Assigning Bookmarks in Perfect Binary Trees

Jurek Czyzowicz[*§] Evangelos Kranakis[†§¶]
Danny Krizanc[‡] Andrzej Pelc[*§]
Miguel Vargas Martin[||§¶††]

Abstract

Consider a tree $T = (V, E)$ with root $r \in V$ and $|V| = N$. Let p_v be the probability that a user wants to access node v. A *bookmark* is an additional link from r to any other node of T. We want to add k bookmarks to T, so as to minimize the expected access cost from r, measured by the average length of the shortest path. We present a characterization of an optimal assignment of k bookmarks in a perfect binary tree with uniform probability distribution of access and $k \leq \sqrt{N+1}$.

1 Introduction

It is common that users of a web site (or any distributed information system) and its designer perceive the web site in a different way. This discrepancy is reflected in users having to traverse "costly" paths in order to reach the pages they are interested in. We say that a path is costly either because it is "too" long or because the pages in it are "too" large (in terms of bytes). A well-designed web site will avoid useless traffic, save time to users, and reduce the web server workload. We endeavour to improve web

[*]Département d'Informatique, Université du Québec en Outaouais. Email: {jurek,pelc@uqo.ca}.

[†]School of Computer Science, Carleton University. Email: kranakis@scs.carleton.ca.

[‡]Department of Mathematics, Wesleyan University. Email: dkrizanc@wesleyan.edu.

[||]University of Ontario Institute of Technology. Email: miguel.vargasmartin@uoit.ca.

[§]Research supported in part by NSERC (Natural Sciences and Engineering Research Council of Canada) grant.

[¶]Research supported in part by MITACS (Mathematics of Information Technology and Complex Systems) NCE (Networks of Centres of Excellence) grant.

[††]Research supported in part by CONACYT (Science and Technology Council of Mexico) grant.

site access by improving the hyperlink structure. We propose to do this via a careful and methodical design of bookmarks (shortcuts) assignment to related pages. The idea is conceptually simple, ***bring the most popular pages closer to the home page.*** This is the approach we follow in this paper, for the special case of distributed information systems which can be represented using perfect binary trees.

Consider a perfect binary tree (i.e., a binary tree with all leaf nodes at depth δ and all internal nodes with two children) $T = (V, E)$ with root $r \in V$ and $|V| = N = 2^\delta - 1$. Let p_v be a value associated to node $v \in V$ representing the probability that a user wants to access node v. A ***bookmark*** is an additional link from r to any other node of T. Assuming this we pose the following k-Bookmark Assignment Problem (k-BAP): Find an assignment of k bookmarks in a perfect binary tree with known access probabilities, that will minimize the expected number of steps to reach any node of the tree from the root.

1.1 Related work and results of the paper

An equivalent problem concerns the placement of k web proxies in the Internet. This latter problem has been studied in [1, 2] and what they call the optimal placement of k proxies is equivalent to our problem k-BAP. It has been proven in [3] that k-BAP is NP-hard for general directed graphs. On the other hand, [2] provides a dynamic programming solution of k-BAP on a tree of N nodes. Their algorithm runs in $O(N^3k^2)$. For a special case of an N-node line an $O(N^2k)$ time algorithm is given in [1]. We present a characterization of an optimal assignment of k bookmarks in a perfect binary tree with uniform probability distribution of access and $k \leq \sqrt{N+1}$. Thus, our work proposes an optimal solution to a special case of the work of [2] not only by providing an efficient solution when the distribution is uniform but also by giving the complete characterization of optimal bookmark assignments in this special case. From our characterization follows a bookmark assignment independent on the size of T.

2 Notation and Terminology

Consider a perfect binary tree (i.e., a binary tree with all leaf nodes at depth δ and all internal nodes with two children) $T = (V, E)$ with root $r \in V$ and $|V| = N = 2^\delta - 1$. Let p_v be the access probability associated to node $v \in V$. Let $d(v)$ be the ***distance*** from the root r to node v in T. The ***expected number of steps*** to reach a node of T from the root is defined by

$$E[T] = \sum_{v \in V} p_v \cdot d(v) \tag{1}$$

The *k-Bookmark Assignment Problem (k-BAP)* consists of minimizing Equation 1 by adding at most k bookmarks to the tree. A *bookmark* b, is an additional directed edge $(r, v), v \neq r$, added to the original tree. A bookmark set on T is a set $B \subset V$; thus we identify a bookmark (r, v) with the node v itself. Consider a set $B = \{b_1, \ldots, b_k\}$ of bookmarks assigned to tree T; the resulting graph is denoted by T^B. The *distance* from the root to a node v in T^B is denoted by $d_B(v)$. Thus, the expected number of steps to reach a node of T^B from the root is defined by

$$E[T^B] = \sum_{v \in V} p_v \cdot d_B(v) \tag{2}$$

The *gain* of B is defined by

$$G(B) = E[T] - E[T^B] \tag{3}$$

The *gain* of a single bookmark, $b \in B$, is denoted by

$$g_B(b) = d(b) - d_B(b), \tag{4}$$

where $d_B(b) = 1$.

