

# **TRADE-OFFS IN NON-REVERSING DIAMETER**

Hans L. Bodlaender, Gerard Tel, and  
Nicola Santoro

SCS-TR-169, FEBRUARY 1990

School of Computer Science, Carleton University  
Ottawa, Canada, K1S 5B6

# Trade-Offs in Non-Reversing Diameter

Hans L. Bodlaender\* Gerard Tel†‡

Department of Computer Science, Utrecht University,  
P.O. Box 80.089, 3508 TB Utrecht, The Netherlands

Nicola Santoro†

Carleton School of Computer Science,  
Ottawa (Ont.) K1S 5B6, CANADA

## Abstract

Consider a tree  $T$  with a number of extra edges (the *bridges*) added. We consider the notion of diameter, that is obtained by admitting only paths  $p$  with the property that for every bridge  $b$  in path  $p$ , no edge that is on the unique path (in  $T$ ) between the endpoints of  $b$  is also in  $p$  or on the unique path between the two endpoints of any other bridge in  $p$ . (Such a path is called *non-reversing*.) We investigate the trade-off between the number of added bridges and the resulting diameter. Upper and lower bounds of the same order are obtained for all diameters of constant size, of size  $\alpha(N) + O(1)$  or of size  $f(N)$  where  $f(N)$  grows faster than  $\alpha(N)$ . Some applications are given.

## 1 The Problem

Let an undirected tree  $T = (V, E)$  be given. The diameter of this tree is to be reduced by the addition of as few edges as possible, where a modified definition of the distance is given below. Calling the set of additional edges (the *bridges*)  $B$ , let  $E'$  denote  $E \cup B$  and  $T' = (V, E')$ . Given a tree  $T$  and a diameter  $D$ , a set  $B$  is

---

\*The work of this author was partially supported by the ESPRIT II Basic Research Actions Program of the EC under contract no. 3075 (project ALCOM).

†The work of this author was partially supported by the ESPRIT II Basic Research Actions Program of the EC under contract no. 3075 (project ALCOM).

‡The second and third author's work was partially supported by the National Science and Engineering Research Counsel of Canada.

called *optimal* if  $T'$  has diameter  $D$  and no set smaller than  $B$  realizes diameter  $D$ . The elements of  $E$  are sometimes referred to as *basic edges*, the elements of  $B$  as *bridges*, and all elements of  $E'$  simply as *edges*.

For the modified definition of distance, a bridge  $(u, v) \in B$  is said to *contain* the basic edges of the unique path from  $u$  to  $v$  in  $T$ . A *non-reversing path* in  $T'$  is a path (of edges in  $E'$ ) in which a basic edge is represented at most once, either by being a basic link in the path or by being contained in one of the bridges of the path. The *distance* between  $u$  and  $v$  (denoted  $d(u, v)$ ) is the length of the shortest non-reversing path between  $u$  and  $v$ , and the diameter of  $T'$  is the maximum distance between any two nodes.

As an example to the definitions, consider the tree in figure 1, with the bridge  $b = (u, w)$ . The bridge  $b$  contains the basic edges 3, 5, 6, and 7. The path  $b, 9$  from

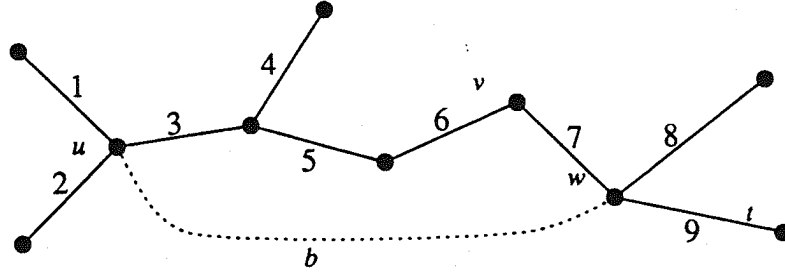


Figure 1: Example tree.

$u$  to  $t$  is non-reversing, but the path  $b, 7$  from  $u$  to  $v$  is not because the edge 7 is contained in  $b$ . In fact the shortest non-reversing path from  $u$  to  $v$  is 3,5,6, which is of length 3. Note that  $d(u, w) = 1$  and  $d(w, v) = 1$ , and hence the distance function as defined does not satisfy the triangle inequality.

This paper is organized as follows. In section 2 the case is considered where  $T$  is a linear chain. In section 3 the results for chains are extended to the general case that  $T$  is an arbitrary tree. In section 4 the results of the paper are summarized. In section 5 the motivation of the problem is given and a generalization to non-tree graphs is suggested.

## 2 Results for linear chains

The *linear chain* on  $N$  nodes is defined as the tree  $T = (V, E)$  where  $V = \{i \in \mathbb{N} : 1 \leq i \leq N\}$  and  $E = \{(i, i + 1) : 1 \leq i < N\}$ . A non-reversing path in  $T'$  in this case visits nodes in either decreasing or increasing order, as a change in direction violates the non-reversibility condition. As each decreasing path is the reverse of an increasing path, we may well restrict ourselves to paths running in increasing

direction and consider all edges as directed towards the higher of the incident nodes (to “the right”). In the rest of this section let  $T$  be a directed chain on  $N$  nodes.

## 2.1 Diameter 1

Let  $F_1(N)$  be the size of the smallest set  $B$  of bridges such that the diameter of  $T'$  is 1. There is only one such set and its size is easily computed.

**Theorem 2.1** ([3])  $F_1(N) = \frac{1}{2}(N-1)(N-2) = \Theta(N^2)$

**Proof.** The diameter of  $T'$  is 1, hence  $E'$  connects  $V$  completely, i.e.,  $E' = \{(i, j) : 1 \leq i < j \leq N\}$  and so the size of  $E'$  is  $\frac{1}{2}N(N-1)$ . Subtracting the  $N-1$  basic edges, find  $|B| = \frac{1}{2}(N-1)(N-2)$ .  $\square$

## 2.2 Diameter 2

Let  $F_2(N)$  be the size of the smallest set  $B$  of bridges such that the diameter of  $T'$  is at most 2. In this subsection it is proved that  $F_2(N) = \Theta(N \log N)$ .

**Theorem 2.2** ([3])  $F_2(N) \leq O(N \log N)$ .

**Proof.** A set of bridges giving  $D = 2$  is defined recursively. If  $N \leq 3$  then  $D \leq 2$  already and no bridges are added. Otherwise (see figure 2) let  $M$  be the “center” node  $\lceil \frac{N}{2} \rceil$ , connect all nodes (except  $M$ ’s neighbors) to  $M$  and apply the construction

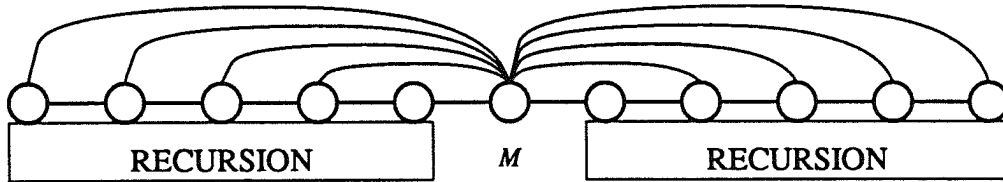


Figure 2: Construction for diameter 2.

recursively to the “lower part” (consisting of nodes 1 through  $M-1$ ) and the “higher part” (consisting of nodes  $M+1$  through  $N$ ) of the chain.

To show that the diameter is now at most 2, consider two nodes  $i$  and  $j$ , where  $i < j$ . If  $i = M$  or  $j = M$ , an edge  $(i, j)$  exists by construction and  $d(i, j) = 1$ . If  $i < M < j$ , the path  $(i, M), (M, j)$  is non-reversing and  $d(i, j) = 2$ . Finally, if  $i$  and  $j$  are on the same side of  $M$ , a non-reversing path exists because of the recursive application of the construction.

To analyse the number of bridges, let  $f(N)$  be the number of bridges used when the construction is applied to a chain of  $N$  nodes, and note that

$$f(N) = \begin{cases} 0 & \text{if } N \leq 3 \\ f(\lceil \frac{N}{2} \rceil - 1) + (N - 3) + f(\lfloor \frac{N}{2} \rfloor) & \text{otherwise.} \end{cases}$$

The solution to this equation is  $f(N) = O(N \log N)$ , hence  $F_2(N) \leq O(N \log N)$ .  
□

To prove lower bounds it turns out to be easier to consider the size of  $E'$  first, rather than look at the size of  $B$  directly.

**Theorem 2.3**  $F_2(N) \geq \Omega(N \log N)$ .

**Proof.** Let  $f(N)$  denote the minimal size of any set  $E'$  that gives a chain of  $N$  nodes a diameter of at most 2, this is the size of  $B$  plus  $N - 1$ . For  $N \leq 3$ ,  $f(N) = N - 1$ . Now let  $N > 3$  and let such an  $E'$  be given. Partition  $V$  in a “lower block”  $\{i : i \leq \frac{1}{2}N\}$  and a “higher block”  $\{i : i > \frac{1}{2}N\}$  (see figure 3). Call an edge

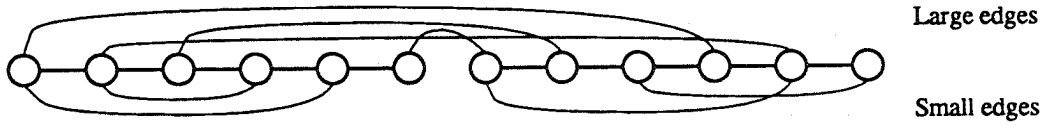


Figure 3: Lower bound for diameter 2.

