14 (1) 1991

VECTOR CONTROLLED CONCURRENT SYSTEMS, PART II: COMPARISONS

N.W. Keesmaat, H.C.M. Kleijn, and G. Rozenberg

Abstract. In this paper the behaviour of models of concurrent systems introduced in [KeeKleRoz] is investigated. In particular (i) static and dynamic models are compared and (ii) the effect of restricting the sequential components to be regular is considered.

MODELLING NONDETERMINISTIC CONCURRENT PROCESSES WITH EVENT STRUCTURES

Rita Loogen, Ursula Goltz

Abstract. We present a non-interleaving model for nondeterministic concurrent processes that is based on labelled event structures. We define operators on labelled event structures like parallel composition, nondeterministic combination, choice, prefixing and hiding. These operators correspond to the operations of the "Theory of Communicating Sequential Processes" (TCSP). Infinite processes are defined using the metric approach. The dynamic behaviour of event structures is defined by a transition relation which describes the execution of partially ordered sets of actions, abstracting from internal events.

ON UNIQUENESS OF THE WULFF SHAPE FOR CELLULAR AUTOMATA

Pawel Wlaź

Abstract. In this paper, ordered transition rules are investigated. Such rules describe an increment of mono-crystals and for every rule one can calculate so called Wulff Shape. It is shown that for some large class of these rules, there exists at most one growth function which generates a given Wulff Shape.

APPLICATIONS OF FINITE MODELS PROPERTIES IN APPROXIMATION AND ALGORITHMIC LOGICS

Jaroslaw Stepaniuk

Abstract. The purpose of this paper is to investigate some aspects concerning elementary theories of finite models and to give the applications in approximation logics and algorithmic theory of dictionaries.

A COMPOSITIONAL SEMANTICS FOR UNMARKED PREDICATE/ TRANSITION NETS(+)

Andrea Maggiolo Schettini, G. Michele Pinna, and Józef Winkowski

Abstract. Predicate/Transition nets are a high level description tool for complex systems. The paper aims at developing a semantics for Unmarked Predicate/Transition nets. This semantics, based on the concept of behavior as isomorphic class of unfoldings of the net, is compositional, i.e. the behavior of a complex net can be obtained by a suitable composition of the behaviors of its parts. The concept of behavior introduced here is similar to that of process for which an algebra

has been developed in [16]. A case in which the behavior of Marked Predicate/Transition nets can be derived from the behaviors of its parts is also discussed.

MINIMIZATION OF RESOLUTION PROOF SYSTEMS

Zbigniew Stachniak *129-146*

Abstract. In this paper we show that minimal resolution counterparts of strongly finite logics can be effectively constructed. Moreover, we show that the class of resolution counterparts of structural propositional logics coincides with the class of resolution counterparts of strongly finite logics.

14 (2) 1991

UNIFIED THEORY OF DATABASE SERIALIZABILITY*!

K. Vidyasankar *147-183*

Abstract. A database system is a collection of data items, read or written by transactions in a possibly interleaved fashion. An interleaved execution is assumed to be correct if the sequence of the steps of the transactions, called *history*, is *serializable*, that is, the effect of the execution is equivalent to that of some serial execution of the same transactions. In this paper we give a new characterization of serializability that brings out the inherent problem of serialization explicitly.

We then give a graph-theoretic analogue of serializable histories. We define a new class of graphs, called *serializable graphs*, whose properties are such that (i) a serializable graph can be associated with each serializable history, and this can be done for various notions of serializability of histories and for serializability under various sets of constraints, and (ii) a serializable history, in fact a serial one, can be associated with each serializable graph. We use serializable graphs to characterize, in an intuitive manner, serializable histories involving general multi-step transactions, where the same data item can be accessed by several read and write steps of a transaction in an arbitrary manner, and those involving nested transactions.

