10 (1) 1987

LOGICAL DECISION PROBLEMS AND COMPLEXITY OF LOGIC PROGRAMS

Egon Börger, and Ulrich Löwen

Abstract. We survey and give new results on logical characterizations of complexity classes in terms of the computational complexity of decision problems of various classes of logical formulas. There are two main approaches to obtain such results: The first approach yields logical descriptions of complexity classes by semantic restrictions (to e.g. finite structures) together with syntactic enrichment of logic by new expressive means (like e.g. fixed point operators). The second approach characterizes complexity classes by (the decision problem of) classes of formulas determined by purely syntactic restrictions on the formation of formulas.

A UNIFIED MODEL IN INFORMATION RETRIEVAL

S.K.M. Wong, and Wojciech Ziarko

Abstract. We introduce a new information retrieval model, the generalized vector space model (GVSM). In the GVSM, Boolean-like queries can be transformed into vectors spanned by the atoms of the free Boolean algebra generated by the index terms. Documents can, therefore, be ranked with respect to a query as in the conventional vector space model. Most significantly, the GVSM provides a unified view of the Boolean retrieval and vector-processing systems.

Keywords: Information Storage and Retrieval, Vector Space Model

ALGEBRAIC APPROACH TO DATABASE CONSTRAINTS

Ewa Orlowska

Abstract. Algebraic methods of proving database constraints are proposed. The first method is based on the formulation of constraints in terms of inclusions of some binary relations associated with database. The relations are generated by indiscernibility relations by means of the operations of the algebra of binary relations. The second method is based on the matrix representation of constraints.

ON SOME PARAMETERS OCCURRING IN CERTAIN EFFECTIVE CONSTRUCTIONS OF GRAMMARS

Gheorghe Paŭn, and Miroslav Novotný

Abstract. J. Ostrayský considered two classes of languages - B^s and B^C - and described an effective construction as-signing a pure grammar $G_k^j(V, L)$ to any $j \in \{s, c\}$, to any language (V, L), and to any integer $k \geq 1$. He proved that $(V, L) \in B^j$ if, and only if there exists $k_0 \geq 1$ such that $G_k^j(v, L) = G_{k_0}^j(V, L)$ for any $k \geq k_0$. The least integer k_0 with this property is a complexity measure of the language $(V, L) \in B^j$. Besides this complexity measure, we study another one introduced in connection with the above mentioned construction. We compare the complexity of languages with the complexity of languages obtained by usual operations from the given ones.

Keywords: pure grammar, s-grammatizable language, c-grammatizable language, minimum k zero of the language, maximum preequality of the language, s-faithful language, c-faithful language.

A COIN-WEIGHING PROBLEM AND ITS CONNECTION WITH THE SECURITY OF A STATISTICAL DATABASE

Zbigniew Michalewicz *81-92*

Abstract. We consider the problem of compromisability of a statistical database. This paper is an attempt to evaluate the implications of using "yes-no" queries instead of the usual statistical queries; for this purpose we solve a certain coin-weighing problem.

DECISION EXPRESSION OPTIMIZATION

J.R.B. Cockett *93-114*

Abstract. A basic concern when using decision trees for the solution of taxonomic or similar problems is their efficiency. Often the information that is required to completely optimize a tree is simply not available. This is especially the case when a criterion based on probabilities is used. It is shown how it is often possible, despite the absence of this information, to improve the design of the tree. The approach is based on algebraic methods for manipulating decision trees and the identification of some particularly desirable forms.

10 (2) 1987

Abstract. We present a simple proof of Hindley's Theorem: that it is decidable whether a term of the untyped lambda calculus is the image under type-erasing of a term of the simply typed lambda calculus. The proof proceeds by a direct reduction to the unification problem for simple terms. This arrangement of the proof allows for easy extensibility to other type inference problems.

Abstract. We show an example of an algebra $\tau^\bullet$, such that every flow-chart program is equivalent in $\tau^\bullet$ to a loop-free approximation of itself, while a program augmented by one binary push-down store is not equivalent in $\tau^\bullet$ to any loop-free program. From this we deduce that the Deterministic Dynamic Logic of regular programs is strictly weaker than the Deterministic Context-Free Dynamic Logic.

Abstract. We prove that every nontrivial even linear language is linearly grammatizable by means of derivatives.

Keywords: generalized grammar with linear productions, derivative, permitting triple with derivatives corresponding to a language, language linearly grammatizable by means of derivatives, even context, even category, even linear language.

Abstract. A non-sequential process can be modelled by a partially ordered set. Conversely, one is led to study the properties to be fulfilled by a poset so that it can reasonably be viewed as the model of a non-sequential process. To this end, C.A. Petri has proposed a set of concurrency axioms and a related property called D-continuity, a generalised version for partially ordered sets of Dedekind's completeness property of the real numbers.