$(i, j)$  in  $E'$  *small* if  $i$  and  $j$  are in the same block and *large* otherwise. As two nodes in the lower block have distance at most 2, and a non-reversing path between these nodes lies entirely within the lower block, there are at least  $f(\lfloor \frac{1}{2}N \rfloor)$  small edges in the lower block. Similarly, there are at least  $f(\lceil \frac{1}{2}N \rceil)$  small edges in the higher block.

As for the large edges, suppose there is an  $i_0$  in the lower block such that there is no large edge  $(i_0, j)$ , i.e., all edges incident to  $i_0$  are small edges and lead to a node  $k \leq \frac{1}{2}N$ . Now for every  $j > \frac{1}{2}N$  there is a path of length  $\leq 2$  from  $i_0$  to  $j$ , hence there exists an edge  $(k, j)$  for some  $k \leq \frac{1}{2}N$ . It follows that all nodes in the lower block are incident to a large edge or all nodes in the higher block are incident to a large edge. So there are at least  $\lfloor \frac{1}{2}N \rfloor$  large edges and thus

$$f(N) \geq \begin{cases} N - 1 & \text{if } N \leq 3 \\ f(\lfloor \frac{1}{2}N \rfloor) + f(\lceil \frac{1}{2}N \rceil) + \lfloor \frac{1}{2}N \rfloor & \text{otherwise} \end{cases}$$

and this equation solves to  $f(N) \geq \Omega(N \log N)$ . Subtracting the  $N - 1$  basic edges we find  $F_2(N) \geq \Omega(N \log N)$ . □

The two theorems show that  $F_2 = \Theta(N \log N)$  as claimed and in the proof of theorem 2.2 a construction is given that uses this number of bridges. The constants hidden in the order-notation are 1 and  $\frac{1}{2}$  for the upper and lower bound, respectively.

## 2.3 Diameter 3

Let  $F_3(N)$  be the size of the smallest set  $B$  of bridges such that the diameter of  $T'$  is at most 3. In this subsection it is proved that  $F_3(N) = \Theta(N \log \log N)$ .

**Theorem 2.4**  $F_3(N) \leq O(N \log \log N)$ .

**Proof.** A set of bridges giving  $D \leq 3$  is defined recursively. If  $N \leq 4$  then  $D \leq 3$  already and no bridges are added. Otherwise, apply the following construction in four steps (see figure 4). (0) Let  $W = \lceil \sqrt{N} \rceil$  and designate the nodes  $W, 2W, \dots$

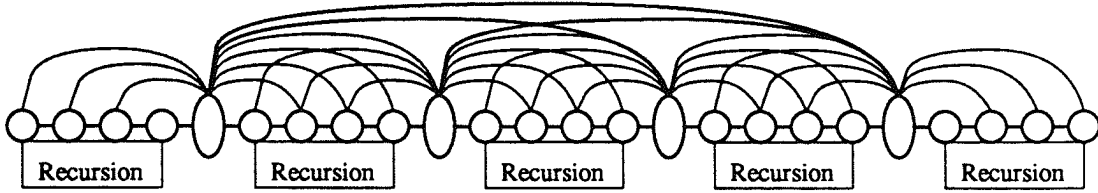


Figure 4: Construction for diameter 3.

to be *backbones* of the construction. (1) Connect every node in  $V$  (including the backbones) to the nearest backbone in each direction (if it exists). (2) Add a bridge between every pair of backbones. (3) Apply the construction recursively on each of the subchains in which the backbones cut the chain.

To show that the diameter is indeed at most 3, consider two nodes,  $i$  and  $j$ ,  $i < j$ , and both *not* a backbone. If  $i$  and  $j$  are in different subchains, let  $M_1$  be the backbone right of  $i$  and  $M_2$  the backbone left of  $j$ . The path  $(i, M_1), (M_1, M_2), (M_2, j)$  shows that  $d(i, j) \leq 3$ . If  $M_1 = M_2$  the middle edge is omitted, and the case that  $i$  or  $j$  is a backbone (or both) is left to the reader. If  $i$  and  $j$  are in the same subchain, a non-reversing path of length at most 3 exists because of the recursive application of the construction on each of the subchains.

To compute the number of bridges used in the construction, note that (1) it takes less than  $2N$  bridges to connect each node to the nearest backbones, (2) the number of backbones is  $\lfloor \frac{N}{\sqrt{N}} \rfloor \leq \sqrt{N}$  so it takes less than  $\frac{1}{2}N$  bridges to connect them completely, and (3) recursion is applied to  $\lceil \frac{N}{\sqrt{N}} \rceil \leq \lceil \sqrt{N} \rceil$  subchains of length at most  $\lceil \sqrt{N} \rceil - 1$ . Hence, with  $f(N)$  the number of bridges used by this construction

on a chain of  $N$  nodes, it follows that

$$f(N) \leq \begin{cases} 0 & \text{if } N \leq 4 \\ 2\frac{1}{2}N + \lceil \sqrt{N} \rceil f(\lceil \sqrt{N} \rceil - 1) & \text{otherwise.} \end{cases}$$

The solution to this equation is  $f(N) \leq O(N \log \log N)$ , and hence it follows that  $F_3(N) \leq O(N \log \log N)$ .  $\square$

**Theorem 2.5**  $F_3(N) \geq \Omega(N \log \log N)$ .

**Proof.** Let  $f(N)$  denote the minimal size of  $E'$  that gives diameter at most 3, again this is the size of  $B$  plus  $N - 1$ . For  $N \leq 4$ ,  $f(N) = N - 1$ , now let  $N > 4$  and let such an  $E'$  be given. Partition  $V$  in  $\lfloor \frac{N}{\sqrt{N}} \rfloor$  blocks of size at least  $\lceil \sqrt{N} \rceil$  and call an edge  $(i, j)$  *small* if  $i$  and  $j$  are in the same block, and *large* otherwise (see figure 5).

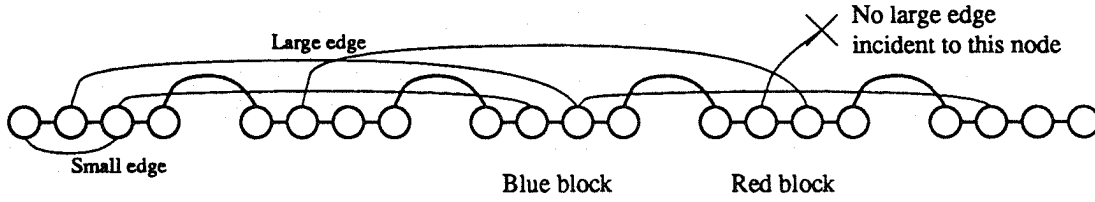


Figure 5: Lower bound for diameter 3.

As two nodes within the same block have distance at most 3, and a non-reversing path of this length lies entirely within the block, a block of size  $s$  contains at least  $f(s)$  small edges. Hence the total number of small edges is at least  $\lfloor \frac{N}{\sqrt{N}} \rfloor f(\lceil \sqrt{N} \rceil)$ .

As for the large edges, call a block *red* if it contains a node that is *not* incident to a large edge, and *blue* if each of its nodes is incident to at least one large edge. (Note that these two cases are complimentary, i.e., each block is either red or blue.) Let  $R$  be the number of red blocks and  $B$  the number of blue blocks. As the blue blocks contain at least  $B \lceil \sqrt{N} \rceil$  nodes, there are at least  $\frac{1}{2} B \lceil \sqrt{N} \rceil$  large edges incident to a node in a blue block. Now let  $S_1$  and  $S_2$  be two red blocks and let  $i_1$  and  $i_2$  be nodes in  $S_1$  and  $S_2$ , respectively, that are not incident to a large edge. There exists a non-reversing path  $(i_1, j), (j, k), (k, i_2)$  from  $i_1$  to  $i_2$  and from the choice of  $i_1$  and  $i_2$  it follows that  $j$  lies in  $S_1$  and  $k$  in  $S_2$ . So  $(j, k)$  is a large edge between  $S_1$  and  $S_2$ . Such an edge exists for every two red blocks  $S_1$  and  $S_2$ , hence the number of large edges incident to two nodes in red blocks is at least  $\frac{1}{2} R(R - 1)$ . Thus the large edges sum up to at least  $\frac{1}{2} B \lceil \sqrt{N} \rceil + \frac{1}{2} R(R - 1)$ , and under the restriction that  $B + R = \lfloor \frac{N}{\sqrt{N}} \rfloor$ , this is  $\Omega(N)$  (viz.,  $\frac{3}{8}N + o(N)$ ). Counting small and large edges

together, it follows that

$$f(N) \geq \begin{cases} N-1 & \text{if } N \leq 4 \\ \lfloor \frac{N}{\sqrt{N}} \rfloor f(\lceil \sqrt{N} \rceil) + \Omega(N) & \text{otherwise} \end{cases}$$

and this implies  $f(N) \geq \Omega(N \log \log N)$ . Subtracting the  $N-1$  basic edges, we find  $F_3(N) \geq \Omega(N \log \log N)$ .  $\square$

The two theorems show that  $F_3(N) = \Theta(N \log \log N)$  as claimed and in the proof of theorem 2.4 a construction is given that uses this number of bridges. The constants hidden in the order-notation are  $2\frac{1}{2}$  and  $\frac{3}{8}$  for the upper and lower bound, respectively.