We also define a new notion of serializability for nested transactions. This enables relating several acceptable concurrent executions of transactions, that are not serializable with the traditional transaction concept, to sequential behaviour. Serializability under this notion is also characterized.

The main graph-theoretic properties used in these characterizations are a directed cutset match-ing property and graph contraction.

A UNIFORM TEST APPROACH FOR RCC-ADDERS1

Bernd Becker, and Uwe Sparmann *185-219*

Abstract. In this paper testability aspects of Recursive Carry Computation adders are considered. The class of RCC-adders has been introduced in [5] and contains a wide range of different adder realizations (e.g., optimal time adders such as the the carry look-ahead adder of [8] and the conditional carry adder of [5]).

We show that symbolic computation can be used to define this class and at the same time offers a uniform test approach which can be applied at an early stage of the design process. The class of RCC-adders itself splits into several subclasses which are specified by structural properties of the overall computation scheme and functional properties of the basic cells. Optimal complete test sets with respect to two commonly used fault models, the single stuck-at fault model and the single cellular fault model, are developed for these RCC-subclasses. The cardinality of the test sets depends on the choice of the fault model and on structural properties of the RCC-subclass.

To summarize our results, we finally obtain tables with upper and lower bounds characterizing the test complexity of classes of RCC-adders. The upper bounds are obtained by the effective construction of complete test sets. The cardinality of these sets varies between a logarithmic or linear number of patterns for an n-bit RCC-adder.

ABOUT SEMANTIC ACTION REFINEMENT

Philippe Darondeau Irisa, and Pierpaolo Degano *221-234*

Abstract. A notion of semantic action refinement is defined both on Synchronization Trees and on Causal Trees, a class of trees recently devised for giving a full account to causality [DD89]. The branching bisimulation, as introduced in [GW89a], is shown to be preserved under semantic action refinement. As a by-product, the axiomatization of the congruence induced by branching bisimulation which was given in [GW89a] is still valid under action refinement. Both results hold for Synchronization Trees and, with the needed extensions, for Causal Trees.

AXIOM SYSTEM INDUCED BY CTL* LOGIC

Maciej Koutny *235-253*

Abstract. We discuss some of the problems concerning bisimulation relations over behaviour expressions representing non-sequential systems. The kind of bisimulation we investigate is derived from the notion of bisimulation relation between two Kripke structures, which provides a full characterisation of semantical equivalence of such structures w.r.t. the CTL* branching time temporal logic without the next-time operator. This new bisimulation on behaviour expressions, called CTL*-bisimulation, induces a congruence on recursion-free behaviour expressions, and in this paper we derive a sound and complete axiom system for this congruence.

A HIERARCHY OF CONTEXT-FREE DERIVATIONS

Erkki Mäkinen *255-259*

Abstract. This note discusses a hierarchy of different kind of context-free derivations by studying the corresponding Szilard languages. Especially, we introduce the leftmost and rightmost breadth-first derivations which are the counterparts of the well-known (depth-first) leftmost and rightmost derivations. It is shown that the Szilard languages related to different derivation types can be recognized by one-state automata which differ from each others by their memory devices.

TEMPORAL LOGIC AND RECURSION

Fred Kröger, and Stephan Merz *261-281*

Abstract. We propose a temporal logic based on structures divided into several layers of linear "time scales" and give a sound and complete derivation system. The logic is applied to the formulation and verification of assertions about sequential recursive programs.

14 (3) 1991

A NOTE ON PF(k) – PARSABLE LANGUAGES

Tudor Balanescu, and Marian Gheorghe *283-286*

Abstract. This paper investigates the class of regular languages which are **pf(k)** - parsabie. It is shown that **k** induces an infinite hierarchy in the family of regular languages and that every regular language is a homomorphic image of a **pf(0)** - parsable language. A pumping lemma for **pf** - parsable languages is also provided.

Keywords: pf(k) - parsable languages, pumping lemma.

