

The Nature Of Computation Book Library

The boundary between physics and computer science has become a hotbed of interdisciplinary collaboration. In this book the authors introduce the reader to the fundamental concepts of computational complexity and give in-depth explorations of the major interfaces between computer science and physics.

Major text/reference work on computer modeling for students and researchers in any quantitative or semi-quantitative discipline, first published in 1998.

The Nature of Computation Oxford University Press

First-ever comprehensive introduction to the major new subject of quantum computing and quantum information.

Presenting a theory of the theoryless, a computer scientist provides a model of how effective behavior can be learned even in a world as complex as our own, shedding new light on human nature.

The classical theory of computation has its origins in the work of Goedel, Turing, Church, and Kleene and has been an extraordinarily successful framework for theoretical computer science. The thesis of this book, however, is that it provides an inadequate foundation for modern scientific computation where most of the algorithms are real number algorithms. The goal of this book is to develop a formal theory of computation which integrates major themes of the classical theory and which is more

directly applicable to problems in mathematics, numerical analysis, and scientific computing. Along the way, the authors consider such fundamental problems as: * Is the Mandelbrot set decidable? * For simple quadratic maps, is the Julia set a halting set? * What is the real complexity of Newton's method? * Is there an algorithm for deciding the knapsack problem in a polynomial number of steps? * Is the Hilbert Nullstellensatz intractable? * Is the problem of locating a real zero of a degree four polynomial intractable? * Is linear programming tractable over the reals? The book is divided into three parts: The first part provides an extensive introduction and then proves the fundamental NP-completeness theorems of Cook-Karp and their extensions to more general number fields as the real and complex numbers. The later parts of the book develop a formal theory of computation which integrates major themes of the classical theory and which is more directly applicable to problems in mathematics, numerical analysis, and scientific computing.

Computational complexity is one of the most beautiful fields of modern mathematics, and it is increasingly relevant to other sciences ranging from physics to biology. But this beauty is often buried underneath layers of unnecessary formalism, and exciting recent results like interactive proofs, phase transitions, and quantum computing are usually considered too advanced for the typical student. This book bridges these gaps by explaining the deep ideas of theoretical computer science in a clear and enjoyable fashion, making them accessible to non-computer scientists and to computer scientists

who finally want to appreciate their field from a new point of view. The authors start with a lucid and playful explanation of the P vs. NP problem, explaining why it is so fundamental, and so hard to resolve. They then lead the reader through the complexity of mazes and games; optimization in theory and practice; randomized algorithms, interactive proofs, and pseudorandomness; Markov chains and phase transitions; and the outer reaches of quantum computing. At every turn, they use a minimum of formalism, providing explanations that are both deep and accessible. The book is intended for graduate and undergraduate students, scientists from other areas who have long wanted to understand this subject, and experts who want to fall in love with this field all over again.

This book addresses the frontier advances in the theory and application of nature-inspired optimization techniques, including solving the quadratic assignment problem, prediction in nature-inspired dynamic optimization, the lion algorithm and its applications, optimizing the operation scheduling of microgrids, PID controllers for two-legged robots, optimizing crane operating times, planning electrical energy distribution systems, automatic design and evaluation of classification pipelines, and optimizing wind-energy power generation plants. The book also presents a variety of nature-inspired methods and illustrates methods of adapting these to said applications. Nature-inspired computation, developed by mimicking natural phenomena, makes a significant contribution toward the solution of non-convex optimization problems that normal

mathematical optimizers fail to solve. As such, a wide range of nature-inspired computing approaches has been used in multidisciplinary engineering applications. Written by researchers and developers from a variety of fields, this book presents the latest findings, novel techniques and pioneering applications.

