_{k+1}})| \geq 2$. So, in this case, S_1 contains no vertex in $V(T_{v_{c-1}}) - V(T_{v_{k+1}})$. Hence, $S_1 - V(T_{v_{c-1}})$ is a k-distance paired-dominating set of T'. Furthermore, $S_1 - V(T_{v_{c-1}})$ is a minimum k-distance paired-dominating set of T'. If not, let D be a minimum k-distance paired-dominating set of T', then $D \cup \{v_k, v_{k+1}\}$ is a smaller k-distance paired-dominating set of T, a contradiction. Since S' is the unique minimum k-distance paired-dominating set of T', we have $S_1 - V(T_{v_{c-1}}) = S'$. Since S' cannot dominate the vertex y and S_1 contains no vertex in $V(T_{v_{c-1}}) - V(T_{v_{k+1}})$, it follows that $S_1 \cap V(T_{v_{c-1}}) = \{v_k, v_{k+1}\}$. Therefore, $S_1 = S$.

Case 2 $|S_1 \cap V(T_{v_{c-1}})| = 3$.

In this case, v_{c-1} must be paired with v_c in S_1. Similarly, there is a neighbor v'_c of v_c with $v'_c \notin S_1$. Then let $S_2 = S_1 - \{v_{c-1}\} \cup \{v'_c\}$. With the same argument in Case 1, we have $S_2 = S$. Hence, v'_c has a private k-neighbor z with regard to S_2 and the length of z-v_{c-1} path is $k + 1$. Therefore, the vertex z cannot be dominated by any vertex in S_1. It contradicts that S_1 is a k-distance paired-dominating set of T.

Case 3 $|S_1 \cap V(T_{v_{c-1}})| = 4$.

If $v_c \in S_1$, by $|S_1 \cap V(T_{v_{c-1}})| = 4$, its paired vertex is not in $T_{v_{c-1}}$. Let S_2= $(S_1 - V(T_{v_{c-1}})) \cup \{v_k, v_{k+1}\}$. Obviously, S_2 is a smaller k-distance paired-dominating set of T, a contradiction. Assume now that $v_c \notin S_1$. There is a neighbor $v'_c \neq v_{c-1}$ of v_c with $v'_c \notin S_1$, for otherwise $(S_1 - V(T_{v_{c-1}})) \cup \{v_k, v_{k+1}\}$ is a smaller k-distance paired-dominating set of T. Let $S_2 = (S_1 - V(T_{v_{c-1}})) \cup \{v_k, v_{k+1}, v_c, v'_c\}$. Then S_2 is also a minimum k-distance paired-dominating set of T. With the same argument in Case 1, we know that $S_2 = S$. Similarly, v_c has a private k-neighbor z with regard to S_2, which cannot be dominated by any vertex in S_1. It contradicts that S_1 is a k-distance paired-dominating set of T.

By the discussion above, we know that $S_1 \cap V(T_{v_{c-1}}) = \{v_k, v_{k+1}\}$ and $S_1 = S$. Therefore, S is the unique minimum k-distance paired-dominating set of T. □

Remark 14. The algorithm k-MPDT can be used to identify whether a given tree has the unique minimum k-distance paired-dominating set. For a given tree T, if the output *kPD* of the algorithm k-MPDT has the property that every vertex in *kPD* has a private k-neighbor with regard to *kPD*, then *kPD* is the unique minimum k-distance paired-dominating set of T.

Acknowledgments

The authors thank the referees for giving many constructive suggestions.

References

[1] K.S. Booth, G.S. Lueker, Testing for consecutive ones property, interval graphs and graph planarity using *PQ*-tree algorithms, J. Comput. Syst. Sci. 13 (1976) 335–379.
[2] G.J. Chang, Algorithmic aspects of domination in graphs, in: D.Z. Du, P.M. Pardalos (Eds.), in: Handbook of Combinatorial Optimization, vol. 3, Kluwer Academic Publishers, Boston, MA, 1998, pp. 339–405.
[3] M. Chellali, T.W. Haynes, Trees with unique minimum paired-dominating sets, Ars Combin. 73 (2004) 3–12.
[4] L. Chen, C. Lu, Z. Zeng, Labelling algorithms for paired-domination problems in block and interval graphs, J. Comb. Optim. (2008), in press (doi:10.1007/s10878-008-9177-6).
[5] T.C.E. Cheng, L.Y. Kang, C.T. Ng, Paired domination on interval and circular-arc graphs, Discrete Appl. Math. 155 (2007) 2077–2086.
[6] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic press, New York, 1980.
[7] T.W. Haynes, S.T. Hedetniemi, P.J. Slater (Eds.), Fundamentals of Domination in Graphs, Marcel Dekker, New York, 1998.
[8] T.W. Haynes, S.T. Hedetniemi, P.J. Slater (Eds.), Domination in Graphs: Advanced Topics, Marcel Dekker, New York, 1998.
[9] T.W. Haynes, P.J. Slater, Paired-domination in graphs, Networks 32 (1998) 199–206.
[10] M.A. Henning, Graphs with large paired-domination number, J. Comb. Optim. 13 (2007) 61–78.
[11] P. Hell, S. Klein, L.T. Nogueira, F. Protti, Partitioning chordal graphs into independent sets and cliques, Discrete Appl. Math. 141 (2004) 185–194.
[12] L.Y. Kang, M.Y. Sohu, T.C.E. Cheng, Paired-domination in inflated graphs, Theoret. Comput. Sci. 320 (2004) 485–494.
[13] H. Qiao, L.Y. Kang, M. Caedei, D.Z. Du, Paired-domination of trees, J.Global Optim. 25 (2003) 43–54.
[14] J. Raczek, Distance paired domination numbers of graphs, Discrete Math. 308 (2008) 2473–2483.
[15] P.D. Sylvain Gravier, M.A. Henning, Paired-domination in generalized claw-free graphs, J. Comb. Optim. 14 (2007) 1–7.
[16] D.B. West, Introduction to Graph Theory, 2nd ed., Prentice Hall, Inc, NJ, 2001.