The gain for a node v in T^B is denoted by

$$g_B(v) = d(v) - d_B(v) \tag{5}$$

A bookmark set B of size k is optimal if $G(B) \geq G(B')$ for any bookmark set B' of size k.

Consider a set of bookmarks B assigned to tree $T = (V, E)$. We say that a bookmark $x \in V$ *dominates* node $v \in V$, if on the path $(x = v_0, v_1, \cdots, v_l = v)$ only x may belong to B. When a bookmark b dominates v, the shortest path from root r to v uses the edge (r, b). The set of all nodes of V dominated by b is called the *domain* of b. We say that two bookmarks b_1 and b_2 are *independent* if one is not an ancestor of the other. Observe that the domains of two bookmarks b_i and b_j are always disjoint and the family of domains of the set $B \cup \{r\}$ partitions V. We say that *B covers T* if each leaf v of T belongs to the domain of some $b_i \in B$. A bookmark $b \in B$ is called *exposed* in T^B if its domain does not contain any other bookmark of B. For $l \geq 0$, the *level* l of tree T consists of all nodes at distance l from r. If $l < q$, we say that l is *above* q or q is *deeper* than l. The *depth* of the tree is the number of its deepest level, n.

3 Assignment of Bookmarks

We construct an optimal assignment of k bookmarks in the perfect binary tree of $N \geq 15$ nodes with uniform probability distribution, for $k \leq \sqrt{N+1}$. First observe that if $k \leq 4$ then the optimal assignment of bookmarks is to place all of them on level 2, since bookmarks on level 0 and 1 would be wasted. Hence we may assume $k > 4$.

Theorem 1. *For a perfect binary tree T of $N \geq 15$ nodes with uniform distribution, a set B of k bookmarks, such that $4 < k \leq \sqrt{N+1}$, is optimal, if and only if all of the following conditions hold.*

1. *B covers T,*
2. *all bookmarks in B are independent,*
3. *all bookmarks in B are placed on two consecutive levels of T.*

In order to prove this theorem, we formulate several transformations showing which transformations of a bookmark set increase its gain, but first we use the following lemma to guarantee the existence of a set of bookmarks satisfying the above conditions.

Lemma 1. *For a perfect binary tree T with m leaves and for any number $k \leq m$, there exists a set of k independent bookmarks covering T, placed on at most two consecutive levels of T.*

Proof. Let $l = \lfloor \log(k) \rfloor$. If we take any set S of $2^{l+1} - k$ bookmarks on level l, we can place exactly $2k - 2^{l+1}$ bookmarks independent on S on level $l+1$. They all cover T.

■

Note that the distribution of bookmarks given in Lemma 1 is unique up to the choice of its subset on level l. Thus in the remainder of this section we will suppose without loss of generality that in the set of bookmarks from Theorem 1 the bookmarks from the higher level are the leftmost nodes of T on this level (cf. Figure 1).

The following fact will be useful below.

Fact 1. *Let B_i be a set of bookmarks defined for a tree T. Suppose that B_i has been altered by replacing some of its bookmarks $b_{i_1}, b_{i_2}, \ldots, b_{i_m}$ by $b_{j_1}, b_{j_2}, \ldots, b_{j_m}$ thus forming a new bookmark set B_j. For each node v which is not in the domain of any node from $X = \{b_{i_1}, b_{i_2}, \ldots, b_{i_m}, b_{j_1}, b_{j_2}, \ldots, b_{j_m}\}$, we have $g_{B_i}(v) = g_{B_j}(v)$.*

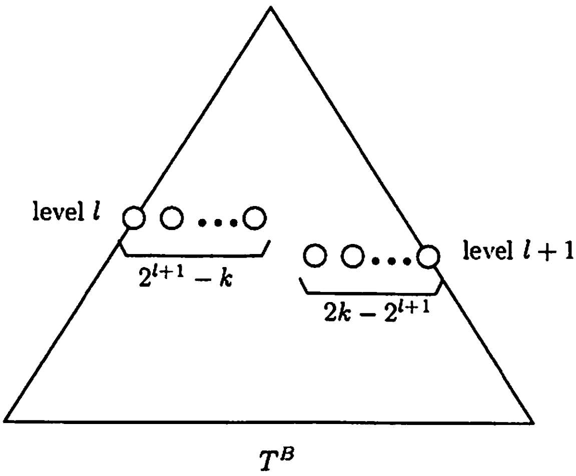

Figure 1: The set B of bookmarks from Theorem 1. B covers T and all the bookmarks are independent and placed on two consecutive levels.