## 2.4 Diameter 4 and higher constants

Let  $F_k(N)$  be the size of the smallest set  $B$  of bridges such that the diameter of  $T'$  is at most  $k$ . In this subsection matching upper and lower bounds on  $F_k(N)$  are proved (up to a constant factor). These bounds are superlinear, but so only by an extremely slowly growing function. For all practical purposes the results may be considered linear.

**Theorem 2.6**  $F_4(N) \leq O(N \log^* N)$ .

**Proof.** A set of bridges giving  $D \leq 4$  is defined recursively in a way similar to the constructions in theorems 2.2 and 2.4. If  $N \leq 5$  then  $D \leq 4$  already and no bridges are added. Otherwise, apply the following construction in four steps. (0) Let  $W = \lceil \log N \rceil$  and designate the nodes  $W, 2W, \dots$  to be *backbones* of the construction. (1) Connect every node in  $V$  (including the backbones) to the nearest backbone in each direction (if it exists). (2) Add a minimal number of bridges between the backbones in such a way that between any pair of backbones there exists a non-reversing path of length at most 2. (3) Apply the construction recursively on each of the subchains in which the backbones cut the chain.

To show that the diameter is indeed at most 4, consider two nodes,  $i$  and  $j$ ,  $i < j$ , and both *not* a backbone. If  $i$  and  $j$  are in different subchains, let  $M_1$  be the backbone right of  $i$  and  $M_2$  the backbone left of  $j$ . By step (2) of the construction a non-reversing path of length at most 2 between  $M_1$  and  $M_2$  exists, and together with the edges  $(i, M_1)$  and  $(M_2, j)$  this gives a non-reversing path of length at most 4 from  $i$  to  $j$ . The case that  $i$  or  $j$  is a backbone (or both) is left to the reader. If  $i$  and  $j$  are in the same subchain, a non-reversing path of length at most 4 exists because of the recursive application of the construction on each of the subchains.

To compute the number of bridges used in the construction, let  $f(N)$  be the number of bridges used and note that (0) there are  $b = \lfloor \frac{N}{\log N} \rfloor \leq \frac{N}{\log N}$  backbones, (1) it takes less than  $2N$  bridges to connect each node to the nearest backbones, (2) it takes  $F_2(b) = O(b \log b) \leq O(\frac{N}{\log N} \log \frac{N}{\log N}) \leq O(N)$  bridges to connect the



backbones as described in step 2 by theorem 2.2, and (3) recursion is applied to  $\lceil \frac{N}{\lceil \log N \rceil} \rceil \leq \lceil \frac{N}{\log N} \rceil$  subchains of length at most  $\log N$  so this takes at most  $\frac{N}{\log N} \times f(\log N)$  bridges. Hence it follows that

$$f(N) \leq \begin{cases} 0 & \text{if } N \leq 5 \\ 2N + O(N) + \frac{N}{\log N} f(\log N) & \text{otherwise.} \end{cases}$$

The solution to this equation is  $f(N) \leq O(N \log^* N)$ , and hence  $F_4(N) \leq O(N \log^* N)$ .  $\square$

**Theorem 2.7**  $F_4(N) \geq \Omega(N \log^* N)$ .

**Proof.** Let  $f(N)$  denote the minimal size of  $E'$  that gives diameter at most 4, again this is the size of  $B$  plus  $N - 1$ . For  $N \leq 5$ ,  $f(N) = N - 1$ , now let  $N > 5$  and let such an  $E'$  be given. Partition  $V$  in  $\lfloor \frac{N}{\lceil \log N \rceil} \rfloor$  blocks of size at least  $\lceil \log N \rceil$  and call an edge  $(i, j)$  *small* if  $i$  and  $j$  are in the same block, and *large* otherwise.

As two nodes within the same block have distance at most 4, and a non-reversing path of this length lies entirely within the block, a block of size  $s$  contains at least  $f(s)$  small edges. Hence the total number of small edges is at least  $\lfloor \frac{N}{\lceil \log N \rceil} \rfloor f(\lceil \log N \rceil)$ .

As for the large edges, call a block *red* if it contains a node that is *not* incident to a large edge, and *blue* if each of its nodes is incident to at least one large edge. Let  $R$  be the number of red blocks and  $B$  the number of blue blocks. Next a lower bound on the number of large edges is shown both in  $R$  and in  $B$ . As the blue blocks contain at least  $B \lceil \log N \rceil$  nodes, there are at least  $\frac{1}{2} B \lceil \log N \rceil$  large edges incident to a node in a blue block. For the bound in  $R$ , form  $R$  "purple" blocks by extending each red block over the blue blocks to the right of it. Consider the graph  $H = (V_H, E_H)$ , where  $V_H$  is the set of purple blocks and an edge between two block exists iff  $E'$  contains an edge between nodes in these two blocks. It can now be shown that  $H$  has a diameter of at most 2: For purple blocks  $S_1$  and  $S_2$ , take  $i_1$  and  $i_2$  in those blocks that are not incident to large edges. The existence of a non-reversing path in  $T'$  of length at most 4 implies the existence of a non-reversing path of length at most 2 in  $H$  between  $S_1$  and  $S_2$ . From theorem 2.3 it follows that  $E_H$  contains at least  $\Omega(R \log R)$  edges, and as each edge in  $E_H$  corresponds to at least one large edge in  $E'$ , there are at least  $\Omega(R \log R)$  large edges. Using that  $B + R = \lfloor \frac{N}{\lceil \log N \rceil} \rfloor$ , it follows that the number of large edges is at least  $\Omega(N)$ .

Counting small and large edges together, it follows that

$$f(N) \geq \begin{cases} N - 1 & \text{if } N \leq 5 \\ \lfloor \frac{N}{\lceil \log N \rceil} \rfloor f(\lceil \log N \rceil) + \Omega(N) & \text{otherwise} \end{cases}$$

and this implies  $f(N) \geq \Omega(N \log^* N)$ . Subtracting the  $N - 1$  basic edges, we find  $F_4(N) \geq \Omega(N \log^* N)$ .  $\square$

The two theorems show that  $F_4(N) = \Theta(N \log^* N)$  as claimed and in the proof of theorem 2.6 a construction is given that uses this number of bridges. The constants hidden in the order-notation are 3 and  $\frac{1}{4}$  for the upper and lower bound, respectively.

The techniques applied up till now suffice for the analysis of higher constant diameters also. Observe that in theorems 2.2, 2.4, and 2.6 the constructions have in common that a linear number of bridges is used in each level of the recursion. In theorem 2.2 the recursion is on subchains of length  $\frac{1}{2}N$ , hence the recursion depth is  $\log N$  and the number of bridges  $O(N \log N)$ . In theorem 2.4 the recursion is on subchains of length  $\sqrt{N}$ , hence the recursion depth is  $\log \log N$  and the number of bridges  $O(N \log \log N)$ . In theorem 2.6 the recursion is on subchains of length  $\log N$ , hence the recursion depth is  $\log^* N$  and the number of bridges  $O(N \log^* N)$ .

The constructions that are to follow in this subsection share this property, too. Clearly, the recursion depth depends on the size of the subchains, which depends in turn on the number of backbones. In the construction for diameter  $k$  the number of backbones is chosen in such a way that they can be connected to have diameter  $k - 2$  using  $O(N)$  bridges, and thus a relation is established between  $F_k$  and  $F_{k-2}$ . The construction used so far is optimal as shown by the following two theorems.

**Theorem 2.8**  $F_k(N) \leq O(N[\frac{N}{F_{k-2}^{-1}(N)}]^*)$ , where  $[g(N)]^*$  denotes the number of times  $g$  is iterated on  $N$  before the result is at most  $k + 1$ .

**Proof.** The following construction realizes the bound. If  $N \leq k + 1$  then  $D \leq k$  already and no edges are added. Otherwise, use the following 4 steps. (0) Choose a number  $b$  of backbones such that  $F_{k-2}(b) = N$  and place the backbones at regular distance in the chain. (1) Connect every node to the nearest backbones in both directions (if they exist). (2) Add  $F_{k-2}(b)$  bridges between the backbones so as to connect them with a diameter of  $k - 2$ . (3) Apply the construction recursively to the subchains.

The proof that the diameter of the resulting graph is indeed  $k$  and the computation of the number of bridges is as in the previous constructions. Note that the recursion is on subchains of length at most  $\frac{N}{F_{k-2}^{-1}(N)}$  and stops as soon as this length is at most  $k + 1$ . Hence the theorem follows.  $\square$

**Theorem 2.9**  $F_k(N) \geq \Omega(N[\frac{N}{F_{k-2}^{-1}(N)}]^*)$ .

**Proof.** Let  $f(N)$  denote the minimal size of  $E'$  that gives diameter at most  $k$ , again this is the size of  $B$  plus  $N - 1$ . For  $N \leq k + 1$ ,  $f(N) = N - 1$ , now let  $N > k + 1$  and let such an  $E'$  be given. Partition  $V$  in  $b = \lfloor \frac{N}{F_{k-2}^{-1}(N)} \rfloor$  blocks of size at least  $s = \frac{N}{F_{k-2}^{-1}(N)}$  and call an edge  $(i, j)$  *small* if  $i$  and  $j$  are in the same block, and *large* otherwise.

