On an Optimal Split Tree Problem

S.Rao Kosaraju *
Department of Computer Science
Johns Hopkins University
kosaraju@cs.jhu.edu

Teresa M. Przytycka †
John Hopkins School of Medicine
Department of Biophysics
przytyck@grserv.med.jhmi.edu

Ryan Borgstrom ‡
Department of Computer Science
Johns Hopkins University
rsean@cnds.jhu.edu

We introduce and study a problem that we refer to as the optimal split tree Problem. This is a tree construction problem motivated by applications to internet access. The problem is a generalization of the Huffman coding problem, for which a simple polynomial time optimal algorithm is known. We show that our problem is NP-complete and analyze a greedy algorithm for its solution.

More formally, consider a set $A = \{a_1, a_2, \ldots a_n\}$, with each element a_i having an associated weight $w(a_i) > 0$. A partition of A into two sets $B, A - B$ is called a *split*. A set $\mathcal{S}$ of splits such that for any pair $a, b \in A$ there exists a split $\{B, A - B\}$ in $\mathcal{S}$ such that $a \in A$ and $b \in B$ is called a *complete set of splits*.

A *split tree* for a set A with a complete set of splits $\mathcal{S}$ is a rooted tree T in which the leaves are labeled with elements of A and internal nodes correspond to splits in $\mathcal{S}$. Then the cost $c(T)$ of a split tree T is defined in the standard way: $c(T) = \sum_{i=1}^{n} l_i w(a_i)$, where l_i is the length of the path from the root to leaf a_i.

The *optimal split tree problem* is to compute for a given $(A, \mathcal{S})$ a minimum cost split tree.

We show that the greedy algorithm guarantees $O(\log n)$ approximation ratio. We construct an example for which this algorithm achieves $\Omega(\frac{\log n}{\log \log n})$ approximation ratio. This leaves a gap between proven approximation ratio and the approximation ratio that we achieved in this construction. Towards closing this gap we show that if all weights are equal and the optimal split tree is of depth $O(\log n)$, then the greedy algorithm guarantees $O(\frac{\log n}{\log \log n})$ approximation ratio. We conjecture that for the general problem the approximation ratio of the greedy algorithm is $\Theta(\frac{\log n}{\log \log n})$.

*Research supported by NSF grant CCR9508545 and ARO grant DAAH04-96-1-0013.

†Research supported by the Sloan and Department of Energy Postdoctoral Fellowship for Computational Biology.

‡Research supported by NSF grant CCR9508545