

## Formal Languages And Applications

The huge number and broad range of the existing and potential applications of fuzzy logic have precipitated a veritable avalanche of books published on the subject. Most, however, focus on particular areas of application. Many do no more than scratch the surface of the theory that holds the power and promise of fuzzy logic. Fuzzy Automata and Languages: Theory and Applications offers the first in-depth treatment of the theory and mathematics of fuzzy automata and fuzzy languages. After introducing background material, the authors study max-min machines and max-product machines, developing their respective algebras and exploring properties such as equivalences, homomorphisms, irreducibility, and minimality. The focus then turns to fuzzy context-free grammars and languages, with special attention to trees, fuzzy dendrolanguage generating systems, and normal forms. A treatment of algebraic fuzzy automata theory follows, along with additional results on fuzzy languages, minimization of fuzzy automata, and recognition of fuzzy languages. Although the book is theoretical in nature, the authors also discuss applications in a variety of fields, including databases, medicine, learning systems, and pattern recognition. Much of the information on fuzzy languages is new and never before presented in book form. Fuzzy Automata and Languages incorporates virtually all of the important material published thus far. It stands alone as a complete reference on the subject and belongs on the shelves of anyone interested in fuzzy mathematics or its applications.

This book provides a concise and modern introduction to Formal Languages and Machine Computation, a group of disparate topics in the theory of computation, which includes formal languages, automata theory, turing machines, computability, complexity, number-theoretic computation, public-key cryptography, and some new models of computation, such as quantum and biological computation. As the theory of computation is a subject based on mathematics, a thorough introduction to a number of relevant mathematical topics, including mathematical logic, set theory, graph theory, modern abstract algebra, and particularly number theory, is given in the first chapter of the book. The book can be used either as a textbook for an undergraduate course, for a first-year graduate course, or as a basic reference in the field. Contents:Computation-Related Mathematics:Logics and ProofsSets, Functions and GraphsDivisibility, Continued Fractions and CongruencesGroups, Rings and FieldsFormal Languages and Automata:Languages, Grammars and AutomataFinite Automata and Regular LanguagesPush-Down Automata and Context-Free LanguagesTuring Machines and Recursively Enumerable LanguagesTuring Computability and Complexity:Computability and NoncomputabilityDecidability and UndecidabilityComputational ComplexityDesign and Analysis of AlgorithmsNumber-Theoretic Computations and Applications:Primality TestingInteger FactorizationDiscrete LogarithmsCryptography and Systems SecurityHigh-Speed ComputationThree More Applications in ComputingNew Models of Computation:Quantum ComputationBiological ComputationComparison of Quantum and DNA Biological ModelsComparison of Connectionist and DNA Biological Models Readership: Students, teachers and researchers in computer science. keywords:Formal Languages;Automata;Computability;Complexity;Models of Computation

This book constitutes the refereed proceedings of the 12th International Conference on Language and Automata Theory and Applications, LATA 2018, held in Ramat Gan, Israel, in April 2018.The 20 revised full papers presented together with 3 invited papers were carefully reviewed and selected from 58 submissions. The papers cover fields like algebraic language theory, algorithms for semi-structured data mining, algorithms on automata and words, automata and logic, automata for system analysis and programme verification, automata networks, automatic structures, codes, combinatorics on words, computational complexity, concurrency and Petri nets, data and image compression, descriptonal complexity, foundations of finite state technology, foundations of XML, grammars (Chomsky hierarchy, contextual, unification, categorial, etc.), grammatical inference and algorithmic learning, graphs and graph transformation, language varieties and semigroups, language-based cryptography, mathematical and logical foundations of programming methodologies, parallel and regulated rewriting, parsing, patterns, power series, string processing algorithms, symbolic dynamics, term rewriting, transducers, trees, tree languages and tree automata, and weighted automata.