Proof. The result follows because the shortest oriented path from r to v in T^{B_i} and T^{B_j} must use exactly the same edges.

■

Fact 1 states that, when replacing one bookmark set by another, $g_B(v)$ may change only for the nodes v which are in the subtrees rooted at the nodes of X. Let $E[T_{-X}]$ denote the expected number of steps to reach a node in the tree T diminished by the nodes of X and their descendants. Let $G_{-X}(B)$ denote the gain attained by the set of bookmarks B in the tree diminished by the nodes of X and their descendants. The statement of Fact 1 then says that $E[T_{-X}^{B_i}] = E[T_{-X}^{B_j}]$ and $G_{-X}(B_i) = G_{-X}(B_j)$.

Lemma 2. ***Transformation lift.*** *Suppose that for a set of bookmarks B on T there exists a node $x \notin B$ on level $l \geq 2$, such that, among the descendants of x, the only bookmark $y \in B$ is a child of x. Then for $B' = B \setminus \{y\} \cup \{x\}$, we have $G(B') > G(B)$ (cf. Figure 2).*

Proof. Let w be the number of nodes in the subtree rooted at y and s be the distance from r to x in T^B.

Note that if there is a bookmark on the oriented path from r to x then $s < l$, otherwise $s = l$. By Fact 1, $G_{-\{x\}}(B) = G_{-\{x\}}(B')$ because $g_B(v) = g_{B'}(v)$, for each node v not in the subtree rooted at x. Therefore, to evaluate the change in the gain function between the sets B and B' we

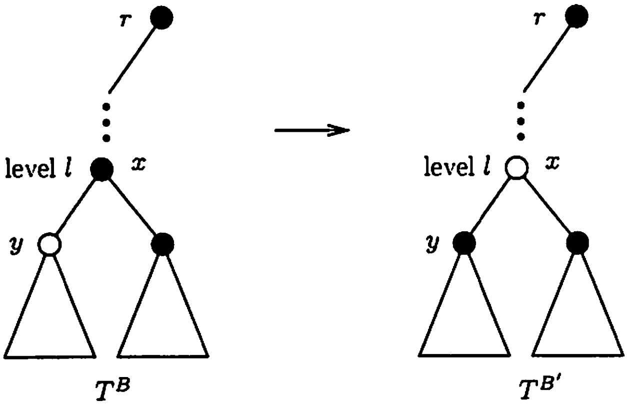

Figure 2: *Transformation lift.* If $B' = B \setminus \{y\} \cup \{x\}$ then $G(B') > G(B)$.

should consider only nodes in the subtree rooted at x. Hence, $g_B(v) = l$, for all w nodes in the subtree rooted at y and $g_B(v) = l - s$, for x and all remaining w descendants of x, while $g_{B'}(v) = l - 1$, for all $2w + 1$ nodes in the subtree rooted at x. Since $l \geq s \geq 2$ we conclude that $G(B) = G_{-\{x\}}(B) + lw + (l-s)(w+1) < G_{-\{x\}}(B) + (l-1)(2w+1) = G(B')$.

■

Lemma 3. ***Transformation adopt.*** *Suppose that T^B contains two nodes x_1 and x_2 at the same level l, such that $x_1 \in B$ and no other node in the subtrees rooted at x_1 and x_2 belongs to B. Suppose that the last bookmark on the oriented path from r to x_1 in the tree T is the node $b_1 \in B$ on level $l_1 < l$ in T. Suppose as well, that if bookmark $b_2 \in B$ on level l_2 dominates x_2, we have $l_1 > l_2$. Then for $B' = B \setminus \{x_1\} \cup \{x_2\}$ we have $G(B') > G(B)$ (cf. Figure 3).*

Proof. Let w be the number of nodes in the subtrees rooted at x_1 or x_2, cf. Figure 3. Let b_2 denote the bookmark dominating x_2 or b_2 is the predecessor of x_2 on the first level if such a bookmark does not exist, and let l_2 be the level of b_2 in G. Again, $G_{-\{x_1,x_2\}}(B) = G_{-\{x_1,x_2\}}(B')$. We have $g_B(v) = l - 1$, for each node v, descendant of x_1 and $g_B(v) = l_2 - 1$, for each v, descendant of x_2 (observe that, for this particular case, if $b_2 = r$ then we have to make $g_B(v) = 0$). Similarly, $g_{B'}(v) = l_1 - 1$, for each v, descendant of x_1 and $g_{B'}(v) = l - 1$, for each v, descendant of x_2. Hence

$G(B) = G_{-\{x_1,x_2\}}(B) + w(l-1) + w(l_2-1) < G_{-\{x_1,x_2\}}(B') + w(l_1 - 1) + w(l-1) = G(B')$.