Neural network research often builds on the fiction that neurons are simple linear threshold units, completely neglecting the highly dynamic and complex nature of synapses, dendrites, and voltage-dependent ionic currents. *Biophysics of Computation: Information Processing in Single Neurons* challenges this notion, using richly detailed experimental and theoretical findings from cellular biophysics to explain the repertoire of computational functions available to single neurons. The author shows how individual nerve cells can multiply, integrate, or delay synaptic inputs and how information can be encoded in the voltage across the membrane, in the intracellular calcium concentration, or in the timing of individual spikes. Key topics covered include the linear cable equation; cable theory as applied to passive dendritic trees and dendritic spines; chemical and electrical synapses and how to treat them from a computational point of view; nonlinear interactions of synaptic input in passive and active dendritic trees; the Hodgkin-Huxley model of action potential generation and propagation; phase space analysis; linking stochastic ionic channels to membrane-dependent currents;

calcium and potassium currents and their role in information processing; the role of diffusion, buffering and binding of calcium, and other messenger systems in information processing and storage; short- and long-term models of synaptic plasticity; simplified models of single cells; stochastic aspects of neuronal firing; the nature of the neuronal code; and unconventional models of sub-cellular computation. *Biophysics of Computation: Information Processing in Single Neurons* serves as an ideal text for advanced undergraduate and graduate courses in cellular biophysics, computational neuroscience, and neural networks, and will appeal to students and professionals in neuroscience, electrical and computer engineering, and physics.

Although we are entirely unaware of it, computation is central to all aspects of our existences. Every day we solve, or try to solve, a myriad of problems, from the utterly trivial to the bafflingly complex. This book explains why it is possible to do computation and what the ultimate limits of it are, as understood by modern science.

This book is based on the outcome of the “2012 Interdisciplinary Symposium on Complex Systems” held at the island of Kos. The book consists of 12 selected papers of the symposium starting with a comprehensive overview and classification of complexity problems, continuing by chapters about complexity,

its observation, modeling and its applications to solving various problems including real-life applications. More exactly, readers will have an encounter with the structural complexity of vortex flows, the use of chaotic dynamics within evolutionary algorithms, complexity in synthetic biology, types of complexity hidden inside evolutionary dynamics and possible controlling methods, complexity of rugged landscapes, and more. All selected papers represent innovative ideas, philosophical overviews and state-of-the-art discussions on aspects of complexity. The book will be useful as instructional material for senior undergraduate and entry-level graduate students in computer science, physics, applied mathematics and engineering-type work in the area of complexity. The book will also be valuable as a resource of knowledge for practitioners who want to apply complexity to solve real-life problems in their own challenging applications. The authors and editors hope that readers will be inspired to do their own experiments and simulations, based on information reported in this book, thereby moving beyond the scope of the book.

An introduction to computational complexity theory, its connections and interactions with mathematics, and its central role in the natural and social sciences, technology, and philosophy Mathematics and Computation provides a broad, conceptual overview of computational complexity theory—the mathematical

study of efficient computation. With important practical applications to computer science and industry, computational complexity theory has evolved into a highly interdisciplinary field, with strong links to most mathematical areas and to a growing number of scientific endeavors. Avi Wigderson takes a sweeping survey of complexity theory, emphasizing the field's insights and challenges. He explains the ideas and motivations leading to key models, notions, and results. In particular, he looks at algorithms and complexity, computations and proofs, randomness and interaction, quantum and arithmetic computation, and cryptography and learning, all as parts of a cohesive whole with numerous cross-influences. Wigderson illustrates the immense breadth of the field, its beauty and richness, and its diverse and growing interactions with other areas of mathematics. He ends with a comprehensive look at the theory of computation, its methodology and aspirations, and the unique and fundamental ways in which it has shaped and will further shape science, technology, and society. For further reading, an extensive bibliography is provided for all topics covered. Mathematics and Computation is useful for undergraduate and graduate students in mathematics, computer science, and related fields, as well as researchers and teachers in these fields. Many parts require little background, and serve as an invitation to newcomers seeking an introduction to the theory of computation.