DIAMOND PROPERTIES OF ELEMENTARY NET SYSTEMS

Hendrik Jan Hoogeboom, and Grzegorz Rozenberg *287-300*

Abstract. An *elementary net system* is the basic system model of net theory. The state space of an elementary net system is formalized through the notion of the case graph, which is an edge-labeled graph with a distinguished initial node. The paper investigates syntactic, i.e., graph theoretic properties of case graphs of elementary net systems. In particular it studies the structure of isomorphisms between case graphs.

ON PROCESSING OF FUZZY PRODUCTION SYSTEMS

Stefka P. Stoeva *301-312*

Abstract. Knowledge bases in the form of fuzzy production systems with weighting coefficients, i.e. sets of weighted fuzzy rules, are studied. A software implementation of a rule-firing algorithm, based on Zadeh's generalized modus ponens, is presented. The time complexity of the algorithm proposed is determined. Different types of parallelism in processing of fuzzy production systems are explored.

ON GENERALIZED AUTOMATA

Rana Barua *313-322*

Abstract. The usual notions of acceptance of automata acting on infinite strings yield sets that are in very low Borel classes. In ./ response to a question by Landweber, Wisniewski introduced the notion of generalized automata that yield sets at any specified level of the Borel hierachy. Here we show that most of / Wismewski's results can be easily derieved (and generalized) using the theory of positive analytical operations and $\delta - s$ operations.

MODELS OF CLARK'S COMPLETION FOR SOME CLASSES OF LOGIC PROGRAMS

Stefano Baratella *323-337*

Abstract. Aim of this paper is to provide an alternative proof of consistency for the completion of a *semi-strict* program. Furthermore, we prove the continuity of the immediate consequence map associated to a *locally call-consistent* program with respect to any preinterpretation that is a model of the equality theory of its Clark's completion.

Abstract. An embedding of minimal sets for default theories into nonmonotonic and nonmonotonic ground logics based on S4 and S5 is presented.

Abstract. In the paper we study a family of modal nonmonotonic logics closely related to the family of modal nonmonotonic logics proposed by McDermott and Doyle. For a modal logic S and a fixed collection of formulas X we introduce the notion of an (S, X)-expansion. We restrict to modal logics which have a complete Kripke semantics. We study the properties of (S, X)-expansions and show that in many respects they are analogous to the properties of S-expansions in nonmonotonic modal logics of McDermott and Doyle.

Abstract. A number of approaches have been taken to represent compound, structured values in relational databases. We review a few such approaches and discuss a new approach, in which every set is represented as a Boolean term. We show that this approach generalizes the other approaches, leading to more flexible representation. Boolean term representation seems to be appropriate in handling incomplete information: this approach generalizes some other approaches (*e.g.* null value mark, null variables, *etc*). We consider definitions of algebraic operations on such sets, like join, union, selection, *etc.* Moreover, we introduce a measure of computational complexity of these operations.

14 (4) 1991

THREE VALUED PREDICATES FOR SOFTWARE SPECIFICATION AND VALIDATION

Andrzej Blikle *387-410*

Abstract. Partial functions, hence also partial predicates, cannot be avoided in algorithms. However, in spite of the fact that partial functions have been formally introduced into the theory of software very early, partial predicates are still not quite commonly recognized. In many programming- and software-specification languages partial Boolean expressions are treated in a rather simplistic way: the evaluation of a Boolean sub-expression to an error leads to the evaluation of the hosting Boolean expression to an error and, in the consequence, to the abortion of the whole program. This technique is known as an *eager evaluation* of expressions.

A more practical approach to the evaluation of expressions – gaining more interest today among both theoreticians and programming-language designers – is *lazy evaluation*. Lazily evaluated Boolean expressions correspond to (non-strict) three-valued predicates where the third value represents both an error and an *undefinedness*. On the semantic ground this leads to a three-valued propositional calculus, three-valued quantifiers and an appropriate logic.