■

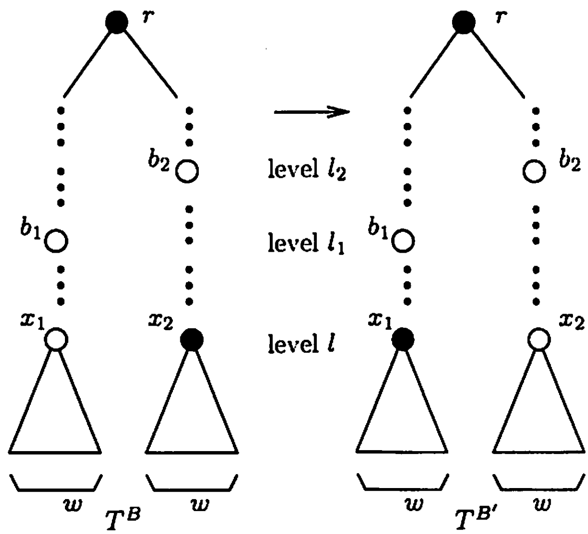

Figure 3: *Transformation adopt.* If $B' = B \setminus \{x_1\} \cup \{x_2\}$ then $G(B') > G(B)$.

Lemma 4. ***Transformation spread.*** *Suppose that for a set of bookmarks B on T there exists a node $x_1 \notin B$ on level $l \geq 2$ in T, such that the only bookmarks contained in the subtree of T rooted at x_1 are its children, y and z. Suppose as well that there exists another node $x_2 \notin B$ on level l, such that the subtree rooted at x_2 contains no bookmarks. Then, for $B' = B \setminus \{y, z\} \cup \{x_1, x_2\}$ we have $G(B') > G(B)$ (cf. Figure 4).*

Proof. Let w be the number of nodes in the subtrees rooted at y or z, cf. Figure 4.

Again, $G_{-\{x_1,x_2\}}(B) = G_{-\{x_1,x_2\}}(B')$. For each node v, descendant of x_1 we have $g_B(v) = l$, while $g_B(x_1) = l - d_B(x_1)$. For all $2w + 1$ nodes in the subtree rooted at x_2, $g_B(v) = l - d_B(x_2)$. Similarly, $g_{B'}(v) = l - 1$, for all $4w + 2$ nodes in the subtrees rooted at x_1 and x_2. Since $d_B(x_i) \geq 2$ for $i = 1, 2$, we conclude that

$$\begin{aligned} G(B) &= G_{-\{x_1,x_2\}}(B) + 2lw + (l - d_B(x_1)) + (l - d_B(x_2))(2w + 1) \\ &< G_{-\{x_1,x_2\}}(B') + (l - 1)(4w + 2) \\ &= G(B') \end{aligned} \tag{6}$$