Comprehensive coverage of computational complexity theory, and beyond High-level, intuitive exposition, which brings conceptual clarity to this central and dynamic scientific discipline Historical accounts of the evolution and motivations of central concepts and models A broad view of the theory of computation's influence on science, technology, and society Extensive bibliography

This text addresses some theoretical issues surrounding computer science. It provides an introduction to the theory of computation, and covers programming languages, finite state machines, grammars, Boolean circuits, computational complexity, feasible problems, and intractable problems.

Takes students and researchers on a tour through some of the deepest ideas of maths, computer science and physics.

An accessible and rigorous textbook for introducing undergraduates to computer science theory *What Can Be Computed?* is a uniquely accessible yet rigorous introduction to the most profound ideas at the heart of computer science. Crafted specifically for undergraduates who are studying the subject for the first time, and requiring minimal prerequisites, the book focuses on the essential fundamentals of computer science theory and features a practical approach that uses real computer programs (Python and Java) and encourages active experimentation. It is also ideal for self-study and reference. The book covers the standard topics in

the theory of computation, including Turing machines and finite automata, universal computation, nondeterminism, Turing and Karp reductions, undecidability, time-complexity classes such as P and NP, and NP-completeness, including the Cook-Levin Theorem. But the book also provides a broader view of computer science and its historical development, with discussions of Turing's original 1936 computing machines, the connections between undecidability and Gödel's incompleteness theorem, and Karp's famous set of twenty-one NP-complete problems. Throughout, the book recasts traditional computer science concepts by considering how computer programs are used to solve real problems. Standard theorems are stated and proven with full mathematical rigor, but motivation and understanding are enhanced by considering concrete implementations. The book's examples and other content allow readers to view demonstrations of—and to experiment with—a wide selection of the topics it covers. The result is an ideal text for an introduction to the theory of computation. An accessible and rigorous introduction to the essential fundamentals of computer science theory, written specifically for undergraduates taking introduction to the theory of computation Features a practical, interactive approach using real computer programs (Python in the text, with forthcoming Java alternatives online) to enhance motivation and understanding Gives equal

emphasis to computability and complexity Includes special topics that demonstrate the profound nature of key ideas in the theory of computation Lecture slides and Python programs are available at whatcanbecomputed.com "Shows how to recognize NP-complete problems and offers practical suggestions for dealing with them effectively. The book covers the basic theory of NP-completeness, provides an overview of alternative directions for further research, and contains an extensive list of NP-complete and NP-hard problems, with more than 300 main entries and several times as many results in total. [This book] is suitable as a supplement to courses in algorithm design, computational complexity, operations research, or combinatorial mathematics, and as a text for seminars on approximation algorithms or computational complexity. It provides not only a valuable source of information for students but also an essential reference work for professionals in computer science"--Back cover.

The foundation of computer science is built upon the following questions: What is an algorithm? What can be computed and what cannot be computed? What does it mean for a function to be computable? How does computational power depend upon programming constructs? Which algorithms can be considered feasible? For more than 70 years, computer scientists are searching for answers to such questions. Their ingenious techniques used in answering these questions form the theory of

computation. Theory of computation deals with the most fundamental ideas of computer science in an abstract but easily understood form. The notions and techniques employed are widely spread across various topics and are found in almost every branch of computer science. It has thus become more than a necessity to revisit the foundation, learn the techniques, and apply them with confidence. Overview and Goals This book is about this solid, beautiful, and pervasive foundation of computer science. It introduces the fundamental notions, models, techniques, and results that form the basic paradigms of computing. It gives an introduction to the concepts and mathematics that computer scientists of our day use to model, to argue about, and to predict the behavior of algorithms and computation. The topics chosen here have shown remarkable persistence over the years and are very much in current use.

