

On the Power of BFS to Determine a Graph's Diameter

Derek G. Corneil

Department of Computer Science, University of Toronto, Toronto, Ontario, Canada

Feodor F. Dragan

Department of Computer Science, Kent State University, Kent, Ohio

Ekkehard Köhler

Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

Recently, considerable effort has been spent on showing that Lexicographic Breadth First Search (LBFS) can be used to determine a tight bound on the diameter of graphs from various restricted classes. In this paper, we show that, in some cases, the full power of LBFS is not required and that other variations of Breadth First Search (BFS) suffice. The restricted graph classes that are amenable to this approach all have a small constant upper bound on the maximum-sized cycle that may appear as an induced subgraph. We show that, on graphs that have no induced cycle of size greater than k , BFS finds an estimate of the diameter that is no worse than $\text{diam}(G) - \lfloor k/2 \rfloor$. © 2003 Wiley Periodicals, Inc.

Keywords: graph's diameter approximation; breadth first search; linear-time algorithm; k -chordal graphs; AT-free graphs

1. INTRODUCTION

Recently, considerable attention has been given to the problem of developing fast and simple algorithms for various classical graph problems. The motivation for such algorithms stems from our need to solve these problems on very large input graphs; thus, the algorithms must be not only fast, but also easily implementable.

Determining the diameter of a graph is a fundamental and seemingly quite time-consuming operation. For arbitrary graphs (with n vertices and m edges), the current fastest algorithm runs in $O(nm)$ time, which is too slow to

be practical for very large graphs. This naive algorithm examines each vertex in turn and performs a Breadth First Search (BFS) starting at the chosen vertex. Such a sweep starting at vertex x immediately determines $\text{ecc}(x)$, the eccentricity of vertex x . Recall that the *eccentricity* of vertex x , $\text{ecc}(x) = \max_{y \in V} d(x, y)$, where $d(x, y)$ denotes the distance between x and y ; the *diameter* of G equals the maximum eccentricity of any vertex in V . It is clear that this algorithm actually computes the entire distance matrix; clearly knowing the distance matrix immediately yields the diameter of the graph.

For dense graphs, the best result known is by Seidel [14], who showed that the all pairs shortest path problem (and, hence, the diameter problem) can be solved in $O(M(n)\log n)$ time, where $M(n)$ denotes the time complexity for fast matrix multiplication involving small integers only. The current best matrix multiplication algorithm is due to Coppersmith and Winograd [4] and has an $O(n^{2.376})$ time bound. Unfortunately, fast matrix multiplication algorithms are far from being practical and suffer from large hidden constants in the running time bound.

Note that no efficient algorithm for the diameter problem in general graphs, avoiding the computation of the whole distance matrix, has been designed. Thus, the question of whether a graph's diameter can be computed more easily than can the whole distance matrix still remains open.

Clearly, performing a BFS starting at a vertex of maximum eccentricity easily produces the graph's diameter. Thus, one way to approximate the diameter of a graph is to find a vertex of high eccentricity; this is the approach taken in this paper. This is not, however, the only approach. For example, Aingworth et al. [1] obtained a ratio of $2/3$ approximation to the diameter in time $O(m\sqrt{n\log n} + n^2\log n)$. Note that a ratio of $1/2$ can easily be achieved by choosing an arbitrary vertex (the eccentricity of any vertex is at least one-half the diameter of the graph) and performing a BFS starting at this vertex. It follows also from the

Received February 2003; accepted July 2003

Correspondence to: F. F. Dragan; e-mail: dragan@cs.kent.edu

Contract grant sponsor: Natural Science and Engineering Research Council of Canada

An expanded abstract of this paper appeared in the Proceedings of the LATIN 2002, S. Rajsbaum (Editor), LNCS 2286, Springer, Berlin, 2002, pp. 209–223

© 2003 Wiley Periodicals, Inc.

results in [1, 7] (see also paper [16] which surveyed recent results related to the computation of exact and approximate distances in graphs) that the diameter problem in unweighted, undirected graphs can be solved in $\tilde{O}(\min\{n^{3/2}m^{1/2}, n^{7/3}\})$ time with an additive error of at most 2 without matrix multiplication. Here, $\tilde{O}(f)$ means $O(f \text{ polylog}(n))$. The motivation behind the work of Aingworth et al. was to find a fast, easily implementable algorithm (they avoided using matrix multiplication), a motivation that we share.

Our approach is to examine the naive algorithm of choosing a vertex, performing some version of BFS from this vertex, and then showing a nontrivial bound on the eccentricity of the last vertex visited in this search. In fact, this algorithm is one of the “classical” algorithms in graph theory; if one restricts one’s attention to trees, then this algorithm produces a vertex of maximum eccentricity (see, e.g., [10]). This approach has already received considerable attention. (In the following, we let v denote the vertex that appears last in a particular search; the definition of the various searches and families of graphs will be presented in the next section.) For example, Dragan et al. [9] showed that if LBFS is used for chordal graphs then $\text{ecc}(v) \geq \text{diam}(G) - 1$, whereas for interval graphs, $\text{ecc}(v) = \text{diam}(G)$. It is clear from the work of Corneil et al. [5], that by using LBFS on AT-free graphs, one has $\text{ecc}(v) \geq \text{diam}(G) - 1$. Dragan [8], again using LBFS, showed that $\text{ecc}(v) \geq \text{diam}(G) - 2$ for HH-free graphs, $\text{ecc}(v) \geq \text{diam}(G) - 1$ for HHD-free graphs, and $\text{ecc}(v) = \text{diam}(G)$ for graphs that are both HHD-free and AT-free.

It is interesting to note that Corneil et al. [5] looked at double-sweep LBFSs (i.e., start an LBFS from a vertex that is last in a previous arbitrarily chosen LBFS) on chordal and AT-free graphs. They provided a forbidden subgraph structure on graphs where $\text{ecc}(v) = \text{diam}(G) - 1$. They also presented both chordal and AT-free graphs, where for no c the c -sweep LBFS algorithm is guaranteed to find a vertex of maximum eccentricity. Furthermore, they showed that for any c there is a graph G where $\text{ecc}(v) \leq \text{diam}(G) - c$, where v is the vertex visited last in a 2-sweep LBFS. This graph G , however, has a large induced cycle whose size depends on c .

These results motivate a number of interesting questions:

- Is it an inherent property of LBFS to end in a vertex of high eccentricity for the various restricted graph families mentioned above? What happens if we use other variants of BFS?
- Why do AT-free and chordal graphs, two families with very disparate structure, exhibit such similar behavior with respect to the efficacy of LBFS to find vertices of high eccentricity?
- Although LBFS “fails” to find vertices of high eccentricity for graphs in general, all known examples that exhibit such failure have large induced cycles. If we bound the size of the largest induced cycle, can we get a bound on the eccentricity of the vertex that appears last in an LBFS?
- If the previous question is answered in the affirmative, is the full power of LBFS needed? What happens if we just use BFS?

This paper addresses these questions. In the next section, we present the various forms of BFS and define the graph theoretic terminology used throughout the paper. In Section 3, we examine the behavior of the different versions of BFS on various restricted graph families. We establish some new bounds and show, by example, that all stated bounds on $\text{ecc}(v)$ are tight. In Section 4, we examine families of graphs where the size of the largest induced cycle is bounded and show that BFS does succeed in getting vertices of high, with respect to k , eccentricity. The paper concludes with some final remarks in Section 5.

2. NOTATION AND DEFINITIONS

First, we formalize the notion of BFS and then discuss various variations of it. We caution the reader that there is some confusion in the literature between BFS and what we call LL, defined below. In defining the various versions of BFS, we are only concerned with identifying the last vertex visited by the search; straightforward modifications produce the list of vertices in the order that they are visited by the search. It should be noted that none of the orderings are unique; instead, each search identifies one of the possible end-vertices.

Algorithm BFS: Breadth First Search

Input: graph $G(V, E)$ and vertex u

Output: vertex v , the last vertex visited by a BFS starting at u

Initialize queue Q to be $\{u\}$ and mark u as “visited.”

while $Q \neq \emptyset$ **do**

 Let v be the first vertex of Q and remove it from Q .

 Each unvisited neighbor of v is added to the end of Q and marked as “visited.”

Note that the above algorithm can easily be modified to obtain the “layers” of V with respect to u . In particular, for each $0 \leq i \leq \text{ecc}(u)$, the i -th layer of V with respect to u is denoted $L_i(u) = \{v : d(u, v) = i\}$. This motivates the next algorithm, LL.

Algorithm LL: Last Layer

Input: graph $G(V, E)$ and vertex u

Output: vertex v , a vertex in the last layer of u

Run BFS to get the layering of V with respect to u .

Choose v to be an arbitrary vertex in the last layer.

Clearly, any vertex returned by BFS can also be returned by LL; the converse is not true as shown by the graph in Figure 1.

Now, we modify this algorithm to obtain a vertex in the last layer that has minimum degree with respect to the vertices in the previous layer.

Algorithm LL+: Last Layer, Minimum Degree

Input: graph $G(V, E)$ and vertex u

Output: vertex v , a vertex in the last layer of u , that has

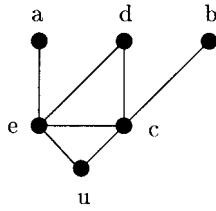


FIG. 1. No BFS starting at u can return vertex d .

minimum degree with respect to the vertices in the previous layer

Run BFS to get the layering of V with respect to u . Choose v to be an arbitrary vertex in the last layer that has minimum degree with respect to the vertices in the previous layer.

Finally, we introduce Lexicographic Breadth First Search (LBFS). This search paradigm was discovered by Rose et al. [13] and was shown to yield a simple linear time algorithm for the recognition of chordal graphs. In light of the great deal of work currently being done on LBFS, it is somewhat surprising that interest in LBFS lay dormant for quite a while after [13] appeared.

Algorithm LBFS: Lexicographic Breadth First Search
Input: graph $G(V, E)$ and vertex u
Output: vertex v , the last vertex visited by an LBFS starting at u

Assign label \emptyset to each vertex in V .
for $i = n$ **downto** 1 **do**
 Pick an unmarked vertex v with the largest (with respect to lexicographic order) label.
 Mark v “visited”.
 For each unmarked neighbor y of v , append i to the label of y .