As two nodes within the same block have distance at most  $k$ , and a non-reversing path of this length lies entirely within the block, a block of size  $s$  contains at least  $f(s)$  small edges. Hence the total number of small edges is at least  $bf(s)$ .

As for the large edges, call a block *red* if it contains a node that is *not* incident to a large edge, and *blue* if each of its nodes is incident to at least one large edge. Let  $R$  be the number of red blocks and  $B$  the number of blue blocks. Next a lower bound on the number of large edges is shown both in  $R$  and in  $B$ . As the blue blocks contain at least  $Bs$  nodes, there are at least  $\frac{1}{2}Bs$  large edges incident to a node in a blue block. For the bound in  $R$ , form  $R$  “purple” blocks by extending each red block over the blue blocks to the right of it. Consider the graph  $H = (V_H, E_H)$ , where  $V_H$  is the set of purple blocks and an edge between two block exists iff  $E'$  contains an edge between nodes in these two blocks. It can now be shown that  $H$  has a diameter of at most  $k-2$ : For purple blocks  $S_1$  and  $S_2$ , take  $i_1$  and  $i_2$  in those blocks that are not incident to large edges. The existence of a non-reversing path in  $T'$  of length at most  $k$  implies the existence of a path of length at most  $k-2$  in  $H$  between  $S_1$  and  $S_2$ .  $E_H$  contains at least  $F_{k-2}(R)$  edges, and as each edge in  $E_H$  corresponds to at least one large edge in  $E'$ , there are at least  $F_{k-2}(R)$  large edges. Using that  $B + R = b$ , it follows that the number of large edges is at least  $\Omega(N)$ .

Counting small and large edges together, it follows that

$$f(N) \geq \begin{cases} N-1 & \text{if } N \leq k+1 \\ bf(s) + \Omega(N) & \text{otherwise} \end{cases}$$

and this implies the result stated in the theorem.  $\square$

The derived formula looks awkward but it turns out that  $F_k$  is essentially found by adding a star to the superlinear part of  $F_{k-2}$ .

**Theorem 2.10** *Let  $F_{k-2}(N) = Nf(N)$  where  $f$  is nondecreasing. Then  $F_k(N) = O(N[f(N)]^*(N))$ .*

**Proof.** From  $F_{k-2}(\frac{N}{f(N)}) = \frac{N}{f(N)}f(\frac{N}{f(N)}) \leq N$  follows  $F_{k-2}^{-1}(N) \geq \frac{N}{f(N)}$ . Hence  $\frac{N}{F_{k-2}^{-1}(N)} \leq f(N)$  and  $[\frac{N}{F_{k-2}^{-1}}]^* \leq [f(N)]^*(N)$ .  $\square$

Thus  $F_2(N) = \Theta(N \log N)$ ,  $F_3(N) = \Theta(N \log \log N)$ ,  $F_4(N) = \Theta(N \log^* N)$ ,  $F_5(N) = \Theta(N \log^* N)$ ,  $F_6(N) = \Theta(N \log^{**} N)$ ,  $F_7(N) = \Theta(N \log^{**} N)$ ,  $F_8(N) = \Theta(N \log^{***} N)$ , and so forth.

## 2.5 Diameter $O(\log N)$

Let  $F_{\log}(N)$  be the size of the smallest set  $B$  of bridges such that the diameter of  $T'$  is at most  $O(\log N)$ . In this subsection matching upper and lower bounds on  $F_{\log}(N)$  are proved. First a linear and subsequently an  $O(\frac{N}{\log N})$  upperbound are proved, and then an  $\Omega(\frac{N}{\log N})$  lower bound.

**Theorem 2.11** *A diameter of at most  $2 \log N$  is realized with less than  $N$  bridges.*

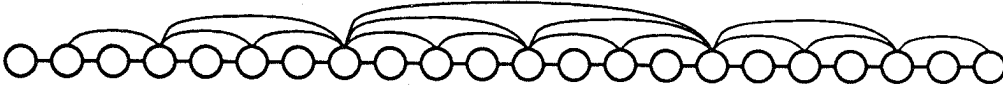


Figure 6: Construction for diameter  $2 \log N$ .

**Proof.** A set of bridges satisfying these bounds contains all bridges whose length is a power of 2 (but  $> 1$ ), and whose incident nodes are multiples of its length. Formally, let  $B = \{(s2^l, (s+1)2^l) : l \geq 1, s \geq 1, (s+1)2^l \leq N\}$  (see figure 6). There are less than  $\frac{N}{2^l}$  bridges of length  $2^l$  and hence the number of bridges is less than  $N$  indeed. (An even lower bound of  $N - \log N - 1$  can be proved and this is sharp for  $N$  a power of 2.)

To prove that the diameter of the resulting graph is indeed logarithmic, consider a shortest non-reversing path between two nodes. First, this path does not contain an edge that is immediately preceded *and* followed by a longer edge. This is because one of its endpoints is *not* a multiple of a *larger* power of 2 than its own length. Second, this path does not contain two edges of equal length, immediately followed *or* preceded by an edge of larger length. This is because in such a configuration the first edge starts in a higher power of 2 than its own length and one bridge exists that spans the two edges (violating the assumption that the path is a shortest path). Third and finally, this path does not contain three consecutive edges of the same length. This is because either the first two or the last two can be replaced by a single bridge (again violating the assumption that the path is a shortest path). It follows that in a shortest path the edge lengths strictly increase to a maximum, which is assumed at most twice, and then strictly decrease. Hence the length of a shortest path is at most twice the number of different edge lengths, which is  $2 \log N$  indeed.  $\square$

**Theorem 2.12** *A diameter of at most  $4 \log N$  is realized with less than  $\frac{2N}{\log N}$  bridges.*

**Proof.** The construction of theorem 2.11 is now applied to nodes (backbones) that are interspaced  $\log N$  apart. Let  $W = \lceil \log N \rceil + 1$  and let  $B = \{(s2^l W, (s+1)2^l W) : l \geq 0, (s+1)2^l W \leq N\}$  (see figure 7). There are less than  $\frac{N}{2^l W}$  bridges of length  $2^l W$  so the total number of bridges is less than  $\frac{2N}{W} < \frac{2N}{\log N}$ .

As in theorem 2.11 it is shown that between any two backbones there is a non-reversing path of length at most  $2 \log(\frac{N}{W}) \leq 2 \log N$ . For arbitrary nodes  $i$  and  $j$  ( $i < j$ ), if  $i$  and  $j$  lie between the same two backbones then there is a path of length  $j - i < W$  between them, consisting of basic edges only. Otherwise, it takes at most  $W - 1$  steps to reach the nearest backbone to the right of  $i$ , at most  $2 \log N$  steps to reach the one to the left of  $j$ , and at most  $W - 1$  to reach  $j$  from there. Hence  $d(i, j) \leq 4 \log N$ .  $\square$

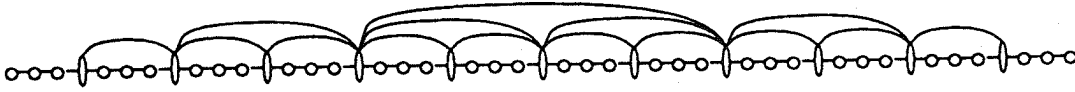


Figure 7: Construction for diameter  $4 \log N$ .

This result is optimal as shown by the following theorem.

**Theorem 2.13** *It takes at least  $\lfloor \frac{N-2}{D} \rfloor$  bridges to have diameter  $D$ .*

**Proof.** As there is a non-reversing path from node  $iD + 1$  to  $N$  of length at most  $D$ , there is a bridge with its left endpoint in the interval  $[iD + 1 \dots (i + 1)D]$  for all  $i$  such that  $0 \leq i, (i + 1)D + 1 < N$ .  $\square$

**Corollary 2.14**  $F_{\log}(N) = O(\frac{N}{\log N})$ .

## 2.6 A Linear Bridge-Diameter Product

For any diameter the product of the number of bridges and the diameter is at least linear in  $N$  as is shown by theorem 2.13. This bound is sharp for a logarithmic diameter as shown in theorem 2.12. In subsections 2.1 through 2.4 however it was shown that a linear product is not realized for any constant diameter (that is, a constant diameter cannot be realized with a linear number of bridges). The following results state that if a linear product is realizable for the diameter being some function  $f(N)$ , then it is also realizable for a function  $g(N)$  if  $g(N) = \Omega(f(N))$ . ( $f$  and  $g$  are nondecreasing functions.)

