RAIRO
INFORMATIQUE THÉORIQUE

BERND MAHR

Algebraic complexity of path problems

RAIRO – Informatique théorique, tome 16, nº 3 (1982), p. 263-292.
<http://www.numdam.org/item?id=ITA_1982__16_3_263_0>

R.A.I.R.O. Informatique théorique/Theoretical Informatics
(vol. 16, n° 3, 1982, p. 263 à 292)

ALGEBRAIC COMPLEXITY OF PATH PROBLEMS (*)

by Bernd MAHR (1)

Communicated by M. NIVAT

Abstract. — *Algebraic complexity of path problems is studied using straight line programs over arbitrary semirings as a computational model. In contrast to n^3 lower bounds in the literature, graph sensitive bounds are proven which apply for any input graph and any semiring. It is shown that these bounds are tight for all cycle-free graphs and some cyclic graphs.*

Résumé. — *On étudie la complexité algébrique des problèmes de cheminement avec des programmes linéaires sur des semi-anneaux quelconques en guise de modèle de calcul. On donne des bornes dépendant du graphe, qui s'appliquent à n'importe quel graphe d'entrée et n'importe quel semi-anneau, par opposition aux bornes en n^3 que l'on trouve dans la littérature. Ces bornes sont stricte pour tous les graphes sans cycles.*

1. INTRODUCTION

Finding shortest paths in a graph with labelled edges is a most famous and important problem in combinatorial optimization. It has been studied now since at least twentyfive years and much progress has been achieved in efficient solution of the problem; so it is likely to be true that "for this problem we have almost reached the theoretical bounds of speed if conventional serial computers are to be used" (*see* introduction of [DP 80]). But, we do not know very well about these theoretical bounds of this problem and there are still many open questions about its complexity.

This paper studies algebraic complexity of path problems and proves lower bounds on the number of operations to be performed by straight line programs which solve path problems. These bounds are not restricted to the so called "all pair shortest path problem", but apply to path problems over arbitrary semirings, and in contrast to lower bounds known in the literature, they reflect

(*) Received in October 1980.
(1) Technische Universität Berlin Fachbereich Informatik, Institut für Software und Theoretische Informatik, Otto-Suhr-Allee 18/20, 1000 Berlin 10, Germany.

the graph structure and not just the number of nodes, (Theorems 5.3, 5.4, 5.7; Theorems 5.9, 5.10, 5.11). A concise survey on path problems, their solution techniques, their algebraic generalization and their complexity is given in [Ma 80]. Recently Deo and Pang [DP 80] have presented an excellent bibliography on path problems which includes more than 200 selected references and about 80 annotations. We therefore abstain here from giving an overview on the literature and results and just mention the main lines which motivate the subsequent sections.

For the shortest path problem efficient algorithms have been developed which mainly fall into two categories: Those which depend on the labelling of the graph, called "label dependent algorithms", and those which are independent of the label of the graph and only produce computations depending on the graph or some of its parameters, like number of nodes or number of edges. This class of algorithms we call "label independent". The most famous algorithms in these classes are those of Dijkstra and Floyd respectively. They solve the shortest path problem (all pair version) in $O(n^3)$ steps. In the label dependent case Dijkstras algorithm for the shortest path problem was improved in several respects (*see* [Ma 80]) namely by Johnson [Jo 77], Wagner [Wa 76] and Bloniarz [Bl 80]. The only known lower bound which addresses this class of algorithms is due to Spira and Pan [SP 73] and proves $n(n-1)$ "steps" for an algorithm which solves the single source shortest path problem. This bound however is in no clear relation to the best algorithms for the all pair problem in the class of label dependent algorithms.

The class of label independent algorithms instead seems easier to model for complexity analysis. All lower bounds which address this class of algorithms use straight line programs and analyse the number of algebraic operations. Murchland [Mu 68], Iri and Nakamori [IN 72] and Johnson [Jo 73] prove that $n(n-1)(n-2)$ additions and min-operations are necessary to solve the shortest path problem. In contrast to this bound Fredman [Fr 76] gives an algorithm which only uses $O(n^{5/2})$ additions and min-operations and Romani [Ro 80] shows in a recent paper that the all pair shortest path problem has a complexity not greater than matrix multiplication over a ring and thus is in $O(n^{2.52})$. But Fredmans and Romanis algorithms do not fall in our classification of label dependent and label independent algorithms (*see* also [Ma 80]).

From the use of straight line programs it is natural to expand complexity analysis for the shortest path problem to generalized path problems over arbitrary semirings. This is one direction of generalization which is emphasized in this paper. The other direction of generalization is to study graphs instead of matrices. The main effort of these generalizations is that on one hand other problems than the shortest path problem are covered in the analysis and on the other hand that "graph sensitive" algorithms are under consideration and so also decomposition algorithms (*see* Brucker [Br 74], Tarjan [Ta 75]) and algorithm for cycle-free graphs are included in the analysis.

In Section 2 of this paper we introduce basic concepts from graph theory, semirings and the notions of labelled graph and adjacency matrix.

Section 3 defines the computational model of "computation scheme" which is a useful representation of straight line programs. In order to analyse complexity of path problems, one has to specify what kind of function the algorithms in the computational model have to compute. An appropriate notion is that of a path function, which is defined in 3.1. Subsequent to this definition we discuss how path functions are related with solutions of path problems. In order to prove the claimed lower bounds, the computation schemes for path functions are restricted to so called "adapted schemes" which in a natural way perform operations controlled by the input graph. That such a restriction is not just a technical need but also a meaningful concept, is subject of a discussion at the end of this section.

Section 4 contains the proofs of two lower bounds on the number of additions and multiplications to be performed by adapted schemes which compute path functions.

Section 5 discusses the bounds of Section 4 and states the main results of this paper (Theorems 5.3 and 5.4). It follows a discussion on the necessity of the assumptions which are made in the statement of Theorems 5.3 and 5.4. Finally in this section our lower bound results are related with those known in the literature, namely the bounds of Murchland, Iri and Nakamori, and Johnson, and of Pratt, Paterson and Mehlhorn. The latter bound for path problems over the Boolean semiring is then extended to path problems over arbitrary positive semirings.

The main results of this paper are already in my thesis [Ma 79]. In many respects, however, this paper is a further development of the complexity analysis of path problems in [Ma 79]; first of all since in 5.6 an answer to the central open question in [Ma 79] is given.

ACKNOWLEDGEMENT

I thank C. Lautemann, H. Ehrig and D. Siefkes for their comments and helpful discussions, and H. Barnewitz for her excellent typing of the manuscript.

2. BASIC CONCEPTS

In this section we introduce the basic concepts of graph theory and semirings which are fundamental for the next sections.

Let $[0]:=\emptyset$ and $[n]:=\{1, \ldots, n\}$ for non-negative integers n. Then we refer to a *graph* as a subset $E \subseteq [n] \times [n]$ for convenience. By $n(E)=n$ and $e(E)=e$ we denote the *number* of *nodes* and *edges* respectively. Concepts from graph theory we adopt from Berge [Be 76] if not explicitly defined.

By $T(E) \subseteq [n] \times [n]$ we denote the *reflexive-transitive closure* of E, which is a graph having an arc $(i, j) \in T(E)$ if and only if E has a path p with initial endpoint i and terminal endpoint j. For $(i, j) \in [n] \times [n]$ we denote by P_{ij} the *set of* (i, j)*-paths*, i. e. the set of paths with initial endpoint i and terminal endpoint j. For $i \in [n]$ we assume $\lambda \in P_{ii}$, where λ denotes the *empty path*. By F_{ij} we denote the *set of elementary* (i, j)*-paths*, i. e. of (i, j)-paths, where no node is encountered more than once. By $P(E)$ we denote the *set of paths* of E and look at $P(E)$ as a subset of E^*, the free monoid generated from E with unit λ and concatenation as the binary operation. Namely for $i \in [n]$ we assume $F_{ii}=\{\lambda\}$.

Given $(i, j) \in T(E)$ then a set of nodes $A_{ij} \subseteq [n]$ is called an (i, j)*-cut*, iff every elementary (i, j)-path has an internal node which is in A_{ij}. A family $\mathscr{A}(E)=(A_{ij})_{(i, j) \in T(E)}$ of (i, j)-cuts we call a *cut system of* E. An (i, j)-cut A_{ij} is called *minimal* if all other (i, j)-cuts contain not less nodes than A_{ij} and a cut system $\mathscr{A}(E)$ is called *minimal* if all its cuts are minimal. Note, a minimal (i, i)-cut is empty for all $i \in [n]$.

A *semiring* $S=(S, +, ., 0, 1)$ is a set together with binary operations $+$ and and two selected elements (i. e. 0-ary operations) 0 and 1 (assumed to be not equal) such that the following equations hold:

$$a+b=b+a, \tag{1}$$

$$(a+b)+c=a+(b+c), \tag{2}$$

$$a+0=a, \tag{3}$$

$$(a.b).c=a.(b.c), \tag{4}$$

$$a.1=1.a=a, \tag{5}$$

$$a.0=0.a=0, \tag{6}$$

$$a.(b+c)=a.b+a.c, \tag{7}$$

$$(a+b).c=a.c+b.c. \tag{8}$$

We call a semiring *idempotent*, if in addition to (1)-(8):

$$a+a=a. \tag{9}$$

Idempotent semirings are partially ordered by:

$$a \leq b \quad \Leftrightarrow \quad a+b=b, \tag{10}$$

and called "path algebras" in Carré [Ca 79].