If vertex a is removed from the graph in Figure 1, we have a graph where a vertex, namely d , can be visited last by a BFS from u but by no LBFS from u .

We now turn to the definitions of the various graph families introduced in the previous section. A graph is *chordal* if it has no induced cycle of size greater than 3. An *interval* graph is the intersection graph of intervals of a line. Lekkerkerker and Boland [11] defined an *asteroidal triple* to be a triple of vertices such that between any two there is a path that avoids the neighborhood of the third and showed that a graph is an interval graph iff it is both chordal and asteroidal triple-free (AT-free). A *claw* is the complete bipartite graph $K_{1,3}$, a *hole* is an induced cycle of length greater than 4, a *house* is a 4-cycle with a triangle added to one of the edges of the C_4 , and a *domino* is a pair of C_4 s sharing an edge. A graph is *HH-free* if it contains no induced houses or holes and is *HHD-free* if it contains no induced houses, holes, or dominos. Finally, to capture the

notion of “small” induced cycles, we define a graph to be *k-chordal* if it has no induced cycles of size greater than k . Note that chordal graphs are precisely the 3-chordal graphs and that AT-free graphs are 5-chordal.

We define the *disk* of radius r centered at u to be the set of vertices of distance at most r to u , that is, $D_r(u) = \{v \in V : d(u, v) \leq r\} = \cup_{i=0}^r L_i(u)$. A pair of vertices x, y of a graph G is called a *diametral pair* if $d(x, y) = \text{diam}(G)$. A vertex s of G is *universal* if $\text{ecc}(s) = 1$, that is, $N[s] = V$, and it is *nonuniversal* otherwise.

3. RESTRICTED FAMILIES OF GRAPHS

We now see how the four search algorithms mentioned in the previous section behave on the following families of graphs: chordal, AT-free, {AT, claw}-free, interval, and hole-free. The results are summarized in the following table. In this table, the references refer to the paper where the lower bound was established; a [*] indicates that the result is new. A figure reference refers to the appropriate figure where it is shown that the lower bound is tight. In each of the figures, the vertex pair a, b forms a diametral pair, that is, $d(a, b) = \text{diam}(G)$. Below each figure, a BFS, LBFS, LL, or LL+ ordering is given that achieves the corresponding bounds; vertex u is always the start-vertex and v the end-vertex of the appropriate search; different BFS-layers are separated by a |. Note that for “= D ” entries we are assuming that the search starts at a nonuniversal vertex.

Graph class	LL	LL+	BFS	LBFS
Chordal graphs	$\geq D - 2$ [3] Fig. 4	$\geq D - 2$ [3] Fig. 5	$\geq D - 1$ [*] Fig. 2	$\geq D - 1$ [9] Fig. 6
AT-free graphs	$\geq D - 2$ [*] Fig. 3	$\geq D - 1$ [*] Fig. 7	$\geq D - 2$ [*] Fig. 3	$\geq D - 1$ [5] Fig. 7
{AT, claw}-free graphs	$\geq D - 1$ [*] Fig. 2	$= D$ [*]	$\geq D - 1$ [*] Fig. 2	$= D$ [2]
Interval graphs	$\geq D - 1$ [*] Fig. 2	$= D$ [*]	$\geq D - 1$ [*] Fig. 2	$= D$ [9]
Hole-free graphs	$\geq D - 2$ [*] Fig. 8	$\geq D - 2$ [*] Fig. 8	$\geq D - 2$ [*] Fig. 8	$\geq D - 2$ [*] Fig. 8

In what follows, we illustrate the types of techniques that are used to establish lower bounds in the table. First, we show that $\text{ecc}(v) \geq \text{diam}(G) - 1$ holds for chordal graphs when BFS is used. This result subsumes the result shown in [9] that this lower bound holds when LBFS is used. Then, we prove our new results for interval graphs and AT-free graphs. Hole-free graphs and general k -chordal graphs are considered in Section 4.

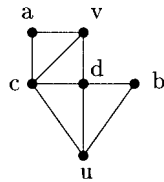


FIG. 2. BFS: u|bcd|av.

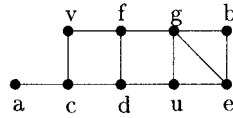


FIG. 3. BFS: u|dgc|cfb|av.

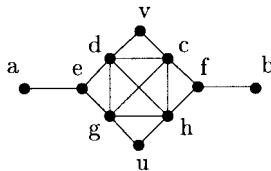


FIG. 4. LL: u|gh|edcf|abv.

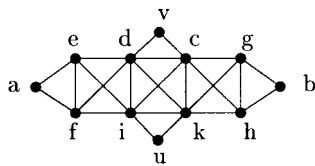


FIG. 5. LL+: u|ik|fedcgh|abv.

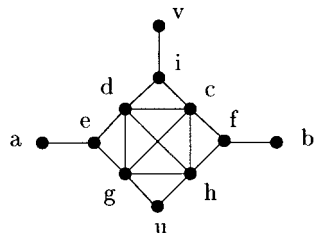


FIG. 6. LBFS: u|gh|edcf|aib|v.

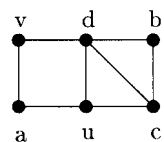


FIG. 7. LBFS: u|cda|bv.

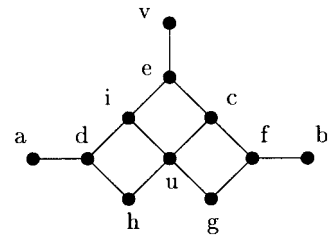


FIG. 8. LBFS: u|ghic|fde|bav.

given the next available number. The last vertex visited, v , is given the number 1. Thus, BFS may be seen to generate a rooted tree T with vertex u as the root. A vertex y is the father in T of exactly those neighbors in G which are inserted into the queue when y is removed.

An ordering $\sigma = [v_1, v_2, \dots, v_n]$ of the vertex set of a graph G generated by a BFS will be called a *BFS-ordering* of G . Let $\sigma(y)$ be the number assigned to a vertex y in this BFS-ordering σ . Denote also by $f(y)$ the father of a vertex y with respect to σ . The following properties of a BFS-ordering will be used in what follows. Since all layers of V considered here are with respect to u , we will frequently use the notation L_i instead of $L_i(u)$:

- (P1) If $y \in L_q$ ($q > 0$), then $f(y) \in L_{q-1}$ and $f(y)$ is the vertex from $N(y) \cap L_{q-1}$ with the largest number in σ .
- (P2) If $x \in L_i$, $y \in L_j$ and $i < j$, then $\sigma(x) > \sigma(y)$.
- (P3) If $x, y \in L_j$ and $\sigma(x) > \sigma(y)$, then either $\sigma(f(x)) > \sigma(f(y))$ or $f(x) = f(y)$.
- (P4) If $x, y, z \in L_j$, $\sigma(x) > \sigma(y) > \sigma(z)$ and $f(x)z \in E$, then $f(x) = f(y) = f(z)$ [in particular, $f(x)y \in E$].

Henceforth, $P(x, y)$ denotes a path connecting vertices x and y . The proof of the following lemma is omitted since it is a particular case of Lemma 9.

Lemma 1. *If vertices a and b of a disk $D_r(u)$ of a chordal graph are connected by a path $P(a, b)$ outside of $D_r(u)$ [i.e., $P(a, b) \cap D_r(u) = \{a, b\}$], then a and b must be adjacent.*

Let σ be a BFS-ordering of a chordal graph G started at a vertex u . Let also $P(a, b) = (a = x_1, x_2, \dots, x_{k-1}, x_k = b)$ be a shortest path of G connecting vertices a and b . We say that $P(a, b)$ is a *leftmost shortest path* if the sum $\sigma(x_1) + \sigma(x_2) + \dots + \sigma(x_{k-1}) + \sigma(x_k)$ is the largest among all shortest paths connecting a and b .

Lemma 2. *Let x and y be two arbitrary vertices of a chordal graph G . Every leftmost shortest path $P(x, y)$ between x and y can be decomposed (see Fig. 9) into three shortest subpaths $P_x = (x = x_1, x_2, \dots, x_l)$ (called the vertical x -subpath), $P_y = (y = y_1, y_2, \dots, y_k)$ (called the vertical y -subpath), and $P(x_l, y_k)$ (called the horizontal subpath) such that*

3.1. Chordal Graphs

First, we comment on the BFS algorithm. In particular, we may regard BFS as having produced a numbering from n to 1 in decreasing order of the vertices in V where vertex u is numbered n . As a vertex is placed on the queue, it is

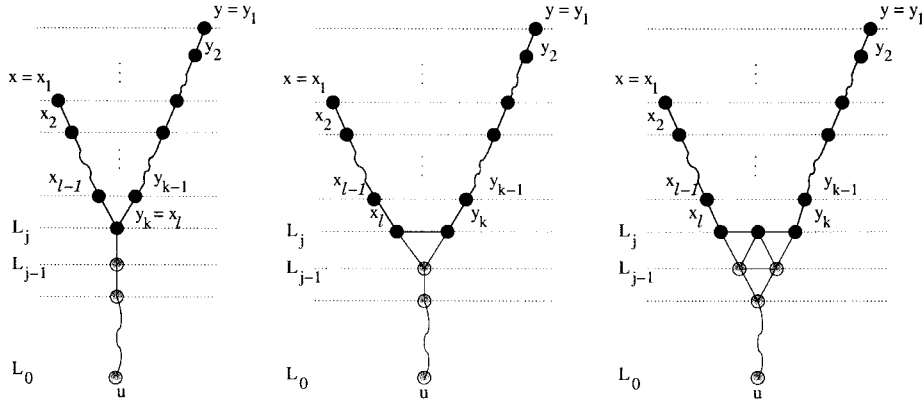


FIG. 9. The structure of leftmost shortest paths in chordal graphs.

1. $P(x_l, y_k) \subseteq L_j$ for some $j \in \{0, 1, \dots, \text{ecc}(u)\}$, and $d(x_l, y_k) \leq 2$ (i.e., x_l and y_k either coincide or are adjacent or have a common neighbor in $G[L_j] \cap P(x_l, y_k)$);
2. $x_{l-i} \in L_{j+i}$ for $0 \leq i \leq l-1$;
3. $y_{k-i} \in L_{j+i}$ for $0 \leq i \leq k-1$.