Written for graduate students and advanced undergraduates in computer science, A Second Course in Formal Languages and Automata Theory treats topics in the theory of computation not usually covered in a first course. After a review of basic concepts, the book covers combinatorics on words, regular languages, context-free languages, parsing and recognition, Turing machines, and other language classes. Many topics often absent from other textbooks, such as repetitions in words, state complexity, the interchange lemma, 2DPDAs, and the incompressibility method, are covered here. The author places particular emphasis on the resources needed to represent certain languages. The book also includes a diverse collection of more than 200 exercises, suggestions for term projects, and research problems that remain open.

This uniquely authoritative and comprehensive handbook is the first to cover the vast field of formal languages, as well as its traditional and most recent applications to such diverse areas as linguistics, developmental biology, computer graphics, cryptology, molecular genetics, and programming languages. No other work comes even close to the scope of this one. The editors are extremely well-known theoretical computer scientists, and each individual topic is presented by the leading authorities in the particular field. The maturity of the field makes it possible to include a historical perspective in many presentations. The work is divided into three volumes, which may be purchased as a set.

This book constitutes the proceedings of the 4th International Conference, LATA 2010, held in May 2010 in Trier, Germany. The 47 full papers presented were carefully selected from 115 submissions and focus on topics such as algebraic language theory , algorithmic learning, bioinformatics, computational biology, pattern recognition, program verification, term rewriting and tree machines.

Formal Languages and Applications provides a comprehensive study-aid and self-tutorial for graduates students and researchers. The main results and techniques are presented in an readily accessible manner and accompanied by many references and directions for further research. This carefully edited monograph is intended to be the gateway to formal language theory and its applications, so it is very useful as a review and reference source of information in formal language theory.

[Theory and Applications](#)

[13th International Conference, LATA 2019, St. Petersburg, Russia, March 26-29, 2019, Proceedings](#)

[Formal Models in the Study of Language](#)

[Formal Languages for Computer Simulation: Transdisciplinary Models and Applications](#)

[Formal Methods for Software Engineering](#)

[Transdisciplinary Models and Applications](#)

[Introduction to Formal Languages](#)

[A Practical Introduction](#)

[Modern Applications of Automata Theory](#)

[Formal Languages and Automata Theory](#)

The theory of formal languages is widely accepted as the backbone of theoretical computer science. It mainly originated from mathematics (combinatorics, algebra, mathematical logic) and generative linguistics. Later, new specializations emerged from areas of either computer science (concurrent and distributed systems, computer graphics, artificial life), biology (plant development, molecular genetics), linguistics (parsing, text searching), or mathematics (cryptography). All human problem solving capabilities can be considered, in a certain sense, as a manipulation of symbols and structures composed by symbols, which is actually the stem of formal language theory. Language – in its two basic forms, natural and artificial – is a particular case of a symbol system. This wide range of motivations and inspirations explains the diverse applicability of formal language theory and all these together explain the very large number of monographs and collective volumes dealing with formal language theory. In 2004 Springer-Verlag published the volume Formal Languages and Applications, edited by C. Martin-Vide, V. Mitran and G. Păun in the series Studies in Fuzziness and Soft Computing 148, which was aimed at serving as an overall course-aid and self-study material especially for PhD students in formal language theory and applications. Actually, the volume emerged in such a context: it contains the core information from many of the lectures delivered to the students of the International PhD School in Formal Languages and Applications organized since 2002 by the Research Group on Mathematical Linguistics from Rovira i Virgili University, Tarragona, Spain.

This volume presents articles that focus on the application of formal models in the study of language in a variety of innovative ways, and is dedicated to Jacques Moeschler, professor at University of Geneva, to mark the occasion of his 60th birthday. The contributions, by seasoned and budding linguists of all different linguistic backgrounds, reflect Jacques Moeschler’s diverse and visionary research over the years. The book contains three parts. The first part shows how different formal models can be applied to the analysis of such diverse problems as the syntax, semantics and pragmatics of tense, aspect and deictic expressions, syntax and pragmatics of quantifiers and semantics and pragmatics of connectives and negation. The second part presents the application of formal models to the treatment of cognitive issues related to the use of language, and in particular, demonstrating cognitive accounts of different types of human interactions, the context in utterance interpretation (salience, inferential comprehension processes), figurative uses of language (irony pretence), the role of syntax in Theory of Mind in autism and the analysis of the aesthetics of nature. Finally, the third part addresses computational and corpus-based approaches to natural language for investigating language variation, language universals and discourse related issues. This volume will be of great interest to syntacticians, pragmaticians, computer scientists, semanticists and psycholinguists.

