Journal of Logical and Algebraic Methods in Programming 83 (2014) 212–224

Contents lists available at ScienceDirect

Journal of Logical and Algebraic Methods in Programming

www.elsevier.com/locate/jlamp

Hopscotch—reaching the target hop by hop

Peter Höfner [a,c,*], Annabelle McIver [b]

[a] NICTA, Australia
[b] Department of Computing, Macquarie University, Australia
[c] Computer Science and Engineering, University of New South Wales, Australia

ARTICLE INFO

Article history:
Available online 12 February 2014

Keywords:
Routing
Semiring
Kleene algebra
Path algebra

ABSTRACT

Concrete and abstract relation algebras have widespread applications in computer science. One of the most famous is graph theory. Classical relations, however, only reason about connectivity, not about the length of a path between nodes. Generalisations of relation algebra, such as the min-plus algebra, use numbers allowing the treatment of weighted graphs. In this way one can for example determine the length of shortest paths (e.g. Dijkstra's algorithm). In this paper we treat matrices that carry even more information, such as the "next hop" on a path towards a destination. In this way we can use algebra not only for determining the length of paths, but also the concrete path. We show how paths can be reconstructed from these matrices, hop by hop. We further sketch an application for message passing in wireless networks.

It is our pleasure to dedicate this paper to Gunther Schmidt at the occasion of his 75th birthday. Gunther has been working in the area of relation algebra for many years, publishing countless papers and two books. His first book particularly addresses the relationship between relations and graphs. Relations are an excellent tool for analysing unweighted graphs; subsequently weighted graphs were "algebraised" using matrices over reals or naturals, such as the min-plus algebra, a long-term interest of Gunther. In this paper we continue this line of research and show how to treat matrices that contain even more information, not only numbers. In this respect we illustrate, as Gunther did on so many occasions, that matrices are a powerful, concise and easy-to-understand tool in computer science. So, all the best Gunther, and many more years of health and success!

1. Introduction

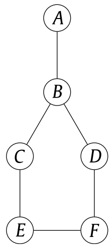
Concrete and abstract relation algebras have widespread applications in computer science. Using concrete relation algebra, it is easy and concise to characterise and calculate with concepts such as orderings, equivalence relations, reflexive closures, Church–Rosser theorems, etc. [1–5]. Concrete relation algebra can be represented either by sets of relations (pairs) or by matrices over the Boolean algebra; it is well known that both representations are isomorphic.

Abstract relation algebra generalises matrices or sets and uses an axiomatic approach. This abstraction allows inherently algebraic reasoning (often equational), which can be automated [6].

One of the most famous applications of relation algebra is graph theory (e.g. [4]). Characterising (unweighted) graphs by their adjacency matrices allows calculations on graphs to be performed algebraically. For example, the reflexive transitive closure can be characterised by the least fixpoint of $x = a \cdot x + 1$, and hence easily calculated by the Knaster–Tarski fixpoint

* Corresponding author.
E-mail addresses: peter.hoefner@nicta.com.au (P. Höfner), annabelle.mciver@mq.edu.au (A. McIver).

http://dx.doi.org/10.1016/j.jlap.2014.02.009



(a) 6-nodes graph

$$\begin{array}{c|cccccc} & A & B & C & D & E & F \\ \hline A & 0 & 1 & 0 & 0 & 0 & 0 \\ B & 1 & 0 & 1 & 1 & 0 & 0 \\ C & 0 & 1 & 0 & 0 & 1 & 0 \\ D & 0 & 1 & 0 & 0 & 0 & 1 \\ E & 0 & 0 & 1 & 0 & 0 & 1 \\ F & 0 & 0 & 0 & 1 & 1 & 0 \end{array}$$