This paper is a survey-essay devoted to the discussion and the comparison of a few three-valued propositional and predicate calculi and to the discussion of the author's claim that a two-valued logic, rather than a three-valued logic, is suitable for the treatment of programs with three-valued Boolean expressions.

The paper is written in a formal but not in a formalized style. All discussion is carried on a semantic ground. We talk about predicates (functions) and a semantic consequence relation rather than about expressions and inference rules. However, the paper is followed by more formalized works which carry our discussion further on a formalized ground, and where corresponding formal logics are constructed and discussed.

A THREE-VALUED LOGIC FOR SOFTWARE SPECIFICATION AND VALIDATION Tertium tamen datur

Beata Konikowska, Andrzej Tarlecki, and Andrzej Blikle *411-453*

Abstract. Different calculi of partial or three-valued predicates have been used and studied by several authors in the context of software specification, development and validation. This paper offers a critical survey on the development of three-valued logics based on such calculi.

In the first part of the paper we review two three-valued predicate calculi, based on, respectively, McCarthy's and Kleene's propositional connectives and quantifiers, and point out that in a three-valued logic one should distinguish between two notions of validity: *strong validity* (always true) and *weak validity* (never false). We define in model-theoretic terms a number of consequence relations for three-valued logics. Each of them is determined by the choice of the underlying predicate calculus and of the weak or strong validity of axioms and of theorems. We discuss mutual relationships between consequence relations defined in such a way and study some of their basic properties.

The second part of the paper is devoted to the development of a formal deductive system of inference rules for a three-valued logic. We use the method of semantic tableaux (slightly modified to deal with three-valued formulas) to develop a Gentzen-style system of inference rules for deriving valid sequents, from which we then derive a sound and complete system of natural deduction rules. We have chosen to study the consequence relation determined by the predicate calculus with McCarthy's propositional connectives and Kleene's quantifiers and by the strong interpretation of both axioms and theorems. Although we find this choice appropriate for applications in the area of software specification, verification and development, we regard this logic merely as an example and use it to present some general techniques of developing a sequent calculus and a natural deduction system for a three-valued logic. We also discuss the extension of this logic by a non-monotone is??true predicate. 1Non-(tertium non datur)

ALGEBRAIC THEORY OF INDEPENDENCE IN INFORMATION SYSTEMS

Miroslav Novotný, and Zdzizlaw Pawlak *454-476*

Abstract. The semilattice of all subsets of a. finite nonempty set provided with the operation of union and a congruence of this semilattice enable to define independent elements of the semilattice. Reducts, subreducts, and superreducts of an arbitrary element in the semilattice are introduced and investigated. A particular type of congruence, the so called rough top equality, is studied.

These algebraic notions and methods are applied to black boxes, to information systems, and to contexts in Wille's sense. The algebraic results are used to solve the problem whether two sets of inputs give the same output in a black box, whether two sets of attributes define the same classification of objects in an information system, and, finally, whether two sets of features generate the same concept in a context.

Partial dependence between two sets of attributes of an information system is introduced and a distance between these sets is defined.

Keywords: dependence space, independence, reduct, subreduct, superreduct, rough top equality, black box, information system, context, partial dependence

AN AXIOMATIC CHARACTERIZATION OF A CLASS OF PETRI NETS

Waldemar Korczynski *477-491*

Abstract. In this paper an algebraic characterization of a class of Petri nets is given. The nets are characterized by a kind of algebras, which can be considered as a generalization of the concept of the case graph of a (marked) Petri net.

AN ALGEBRAIC APPROACH TO THE APPROXIMATION OF INFORMATION

Stephen D. Comer *492-502*

Abstract. This paper is based on the notion of an information system U, Ω, V, f in the sense of Pawlak. Every set of knowledge $P \subseteq \Omega$ determines a closure operator on U. The class of Boolean algebras with added operations determined by all sets of knowledge are axiomatixed. As a consequence of the representation theorem information systems can be constructed that have a prescribed lattice of functional dependencies.