Journal of Computer and System Sciences 82 (2016) 229–259

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Structural characterizations of the navigational expressiveness of relation algebras on a tree ☆

George H.L. Fletcher [a,*], Marc Gyssens [b], Jan Paredaens [c], Dirk Van Gucht [d], Yuqing Wu [e]

[a] Eindhoven University of Technology, Eindhoven, The Netherlands
[b] Hasselt University, Hasselt, Belgium
[c] University of Antwerp, Antwerp, Belgium
[d] Indiana University, Bloomington, IN, USA
[e] Pomona College, Claremont, CA, USA

ARTICLE INFO

Article history:
Received 3 June 2013
Accepted 12 July 2015
Available online 10 November 2015

Keywords:
Trees
Relation algebra
XML
XPath
Bisimulation
Instance expressivity

ABSTRACT

We study the expressiveness on a given document of various fragments of XPath, the core navigational language on XML documents. Viewing these languages as fragments of Tarski's relation algebra, we give characterizations for when a binary relation on the document's nodes (i.e., a set of paths) is definable by an expression in these algebras. In contrast with this "global view" on language semantics, there is also a "local view" where one is interested in the nodes to which one can navigate starting from a particular node. In this view, we characterize when a set of nodes can be defined as the result of applying an expression to a given node. All of these global and local definability results are obtained using a two-step methodology, which consists of first characterizing when two nodes cannot be distinguished by an expression in the language, and then bootstrapping these characterizations to the desired results.

1. Introduction

In this paper, we investigate the expressive power of several basic fragments of Tarski's relation algebra [2] on finite tree-structured graphs. Tarski's algebra is a fundamental tool in the field of algebraic logic which finds various applications in computer science [3–6]. Our investigation is specifically motivated by the role the relation algebra plays in the study of database query languages [7–15], and, more in particular, these query languages restricted to tree-structured graphs [16]. The algebras we consider in this paper correspond to natural fragments of XPath. XPath is a simple language for navigation in XML documents (i.e., a standard syntax for representing node-labeled trees), which is at the heart of standard XML transformation languages and other XML technologies [17]. Keeping in the spirit of XML, we will continue to speak in what follows of trees as "documents" and the algebras we study as "XPath" algebras.

XPath can be viewed as a query language in which an expression associates to every document a binary relation on its nodes representing all navigation paths in the document defined by that expression [9,18,19]. From this query-level

☆ A preliminary version of some of the results given here were presented at the 25th ACM Symposium on Principles of Database Systems [1].

* Corresponding author.
E-mail addresses: g.h.l.fletcher@tue.nl (G.H.L. Fletcher), marc.gyssens@uhasselt.be (M. Gyssens), jan.paredaens@ua.ac.be (J. Paredaens), vgucht@cs.indiana.edu (D. Van Gucht), melanie.wu@pomona.edu (Y. Wu).

http://dx.doi.org/10.1016/j.jcss.2015.10.002

perspective, several natural semantic issues have been investigated in recent years for various fragments of XPath. These include expressibility, closure properties, and complexity of evaluation [9,10,19–21], as well as decision problems such as satisfiability, containment, and equivalence [22–24].

Alternatively, we can view XPath as a navigational tool on a particular given document, and study expressiveness issues from this document-level perspective. (A similar duality exists in the relational database model, where Bancilhon [25] and Paredaens [26] considered and characterized expressiveness at the instance level, which, subsequently, Chandra and Harel [27] contrasted with expressiveness at the query level.)

In this setting, our goal is to characterize, for various natural fragments of XPath, when a binary relation on the nodes of a given document (i.e., a set of navigation paths) is definable by an expression in the fragment.

To achieve this goal, we develop a robust two-step methodology. The first step consists of characterizing when two nodes in a document cannot be distinguished by an expression in the fragment under consideration. It turns out for those fragments we consider that this notion of expression equivalence of nodes is equivalent to an appropriate generalization of the classic notion of bisimilarity [28]. The second step of our methodology then consists of bootstrapping this result to a characterization for when a binary relation on the nodes of a given document is definable by an expression in the fragment.

We refer to this perspective on the semantics of XPath at the document level as the "global view." In contrast with this global view, there is also a "local view" which we consider. In this view, one is only interested in the nodes to which one can navigate starting from a particular given node in the document under consideration. From this perspective, a set of nodes of that document can be seen as the end points of a set of paths starting at the given node. For each of the XPath fragments considered, we characterize when such a set represents the set of *all* paths starting at the given node defined by some expression in the fragment. These characterizations are derived from the corresponding characterizations in the "global view," and turn out to be particularly elegant in the important special case where the starting node is the root.

In this paper, we study several natural XPath fragments. The most expressive among them is the *XPath algebra* which permits the self, parent, and child operators, predicates, compositions, and the boolean operators union, intersection, and difference. (Since we work at the document level, i.e., the document is given, there is no need to include the ancestor and descendant operators as primitives.) We also consider the *core XPath algebra*, which is the XPath algebra without intersection and difference at the expression level. The core XPath algebra is the adaptation to our setting of Core XPath of Gottlob et al. [18,21,29]. Of both of these algebras, we also consider various "downward" and "upward" fragments without the parent and child operator, respectively. We also study "positive" variants of all the fragments considered, without the difference operator.

Our strategy is to introduce and characterize generalizations of each of these practical fragments, towards a broader perspective on relation algebras on trees. These generalizations are based on a simple notion of path counting, a feature which also appears in XPath.

The robustness of the characterizations provided in this paper is further strengthened by their feasibility. As discussed in Section 9, the global and local definability problems for each of the XPath fragments are decidable in polynomial time. This feasibility hints towards efficient partitioning and reduction techniques on both the set of nodes and the set of paths in a document. Such techniques may be applied fruitfully towards, e.g., document compression [30], access control [31], and designing indexes for efficient query processing [11,32,33].

We proceed in the paper as follows. In Section 2, we formally define documents and the algebras, and then in Section 3, we define a notion of "signatures" which will be essential in the sequel. In Section 4, we define the semantic and syntactic notions of node distinguishability necessary to obtain our desired structural characterizations. In the balance of the paper, we apply our two-step methodology to link semantic expression equivalence in the languages to appropriate structural syntactic equivalence notions. In particular, we give structural characterizations, under both the global and local views,

- of "strictly" (Section 5) and "weakly" (Section 6) downward languages, and their positive variants;
- of upward languages and their positive variants (Section 7); and
- of languages with both downward and upward navigation, and their positive variants (Section 8).

Along the way, we also establish the equivalence of some of these fragments, using the structural characterizations obtained. We conclude in Section 9 with a discussion of some ramifications of our results and directions for further study.

2. Documents and navigation

In this paper, we are interested in navigating over documents in the form of unordered labeled trees. Formally, we denote such a document as $D = (V, Ed, r, \lambda)$, with D the document name, V the set of nodes of the tree, Ed the set of edges of the tree, r the root of the tree, and $\lambda : V \to \mathcal{L}$ a function assigning to each node a label from some infinite set of labels $\mathcal{L}$.

Example 2.1. Fig. 1 shows an example of a document that will be used throughout the paper. Here, $r = v_1$ is the root of the tree with label $\lambda(v_1) = a$.