Proof. First, we prove that $|P(x, y) \cap L_i| \leq 3$ for any $i = 1, 2, \dots, \text{ecc}(u)$. Assume that the intersection of $P(x, y)$ and a layer L_q , for some index q , contains at least four vertices. Let a, b, c, d be the first four vertices of $P(x, y) \cap L_q$ on the way from x to y . We claim that $ab, bc, cd \in E$.

If $ab \notin E$, then, by Lemma 1, subpath $P(a, b)$ of the path $P(x, y)$ is completely contained in disk $D_q(u)$. In particular, the neighbor b' of b on $P(a, b)$ belongs to $D_{q-1}(u)$. Using the same arguments, we conclude that $bc \in E$ or subpath $P(b, c)$ of the path $P(x, y)$ is contained in $D_q(u)$. If $bc \in E$, then a neighbor v of c in L_{q-1} must be adjacent to b' (by Lemma 1). Since $\sigma(v) > \sigma(b)$, we get a contradiction to $P(x, y)$ being a leftmost path [we can replace vertex b of $P(x, y)$ with v and get a shortest path between x and y with larger sum]. If $bc \notin E$, then the neighbor b'' of b on $P(b, c)$ is also contained in $D_{q-1}(u)$. By Lemma 1, vertices b' and b'' must be adjacent, but this is impossible since $P(x, y)$ is a shortest path.

Thus, vertices a and b have to be adjacent. If vertices b and c are not adjacent, then the neighbor b'' of b on $P(b, c)$ belongs to $D_{q-1}(u)$ and, by Lemma 1, it must be adjacent to any neighbor w of a in L_{q-1} . Since $\sigma(w) > \sigma(b)$ holds, again we have a contradiction to $P(x, y)$ being a leftmost path. Consequently, vertices b and c are also adjacent. In exactly the same way, one can show that c and d have to be adjacent. Note also that the adjacency of a with b and b with c is proved without using the existence of the vertex d .

We have now $ab, bc, cd \in E$ and the induced path (a, b, c, d) is a leftmost shortest path [as a subpath of the leftmost shortest path $P(x, y)$]. Consider neighbors a' and d' in L_{q-1} of a and d , respectively. Since $\sigma(a') + \sigma(d') > \sigma(b) + \sigma(c)$ and the path (a, b, c, d) is leftmost and shortest, vertices a' and d' can neither coincide nor be

adjacent. But, then, we get a path (a', a, b, c, d, d') connecting two nonadjacent vertices of $D_{q-1}(u)$ outside of the disk. This contradiction to Lemma 1 shows that $|P(x, y) \cap L_i| \leq 3$ for any $i = 1, 2, \dots, \text{ecc}(u)$.

Now, let $|P(x, y) \cap L_q| = 3$ and let a, b, c be the vertices of $P(x, y) \cap L_q$ on the way from x to y . It follows from the discussion above that $ab, bc \in E$ and both the neighbor a' of a on subpath $P(x, a)$ and the neighbor c' of c on subpath $P(c, y)$ (if they exist) belong to layer L_{q+1} . If, for example, $a' \in L_{q-1}$, then, by Lemma 1, a' must be adjacent to any neighbor v of b in L_{q-1} . Again, since $\sigma(v) > \sigma(a)$, we have a contradiction to $P(x, y)$ being a leftmost path.

Furthermore, if for some index q , $P(x, y) \cap L_q = \{a, b\}$, and $P(x, y) = (x, \dots, a', a, b, b', \dots, y)$, then both a' and b' belong to layer L_{q+1} .

Summarizing, we conclude that, while moving from x to y along the path $P(x, y)$, we can have only one horizontal edge or only one pair of consecutive horizontal edges. Here, by horizontal edge, we mean an edge with both end-vertices from the same layer. All other vertices of the path $P(x, y)$ belong to higher layers. ■

Having the structure of a leftmost shortest path established, we can now prove the main result for chordal graphs. In presenting a leftmost shortest path, we use “ l ”s to differentiate the appropriate subpaths.

Theorem 3. *Let v be the vertex of a chordal graph G last visited by a BFS. Then, $\text{ecc}(v) \geq \text{diam}(G) - 1$.*

Proof. Let x, y be a pair of vertices such that $d(x, y) = \text{diam}(G)$, and consider two leftmost shortest paths

$$P(x, v) = (x = x_1, x_2, \dots, x_{l-1}/x_l, \dots, v'_l/v'_{l-1}, \dots, v'_2, v'_1 = v)$$

and

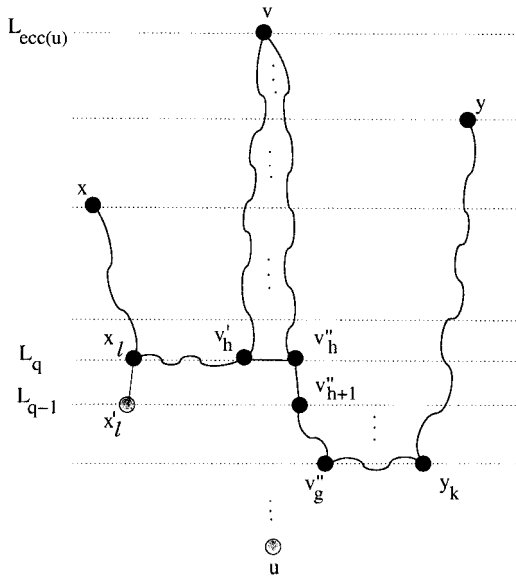


FIG. 10. Rightmost shortest paths $P(x, v)$ and $P(y, v)$.

$$P(y, v) = (y = y_1, y_2, \dots, y_{k-1}/y_k, \dots, v''_g/v''_{g-1}, \dots, v''_2, v''_1 = v)$$

connecting vertex v with x and y , respectively (see Fig. 10). By Lemma 2, each of these paths consists of two (perhaps of length 0) vertical subpaths and one horizontal path of length not greater than 2. Assume, without loss of generality, that $h \leq g$ and let $x_l, \dots, v'_h \in L_q$. Since $v \in L_{ecc(u)}$, we also have $l \leq h$ and $k \leq g$. By Lemma 1, vertices v'_h, v''_h in L_q either coincide or are adjacent. Note that, if $d(x_l, v''_h) \leq 1$, then $d(x, y) \leq d(x, x_l) + 1 + d(v''_h, y) \leq d(v, v''_h) + 1 + d(v''_h, y) = d(v, y) + 1 \leq ecc(v) + 1$, that is, $ecc(v) \geq d(x, y) - 1 = diam(G) - 1$, and we are done.

Hence, we may assume that $d(x_l, v''_h) \geq 2$ and, therefore, $x_l \neq v''_h$.

We distinguish between two cases. The first one is simple. Only for the second case we will need to use the special properties of a BFS-ordering.

CASE $g > h$.

In this case, there exists a vertex v''_{h+1} in the intersection $P(y, v) \cap L_{q-1}$. Consider also a neighbor x'_l of x_l in L_{q-1} . Since vertices x'_l and v''_{h+1} are connected by path $(x'_l, x_l, \dots, v'_h, v''_h, v''_{h+1})$ outside the disk $D_{q-1}(u)$, by Lemma 1, they are adjacent if they do not coincide. Hence, $d(x_l, v''_{h+1}) \leq 2$ and, therefore, $d(x, y) \leq d(x, x_l) + 2 + d(v''_{h+1}, y) \leq d(v, v''_h) + 2 + d(v''_{h+1}, y) = d(v, y) + 1 \leq ecc(v) + 1$, that is, again $ecc(v) \geq d(x, y) - 1 = diam(G) - 1$.

CASE $g = h$.

From the discussion above (now, since $g = h$, we have a symmetry), we may assume that $d(y_k, v'_h) \geq 2$ and $y_k \neq v'_h$. Consider neighbors x'_l and y'_k in L_{q-1} of vertices x_l and y_k , respectively [see Fig. 11(a)]. By Lemma 1, they are adjacent if they do not coincide, that is, $d(x_l, y_k) \leq 3$. Now, if at least one of the equalities $d(x_l, v'_h) = 2, d(y_k, v''_h) = 2$ holds, then we are done. Indeed, if, for example, $d(x_l, v'_h) = 2$, then $d(x, v) = d(x, x_l) + 2 + d(v'_h, v)$ and, therefore, $d(x, y) \leq d(x, x_l) + d(x_l, y_k) + d(y_k, y) \leq d(x, x_l) + 3 + d(v'_h, v) = d(x, v) + 1 \leq ecc(v) + 1$, implying that $ecc(v) \geq d(x, y) - 1 = diam(G) - 1$.