Semirings are called *simple*, if in addition to (1)-(8):

$$1+a=1. \tag{11}$$

We also use generalized addition $\sum_{i\in I} a_i$ of finite indexed families $\{a_i \mid i \in I\}$ and refer to Eilenberg [Ei 74] for a formal characterization of semirings by generalized addition. Namely if $I=\emptyset$, we define $\sum_{i\in I} a_i := 0$. Eilenberg also defines *positive semirings*, i. e. semirings with:

$$\left.\begin{array}{llll} a+b=0 & \text{implies} & a=0 \text{ and } & b=0, \\ a.b=0 & \text{implies} & a=0 \text{ or } & b=0. \end{array}\right\} \tag{12}$$

Positive semirings are characterized as follows:

S positive if and only if the mapping $h: S \to \mathbb{B}$ into the Boolean semiring $B=\{0, 1\}$, defined by:

$$h(a)=\begin{cases} 0 & \text{if } a=0, \\ 1 & \text{otherwise}, \end{cases} \tag{13}$$

is a semiring homomorphism.

It is easily seen, that the class of semirings can be embedded into the class of positive semirings by adjoining to each semiring a new zero-element $\mathit{0\!\!\!-}$. This *positive extension* preserves properties (9) and (11), *see* [Ma 82]. Finally we closure over semirings and are studied in [Ma 82].

$$x=1+a.x, \tag{14}$$

is solvable. Such semirings play an important role in the study of transitive closure over semirings and are studied in [Ma 82].

From the large list of examples for semirings we only mention:

$$\mathbb{N}_0=(\mathbb{N}_0, +, ., 0, 1), \tag{15}$$

the *natural numbers* with addition and multiplication. This semiring is initial in the variety of semirings.

$$\mathbb{B}=(\{0, 1\}, \vee, \wedge, 0, 1), \tag{16}$$

the *Boolean semiring* with *and* and *or*.

$$\mathbb{M}=(\mathbb{R}^+ \cup \{\infty\}, \min, +, \infty, 0), \tag{17}$$

the *positive real numbers* with minimum and addition and zero as an adjoint "infinity"-element.

$$\mathbb{E}=(2^{E^*}, \cup, \circ, \emptyset, \{\lambda\}), \tag{18}$$

the *language semiring* over E with union and complex product and the empty string as one.

All these semirings are positive. (16), (17) and (18) are idempotent and (16) and (17) are simple. (16), (17) and (18) are closed.

Let $E \subseteqq [n] \times [n]$ be a graph and S be a semiring, then a mapping $m : E \to S$ we call a *labelling of* E. Since $P(E) \subseteqq E^*$, we have the unique homomorphic extension $\bar{m} : E^* \to S$ from the free monoid E^* into the semiring S, taken as the monoid $(S, ., 1)$, to define a *path labelling of* E:

$$\bar{m} : P(E) \to S.$$

Explicitly $\bar{m}(p) = m(e_1). \dots .m(e_r)$ for each path $p = e_1 \dots e_r$, and $\bar{m}(\lambda) = 1$. If $m : E \to S$ is a labelled graph, then its *adjacency matrix* is defined by $M : [n] \times [n] \to \bar{S}$ with:

$$M(i, j) = \begin{cases} m(i, j) & \text{if } (i, j) \in E, \\ \mathbf{0} & \text{otherwise.} \end{cases}$$

Here $\bar{S}$ denotes the positive extension of S. The $n \times n$-matrices over some semiring S build themselves a *semiring* $M_n(S)$ with addition and multiplication induced from S, the zero matrix as null and unit matrix as one.

3. PATH FUNCTIONS AND ADAPTED COMPUTATION SCHEMES

In this section we define path functions and show how they are related with solutions of path problems. As an algorithmic model we introduce adapted computation schemes. For those schemes the lower bounds in the next section are valid.

3.1. Definition (Path Function): Let $E \subseteqq [n] \times [n]$ be a graph and $T(E)$ its transitive closure. Let S be a semiring, then a function:

$$T : \quad S^E \to S^{T(E)},$$

such that for $m \in S^E$ and $(i, j) \in T(E)$:

$$T(m)(i, j) = \sum_{p \in B_{ij}} \bar{m}(p) \tag{1}$$

with finite B_{ij} satisfying $F_{ij} \subseteqq B_{ij} \subseteqq P_{ij}$, is called a *path function*.

3.2. REMARK (Uniqueness of Path Functions): If E is cycle-free or S is simple, then path functions are *unique*, i.e. they are uniquely specified by giving a graph E and a semiring S:

If E is cycle-free, then all paths are elementary so that for all B_{ij} with $F_{ij} \subseteqq B_{ij} \subseteqq P_{ij}$ already $F_{ij} = B_{ij}$ holds.

If S is simple, then for all graphs E and all labellings $m : E \to S$ we have: Let p be a cyclic path and $p = p_1 c p_2$ be a decomposition of p such that c is a cycle, then $m(p_1 c p_2) + m(p_1 p_2) = m(p_1 p_2)$. This observation can be used to show that for all B_{ij} with $F_{ij} \subseteqq B_{ij} \subseteqq P_{ij}$ the equation:

$$\sum_{p \in B_{ij}} \bar{m}(p) = \sum_{p \in F_{ij}} \bar{m}(p), \qquad (\star)$$

holds. If S is idempotent this needs not to be true for all labellings $m : E \to S$. Then $(\star)$ holds exactly in those cases, where $m : E \to S$ is *absorbtive* (*see* [Ca 79]) which assumes that all cycles c in E satisfy $\bar{m}(c) \leqq 1$.

Path functions are mainly motivated from a study of path problems. They serve to describe solutions of path problems and to discuss their arithmetic complexity. Path problems are differently defined in the literature. One version is to find the values $\mu(i, j) = \sum_{p \in P_{ij}} \bar{m}(p)$ for all $(i, j) \in [n] \times [n]$ (*see* e. g. [Me 77]). Here infinite sums in a semiring are used which cause some notational problems if not in advance their existence is assumed for all countable indexed families. This assumption, however, is quite restrictive (for a discussion of this problem *see* [Ma 82]).

Another version of defining path problems is to find a solution of the equation $X = 1 + AX$ in the semiring of matrices $M_n(S)$ where A is adjacency matrix of a labelled graph $m : E \to S$, (*see* e. g. [Ca 79]). This version of a path problem is most general and avoids to talk about existence of infinite sums. It is an elegant approach to discuss solvability of path problems over general semirings, *see* [Ca 79], [Zi 81] and [Ma 82]. Most useful here is the notion of a stable matrix:

Let S be a semiring, then a matrix $A \in M_n(S)$ is called *stable of index r*, if:

$$\sum_{j=0}^{r} A^j = \sum_{j=0}^{r+1} A^j,$$

where $A^0 := 1$, the unit matrix, and $A^j := A^{j-1} . A$ is assumed. Let us abbreviate the matrix $\sum_{j=0}^{r} A^j$ by $A^{\langle r \rangle}$, then one immediately derives:

3.3. FACT (Stable Matrices): If A is stable of index r, then $A^{\langle r \rangle}$ solves the equation $X=1+AX$.

The following lemma gives a graph theoretic interpretation of the matrices $A^{\langle r \rangle}$ and so relates solutions of equations $X=1+AX$ with the intention expressed by a path problem:

3.4. LEMMA (Graph Theoretic Interpretation): *Let* $m: E \to S$ *be a labelled graph with adjacency matrix* $A=A(m)$. *Then for* $(i, j) \in [n] \times [n]$ *and* $r \geqq 0$ *we have:*

$$A_{ij}^{\langle r \rangle} = \sum_{p \in P_{ij}^{\langle r \rangle}} \bar{m}(p),$$

where $P_{ij}^{\langle r \rangle}$ *denotes the set of* (i, j)*-paths of length less or equal to* r *in* E (*note, if* $P_{ij}^{\langle r \rangle} = \emptyset$ *then* $A_{ij}^{\langle r \rangle}=0$).

Proof: (*see* e. g. [Zi 81] or [Ma 82]). □

Since by Lemma 3.4 the entries of a matrix $A^{\langle r \rangle}$ can be interpreted as "sums of path labels", where all paths between the same endpoints up to length r are considered, the relation between path functions and algorithms, which solve path problems, is apparent:

3.5. PROPOSITION (Path Functions and Path Problems): *Let* E *be a graph and* S *a semiring. Let:*

$$T: \quad S^E \to S^{T(E)},$$

be a mapping and for $m \in S^E$ *and* $T(m) \in S^{T(E)}$ $A(m)$ *and* $A(T(m))$ *denote the adjacency matrices of* m *and* $T(m)$ *respectively. Then* T *is a path function if* $A(T(m))=A(m)^{\langle r \rangle}$ *for* $r \geqq n(E)-1$.

Moreover in the following cases the adjacency matrix $A(m)^{\langle r \rangle}$ *of* $T(m)$ *solves the path problem* $X=1+A(m)X$:

(1) E *cycle-free;*

(2) S *simple;*

(3) S *idempotent and* $m: E \to S$ *absorbtive, i.e.* $\bar{m}(c) \leqq 1$ *for all cycles* c.

Proof: The first part of 3.5 follows from Lemma 3.4 since for all $(i, j) \in [n] \times [n]$ $F_{ij} \subseteqq P_{ij}^{\langle r \rangle} \subseteqq P_{ij}$ if $r \geqq n(E)-1$.

(1) follows from the fact that all paths in a cycle-free graph E have at most length $n(E)-1$, so that for $r=n(E)-1$ the matrix $A(m)$ is stable of index r, for all $m: E \to S$ and all semirings S.