A collection of articles by leading experts in theoretical computer science, this volume commemorates the 75th birthday of Professor Rani Siromoney, one of the pioneers in the field in India. The articles span the vast range of areas that Professor Siromoney has worked in or influenced, including grammar systems, picture languages and new models of computation.

This revised and expanded new edition elucidates the elegance and simplicity of the fundamental theory underlying formal languages and compilation. Retaining the reader-friendly style of the 1st edition, this versatile textbook describes the essential principles and methods used for defining the syntax of artificial languages, and for designing efficient parsing algorithms and syntax-directed translators with semantic attributes.

Features: presents a novel conceptual approach to parsing algorithms that applies to extended BNF grammars, together with a parallel parsing algorithm (NEW); supplies supplementary teaching tools at an associated website; systematically discusses ambiguous forms, allowing readers to avoid pitfalls; describes all algorithms in pseudocode; makes extensive usage of theoretical models of automata, transducers and formal grammars; includes concise coverage of algorithms for processing regular expressions and finite automata; introduces static program analysis based on flow equations.

The contributors present the main results and techniques of their specialties in an easily accessible way accompanied with many references: historical, hints for complete proofs or solutions to exercises and directions for further research. This volume contains applications which have not appeared in any collection of this type. The book is a general source of information in computation theory, at the undergraduate and research level.

This book is based on notes for a master’s course given at Queen Mary, University of London, in the 1998/9 session. Such courses in London are quite short, and the course consisted essentially of the material in the first three chapters, together with a two-hour lecture on connections with group theory. Chapter 5 is a considerably expanded version of this. For the course, the main sources were the books by Hopcroft and Ullman ([20]), by Cohen ([4]), and by Epstein et al. ([7]). Some use was also made of a later book by Hopcroft and Ullman ([21]). The ulterior motive in the first three chapters is to give a rigorous proof that various notions of recursively enumerable language are equivalent. Three such notions are considered. These are: generated by a type 0 grammar, recognised by a Turing machine (deterministic or not) and defined by means of a Godel numbering, having defined "recursively enumerable" for sets of natural numbers. It is hoped that this has been achieved without too many arguments using complicated notation. This is a problem with the entire subject, and it is important to understand the idea of the proof, which is often quite simple. Two particular places that are heavy going are the proof at the end of Chapter 1 that a language recognised by a Turing machine is type 0, and the proof in Chapter 2 that a Turing machine computable function is partial recursive.

This volume contains talks given at a joint meeting of three communities working in the fields of difference equations, special functions and applications (ISDE, OPSFA, and SIDE). The articles reflect the diversity of the topics in the meeting but have difference equations as common thread. Articles cover topics in difference equations, discrete dynamical systems, special functions, orthogonal polynomials, symmetries, and integrable difference equations.

[Automata Theory with Modern Applications](#)

[Volume 1: An Introduction to the Theory of Formal Languages and Automata, Volume 2: Applications in Linguistic Theory, Volume 3: Psycholinguistic Applications](#)

[Marcus Contextual Grammars](#)

[Fuzzy Automata and Languages](#)

[A Course in Formal Languages, Automata and Groups](#)

[Volume 1 Word, Language, Grammar](#)

[Automata, Computability and Complexity](#)

[A Collection of Papers in Honor of Sheng Yu](#)

[Combinatorial Applications of Formal Languages and Formal Series](#)

[Models and Their Applications](#)

Theory of Automata is designed to serve as a textbook for undergraduate students of B..E, B.Tech. CSE and MCA/IT. It attempts to help students grasp the essential concepts involved in automata theory.