So, we may assume that $x_l v'_h, y_k v''_h \in E$. Moreover, since $d(y_k, v'_h) \geq 2$, vertices v'_h and v''_h cannot coincide, that is, they are adjacent. If $l < h$ or $k < h$ or $d(x_l, y_k) < 3$, again, we will get $d(x, y) \leq ecc(v) + 1$ by comparing distances $d(v, y) = h + k - 1, d(v, x) = h + l - 1$ with $d(x, y) \leq l - 1 + d(x_l, y_k) + k - 1 \leq l + 1 + k$. Thus, we

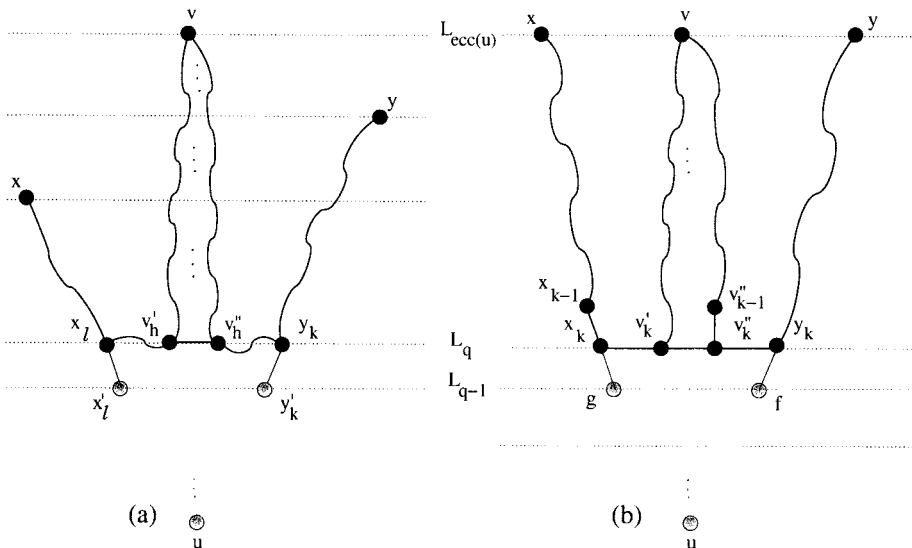


FIG. 11. (a) Horizontal subpaths of $P(x, v)$ and $P(y, v)$ are in the same layer; (b) paths $P(x, v)$ and $P(y, v)$ have similar shape and the same length $2k - 1$.

arrive at a situation when $l = k = h$, $x_l v'_h, y_k v''_h, v'_h v''_h \in E$ and $d(x_l, y_k) = 3$, $d(v, x) = d(v, y) = 2k - 1$ [see Fig. 11(b)]. We may also assume that $d(x, y) = 2k + 1$, since, otherwise, $d(x, y) \leq 2k = d(v, y) + 1 \leq \text{ecc}(v) + 1$ and we are done. We show that this final configuration [with $d(x, y) = 2k + 1$] is impossible because of the properties of BFS-orderings.

Assume, without loss of generality, that $\sigma(y_k) > \sigma(x_k)$ and consider the fathers $f = f(y_k)$, $g = f(x_k)$ of y_k and x_k , respectively. Since $d(x_k, y_k) = 3$, we have $f \neq g$ and $fx_k, gy_k \notin E$. By Lemma 1 and property (P3) of BFS-ordering, vertices f and g are adjacent and $\sigma(f) > \sigma(g)$. Chordal graphs cannot contain an induced cycle of length greater than 3. Therefore, in the cycle formed by $g, x_k, v'_k, v''_k, y_k, f$, at least chords gv'_k and fv''_k must be present. Since for the father $f(v''_k)$ of v''_k we have $\sigma(f(v''_k)) \geq \sigma(f) > \sigma(g)$, inequality $\sigma(v''_k) > \sigma(x_k)$ must hold [here, we used properties (P1) and (P3) of BFS-orderings]. We will need the inequality $\sigma(v''_k) > \sigma(x_k)$ later to get our final contradiction.

Now, consider vertices x_{k-1} and v''_{k-1} . We claim that $\sigma(v''_{k-1}) < \sigma(x_{k-1})$. Assume that this is not the case, and let j ($j \in \{1, 2, \dots, k-2\}$) be the largest index such that $\sigma(v''_j) < \sigma(x_j)$ [recall that $\sigma(v''_1) = \sigma(v) = 1 < \sigma(x) = \sigma(x_1)$]. Then, $\sigma(v''_{j+1}) > \sigma(x_{j+1})$ holds, and since $j \leq k-2$ and $d(v, x) = 2k-1$, we obtain $d(x_j, v''_j) \geq 5$ [because of $2k-1 = d(v, x) \leq d(v, v''_j) + d(v''_j, x_j) + d(x_j, x) = 2(j-1) + d(v''_j, x_j) \leq 2(k-3) + d(v''_j, x_j) = 2k-6 + d(v''_j, x_j)$]. Consider the father $t = f(x_j)$ of x_j . From the distance requirement and properties of BFS-orderings, we conclude that $t \neq v''_{j+1}$, $t \neq x_{j+1}$, and $\sigma(t) > \sigma(v''_{j+1}) > \sigma(x_{j+1})$. Moreover, vertex t has to be adjacent to x_{j+1} (by Lemma 1), but cannot be adjacent to x_{j+2} [since $\sigma(t) > \sigma(x_{j+1})$ and the path $P(x, v)$ is leftmost]. Consider now the father $z = f(t)$ of the vertex t . It is adjacent to x_{j+2} , by Lemma 1, and has to be adjacent to x_{j+1} , to avoid an induced cycle $(z, t, x_{j+1}, x_{j+2}, z)$ of length 4. Applying property (P4) to $\sigma(t) > \sigma(v''_{j+1}) > \sigma(x_{j+1})$ and $zx_{j+1} \in E$, we get $zv''_{j+1} \in E$, which is impossible since $d(x_j, v''_j) \geq 5$. This contradiction shows that, indeed, the inequality $\sigma(v''_{k-1}) < \sigma(x_{k-1})$ must hold.

So, we have $\sigma(v''_{k-1}) < \sigma(x_{k-1})$ and $\sigma(v''_k) > \sigma(x_k)$. We repeat our arguments from the previous paragraph considering index $k-1$ instead of j . [The only difference is that now we do not have the vertex $x_{j+2} = x_{k+1}$ on the path $P(x, v)$.] Again, consider the father $t = f(x_{k-1})$ of x_{k-1} . Clearly, $t \neq v'_k$. Since $d(x_k, y_k) = 3$, vertex t is not adjacent to y_k . Furthermore, t does not coincide with v''_k since Lemma 1 would require $x_k v''_k \in E$. Hence, $t \neq v''_k$ and, by property (P3), $\sigma(t) > \sigma(v''_k) > \sigma(x_k)$. Vertex t must be adjacent to x_k , by Lemma 1, but cannot be adjacent to v'_k , since the path $P(v, x)$ is leftmost and $\sigma(t) > \sigma(x_k)$. To avoid an induced cycle of length 4, vertex t is not adjacent to v''_k as well.

Consider also the father $z = f(t)$ of the vertex t (see Fig. 12). If $zy_k \in E$, then $d(x, y) \leq d(y, y_k) + 1 + d(z, x_{k-1}) + d(x_{k-1}, x) \leq k-1 + 1 + 2 + k-2 = 2k$, and a

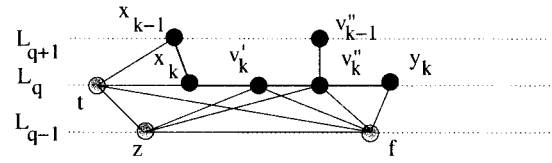


FIG. 12. Illustration of the proof of Theorem 3.

contradiction to the assumption $d(x, y) = 2k + 1$ arises. Therefore, z and y_k are not adjacent and, hence, $z \neq f$ (recall that f is the father of y_k and $fv''_k \in E, fx_k \notin E$). By Lemma 1, $zf \in E$. Since $\sigma(t) > \sigma(v''_k)$ and $fv''_k \in E$, by properties (P3) and (P1), we get $\sigma(z) \geq \sigma(f(v''_k)) \geq \sigma(f)$, that is, $\sigma(z) > \sigma(f)$. Consequently, $\sigma(t) > \sigma(y_k)$. Now, vertex z cannot be adjacent to x_k , since this would imply also the adjacency of z with y_k [by $\sigma(t) > \sigma(y_k) > \sigma(x_k)$ and property (P4)]. But, then, in the cycle $(z, t, x_k, v'_k, v''_k, f, z)$, only chords zv'_k, zv''_k, fv'_k, ft are possible, which are not enough to avoid an induced cycle of length greater than 3 in G . A contradiction to the chordality of G completes the proof of the theorem. ■

3.2. Interval Graphs

Now, we consider interval graphs and LL+. Recall that G is an *interval* graph if and only if G is chordal and AT-free [11].

Theorem 4. *Let G be an interval graph. Consider a BFS, starting at some nonuniversal vertex u of G , and let v be a vertex of the last BFS layer having the fewest neighbors in the previous BFS layer. Then, $\text{ecc}(v) = \text{diam}(G)$.*

Proof. Let L be the last, and L' , the second to last layers of a BFS started at vertex u . Suppose that x, y is a diametral pair of vertices of G with

$$\text{ecc}(v) < d(x, y). \quad (1)$$

Suppose, first, that $\text{diam}(G) = 2$. Since u is not universal, u is not contained in L' . Consequently, for any vertex $v \in L$, we have $\text{ecc}(v) = \text{diam}(G)$, contradicting (1). Hence, from now on, we can assume that $\text{diam}(G) > 2$ and, thus, at most one of x and y is adjacent to v .

CASE 1. Both x and y are not adjacent to v .

Let $P(x, v)$ be a path of shortest length, connecting x and v . Vertex y cannot have a neighbor on $P(x, v)$, because $d(x, y) > d(x, v)$, and neither x nor v is adjacent to y . Similarly, x has no neighbor on shortest path $P(y, v)$, connecting y and v . In particular, x, y, v form an independent triple of G .

Since G is AT-free, every (x, y) -path (path connecting x and y) has to contain a neighbor of v ; in other words, removing $N[v]$ leaves x and y in different connected components. This implies that at least one of x and y , say x , has to be contained in L , since, otherwise, there is a path

between x and y in G (via u) that does not contain a neighbor of v .

By definition of BFS, vertex x has to have neighbors in L' . Let $U_x = N[x] \cap L'$. If $y \notin L$, v is adjacent to all vertices from U_x because, otherwise, there is an (x, y) -path (via U_x and u) not containing a neighbor of v , contradicting the fact that G is AT-free. If also $y \in L$, and v misses a vertex in U_x as well as a vertex in $U_y = N[y] \cap L'$, we again have an (x, y) -path not containing a neighbor of v . Thus, without loss of generality, we may assume that v is adjacent to all vertices in U_x . Since v is a vertex of L , which has the fewest neighbors in L' , we conclude that $U_x = N[v] \cap L'$.

Now, consider a vertex $u_x \in U_x$ and a shortest path $P(v, y) = (v = y_0, y_1, \dots, y_{k-1}, y_k = y)$ between v and y . Since $d(x, y) > d(v, y)$, $xv \notin E$ and $U_x = N[v] \cap L'$, we have $xy_i, u_x y_j \notin E$ for each $i = 0, 1, \dots, k$ and each $j = 2, 3, \dots, k$, and also $y_1 \in L$. We claim that the whole path $P(v, y)$ is contained in L . Assume not and let y_i be a vertex of $P(v, y) \cap L'$ with smallest index i . Vertices u_x and y_i belong to disk $D_{\text{ecc}(u)-1}(u)$ and are connected by a path outside of the disk. By Lemma 1, these vertices have to be adjacent. But since $i \geq 2$, this is impossible.