■

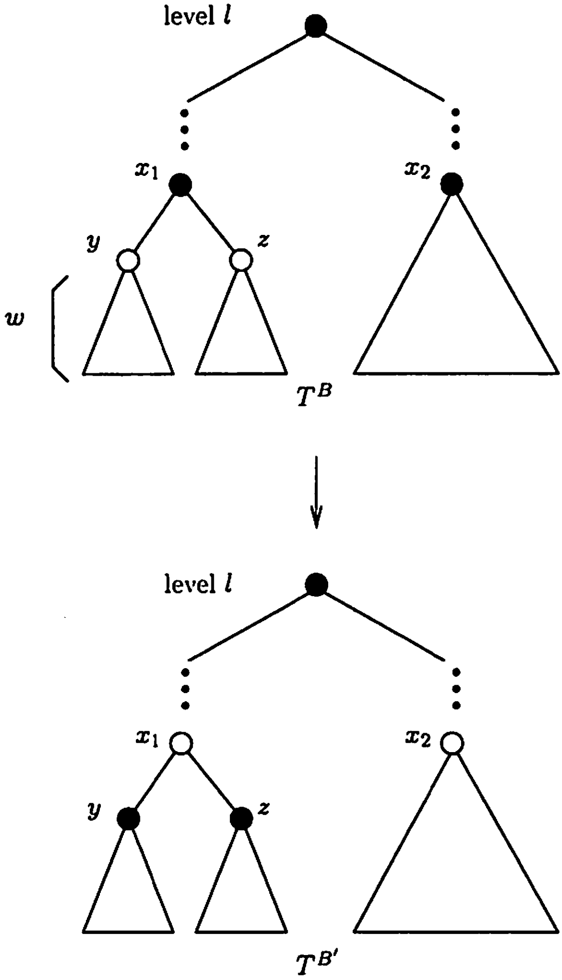

Figure 4: *Transformation spread.* If $B' = B \setminus \{y, z\} \cup \{x_1, x_2\}$ then $G(B') > G(B)$.

Lemma 5. ***Transformation floor.*** *If a set B of k bookmarks is optimal then each bookmark is placed on level $l = \lceil \log k \rceil$ of T or on a level above l.*

Proof. Let b be a bookmark at the deepest level l_b among all the bookmarks of B. We will prove that if $l_b > \lceil \log k \rceil$, then we can always replace b, getting a new bookmark set B', such that $G(B) < G(B')$. There are three possible cases:

1. Neither the parent nor the sibling of b belongs to B. We can apply *transformation lift* (Lemma 2) replacing b by b' on a level above l_b in T^B. Hence $B' = B \setminus \{b\} \cup \{b'\}$.
2. The node a which is the parent of b belongs to B. As for the level l_a of a we have $l_a \geq \lceil \log k \rceil$. Hence there are at least $p = 2^{\lceil \log k \rceil}$ nodes on level l_a. As $p \geq k$, there exists a node u on level l_a, such that the subtree of T rooted at u contains no bookmarks. Thus the conditions of *transformation adopt* (Lemma 3) are met, with $b_1 = a$, $x_1 = b$ and x_2 being a child of u. Hence we can set $B' = B \setminus \{b\} \cup \{x_2\}$.
3. The parent of b does not belong to B, but its sibling c does. Again, there must exist a node u on level $l_b - 1$ such that the subtree of T rooted at u contains no bookmarks and the conditions of *transformation spread* (Lemma 4) are met. We set $B' = B \setminus \{b, \text{sibling}(b)\} \cup \{\text{parent}(b), u\}$.

In each case, replacing b improves the gain of the bookmark set. This concludes the proof.

■

Lemma 6. ***Transformation inherit.*** *Suppose that for a set B of k bookmarks, defined on the tree T of $N \geq 15$ nodes, there exists a pair of nodes $y, x \in B$ at levels l and $l-1$ respectively, such that $2 \leq l \leq \lceil \log k \rceil$. Suppose, as well, that y is the only bookmark among the descendants of x. If $4 < k \leq \sqrt{N+1}$, then for $B' = B \setminus \{x\} \cup \{z\}$, where z is the sibling of y, we have $G(B') > G(B)$ (cf. Figure 5).*

Proof. Let w be the number of nodes in the subtree rooted at y, cf. Figure 5. Again, $G_{\{x\}}(B) = G_{-\{x\}}(B')$. For each node v, descendant of y, we have $g_B(v) = l - 1$, while for all $w + 1$ remaining nodes in the subtree rooted at x we have $g_B(v) = l - 2$. Similarly, $g_{B'}(v) = l - 1$ for all $2w$ descendants of x, and $g_{B'}(x) = l - 1 - d_B(x)$. We have to prove that

$$\begin{aligned} G(B) &= G_{-\{x\}}(B) + (l-2)(w+1) + (l-1)w \\ &< G_{-\{x\}}(B') + 2(l-1)w + (l-1-d_{B'}(x)) \\ &= G(B') \end{aligned} \tag{7}$$

which is equivalent to

$$w + 1 > d_B(x) \quad (8)$$

Note that $d_B(x) \leq l$. Observe also that, since the depth of T equals $\log(N+1) - 1$, we have $w = 2^{\log(N+1)-l} - 1$. Therefore, since $4 < k \leq \sqrt{N+1}$, which for $N \geq 15$ implies $\log k \leq \frac{\sqrt{N+1}}{2}$, we have $w + 1 \geq 2^{\log(N+1)-l} \geq 2^{\log(N+1)-\lceil \log k \rceil} \geq 2^{\log(N+1)-\log k-1} = \frac{N+1}{2k} > \frac{\sqrt{N+1}}{2} \geq l \geq d_B(x)$

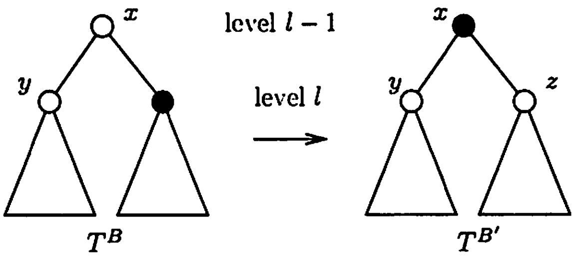

Figure 5: *Transformation inherit.* If $B' = B \setminus \{x\} \cup \{z\}$ then $G(B') > G(B)$.