This book provides a comprehensive introduction to the computational material that forms the underpinnings of the currently evolving set of brain models. It is now clear that the brain is unlikely to be understood without recourse to computational theories. The theme of *An Introduction to Natural Computation* is that ideas from diverse areas such as neuroscience, information theory, and optimization theory have recently been extended in ways that make them useful for describing the brain's programs. This book provides a comprehensive introduction to the computational material that forms the underpinnings of the currently evolving set of brain models. It stresses the broad spectrum of learning models—ranging from neural network learning through

reinforcement learning to genetic learning—and situates the various models in their appropriate neural context. To write about models of the brain before the brain is fully understood is a delicate matter. Very detailed models of the neural circuitry risk losing track of the task the brain is trying to solve. At the other extreme, models that represent cognitive constructs can be so abstract that they lose all relationship to neurobiology. An Introduction to Natural Computation takes the middle ground and stresses the computational task while staying near the neurobiology.

Engineering systems operate through actuators, most of which will exhibit phenomena such as saturation or zones of no operation, commonly known as dead zones. These are examples of piecewise-affine characteristics, and they can have a considerable impact on the stability and performance of engineering systems. This book targets controller design for piecewise affine systems, fulfilling both stability and performance requirements. The authors present a unified computational methodology for the analysis and synthesis of piecewise affine controllers, taking an approach that is capable of handling sliding modes, sampled-data, and networked systems. They introduce algorithms that will be applicable to nonlinear systems approximated by piecewise affine systems, and they feature several examples from areas such as switching electronic circuits, autonomous vehicles, neural networks, and aerospace applications. Piecewise Affine Control: Continuous-Time, Sampled-Data, and Networked Systems is intended for graduate students, advanced senior undergraduate

students, and researchers in academia and industry. It is also appropriate for engineers working on applications where switched linear and affine models are important. Nature-inspired computation and swarm intelligence have become popular and effective tools for solving problems in optimization, computational intelligence, soft computing and data science. Recently, the literature in the field has expanded rapidly, with new algorithms and applications emerging. Nature-Inspired Computation and Swarm Intelligence: Algorithms, Theory and Applications is a timely reference giving a comprehensive review of relevant state-of-the-art developments in algorithms, theory and applications of nature-inspired algorithms and swarm intelligence. It reviews and documents the new developments, focusing on nature-inspired algorithms and their theoretical analysis, as well as providing a guide to their implementation. The book includes case studies of diverse real-world applications, balancing explanation of the theory with practical implementation. Nature-Inspired Computation and Swarm Intelligence: Algorithms, Theory and Applications is suitable for researchers and graduate students in computer science, engineering, data science, and management science, who want a comprehensive review of algorithms, theory and implementation within the fields of nature inspired computation and swarm intelligence. Introduces nature-inspired algorithms and their fundamentals, including: particle swarm optimization, bat algorithm, cuckoo search, firefly algorithm, flower pollination algorithm, differential evolution and genetic algorithms as well as multi-objective optimization

algorithms and others Provides a theoretical foundation and analyses of algorithms, including: statistical theory and Markov chain theory on the convergence and stability of algorithms, dynamical system theory, benchmarking of optimization, no-free-lunch theorems, and a generalized mathematical framework Includes a diversity of case studies of real-world applications: feature selection, clustering and classification, tuning of restricted Boltzmann machines, travelling salesman problem, classification of white blood cells, music generation by artificial intelligence, swarm robots, neural networks, engineering designs and others

