Chapter 21
A Fast and Space-Economical Algorithm for the Tree Inclusion Problem

Yangjun Chen
University of Winnipeg, Canada

Yibin Chen
University of Winnipeg, Canada

ABSTRACT

The ordered tree inclusion is an interesting problem, by which the authors will check whether a pattern tree P can be included in a target tree T, where the order of siblings in both P and T is significant. In this chapter, the authors propose an efficient algorithm for this problem. Its time complexity is bounded by O(|T|·loghP) with O(|T| + |P|) space being used, where hP represents the height of P. Up to now the best algorithm for this problem needs Θ(|T|·|leaves(P)|) time, where leaves(P) stands for the set of the leaves of P.

INTRODUCTION

Let T be a rooted tree. We say that T is *ordered* and *labeled* if each node is assigned a symbol from an alphabet Σ and a left-to-right order among siblings in T is specified. Let v be a node different of the root in T with parent node u. Denote by *delete*(T, v) the tree obtained by removing the node v from T, by which the children of v become part of the children of u as illustrated in Figure 1.

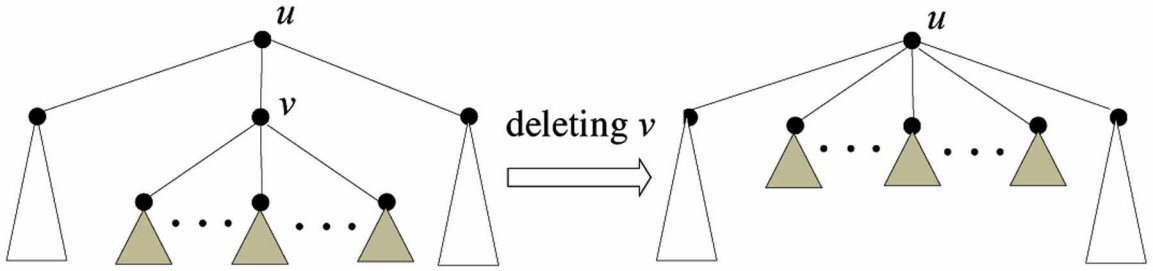
Given two ordered labeled trees P and T, called the pattern and the target, respectively. We may ask: Can we obtain pattern P by deleting some nodes from target T? That is, is there a sequence v_1, ...,v_k of nodes such that for

$T_0 = T$ and

$T_{i+1} = delete(T_i, v_{i+1})$ for $i = 0, ..., k - 1$,

DOI: 10.4018/978-1-5225-7659-4.ch021

Figure 1. Illustration of node deletion



we have $T_k = P$? If this is the case, we say, P is included in T (Kilpeläinen and Mannila, 1995). Such a problem is called the *tree inclusion problem*. It has many applications in the computer engineering as described below.

BACKGROUND

The first interesting application of the tree inclusion is used as an important query primitive for XML data (Mannila and Räiha, 1990), where a structured document database is considered as a collection of parse trees that represent the structure of the stored texts and the tree inclusion is used as a means of retrieving information from them. As an example, consider the tree shown in Figure 2, representing an XML document for the book *Arts of Programming* authored by (Knuth, 1969). One might want to find this book in an XML database by forming a pattern tree as shown in Figure 3 as a query, which can be obtained by deleting some nodes from the tree shown in Figure 2. Thus, a tree inclusion checking needs to be conducted to evaluate this query.

Figure 2. A XML document (target) tree