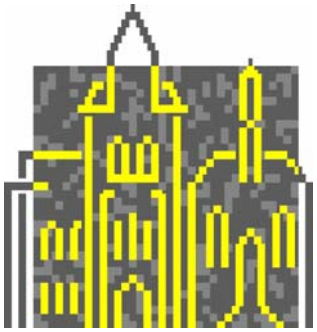

SCHLOSS DAGSTUHL

INTERNATIONAL
CONFERENCE
AND RESEARCH CENTER
FOR COMPUTER SCIENCE

Dagstuhl News

January - December 2002

Volume 5
2003

ISSN 1438-7581

Period: January - December 2002

Frequency: 1 per year

The International Conference and Research Center for Computer Science is operated by a non-profit organization. Its objective is to promote world-class research in computer science and to host research seminars which enable new ideas to be showcased, problems to be discussed and the course to be set for future development in this field.

Associates: Gesellschaft für Informatik e.V., Bonn
Technische Universität Darmstadt
Universität Frankfurt
Universität Kaiserslautern
Universität Karlsruhe
Universität Stuttgart
Universität Trier
Universität des Saarlandes

Funding: The state governments of Saarland and Rhineland Palatinate

Address: IBFI Schloß Dagstuhl
Octavieallee
D-66687 Wadern
Tel.: +49 - 6871 - 905127
Fax: +49 - 6871 - 905130
E-mail: service@dagstuhl.de
Internet: http://www.dagstuhl.de/

Welcome

You have in your hands the fifth edition of the "Dagstuhl News", a publication for the members of the Foundation "Informatikzentrum Schloss Dagstuhl", the *Dagstuhl Foundation* for short. As always, we are a bit late, which as always has its reasons in the fact that Dagstuhl (and our other obligations) keeps us busy.

The main part of this leaflet consists of collected resumees and other hopefully interesting information excerpt from the Dagstuhl-Seminar Reports. We hope that you will find this information valuable for your own work or informative as to what colleagues in other research areas of Computer Science are doing. The full reports for 2002 are on the Web under URL: `http://www.dagstuhl.de/Seminars/02/` Several things related to the Dagstuhl News have changed. First, the layout. We were told by the Evaluation Committee and by our Curatory Board that our public relations should be improved. Well, we have not really put much emphasis here, I must confess. Actually, at the end of the hearing by the evaluation committee the chairman stated that it was touching to evaluate an institution which neglected public relations in favour of the proper work.

Biggest (positive) news last year was the decision by the Federal-State Commission (Bund-Länder Kommission) that Dagstuhl should move onto the *Blue List* of research institutions with combined federal and state funding. This will secure Dagstuhl's financial support for quite a while.

One of the other changes was that we switched to publishing online proceedings of our Dagstuhl Seminars instead of the old Seminar Reports. Authors keep the copyrights to their contributions in order not to harm their rights to submit them to conferences or journals. We hope that the reputation of our Dagstuhl Seminars will make their proceedings a valuable source of information. I felt that this was a good starting point for becoming an online-publishing house. However, a colleague of mine from our university's law school convinced me that much more is involved. Hence, we are still working on this delicate subject.

The State and the Activities of the *Dagstuhl Foundation*

The foundation currently has 46 personal members and 8 institutional members.

In 2002, the foundation has supported a few guests with travel grants and a reduction of the Seminar fees. In 2003, we were very grateful for having interests from the Foundation available to support our first Seminar on e-accessibility, which had a significant number of handicapped people as participants. These needed more financial support for their travel than a usual seminar.

Thanks

I would like to thank you for supporting Dagstuhl through your membership in the *Dagstuhl Foundation*. Thanks go to Fritz Müller for editing the resumees collected in this volume.

Reinhard Wilhelm (Scientific Director)

Contents

Chapter 1

Data Structures, Algorithms, Complexity

1.1 Algorithmic Combinatorial Game Theory

Seminar No. **02081** Report No. **334** Date **17.02.–22.02.2002**
Organizers: Erik Demaine, Rudolf Fleischer, Aviezri Fraenkel, Richard Nowakowski

Games are as old as humanity. The combinatorial game theory community has studied games extensively, resulting in powerful tools for their analysis, like the notion of game-theoretic value. This theory provides a high-level understanding of how to play combinatorial games, but to completely solve specific games requires algorithmic techniques. So far, algorithmic results are rare and mainly negative, e.g., the proofs that Chess and Go are EXPTIME-complete. There are also some positive results on endgames of Go and on various classes of impartial games. But most games lie in "Wonderland", i.e., we are wondering about their complexity/efficiency. (We are not normally interested in exhaustive approaches like the recent world-class computer players for Checkers and Chess.)

The two large communities of combinatorial game theory and algorithmics rarely interact. This is unfortunate. Game theory could benefit from applying algorithmic techniques to games with known outcomes but no known efficient strategies, e.g., Hex and poset games such as Chomp. On the other hand, better knowledge of the game-theoretic tools could help researchers in algorithmics to develop more efficient or more general algorithms for games whose complexity is barely known, e.g., Hex and Chomp and epidemiography games such as Nimania. Maybe the game-theoretic framework can even be extended to noncombinatorial games, like geometric games.

There has been a recent surge of interest in algorithmic combinatorial game theory from both communities. The goal of this workshop was to bring these two communities together, to advance the area of algorithmic combinatorial game theory from infancy to maturity.

In all, 46 researchers with affiliations in Austria (1), Canada (5), the Czech Republic (4), Germany (16), Hong Kong (1), Israel (2), the Netherlands (4), Poland (1), Sweden (1), Switzerland (1), and the USA (10) participated in the meeting (some of them EU citizens

working abroad). Nineteen participants were graduate students or postdocs. Four invited keynote speakers, Elwyn Berlekamp, Aviezri Fraenkel, Joel Spencer, and Jürg Nievergelt, gave one-hour position talks. The remaining 31 presentations given by participants of the meeting covered a wide range of topics, ranging from complexity theoretic results up to experimental studies. Game-theoretic analysis of popular board games like Go and Amazons never ceases to be interesting. The algorithmicians on the other hand provided NP-hardness proofs of games like Clickomania and variants of Pushing Block games, or efficient strategies for game playing. And all younger participants were eager to learn the differences between the US and European tenure game. A special issue of TCS-A (Theoretical Computer Science, series A), edited by R. Fleischer and R. Nowakowski, containing selected papers presented at this Workshop is in preparation.

The evening sessions were devoted to the discussion of open problems and a Clobber tournament (played on a 5×6 board). The winners of this tournament were Tomáš Tichý and Jiří Sgall (runner-up). The computer Clobber tournament (all programs were written on the first day of the Workshop) was won by R. Hearn. Clobber is a new two-player game, recently invented by Albert, Grossmann, and Nowakowski, and not much is known about it (inspired by our tournament, the upcoming Third International Conference on Computers and Games in Edmonton will have a Clobber Problem Composition Contest). During the workshop, two papers on Clobber were written that will also be submitted to the TCS special volume. Actually, we expect many more papers to originate from this very successful workshop, as several of the proposed open problems were already solved during the week (and solutions presented in a special session at the end of the Workshop), and other problems at least partially solved.

1.2 Data Structures

Seminar No. **02091** Report No. **335** Date **24.02.–01.03.2002**
Organizers: Susanne Albers, Robert Sedgewick and Peter Widmayer

The area of Data Structures continues to be an important and vibrant aspect of computer science. The topic is an essential component in the algorithmic solution of many problems. Although data structures have been studied for four decades, there is still a large research community working on exciting and challenging problems. The Sixth Dagstuhl Seminar on Data Structures was attended by 59 people and hence it was larger than all previous meetings. Attendees came from 13 different countries and included many young colleagues. About a third of the participants were attending the seminar for the first time, bringing new ideas and points of view.

There were 40 workshop presentations and, despite of the high attendance, there was sufficient time for scientific discussions and research in teams. The presentations addressed classical data structuring problems as well as new problems arising in important applications. Many interesting results were presented on classical issues such as dictionaries, ordered lists, ordinary search trees, finger trees, B-trees and priority queues. A number of lectures considered classical graph problems. Several presentations investigated data

structuring problems in computational geometry, in particular geometric problems with moving objects. With respect to external memory algorithms, several talks presented cache oblivious solutions that need no knowledge of the exact parameters of the memory hierarchy. Last but not least, there were several contributions investigating data structure problems in specific application areas such as Networks, Parallel Computing and Database Systems.