Thus, $P(v, y) \subset L$, and, therefore, this path avoids the neighborhood $N[u]$ of u . Consider also shortest paths $P(v, u)$, $P(y, u)$ between u and v and y , respectively. Since $d(x, y) > 2$ and $U_x = N[v] \cap L'$, v cannot have neighbors in U_y . From this and $u_x y \notin E$, we conclude that neither v has a neighbor in $P(y, u)$ nor y has a neighbor in $P(v, u)$. Consequently, vertices v, y, u form an AT in G .

CASE 2. One of x and y , say y , is adjacent to v .

Consider a shortest (v, x) -path $P(v, x) = (v = x_0, x_1, \dots, x_{k-1}, x_k = x)$. Since $d(x, v) < d(x, y)$, the only neighbor of y on $P(v, x)$ is v .

CASE 2.1. $y \in L$.

Let $x_1 \in L'$. Since v is a vertex of L with the fewest neighbors in L' and $x_1 y \notin E$, there has to be a neighbor u_y of y in L' , not adjacent to v . By Lemma 1, vertices x_1 and u_y have to be adjacent. But, then, this edge together with path (x_1, v, y, u_y) create an induced cycle of length 4 in G , which is forbidden. Therefore, x_1 has to be in L .

Now, we prove that the whole path $P(v, x)$ is contained in L . Assume not and let x_i be the vertex of $P(v, x) \cap L'$ with smallest index i . Vertices $u_y \in N[y] \cap L'$ and x_i belong to disk $D_{\text{ecc}(u)-1}(u)$ and are connected by a path outside of the disk. By Lemma 1, these vertices have to be adjacent. But since $i \geq 2$, we get $d(y, x) \leq d(x, x_i) + 2 \leq d(x, v)$, which is impossible. Hence, $P(v, x) \subset L$, and, therefore, the path formed by $P(v, x)$ and edge vy avoids the neighborhood $N[u]$ of u . Consider also arbitrary shortest paths $P(x, u)$, $P(y, u)$ between u and x and y , respectively. Since $d(x, y) > 2$ and $x, y \in L$, neither x has a neighbor in $P(y, u)$ nor y has a neighbor in $P(x, u)$. Consequently, vertices x, y, u form an AT in G .

CASE 2.2. $y \notin L$ (i.e., $y \in L'$).

Assume that $P(v, x)$ has a vertex in L' , and let x_i be the vertex of $P(v, x) \cap L'$ with smallest index i . Vertices y and x_i belong to disk $D_{\text{ecc}(u)-1}(u)$ and are connected by a path outside of the disk. By Lemma 1, they have to be adjacent, contradicting $d(x, v) < d(y, x)$. Therefore, again, $P(v, x) \subseteq L$ must hold. From the choice of v and $xy \notin E$, we deduce that there has to be a neighbor u_x of x in L' , not adjacent to v . By Lemma 1, $u_x y \in E$, which contradicts $d(x, y) > 2$. ■

From this proof, we also conclude the following:

Corollary 5. *Let G be an interval graph. Consider an LL, starting at some vertex u of G and let v be a vertex of the last BFS layer. Then, $\text{ecc}(v) \geq \text{diam}(G) - 1$.*

3.3. AT-free Graphs

In this subsection, we consider AT-free graphs, claw-free AT-free graphs, and LL+.

Theorem 6. *Let G be an AT-free graph. Consider a BFS, starting at some vertex u of G , and let v be a vertex of the last BFS layer having the fewest neighbors in the previous BFS layer. Then, $\text{ecc}(v) \geq \text{diam}(G) - 1$.*

Proof. If u is universal, then $\text{diam}(G) \leq 2$ and $\text{ecc}(v) \leq 2$, so we may assume there are at least three layers (including $\{u\}$) in the BFS. Let L be the last, and L' , the second to last layer, of a BFS started at u . Suppose that x, y is a diametral pair of vertices of G with

$$\text{ecc}(v) < d(x, y) - 1. \quad (2)$$

Let $P(x, v)$, $P(y, v)$ be paths of shortest length connecting v with x and y , respectively. Vertex y cannot have a neighbor on the path $P(x, v)$, because, otherwise, $d(x, y) \leq d(x, v) + 1 \leq \text{ecc}(v) + 1$, contradicting (2). Similarly, there is no neighbor of x on the path $P(y, v)$. In particular, x, y, v form an independent triple of G .

Since G is AT-free, every (x, y) -path has to contain a neighbor of v ; in other words, removing $N[v]$ leaves x and y in different connected components. This implies that at least one of x and y , say x , has to be contained in L , since, otherwise, there is an (x, y) -path in G (via u) that does not contain a neighbor of v .

By definition of BFS, vertex x has to have neighbors in L' . Let $U_x = N(x) \cap L'$. If $y \notin L$, v is adjacent to all those neighbors U_x because, otherwise, there is an (x, y) -path (via U_x and u) not containing a neighbor of v , contradicting the fact that G is AT-free. If $y \in L$, and v misses a vertex in U_x as well as a vertex in $U_y = N(y) \cap L'$, we again have an (x, y) -path not containing a neighbor of v . Thus, without loss of generality, we may assume that v is adjacent to all vertices in U_x . However, $d(x, y) > d(v, x) + 1$.

y); therefore, the shortest path $P(v, y)$ does not use any of the U_x vertices. The neighbor of v on $P(v, y)$ must be in L , since, otherwise, v would not have the fewest neighbors in L' .

First, assume that $P(v, y)$ has a vertex in L' and let y' be the first such vertex in the v to y direction. Let x'' be a vertex in U_x and let x'' be one of its ancestors in the previous BFS layer. If y' is adjacent to x'' , then $d(x, y) \leq 3 + d(y, y')$. But $d(y, y') = d(v, y) - d(v, y')$ and $d(v, y') \geq 2$. Thus, $d(x, y) \leq d(v, y) + 1$ and $d(x, y) \leq \text{ecc}(v) + 1$ contradicting (2). Therefore, $x''y' \notin E$, and we claim that $\{x'', v, y'\}$ forms an AT in G . Indeed, vertex v misses the concatenation of the direct paths from x'' to u and from y' to u . Vertex x'' misses the subpath of $P(v, y)$ from v to y' since all vertices other than y' are in L and $x''y' \notin E$. Vertex y' misses the path (v, x', x'') , since if $x'y' \in E$, then $d(x, y) \leq d(v, y)$, which is impossible.

So, $P(v, y)$ has to be contained in L . Now $\{v, y, u\}$ forms an AT in G . Indeed, u clearly misses $P(v, y)$ since it is in L . Let x' be an arbitrary vertex in U_x . We have $x'y \notin E$, since, otherwise, $d(x, y) = 2 \leq d(v, y) \leq \text{ecc}(v)$. Hence, y misses the (v, u) -path formed by the edge vx' together with a direct path from x' to u . Let y' be an arbitrary ancestor of y in L' . We have $vy' \notin E$, since, otherwise, v would no longer have the fewest neighbors in L' and, thus, v misses the (y, u) -path formed by the edge $y'y$ together with a direct path from y' to u . ■

This result can be strengthened further for claw-free AT-free graphs.

Theorem 7. *Let G be a claw-free, AT-free graph. Consider a BFS, starting at some (nonuniversal) vertex u of G , and let v be a vertex of the last BFS layer having the fewest neighbors in the previous BFS layer. Then, $\text{ecc}(v) = \text{diam}(G)$.*

Proof. Let L be the last, L' , the second to last, and L'' , the third to last layer of a BFS started at vertex u . Suppose that x, y is a diametral pair of vertices of G with

$$\text{ecc}(v) < d(x, y). \quad (3)$$

First, assume that $\text{diam}(G) = 2$. Since u is not universal, u is not contained in L' . Consequently, for any vertex $v \in L$, we have $\text{ecc}(v) = \text{diam}(G)$, contradicting (3). Hence, from now on, we can assume that $\text{diam}(G) > 2$ and, thus, at most one of x and y is adjacent to v .

CASE 1. Both x and y are not adjacent to v .

Let $P(x, v)$ be a path of shortest length, connecting x and v . Vertex y cannot have a neighbor on $P(x, v)$, because $d(x, y) > d(x, v)$ and neither x nor v is adjacent to y . Similarly, x has no neighbor on shortest path $P(y, v)$ connecting vertices y and v . In particular, x, y, v form an independent triple of G .

Since G is AT-free, every (x, y) -path has to contain a neighbor of v . This implies that at least one of x and y , say x , has to be contained in L , since, otherwise, there is an (x, y) -path in G (via u) that does not contain a neighbor of v .

By definition of BFS, vertex x has to have neighbors in L' , that is, $U_x = N(x) \cap L' \neq \emptyset$. If $y \notin L$, v is adjacent to all those neighbors of x because, otherwise, there is an (x, y) -path (via U_x and u) not containing a neighbor of v , contradicting the fact that G is AT-free. If, also, $y \in L$, and v misses a vertex in U_x as well as a vertex in $U_y = N(y) \cap L'$, we again have an (x, y) -path not containing a neighbor of v . Thus, without loss of generality, we may assume that v is adjacent to all vertices in U_x .

Now, consider some vertex $x' \in U_x$, adjacent to both x and v . Vertex x' has to be adjacent to some vertex x'' in L'' . By definition of BFS, x'' is not adjacent to v and x , which are not adjacent. Hence, x', x'', x, v form a claw, contradicting G being claw-free.

CASE 2. One of x and y , say y , is adjacent to v .

CASE 2.1. $y \in L$.

CASE 2.1.1. $N(v) \cap L' \subseteq N(y) \cap L'$.

