

An Experimental Study of Recent Hotlink Assignment Algorithms

Tobias Jacobs*

Abstract

The concept of *hotlink assignment* aims at enhancing the structure of web sites such that the user’s expected navigation effort is minimized. We concentrate on sites that are representable by trees and assume that each leaf carries a weight representing its popularity.

The problem of optimally adding at most one additional outgoing edge (“hotlink”) to each inner node has been widely studied. A considerable number of approximation algorithms have been proposed and worst-case bounds for the quality of the computed solutions have been given. However, only little is known about the practical behaviour of most of these algorithms yet.

This paper contributes to close this gap by evaluating all recent strategies experimentally. Our experiments are based on trees extracted from real websites as well as on synthetic instances. The latter are generated by a new method that simulates the growth of a web site over time. We also propose a memory-efficient way to implement an optimal hotlink assignment algorithm, making it possible to compute optimal solutions for larger instances than before. Finally, we present a new approximation algorithm that is easy to implement and exhibits an excellent behaviour in practice.

1 Introduction

The design of web sites typically aims at making a large amount of information conveniently accessible. Web designers cannot arbitrarily distribute the contents among the pages as this would make information retrieval too complex for the users. The site’s structure must rather somehow represent structural properties of the information. On the other hand, the interests of users on most web sites are highly correlated. Typically, about 80% of the users access only about 20% of the pages (cf. [14]). Moreover, the popularity of pages is likely to change over time.

By automatically moving popular pages closer to the home page one can both reduce web traffic and increase user-friendliness. However, restructuring the whole web site is not practicable for reasons mentioned above. In contrast, the concept of adding additional hyperlinks called *hotlinks* to the pages is a non-destructive approach which preserves the original site’s structure.

We concentrate on web sites that are representable by a rooted tree, where the root is the home page, inner

nodes represent navigation pages and the information is contained in the leaves, each having a certain popularity. This model has been widely studied in literature ([2, 3, 4, 5, 6, 9, 10, 11, 12, 13]) and a number of algorithms with provable good worst-case behaviour have been proposed. Our main purpose is to evaluate these algorithms experimentally and to make a statement about which of them are recommendable in practice.

Problem definition. A weighted tree T is a triple (V, E, ω) , where (V, E) is a tree rooted at a node $r \in V$. Let $L \subseteq V$ be the set of leaves. The weight function $\omega : L \rightarrow \mathbb{R}_0^+$ assigns a non-negative weight to each leaf.

For nodes $u \in V$ we also write $u \in T$. The set of ancestors (descendants) of u , not including u itself, is denoted $\text{anc}(u)$ ($\text{desc}(u)$), $\text{ch}(u)$ is the set of u ’s direct children and $\text{par}(u)$ denotes the parent of u .

A *hotlink* (u, v) is an additional directed edge between nodes in T . We say that u is the *hotparent* of v and v is the *hotchild* of u . We further say that the hotlink *starts* in u and *ends* in v . A set $A \subset V \times V$ of hotlinks is called a *hotlink assignment (HLA)*.

We assume that a user only knows about hotlinks that start in nodes she has already visited, and she always uses any hotlink taking her closer to her destination leaf. This natural behaviour is called the *greedy user model* or *obvious navigation assumption*.

Referring to [13], a hotlink assignment A is called *feasible* if it satisfies the following properties:

- (i) $v \in \text{desc}(u)$ for any $(u, v) \in A$.
- (ii) If $(u, v) \in A$, then there is no $(u', v') \in A$ with $u' \in \text{desc}(u) \cap \text{anc}(v)$ and $v' \in \text{desc}(v)$.
- (iii) For any node $u \in T$ there is at most one $(u, v) \in A$.

Properties (i) and (ii) exclude hotlinks that would never be taken by a greedy user. Property (iii) reflects the requirement that the number of hotlinks on a concise web page must be somehow limited. However, many algorithms naturally generalize to relaxations of that property. In the remainder of the paper, when talking about hotlink assignments, we always mean feasible HLAs.

*Department of Computer Science, University of Freiburg.