**Theorem 2.15** *If  $F_f(N)$  bridges suffice to realize a diameter of  $f(N)$ , then  $\frac{N}{g(N)} + F_f(\frac{N}{g(N)})$  bridges suffice to realize a diameter of  $2g(N) + f(\frac{N}{g(N)})$ .*

**Proof.** The construction to show this bound is a generalization of the construction in the proof of theorem 2.12. Designate each  $g(N)^{\text{th}}$  node to be a backbone. Connect each backbone with a bridge to the next one (using  $\frac{N}{g(N)}$  bridges) and build bridges between the backbones so that a non-reversing path of length at most  $f(\frac{N}{g(N)})$  exists between any two backbones (using  $F_f(\frac{N}{g(N)})$  bridges). The number of bridges is now  $\frac{N}{g(N)} + F_f(\frac{N}{g(N)})$  and the length of a shortest non-reversing path between any two nodes is at most  $g(N)$  (to get to the nearest backbone) *plus*  $f(\frac{N}{g(N)})$  (to get to the other backbone) *plus*  $g(N)$  (to get to the destination node), and that is as indicated.  $\square$

**Theorem 2.16** *If  $F_f(N) = O(\frac{N}{f(N)})$  then  $F_{O(g)}(N) = O(\frac{N}{g(N)})$  for  $g(N) = \Omega(f(N))$ .*

**Proof.** Use the previous result, remarking that  $2g(N) + f(\frac{N}{g(N)}) = O(g(N))$  and  $\frac{N}{g(N)} + F_f(\frac{N}{g(N)}) = O(\frac{N}{g(N)})$ .  $\square$

The natural question arises what is the smallest function for which a linear product is possible. It turns out that a linear product is possible when the diameter is  $\log N$ ,  $\log^* N$ ,  $\log^{**} N$ , or, in general, a log with any number of stars. Call these functions *Milky Way functions* (because of all the stars) and write  $M_k$  for the Milky Way function with  $k$  stars. The following theorem establishes that there is a linear product for Milky Way functions.

**Theorem 2.17**  $F_{O(M_k)} = \Theta(\frac{N}{M_k(N)})$ .

**Proof.** The lower bound follows from theorem 2.13. To show the upper bound, recall that  $F_{2k+2}(N) = O(N \times M_k(N))$ . Apply theorem 2.15 to show that  $F_{2M_k(N)+2k+2}(N) = \frac{N}{M_k(N)} + O(\frac{N}{M_k(N)} \times M_k(\frac{N}{M_k(N)})) = O(N)$ . Apply the same theorem again to show that  $F_{4M_k(N)+2k+2}(N) = \frac{N}{M_k(N)} + O(\frac{N}{M_k(N)}) = O(\frac{N}{M_k(N)})$ .  $\square$

As a result (use theorem 2.16) a linear bridge-diameter product is realizable for any function  $f$  that dominates any Milky Way function.

## 2.7 A linear number of bridges

There is a (very small) gap between the constants (considered in subsections 2.1 through 2.4) and the Milky Way functions (considered in subsection 2.6). The former need a superlinear number of bridges, while the latter need only a sublinear number of bridges. The natural question arises what diameter can be realized with a linear number of bridges. This subsection addresses this question briefly.

It turns out that the  $M_k$  differ only a constant factor from the row inverses of the Ackermann function as defined by LaPoutré [1]. It is shown in [1] that these functions satisfy  $M_{\alpha(N)+O(1)}(N) = O(1)$ , where  $\alpha$  is the function commonly known as the *inverse Ackermann function*. Using these results it follows that a diameter of  $\alpha(N)+O(1)$  is realizable with a linear number of bridges.

The lower bound proofs in subsection 2.4 can be modified such that the constant hidden in the big- $\Omega$  notation is really a constant, that is, does not shrink when  $D$  grows. Using these results it follows that a linear number of bridges is necessary to obtain a diameter of  $\alpha(N) + O(1)$ .

**Theorem 2.18**  $F_{\alpha(N)+O(1)}(N) = \Theta(N)$ .

Thus it follows also that a diameter of  $\alpha(N) + O(1)$  is the best one can do with a linear number of edges. Interestingly, by applying theorem 2.15 in a way similar to the proof of theorem 2.17 it follows that a diameter of a *multiple* of this function can be realized using a *sublinear* number of bridges (namely,  $O(\frac{N}{\alpha(N)})$ ).

**Theorem 2.19** For all  $\epsilon > 0$ ,  $F_{(1+\epsilon)\alpha(N)}(N) = \Theta(\frac{N}{\epsilon\alpha(N)})$ .

### 3 Results for the general case

In the sequel let  $T$  be a general tree (rather than a linear chain). As the chain (considered in section 2) is a special case of the tree, the lower bounds proved in section 2 are valid for the general case, too. The emphasis in this section will therefore be on proving upper bounds. More specifically, it is shown that the bounds obtained for constant diameters are valid also for the tree case.

The constructions by which the results are obtained are similar to those used for the chain case. For clarity the general skeleton is repeated here.

0. Choose an integer  $b$  and a subset of size  $b$  of the nodes to be *backbones*.
1. Connect every node to the nearest backbones. (This is done only for non-backbone nodes in the constructions in this section.)
2. Connect the backbones with bridges such that the diameter of the “backbone subnetwork” is 2 less than the required diameter.
3. Apply the construction recursively on the subtrees in which the backbones cut the tree.

This approach, so successfull in the chain case, faces problems in the general case. Some of the problems are highlighted here to serve as an overview of the material presented in this section.

- a. Is it possible to choose  $b$  backbones in a tree in such a way that they cut the tree in pieces of size  $O(\frac{N}{b})$ ? An affirmative answer to this question is necessary in order to bound the recursion depth in the same fashion as in subsection 2.4.
- b. A (non-backbone) node may (in step 1) be connected to more than 2 (or another constant number of) backbones. Is the number of bridges used in step 1 still linear? It turns out that it is, provided that the answer to the previous question is affirmative.
- c. In step 2 an “implicit recursion” is applied because the problem is solved for a smaller diameter. However it is not clear how the backbones form themselves a tree. (Recall that the backbones in a chain form a chain themselves.) This problem is solved by allowing some network nodes that are not backbones themselves, to serve as “super backbones”.

As in section 2, let  $F_k(N)$  denote the minimal number of bridges that is necessary to give a tree on  $N$  nodes a diameter of at most  $k$ . The structure of this section resembles the structure of section 2. The depth of the graph-theoretical results needed in the constructions increases as one goes from diameter 1 to 2, 3, and 4. This is why we chose to devote a separate subsection to each of these diameters.

### 3.1 Diameter 1

The results are here completely the same as for chains, as only the complete graph has diameter 1 and a path of length 1 is always non-reversing.

**Theorem 3.1**  $F_1(N) = \frac{1}{2}(N-1)(N-2) = \Theta(N^2)$

**Proof.** The diameter of  $T'$  is 1, hence  $E'$  connects  $V$  completely, i.e.,  $E' = \{\{i, j\} : i, j \in V\}$  and so the size of  $E'$  is  $\frac{1}{2}N(N-1)$ . Subtracting the  $N-1$  basic edges, find  $|B| = \frac{1}{2}(N-1)(N-2)$ .  $\square$

### 3.2 Diameter 2

This subsection employs the first of a series of non-trivial “cut-lemmas” on trees, stating that backbones can be found as required for a recursive division of the problem. For a node  $v \in V$ , the *subtrees* of  $v$  are the trees that remain when  $v$  and its incident edges are removed from  $T$ .  $T_{vu}$  denotes the subtree of  $v$  that contains  $v$ 's neighbor  $u$  and  $t_{vu}$  denotes its size.

**Lemma 3.2** *There is a node  $c \in V$  such that every subtree of  $c$  has size at most  $\frac{1}{2}N$ .*

**Proof.** Let  $m_v$  be the largest size of a subtree of  $v$  and choose  $c$  to be a node that minimizes  $m_c$ . It will be shown that  $m_c \leq \frac{1}{2}N$ .

Let  $u$  be a neighbor of  $c$  such that  $t_{cu} = m_c$ . By the choice of  $c$ ,  $m_u \geq m_c$  so  $u$  has a neighbor  $v$  such that  $t_{uv} \geq t_{cu}$ . Let in the following  $x$  range over the neighbors of  $u$  other than  $c$ . As  $t_{cu} = 1 + \sum_x t_{ux}$ ,  $t_{ux} < t_{cu}$  and it follows that  $v = c$ . Thus  $t_{uc} \geq t_{cu}$  and, as  $t_{uc} + t_{cu} = N$ ,  $t_{cu} \leq \frac{1}{2}N$  follows.  $\square$

**Theorem 3.3** ([3])  $F_2(N) \leq O(N \log N)$ .

**Proof.** The claimed number is realized with the following construction. If the diameter of  $T$  is at most 2 already, no bridges are added. Otherwise, choose  $M$  to be a node such that all subtrees of  $M$  have size at most  $\frac{1}{2}N$ . (This choice is possible by lemma 3.2.) Connect all nodes to  $M$ . Apply the construction recursively to every subtree of  $M$ .

To show that the diameter of the resulting graph is at most 2, let  $i$  and  $j$  be nodes in  $V$ . If  $i = M$  or  $j = M$ , an edge  $(i, j)$  exists by construction. If  $i$  and  $j$  are in different subtrees of  $M$ , edges  $(i, M)$  and  $(M, j)$  exist and the path  $(i, M), (M, j)$  is non-reversing. If  $i$  and  $j$  are in the same subtree a non-reversing path of length at most 2 exists because of the recursive application of the construction.