Consider a shortest (x, v) -path $P(x, v)$. Since $d(x, v) < d(x, y)$, the neighbor v' of v on $P(x, v)$ is not adjacent to y ; thus, $v' \in L$. In fact, the only neighbor of y on $P(x, v)$ is v . In particular, using $P(x, v)$, one can construct a (v', u) -path P_1 , not containing a neighbor of y . Indeed, extend the (v', x) -subpath of $P(x, v)$ by any shortest (x, u) -path. If $x \in L$, then any neighbor x' of x in L' is not adjacent to y [because $d(x, y) > 2$] and, by the definition of BFS, y is not adjacent to any vertex of a shortest (x', u) -path. If x itself is not contained in L , then y is not adjacent to any shortest (x, u) -path.

Now, we claim that v', u, y form an AT in G . To see this, observe the following: Vertices v' and y are not adjacent and are contained in L . Thus, v', u, y form an independent triple. Path (v', v, y) is a (v', y) -path which avoids $N[u]$. To see that there is a path (y, y', \dots, u) which avoids $N[v']$, one just has to observe that any neighbor y' of y in L' cannot be adjacent to v' (because, otherwise, y', y, v', y'' form a claw for any neighbor y'' of y' in L'') and any shortest (y', u) -path cannot contain a neighbor of v' . A similar argument shows there is a (v', u) -path which avoids $N[y]$.

CASE 2.1.2. There exists $w \in N(v) \cap L'$ with $w \notin N(y)$.

Since v is a vertex of L with the fewest neighbors in L' , there has to be a neighbor y' of y in L' , not adjacent to v . Consider a shortest (x, v) -path $P(x, v)$. If v' , the neighbor of v in $P(x, v)$, is contained in L , then, by the same arguments as above, v', u, y form an AT of G . Hence, $v' \in L'$ and we can assume that $v' = w$.

Let v'' be the neighbor of v' in $P(x, v)$. Since $d(x, y) > d(x, v)$, we have $yv'', y'v'' \notin E$, and since $P(x, v)$ is a shortest (x, v) -path, we have $vv'' \notin E$. Consequently, $v'y'$

$\notin E$, since, otherwise, v', v'', v, y' form a claw. Furthermore, v'' has to be adjacent to all neighbors v^* of v' in L'' , because, otherwise, we have a claw v', v, v^*, v'' . This implies that $v'' \notin L$ (by definition of BFS) and that there is no common neighbor y'' of v' and y' in the previous BFS layer, since, otherwise, v'', v, y' form an AT with the paths $P(v'', v) = (v'', v', v)$, $P(v, y') = (v, y, y')$, $P(y', v'') = (y', y'', v'')$. (Recall that $v''y'' \in E$ since v'' has to be adjacent to all neighbors of v' in L'' .)

However, by the definition of BFS, both v' and y' have to have neighbors on L'' , say v^*, y'' , respectively. Now, v, v^*, y' form an AT, where v^* and y' are joined by a path via u ; $P(v, v^*) = (v, v', v^*)$; $P(y', v) = (y', y, v)$, a contradiction.

CASE 2.2. $y \notin L$ (i.e., $y \in L'$).

CASE 2.2.1. $uy \in E$.

First, observe that $xv \notin E$ since $d(x, y) > 2$. Now, let v' be the neighbor of v on a shortest (x, v) -path $P(x, v)$. Since $d(x, v) < d(x, y)$, $v'y \notin E$ and u is not adjacent to any vertex of $P(x, v) \setminus \{v'\}$. In particular, $xu \notin E$ and $x \in L$. Then, there must exist a vertex $x' \in L'$ such that $xx', ux' \in E$. Furthermore, since $d(x, y) = 3 > d(v, x)$ [i.e., $P(x, v)$ is of length 2], to avoid a claw formed by v, u, v', x , vertices v' and u cannot be adjacent. Also, vertices x', v are not adjacent, since, otherwise, a claw on x, x', u, v will occur. However, now x, v, u form an AT with the paths $P(x, v)$, $P(v, u) = (v, y, u)$, $P(u, x) = (u, x', x)$.

CASE 2.2.2. $uy \notin E$.

Note that since $uy \notin E$, $\text{ecc}(v) > 2$ and, thus, by (3), $\text{diam}(G) = d(x, y) > 3$. Again, let v' be the neighbor of v on a shortest (x, v) -path $P(x, v)$. Since $d(x, v) < d(x, y)$, y is not adjacent to any vertex of $P(x, v)$, except v . Further, consider a shortest (x, u) -path P' . No matter whether x is contained in L, L' or any other layer, y cannot have a neighbor on P' , since $y \in L'$ and $d(x, y) > 3$. Thus, there is a (v', u) -path which avoids $N[y]$. Let $P(y, u)$ be a shortest (y, u) -path and let y' be the neighbor of y on this path. We claim that $P(y, u)$ does not contain any neighbor of v' . To see this, first note that the only possible neighbors of v' on $P(y, u)$ are y and y' , since v' is either contained in L or L' . We have $v'y \notin E$, since the only neighbor of y on $P(x, v)$ is v . However, if $v'y' \in E$, then y', y, v', y'' form a claw, where y'' is a neighbor of y' in the previous layer (which has to exist because $uy \notin E$). Consequently, v', y, u form an AT, contradicting G being AT-free.

Thus, none of the above cases can occur, showing that (3) cannot be satisfied. This completes the proof. ■

It is easy to adapt this proof to show the following:

Corollary 8. *Let G be a claw-free, AT-free graph. Consider an LL, starting at some vertex u of G and let v be a vertex of the last BFS layer. Then, $\text{ecc}(v) \geq \text{diam}(G) - 1$.*

4. K-CHORDAL GRAPHS

As mentioned in the Introduction, the examples that show that LBFS fails to find vertices of high eccentricity all have large induced cycles. Furthermore, both chordal and AT-free graphs have constant bounds on the maximum size of induced cycles, namely, 3 and 5, respectively. Thus, one would hope that for k -chordal graphs, where k is a constant, some form of BFS would succeed in finding a vertex whose eccentricity is within some function of k of the diameter. In fact, we show that LL is sufficiently strong to ensure this. First, we establish a lemma that is used in the proof.

Lemma 9. *If vertices a and b of a disk $D_r(u)$ of a k -chordal graph are connected by a path $P(a, b)$ outside of $D_r(u)$ [i.e., $P(a, b) \cap D_r(u) = \{a, b\}$], then $d(a, b) \leq \lfloor k/2 \rfloor$.*

Proof. Assume that $d(a, b) > \lfloor k/2 \rfloor$, and let P be an induced subpath of $P(a, b)$ connecting vertices a and b . Consider shortest paths $P(a, u)$ and $P(b, u)$ (connecting a with u and b with u , respectively). Using vertices of these paths, we can construct an induced path $Q(a, b)$ with the property that all its vertices except a and b are contained in $D_{r-1}(u)$. By our construction, the cycle C obtained by the concatenation of P and $Q(a, b)$ is induced. Since $d(a, b) > \lfloor k/2 \rfloor$, both paths P and $Q(a, b)$ must be of length greater than $\lfloor k/2 \rfloor$. Therefore, the cycle C has length at least $\lfloor k/2 \rfloor + 1 + \lfloor k/2 \rfloor + 1 > k$, which is impossible. ■

Theorem 10. *Let G be a k -chordal graph ($k \geq 4$). Consider an LL, starting at some vertex u of G , and let v be a vertex of the last BFS layer. Then, $\text{ecc}(v) \geq \text{diam}(G) - \lfloor k/2 \rfloor$.*

Proof. Let x, y be a pair of vertices such that $d(x, y) = \text{diam}(G)$, and consider two shortest paths $P(x, v)$ and $P(y, v)$ connecting vertex v with x and y , respectively. Also, let q be the minimum index such that

$$L_q \cap (P(x, v) \cup P(y, v)) \neq \emptyset.$$

Consider a vertex $z \in L_q \cap (P(x, v) \cup P(y, v))$, and assume, without loss of generality, that z belongs to $P(y, v)$ and is the vertex of $L_q \cap P(y, v)$ closest to v . We have $d(v, z) + d(z, y) = d(v, y)$. Consider also the shortest paths $Q(x, u)$ and $Q(y, u)$ connecting vertex u with x and y , respectively, and vertices $x' \in L_q \cap Q(x, u)$ and $y' \in L_q \cap Q(y, u)$ [see Fig. 13(a)].

Suppose that $\text{ecc}(v) < d(x, y) - \lfloor k/2 \rfloor$. In particular, we have

$$d(x, y) > \max\{d(x, v), d(y, v)\} + \lfloor k/2 \rfloor. \quad (4)$$

Then, the following claims hold:

Claim 1. $L_q \cap P(x, v) \neq \emptyset$.

If $L_q \cap P(x, v) = \emptyset$, then vertices x' and z belong to $D_q(u)$ and can be connected outside of $D_q(u)$. By Lemma

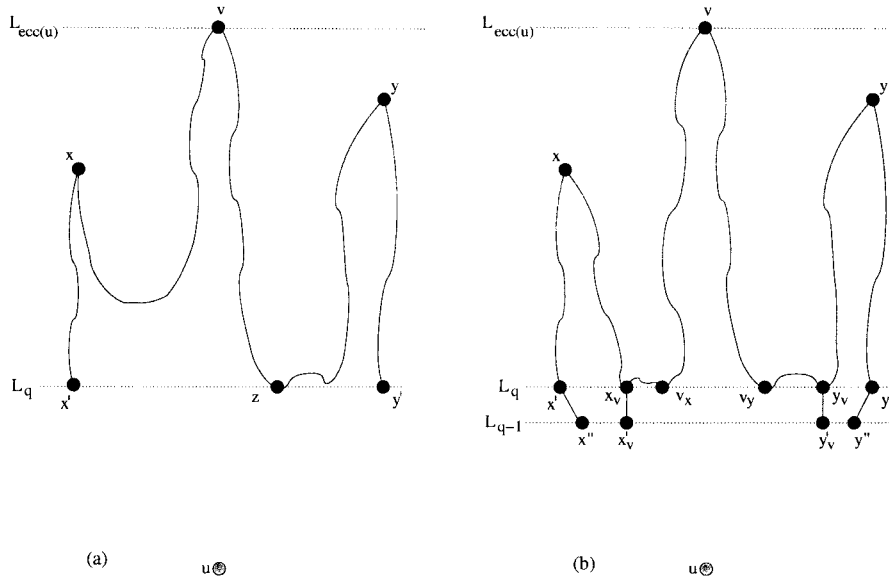


FIG. 13. Shortest paths $P(x, v)$, $P(y, v)$, $Q(x, x')$, and $Q(y, y')$.