(2) is a special case of (3) since in a simple semiring for all elements $a \in S$ $a \leqq 1$ holds, where $\leqq$ is the parial order defined in 2 (10).

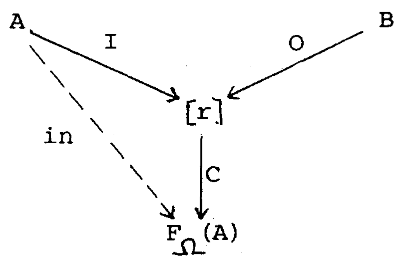
(3) follows from equation ($\star$) in Remark 3.2, which holds for absorbtive graphs. Using Lemma 3.4 one shows that $A_{ij}^{\langle r \rangle} = A_{ij}^{\langle r+1 \rangle}$ for all $(i, j) \in [n] \times [n]$ and $r \geqq n(E)-1$, which implies stability of index $n-1$ and thus the assertion (for a detailed proof *see* also [Ca 79]). □

Properties (1), (2), (3) in 3.5 all reduce to stability of the adjacency matrices $A(m)$. There are other sufficient properties for stability discussed in [Zi 81] which we do not want to discuss here.

In fact most algorithms which solve path problems also compute path functions so that results about complexity of path functions carry over to the complexity of solving path problems.

Next we want to introduce a computational model for path functions which is appropriate to study algebraic complexity of path functions. We use straight line programs (*see* [AHU 74]), which we describe by computation schemes as follows:

3.6. DEFINITION (Computation Scheme): Let A and B be finite sets and $F_\Omega(A)$ be the free term algebra generated by A with the operation symbols $\Omega = \{ +, ., 0, 1 \}$. Then a *computation scheme* is a triple (I, C, O) of mappings



such that:

(1) the canonical embedding in satisfies $\text{in} = C \circ I$;

(2) for all $j \in [r]$ either $C(j) \in \text{in}(A) \cup \{0, 1\}$ or there exist $j_1, j_2 < j$ such that:

$$C(j) = C(j_1) . C(j_2) \text{ or } C(j) = C(j_1) + C(j_2).$$

The elements $j \in [r]$ we call *steps* and distinguish between *input steps*, where $C(j) \in \text{in}(A)$, *output steps*, where $j \in O(B)$, *addition steps*, where $C(j) = C(j_1) + C(j_2)$ and *multiplication steps*, where $C(j) = C(j_1) . C(j_2)$.

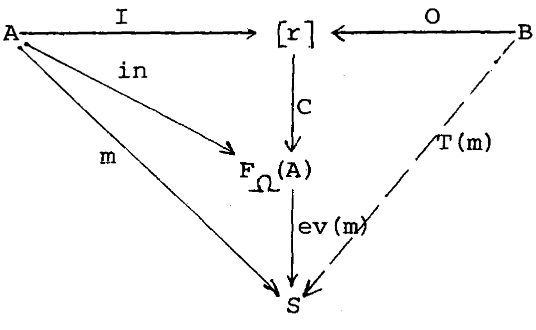
Note, we keep Ω fixed from now on.

Computation schemes can be looked at as a sequence of terms and are obviously derived from straight line programs by inductively evaluating the program steps in the free Ω-algebra $F_\Omega(A)$. The semantics of computation schemes now is declared as follows:

3.7. DEFINITION (Scheme for a Function): Let A and B be finite sets, S an Ω-algebra and $T: S^A \to S^B$ be a function, then a computation scheme (I, C, O) is a *correct scheme for* T (*or computes* T) if for all $m \in S^A$ the equation:

$$T(m) = ev(m) \circ C \circ O. \qquad (1)$$

in the following diagram holds:



where $ev(m)$ is the unique homomorphic extension of m to $F_\Omega(A)$ which describes the evaluation of terms with respect to m.

We will mainly use computation schemes to compute path functions. Then we assume that $A = E \subseteq [n] \times [n]$ and $B = T(E) \subseteq [n] \times [n]$ the transitive closure of E and S a semiring.

In order to prove the lower bounds in the next section we have to restrict the class of computation schemes for path functions to so called "adapted computation schemes" which in a natural way perform computation steps controlled by the graph structure. For this purpose we define adapted terms.

3.8. DEFINITION (Adapted Terms, Adapted Computation Schemes): Let $E \subseteq [n] \times [n]$ be a graph and $F_\Omega(E)$ be the free term algebra generated by E and Ω. Then:

(1) the set of *null terms* NULL (E) is inductively defined by:
- $0 \in \mathrm{NULL}(E)$,
- $(e.e') \in \mathrm{NULL}(E)$ if $e \in \mathrm{NULL}(E)$ or $e' \in \mathrm{NULL}(E)$,
- $(e+e') \in \mathrm{NULL}(E)$ if $e, e' \in \mathrm{NULL}(E)$;

(2) the set of *one terms* ONE (E) is inductively defined by:
- $1 \in \mathrm{ONE}(E)$,
- $(e.e') \in \mathrm{ONE}(E)$ if $e, e' \in \mathrm{ONE}(E)$,

– $(e+e') \in \mathrm{ONE}(E)$ if $e, e' \in \mathrm{ONE}(E)$:

$$\text{or } e \in \mathrm{NULL}(E) \text{ and } e' \in ONE(E),$$
$$\text{or } e \in \mathrm{ONE}(E) \text{ and } e' \in \mathrm{NULL}(E);$$

(3) the set of *regular terms* $\mathrm{REG}(E)$ is inductively defined by $\mathrm{REG}(E) = \bigcup_{(i,j) \in T(E)} \mathrm{REG}(i, j)$ with;

– (i, j) ∈ E implies $(i, j) \in \mathrm{REG}(i, j)$,
– $e \in \mathrm{REG}(i, k)$ and $e' \in \mathrm{REG}(k, j)$ implies $e.e' \in \mathrm{REG}(i, j)$,
– $e \in \mathrm{REG}(i, j)$ and $e' \in \mathrm{REG}(i, j)$ implies $(e+e') \in \mathrm{REG}(i, j)$,
– $e \in \mathrm{REG}(i, j)$ and $e' \in \mathrm{NULL}(E)$ implies $(e+e') \in \mathrm{REG}(i, j)$

$$\text{and } (e'+e) \in \mathrm{REG}(i, j)$$

– $e \in \mathrm{REG}(i, j)$ and $e' \in \mathrm{ONE}(E)$ implies $(e.e') \in \mathrm{REG}(i, j)$

$$\text{and } (e'.e) \in \mathrm{REG}(i, j)$$

– $e \in \mathrm{REG}(i, i)$ and $e' \in ONE(E)$ implies $(e+e') \in \mathrm{REG}(i, i)$

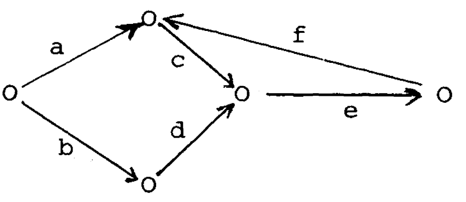
$$\text{and } (e'+e) \in \mathrm{REG}(i, i).$$

The set of *adapted terms* then is defined by:

$$\mathrm{ADAPT}(E) = \mathrm{NULL}(E) \cup \mathrm{ONE}(E) \cup \mathrm{REG}(E),$$

and a *scheme* (I, C, O) is called *adapted* if for all $(i, j) \in T(E)$ the terms $C \circ O(i, j)$ are adapted.

Null terms have the value 0 under every evaluation in a semiring and one terms denote constants defined as a finite sum of one's. Regular terms reflect the graph structure, as is seen in the following example:



adapted terms:

$$((a.c)+(b.d)).e,$$
$$((a.c).e+(b.(d.e)),$$
$$1+(c.(e.f)),$$

not adapted terms are $(a+d).c$ or $c.f$ or $1+(c.e)$.

The following lemma characterizes adapted terms:

3.9 LEMMA (Language of Adapted Terms): *Let* $E \subseteq [n] \times [n]$ *be a graph and* $(2^{E^*}, \cup, \circ, \emptyset, \{\lambda\})$ *be the language semiring generated by E. there is a unique homomorphic extension, called* language,

$$L: \quad F_\Omega(E) \to 2^{E^*},$$

of the mapping $1: E \to 2^{E^*}$ with $1(i, j) = \{(i, j)\}$ for all $(i, j) \in E$, and L satisfies:

(1) $L(e) = \emptyset$ iff $e \in \mathrm{NULL}(E)$;

(2) $L(e) = \{\lambda\}$ if $e \in ONE\ (E)$;

(3) $L(e) \subseteq P_{ij}$ iff $e \in \mathrm{REG}(i, j)$.

Proof: The unique homomorphic extension L of 1 exists from the universal properties of the free Ω-algebra $F_\Omega(E)$. The (if)-directions of (1), (2) and (3) are directly verified from the inductive definition of adapted terms in 3.8. The (only if)-directions are easily derived by induction on the depth of terms in $F_\Omega(E)$. A full proof is in [Ma 79]. □

From 3.9 we can also conclude that the property "adapted" for terms is preserved by all equations 2 (1)-2 (9) which specify idempotent semirings. This conclusion is closely related to Theorem 5.7 below (since the language semiring is a free idempotent semiring, generated by E).

We have now introduced adapted computation schemes as an algorithmic model for path functions. That this is a natural model, might be apparent from Lemma 3.9. But this is also evident from the large number of algorithms in the literature, which are designed to solve path problems and which are "based" on adapted computation schemes for path functions. First of all from the definition of a path function it is clear that each path function has an adapted computation scheme.