Brain and Nature-Inspired Learning, Computation and Recognition presents a systematic analysis of neural networks, natural computing, machine learning and compression, algorithms and applications inspired by the brain and biological mechanisms found in nature. Sections cover new developments and main applications, algorithms and simulations. Developments in brain and nature-inspired learning have promoted interest in image processing, clustering problems, change detection, control theory and other disciplines. The book discusses the main problems and applications pertaining to bio-inspired computation and recognition, introducing algorithm implementation, model simulation, and practical application of parameter setting. Readers will find solutions to problems in computation and recognition, particularly neural networks, natural computing, machine learning and compressed sensing. This volume offers a comprehensive and well-structured introduction to brain and nature-

inspired learning, computation, and recognition. Presents an invaluable systematic introduction to brain and nature-inspired learning, computation and recognition Describes the biological mechanisms, mathematical analyses and scientific principles behind brain and nature-inspired learning, calculation and recognition Systematically analyzes neural networks, natural computing, machine learning and compression, algorithms and applications inspired by the brain and biological mechanisms found in nature Discusses the theory and application of algorithms and neural networks, natural computing, machine learning and compression perception

The abstract branch of theoretical computer science known as Computation Theory typically appears in undergraduate academic curricula in a form that obscures both the mathematical concepts that are central to the various components of the theory and the relevance of the theory to the typical student. This regrettable situation is due largely to the thematic tension among three main competing principles for organizing the material in the course. This book is motivated by the belief that a deep understanding of, and operational control over, the few "big" mathematical ideas that underlie Computation Theory is the best way to enable the typical student to assimilate the "big" ideas of Computation Theory into her daily computational life.

This book presents a study of digital computation in contemporary cognitive science. Digital computation is a highly ambiguous concept, as there is no common core definition for it in cognitive science. Since this concept plays a central role in cognitive

theory, an adequate cognitive explanation requires an explicit account of digital computation. More specifically, it requires an account of how digital computation is implemented in physical systems. The main challenge is to deliver an account encompassing the multiple types of existing models of computation without ending up in pancomputationalism, that is, the view that every physical system is a digital computing system. This book shows that only two accounts, among the ones examined by the author, are adequate for explaining physical computation. One of them is the instructional information processing account, which is developed here for the first time. "This book provides a thorough and timely analysis of differing accounts of computation while advancing the important role that information plays in understanding computation. Fresco's two-pronged approach will appeal to philosophically inclined computer scientists who want to better understand common theoretical claims in cognitive science." Marty J. Wolf, Professor of Computer Science, Bemidji State University "An original and admirably clear discussion of central issues in the foundations of contemporary cognitive science." Frances Egan, Professor of Philosophy, Rutgers, The State University of New Jersey

The study of the genetic basis for evolution has flourished in this century, as well as our understanding of the evolvability and programmability of biological systems. Genetic algorithms meanwhile grew out of the realization that a computer program could use the biologically-inspired processes of mutation, recombination, and selection to solve

hard optimization problems. Genetic and evolutionary programming provide further approaches to a wide variety of computational problems. A synthesis of these experiences reveals fundamental insights into both the computational nature of biological evolution and processes of importance to computer science. Topics include biological models of nucleic acid information processing and genome evolution; molecules, cells, and metabolic circuits that compute logical relationships; the origin and evolution of the genetic code; and the interface with genetic algorithms and genetic and evolutionary programming.

With the objective of making into a science the art of verifying computer programs (debugging), the author addresses both practical and theoretical aspects. Subjects include computability (with discussions of finite automata and Turing machines); predicate calculus; verification of programs (both flowchart and algol-like programs); flowchart schemas; and the fixpoint theory of programs. 1974 edition. Includes 77 figures.

Computation permeates our world, but a satisfactory philosophical theory of what it is has been lacking. Gualtiero Piccinini presents a mechanistic account of what makes a physical system a computing system. He argues that computation does not entail representation or information-processing, although information-processing entails computation.

This book constitutes the refereed proceedings of the 9th Conference on Computability in Europe, CiE 2013, held in Milan, Italy, in July 2013. The 48 revised papers presented together with 1 invited lecture and 2 tutorials were carefully reviewed and selected with an acceptance rate of under 31,7%. Both the conference series and the association promote the development of computability-related science, ranging over mathematics, computer science and applications in various natural and engineering sciences such as physics and biology, and also including the promotion of related non-scientific fields such as philosophy and history of computing.