1.3 Complexity of Boolean Functions

Seminar No. **02121** Report No. **338** Date **17.03.–22.03.2002**
Organizers: David Mix Barrington, Johan Håstad, Matthias Krause, Rüdiger Reischuk

Summary of the Proceedings Many talks of the seminar dealt with new techniques for analyzing the computational power of basic models to compute Boolean functions. In particular, branching programs were dicussed most extensively. At the first day we had a keynote talk in the morning and an evening discussion on time-space tradeoff results on the level of branching programs (Beame). Several talks on refined lower bound methods for nondeterministic and randomized free BDDs (Okol'nishnikova, Žák, Sauerhoff, Wölfel) and the approximability of Boolean functions by OBDDs (Wegener) followed. Other important topics were new results concerning distributed computing of Boolean functions (Jakoby) and communication complexity (Forster, Thérien, and several BDD talks). One highlight here was the presentation of and the discussions on Forsters technique to prove almost optimal lower bounds on the unbounded error probabilistic communication complexity of particular Boolean functions (Forster, Simon). Further talks considered the comparison of classical models and related quantum models for computing Boolean functions (Sieling, Klauck, van Melkebeek, Buhrman, Kerntopf). In addition, besides presenting his own results, Klauck discussed Razborov's very recent solution to a long open problem on deterministic versus probabilistic quantum communication complexity.

Other talks of the seminar dealt with methods for better determining the complexity of hardware relevant Boolean functions (like integer multiplication) with respect to models used as data structures in hardware verification (Bollig, Wölfel), the computational power of decision lists (Krause), and new results on the power of span programs (Gál).

Efficient algorithms was another main topic, especially concerning restricted types of circuits and branching programs as data structures for manipulating, minimizing and learning Boolean functions. Here we had several interesting talks about latest progress in SAT algorithms (Hofmeister, Goerdt, Alekhnovich), new developments in proof complexity (Ben-Sasson, Alekhnovich), new positive and negative results on the learnability of DNFs and AND-decision lists (Maruoka, Krause), and fixed-parameter tractability (Ragde).

Further talks were concerned with relations between Boolean complexity topics and uniform complexity theory, especially with the complexity of derandomizing probabilistic algorithms (Allender, Kabanets), and the closely connected topics of characterizing logspace-classes (Thierauf) and the uniform complexity of reachability problems (Koucký, Barrington). Several talks stressed, at least implicitely, cryptographic implications of structural

and complexity-theoretic results on Boolean functions, especially from the viewpoint of design and security criterions for cryptographic primitives like pseudorandom functions and permutations and S-Box functions (Golic, Lucks).

The contributions of this seminar showed that several new trends in Boolean complexity have gained increased consideration, in particular proof complexity and computing with quantum bits. We have discussed in detail how far our current proof methods have brought us to precisely determine the computational complexity of Boolean functions for general computational models.

The seminar had a number of younger European researchers who for the first time had a chance to take part in such a detailed discussion on current research topics in Boolean complexity. About half of the presentations were given by participants from outside the European Union. The research on Boolean functions is conducted in a broad international exchange. We felt that this meeting at the IFBI was quite productive for all participants concerning their own future research.

Public Outreach To determine the complexity of Boolean functions with respect to various hardware models – like Boolean circuits, branching programs or constant layer feedforward neural networks – is one of the central and classical topics in the theory of computation. This includes the search for efficient implementations of hardware relevant functions, like address functions and arithmetic and logical operations. On the other hand, we strive for establishing lower bounds on the computational complexity showing that a certain function cannot be computed if a certain amount of resources is not available. In this respect, a lot of interesting and surprising results have been obtained, which in many cases are based on the development of elegant, highly nontrivial mathematical proof techniques. However, in spite of enormous efforts, there still seems to be quite a long way to go before getting tight characterizations of the complexity of important functions for general types of circuits and branching programs. Methods originally designed to analyze the complexity of Boolean functions turned out to have interesting implications in other areas like hardware verification, computational intelligence and cryptography.

The aim of this seminar was to collect the leading experts of Boolean complexity theory and to present the latest results in this area. One main focus was to discuss successfull applications of Boolean complexity methods in other more applied fields like hardware design and verification, algorithmic learning, neural computing, proof complexity theory, quantum computing, design of cryptographic primitives, and cryptoanalysis of block and stream ciphers.

1.4 The Travelling Salesman Problem

Seminar No. **02261** Report No. **346** Date **23.06.–28.06.2002**
Organizers: D.S. Johnson, J.K. Lenstra, G. Woeginger

The Traveling Salesman Problem belongs to the most basic, most important, and most investigated problems in optimization and theoretical computer science: A salesman has

to visit each city from a given set exactly once. In doing this, he starts from his home city, and in the very end he has to return again to this home city. He wants to visit the cities in such an order that the total of the distances traveled in his tour becomes as small as possible, since this will save him time and gas. The Traveling Salesman Problem (TSP) consists in identifying this shortest tour through the cities.

The TSP has many important applications in vehicle routing, VLSI design, production scheduling, cutting wallpaper, job sequencing, data clustering, curve reconstruction, etc etc etc. Research on the TSP has followed many different paths: There are studies of its computational complexity, of its approximability, of the complexity and approximability behavior of various of its special cases, there are many implementations e.g. via cutting planes, there are studies and comparisons of implementations, there are approaches via graph theory that study certain Hamiltonian structures etc. etc. etc.

The Dagstuhl seminar on the TSP brought together researchers from Theoretical Computer Science, Operations Research, Mathematical Programming, Discrete Applied Mathematics, and Combinatorics who discussed new developments and new progress made on the TSP during the last 15 years.

1.5 Online Algorithms

Seminar No. **02271** Report No. **347** Date **30.06.–05.07.2002**
Organizers: Susanne Albers, Amos Fiat, Gerhard Woeginger

Online algorithms have received considerable research interest during the last 15 years. In an online problem the input arrives incrementally, one piece at a time. In response to each input portion, an online algorithm must generate output, not knowing future input. Online problems arise in very many areas of computer science, including e.g. resource allocation in operating systems, data structuring, robotics or large networks. The performance of online algorithms is usually evaluated using competitive analysis. An online algorithm A is called c-competitive if, for all input sequences, the solution computed by A is at most a factor of c away from the solution generated by an optimal offline algorithm that knows the entire input in advance.

The Dagstuhl meeting on *Online Algorithms* brought together 58 researchers with affiliations in Austria, the Czech Republic, Denmark, Germany, Hong Kong, Israel, Italy, Japan, the Netherlands, Switzerland, the UK and the USA. 11 participants were young scientists. There were 40 workshop presentations and, despite of the large number of talks, there was sufficient time for scientific discussions and research in teams. The presentations addressed classical online problems as well as new problems arising in important applications of current interest. Many interesting results were presented on classical issues such as paging and caching, bin packing, coloring, the k-server problem and metrical task systems. There was also a considerable number of lectures on online scheduling problems. Several presentations considered the fresh and interesting field of competitive auctions and game theory. With respect to application areas, many talks investigated problems that arise in large networks. Moreover there were talks studying problems in robotics,

online learning, media-on-demand, power saving, seat reservation and vehicle routing. On Thursday evening there was an open problem session where interesting and new problems were presented.

1.6 Experimental Algorithmics

Seminar No. **02371** Report No. **353** Date **08.09.–13.09.2002**
Organizers: Jon Bentley, Rudolf Fleischer, Bernard Moret, Erik Meineche Schmidt

In September 2000, the Dagstuhl Seminar on *Experimental Algorithmics* brought together researchers from both worlds of algorithmics, theoreticians and practitioners. The main question of that seminar was whether and how theoretical and experimental research can co-exist as equal partners under the big roof of *algorithmics*. At the end, the nearly 50 participants agreed that the seminar had been very successful in bridging the two worlds, and they decided to summarize their findings in a Springer Lecture Notes volume *Experimental Algorithmics — The State of the Art*, which was published in 2002. They also agreed that they were still far away from their main goal, namely to characterize the different roles of theory and practice in the field of algorithmics, and that there should be another seminar on this topic in the future.

Therefore, another Seminar was held in September 2002 to further discuss the fundamental question of the value of experiments as opposed to purely theoretical analysis of algorithms. It was also discussed what happens when computer scientists (theoretical or practical) venture out in the world of real systems building and testing (networks, bioinformatics, natural language systems, ...) where they usually meet non-CS engineers or physicists with their own methodological framework of experimental evaluation. Is there a fruitful interaction between CS and non-CS? Can we (the experimental algorithmicists) learn from them? Or they from us?