In this paper we study Petri's axioms of concurrency and some of their interdependencies. We also derive several characterisations of D-continuity and exhibit its relation with the axioms of concurrency. Furthermore we apply our work to Petri nets: we introduce occurrence nets, some special posets which model the processes of a system net and we present their relations to D-continuity and the axioms of concurrency. Finally we identify the class of the system nets whose processes are D-continuous and satisfy the axioms of concurrency.

PROCESS ALGEBRA SEMANTICS FOR QUEUES

Jan Bergstra, and Jerzy Tiuryn *213-224*

Abstract. Centre of Mathematics and Computer Science, Amsterdam ft University of Warsaw, Institute of Mathematics An unbounded queue over a finite set of data values is a process Q in A^∞ defined by an infinite system of guarded equations. The aim of this paper is to show that no finite system of guarded equations is capable of defining Q.

10 (3) 1987

VARIANTS OF DETERMINACY IN PARALLEL PROGRAM SCHEMATA

Udo Kelter *225-246*

Abstract. Systems which use parallel processing to speed up response times are normally required to be determinate. Parallel program schemata are formal models of parallel systems which serve to exactly define determinacy. Three variants of determinacy have frequently been used in the literature: Two "semantic" ones, which refer to the final contents of variables and to the sequences of contents, and a "syntactic" one, which is based upon the notion of conflict. Variants of determinacy can be derived with a universal definition of determinacy (as presented in [K73]) which uses a parameter, namely an equivalence relation on computations. Each of the above variants corresponds to a certain equivalence relation.

The main aim of tnis paper is to investigate the role of "read-only"-operations, e.g. pure tests. Results of tests do not matter for both semantic forms of determinacy, wnereas they are relevant for the syntactic form. We will snow that different variants of semantic and syntactic determinacy can be obtained by judging read-only operations as relevant or not.

Keywords: determinacy, equivalence of computations, program schemata.

FAIR DERIVATIONS IN LOGIC PROGRAMMING: OPERATIONAL AND GREATEST FIXPOINT SEMANTICS

Mohand-Areski Nait Abdallah *247-308*

Abstract. Using topological methods, we study the operational and greatest fixpoint semantics of infinite computations in logic programming. We show the equivalence of the operational and greatest fixpoint semantics in the case of fair derivations. We give some canonical partition, and soundness and completeness results which generalize already known results about the finitary case. Since fair derivations are a generalization of successful derivations, we thus give a uniform treatment of all meaningful computations in logic programming.

THE REGULAR EXPRESSION DESCRIPTIONS OF UNIFIER SET IN THE TYPED X-CALCULUS

Marec Zaionc *309-322*

Abstract. X-language over simple type structure is considered. We investigate unification problem such that number of elementary substitutions in a potentially infinite matching tree introduced by Huet is finite (compare [8]). The substitutions are generated using term gram-mar technique. Such a grammar consists of productions which are on the form; nonterminal variable of certain type produces a term of the same type. The most interesting is the case when matching tree is infinite and includes an infinite number of most general unifiers. We will show an interesting property of the infinite unification trees from examined class: for every infinite branch there is a node which occurs earlier in this tree. It means that there are some fragments which are repeated in this tree. Therefore the set of most general unifiers which are represented by terminal nodes can be described by regular expression over finite alphabet. Regular expression are constructed as a solution of linear language equalities. For every problem from this class the set of unifiers is described by some language. This language can be approximated by regular languages.

Keywords: typed X-calculus, unification, infinite trees, regular expressions, matching trees.

UNIQUE DECIPHERABILITY FOR PARTIALLY COMMUTATIVE ALPHABETS

Marek Chrobak, and Wojciech Rytter *323-336*

Abstract. We consider the unique decipherability problem for partially commutative alphabets. It is shown that the decidability of the problem depends heavily on the size of the alphabet and the structure of the graph of the commutativity relation. In particular, for alphabets with at most three letters the problem is decidable and for alphabets of bigger size the problem is undecidable.

10 (4) 1987

Abstract. We study the programming formalism FD of "flow-diagrams" to which we gradually add various features of concurrency. The weakest form of concurrency is introduced by the construct "**and**", which is dual to the nondeterministic choice "**or**" and plays a role similar to universal states in alternating Turing machines. Stronger (and more realistic) forms of concurrency are obtained when processes are allowed to communicate. We consider communication by channels and communication by messages. We calibrate the computational power of classes of concurrent programs $FD + \alpha$ against that of sequential pro-grams, where a is the addition of one of the following features: {and}, {and, or}, {and, or, channels}, or {and, or, messages}.

Abstract. By a permutation machine we mean a triple (Q, S, F), where Q and S are finite sets and F is a function $Q \times S \to Q$ which defines a permutation on Q for every element from S. These permutations generate a permutation group G and by considering the structure of G we can obtain efficient ways to decompose the transformation group (Q, G). In this paper we first consider the situation where G is half-transitive and after this we show how to use our result in the general non-transitive case.

Abstract. This paper recalls some fixpoint theorems in ordered algebraic structures and surveys some ways in which these theorems are applied in computer science. We describe via examples three main types of applications: in semantics and proof theory, in logic programming and in deductive data bases.