In [Ma 80] a class of algorithms for path problems is defined by a general solution scheme which contains almost all known algorithms for path problems [Fredmans $0(n^3)$ algorithm is the only exception known to the author]. This class is devided into the class of label dependent algorithms and that of label independent algorithms. *Label dependent algorithms*, like Dijkstras algorithm which solves path problems over Dijkstra semirings (*see* [Le 77]) are not "based" on computation schemes, while the class of *label independent algorithms*, like Floyds algorithm which solves path problems if the underlying graph is absorptive (*see* [Br 74]), are essentially based on adapted computation schemes. In this class of label independent algorithms are the algorithms of Roy-Warshall, Floyd, Dantzig, all decomposition algorithms in [Br 74], the sensitive algorithms of Tarjan [Ta 75] and all known algorithms especially designed for cycle-free graphs (*cf.* [Ma 79]), only to mention the most prominent. A further distinction between matrix oriented (or rigorous) and graph oriented (or sensitive) algorithms is useful. While matrix oriented algorithms do not reflect the underlying graph structure, like Floyds algorithm, graph oriented algorithms do reflect the graph structure. Among those algorithms are Tarjans algorithms and those for cycle-free graphs.

The lower bounds on the algebraic complexity of path functions, which we state in Section 5, so apply to all label independent algorithms, while the known lower bounds in the literature only apply to matrix oriented label independent algorithms.

4. PROVING LOWER BOUNDS

In this section we give the proofs for the lower bounds on the number of multiplication – and addition-steps of adapted computation schemes for path functions. The idea of the proof is shortly sketched as follows: A computation scheme can be considered as a sequence of terms where input steps denote input variables and output steps denote term definitions of the desired result. To each term in the sequence we assign a set and discuss the sequence of sets. From the property of the set of the first step (precondition) and the property of the set of the last step (postcondition) and the change of the sets from one step to the other we infer the number of steps. As a tool in the proof we use the following:

4.1. LEMMA (Normal Form Presentation of Terms): *Let* $E \subseteq [n] \times [n]$ *be a graph and* E^* *be the free monoid generated by* E. *Then the set of functions* $f : E^* \to \mathbb{N}_0$ *with* finite support $\mathrm{sp}(f) = \{ e \in E^* \mid f(e) \neq 0 \}$ forms a semiring Fin $(E^*, \mathbb{N}_0)$ *with respect to the following operations:*

$$(f+g)(a) = f(a) + g(a),$$
$$(f.g)(a) = \sum_{a=xy} f(x).g(y),$$
$$O(a) = 0$$
$$1(a) = \begin{cases} 1 & \text{if } a = \lambda, \\ 0 & \text{otherwise.} \end{cases}$$

Moreover there is an Ω*-homomorphism* $N : F_\Omega(E) \to \mathrm{Fin}(E^*, \mathbb{N}_0)$ *such that for each mapping* $m : E \to S$ *into a semiring* S *and for each* $e \in F_\Omega(E)$ *the equation:*

$$\text{(1)} \qquad \mathrm{ev}(m)(e) = \sum_{x \in \mathrm{sp}(N(e))} \sum_{i=1}^{N(e)(x)} \bar{m}(x),$$

holds, where m is the unique homomorphic extension of m to E^*.

Finally, the language $L : F_\Omega(E) \to 2^{E^*}$ *of Lemma 3.9 satisfies for each* $e \in F_\Omega(E)$:

$$L(e) = \mathrm{sp}(N(e)). \tag{2}$$

Proof: The sum which defines multiplication in $\mathrm{Fin}(E^*, \mathbb{N}_0)$ is finite since the free monoid E^* is locally finite, i.e. for $a \in E^*$ there is only a finite number of decompositions $a = x.y$. Since $\mathrm{sp}(f+g) = sp(f) \cup \mathrm{sp}(g)$ and $\mathrm{sp}(f.g) = \mathrm{sp}(f) \circ \mathrm{sp}(g)$ where $\circ$ denotes the complex product, it follows that all operations in $\mathrm{Fin}(E^*, \mathbb{N}_0)$ are well defined. To check validity of the equations 2(1) to 2(8) in $\mathrm{Fin}(E^*, \mathbb{N}_0)$ is straight forward.

We define a mapping $h : E \to \mathrm{Fin}(E^*, \mathbb{N}_0)$ by:

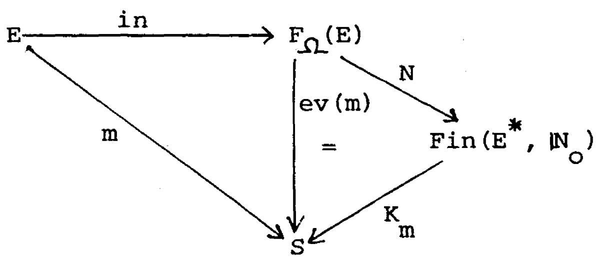
$$h(i, j)(e) = \begin{cases} 1 & \text{if } e = (i, j), \\ 0 & \text{otherwise,} \end{cases}$$

and obtain, since $\mathrm{Fin}(E^*, \mathbb{N}_0)$ is semiring, a unique homomorphic extension $\mathrm{ev}(h) : F_\Omega(E) \to \mathrm{Fin}(E^*, \mathbb{N}_0)$ which we call N.

For each $f \in \mathrm{Fin}(E^*, \mathbb{N}_0)$ one shows that $K_m(f) = \sum_{x \in \mathrm{sp}(f)} \sum_{i=1}^{f(x)} \bar{m}(x)$ is semiring homomorphism for each $m : E \to S$ and derives immediately from the definitions for each $(i, j) \in E$:

$$m(i, j) = (K_m \circ N) \circ \mathrm{in}\ (i, j),$$

in the following diagram:



Since $K_m \circ N$ is semiring homomorphism and $\mathrm{ev}(m)$ unique with the property $m(i, j) = \mathrm{ev}(m) \circ \mathrm{in}(i, j)$ for $(i, j) \in E$, it follows that:

$$\mathrm{ev}(m)(\mathrm{e}) = K_m \circ N = \sum_{x \in \mathrm{sp}(N(e))} \sum_{i=1}^{N(e)(x)} \bar{m}(x),$$

for all $e \in F_\Omega(E)$.

Finally equation 4.1(2) follows from the fact that the support $\mathrm{sp} : \mathrm{Fin}(E^*, \mathbb{N}_0) \to 2^{E^*}$ is a semiring homomorphism which satisfies for $(i, j) \in E$, *see* 3.9,

$$1(i, j) = (\mathrm{sp} \circ N) \circ \mathrm{in}(i, j),$$

so that for all $e \in F_\Omega(E)$ from the uniqueness of the homomorphic extension $L(e) = \mathrm{sp}(N(e))$ follows. □

The next lemma is the key-lemma in the lower bound proofs and used to state the "post condition" for correct computation schemes.

4.2. Lemma: *Let $E \subseteqq [n] \times [n]$ be a graph, $e \in F_\Omega(E)$ be an adapted term and S a semiring. Suppose there is $(i, j) \in T(E)$, $i \neq j$ and $B_{ij} \subseteqq P_{ij}$ such that for all $m : E \to S$:*

$$\mathrm{ev}(m)(e) = \sum_{p \in B_{ij}} \bar{m}(p), \tag{$\star$}$$

then $F_{ij} \subseteqq B_{ij}$ implies $F_{ij} \subseteqq L(e) \subseteqq P_{ij}$.

Proof: From Lemma 4.1 (1) and 4.1 (2) we conclude:

$$\mathrm{ev}(m)(e) = \sum_{x \in L(e)} \sum_{i=1}^{N(e)(x)} \bar{m}(x), \tag{$\star\star$}$$

for all $m : E \to S$.

Since $(i, j) \in T(E)$ and $i \neq j$ is assumed, it follows that $F_{ij} \neq \emptyset$ and $F_{ij} \neq \{\lambda\}$. Thus there is $p \in F_{ij}$ with $p \neq \lambda$ which defines a labelling $m_p : E \to S$ by:

$$m_p(x) := \begin{cases} 1 & \text{if } x \text{ in } p, \\ 0 & \text{otherwise.} \end{cases}$$

Then we conclude for all $q \in P_{ij}$ with $q \neq p$:

$$\bar{m}_p(q) = 0, \tag{1}$$

and from ($\star$) also:

$$\mathrm{ev}(m_p)(e) = 1. \tag{2}$$

Another labelling $m_0 : E \to S$ is defined by $m_0(x) = 0$ for all $x \in E$ and we conclude from ($\star$):

$$\mathrm{ev}(m_0)(e) = 0. \tag{3}$$

1. Suppose $L(e) = \emptyset$; then by ($\star\star$) for all $m : E \to S$ ev (m) (e) $= 0$, which is in contradiction to (2). Thus $L(e) \neq \emptyset$.

2. Suppose $L(e)=\{\lambda\}$; then by $(\star\star)$ for all $m: E \to S$:

$$\mathrm{ev}(m)(\mathrm{e})=\sum_{i=1}^{N(e)(\lambda)} 1.$$

There are two cases: either $\mathrm{ev}(m)(e)=0$ or $\mathrm{ev}(m)(e)\neq 0$. The first case is in contradiction with (2) and the second case in contradiction with (3).

Since e is adapted, it follows from Lemma 3.9 that $L(e)\subseteqq P_{k,1}$ for some $(k, 1)\in T(E)$. Similar to 1 and 2 one shows with the help of $(\star)$ and $(\star\star)$ that $k=i$ and $1=j$ and thus $L(e)\subseteqq P_{ij}$. In detail this is done in [Ma 79].