The aim of this workshop was to bring together three groups, more theoretical oriented researchers, more practical oriented researchers, and people working on real systems. In all, 44 researchers with affiliations in Australia, Austria, Canada, Denmark, Germany, Greece, Hong Kong, Italy, Japan, Spain, and the USA participated in the meeting. Four invited keynote speakers, Jon Bentley, Robert Bixby, Mike Fellows, and Tandy Warnow, gave one-hour position talks. The remaining 21 presentations given by participants of the meeting covered a wide range of topics in experimental algorithmics. One evening was reserved for an open problems session, included below.

Open Problems Session

Collected by E. D. Demaine

The following is a list of the problems presented on September 10, 2002 at the open-problem session of the 2nd Dagstuhl Seminar on Experimental Algorithmics held in Wadern, Germany.

Estimating Running Time: Easy Cases?
Robert Sedgewick

How should we design an experiment to estimate the running time of a program as a function of n? In general, of course, this problem is unsolvable (cf. the halting problem). The idea here is to focus on a very restricted class of programs, and to focus on just estimating the coefficient of the (known) lead term, possibly with knowledge of the entire asymptotic expansion of the running time. One of the main questions here is whether it makes sense to run the program on several instances of the same (large) size, or to run the program on several instances all of different sizes, or with what distribution of sizes, etc.

Determining exactly which restricted class of programs makes sense is part of the open problem. An example of something that *should* be easy is insertion sort; there are many other natural candidates. By making some progress on problems with known solutions computed analytically by hand, we would hope to obtain techniques for estimating the solution for similar unknown problems. In particular, when we make a slight modification to an algorithm whose performance is well-understood, we might not be able to redo the analysis easily, but we can easily run empirical studies.

A few issues that arose in discussion: The entire functional form of the asymptotic running time might be necessary to get a good estimate even for the lead term; at least it may help eliminate noise. A particularly tricky aspect is when lower-order terms oscillate; in this case, we might bound the term by e.g. proving a theorem, and use this bound to estimate the lead term.

Intrinsically Hard Instances: How to Find?
Michael Fellows

How do we find intrinsically hard instances for NP-hard problems that defy all algorithms? What is the value of finding such instances for evaluating the performance of heuristics? In particular, the restricted domain of parameterized complexity may make this task easier, because of the tighter constraints it places on instances.

Three natural suggestions that came up during discussion:

1. Take a random example, kernelize (reduce while preserving the answer, a notion standard in parameterized complexity), and see how much of the instance is left. (Is a large kernel always "hard"?)

2. Internet-based competition ("gambling"). The idea is to run a "hard-instance stock market" which people (even kids) invest a small amount of money to have their examples considered; this is a sort of random parallel search driven by humans.

3. Reduction from hard 3SAT instances. A fair amount is known about hard 3SAT instances, and the reduction from 3SAT to graph 3-coloring doesn't blow up the size much.

Can intrinsically hard instances help us compare multiple implementations, as well as determine whether an implementation is "good enough"? One example discussed was the problem of graph 3-coloring. In this context, is the following conjecture true?

Hard puzzle conjecture: There exists an infinite sequence of 3-colorable graphs such that *every algorithm* (of constant size) performs poorly on *all* sufficiently large instances in the sequence.

Make LEDA Look Bad
Peter Sanders

The 'Make-LEDA-Look-Bad' Contest challenges you to find difficult worst-case instances for two polynomial-time graph algorithms: general weighted matching and max-flow. Even more difficult is to develop a worst-case instance generator that creates an infinite family of difficult instances. The idea is to collect a good set of instances for benchmarking implementations of these algorithms.

Algorithm Sets
Jon Bentley

Let's build algorithm sets analogous to chemistry sets, which allow kids to play and experiment with algorithms instead of chemicals. The idea is to have a classic set of experiments on algorithms, each of which has the following components:

1. Problem statement
2. Application it came from (for the really juicy problems)
3. Environment for kids to work with
   (a) Code for the algorithms
   (b) Testbed for exercising the algorithms
   (c) Animation so that they could see it work
   (d) Inputs
   (e) Generators to make more inputs
4. Classic form of the experiment
5. Discussion about the design of the experiment: why it was set up this way as opposed to various other ways, and how it was implemented.
6. Interaction between theory and experiments

Some candidates arose during the discussion:

1. Sorting (insertion sort, quicksort, etc.)
2. Binary search trees (random inserts, and then random inserts and deletes, an actual set of experiments that was active for over 10 years)
3. Longest common subsequence for DNA sequences (easily motivated to most age groups)

4. Bin packing
5. Traveling Salesman Problem
6. Minimum spanning tree
7. 2-coloring (for a younger audience)

"Little kids" might mean first-year graduate students, or undergraduates, or indeed little kids.

Why Are Solution Spaces So Lumpy?
Michael Fellows

There are several examples of problems whose solution spaces tend to be (but aren't universally) "lumpy" in practice, in the sense that many desired solutions are clustered together instead of being evenly distributed. Can we prove anything giving insight into why solution spaces are lumpy?

For example, with k-leaf spanning tree (is there a spanning tree with at least k leaves?), solutions seem to be clustered among the leaves of the height-k search tree. This solution structure has been exploited by Frank Dehne in some experiments where, by partitioning the search space into pieces and searching each in parallel, he seems to obtain solutions much faster. (Here the problem has already been kernel-reduced.)

Another example is Bill Cook's code for the Traveling Salesman Problem which picks 7 candidate tours out of a soup, takes their union, solves TSP exactly on that union, and adds the result to the soup. The union tends to be a graph with treewidth around 10, which makes TSP solvable exactly in a reasonable amount of time. But theoretically the treewidth is unbounded; perhaps the low treewidth is caused by lumpyness.

A few issues arose in discussion: Some insight might come from problems engineered to have unique solutions, because then there are "no lumps" (in an exploitable way—from another point of view, all solutions are lumped together). Additional light may be shed from the extensive study of 3SAT instances.

1.7 Algorithms and Complexity for Continuous Problems

Seminar No. **02401** Report No. **356** Date **29.09.-04.10.2002**
Organizers: L. Plaskota, K. Ritter, I.H. Sloan, J.F. Traub

Scientific highlights of the Seminar

The seminar was devoted to the computational solution of continuous problems. Concrete algorithms and their analysis were discussed as well as complexity results were presented.

Important continuous problems arise in different areas, and different techniques for analysis of these problems are necessary. Therefore the seminar attracted researchers from computer science, mathematics and applied mathematics, and statistics. There were altogether 46 participants representing 13 countries, among them 20 from Germany and 8 from the US. Together with senior and well recognized scientists, young prospective colleagues, some of them having just finished their diploma or master thesis, were also invited and presented their results.

The lectures on quantum computing for continuous problems built one of the *scientific highlights* of the seminar. Since quantum computers are potentially much more powerful than the classical ones, the quantum model is attracting great attention, mainly for discrete problems. At the Dagstuhl seminar 00931 in 2000 a single talk was devoted to quantum algorithms for continuous problems, and the first results in this field were presented. Thereafter the quantum model has been included into several research projects related to the topics of the seminar. The current question is for what continuous problems the quantum model of computation offers an essential speed-up in solving them. There were 6 talks in which results on quantum complexity of summation, recovery of functions, and integration were presented. E.g., for recovery of functions of many variables from finitely many function values the quantum computer offers a significant speed-up compared to deterministic and randomized algorithms on a real-number machine.

A substantial part of the seminar was devoted to numerical integration, with emphasis again on problems with a large number of variables, and the algorithms under investigation were mainly Monte Carlo or quasi Monte Carlo methods. In some of these talks the computer-based construction of good deterministic cubature formulas was addressed.

A number of talks dealt with non-linear or operator equations, the latter being sometimes analyzed in a statistical setting with noisy data. Probabilistic concepts also played a role as a tool for analysis, e.g., for a problem from computational geometry or for global optimization, or as a part of the problem formulation itself, e.g., for solving stochastic differential equations.

A large proportion of talks, namely 15 out of 40, were presented by *young researchers*. The contributions to quantum computing, e.g., involved senior scientists from Australia, Germany, and the USA together with young colleagues from Poland and Germany. This may illustrate the general fact that the young participants of the seminar have been very well included into joint research efforts.

Research on 'Algorithms and Complexity for Continuous Problems' is done in different places worldwide. At the Dagstuhl seminar most of the participants and almost all of the young scientists were from *Europe*, representing 9 EU member or associated states. Most of the further participants were leading experts from the USA and other countries.

A selection of results presented at this conference will be published as invited papers in the Journal of Complexity.

Explanation of the Subject

To a large extend real-world problems are modelled in terms of concepts from continuous mathematics, e.g., real numbers, derivatives, and integrals. Major examples of continuous models are differential or integral equations, and typically continuous problems can only be solved approximately on a computer, i.e., up to some error $\epsilon > 0$.