Covers all areas, including operations on languages, context-sensitive languages, automata, decidability, syntax analysis, derivation languages, and more. Numerous worked examples, problem exercises, and elegant mathematical proofs. 1983 edition.

This book constitutes the refereed proceedings of the 13th International Conference on Language and Automata Theory and Applications, LATA 2019, held in St. Petersburg, Russia, in March 2019. The 31 revised full papers presented together with 5 invited talks were carefully reviewed and selected from 98 submissions. The papers cover the following topics: Automata; Complexity; Grammars; Languages; Graphs, trees and rewriting; and Words and codes.

This is a graduate-level introduction to formal methods. The first part presents two formal languages: logic, in various forms, and Communicating Sequential Process (CSP) as a process algebra. The second part offers specification and testing methods for formal development of software. Building on the foundations from the first part, the reader is allowed to embrace methods for practical applications. The reader will find the examples cutting across chapters valuable for this purpose. The final section takes the reader further into application domains. Models and simulations are an important first step in developing computer applications to solve real-world problems. However, in order to be truly effective, computer programmers must use formal modeling languages to evaluate these simulations. Formal Languages for Computer Simulation: Transdisciplinary Models and Applications investigates a variety of programming languages used in validating and verifying models in order to assist in their eventual implementation. This book will explore different methods of evaluating and formalizing simulation models, enabling computer and industrial engineers, mathematicians, and students working with computer simulations to thoroughly understand the progression from simulation to product, improving the overall effectiveness of modeling systems.

Recent applications to biomolecular science and DNA computing have created a new audience for automata theory and formal languages. This is the only introductory book to cover such applications. It begins with a clear and readily understood exposition of the fundamentals that assumes only a background in discrete mathematics. The first five chapters give a gentle but rigorous coverage of basic ideas as well as topics not found in other texts at this level, including codes, retracts and semiretracts. Chapter 6 introduces combinatorics on words and uses it to describe a visually inspired approach to languages. The final chapter explains recently-developed language theory coming from developments in bioscience and DNA computing. With over 350 exercises (for which solutions are available), many examples and illustrations, this text will make an ideal contemporary introduction for students; others, new to the field, will welcome it for self-learning.

Formal languages provide the theoretical underpinnings for the study of programming languages as well as the foundations for compiler design. They are important in such areas as data transmission and compression, computer networks, etc. This book combines an algebraic approach with algorithmic aspects and decidability results and explores applications both within computer science and in fields where formal languages are finding new applications such as molecular and developmental biology. It contains more than 600 graded exercises. While some are routine, many of the exercises are in reality supplementary material. Although the book has been designed as a text for graduate and upper-level undergraduate students, the comprehensive coverage of the subject makes it suitable as a reference for scientists.

[12th International Conference, LATA 2018, Ramat Gan, Israel, April 9-11, 2018, Proceedings](#)

[A Second Course in Formal Languages and Automata Theory](#)

[Perspectives and Open Problems](#)

[Recent Advances in Formal Languages and Applications](#)

[Applications in Interdisciplinary Contexts](#)

[Formal Language](#)

[New Developments in Formal Languages and Applications](#)

[Automata and Languages](#)

[Theory of Formal Languages with Applications](#)

[An Introduction to Formal Languages and Machine Computation](#)

Almost four decades have passed since Formal Grammars first appeared in 1974. At that time it was still possible to rather comprehensively review for (psycho)linguists the relevant literature on the theory of formal languages and automata, on their applications in linguistic theory and in the psychology of language. That is no longer feasible. In all three areas developments have been substantial, if not breathtaking. Nowadays, an interested linguist or psycholinguist opening any text on formal languages can no longer see the wood for the trees, as it is by no means evident which formal, mathematical tools are really required for natural language applications. An historical perspective can be helpful here. There are paths through the wood that have been beaten since decades; they can still provide useful orientation. The origins of these paths can be traced in the three volumes of Formal Grammars, brought together in the present re-edition. In a newly added postscript the author has sketched what has become, after all these years, of formal grammars in linguistics and psycholinguistics, or at least some of the core developments. This chapter may provide further motivation for the reader to make a trip back to some of the historical sources.