■

Lemma 7. ***Transformation expose.*** *Suppose that for a set B of k bookmarks on T there exists $b \in B$ at level $t \geq 3$, such that the only bookmarks contained in the subtree of T rooted at b, are $\tilde{B} = \{b_1, \cdots, b_m\}$, and*

1. *$\{b_1, \cdots, b_p\}$, where $1 \leq p \leq m$ are placed on level l,*
2. *$\{b_{p+1}, \cdots, b_m\}$ are placed on level $l+1$,*
3. *all bookmarks of $\tilde{B}$ are independent,*
4. *each leaf of T^B is in the domain of some bookmark from level l or $l+1$.*

If $k \leq \sqrt{N+1}$, then for $B' = B \setminus \{b, b_i\} \cup \{c_i, d_i\}$, where $b_i \in \tilde{B}$ is on level l and c_i, d_i are children of b_i we have $G(B') > G(B)$ (cf. Figure 6).

Proof. As $G_{-\{b\}}(B) = G_{-\{b\}}(B')$, we need only to compute the gain function for the nodes in the subtree rooted at b.

Let n be the depth of T. Observe that there are $w = 2^{n-l} - 1$ nodes in the domain of each node $b_i, i = p+1, \ldots, m$, and for each of them $g_B(v) = l$. Similarly $g_B(v) = l - 1$, for each of $2w + 1$ descendants of every node $b_i, i = 1, \ldots, p$. Finally, for v being one of $2^{l-t+1} - p - 1$ nodes in the domain of b, we have $g_B(v) = t - 1$. In the graph $T^{B'}$, there are $p - 1$ bookmarks which are descendants of b on level l and $m - p + 2$ bookmarks on level $l+1$. Similarly as in T^B, for each of $2w + 1$ nodes in the domain of

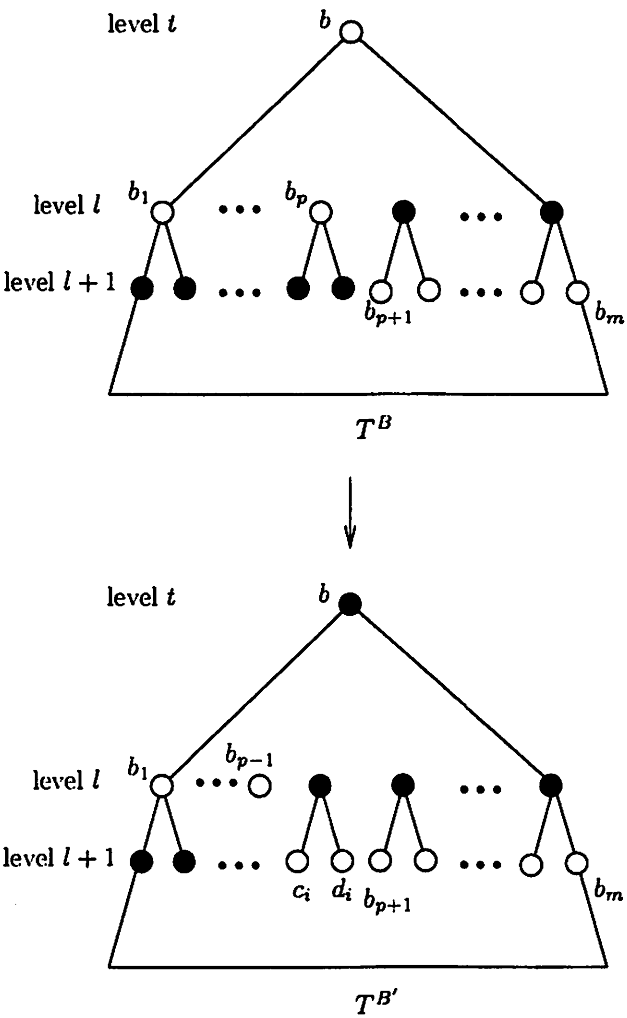

Figure 6: *Transformation expose.* "Exposing" a node from a higher level. If $B' = B \setminus \{b, b_p\} \cup \{c_i, d_i\}$ then $G(B') > G(B)$.