The basic question that was addressed at the seminar for a range of continuous problems is: what is the minimal computational cost needed by any algorithm to solve a problem with error at most ϵ? This minimal cost is called the ϵ-complexity, and it quantifies the intrinsic difficulty of a problem. An answer to the basic questions usually includes the construction of an (almost) optimal algorithm: its error is at most ϵ and its computational cost is close to the ϵ-complexity.

The list of problems that where studied at the seminar includes operator equations, optimization, and recovery and integration of functions, with applications in engineering sciences and finance, e.g. Different techniques for analysis of these problems are necessary, and therefore the seminar attracted researchers from computer science, mathematics and applied mathematics, and statistics.

A substantial part of the seminar was devoted to high-dimensional problems, i.e., problems with a large number of variables, where classical algorithms often fail. Here the question of tractability arises: does the ϵ-complexity increase only polynomially in the dimension? In case of a negative answer it is impossible in practice to solve such problems in high dimensions by any algorithm. A positive answer usually comes together with the construction of a new algorithm that turns the high-dimensional problem into a tractable one.

Within the new quantum model of computation the same set of questions has to be addressed again. In fact, the quantum computer is potentially more powerful than the classical one, and therefore it is important to identify those problems where quantum computing offers an essential speed-up. This speed-up was reported to be present, e.g., for recovery of functions of many variables from finitely many function values. While only a first result for quantum computing for continuous problems was available about two years ago, this area of research has rapidly developed since then and it played an important role at the seminar.

1.8 Algebraic Methods in Quantum and Classical Models of Computation

Seminar No. **02421** Report No. **357** Date **13.10.–18.10.2002**
Organizers: Harry Buhrman, Lance Fortnow, Thomas Thierauf

Scientific Report

The seminar brought together groups from two research areas: *quantum information processing* and *computational complexity*. Having said that the most important talk of the

workshop dealt with neither. Manindra Agrawal gave a presentation on the new primality algorithm he developed with his students. They discovered the first provably deterministic efficient algorithm for determining whether a number is prime. This is the most important theoretical computer science result in at least a decade. We were very lucky at Dagstuhl to have Agrawal give this talk, the first talk he gave on the subject outside of his native India.

Steve Fenner gave the first talk giving a wonderful overview of quantum computation for classical complexity theorists. In addition, Steve Høyer showed how to use quantum algorithm as black box subroutines to create new quantum algorithms. These two talks helped produce the synergy of the two areas for the rest of the conference.

The main theme of the workshop considered algebraic methods in the study of both areas and we had several talks along these lines. Scott Aaronson and Andris Ambainis gave talks showing how polynomials and group representations give lower bounds for quantum machines while Ken Regan described how the algebraic degree can lead to lower bounds in classical complexity. Eldar Fischer showed how Fourier transforms play a role in the recently exciting area of property testing.

The graph isomorphism question, a special case of the hidden subgroup problem, has interest to both classical and quantum theorists. Jacobo Torn and V. Arvind discussed the classical complexity of graph isomorphism while Wim van Dam talked about quantum algorithms for cases of the hidden subgroup problem.

Other quantum talks include work on quantum branching programs (Ablayev), quantum circuits (Fenner, Green, Spalek) and quantum Kolmogorov Complexity (Vitanyi).

In addition to Agrawal's presentation on primality, we had a wide-range of talk on classical complexity. Pierre McKenzie described circuits over sets of natural numbers. He gave an exciting open question that many of the attendees struggled over (unsuccessfully) for many hours during the workshop. Bill Gasarch talked about the cake-cutting problem, how to cut a cake so all are happy with the outcome that had equally intriguing open questions.

Jack Lutz talked about his recent interests in effective Fractal dimension, an extension of his work on resource-bounded measure. Denis Therien classified the communication complexity for regular languages.

Rounding out the conference were talks on classical subjects by Stephan, Hertrampf, Reischuk, and Miltersen.

Public Outreach

In the past fifteen years, we have seen several surprising results in computational complexity based on algebraic techniques. For example Barrington's Theorem showing that majority can be computed by bounded-width branching programs uses noncommutative groups, or the research on interactive proofs and probabilistically checkable proofs that led to hardness of approximability results rely heavily on the structure of the zeros of low-degree polynomials.

Nowhere though has the power of algebra played a larger role than in the study of quantum computation. One can view quantum computation as multiplication of unitary matrices.

Shor's famous quantum algorithm for factoring relies heavily on the algebraic structure of the groups Z_m and can be seen as a special case of the hidden subgroup problem for abelian groups. The more general case for non-abelian groups is still a tantalizing open problem and could lead to a polynomial time quantum algorithm for the Graph Isomorphism problem.

Our proposed workshop would bring together leading researchers using algebraic techniques from both the quantum computation area and those studying classical models. Combining these groups of researchers will hopefully lead to a greater understanding of the computational power of both quantum and classical models of computation through new applications of algebraic techniques.

Chapter 2

Verification, Logic, Artificial Intelligence

2.1 The Logic of Rational Agency

Seminar No. **02041** Report No. **331** Date **20.01.–25.01.2002**
Organizers: Wiebe van der Hoek, Michael Wooldridge

Description

The notion of a rational agent is one that has found currency in many disciplines, most notable economics, philosophy, cognitive science, biology, social sciences and, most recently, computer science and artificial intelligence. Crudely, a rational agent is an entity that is capable of acting on its environment, and which chooses to act in such a way as to further its own interests. There is much research activity in the formal foundations of such agents and multi-agent systems. Many mathematical approaches to developing theories of rational agency have been developed, including decision theory, game theory, and mathematical logic. In this seminar, we focussed on logical approaches to rational agency.

There are three aspects to the study of logical approaches to rational agency:

1. Philosophy
2. Logical foundations
3. Application

The first aspect is concerned with the primarily philosophical questions of what rational agency is and how we might go about characterising it. Within the artificial intelligence and AI communities, one approach in particular has come to dominate – the view of rational agents as practical reasoners, continually making decisions about what actions to

perform in the furtherance of their intentions and desires. This view of rational agents is largely seen as going hand-in-hand with the view of agents as intentional systems – systems that may best be characterised in terms of mentalistic notions such as belief and desires.

The logical foundations aspect of the study is concerned with the extent to which these aspects of agents (practical reasoning and mentalistic notions such as beliefs and intentions) can be captured within a logical framework of some kind. There are many well-documented difficulties with using classical (first-order) logic to express these aspects of agency, and so a key component of the logical aspect is finding an appropriate logical framework within which to express an agent's (different kinds of) beliefs, goals, plans, intentions, and how his actions can affect them over time. Although much has been done on modelling such attitudes in isolation, it is still not clear how easy it is to combine several of them into one framework, let alone if one changes the perspective to multi-agent system. From a technical point of view, the logics of choice for expressing these aspects are extremely complex, combining temporal, modal, and dynamic aspects in a single framework. The theoretical and meta-logical properties of such logics (computational complexity, expressive power, completeness results, theorem proving techniques) are not well understood.

Finally, the application aspect is concerned with how we might apply logical theories of agency in the construction of automated agents. Logical theories of agency can be used as (1) a specification language, (2) a programming language, and (3) a verification language. Viewed as a specification language, a logic of rational agency can be used to specify the desirable properties of a system that is to be built. The development of formal methods for specifying the desirable properties of computer systems is a major ongoing area of research activity in computer science, and the view of computer systems as rational agents brings a new dimension to this study. Executable logics have also been a major research topic in computer science, with the programming language PROLOG being perhaps the best-known example of an executable logic framework. While the kinds of logics used in the development of agent theory are typically much more complex than those which underpin languages such as PROLOG, there is nevertheless some potential for developing executable fragments of agent logics. Finally, an interesting issue is the extent to which a computer system can formally be shown to embody some theory of agency. It is an as yet open question how we might go about attributing attitudes such as beliefs, desires, and the like to computer programs. Verifying that a system implements some theory of agency is thus a major research issue.

The structure of the seminar reflected the discussion above:

1. Philosophical foundations

   What is rational agency? What are the right primitives (beliefs, desires, etc) for modelling rational agents? How do these primitives relate to one-another?

2. Logical foundations

   What are the alternatives (e.g., classical logics, modal logics, first-order meta-logics, dynamic/action logics, deontic logics, temporal logics, ...) for modeling of the primitive components of rational agency? What are appropriate semantic frameworks

for these logics (Tarskian model theoretic semantics? Kripke semantics? computationally grounded Kripke semantics? other approaches?) What are the relative advantages of these different frameworks? How do we combine these primitives into a single logic? What are the theoretical properties (expressive power, completeness, decidability/undecidability, computational complexity, proof procedures) of these combined logics? How do we use these logics to capture macro (non-atomic) aspects of rational agency, such as decision making (games, distributed utilities,...), communication, perception, collective action?