3. Suppose $F_{ij}\subseteqq L(e)$; then there is a path $p\in F_{ij}$ which is not in $L(e)$. With respect to this path p we conclude from (1) and $(\star\star)$ $\mathrm{ev}(m_p)(e)=0$ which is in contradiction to (2). Thus $F_{ij}\subseteqq L(e)$. □

A closer look to adapted computation schemes shows that there are paths, which are never taken into consideration in the same computation step. Such paths we call "independent". Thus a system of pairwise independent paths gives a lower bound on the number of steps. This is the basic idea of lower bound proofs.

4.3. Definition (Independent Paths): Let $E\subseteqq[n]\times[n]$ be a graph, then two paths p and q in E are called *independent*, if for all adapted terms of the form $e_1 . e_2$ with $e_1, e_2\notin \mathrm{ONE}(E)$:

$$p\in L(e_1 . e_2) \quad\Rightarrow\quad q\notin L(e_1 . e_2).$$

Two paths are independent if and only if they have no common internal node or have different endpoints. This we use in the next section.

4.4. Proposition (Lower Bound for Multiplication Steps): *Let $T: S^E \to S^{T(E)}$ be a path function and $Z\subseteqq \bigcup_{(i,j)\in T(E)} F_{ij}$ be a set of pairwise independent paths of length $\geqq 2$, then each adapted computation scheme (I, C, O) for T has at least:*

$$\mathrm{card}(Z)$$

multiplication steps.

Proof: From 3.1 (1) and 3.7 (1) we derive for all $(i, j)\in T(E)$ and for all $m: E\to S$:

$$\mathrm{ev}(m)\circ(\mathrm{C}\circ O(i, j))=\sum_{p\in B_{ij}} \bar{m}(p),$$

with $F_{ij}\subseteqq B_{ij}\subseteqq P_{ij}$. From Lemma 4.2 we conclude for $(i, j)\in T(E)$ with $i\neq j$.

$$F_{ij}\subseteqq L(C\circ O(i, j))\subseteqq P_{ij}. \tag{1}$$

Let (I, C, O) have r computation steps, then with the abbreviation $L(k) := L(C(k))$ for $k \in [r]$ we have:

$$Z \subseteq \bigcup_{(i,j) \in T(E)} F_{ij} \subseteq \bigcup_{k \in [r]} L(k). \tag{2}$$

Let a mapping $Q : \{0, \ldots, r\} \to 2^{E^*}$ be defined by:

$$Q(0) := \emptyset,$$

$$Q(k) := \begin{cases} \emptyset & \text{if } L(k) \subseteq P_{ij} \text{ for some } (i, j) \in T(E), \\ L(k) & \text{otherwise,} \end{cases}$$

and from Q a mapping $R : \{0, \ldots, r\} \to 2^Z$ defined by:

$$R(k) := Z \bigcup_{j \leq k} Q(j). \tag{3}$$

Then we have as *precondition* $R(0) = \emptyset$ and as *postcondition* $R(r) = Z$ which follows from (1), (2), (3) and the definition of Q.

We now show that for $k = 1, \ldots, n$:

$$\operatorname{card}(R(k-1)) \leq \operatorname{card}(R(k)) \leq \operatorname{card}(R(k-1)) + 1,$$

if k is a multiplication step and $R(k) = R(k-1)$ otherwise:

1. If k is *input step*, then from the definition of L we have $L(k) = \{(i, j)\}$ for some $(i, j) \in E$ and $Q(k) = L(k)$. Since $p \in Z$ has length ≥ 2, we derive $R(k) = R(k-1)$.

2. If k is *null step*, i. e. $C(k) = 0$, then from 3.9 (1) it follows that $L(k) = \emptyset$ and thus $R(k) = R(k-1)$.

3. If k is *one step*, i. e. $C(k) = 1$, then from 3.9 (2) it follows that $L(k) = \{\lambda\}$ and thus $R(k) = R(k-1)$.

4. If k is *addition step*, then $L(k) = L(j_1) \cup L(j_2)$ for some $j_1, j_2 < k$. If $L(k) \subseteq P_{ij}$, for some $(i, j) \in T(E)$, then $R(k) = R(k-1)$. Otherwise from $L(k) \subseteq P_{ij}$ it follows that $L(j_1) \subseteq P_{ij}$ and $L(j_2) \subseteq P_{ij}$. In this case also $Q(k) = Q(j_1) \cup Q(j_2)$ and since $j_1, j_2 \leq k-1$ finally $\bigcup_{j \leq k} Q(j) = \bigcup_{j \leq k-1} Q(j)$ and thus $R(k) = R(k-1)$.

5. If k is *multiplication step*, then $L(k) = L(j_1) \circ L(j_2)$ for some $j_1, j_2 < k$. If $L(k) = \emptyset$ or $L(k) \subseteq P_{ij}$ for some $(i, j) \in T(E)$, then $R(k) = R(k-1)$. If $L(j_1)$ or $L(j_2)$ equals $\{\lambda\}$, then $L(k) = L(j_2)$ or $L(k) = L(j_1)$ respectively and also in this case $R(k) = R(k-1)$.

Finally if $L(k) \subseteq P_{ij}$, $L(j_1) \subseteq P_{iq}$ and $L(j_2) \subseteq P_{qj}$ for some node $q \in [n]$, then $R(k) = R(k-1) \cup (Z \cap L(j_1) \circ L(j_2))$. Since $C(j_1)$ and $C(j_2)$ are adapted terms not in ONE (E), from the definition of independent paths 4.3 it follows that $Z \cap L(j_1) \circ L(j_2)$ has at most one element, so that:

$$\text{card}(R(k-1)) \leqq \text{card}(R(k)) \leqq \text{card}(R(k-1))+1.$$

As we have shown, the sequence $R(k)$ for $k=0, \ldots, r$ grows at most in a multiplication step by one element. From the precondition $R(0)=\emptyset$ and the postcondition $R(r)=Z$ it follows that (I, C, O) has at least card(Z) many multiplication steps. □

The next proposition concerns the number of addition steps. The proof is similar but a bit more complicated. The main idea is that each independent path which is taken into account has to be used in an addition step in order to obtain a correct result.

In fact, the next proposition states a lower bound which is independent from the lower bound for multiplication steps.

4.5. PROPOSITION (Lower Bound for Addition Steps): *Let* $T : S^E \to S^{T(E)}$ *be a pathfunction and for* $(i,j) \in T(E)$ $Z_{ij} \subseteqq F_{ij}$ *a set of pairwise independent paths. Let* sum (Z_{ij}) : $=$card $(Z_{ij})-1$ *if card* $(Z_{ij})>0$ *and* sum (Z_{ij}) : $=0$ *otherwise. Then each adapted computation scheme* (I, C, O) *for* T *has at least:*

$$\sum_{(i,j) \in T(E)} \text{sum}(Z_{ij}),$$

addition steps.

Proof: By Lemma 4.1 there is a step r_{ij} for each $(i,j) \in T(E)$ with $i \neq j$ such that $F_{ij} \subseteqq L(C(r_{ij})) \subseteqq P_{ij}$; thus $Z_{ij} \subseteqq L(C(r_{ij}))$. We show that each adapted computation scheme for T performs at least sum (Z_{ij}) addition steps k, each of which satisfies $C(k) \in \text{REG}(i, j)$. Since this is true for all $(i, j) \in T(E)$ with $i \neq j$ and sum$(Z_{ii})=0$ for all $i \in [n]$, the assertion follows.

Let a mapping $Q : \{0, \ldots, r_{ij}\} \to 2^{E^*}$ be defined by:

$$Q(0) : =\emptyset,$$

$$Q(k) : = \begin{cases} \emptyset & \text{if} \quad L(k) \subseteqq P_{ij}, \\ L(k) & \text{otherwise,} \end{cases}$$

and from Q a mapping $R : \{0, \ldots, r_{ij}\} \to 2^Z ij$ defined by:

$$R(k) : =Z_{ij} \cap \bigcup_{j \leqq k} Q(j).$$

Then we have $R(0)=\emptyset$ and $R(r_{ij})=Z_{ij}$ like in 4.4. The mapping R is appropriate to count multiplication steps. We use it for counting addition steps.

But we need some kind of book keeping for all "partial sums", available in a computation step. For this purpose we define a mapping:

$$K : \{0, \ldots, r_{ij}\} \to 2^{2^{Z_{ij}}},$$

by:

$$K(0) := \{\emptyset\},$$
$$K(k) := A(k) \setminus D(k),$$

where the set $A(k)$ collects all sets $Q(1) \cap Z_{ij}$, for $1 \leqq k$ and $D(k)$ collects all such sets, which are contained in other such sets, so that:

$$A(k) := \{Q(1) \cap Z_{ij} \mid 1 \leqq k\},$$
$$D(k) := \{X \in A(k) \mid X \neq \emptyset, \exists\, Y \in A(k) : X \subsetneqq Y\}.$$

The set $D(k)$ serves to avoid book keeping of "partial sums" of "partial sums".

Then we have as *precondition* $K(0) = \{\emptyset\}$ and as *postcondition* $K(r_{ij}) = \{\emptyset, Z_{ij}\}$.