To analyse the number of bridges used in the construction, note that a linear number of bridges is used in each level of recursion (because there is at most one



bridge from each node to  $M$ ) and the recursion has depth less than  $\log N$ . Hence less than  $N \log N$  bridges are used.  $\square$

The constant hidden in the order-notation is 1. The result is asymptotically optimal by theorem 2.3.

### 3.3 Diameter 3

This section requires some more complex graph theoretic results. The idea of the construction to follow is to use  $\sqrt{N}$  backbones and apply the construction recursively to subtrees of size  $\frac{N}{\sqrt{N}}$ . Hence a result is necessary that an appropriate choice of the backbones is possible. Also it is necessary to establish that all nodes may be connected to the surrounding backbones using a linear number of bridges.

Some notations are introduced first. Given a selected set of backbones (sometimes called *cutpoints*), the *subtrees* are defined to be the trees that remain after removal of the backbones and their adjacent edges. The *borders* of a subtree are the backbones that were connected to this subtree prior to the removal.

**Lemma 3.4** *For a tree  $T$  and an integer  $K \leq N$ , there is a node  $c$  such that*

1. *at most one subtree of  $c$  has size  $> K$ , and*
2. *the subtrees of  $c$  whose size is  $\leq K$  contain at least  $K - 1$  nodes together.*

**Proof.** Call a subtree *heavy* if its size is  $> K$  and *light* if its size is  $\leq K$ . Consider the following two cases:

1. **There are nodes that have two (or more) heavy subtrees.** The set  $D$  of such nodes is connected: let  $d_1$  and  $d_2$  be in  $D$  and  $e$  on the path between  $d_1$  and  $d_2$ . The two subtrees of  $e$ , containing  $d_1$  and  $d_2$ , respectively, are heavy and hence  $e$  is in  $D$ . Hence  $D$  is connected. As  $D$  is a connected subgraph of a tree,  $D$  is a tree itself, let  $d$  be a leaf of  $D$ .  $d$  has two or more heavy subtrees, but has at most one neighbor that has two or more heavy subtrees. It follows that  $d$  has a neighbor  $c$  such that  $T_{dc}$  is heavy and  $c$  has at most one heavy subtree. As  $T_{cd}$  contains another heavy subtree of  $d$  it is heavy, hence all the other subtrees of  $c$  are light (by the choice of  $c$ ). As  $T_{dc}$  is heavy, these subtrees of  $c$  have a total weight of at least  $K - 1$ , hence  $c$  has the required properties.
2. **All nodes have at most one heavy subtree.** Let  $c$  be the node for which the sum of the sizes of the *light* subtrees is maximal. It remains to show that this sum is at least  $K - 1$ . If *all* subtrees of  $c$  are light the sum is  $N - 1$ . Otherwise, let  $d$  be the (only) neighbor of  $c$  for which  $T_{cd}$  is heavy. The sum of the sizes of light subtrees of  $c$  is now  $t_{dc} - 1$ . By the choice of  $c$ , the sum of the sizes of the light subtrees of  $d$  is at most  $t_{dc} - 1$  also, hence  $T_{dc}$  is not light. It follows that  $t_{dc} - 1 > K - 1$ .

As these two cases are complimentary, the proof is complete.  $\square$

**Lemma 3.5**  *$b$  cutpoints can be chosen in such a way that all the subtrees have size at most  $\lceil \frac{N}{b+1} \rceil$ .*

**Proof.** Use induction on  $b$ . If  $b = 0$  there are no cutpoints and the result is true. If  $b > 0$ , let  $K = \lceil \frac{N}{b+1} \rceil$  and choose a node  $c$  as in lemma 3.4.  $c$  has at most one heavy subtree of size  $\leq N - 1 - (K - 1) = N - K$ . Use the induction hypothesis to show that in this subtree  $b - 1$  cutpoints can be chosen so as to cut it in subtrees of size at most  $\lceil \frac{N-K}{(b-1)+1} \rceil \leq K$ .  $\square$

For the following result, assume a tree is given with  $b$  cutpoints in such a way that all subtrees have size at most  $\lceil \frac{N}{b+1} \rceil$ .

**Lemma 3.6** *All nodes other than the cutpoints can be connected with all borders of their subtree using at most  $2N$  bridges.*

**Proof.** To count the bridges, choose one arbitrary cutpoint  $r$  as the root of the tree. Note that every bridge is incident to exactly one cutpoint. Call a bridge *upstream* if the cutpoint of that bridge is closer to  $r$ , and *downstream* if the cutpoint is further away from  $r$  than the non-cutpoint endpoint of the bridge (see figure 8). Each

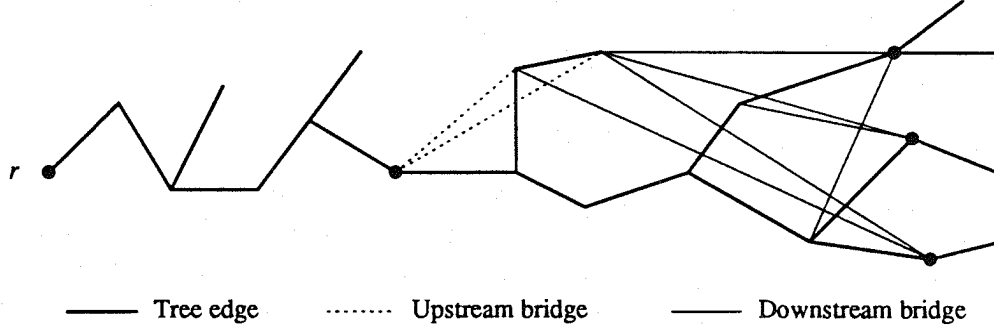


Figure 8: Upstream and downstream bridges.

non-cutpoint node is incident to at most one upstream bridge so there are at most  $N - b$  upstream bridges. The downstream bridges are counted per cutpoint. Each cutpoint (other than  $r$ ) is incident only to one downstream bridge from each node in the subtree in the direction of the root (see figure 8). Hence there are at most  $(b - 1)\lceil \frac{N}{b+1} \rceil < N + b$  downstream bridges, and the total number of bridges is less than  $2N$ .  $\square$

**Theorem 3.7**  $F_3(N) \leq O(N \log \log N)$ .

**Proof.** A recursive construction is given that uses the claimed number of bridges. If the diameter is at most 3 already, no bridges are added. Otherwise, set  $b = \sqrt{N}$ , (0) select  $b$  backbones in such a way that the subtrees have size at most  $\sqrt{N}$  (such a choice is possible by lemma 3.5), and (1) connect every non-backbone with all borders of its subtree. (2) Connect the backbones completely (i.e., add a bridge between every pair of backbones) and (3) apply the construction recursively to each of the subtrees.

To prove that the diameter of the resulting graph is at most 3, let  $i$  and  $j$  be two nodes from  $V$ . If  $i$  and  $j$  are in the same subtree, a non-reversing path of length at most 3 between them exists because of the recursive application of the construction. Otherwise, let  $M_1$  be the first backbone on the path in  $T$  from  $i$  to  $j$  and  $M_2$  the last backbone on this path. (If  $i$  or  $j$  is a backbone, choose  $M_1 = i$  or  $M_2 = j$ .) The path  $(i, M_1), (M_1, M_2), (M_2, j)$  exists and is non-reversing.

To analyse the number of bridges used in the construction, use lemma 3.6 to show that a linear number of bridges is used in step (1) of the construction.  $\frac{1}{2}b(b-1) < \frac{1}{2}N$  bridges are used in step (2), and the recursion depth is less than  $\log \log N$  because recursion is on subtrees of size  $\sqrt{N}$ . Hence less than  $O(N \log \log N)$  bridges are used.  $\square$

The constant hidden in the order-notation is  $2\frac{1}{2}$  as in the case of linear chains. The result is asymptotically optimal by theorem 2.5.

### 3.4 Diameter 4 and higher constants

This subsection gives a solution for trees for any constant non-reversing diameter. The number of bridges used is the same as in the constructions for linear chains and hence (see subsection 2.4) they are asymptotically optimal. In the introduction of this section it was noted that the backbones of a tree do not form a tree themselves. To illustrate this, consider a subtree with borders  $a, b, c, \dots$ , see figure 9. A non-

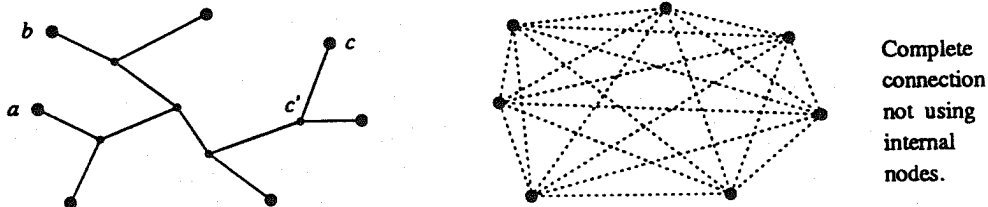


Figure 9: The Backbones do not form a tree.

reversing path from  $a$  to  $b$  does not run through  $c$  (because the edge  $(c, c')$  would be represented twice). So node  $c$  cannot be used as a “super backbone” to divide the set of borders. However, connecting each pair of borders directly by a bridge

results in a graph that is not a tree, so that the result for smaller diameters cannot be applied. Moreover, the number of bridges would be too high. The solution to connect the backbones is to choose a “higher level” backbone from the nodes in the subtree (where necessary). There a cutpoint can be found to divide the set of backbones appropriately.