3. Application

   How can we use agent logics in the specifiation of agent systems? How can we manipulate or otherwise refine these specifications to generate implementations? Can we directly execute these logics, and if so how? How do we verify that implemented systems satisfy some theory of agency (deductive approaches, model checking, ...)?

Evaluation

We think the workshop was very successful. We know that some collaborations have been intiated during the event. Moreover, the following two special issues came out as spin off from the workshop, both refering explicitly in a forword to the event at Dagstuhl:

W. van der Hoek and M.J.W. Wooldridge (eds),
Towards a Logic of Rational Agency,
special issue of *Logic Journal of the IGPL*, **11**:2, 2003.
see `http://www3.oup.co.uk/igpl/Volume_11/Issue_02/`

W. van der Hoek and M.J.W. Wooldridge (eds),
The Dynamics of Knowledge, special issue of *Studia Logica*, **75**:1, 2003.

Abstracts selected by the Dagstuhl News editor:

Proof methods for the KARO framework

Ullrich Hustadt

We give a short overview of a method for realising automated reasoning about agent-based systems. The framework for modelling intelligent agent behaviour that we focus on is a core of KARO logic, an expressive combination of various modal logics including propositional dynamic logic, a modal logic of knowledge, a modal logic of wishes, and additional non-standard operators. The method we present is based on a translation of core KARO logic to first-order logic combined with first-order resolution. We discuss the advantages and shortcomings of the approach and suggest ways to extend the method to cover more of the KARO framework.

2.2 Nonmonotonic Reasoning, Answer Set Programming and Constraints

Seminar No. **02381** Report No. **354** Date **15.09.–20.09.2002**
Organizers: G. Brewka, I. Niemelä, T. Schaub, M. Truszczynski

Scientific highlights of the event

Answer set programming is an emerging programming/problem solving paradigm. The fundamental underlying idea is to describe a problem declaratively in such a way that models of the description provide solutions to problems. One particular instance of this paradigm are logic programs under stable model semantics (respectively answer set semantics if an extended class of logic programs is used). Tremendous progress has been made recently in this area concerning both the theoretical foundations of the approach and implementation issues. Several highly efficient systems are available now which make it possible to investigate some serious applications.

The talks of the workshop were centered around the following main research topics:

- Useful language extensions and their theoretical foundations, with a particular focus on cardinality, weight and other types of constraints.
- Preferences in answer set programming and their implementation, where the preferences considered are among rules, among literals, or among disjuncts in heads of rules.
- Implementation techniques for answer set solvers. Several new methods or improvements of existing methods were presented, some of them based on highly efficient existing satisfiability solvers.
- New attempts to handle programs with variables. Existing solvers produce the ground instantiation of a program before computing answer sets and disallow function symbols. More flexible and less space consuming techniques are needed for large applications.
- Applications of the answer set paradigm in planning, scheduling, linguistics etc.

In addition to the talks a system competition took place during the workshop. Five systems participated in the competition, namely *dlv* (TU Vienna/Univ. Calabria), *Smodels* (Helsinki UT), *ASSat* (Univ. Honkong), *cmodels* (UT Austin) and *aspps* (University of Kentucky). In a meeting at the beginning of the seminar the participants agreed about the benchmark problems to be used in the competition. The problems were encoded and tested and results presented in a plenary session at the end of the week.

Another topic of interest was standardization. There was an panel on the subject followed by open discussion. A general feeling was that the matter of standardization is a topic that requires a thorough attention on the part of the community in the near future.

Training

Among the participants of the workshop were 11 young researchers, most of them PhD students. The students were allotted the same amount of time as everybody else for their talks to make sure they received enough attention from senior scientists. For many of the students it was the first time they presented their results/projects to an international audience. The students had a chance to discuss with world leading researchers in their area. This will certainly have an impact on their future work.

European added value

It is fair to say that in the field of answer set programming, and in particular in implementing advanced answer set solvers, Europe is currently on par with research in North America, if not leading. There is a number of European research groups active in this area. The EC just started to fund a Working Group on Answer Set Programming. The major goals of the Working group are the further advancement of the theoretical understanding of ASP (this includes the investigation of new potentially useful language constructs and their semantics), the further development of efficient advanced reasoning systems which make ASP techniques widely available (this includes the development of front ends for specific application problems), and the investigation of the applicability of ASP to areas such as planning, configuration, encryption, verification, knowledge extraction and others.

During the seminar the kickoff meeting of the working group took place, and the members had an excellent opportunity to get first hand information about current research developments in each group.

Given the numerous application areas for which promising answer set programming solutions already exist today, we expect tremendous economic benefit of this research. The seminar was important to keep Europe at the forefront of research in this area.

Public Outreach

Answer set programming is a new declarative programming methodology. The basic idea is that programmers, rather than having to specify how a computer should solve a problem, just describe what the problem is. Each model of the problem description then provides a possible solution to the problem. The exact notion of a model used here depends on the language used for describing problems, but in all cases the models (also called answer sets in this context) can be thought of as sets of facts representing what is true and what is false.

Although theoretical foundations have been laid and some highly efficient implemented systems are available, there are still numerous challenging scientific questions which need to be answered: improved implementation techniques, extensions of the declarative languages which facilitate the problem description, methods for applying these techniques to problems like planning, scheduling, configuration etc. Contributions to all of these topics were presented and discussed during the seminar.

2.3 Programming Multi Agent Systems based on Logic

Seminar No. **02481** Report No. **361** Date **24.11.–29.11.2002**
Organizers: Juergen Dix, Michael Fischer, Yingqian Zhang

Nature and importance of the subject

Multi-agent systems are set to be the key technology for software organisation during the next decade. While there have already been a number of multi-agent systems developed, the programming technology available for constructing such systems is relatively immature. Hence, there is a need for a powerful, general purpose programming technology for multi-agent systems.

The intention of this seminar is to bring together the leading researchers in these areas and to foster interaction between the various groups and thus get a better understanding of the ways in which multi-agent systems may be programmed in the future. As well as targeting logical approaches, a key element is to consider the requirements for efficient systems scaling within real world applications.

Over many years, work on computational logic has spawned research areas such as knowledge representation (KR), nonmonotonic reasoning (NMR), automated deduction (AR), and deductive databases (DDB). Each of these can be seen as an essential component within multi-agent systems, as agents need to

- *describe* the world (KR),
- *reason* somehow about how the world behaves (AR),
- *decide* in the light of uncertain information (NMR), and
- *deal* with massive data stored in heterogeneous formats (DDB).

In parallel, work within the multi-agent systems community has involved developing, often via logic, concepts concerned with communication languages and distributed computation (CC), cooperation and teamwork (TW), and the dynamic development of agent organisations (ORG). Again, each of these aspects can be seen as being required in complex multi-agent systems, as agents need to *communicate* with other distributed agents (CC), *cooperate* with other agents in order to achieve some goal (TW), and *evolve*, dynamically, organisational structures appropriate to the particular situation (ORG).

Goals of the Seminar

The seminar was set up in a way to allow ample time for discussions. We restricted the presentations to 30-35 minutes and allowed 10-15 minutes time for discussion after each presentation. This concept allowed for four talks in the morning and two talks after lunch.

We also set up four working groups: (1) Programming negotiation in agents, (2) Programming deliberation/rationality in agents, (3) Information/Data management via logic-based agents, (4) Programming cooperation in agents. Participants had been allocated to these groups three weeks before the seminar started. Each working group was chaired by two senior researchers who contacted the participants and distributed material before the seminar. The groups met on Monday and Tuesday from 4-6 pm.

The idea behind these working groups was:

1. to identify key exemplars/problems that are relevant to that area;
2. to describe these exemplars/problems concisely/abstractly (can some of them be used as benchmarks/prototypical examples to check particular frameworks against?); and
3. to find out if, and to what extent, logic-based programming of multi-agent systems is useful for solving these problems.

Results were presented on Thursday, where all participants met from 4-6 pm.

An ambitious outcome that we aimed for was

> *A set of challenge problems/exemplars for logic-based programming of multi-agent systems. In addition, some criteria to determine whether a logic-based approach is useful or not. Or a list of problems where other methods are superior.*

Outcomes of the Seminar

A homepage for the seminar has been set up, at `http://www.cs.man.ac.uk/~zhangy/dagstuhl`, containing all the presentations, the results of the working groups, and, last but not least, some photos of our official excursion: a wine tasting in Riol. As can be seen from the programme of presentations available on that web site, the seminar contained a wide variety of high-quality talks. Many participants commented on the excellent programme.