9, $d(x', z) \leq \lfloor k/2 \rfloor$ must hold. We also have $d(x, x') \leq d(v, z)$ because $v \in L_{ecc(u)}$. Therefore, $d(x, y) \leq d(x, x') + d(x', z) + d(z, y) \leq d(v, z) + \lfloor k/2 \rfloor + d(z, y) = d(v, y) + \lfloor k/2 \rfloor$, which contradicts (4).

In what follows, we may assume that $z \neq u$, since if $z = u$, then $\text{diam}(G) = d(x, y) \leq d(y, u) + d(u, x) \leq d(y, u) + d(u, v) = d(v, y) \leq \text{ecc}(v)$.

Now, let x_v and v_x be vertices of $L_q \cap P(x, v)$ closest to x and v , respectively. Analogously, we can define vertices y_v and v_y of $L_q \cap P(y, v)$. Clearly, $z = v_y$. Consider also in L_{q-1} neighbors x'', y'', x'_v, y'_v of vertices x', y', x_v, y_v , respectively [see Fig. 13(b)]. Since vertices x'' and y'' belong to $D_{q-1}(u)$ and can be connected outside of $D_{q-1}(u)$, by Lemma 9, $d(x'', y'') \leq \lfloor k/2 \rfloor$. Similarly, by considering disk $D_q(u)$ and its vertices v_x and v_y , we deduce that $d(v_x, v_y) \leq \lfloor k/2 \rfloor$.

Claim 2. $d(x_v, v_x) \leq 1$ and $d(y_v, v_y) \leq 1$.

Because of symmetry, we will only show that $d(y_v, v_y) \leq 1$. Assume that $d(y_v, v_y) \geq 2$. Since vertices x'' and y'_v belong to $D_{q-1}(u)$ and can be connected outside of $D_{q-1}(u)$, by Lemma 9, $d(x'', y'_v) \leq \lfloor k/2 \rfloor$. We also have $d(x, x') \leq d(v, v_y)$. Therefore, $d(x, y) \leq d(x, x') + 1 + d(x'', y'_v) + 1 + d(y'_v, v_y) \leq d(v, v_y) + \lfloor k/2 \rfloor + 2 + d(y_v, v_y) \leq d(v, v_y) + d(v_y, y_v) + d(y_v, v_y) + \lfloor k/2 \rfloor = d(v, y) + \lfloor k/2 \rfloor$, which contradicts (4).

Claim 3. If $d(x_v, v_x) = 1$, then $d(x, x_v) = d(x, x')$ and $d(y, y') = d(v, v_x)$. Symmetrically, if $d(y_v, v_y) = 1$, then $d(y, y_v) = d(y, y')$ and $d(x, x') = d(v, v_y)$. Evidently, $d(x, x') \leq d(x, x_v)$. Also, since $v \in L_{ecc(u)}$, we have $d(y, y') \leq d(v, v_x)$ and, by Lemma 9, $d(x'', y'') \leq \lfloor k/2 \rfloor$. Let now $d(x_v, v_x) = 1$ and assume that $d(x, x_v) > d(x, x')$ or $d(v, v_x) > d(y, y')$. Then, $d(x, y) \leq d(x, x') + 1 + d(x'', y'') + 1 + d(y', y) < d(x, x_v) + 1$

$+ \lfloor k/2 \rfloor + 1 + d(v, v_x) = d(x, v) + 1 + \lfloor k/2 \rfloor$, and a contradiction to (4) arises.

Claim 4. If $x_v = v_x$, then $d(x, x_v) = d(x, x') + 1$ and $d(y, y') = d(v, v_x)$. Symmetrically, if $y_v = v_y$, then $d(y, y_v) = d(y, y') + 1$ and $d(x, x') = d(v, v_y)$.

We have $d(y, y') \leq d(v, v_x) = d(v, x_v)$ and $d(x, x') \leq d(v, v_y)$. If $d(x, x') = d(x, x_v)$, then $d(x, y) \leq d(x, x_v) + d(x_v, v_y) + d(v_y, y) = d(x, x') + d(v_x, v_y) + d(v_y, y) \leq d(v, v_y) + \lfloor k/2 \rfloor + d(v_y, y) = d(v, y) + \lfloor k/2 \rfloor$, and we arrive at a contradiction to (4). Thus, $d(x, x_v) \geq d(x, x') + 1$ must hold. Now, if $d(x, x_v) \geq d(x, x') + 2$ or $d(v, v_x) > d(y, y')$ [i.e., $d(x, x_v) + d(v, v_x) \geq d(x, x') + d(y, y') + 2$], then $d(x, y) \leq d(x, x') + 1 + d(x'', y'') + 1 + d(y', y) \leq d(x, x_v) + d(v, v_x) - 2 + 1 + \lfloor k/2 \rfloor + 1 = d(x, v) + \lfloor k/2 \rfloor$, and, again, a contradiction to (4) occurs.

From Claims 3 and 4, we conclude that equalities $d(y, y') = d(x, x') = d(v, v_x) = d(v, v_y)$ must always hold. Denote this distance by t . Clearly, $t = \text{ecc}(u) - q$. We will distinguish between three (up to symmetry) cases, depending on whether vertices v_x, x_v (respectively, v_y, y_v) are adjacent or coincide.

CASE 1. $d(v_x, x_v) = d(v_y, y_v) = 1$.

By Claim 3, we have $t = d(y, y') = d(y, y_v) = d(v, v_x) = d(x, x') = d(x, x_v) = d(v, v_y)$. We may choose $x' = x_v$ and $y' = y_v$ [see Fig. 14(a)]. Since $d(x, y) \leq 2t + 2 + \lfloor k/2 \rfloor$ and $d(v, x) = d(v, y) = 2t + 1$, by assumption (4), we conclude that $d(x, y) = 2t + 2 + \lfloor k/2 \rfloor$, that is, $d(y', x') = \lfloor k/2 \rfloor + 2 \geq 4$, $d(x'', y'') = d(v_x, v_y) = \lfloor k/2 \rfloor \geq 2$. (Note that Lemma 9 applies to x'', y'' as well as to v_x, v_y .) We will show that this situation is impossible by creating in G an induced cycle of length at least $k + 1$. Let $P(v_x, v)$ and $P(v_y, v)$ be subpaths of shortest paths $P(x,$

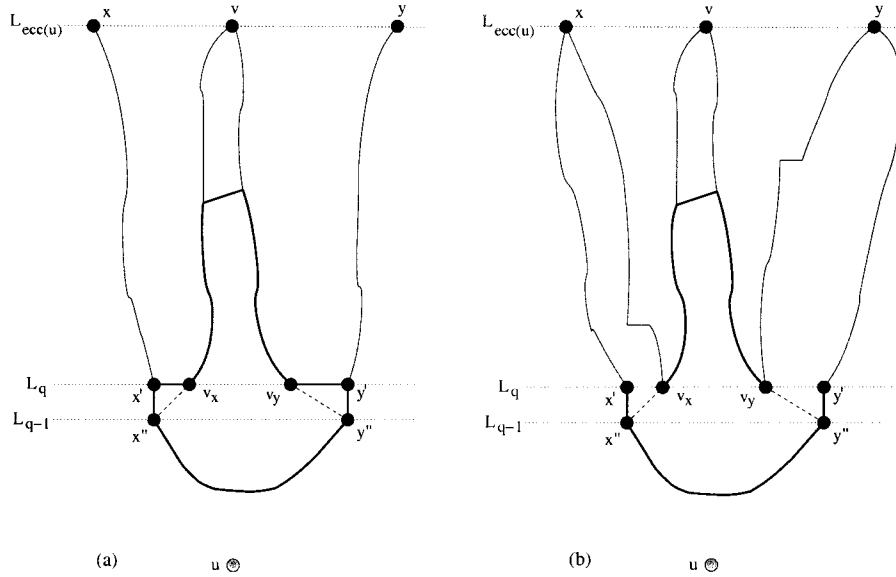


FIG. 14. Illustration for Cases 1 and 2.

v) and $P(y, v)$, connecting corresponding vertices. Consider in a subgraph of G induced by $P(v_x, v) \cup P(v_y, v)$ a shortest (v_y, v_x) -path P' . The length of P' is at least $\lfloor k/2 \rfloor$. Analogously, in a subgraph of G induced by subpaths $Q(x'', u)$, $Q(y'', u)$ of paths $Q(x, u)$, $Q(y, u)$, consider a shortest (x'', y'') -path Q' . Its length again is at least $\lfloor k/2 \rfloor$. Using paths P' , Q' and edges $y''y'$, $y'v_y$, $v_x x'$, and $x'x''$ of G , we can create a cycle of length at least $\lfloor k/2 \rfloor + 2 + \lfloor k/2 \rfloor + 2$. In this cycle, because of distance requirements, only chords $v_y y''$ and $v_x x''$ are possible. Even having these chords, we still have an induced cycle of length at least $\lfloor k/2 \rfloor + 1 + \lfloor k/2 \rfloor + 1 > k$ in G , which is impossible.

CASE 2. $d(v_x, x_v) = d(v_y, y_v) = 0$.

By Claim 4, we have $t = d(y, y') = d(y, v_y) - 1 = d(v, v_x) = d(x, x') = d(x, v_x) - 1 = d(v, v_y)$ [see Fig. 14(b)]. Again, since $d(x, y) \leq 2t + 2 + \lfloor k/2 \rfloor$ and $d(v, x) = d(v, y) = 2t + 1$, by assumption (4), we get $d(x, y) = 2t + 2 + \lfloor k/2 \rfloor$. Therefore, $d(y', x') = \lfloor k/2 \rfloor + 2 \geq 4$, $d(x'', y'') = d(v_x, v_y) = \lfloor k/2 \rfloor \geq 2$. Note also that $d(v, y') > t$ and $d(v, x') > t$ as $d(v, x) = d(v, y) = 2t + 1$ and $d(x, x') = d(y, y') = t$. Consider a shortest (x', y') -path P' in a subgraph of G induced by vertices of $Q(x, x') \cup P(x, v) \cup P(y, v) \cup Q(y, y')$, where $Q(x, x')$ and $Q(y, y')$ are corresponding subpaths of paths $Q(x, u)$ and $Q(y, u)$.