It is thus necessary to express the different status of backbones and subtree nodes in the construction of the “interbackbone” subnetwork. The latter may be used in the construction but only the former need be connected through non-reversing paths. This leads to the formulation of the *Restricted Bridge Problem*.

For this problem let  $T = (V, E)$  be a tree whose nodes are colored either *black* or *white*. It is required to add bridges in such a way that in  $E'$  a non-reversing path exists between any pair of *black* nodes. Bridges may however be incident to white nodes also. The *restricted diameter* of  $T$  is the largest distance between any two black nodes. Recall that  $F_k(N)$  is the number of bridges that is necessary to give a tree on  $N$  nodes a diameter of at most  $k$ . Define  $G_k(N)$  to be the number of bridges necessary to give a colored tree with  $N$  black nodes a restricted diameter of at most  $k$ .

**Lemma 3.8**  $F_k(N) \leq G_k(N)$ .

**Proof.** Assume all the nodes of  $T$  are black.  $\square$

To prove bounds on  $G_k(N)$  it is necessary to have “colored” versions of lemmas 3.5 and 3.6. In the sequel let  $T$  be a colored tree with  $N$  black nodes.

**Lemma 3.9** For  $K \leq N$  there is a node  $c$  such that

1. at most one subtree of  $c$  contains more than  $K$  black nodes, and
2. the subtrees of  $c$  that contain at most  $K$  black nodes, together with  $c$  itself, contain at least  $K$  black nodes.

**Proof.** Call a subtree heavy if it contains more than  $K$  black nodes and light otherwise. The proof is now as for lemma 3.4.  $\square$

**Lemma 3.10**  $b$  cutpoints can be chosen in a colored tree in such a way that every subtree contains at most  $\lceil \frac{N}{b+1} \rceil$  black nodes.

**Proof.** As for lemma 3.5.  $\square$

**Lemma 3.11** Assume cutpoints as in lemma 3.10 are given. All black nodes that are not a cutpoint can be connected with all borders of their subtrees using at most  $2N + b$  bridges.

**Proof.** As for lemma 3.6. Now there are at most  $N$  upstream bridges and at most  $N + b$  downstream bridges.  $\square$

In the constructions to follow  $b \leq N$  always and so the number of bridges is at most  $3N$ . It is now easily established that the  $G_k$  are the same functions as those found in section 2.4.

**Theorem 3.12** *The following bounds hold for  $G_k$ .  $G_1(N) \leq O(N^2)$ ,  $G_2(N) \leq O(N \log N)$ , and  $G_k(N) \leq O(N[\frac{N}{G_{k-2}(N)}]^*)$  for  $k \geq 3$ . (Here  $[g(N)]^*$  denotes the number of times  $g$  is iterated on  $N$  before the result is at most 1.)*

**Proof.** For the claim about  $G_1(N)$ , recall that only a complete connection of the black nodes realizes a restricted diameter of 1.

For the claim about  $G_2(N)$  use the following construction. If  $N \leq 1$  then the restricted diameter is at most 2 already and no bridges are added. Otherwise, (0) let  $M$  be a node such that every subtree of  $M$  contains at most  $\frac{1}{2}N$  black nodes. (1) Connect all the black nodes to  $M$ . (2) Apply the construction recursively to the subtrees of  $M$ . The proof of correctness of this construction as well as the analysis of the number of bridges is as in the proof of theorem 3.3.

For  $k \geq 3$  use the following construction. If  $N \leq 1$  then the restricted diameter is at most  $k$  already and no bridges are added. Otherwise, (0) take  $b$  such that  $G_{k-2}(b) = N$  and choose  $b$  cutpoints in such a way that every subtree contains at most  $\lceil \frac{N}{b+1} \rceil$  black nodes. (This is possible according to lemma 3.10). (1) Connect every black node with all the borders of its subtree. (2) Add bridges between the backbones so that a non-reversing path of length at most  $k - 2$  exists between any two of them. (3) Apply the construction recursively to each of the subtrees.

To show that the restricted diameter of the resulting graph is at most  $k$  indeed, let  $i$  and  $j$  be black nodes in  $V$ . If  $i$  and  $j$  are in the same subtree, a non-reversing path of length at most  $k$  exists because of the recursive application of the construction. Otherwise, let  $M_1$  be the first backbone on the (unique) path from  $i$  to  $j$  in  $T$  and  $M_2$  the last backbone on this path. By construction bridges  $(i, M_1)$  and  $(M_2, j)$  exist, as well as a non-reversing path of length at most  $k - 2$  between  $M_1$  and  $M_2$ . The concatenation of the two bridges and the non-reversing path is a non-reversing path of length at most  $k$  between  $i$  and  $j$ .

To analyse the number of bridges used, note that at most  $3N$  bridges are used in step (1) (lemma 3.11), and at most  $G_{k-2}(b) = N$  bridges are used in step (2). Hence a linear number of bridges is used in every level of the recursion. Recursion is on subtrees of size bounded by  $\lceil \frac{N}{b+1} \rceil \approx \frac{N}{G_{k-2}(N)}$ , hence the recursion depth is at most  $\lceil \frac{N}{G_{k-2}(N)} \rceil^*$ . The result claimed in the theorem follows.  $\square$

For a convenient representation of the functions  $G_k$ , see theorem 2.10 and the remarks following it. It has now been shown that the complexity of the problem for trees is asymptotically the same as for linear chains. The actual values may be a

Diameter	for the chain	remarks
1	$\Theta(N^2)$	for trees also
2	$\Theta(N \log N)$	for trees also
3	$\Theta(N \log \log N)$	for trees also
4	$\Theta(N \log^* N)$	for trees also
5	$\Theta(N \log^* N)$	for trees also
6	$\Theta(N \log^{**} N)$	for trees also
7	$\Theta(N \log^{**} N)$	for trees also
8	$\Theta(N \log^{***} N)$	for trees also
etc.	etc.	
$\alpha(N) + O(1)$	$\Theta(N)$	
$(1 + \epsilon)\alpha(N)$	$\Theta(\frac{N}{\epsilon \alpha(N)})$	
$O(f(N))$	$\Theta(\frac{N}{f(N)})$	$f(N) = \Omega(\alpha(N))$
$N - 1$	0	

Figure 10: Optimal number of bridges for various diameters.

little bit higher, as some more bridges may be used in step (1) of the recursion, and recursion stops when the sizes of subtrees is reduced to 1 rather than  $k + 1$ .

## 4 Summary of results

The results obtained in the previous two sections are summarized in figure 10. The order-optimal number of needed bridges are given for the linear chain for diameters of constant size, of size  $\alpha(N) + O(1)$ , and of size  $O(f(N))$  where  $f(N)$  grows faster than  $\alpha(N)$ . This leaves only few questions open. Most interesting is probably the question to determine the exact constant factors involved. Less interesting perhaps is what happens if we require a diameter of size  $f(N)$ , where  $\lim_{N \rightarrow \infty} = \infty$ , but  $f(N)$  grows slower than  $\alpha(N)$ .

## 5 Applications of the problem

### 5.1 Subsequence composition

Given are sets  $S_1$  through  $S_N$  and functions  $g_1$  through  $g_{N-1}$  where  $g_i$  is a function from  $S_i$  to  $S_{i+1}$ . Let  $G_{ij}$  ( $i < j$ ) denote the function from  $S_i$  to  $S_j$  defined by the composition of  $g_i$  through  $g_{j-1}$ . Compositions of functions can be computed at unit cost and functions as well as compositions can be stored at unit cost. It is required that the  $g_i$  are stored in such a way that the  $G_{ij}$  can be retrieved efficiently.

The results for the linear chain (section 2) are helpful here to decide what information to store. Let a bridge  $(i, j)$  in a solution correspond to function  $G_{ij}$  that is precomputed and stored. The number of bridges corresponds to the space complexity of the resulting data structure. The diameter of the resulting graph corresponds to the time complexity of a query.

A special case of this problem is matrix subsequence product, where  $S_i$  is  $\mathbb{Z}^{d_i}$  and  $g_i$  is a  $d_i \times d_{i+1}$  matrix.

## 5.2 Database queries, generalisation to graphs

Let a set  $V$  of domains be given and a set  $E$  of binary relations between these domains. A relation defines a function in each direction, but these two functions are not each others inverse. (In general the functions have no inverse.) Given two domains  $x, y \in V$  and a path  $p$  of relations in  $E$  between  $x$  and  $y$ , a unique relation  $G_p$  from  $x$  to  $y$  is defined (*viz.*, by the composition of the relations on the path). It is required to store relations and their compositions in such a way that the  $G_p$  can be retrieved efficiently. As in section 5.1 solutions to the problem are helpful to organize the data structure. In the case that  $G$  is a tree, there is exactly one path between any two nodes, hence  $p$  is uniquely determined by  $x$  and  $y$ . The non-reversibility condition is necessary because the result of a relation composed with its reversed relation is in general not the identity relation. The problem for trees was addressed in this paper. The problem was suggested in [2] and some results (theorems 2.1 and 3.3) are found in [3].