If we discuss the behavior of the sets $R(k)$ and $K(k)$ for all computation steps $k = 1, \ldots, r_{ij}$, the following situation results:

In each step k the set $R(k)$ grows at most by one element. Whenever $R(k)$ grows by one element, also $K(k)$ grows by one element. Only if k is addition step with $C(k) \in \mathrm{REG}(i, j)$, $K(k)$ can decrease; and then only by one element. The proof then completes as follows:

There have to be card (Z_{ij}) steps where $R(k)$ and thus $K(k)$ increases by one element, since $R(r_{ij}) = Z_{ij}$. Since by the precondition $K(0) = \{\emptyset\}$ and by the postcondition $K(r_{ij}) = \{\emptyset, Z_{ij}\}$, there must be card (Z_{ij})-1 addition steps k with $C(k) \in \mathrm{REG}(i, j)$ if card $(Z_{ij}) > 0$ and none otherwise, which proves the assertion. A formal discussion of the behavior of the sets $R(k)$ and $K(k)$ for all $k = 1, \ldots, r_{ij}$ is in [Ma 79] and immediate from the previous definitions, yet lengthy and technical. □

5. ALGEBRAIC COMPLEXITY OF PATH FUNCTIONS

In this section we study the algebraic complexity of path functions. Namely we investigate how many multiplication and addition steps are necessary and sufficient to be performed by correct computation schemes for path functions. In order to apply the lower bounds from the last section we refer to the notion of independent paths and state (for a proof *see* [Ma 79]).

5.1. LEMMA (Characterizing Independent Paths): *Let $E \subseteqq [n] \times [n]$ be a graph, then two paths p and q in E are independent if and only if p and q have no common internal point or have different endpoints.* □

Given a graph E and let $t(E)$ denote the number of nodes in a minimal cut system of E, i. e.:

$$t(E) = \sum_{(i,j) \in T(E)} t_{ij},$$

with t_{ij} denoting the cardinality of a minimal (i, j)-cut, then we call $t(E)$ the *cut index of E*.

Derived from $t(E)$ we define the *parallel index of E*, denoted by $t^*(E)$, to be the number:

$$t^*(E) = \sum_{(i,j) \in T(E)} t^*_{ij},$$

with $t^*_{ij} = t_{ij}$ if $(i, j) \in E$ or $i = j$ and $t^*_{ij} = t_{ij} - 1$ otherwise.

5.2. FACTS: Let $E \subseteqq [n] \times [n]$ be a graph, then:

(1) $t^*(E) = t(E) - (e(T(E)) - e(E))$;

(2) $t^*(E) \leqq t(E) \leqq n(E).e(E)$;

(3) if E is complete, i. e. $E = [n] \times [n]$, then:

$$t^*(E) = t(E) = n(E).(n(E) - 1).(n(E) - 2).$$

Cut index and parallel index of a graph tell about the complexity of path functions:

5.3. THEOREM (Lower Bounds, Adapted Computations): *Let E be a graph, S a semiring and $T: S^E \to S^{T(E)}$ a path function. Then every adapted computation scheme for T performs at least*

$t(E)$ multiplication steps,

$t^*(E)$ addition steps,

where $t(E)$ is the cut index and $t^*(E)$ is the parallel index of E.

Proof: To show that $t(E)$ multiplication steps are necessary, we use proposition 4.4 and show that there is a set of pairwise independent paths of length $\geqq 2$ which has cardinality $t(E)$. From Lemma 5.1 we conclude, that the maximum number of paths of length $\geqq 2$ in E which are independent and with common endpoints i and j equals the number of (i, j)-paths with no common internal point. Menger's theorem (*see* [Be 76]) shows that this number equals t_{ij}, the cardinality of a minimum (i, j)-cut. Thus it follows that one can find a system Z of pairwise independent paths of length $\geqq 2$ with:

$$\operatorname{card}(Z)=\sum_{(i,j)\in T(E)} t_{ij}=t(E).$$

Necessity of $t^*(E)$ addition steps now follows immediately from proposition 4.5. □

For cycle-free graphs and idempotent semirings we can show that the bounds of Theorem 5.3 are tight:

5.4. THEOREM (Tightness for Cycle-Free Graphs and Idempotent Semirings): *Let E be a cycle-free graph, S an idempotent semiring and $T: S^E \to S^{T(E)}$ a path function. Then there is an adapted computation scheme for T which performs exactly:*

$t(E)$ multiplication steps,

$t^*(E)$ addition steps.

Proof: Let a minimal cut system $\mathscr{A}(E)=(A_{ij})_{(i,j)\in T(E)}$ be given, then we define recursively a family:

$$\mathscr{C}(E)=(e_{ij})_{(i,j)\in T(E)}$$

of terms e_{ij} in $F_\Omega(E)$, $\Omega=\{+, ., 0, 1\}$, as follows:

Let with simplified notation $A_{ij}=\{1, \ldots, t_{ij}\}$, then:

$$(\star)\qquad e_{ij}:=\begin{cases} e_{i1}.e_{1j}+\ldots+e_{it_{ij}}.e_{t_{ij}j}+(i,j) & \text{if } (i,j)\in E,\\ e_{i1}.e_{1j}+\ldots+e_{it_{ij}}.e_{t_{ij}j} & \text{otherwise,}\end{cases}$$

where brackets are neglected without affecting correctness.

Since E is cycle-free, there exists an ordering $h: T(E)\to\mathbb{N}_0$ such that for all $(i,j)\in T(E)$ and all $k\in A_{ij}$:

$$h(e_{ik})<h(e_{ij}) \qquad \text{and} \qquad h(e_{kj})<h(e_{ij}),$$

which proves that $\mathscr{C}(E)$ is well-defined by $(\star)$. One easily derives from $\mathscr{C}(E)$ a computation scheme (I, C, O) such that $C\circ I(i,j)=(i,j)$ for all $(i,j)\in E$ and $C\circ O(i,j)=e_{ij}$ for all $(i,j)\in T(E)$. Actually the set of steps $\{C(k)\mid k\leqq r\}$ (*see* 3.2) is defined to be the set of all subterms of the set of terms $\{e_{ij}\mid(i,j)\in T(E)\}$.

For this computation scheme one can show:

(1) for all $(i, j) \in T(E)$ is e_{ij} an adapted term, i. e. (I, C, O) is an adapted scheme;

(2) for all idempotent semirings S and all labellings $m : E \to S$ and all $(i, j) \in T(E)$ we have:

$$\mathrm{ev}(m)(e_{ij}) = \sum_{p \in P_{ij}} \bar{m}(p),$$

i. e. (I, C, O) is a correct scheme for the path function T which is uniquely specified by S and E;

(3) (I, C, O) performs exactly $t(E)$ multiplication steps and $t^*(E)$ addition steps.

The properties (1), (2), (3) of (I, C, O) prove the assertion. A complete proof is in [Ma 79]. □

The Theorems 5.3 and 5.4 form the main result of this article.

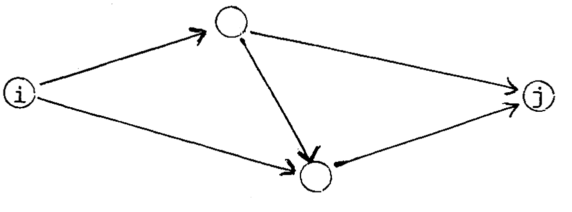
For a better understanding of this result we continue with a discussion about the assumptions made in the statement of the theorems.

Concerning the assumption, that semirings in Theorem 5.4 have to be idempotent we make the following remark:

5.5. Remark (Idempotent Semirings in 5.4): From the definition of $\mathscr{C}(E)$ in the proof of 5.4 one can show that for all $(i, j) \in T(E)$ and all $m : E \to S$:

$$\mathrm{ev}(m)(e_{ij}) = \sum_{p \in P_{ij}} \sum_{i=1}^{N(e_{ij})(p)} \bar{m}(p)$$

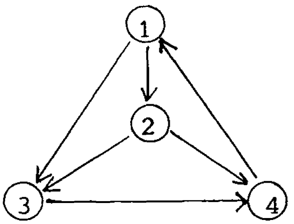
(*see* Lemma 4.1). Since the following example of a cycle-free graph E the unique



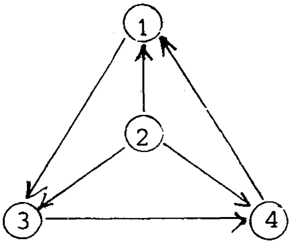
(i, j)-cut decomposes the set of paths P_{ij} not disjointly, it follows that $N(e_{ij})$ in this case is not equal to 1 for all paths $p \in P_{ij}$. In order to prove (I, C, O) in 5.4 correct, we have to assume idempotency of S. In fact for this graph E a different adapted scheme can be constructed which is correct for all path-functions and all semirings and which performs only $t(E)$ multiplications and $t^*(E)$ additions. It is open if this is true for all cycle-free graphs E, which would show that the assumption "S idempotent" can be dropped.

Concerning the assumption that graphs in 5.4 have to be cycle-free, we make the following remark:

5.6. REMARK (Cycle-free Graphs in 5.4): (1) There are cyclic graphs E such that for all simple semirings S the unique path function T can be computed by an adapted scheme (I, C, O) which performs $t(E)$ multiplications and $t^*(E)$ additions, i. e. the bounds of Theorem 5.3 are *tight for some cyclic graphs* and simple semirings. The following graph is an example:



(2) There are cyclic graphs E such that for all simple semirings S the unique path function T can not be computed by an adapted scheme (I, C, O) which performs $t(E)$ multiplications and $t^*(E)$ additions, i. e. the bounds of Theorem 5.3 are *not tight for some cyclic graphs* and simple semirings. The following graph is an example:



In the first case (1) one easily finds a scheme with $t(E)$ multiplication and $t^*(E)$ addition steps. In the second case one observes that E has a unique cut system and shows that any adapted scheme with $t(E)$ multiplication [and $t^*(E)$ addition steps] is incorrect. More details concerning this remark can be found in [LM 80].