We claim that $v_x, v_y \in P'$. Indeed, if neither v_x nor v_y is on P' , then P' connects vertices x', y' outside of $D_q(u)$ and, therefore, by Lemma 9, we would have $d(x', y') \leq \lfloor k/2 \rfloor$, which contradicts $d(x', y') = \lfloor k/2 \rfloor + 2$. If only one of v_x, v_y (say, without loss of generality, v_x) is on P' , then now a subpath of P' connects vertices v_x and y' outside of $D_q(u)$, and, hence, $d(v_x, y') \leq \lfloor k/2 \rfloor$, implying that $d(x, y) \leq d(x, v_x) + d(v_x, y') + d(y', y) \leq t + 1 + \lfloor k/2 \rfloor + t = 2t + 1 + \lfloor k/2 \rfloor$, which is impossible.

As in Case 1, we also consider a shortest (x'', y'') -path Q' in a subgraph of G induced by subpaths $Q(u, x'')$, $Q(u, y'')$. Paths P' , Q' together with edges $x'x''$, $y'y''$ form a cycle C in G . Since $P' \setminus \{x', v_x, v_y, y'\} \subseteq \setminus D_q(u)$ and $Q' \setminus \{x'', y''\} \subseteq D_{q-2}(u)$, any possible chord of C must go from vertices x'', y'' to vertices x', v_x, v_y, y' .

By the distance requirement $d(y', x') \geq 4$, chords $x''y'$ and $y''x'$ are impossible. Also, $x''v_y, y''v_x \notin E(G)$, because, otherwise, $d(x, y) \leq t + 2 + t + 1 = 2t + 3$, which is less than $2t + 2 + \lfloor k/2 \rfloor$ for $k \geq 4$. Hence, the only possible chords in C are $x''v_x$ and $y''v_y$. Since $d(x'', y'') = d(v_x, v_y) = \lfloor k/2 \rfloor$, even with those chords we obtain an induced cycle (subcycle of C) in G of length at least $\lfloor k/2 \rfloor + 1 + \lfloor k/2 \rfloor + 1 > k$.

CASE 3. $d(v_x, x_v) = 0$ and $d(v_y, y_v) = 1$.

Similar to Cases 1 and 2, we can show that this case is also impossible; we can create an induced cycle of length at least $k + 1$ in G . ■

Note that Theorem 10 does not hold for chordal graphs (i.e., $k = 3$) as Figure 4 shows. To get the bound of $\text{diam}(G) - \lfloor k/2 \rfloor = \text{diam}(G) - 1$ on $\text{ecc}(v)$, for chordal graphs, we need to choose a vertex v visited last by BFS.

For 4-chordal graphs and 5-chordal graphs (in particular, for AT-free graphs), we have the following corollary:

Corollary 11. *Let G be a 5-chordal graph. Consider an LL, starting at some vertex u of G , and let v be a vertex of the last BFS layer. Then, $\text{ecc}(v) \geq \text{diam}(G) - 2$.*

Again, this bound on $\text{ecc}(v)$ is tight. Figure 8 represents a 4-chordal graph G for which an LBFS exists such that the vertex v , last visited by this LBFS, has eccentricity equal to $\text{diam}(G) - 2$.

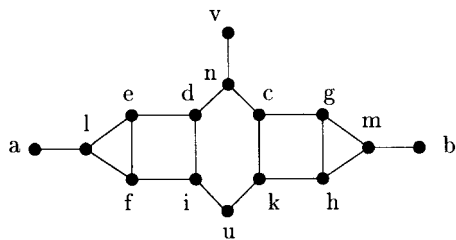


FIG. 15. LL: $u|ik|fdch|lengm|abv$.

Examples of k -chordal graphs with a larger difference between the diameter and the eccentricity of the vertex visited last in some variant of BFS are given in the following two figures. Figure 15 shows a 6-chordal graph. For this graph G , there is an LL, starting at vertex u , that ends at vertex v , where $\text{ecc}(v) = \text{diam}(G) - 3$. This shows that the bound of $\text{diam}(G) - 3$ given by Theorem 10 is tight for 6-chordal graphs.

In Figure 16, each of the dashed edges stand for a path of length k . Thus, this graph G is $4k$ -chordal. The diameter of G is also $4k$, and the eccentricity of v , the vertex visited last by some LBFS started at u , is $\text{ecc}(v) = 2k + 1$. Hence, the difference between the diameter and the eccentricity $\text{ecc}(v)$ is $2k - 1$. This shows that, at least for the $4k$ -chordal graphs, the bound on $\text{ecc}(v)$ (for LL) given in Theorem 10 is close to the best possible. It is within 1 of the bound that could be achieved by LBFS.

5. CONCLUDING REMARKS

In this paper, we have shown that various versions of BFS, when applied to certain restricted families of graphs, end at a vertex whose eccentricity is close to the diameter of the graph. All these restricted families have no induced cycles greater than a small constant. This motivated our look at the behavior of LL on k -chordal graphs and we discovered that the difference between the graph's diameter and the eccentricity of the last vertex visited by LL is at most $\lfloor k/2 \rfloor$. This raises the question of whether other versions of BFS or a multisweep algorithm could improve on this bound. Furthermore: Are there interesting restricted families of graphs where there is no bound on the size of induced cycles, yet some variant of BFS ends at a vertex

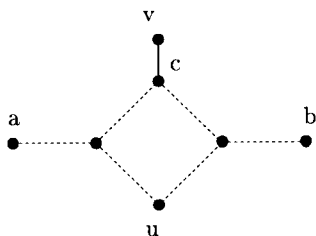


FIG. 16. LBFS: $u|..|cab|v$.

whose eccentricity is within a small constant of the graph's diameter?

One of the referees of the paper raised the following interesting question: Is there a "robust" algorithm (see [12, 15]) for determining, in linear time, the diameter of an $\{\text{AT}, \text{claw}\}$ -free graph? Note that we have shown that, given an $\{\text{AT}, \text{claw}\}$ -free graph, both LL+ and LBFS will end at a vertex of maximum eccentricity. Although this algorithm requires linear time, no linear time recognition algorithm is known for $\{\text{AT}, \text{claw}\}$ -free graphs, and, thus, one would like to have a linear time algorithm that would either find a vertex of maximum eccentricity or show that the graph is not $\{\text{AT}, \text{claw}\}$ -free by demonstrating either a claw or an AT. Note that the same type of question can be asked for AT-free graphs, in particular: Is there a linear time algorithm that can get within 1 of the graph's diameter or show that the graph is not AT-free by demonstrating an AT? We leave such questions for future investigation.

REFERENCES

- [1] D. Aingworth, C. Chekuri, P. Indyk, and R. Motwani, Fast estimation of diameter and shortest paths (without matrix multiplication), *SIAM J Comput* 28 (1999), 1167–1181.
- [2] A. Brandstädt and F.F. Dragan, On linear and circular structure of a (claw, net)-free graph, *Discr Appl Math*, to appear.
- [3] V.D. Chepoi and F.F. Dragan, Linear-time algorithm for finding a central vertex of a chordal graph, "Algorithms—ESA'94" Second Ann European Symp, Utrecht, The Netherlands, September 1994, Jan van Leeuwen (Editor), LNCS 855, Springer, Berlin, 1994, pp. 159–170.
- [4] D. Coppersmith and S. Winograd, Matrix multiplication via arithmetic progression, *Proc 19th ACM Symp on Theory of Computing*, 1987, pp. 1–6.
- [5] D.G. Corneil, F.F. Dragan, M. Habib, and C. Paul, Diameter determination on restricted graph families, *Discr Appl Math* 113 (2001), 143–166.
- [6] D.G. Corneil, S. Olariu, and L. Stewart, Linear time algorithms for dominating pairs in asteroidal triple-free graphs, *SIAM J Comput* 28 (1999), 1284–1297.
- [7] D. Dor, S. Halperin, and U. Zwick, All pairs almost shortest paths, *Proc 37th Ann IEEE Symp on Foundations of Computer Science*, 1996, pp. 452–461.
- [8] F.F. Dragan, Almost diameter of a house-hole-free graph in linear time via LexBFS, *Discr Appl Math* 95 (1999), 223–239.
- [9] F.F. Dragan, F. Nicolai, and A. Brandstädt, LexBFS-orderings and powers of graphs, *Proc 22nd Int Workshop "Graph-Theoretic Concepts in Computer Science" (WG'96)*, Cadenabbia, Italy, 1996, Fabrizio d'Amore, Paolo G. Franciosa, and Alberto Marchetti-Spaccamela (Editors), LNCS 1197, Springer, Berlin, 1997, pp. 166–180.
- [10] G. Handler, Minimax location of a facility in an undirected tree graph, *Trans Sci* 7 (1973), 287–293.

- [11] C.G. Lekkerkerker and J.C. Boland, Representation of a finite graph by a set of intervals on the real line, *Fundam Math* 51 (1962), 45–64.
- [12] V. Raghavan and J. Spinrad, Robust algorithms for restricted domains, *Proc 12th ACM–SIAM Symp on Discrete Algorithms*, 2001, pp. 460–467.
- [13] D. Rose, R.E. Tarjan, and G. Lueker, Algorithmic aspects of vertex elimination on graphs, *SIAM J Comput* 5 (1976), 266–283.
- [14] R. Seidel, On the all-pair-shortest-path problem, *Proc 24th ACM Symp Theory of Computing*, 1992, pp. 745–749.
- [15] J.P. Spinrad, *Efficient graph representations*, Fields Institute Monographs #19, AMS, Providence, 2003.
- [16] Uri Zwick, Exact and approximate distances in graphs—A survey, “Algorithms—ESA’01” 9th Ann European Symp, Aarhus, Denmark, August 2001, LNCS 2161, Springer, Berlin, 2001, pp. 33–48.