The working group idea generally worked well, with the groups often meeting outside their scheduled times. While the overall goal of the groups was perhaps too ambitious (after just two meetings), some interesting results have already emerged. We are currently trying to get the groups to continue their work (and, indeed, most seem keen) and hope that something useful and publishable will come out of it.

Following interactions during the seminar, it was decided to propose a new workshop on *Languages, Tools and Techniques for Programming Multi-Agent Systems* for AAMAS 2003 in Melbourne, Australia. This event is the most important conference on agent-based systems and is held annually. Over 12 seminar participants are now involved in the programme committee for this proposed workshop, and the time at Dagstuhl allowed us to work together on the application.

It has also been decided by several participants of the seminar to set up a steering committee for organising and continuing the CLIMA workshop series (*Computational Logic in Multi-Agent Systems*), which is closely related to the topic of the seminar.

Another important outcome of the seminar was to develop the details of a special issue of *Annals of Mathematics and Artificial Intelligence* on the topic of "Logic-Based Agent Implementation". Again, interactions at the seminar led to the publication of the *call for papers* for this initiative; see `http://www.csc.liv.ac.uk/~michael/LBAI03`.

Chapter 3

Image Processing, Graphics

3.1 Content-Based Image and Video Retrieval

Seminar No. **02021** Report No. **329** Date **06.01.–11.01.2002**
Organizers: Jitendra Malik, Hans-Peter Kriegel, Linda Shapiro, Remco Veltkamp

Images and video play a crucial role in Visual Information Systems and Multimedia. There is an extraordinary number of applications of such systems in entertainment, business, art, engineering, and science. Such applications often involve huge collections of images, so that efficient and effective searching for images and video is an important operation.

The previous Dagstuhl Seminar on Content-Based Image and Video Retrieval was the first one on this topic, and turned out to be a big success, as demonstrated by the following two results:

- During the seminar we collectively discussed the problems of performance evaluation and quality assessment of retrieval systems.
- A selection of the presentations has been published as a book in the Kluwer series on Computational Imaging and Vision with the title State-of-the-Art in Content-Based Image and Video Retrieval, Kluwer, 2001.

This motivated us to organize a follow-up seminar, with the central theme "Object recognition for image retrieval". The emphasis of this second seminar will lie on identifying the principal obstacles that hamper progress in content-based retrieval. Fundamental questions such as whether image 'understanding' is necessary for effective image 'retrieval' and whether 'low' level features are sufficient for 'high' level querying. We strongly believe that image and video retrieval need an integrated approach from fields such as image processing, shape processing, perception, data base indexing, visualization, querying, etc.

Topics to be discussed at the seminar include:

Object recognition

Semantic-based retrieval

Indexing schemes

Shape, texture, color, and lay-out matching

Relevance feedback

Visual data modeling

MPEG7 and JPEG2000 issues

Retrieval system architectures

Image and video databases

Feature recognition

Visualizing pictorial information

Video segmentation

Picture representation

Query processing

Perception issues

Searching the web

Delivery of visual information

Benchmarking

Application areas of image and video retrieval

3.2 Theoretical Foundations of Computer Vision — Geometry, Morphology and Computational Imaging

Seminar No. **02151** Report No. **339** Date **07.04.–12.04.2002**
Organizers: Tetsuo Asano, Reinhard Klette, Christian Ronse

Image analysis and computer graphics depend on geometric modelling and analysis of objects in two- or multidimensional spaces. Different disciplines such as digital geometry, mathematical morphology, polyeder geometry or computational geometry, just to cite a few, are closely related to progress in image analysis and computer graphics.

The workshop discussed theoretical fundamentals related to those issues and specified open problems and major directions of further development in the field of geometric problems related to image analysis and computer graphics. The seminar schedule was characterized by exibility, working groups, and sufficient time for focused discussions. There will be an edited volume of seminar papers (within the Springer LNCS series).

The contributions during the workshop have been related to one of the following subjects:

(1) geometric algorithms for image processing or computer vision for extracting structures from images, geometric shape matching, image segmentation and image restoration, or image halftoning,

(2) mathematical morphology (multiresolution representations, texture models, lattice-theoretical and fuzzy models),

(3) geometric feature analysis (length of a curve, area of surfaces in 3D, curvature), or

(4) further geometric aspects of computer vision or image processing occurring in image acquisition, optical illusions, shape recovery or depth analysis, or modelling of complex situations in vision-based robotics.

The workshop had 41 participants: 10 from Japan, 9 from France, 5 from Germany, 3 from Israel, New Zealand and USA each, 2 from The Netherlands and Slovakia each, and one from Australia, Belgium, Canada and Italy each.

3.3 Geometric Modelling

Seminar No. **02201** Report No. **341** Date **12.05.–17.05.2002**
Organizers: Guido Brunnet, Gerald Farin, Ron Goldman, Stefanie Hahmann

Geometric Modelling is the branch of Computer Science concerned with the efficient representation, manipulation, and analysis of geometry on a computer. The origin of this discipline is curve and surface design for CAD/CAM systems. Today, Geometric Modelling is a well established field with a wide range of applications, including computer graphics, scientific visualization, virtual reality, simulation, and medical imaging, and it attracts researchers with backgrounds in computer science as well as mathematics and engineering.

The 5th Dagstuhl seminar on geometric modelling was attended by 51 participants. The participants came from 3 continents and 13 countries, and included 6 industrial scientists as well as the leading academic experts in the field. Several young invited researchers were funded by the HLSC program of the European community. A very special event during the conference was the award ceremony for the John Gregory Memorial award. This time Prof. Hans Hagen, Prof. Gerald Farin, Prof. Joseph Hoschek, and Prof. Tom Lyche have been awarded with this price for their fundamental contributions to the field of geometric modelling. After the conference, as with all previous Dagstuhl Seminars on Geometric Modelling, a conference proceedings will be published.

There were a total of 42 technical presentations at the conference related to the following diverse topics:

- curve and surface modelling
- non-manifold modelling in CAD
- multiresolution analysis of complex geometric models
- surface reconstruction
- variational Design

- computational geometry of curves and surfaces
- 3D meshing
- geometric modelling for scientific visualization
- geometric models for Biomedical application

Despite the large number of presentations during the conference and the high attendance at these talks, there was ample time for scientific discussions and research.

Chapter 4

Software Technology

4.1 Supporting Customer-Supplier Relationships: Requirements Engineering and Quality Assurance

Seminar No. **02361** Report No. **352** Date **01.09.–06.09.2002**
Organizers: Barbara Paech, David Parnas, Jesse Poore, Dieter Rombach, Rudolf van Megen

Increasingly, product engineers need to buy software components or to outsource part of their software development. For the cooperation with (external or internal) suppliers or for software procurement, the upper most level of the V-model, namely requirements engineering and quality assurance of the software product, are of utmost importance.

However, traditionally, requirements engineering and quality assurance are seen as separate activities carried out in quite different time frames during system development and through quite different people. Similarly, there is not much overlap in the corresponding research communities.

The purpose of this seminar was to bring together researchers and practitioners in the areas of requirements engineering and quality assurance such as inspection, testing and formal verification that are interested in a coherent support for software contracting. In the course of the seminar synergies and tradeoffs like the following have been discussed:

- How to support the communication between customer and supplier through elicitation and documentation of requirements?
- Which requirements documents can serve as the basis for software purchase?
- What quality assurance methods and products support the monitoring of the supplier?
- How to use quality assurance techniques for software product assessments?
- Can test models substitute a requirements specification as e.g. suggested by Extreme Programming (XP)?

- When to integrate quality assurance in the requirements engineering activities, e.g. what degree of stability is necessary for a requirements specification to serve as a starting point for the specification of the tests, and when to involve quality assurance during requirements specification?
- How can different kinds of quality assurance products be derived from different kinds of requirements specifications, e.g. how to derive test cases from use cases?
- How to distribute effort between requirements specification and quality assurance, e.g. when should the customer require a formal specification or a requirements traceability model, when should the customer sacrifice requirements engineering activities for testing activities?
- How to combine different quality assurance techniques such as inspection, testing and formal verification for supplier monitoring?
- How to assure the quality of non-functional requirements?
- How can the experience gained in quality assurance, be used to improve the determination and documentation of requirements?

The discussions fostered the understanding of both communities and helped to stimulate technology transfer of existing methods into practice as well as research on integrated methods. By inviting both researchers and practitioners from different domains like telecommunication system, embedded systems, information systems or web applications, the identification of context factors for the success of integrated methods was supported.