Combining physics, mathematics and computer science, topological quantum computation is a rapidly expanding research area focused on the exploration of quantum evolutions that are immune to errors. In this book, the author presents a variety of different topics developed together for the first time, forming an excellent introduction to topological quantum computation. The makings of anyonic systems, their properties and their computational power are presented in a pedagogical way. Relevant calculations are fully explained, and numerous worked examples and exercises support and aid understanding. Special emphasis is given to the motivation and physical intuition behind every mathematical concept. Demystifying difficult topics by using accessible language,

this book has broad appeal and is ideal for graduate students and researchers from various disciplines who want to get into this new and exciting research field. New and classical results in computational complexity, including interactive proofs, PCP, derandomization, and quantum computation. Ideal for graduate students.

An accessible introduction to an exciting new area in computation, explaining such topics as qubits, entanglement, and quantum teleportation for the general reader. Quantum computing is a beautiful fusion of quantum physics and computer science, incorporating some of the most stunning ideas from twentieth-century physics into an entirely new way of thinking about computation. In this book, Chris Bernhardt offers an introduction to quantum computing that is accessible to anyone who is comfortable with high school mathematics. He explains qubits, entanglement, quantum teleportation, quantum algorithms, and other quantum-related topics as clearly as possible for the general reader. Bernhardt, a mathematician himself, simplifies the mathematics as much as he can and provides elementary examples that illustrate both how the math works and what it means. Bernhardt introduces the basic unit of quantum computing, the qubit, and explains how the qubit can be measured; discusses entanglement—which, he says, is easier to describe mathematically than

verbally—and what it means when two qubits are entangled (citing Einstein's characterization of what happens when the measurement of one entangled qubit affects the second as “spooky action at a distance”); and introduces quantum cryptography. He recaps standard topics in classical computing—bits, gates, and logic—and describes Edward Fredkin's ingenious billiard ball computer. He defines quantum gates, considers the speed of quantum algorithms, and describes the building of quantum computers. By the end of the book, readers understand that quantum computing and classical computing are not two distinct disciplines, and that quantum computing is the fundamental form of computing. The basic unit of computation is the qubit, not the bit.

This book presents in their basic form the most important models of computation, their basic programming paradigms, and their mathematical descriptions, both concrete and abstract. Each model is accompanied by relevant formal techniques for reasoning on it and for proving some properties. After preliminary chapters that introduce the notions of structure and meaning, semantic methods, inference rules, and logic programming, the authors arrange their chapters into parts on IMP, a simple imperative language; HOFL, a higher-order functional language; concurrent, nondeterministic and interactive models; and probabilistic/stochastic models. The authors have class-tested the book content over many years, and it

will be valuable for graduate and advanced undergraduate students of theoretical computer science and distributed systems, and for researchers in this domain. Each chapter of the book concludes with a list of exercises addressing the key techniques introduced, solutions to selected exercises are offered at the end of the book.

This book constitutes the proceedings of the 17th International Conference on Unconventional Computation and Natural Computation, UCNC 2018, held in Fontainebleau, France, in June 2018. The 15 full papers presented were carefully reviewed and selected from 22 submissions. The paper cover topics such as hypercomputation; chaos and dynamical systems based computing; granular, fuzzy and rough computing; mechanical computing; cellular, evolutionary, molecular, neural, and quantum computing; membrane computing; amorphous computing, swarm intelligence; artificial immune systems; physics of computation; chemical computation; evolving hardware; the computational nature of self-assembly, developmental processes, bacterial communication, and brain processes.

