Algorithms for Computing Maximum Agreement Subtrees

Nikolaj Skipper Rasmussen 20114373

Thomas Hedegaard Lange 20113788

Master's Thesis, Computer Science

February 2016

Advisor: Christian Nørgaard Storm Pedersen

AARHUS
UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

Abstract

►in English. . . ◄

Resumé

►in Danish. . . ◄

Acknowledgements

►. . . ◄

Karsken Bælg,
Aarhus, February 11, 2016.

Contents

Chapter 1

Introduction

The Maximum Agreement Subtree problem (MAST) provides a measure of similarity, and is defined as such: Given two rooted trees, T1 and T2, created over the same leaf-set {1,2,3...,n}, determine the largest possible subset of leaves inducing an agreeing subtree of T1 and T2. For a set of leaves to induce an agreeing subtree for T1 and T2, the subtrees restricted to the set of leaves must be isomorphic, which means that they are structurally equivalent.

Let us start by motivating the interest in MAST by giving an example of its application. Suppose that we are interested in inspecting the relationship between DNA obtained from different species. This is typically done by the use of Hierachical Clustering (REF) or Neighbor Joining (REF) to construct evolutionary trees. However, finding the true evolutionary tree is often hard (find another way of expressing hard), and evidence is required to support any suggested tree topology.

The MAST problem is one of several ways of defining tree distances. -which one is supeiror?

The MAST problem applies to all trees, but we will choose to focus on the rooted, binary trees given that the motivation for the problem is primarily rooted in biology and linguistics, where these trees are most common.

►. . . ◄

Chapter 2

►. . . ◄

►example of a citation to primary literature: [1], and one to secondary literature: [?]◄

Chapter 3

►The NSquared algorithm◄

Goddard et. al.[1] describes a **►O(n2)◄** algorithm for finding the largest agreement subtree for two rooted binary trees. Given two trees T and U of size m and n, the idea is to iteratively find the largest agreement subtree and its size for every pair of subtrees from T and U. This can be done in quadratic time by using Lemma 1: ►. . . ◄

3.1 Implementation

We implemented the algorithm in java (using the forrester [**?**] library to represent the trees?). We computed the agreement subtrees for each pair of nodes in the two trees by doing a postorder traversal of the first tree and for each node did a postorder traversal of the second tree.

Algorithm 1 My algorithm

```
 1: procedure MyProcedure
 2:     stringlen ← length of string
 3:     i ← patlen
 4: top:
 5:     if i > stringlen then return false
 6:     end if
 7:     j ← patlen
 8: loop:
 9:     if string(i) = path(j) then
10:         j ← j − 1.
11:         i ← i − 1.
12:         goto loop.
13:         close;
14:     end if
15:     i ← i + max(delta_1(string(i)), delta_2(j)).
16:     goto top.
17: end procedure
```

►. . . ◄

Chapter 4

Conclusion

►. . . ◄

Bibliography

[1] Wayne Goddard, Ewa Kubicka, Grezegorz Kubicki, and F. R. McMorris. The agreement metric for labeled binary trees. *Mathematical Biosciences*, 123(2):215–226, October 1994.