Next we discuss the assumption that computation schemes have to be adapted in Theorem 3.3. A stronger correctness condition as in 5.3 leads to the following:

5.7. THEOREM (Lower Bounds, Arbitrary Computations): *Let E be a graph, then every computation scheme which for every semiring S is correct for some path function* $T : S^E \rightarrow S^{T(E)}$, *performs at least:*

$$t\,(E) \text{ multiplication steps,}$$

$$t^*\,(E) \text{ addition steps.}$$

Proof: We show that there is a semiring S_0 such that every path function $T : S_0^E \rightarrow S_0^{T(E)}$ has only adapted computation schemes which are correct. Then the assertion follows from 5.3. Let $S_0 = (2^{E^*}, \cup, \circ, \emptyset, \{\lambda\})$ be the language semiring; then any path function $T : S_0^E \rightarrow S_0^{T(E)}$ satisfies for the argument $m_0 : E \rightarrow S_0$, defined by:

$$m_0(i, j) := \{(i, j)\},$$

for all $(i, j) \in E$, the following equation by 3.1:

$$T(m_0)(i, j) = \sum_{p \in B_{ij}} \bar{m}_0(p),$$

for all $(i, j) \in T(E)$ and some B_{ij} with $F_{ij} \subseteq B_{ij} \subseteq P_{ij}$. In S_0 the equation has the form:

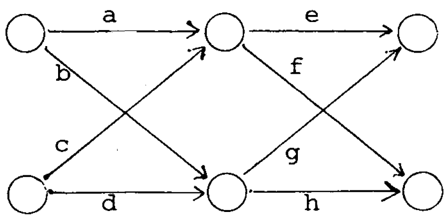
$$T(m_0)(i, j) = B_{ij}.$$

Now let e_{ij} be a term with $\text{ev}(m_0)(e_{ij}) = T(m_0)(i, j)$, then by Lemma 3.5 we have $\text{ev}(m_0)(e_{ij}) = L(e_{ij}) = B_{ij}$ which implies that e_{ij} is a regular (i, j)-term. Consequently every scheme which is correct for $T : S_0^E \rightarrow S_0^{T(E)}$, has to be adapted. □

The stronger correctness condition in 5.7 for example is satisfied by the scheme constructed in the proof of 5.4 and is a quite realistic assumption. Concerning the weaker correctness condition of Theorem 5.3 we make the following remark:

5.8. REMARK (Adapted Schemes in 5.3): It is open if the assumption of adapted schemes in 5.3 can be replaced by some weaker condition without affecting the statement of the lower bounds. In fact some assumption is necessary as the following example shows where additional operations are used (*see* also [AHU 74]).

Let $S = \mathbb{Z}$, the ring of integers, and E be the following (cycle-free) graph:



Then any scheme which computes the matrix product $A.B$ with:

$$A=\begin{pmatrix} a & b \\ c & d \end{pmatrix}, \qquad B=\begin{pmatrix} e & f \\ g & h \end{pmatrix},$$

can trivially (without use of additional operations) be extended to be correct for the unique path function:

$$T: \quad \mathbb{Z}^E \quad \rightarrow \quad \mathbb{Z}^{T(E)}.$$

Using Strassens method to multiply 2×2 matrices with only 7 multiplications instead of 8 (*see* e. g. [AHU 74]) we obtain, with the help of subtraction in $\mathbb{Z}$, 7 multiplication steps instead of $t(E)=8$.

This example however employs a computation scheme which is not covered by the Definition 3.6 since it uses a larger operator domain Ω. But it shows a limit of our results. That there are non adapted schemes over the operator domain $\Omega=\{+, ., 0, 1\}$ with less operations than $t(E)$ and $t^*(E)$ respectively, is not disproved yet.

Finally in this section we want to relate our lower bound results with those known in the literature. The lower bound for path problems, stated by Spira and Pan [SP 73] states $1/2\ (n-1)\ (n-2)$ steps for the single source shortest path problem and applies to label dependent algorithms. The computational model is that of an analytic tree program and is not comparable with computation schemes.

Lower bounds on the algebraic complexity of path problems trace back to Murchland [Mu 68] who informally argues that $n(n-1)(n-2)$ triple operations are necessary to solve the shortest path problem by a label independent algorithm. The same bound is claimed by Iri and Nakamori [IN 71]. Algorithms which use "triple operations" follow the general solution scheme, mentioned in Section 3 (*see* [Ma 80]), and are therefore based on adapted computation schemes. Johnson [Jo 73] proves that $n(n-1)\ (n-2)$ (min, +)-operations are necessary (not just triple operations), following a similar result by Kerr [Ke 70].

Expressed in our terminology, we have:

5.9. THEOREM (Murchland, Iri and Nakamori, Johnson): *Every computation scheme for the (unique) path function:*

$$T : M^{[n] \times [n]} \quad \rightarrow \quad M^{[n] \times [n]},$$

performs at least:

$$e\,(n-1)\,(n-2) \quad \textit{multiplication steps},$$
$$n\,(n-1)\,(n-2) \quad \textit{addition steps},$$

where $\mathbb{M} = (\mathbb{R} \overset{+}{\cup} \{ \infty \}, \min, +, \infty, 0)$. □

Based on a result by Pratt [Pr 75] who shows that transitive closure of Boolean matrices needs $C . n^3$ *and*-gates in monotone (i. e. only *and* and *or* using) networks, Paterson [Pa 74] and Mehlhorn [Me 77] show independently in the same year that $\Theta(n^3)$ *and*- and *or*-gates in monotone networks are necessary to compute transitive closure.

Again expressed in our terminology we have:

5.10. THEOREM (Pratt, Paterson, Mehlhorn): *Every computation scheme for the (unique) path function:*

$$T : \mathbb{B}^{[n] \times [n]} \quad \rightarrow \quad \mathbb{B}^{[n] \times [n]},$$

performs at least:

$$\Theta\,(n^3) \quad \textit{multiplication steps},$$
$$\Theta\,(n^3) \quad \textit{addition steps},$$

where $\mathbb{B} = (\{ 0, 1 \}, \vee, \wedge, 0, 1)$. □

The proof of this bound is making use of the fact that in closed semirings matrix multiplication is not harder than transitive closure (*see* [FM 71]). The main part of the proof, however, states that matrix multiplication over the semiring $\mathbb{B}$ costs at least n^3 multiplication and $n^2\,(n-1)$ additions for arbitrary computations over the operator domain $\Omega = \{ \vee, \wedge, 0, 1 \}$ (*see* [Me 77]). We use this result and extend Theorem 5.10 to positive semirings:

5.11. THEOREM (Path Functions with Positive Semirings): *Let S be a positive semiring and* $T : S^{[n] \times [n]} \rightarrow S^{[n] \times [n]}$ *be a path function. Then every computation scheme for T performs at least:*

$$\Theta\,(n^3) \quad \textit{multiplication steps},$$
$$\Theta\,(n^3) \quad \textit{addition steps}.$$

Proof: First of all the lower bound of n^3 multiplications and $n^2(n-1)$ additions for matrix multiplication over the semiring $\mathbb{B}$ is true for positive semirings in general: Since by the characterization of positive semirings S through the homomorphism $h : S \to \mathbb{B}$ [*see* 2(13)] every scheme which performs matrix multiplication over S also performs it over the semiring $\mathbb{B}$, no positive semiring can exist over which less than n^3 multiplications and $n^2(n-1)$ additions suffice.

Next we state that multiplying matrices is not harder than computing path functions, which is a reformulation of the result in [FM 71]; more precisely: Let for $n>0$ $C(n)$ be a computation scheme which computes a path function $T : S^{[n]\times[n]} \to S^{[n]\times[n]}$ over some (arbitrary) semiring S in not more than $\mu(n)$ multiplication and $\alpha(n)$ addition steps, then there are positive real constants a, b and a computation scheme $D(n)$ which multiplies two $n\times n$ matrices over S in not more than $a.\mu(n)$ multiplication and $b.\alpha(n)$ addition steps, provided that for some positive real constants c, d the relations $\mu(3.n)\leqq c.\mu(n)$ and $\alpha(3.n)\leqq d.\alpha(n)$ hold:

Given $n\times n$-matrices A, B over S and Let $T : S^{[3n]\times[3n]} \to S^{[3n]\times[3n]}$ be an arbitrary path function. Consider the argument m of T in matrix form; it has the value $T(m)$ in matrix form:

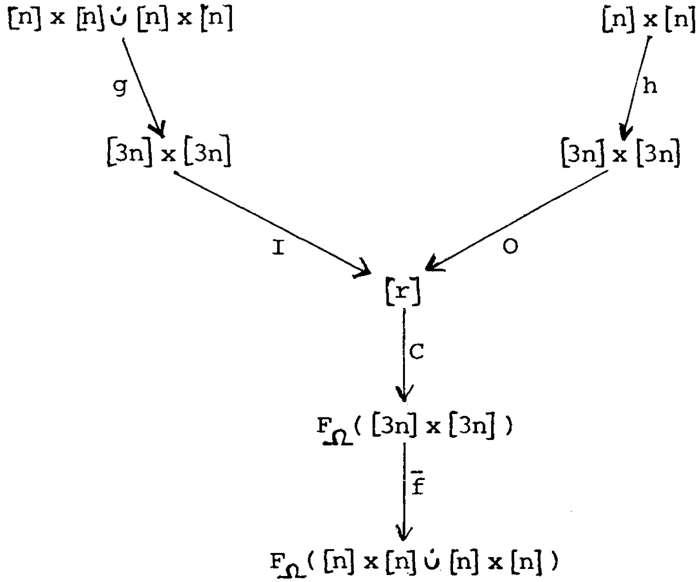
$$m=\begin{pmatrix} 0 & A & 0 \\ 0 & 0 & B \\ 0 & 0 & 0 \end{pmatrix}, \qquad T(m)=\begin{pmatrix} 1 & A & A.B \\ 0 & 1 & B \\ 0 & 0 & 1 \end{pmatrix},$$

as can be easily seen from the definition of a path function (*see* 3.1 and compair [AHU 74]).