A step-by-step development of the theory of automata, languages and computation. Intended for use as the basis of an introductory course at both junior and senior levels, the text is organized so as to allow the design of various courses based on selected material. It features basic models of computation, formal languages and their properties; computability, decidability and complexity; a discussion of modern trends in the theory of automata and formal languages; design of programming languages, including the development of a new programming language; and compiler design, including the construction of a complete compiler. Alexander Meduna uses clear definitions, easy-to-follow proofs and helpful examples to make formerly obscure concepts easy to understand. He also includes challenging exercises and programming projects to enhance the reader’s comprehension, and many ‘real world’ illustrations and applications in practical computer science.

Formal languages, automata, computability, and related matters form the major part of the theory of computation. This textbook is designed for an introductory course for computer science and computer engineering majors who have knowledge of some higher-level programming language, the fundamentals of the theoretical underpinnings of computing form a standard part of almost every computer science curriculum. But the classic treatment of this material isolates it from the myriad ways in which the theory influences the design of modern hardware and software systems. The goal of this book is to change that. The book is organized into a core set of chapters (that cover the standard material suggested by the title), followed by a set of appendix chapters that highlight application areas including programming language design, compilers, software verification, networks, security, natural language processing, artificial intelligence, game playing, and computational biology. The core material includes discussions of finite state machines, Markov models, hidden Markov models (HMMs), regular expressions, context-free grammars, pushdown automata, Chomsky and Greibach normal forms, context-free parsing, pumping theorems for regular and context-free languages, closure theorems and decision procedures for regular and context-free languages, Turing machines, nondeterminism, decidability and undecidability, the Church-Turing thesis, reduction proofs, Post Correspondence problem, tiling problems, the undecidability of first-order logic, asymptotic dominance, time and space complexity, the Cook-Levin theorem, NP-completeness, Savitch’s Theorem, time and space hierarchy theorems, randomized algorithms and heuristic search. Throughout the discussion of these topics there are pointers into the application chapters. So, for example, the chapter that describes reduction proofs of undecidability has a link to the

security chapter, which shows a reduction proof of the undecidability of the safety of a simple protection framework.

This uniquely authoritative and comprehensive handbook is the first work to cover the vast field of formal languages, as well as their applications to the divergent areas of linguistics, developmental biology, computer graphics, cryptology, molecular genetics, and programming languages. The work has been divided into three volumes.

Business ethics has largely been written from the perspective of analytical philosophy with very little attention paid to the work of continental philosophers. Yet although very few of these philosophers directly discuss business ethics, it is clear that their ideas have interesting applications in this field. This innovative textbook shows how the work of continental philosophers — Deleuze and Guattari, Foucault, Levinas, Bauman, Derrida, Levinas, Nietzsche, Zizek, Jonas, Sartre, Heidegger, Latour, Nancy and Sloterdijk — can provide fresh insights into a number of different issues in business ethics. Topics covered include agency, stakeholder theory, organizational culture, organizational justice, moral decision-making, leadership, whistle-blowing, corporate social responsibility, globalization and sustainability. The book includes a number of features designed to aid comprehension, including a detailed glossary of key terms, text boxes explaining key concepts, and a wide range of examples from the world of business.