The seminar was conducted as an open space. On the first day the participants collected the topics they wanted to discuss and present. Based on this, an agenda for the whole week was developed with plenary sessions and working sessions in parallel tracks. Over the course of the seminar the agenda was restructured based on the needs of the participants. Every day the participants assigned themselves to the parallel sessions. In the plenary sessions overview talks were given, summaries of the parallel tracks presented and discussed. This scheme ensured that the groups in each track were small enough for intensive discussions, but on the other hand every participant was informed about the overall results. In the final session the results were put together into a general picture of the pros and cons of the integration of requirements engineering and quality assurance.

4.2 Dependability of Component Based Systems

Seminar No. **02451** Report No. **359** Date **03.11.–08.11.2002**
Organizers: S. Anderson, R. Bloomfield, M. Heisel, B. Krämer

It is now commonplace to develop software based systems from components (e.g. these may be so called commercial off the shelf components, the results of an object oriented development, the evolution of existing product lines). The goal is to describe, design or

select components and then assemble large systems according to architectural principles. Approaches are often sought that minimise the need to know implementation details of the components and to rely on specification of the interface behaviour.

There is usually uncertainty in the evidence that would support claims of dependability of the components. But such evidence is indispensable for critical applications such as medical, aerospace, automobile, financial applications in national infrastructure and embedded systems in the home. Another trend is the proliferation of applications where dependability of software is critical. For such applications

- Dependability-related attributes of components whose implementation details are not known or are uncertain must be assessed,
- The overall system attributes (functionality, reliability, robustness etc.) must be translated into requirements for components or synthesised from the component attributes,
- Techniques are needed that can guarantee or at least assure certain dependability-related properties of a system even it is assembled of components for which no guarantee is given,

To tackle these problems an interdisciplinary approach is needed that combines safety and requirements analysis techniques, specification techniques, design adaptation techniques such as wrappers and adapters and probabilistic modelling of decision making under uncertainty.

The integration of disparate sources of evidence is another challenge of component-based dependable systems.

The aim of the seminar is to bring together researchers and practitioners in order to achieve a common understanding of the problems and collect possible solutions. We hope to experience synergetic effects by inter-disciplinary working.

Besides the technical aspects of safety and component-orientation, questions of certification and standardisation will be discussed. The week will be structured to facilitate industrial involvement.

Chapter 5

Applications, Interdisciplinary Work

5.1 Aesthetic Computing

Seminar No. **02291** Report No. **348** Date **14.07.–19.07.2002**
Organizers: Paul Fishwick, Roger Malina, Christa Sommerer

The Aesthetic Computing Seminar was organized by Paul Fishwick (University of Florida), Roger Malina (University of California Berkeley), and Christa Sommerer (ATR Media Integration and Communications Research Lab), and took place at Schloss Dagstuhl in July 2002.

The initial motivation for the seminar was to investigate into alternative, cultural and aesthetically motivated representations for computer science models such as automata networks, flow graphs, software visualization structures, semantic networks, and information graphs. This was seen as increasingly relevant as the wave of rich, personalized sensory modes became more economic by the perpetual march toward faster and better interfaces. If it were possible to build software models from any material, and with great speed and agility, what new forms of expression would be crafted? It was expected that aesthetics and artist-driven approaches to model representation was about to emerge from more efficient and expressive methods of representation based on advanced technologies. So it was hoped that the advanced possibilities could bring e.g. visualization to be not only about presenting output but also to be about completely new methods of modeling. Thus, Aesthetic Computing was understood as a new trend in modeling and representation where art and science would come together, with art in direct support of science.

The mix of artists and academics from all sorts of fields resulted in a fruitful week with inspiring presentations, divergent discussions, and even constructive group work, bringing us closer to an understanding of what aesthetic computing might be, but further away from a definition. In the last session we tried to formulate what aesthetic computing could be about, based on that discussion Paul wrote the aesthetic computing "manifesto".

Aesthetic Computing "Manifesto"

Recorded by Paul Fishwick

The application of computing to aesthetics, and the formation of art and design, has a long history, which reached a substantial state in the 1960s, with the use of hardware, software, and cybernetics to assist in creating art. We propose to look at the complementary area of applying aesthetics to computing. Computing, and its mathematical foundations, have their own significant aesthetics; however, there is currently a difference between the relative plurality and scope of aesthetics in computing when contrasted with art, which has a long history containing a multitude of historical genres and movements. For example, software as written in text or drawn with flow-charting may be considered elegant. But that is not to say that the software could not be rephrased or represented given more advanced media technologies that are available to us today, as compared with when printing was first developed. Such representation need not compromise the goals of abstraction, nor the material or sensory engagement used to formulate the constituent signs for a given level. Abstraction is a necessary but not sufficient condition for mathematics and computing, as meaning, comprehension, and motivation may be enhanced if the presentation includes additional cognitive or aesthetic elements. Such presentation may involve multiple sensory modalities.

Computer programs have been traditionally presented in standard mathematical notation even though, recently, substantial progress has been made in areas such as software and information visualization to enable formal structures to be comprehended and experienced by larger and more diverse populations. And yet, even in these visualization approaches, there is a tendency toward the mass-media approach of standardized design, rather than an approach that takes account of a more cultural, personal, and customized set of aesthetics. The benefits of these latter qualities are:

1. an emphasis on creativity and innovative exploration of media for software and mathematical structures,
2. leveraging personalization and customization of computing structures at the group and individual levels, and
3. enlarging the set of people who can use and understand computing.

The computing professional gains flexibility in aesthetics, and associated psychological attributes such as improved mnemonics, comprehension, and motivation. The artist gains the benefits associated with thinking of software, and underlying mathematical structures, as raw material for making art. With these benefits in mind, we have created a new term Aesthetic Computing, which we define as the theory, practice and application of aesthetics in computing.

5.2 Computational Biology

Seminar No. **02471** Report No. **360** Date **17.11.–22.11.2002**
Organizers: Russ Altman, David Gilbert, Thomas Lengauer

This seminar was the fourth seminar on general issues in Computational Biology that was held at Dagstuhl. Three previous seminars on this topic have been held in 1992, 1995 and 2000.

Computational Biology addresses the problems of interpreting genomic data with computational methods. These data harbor the biological secrets of life, however, these secrets are encoded in intricate ways that we do not understand yet. The genome tells which molecules should be manufactured and when they should be manufactured in what quantities. It says how the molecules should be arranged and harbors information on how they interact with each other. All of this information is so cryptically encoded in the genome, however, that we need computers to learn biology from the genomic information.

With the great advance of the underlying experimental techniques in biology which provided complete genomes of several hundred organisms by now, and is unearthing additional voluminous data on the difference of the molecular makeup of different tissues in healthy and diseased conditions, computational biology has experienced rapid development. The field is highly interdisciplinary, with aspects from physics, chemistry, biology and medicine as well as mathematics, statistics and computer science. Therefore the need of scientific exchange is enormous. This seminar series addresses this need and brings together active researchers for a wide variety of backgrounds that participate in the quest of understanding the molecular basis of life with computational methods.

The seminar explored traditional as well as some more novel issues in computational biology. The field has expanded greatly in the past years, and the danger has grown of splitting the field into more and more separate sub-disciplines. This seminar attempted to slow down this trend by giving all attendees an overview of the state of the art in widely differing sub-areas of computational biology. These included haplotype analysis, sequence analysis, molecular structure analysis, molecular docking, analysis of gene expression data and biochemical networks as well as issues in medical applications and software issues in project design.

The days were filled with lectures that had extended discussion periods. Some of the talks had decidedly tutorial character. Early afternoons were set aside for informal discussions. There were evening discussion sessions on Biochemical Pathways, and Bioinformatics and Disease. It was a common sentiment that the broad scope of the seminar is worthwhile and should be maintained in future seminars.

Chapter 6

Semantics, Specification

6.1 Theory and Application of Abstract State Machines

Seminar No. **02101** Report No. **336** Date **03.03–08.03.2002**
Organizers: Andreas Blass, Egon Börger, Yuri Gurevich

The seminar was proposed to the participants with the following goal which we restate here from the Call for Participation:

The advances in the theory, the tool development, and the progressive industrial employment of Abstract State Machines (ASMs) in the 90's have turned ASMs into a practical technique for disciplined rigorous software engineering in the large. The proposed seminar aims at bringing together ASM researchers from academia and industrial users of ASMs to strengthen this fruitful interaction between theory and practice.