some bookmark on level l, we have $g_B(v) = l-1$ and for each of w nodes from the domain of a bookmark of level $l+1$, $g_B(v) = l$. The subtree of $T^{B'}$ rooted at b has $2^{l-t+1} - p$ nodes, which are not in the domain of any of $b_i, i = 1, \ldots, m$. For every such node $g_B(v) = t - d_B(x)$. Thus we have to prove that

$$\begin{aligned} G(B) &= G_{-\{b\}}(B) + (t-1)(2^{l-t+1} - p - 1) \\ &\quad + p(2w+1)(l-1) + (m-p)wl \\ &< G_{-\{b\}}(B') + (t - d_{B'}(b))(2^{l-t+1} - p) \\ &\quad + (p-1)(2w+1)(l-1) + (m-p+2)wl \\ &= G(B') \end{aligned} \tag{9}$$

or, equivalently, that $(t-1)(2^{l-t+1} - p - 1) + (2w+1)(l-1) < (t - d_{B'}(b))(2^{l-t+1} - p) + 2wl$. However, since $2 \le d_{B'}(b) \le t$ and $1 \le p \le m$, it is sufficient to prove a stronger condition substituting $d_{B'}(b) = t$ and $p = 1$. For these values, the above inequality becomes $(t-1)(2^{l-t+1} - 2) + (2w+1)(l-1) < 2wl$, which is equivalent to $(t-1)(2^{l-t+1} - 2) + l < 2w + 1$. As $t \ge 2$, it is sufficient to prove the stronger condition $(t-1)2^{l-t+1} + (l-2) < 2w+1$. The function $f(t) = (t-1)2^{l-t+1} + l - 2$ attains its maximum when $f'(t) = 2^{l-t+1} - (t-1)2^{l-t+1} \ln 2 = 0$, i.e. when $t = \frac{1}{\ln 2} + 1$. Since t is a level number and f is an integer argument unimodal function, it follows that that $f(t)$ obtains the same maximal value for $t = 2$ and $t = 3$. Hence, we can strengthen again the inequality we have to prove, getting $2^{l-1} + (l-2) < 2w+1$. Since $w = 2^{n-l} - 1$ and $N = 2^{n+1} - 1$, the condition becomes

$$2^{l-1} + (l-1) < (N+1)2^{-l} \tag{10}$$

However, since $2^{l-1} \ge l-1$, once again we can strengthen the condition obtaining $2^l < (N+1)2^{-l}$ or

$$2^{2l} < (N+1) \tag{11}$$

Observe that, since each leaf of T^B is in the domain of some bookmark from level l or $l+1$, each of 2^l nodes on level l, or one of its children, must belong to B. As $b \in B$, we identified at least $2^l + 1$ bookmarks belonging to B, proving that $k > 2^l$. Since $k \le \sqrt{N+1}$, we conclude that $2^{2l} < k^2 \le (N+1)$, thus proving Inequality 11.

■

3.1 Correctness of the Characterization

In this section we prove the correctness of the characterization given in Theorem 1.

Proof. Let B be an optimal set of bookmarks. By ***transformation floor*** (Lemma 5), we can suppose that each bookmark $b \in B$ is on or above the ***floor*** level $l = \lceil \log k \rceil$. Take a bookmark $b \in B$ on the deepest level l_b. Let c be the sibling of b and let a be its parent. Observe that at most one node among a and c can belong to B, otherwise we could improve B by means of ***transformation expose*** (Lemma 7). However, in such a case, $a \notin B$, since we could apply ***transformation inherit*** (Lemma 6), again improving the gain of B. If $c \notin B$, we could apply ***transformation lift*** (Lemma 2), once again getting a better bookmark set. So $c \in B$ and $a \notin B$. Similarly, for any other bookmark on level l_b its sibling must belong to B and its parent must not. Take any node x on level $l_b - 1$, whose children do not belong to B. $x \in B$, otherwise we could apply ***transformation spread*** (Lemma 4) using $x_1 = a$ and $x_2 = x$. So for each node x on level $l_b - 1$, either x itself or both its children belong to B. Consider the deepest level bookmark $y \in B$ above level $l_b - 1$. If y exists, y along with all the bookmarks of B which are descendants of y verify the conditions of ***transformation expose*** (Lemma 7). If y does not exist, the bookmarks of B are all independent, cover T and are placed on two consecutive levels $l_b - 1$ and l_b.