The general graph problem leads to the following generalisation of the problem. Given is a graph  $G = (V, E)$ . It is required to find a set  $B$  of (simple) paths in  $G$  (the bridges) in such a way that any arbitrary (simple) path in  $G$  is the concatenation of as few paths as possible from  $B$ . We did not study the general problem.

## 5.3 Sparse matrices

Graphs can be represented by a boolean adjacency matrix  $M$  with  $M[i, j] = 1$  if  $(i, j) \in E$  and 0 otherwise. Directed chains correspond with upper triangular matrices with 1's just above the main diagonal. The results in section 2 can be reformulated to say that if the  $k^{\text{th}}$  power of an  $N \times N$  upper triangular matrix  $M$  has all 1's above the diagonal, then  $M$  has at least  $F_k(N)$  1's above the first super diagonal. (The first super diagonal is the diagonal above the main diagonal. All its entries are 1 for a chain because it represents edges from  $i$  to  $i + 1$ .)

A matrix is called *sparse* if many of its entries are zero's and often these matrices are stored using schemes that suppress the zero's and only store non-zero's. When such a scheme is used it is interesting to know how sparse the result of a matrix operation, like a multiplication, is. The statement for boolean matrices implies that the  $k^{\text{th}}$  power of an upper triangular matrix can be completely filled (have no zero's

above the diagonal) only if it has at least  $F_k(N)$  non-zero entries above the first super diagonal.

**Acknowledgements:** We thank Han LaPoutré, Una-May O'Reilly, Anneke Schoone, Jeff Sidney, Richard Tan, and Jan van Leeuwen for their participation in stimulating discussions on the non-reversing diameter problem.

## References

- [1] La Poutré, J.A., *New Techniques for the Union-Find Problem*, Technical Report RUU-CS-89-19, Utrecht University, Utrecht.
- [2] Santoro, N., *Efficient Abstract Implementations for Relational Data Structures*, Ph.D. Thesis, University of Waterloo, 1979.
- [3] Santoro, N., *(Time  $\times$  Space)-Efficient Implementations of Hierarchical Conceptual Models*, in: J. van Leeuwen (ed.), *Graph-Theoretic Concepts in Computer Science*, Springer Verlag LNCS 344, 1988, pp. 180-189.



**School of Computer Science, Carleton University  
Bibliography of Technical Reports**

- SCS-TR-127     **On the Packet Complexity of Distributed Selection**  
A. Negro, N. Santoro and J. Urrutia, November 1987.
- SCS-TR-128     **Efficient Support for Object Mutation and Transparent Forwarding**  
D.A. Thomas, W.R. LaLonde and J. Duimovich, November 1987.
- SCS-TR-129     **Eva: An Event Driven Framework for Building User Interfaces in Smalltalk**  
Jeff McAffer and Dave Thomas, November 1987.
- SCS-TR-130     **Application Frameworks: Experience with MacApp**  
Out of print     John R. Pugh and Cefee Leung, December 1987.  
Available in an abridged version in the Proceedings of the Nineteenth ACM SIGSCE  
Technical Symposium, February 1988, Atlanta, Georgia.
- SCS-TR-131     **An Efficient Window Based System Based on Constraints**  
Danny Epstein and Wilf R. LaLonde, March 1988.
- SCS-TR-132     **Building a Backtracking Facility in Smalltalk Without Kernel Support**  
See Third International Conference on OOPSLA, San Diego, Calif., Sept. '88.  
Wilf R. LaLonde and Mark Van Gulik, March 1988.
- SCS-TR-133     **NARM: The Design of a Neural Robot Arm Controller**  
Daryl H. Graf and Wilf R. LaLonde, April 1988.
- SCS-TR-134     **Separating a Polyhedron by One Translation from a Set of Obstacles**  
Otto Nurmi and Jörg-R. Sack, December 1987.
- SCS-TR-135     **An Optimal VLSI Dictionary Machine for Hypercube Architectures**  
Frank Dehne and Nicola Santoro, April 1988.
- SCS-TR-136     **Optimal Visibility Algorithms for Binary Images on the Hypercube**  
Frank Dehne, Quoc T. Pham and Ivan Stojmenovic, April 1988.
- SCS-TR-137     **An Efficient Computational Geometry Method for Detecting Dotted  
Lines in Noisy Images**  
F. Dehne and L. Ficocelli, May 1988.
- SCS-TR-138     **On Generating Random Permutations with Arbitrary Distributions**  
B. J. Oommen and D.T.H. Ng, June 1988.
- SCS-TR-139     **The Theory and Application of Uni-Dimensional Random Races With  
Probabilistic Handicaps**  
D.T.H. Ng, B.J. Oommen and E.R. Hansen, June 1988.
- SCS-TR-140     **Computing the Configuration Space of a Robot on a Mesh-of-Processors**  
F. Dehne, A.-L. Hassenklover and J.-R. Sack, June 1988.
- SCS-TR-141     **Graphically Defining Simulation Models of Concurrent Systems**  
H. Glenn Brauen and John Neilson, September 1988
- SCS-TR-142     **An Algorithm for Distributed Mutual Exclusion on Arbitrary Networks**  
H. Glenn Brauen and John E. Neilson, September 1988
- SCS-TR-143 to 146 are unavailable.
- SCS-TR-147     **On Transparently Modifying Users' Query Distributions**  
B.J. Oommen and D.T.H. Ng, November 1988
- SCS-TR-148     **An  $O(N \log N)$  Algorithm for Computing a Link Center in a Simple Polygon**  
H.N. Djidjev, A. Lingas and J.-R. Sack, July 1988  
Available in STACS 89, 6th Annual Symposium on Theoretical Aspects of Computer Science,  
Paderborn, FRG, February 16-18, 1989, Lecture Notes in Computer Science, Springer-Verlag No.  
349

- SCS-TR-149      **Smallscript: A User Programmable Framework Based on Smalltalk and Postscript**  
 Kevin Haaland and Dave Thomas, November 1988
- SCS-TR-150      **A General Design Methodology for Dictionary Machines**  
 Frank Dehne and Nicola Santoro, February 1989
- SCS-TR-151      **On Doubly Linked List ReOrganizing Heuristics**  
 D.T.H. Ng and B. John Oommen, February 1989
- SCS-TR-152      **Implementing Data Structures on a Hypercube Multiprocessor, and Applications in Parallel Computational Geometry**  
 Frank Dehne and Andrew Rau-Chaplin, March 1989
- SCS-TR-153      **The Use of Chi-Squared Statistics in Determining Dependence Trees**  
 R.S. Valiveti and B.J. Oommen, March 1989
- SCS-TR-154      **Ideal List Organization for Stationary Environments**  
 B. John Oommen and David T.H. Ng, March 1989
- SCS-TR-155      **Hot-Spot Contention in Binary Hypercube Networks**  
 Sivarama P. Dandamudi and Derek L. Eager, April 89
- SCS-TR-156      **Some Issues in Hierarchical Interconnection Network Design**  
 Sivarama P. Dandamudi and Derek L. Eager, April 1989
- SCS-TR-157      **Discretized Pursuit Linear Reward-Inaction Automata**  
 B.J. Oommen and Joseph K. Lancot, April 1989
- SCS-TR-158      **Parallel Fractional Cascading on a Hypercube Multiprocessor**  
 Frank Dehne, Afonso Ferreira and Andrew Rau-Chaplin, May 1989
- SCS-TR-159      **Epsilon-Optimal Stubborn Learning Mechanisms**  
 J.P.R. Christensen and B.J. Oommen, June 1989
- SCS-TR-160      **Disassembling Two-Dimensional Composite Parts Via Translations**  
 Doron Nussbaum and Jörg-R. Sack, June 1989
- SCR-TR-161      **Recognizing Sources of Random Strings**  
 (revised) R.S. Valiveti and B.J. Oommen, January 1990  
 Revised version of SCS-TR-161 "On the Data Analysis of Random Permutations and its Application to Source Recognition", published June 1989
- SCS-TR-162      **An Adaptive Learning Solution to the Keyboard Optimization Problem**  
 B.J. Oommen, R.S. Valiveti and J. Zgierski, October 1989
- SCS-TR-163      **Finding a Central Link Segment of a Simple Polygon in  $O(N \log N)$  Time**  
 L.G. Alexandrov, H.N. Djidjev, J.-R. Sack, October 1989
- SCS-TR-164      **A Survey of Algorithms for Handling Permutation Groups**  
 M.D. Atkinson, January 1990
- SCS-TR-165      **Key Exchange Using Chebychev Polynomials**  
 M.D. Atkinson and Vincenzo Acciari, January 1990
- SCS-TR-166      **Efficient Concurrency Control Protocols for B-tree Indexes**  
 Ekow J. Otoo, January 1990
- SCS-TR-167      **A Hierarchical Stochastic Automaton Solution to the Object Partitioning Problem**  
 B.J. Oommen, January 1990
- SCS-TR-168      **Adaptive List Organizing for Non-stationary Query Distributions. Part I: The Move-to-Front Rule**  
 R.S. Valiveti and B.J. Oommen, January 1990