(b) Corresponding Relation

$$\begin{array}{c|cccccc} & A & B & C & D & E & F \\ \hline A & \infty & 1 & \infty & \infty & \infty & \infty \\ B & 1 & \infty & 1 & 1 & \infty & \infty \\ C & \infty & 1 & \infty & \infty & 1 & \infty \\ D & \infty & 1 & \infty & \infty & \infty & 1 \\ E & \infty & \infty & 1 & \infty & \infty & 1 \\ F & \infty & \infty & \infty & 1 & 1 & \infty \end{array}$$

(c) Corresponding Matrix (in min-plus algebra)

$$\begin{array}{c|cccccc} & A & B & C & D & E & F \\ \hline A & 0 & 1 & 2 & 2 & 3 & 3 \\ B & 1 & 0 & 1 & 1 & 2 & 2 \\ C & 2 & 1 & 0 & 2 & 1 & 2 \\ D & 2 & 1 & 2 & 0 & 2 & 1 \\ E & 3 & 2 & 1 & 2 & 0 & 1 \\ F & 3 & 2 & 2 & 1 & 1 & 0 \end{array}$$

(d) Reflexive transitive closure (in min-plus algebra)

Fig. 1. Graph, relation and reflexive transitive closure (min-plus algebra).

theorem. However, the reflexive transitive closure of a relation only determines whether two nodes are connected via a path or disconnected. Relation algebra cannot be used to determine the length of a path between nodes, nor to handle weighted graphs.

To overcome this deficiency (abstract) relation algebra was generalised and the developed techniques were extended to semirings, Kleene algebras and similar structures (e.g. [7–9]). For weighted graphs the max-plus algebra [10], the min-plus algebra[1] (e.g. [11]), the min-max algebra and fuzzy relations [12] turned out to be useful. Using these semiring-like structures it is possible to formally derive algorithms, such as Dijkstra's shortest-path algorithm or the algorithm of Floyd and Warshall [13]. Other applications for this type of algebra include refinement [14], knowledge representation [15], social choice theory [16], preference modelling [17,18] and program analysis [19,20].

This paper aims to formalise concepts of (wireless) networks algebraically. Based on previous work [21], we show that path finding in a matrix-presentation of a network is not an easy task and that some new theory is needed. As a result, this paper presents new definitions of "hops" and "paths" in an algebraic setting and discusses fundamental properties. To guarantee applicability, we relate our formalism to routing protocols for wireless mesh networks. In this paper we focus on characterising the basic concepts, not on algebraic reasoning.

The paper is organised as follows. In Section 2 we explain the problem of describing paths algebraically in the context of routing protocols for wireless mesh networks. Section 3 recalls the basic algebraic constructions we need, and in Section 4.1 we set out an algebraic development for paths as sequences of nodes. Finally, in Section 5, we illustrate the ideas on a small example taken from routing.

2. A hidden path

Let us first present the problem at hand. For this purpose, we assume that the reader is familiar with relations and relation algebra.[2] The remaining necessary algebraic foundations are formally introduced in Section 3.

2.1. The "classical" encoding

It is well known that matrices over Boolean algebras can be used to describe (unweighted) graphs by their adjacency matrices. This (essentially) yields relation algebra. An example is given in Fig. 1(b), which encodes the graph presented in Fig. 1(a). It is also known that the reflexive transitive closure operation (Kleene star) determines the connectivity between nodes. In the example there is a path between all nodes, i.e., all nodes are connected with each other. Hence the reflexive transitive closure is the all-relation $\top\!\!\top$ ($\top\!\!\top_{ij} = 1$ for all i, j). However, this result neither indicates the distances between nodes nor the actual paths.

The min-plus semiring can be used to determine path lengths. This algebra uses $\mathbb{R}_{\geqslant 0} \cup \{\infty\}$ as carrier set, defines addition as min, multiplication as $+$, and Kleene star as $a^* = 0$. As usual, these operations can be lifted to matrices. Fig. 1(c) shows the characterisation of the graph of Fig. 1(a), using the min-plus algebra; Fig. 1(d) the reflexive, transitive closure—it

1 This algebra is sometimes called tropical semiring.

2 The definitions of abstract and concrete relation algebras are given in Appendix A.