■

The following proposition shows that the property of optimal bookmark sets described in Theorem 1 does not hold for larger numbers of bookmarks.

Proposition 1. *Let T be a perfect binary tree with N nodes and let $k = 2\sqrt{N+1}+1$. There exists a set B of k bookmarks in T such that:*

1. *B covers T,*
2. *all bookmarks in B are independent,*
3. *all bookmarks in B are placed on two consecutive levels of T,*

but B is not optimal.

Proof. Consider a perfect binary tree T of depth $n = 2l - 3$ and let B be a set of $k = 2^l + 1$ bookmarks. Note that $k = 2\sqrt{N+1}+1$. According to Lemma 1 we can place 2^l bookmarks on level l and one bookmark on level $(l+1)$, so that the bookmarks are independent and cover T. To show that B is not optimal, let B' be a set of $2^l + 1$ bookmarks consisting of all nodes of level l and of one bookmark on level two, cf. Figure 7. We have

$$\begin{aligned} G(B') - G(B) &= [(l-1)(2^l)(2^{n-l+1}-1) + 1(2^{l-2}-1)] \\ &\quad -[(l-1)(2^l-1)(2^{n-l+1}-1) + 2l(2^{n-l}-1)] \qquad (12) \\ &= l+1 > 0 \end{aligned}$$

thus proving that B is not optimal.

■

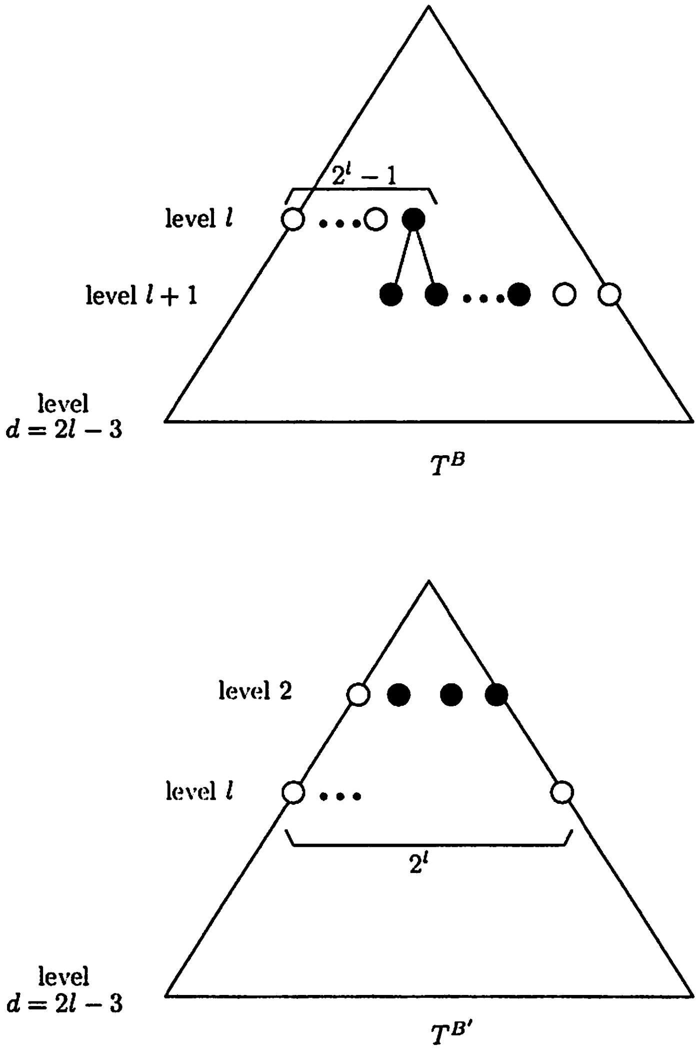

Figure 7: For Proposition 1. Suppose that B is optimal. Composing B' of all nodes at level l plus a node at level 2 will make $G(B') > G(B)$, proving that B is not optimal.

References

[1] B. Li, X. Deng, M.J. Golin, and K. Sohraby. On the optimal placement of Web proxies in the Internet: the linear topology. In *Proceedings of the Eighth IFIP Conference on High Performance Networking, HPN'98*, pages 485–495, Vienna, Austria, September 1998.

[2] B. Li, M.J. Golin, G.F. Italiano, X. Deng, and K. Sohraby. On the optimal placement of Web proxies in the Internet. In *Proceedings of the IEEE InfoCom Conference*, pages 1282–1290, March 1999.

[3] M. Vargas Martin. *Enhancing Hyperlink Structure for Improving Web Performance*. PhD thesis, School of Computer Science, Carleton University, 2002.