Now let $C(3n)=(I, C, O)$ be a computation scheme for T, then we construct a new computation scheme $D(n)=(I', C', O')$ which performs matrix multiplication. Now let $I' := I \circ g$, $O' := O \circ h$ and $C' := \bar{f} \circ C$, where g and h are injections marking the matrix blocks corresponding to A, B and $A.B$ in the above matrices, and $\bar{f}$ is the unique homomorphic extension of f: $[3n]\times[3n] \to F_\Omega([n]\times[n] \dot{\cup} [n]\times[n])$ defined by:

$$f(i,j)=\begin{cases} (i_1, j_1) & \text{if } 1\leqq i\leqq n, \quad n+1\leqq j\leqq 2n, \\ (i_2, j_2) & \text{if } n+1\leqq i\leqq 2n, \quad 2n+1\leqq j\leqq 3n, \\ 0 & \text{otherwise}, \end{cases}$$

with (i_1, j_1) and (i_2, j_2) denoting elements of the first and second "component" of $[n]\times[n] \cup [n]\times[n]$ respectively.



The scheme $D(n)=(I', C', O')$ correctly multiplies $n \times n$-matrices over S in not more than $\mu(3n) \leqq c.\mu(n)$ multiplications and $\alpha(3n) \leqq d.\alpha(n)$ additions. Now suppose there is a computation scheme which computes a path function over positive semirings in $\mu(n)$ multiplication and $\alpha(n)$ addition steps such that $\mu(n)$ or $\alpha(n)$ are not in $\Theta(n^3)$. Then only for μ, also $\mu(3n) \notin \Theta((3n)^3)$ and so $\mu(3n) \notin \Theta(n^3)$. Since then matrix multiplication is not as hard as $\Theta(n^3)$, a contradiction follows from the lower bound of n^3 multiplications and $n^2(n-1)$ additions for matrix multiplication over positive semirings. □

This bound considers a larger class of computation schemes, as the one in Theorem 5.3, but it is more restrictive in the underlying semiring and the type of path function. It is less precise and does not reflect the graph structure.

5.12. Conclusion (Algorithmic Complexity of Path Problems): Algebraic complexity of path problems only considers a certain aspect of algorithms which solve path problems. Organizational overhead, which is part of any path algorithm, is excluded in the analysis. The reason is the following: An algorithm for path problems is usually designed to accept an arbitrary labelled graph G from some class Γ of labelled graphs as input and to result a solution to the problem, specified by G. Straight line programs instead only accept a specific "segment" of Γ, defined by a graph E or its number of nodes $n(E)$. So our complexity analysis employs a non uniform model of computation for a problem which, by its nature, asks for a uniform algorithmic model (*see* [MS 81]). Since we have no good tools up to now to analyze "algorithmic complexity" directly we rely on models that are easier to handle and that admit complexity analysis at all. But we should know that this leaves a number of questions open. The results obtained with non uniform models may not always carry over to complexity bounds for uniform models. We therefore have discussed types of algorithms in Section 3 and claimed that our lower bounds on the base of straight line programs carry over to algorithmic complexity of label independent algorithms. Upper bounds on the number of algebraic operations instead may not always be such for algorithmic complexity as the following results in [Ma 79] show:

Consider the problem of writing an algorithm which solves path problems over cycle-free graphs and idempotent semirings such that it performs only $t(E)$ multiplications and $t^*(E)$ additions. By Theorem 5.4 such an algorithm exists. An easy way is first to compute a minimal cut system of the input graph E and then to proceed with a recursive procedure which follows the definition of the term-family $\mathscr{C}(E)$ in the proof of 5.4. Such an algorithm is described in [Ma 79], which shows that the number of steps of such an algorithm is already determined by the problem of finding the minimal cut system. On the other hand it can be shown that any algorithm which solves path problems over cycle-free graphs and idempotent semirings with graph adapted computations in only $t(E)$ multiplications and $t^*(E)$ additions can be used to find a minimal cut system by simply letting it run on an other algebraic structure, which extends its number of steps by at most a factor of $n=n(E)$.

So solving path problems over cycle-free graphs and idempotent semirings by graph adapted algorithms in a minimum number of operations is at most a factor of $1/n$ "easier" than finding a minimal cut system in a cycle-free graph. While we can solve the path problem in $O(n(E).m(E))$ steps, a solution with minimum number of operations is probably more expensive. This conjectured trade off between algebraic and algorithmic complexity can show that algebraic complexity analysis is not just a tool to study the algorithmic complexity of path problems – a tool that needs to be used carefully – but also a matter in its own right.

REFERENCES

[AHU 74] A. V. AHO, J. E. HOPCROFT and J. D. ULLMAN, *Design and Analysis of Computer Algorithms*, Addison-Wesley, 1974.

[Be 76] C. BERGE, *Graphs and Hypergraphs*, North Holland, 1976.

[Bl 80] P. BLONIARZ, *A Shortest Path Algorithm with Expected Time O (n^2 log n log* n)*, *Proc. of the 12th A.C.M. Symp. on Theory of Computing*, Los Angeles, 1980.

[Br 74] P. BRUCKER, *Theory of Matrix Algorithms*, Math. Systems in Economics, 13, Anton Hain, 1974.

[Ca 79] B. A. CARRÉ, *Graphs and Networks*, Clarendon Press, Oxford, 1979.

[DP 80] N. DEO and C. PANG, *Shortest Path Algorithms: Taxonomy and Annotation*, Washington State University, CS-80-057, 1980.

[Ei 74] S. EILENBERG, *Automata, Languages and Machines*, Vol. A, Academic Press, 1974.

[FM 71] M. J. FISCHER and A. R. MEYER, *Boolean Matrix Multiplication and Transitive Closure*, *I.E.E.E. 12th Ann. Symp. on Switching and Automata Theory*, 1971.

[Fr 76] M. L. FREDMAN, *New Bounds on the Complexity of the Shortest Path Problem*, S.I.A.M. J. Comp., Vol. 5, 1976.

[IN 72] M. IRI and M. NAKAMORI, *Path Sets, Operator Semigroups and Shortest Path Algorithms on a Network*, R.A.A.G, Research Notes, Third Series, No. 185, Univ. Tokyo, 1972.

[Jo 73] D. B. JOHNSON, *Algorithms for Shortest Paths*, TR 73-169, Cornell University, 1973.

[Jo 77] D. B. JOHNSON, *Efficient Algorithms for Shortest Paths in Networks*, J. A.C.M., Vol. 24, 1977.

[Ke 70] L. R. KERR, *The Effect of Algebraic Structures on the Computational Complexity of Matrix Multiplication*, *Ph.D.Thesis*, Cornell, 1970.

[Le 77] D. J. LEHMANN, *Algebraic Structures for Transitive Closure*, Theoretical Computer Science, Vol. 4, 1977.

[LM 80] C. LAUTEMANN and B. MAHR, *A Note on the Complexity of Path Problems*, unpublished, 1980.

[Ma 79] B. MAHR, *Algebraische Komplexität des Allgemeinen Wegeproblems in Graphen*, Techn. Univ. Berlin, Fachbereich Informatik, Vol. 79-14, 1979 (Thesis).

[Ma 80] B. MAHR, *A Birds-Eye View to Path Problems*, LNCS, Vol. 100, Springer, Ed. Noltemeier, 1981.

[Ma 82] B. MAHR, *Semirings and Transitive Closure*, Techn. Univ. Berlin, Fachbereich Informatik, Vol. 82-5, 1982.

[MS 81] B. MAHR and D. SIEFKES, *Relating Uniform and Nonuniform Models of Computation*, Informatik Fachberichte, Springer, Vol. 50, Braner, Ed., 1981.

[Me 77] K. MEHLHORN, *Effiziente Algorithmen*, Teubner, 1977.

[MU 68] J. D. MURCHLAND, *Shortest Distances by a Fixed Matrix Method*, Report LBS-TNT-64, London, Graduate School of Business Studies, 1968 (see [Br 74]).

[Pa 74] M. S. PATERSON, *Complexity of Matrix Algorithms*, handwritten copy, May 1974.

[PR 75] V. R. PRATT, *The Power of Negative Thinking in Multiplying*, Boolean Matrices, S.I.A.M. J. Comp., Vol. 4, 1975.

[Ro 80] F. ROMANI, *Shortest Path Problem is not Harder than Matrix Multiplication*, Information Processing Letters, Vol. 11, No. 3, 1980.

[SP 73] P. M. SPIRA and A. PAN, *On Finding and Updating Shortest Paths and Spanning Trees*, 14th Ann. Symp. on Switching and Automata Theory, 1973.

[TA 75] R. E. TARJAN, *Solving Path Problems on Directed Graphs*, Comp. Sc. Dept. Univ. Stanford, 1975.

[Wa 76] R. A. WAGNER, *A Shortest Path Algorithm for Edge Sparse Graphs*, J. A.C.M., Vol. 23, 1976.

[Zi 81] U. ZIMMERMANN, *Linear and Combinatorial Optimization in Ordered Algebraic Structures*, 1981 (to appear).