Formal Language Theory: Perspectives and Open Problems focuses on the trends and major open problems on the formal language theory. The selection first ponders on the methods for specifying families of formal languages, open problems about regular languages, and generators of cones and cylinders. Discussions focus on cylinders of algebraic languages, cone of algebraic languages, regularity of noncounting classes, group complexity, specification formalism, and grammars. The publication then elaborates on very small families of algebraic nonrational languages and formal languages and their relation to automata. The book tackles morphisms on free monoids and language theory, homomorphisms, and survey of results and open problems in the mathematical theory of L systems. Topics include single finite substitutions iterated, single homomorphisms iterated, representation of language families, homomorphism equivalence on a language, and problems about infinite words. The selection is a valuable source of data for researchers interested in the formal language theory.

[Introduction to Switching and Automata Theory](#)

[Formal Language Theory](#)

[An Introduction to Formal Languages and Automata](#)

[Formal Grammars in Linguistics and Psycholinguistics](#)

[Formal Languages and Computation](#)

[Formal Languages and Applications](#)

[Second International Conference, LATA 2008, Tarragona, Spain, March 13-19, 2008, Revised Papers](#)

[Volume 1, Word, Language, Grammar](#)

[Formal Models, Languages and Applications](#)

[Handbook of Formal Languages](#)

This classroom-tested and clearly-written textbook presents a focused guide to the conceptual foundations of compilation, explaining the fundamental principles and algorithms used for defining the syntax of languages, and for implementing simple translators. This significantly updated and expanded third edition has been enhanced with additional coverage of regular expressions, visibly pushdown languages, bottom-up and top-down deterministic parsing algorithms, and new grammar models. Topics and features: describes the principles and methods used in designing syntax-directed applications such as parsing and regular expression matching; covers translations, semantic functions (attribute grammars), and static program analysis by data flow equations; introduces an efficient method for string matching and parsing suitable for ambiguous regular expressions (NEW); presents a focus on extended BNF grammars with their general parser and with LR(1) and LL(1) parsers (NEW); introduces a parallel parsing algorithm that exploits multiple processing threads to speed up syntax analysis of large files; discusses recent formal models of input-driven automata and languages (NEW); includes extensive use of theoretical models of automata, transducers and formal grammars, and describes all algorithms in pseudocode; contains numerous illustrative examples, and supplies a large set of exercises with solutions at an associated website. Advanced undergraduate and graduate students of computer science will find this reader-friendly textbook to be an invaluable guide to the essential concepts of syntax-directed compilation. The fundamental paradigms of language structures are elegantly explained in terms of the underlying theory, without requiring the use of software tools or knowledge of implementation, and through algorithms simple enough to be practiced by paper and pencil.

Formal Languages and Computation: Models and Their Applications gives a clear, comprehensive introduction to formal language theory and its applications in computer science. It covers all rudimentary topics concerning formal languages and their models, especially grammars and automata, and sketches the basic ideas underlying the theory of computation, including computability, decidability, and computational complexity. Emphasizing the relationship between theory and application, the book describes many real-world applications, including computer science engineering techniques for language processing and their implementation. Covers the theory of formal languages and their models, including all essential concepts and properties Explains how language models underlie language processors Pays a special attention to programming language analyzers, such as scanners and parsers, based on four language models—regular expressions, finite automata, context-free grammars, and pushdown automata Discusses the mathematical notion of a Turing machine as a universally accepted formalization of the intuitive notion of a procedure Reviews the general theory of computation, particularly computability and decidability Considers problem-deciding algorithms in terms of their computational complexity measured according to time and space requirements Points out that some problems are decidable in principle, but they are, in fact, intractable problems for absurdly high computational requirements of the algorithms that decide them In short, this book represents a theoretically oriented treatment of formal languages and their models with a focus on their applications. It introduces all formalisms concerning them with enough rigors to make all results quite clear and valid. Every complicated mathematical passage is preceded by its intuitive explanation so that even the most complex parts of the book are easy to grasp. After studying this book, both student and professional should be able to understand the fundamental theory of formal languages and computation, write language processors, and confidently follow most advanced books on the subject.