How can we capture the unpredictable evolutionary and emergent properties of nature in software? How can understanding the mathematical principles behind our physical world help us to create digital worlds? This book focuses on a range of programming strategies and

techniques behind computer simulations of natural systems, from elementary concepts in mathematics and physics to more advanced algorithms that enable sophisticated visual results. Readers will progress from building a basic physics engine to creating intelligent moving objects and complex systems, setting the foundation for further experiments in generative design. Subjects covered include forces, trigonometry, fractals, cellular automata, self-organization, and genetic algorithms. The book's examples are written in Processing, an open-source language and development environment built on top of the Java programming language. On the book's website (<http://www.natureofcode.com>), the examples run in the browser via Processing's JavaScript mode.

This book discusses all the major nature-inspired algorithms with a focus on their application in the context of solving navigation and routing problems. It also reviews the approximation methods and recent nature-inspired approaches for practical navigation, and compares these methods with traditional algorithms to validate the approach for the case studies discussed. Further, it examines the design of alternative solutions using nature-inspired techniques, and explores the challenges of navigation and routing problems and nature-inspired metaheuristic approaches.

Now you can clearly present even the most complex computational theory topics to your students with Sipser's distinct, market-leading INTRODUCTION TO THE THEORY OF COMPUTATION, 3E. The number one choice for today's computational theory course, this highly anticipated revision retains the unmatched clarity and thorough coverage that make it a leading text for upper-level undergraduate and introductory graduate students. This edition continues author Michael Sipser's well-known, approachable style with timely revisions,

additional exercises, and more memorable examples in key areas. A new first-of-its-kind theoretical treatment of deterministic context-free languages is ideal for a better understanding of parsing and LR(k) grammars. This edition's refined presentation ensures a trusted accuracy and clarity that make the challenging study of computational theory accessible and intuitive to students while maintaining the subject's rigor and formalism. Readers gain a solid understanding of the fundamental mathematical properties of computer hardware, software, and applications with a blend of practical and philosophical coverage and mathematical treatments, including advanced theorems and proofs. INTRODUCTION TO THE THEORY OF COMPUTATION, 3E's comprehensive coverage makes this an ideal ongoing reference tool for those studying theoretical computing. Important Notice: Media content referenced within the product description or the product text may not be available in the ebook version.

What does it mean to say that an object or system computes? What is it about laptops, smartphones, and nervous systems that they are considered to compute, and why does it seldom occur to us to describe stomachs, hurricanes, rocks, or chairs that way? Though computing systems are everywhere today, it is very difficult to answer these questions. The book aims to shed light on the subject by arguing for the semantic view of computation, which states that computingsystems are always accompanied by representations. This view is presented as an alternative to non-semantic views such as the mechanistic account of computation.

This book is about nature considered as the totality of physical existence, the universe, and our present day attempts to understand it. If we see the universe as a network of networks of computational processes at many different levels of organization, what can we learn about

physics, biology, cognition, social systems, and ecology expressed through interacting networks of elementary particles, atoms, molecules, cells, (and especially neurons when it comes to understanding of cognition and intelligence), organs, organisms and their ecologies? Regarding our computational models of natural phenomena Feynman famously wondered: “Why should it take an infinite amount of logic to figure out what one tiny piece of space/time is going to do?” Phenomena themselves occur so quickly and automatically in nature. Can we learn how to harness nature’s computational power as we harness its energy and materials? This volume includes a selection of contributions from the Symposium on Natural Computing/Unconventional Computing and Its Philosophical Significance, organized during the AISB/IACAP World Congress 2012, held in Birmingham, UK, on July 2-6, on the occasion of the centenary of Alan Turing’s birth. In this book, leading researchers investigated questions of computing nature by exploring various facets of computation as we find it in nature: relationships between different levels of computation, cognition with learning and intelligence, mathematical background, relationships to classical Turing computation and Turing’s ideas about computing nature - unorganized machines and morphogenesis. It addresses questions of information, representation and computation, interaction as communication, concurrency and agent models; in short this book presents natural computing and unconventional computing as extension of the idea of computation as symbol manipulation.

[Copyright: f0c76b15f90a32328eacd819879d80a2](https://www.researchgate.net/publication/312222222)