This book constitutes the refereed proceedings of the Second International Conference on Language and Automata Theory and Applications, LATA 2008, held in Tarragona, Spain, in March 2008. The 40 revised full papers presented were carefully reviewed and selected from 134 submissions. The papers deal with the various issues related to automata theory and formal languages

Marcus Contextual Grammars is the first monograph to present a class of grammars introduced about three decades ago, based on the fundamental linguistic phenomenon of strings-contexts interplay (selection). Most of the theoretical results obtained so far about the many variants of contextual grammars are presented with emphasis on classes of questions with relevance for applications in the study of natural language syntax: generative powers, descriptive and computational complexity, automata recognition, semilinearity, structure of the generated strings, ambiguity, regulated rewriting, etc. Constant comparison with families of languages in the Chomsky hierarchy is made. Connections with non-linguistic areas are established, such as molecular computing. Audience: Researchers and students in theoretical computer science (formal language theory and automata theory), computational linguistics, mathematical methods in linguistics, and linguists interested in formal models of syntax.

The theory of finite automata on finite strings, infinite strings, and trees has had a distinguished history. First, automata were introduced to represent idealized switching circuits augmented by unit delays. This was the period of Shannon, McCulloch and Pitts, and Howard Aiken, ending about 1950. Then in the 1950s there was the work of Kleene on representable events, of Myhill and Nerode on finite coset congruence relations on strings, of Rabin and Scott on power set automata. In the 1960s, there was the work of Bichi on automata on infinite strings and the second order theory of one successor, then Rabin's 1968 result on automata on infinite trees and the second order theory of two successors. The latter was a mystery until the introduction of forgetful determinacy games by Gurevich and Harrington in 1982. Each of these developments has successful and prospective applications in computer science. They should all be part of every computer scientist's toolbox. Suppose that we take a computer scientist's point of view. One can think of finite automata as the mathematical representation of programs that run using fixed finite resources. Then Bichi's SIS can be thought of as a theory of programs which run forever (like operating systems or banking systems) and are deterministic. Finally, Rabin's S2S is a theory of programs which run forever and are nondeterministic. Indeed many questions of verification can be decided in the decidable theories of these automata.

This book contains original reviews by well-known workers in the field of mathematical linguistics and formal language theory, written in honour of Professor Solomon Marcus on the occasion of his 70th birthday. Some of the papers deal with contextual grammars, a class of generative devices introduced by Marcus, motivated by descriptive linguistics. Others are devoted to grammar systems, a very modern branch of formal language theory. Automata theory and the algebraic approach to computer science are other well-represented areas. While the contributions are mathematically oriented, practical issues such as cryptography, grammatical inference and natural language processing are also discussed. Contents: Substitutions on Words and Languages Applications to Cryptography (A Atanasiu) Grammar Systems: A Multi-Agent Framework for Natural Language Generation (E Csuhaj-Varjú) Normal Forms for Contextual Grammars (A Ehrenfeucht et al.) Control Mechanisms on #-Context-Free Array Grammars (R Freund) On Transitive Cofinal Automata (M Ito & M Katsura) Algebraic Foundations for Montague Grammars (H Jürgensen & K Tent) A Periodic Languages and Generalizations (J Kari & G Thierrin) Matrix Grammars Versus Parallel Communicating Grammar Systems (V Mihalache) Reducts Versus Reducing Operators (M Novotný) On Conditional Grammars and Conditional Petri Nets (F-L Tiplea) and other papers Readership: Computer scientists. keywords: Algebra; Array Grammar; Automaton; Chomsky Grammar; Combinatorics on Words; Cryptography; Grammar System; Marcus Grammar; Mereology; Montague Grammar; Natural Language; Petri Net

[Mathematical Aspects of Natural and Formal Languages](#)

[Language and Automata Theory and Applications](#)

[Semigroups: Algebraic Theory And Applications To Formal Languages And Codes](#)

[Automata Theory and its Applications](#)

[4th International Conference, LATA 2010, Trier, Germany, May 24-28, 2010, Proceedings](#)

[Formal Languages and Compilation](#)

[Languages, Methods, Application Domains](#)