As a result of the research and the applications of Abstract State Machines during the last decade, ASMs offer a certain number of theoretically well founded and industrially useful methods, which support the entire software development cycle. These include rigorous modeling, analysis and validation methods a) for the requirements, during the early phases of industrial software development, and b) for the refinement of the high level models through a design process which reliably connects the requirements to the code development. Via the definition of appropriate ground models, which can be made executable, ASMs support the elicitation, specification, inspection and testing of requirements. Building the high-level models leads to a good understanding of the requirements. It contributes to practical inspections and to testing which help to detect errors at the earliest possible stage of software production - well known to be responsible for most of the costly errors occurring during the software development process. The controlled stepwise refinement of high-level models which turns them into efficiently executable code also supports a good documentation discipline, which is helpful for the maintenance and the reusability of the intermediate models which reflect critical design decisions.

The specific goal of the seminar is to survey and to critically evaluate the current academic and industrial developments and new results concerning ASMs. In particular we want

to provide guidelines for future research and development by identifying new challenges coming e.g. from component based design techniques, software architecture patterns, mobile computing, security concerns, etc. Corresponding to the goal to evaluate ASM related scientific achievements and their current industrial employment and development, the list of persons to be invited tries to reflect both the academic and industrial aspects of current work on ASMs.

The seminar realized those goals. It was attended by over 60 participants from all over Europe and the US. The presentations ranged from highly theoretical work to genuine industrial applications, and so did the discussions.

6.2 Concurrency and Dynamic Behaviour Modelling: Pragmatics & Semantics

Seminar No. **02111** Report No. **337** Date **10.03.–15.03.2002**
Organizers: Gregor Engels, Rob van Glabbeek, Ursula Goltz

A topic which has gained increasing interest in the past years is the modeling of distributed and concurrent systems. Typical applications are for example in the area of real-time, embedded, and component-based systems, Web-based and multi-agent systems. The complexity of such systems in combination with high demands on their reliability call for adequate design methods.

Concurrency theory provides a formal basis for specifying such systems, consisting of approaches such as process algebra and Petri nets for modeling, logics for expressing properties of concurrent systems, and methods for analysis and verification; Pi-calculus, ambients and control structures provide mobility concepts. Semantic models underlying these concepts were investigated, for example transition systems with various notions of equivalence and event structures. Coalgebras and hidden algebras provide a uniform framework for modeling dynamic behavior and modularization. However, the impact of these developments on practical software development has been limited. One reason is the lack of integration of specification techniques for different aspects of software development, and the missing support for specific application domains and methodologies. Another reason lies in the difficulties of practitioners in reading and writing formal specifications.

Software engineering methods are being developed which specifically address these issues. For example, the Unified Modeling Language (UML) integrates design notations for specifying the logical and physical structure of a system, its dynamic behavior, the interaction with other systems, etc. Being a general-purpose language, the UML provides mechanisms for defining domain-specific profiles of the language. An intuitive diagrammatic notation allows its use by application developers without background in formal methods. However, as UML lacks a formal foundation, models are often ambiguous, and there is no satisfactory support for analysis and verification of models.

The goal of this seminar was to bring together people from both areas of research for the mutual benefit of

- discussing the technology transfer from concurrency theory to (in particular) object-oriented modeling, and
- deriving new challenges for concurrency theory from problems in practical software development.

In particular, the following topics have been addressed:

- Semantics of behavioral models, including problems of under-specified and open systems.
- Consistency between between non-orthogonal sub-models.
- Support for methodologies and specific application domains.
- Adequacy and expressiveness of behavior models, abstraction levels in modeling.
- Analysis and verification (model checking, etc.), code generation.
- Advanced concepts like time and mobility.

The discussion of these and other issues between experts from the research fields outlined above led to a better understanding of the semantics of models for dynamic behavior of concurrent systems. In a working group, perspectives on further developments both from the theoretic and pragmatic point of view have been discussed.

Chapter 7

Distributed Computation, Nets, VLSI, Architecture

7.1 Concepts and Applications of Programmable and Active Networking Technologies

Seminar No. **02071** Report No. **333** Date **13.02.–15.02.2002**
Organizers: David Hutchinson, Bernhard Plattner, Peter Steenkiste, Martina Zitterbart

One of the major challenges of emerging networks (fixed and mobile) lies in the flexible creation and rapid deployment of a large variety of existing and newly emerging services. However, existing networks are highly inflexible and do not easily allow for the provisioning of new services. This explains why novel and useful services are not appearing more rapidly. Examples are multicast services, security, accounting and charging services, Quality of Service support and the like.

An attractive vision is to make future networks programmable in the same way that computers are programmable today. This calls for a stronger convergence of computing, storage and communication within networks.

One of the goals of this Dagstuhl seminar was to assess the state of the art in active and programmable networks. To evaluate how this current technology supports rapid service creation for a diversity of services and applications. In this context, positive and negative experiences in applying active and programmable networking technology need to be addressed. Ultimately, a research agenda for future research in this area should be one important outcome of this seminar.

The seminar brought together researchers and engineers who have gained experience in different aspects of active and programmable networks.

Areas of interest include the following:

Experiences with prototypes and testbeds

Dynamically deployable services

Service location and description

QoS support mechanisms

Congestion control and traffic engineering

Multicast and group communication services

Applications of active networks

Active networking architectures

Accounting and charging

Safety and security

Active signaling

Transition strategies

Interaction of mobile agents and active networks

Evaluation criteria and performance measures

7.2 Approximation and Randomized Algorithms in Communication Networks

Seminar No. **02251** Report No. **345** Date **16.06.–21.06.2002**
Organizers: Evripidis Bampis, Klaus Jansen, Giuseppe Persiano, Roberto Solis-Oba, Gordon Wilfong

During the week of June 16 - 21, 2002, the seminar on Approximation and Randomized Algorithms in Communication Networks was organized by E. Bampis (Evry, France), K. Jansen (Kiel, Germany), G. Persiano (Salerno, Italy), R. Solis-Oba (London, Canada), G. Wilfong (Bell Labs, Murray Hill, USA). 45 Participants came from universities or research institutes from Canada, Cyprus, Greece, France, Germany, Israel, Italy, Netherlands, Switzerland, United Kingdom and United States of America.

The recent progress in network technologies and availability of large distributed computer systems has increased the need for efficient algorithms for solving the diverse optimization problems that arise in the management and usage of communication networks. Technological developments in communication networks, like broad-band, all-optical, and ATM networks have made this area very interesting and important in recent years. They have also created new research directions and projects. The objectives of this seminar are of both theoretical and practical significance. The seminar aims to contribute to the theory of approximation, randomized, and on-line algorithms for problems arising in communication networks. It also has as a goal to explore the use of this theory in the solution of real world applications and in the development of practical algorithmic tools, thus fostering the cooperation among theoretical and practical researchers in this field.

The topics of the seminar included: routing and communication in networks, design of high performance networks, wavelength routing in optical networks, ATM network problems,

quality of service, robustness issues, frequency assignment in radio networks, time and resource constrained scheduling, scheduling with communication delays, load balancing, and resource allocation.

The seminar was intended to bring together researchers from different areas in combinatorial optimization and from applications. It would support the collaboration between researchers in Computer Science, Engineering, Mathematics, and related areas. Different algorithmic methods and techniques have been covered by 31 lectures.

The seminar had the following goals:

- pose new optimization problems arising from applications in communication networks,
- design improved approximation algorithms for optimization problems in communication networks,
- study new algorithmic methods using randomization, linear, and nonlinear programming,
- discuss the practical implementation of different techniques and methods proposed for solving network communication problems,
- exchange information on recent research and stimulate further research in this area.

7.3 Performance Analysis and Distributed Computing

Seminar No. **02341** Report No. **349** Date **18.08.–23.08.2002**
Organizers: Michael Gerndt, Vladimir Getov, Adolfy Hoisie, Allen Malony, Barton Miller

The performance of parallel and distributed systems and applications - its evaluation, analysis, prediction and optimization - is a fundamental topic for research investigation and a technological problem that requires innovations in tools and techniques to keep pace with system and application evolution. This dual view of performance "science" and performance "technology" jointly spans broad fields of performance modeling, evaluation, instrumentation, measurement, analysis, monitoring, optimization, and prediction.

Most of the past and current research on performance analysis is focused on high-performance computing using dedicated parallel machines since performance is the ultimate goal in this environment. Future applications in the area of high-performance computing will not only use individual parallel systems but a large set of networked resources. This scenario of computational and data grids is attracting a lot of attention from application scientists as well as from computer scientists. In addition to the inherent complexity of program tuning on parallel machines, the sharing of resources and the transparency of the actual available resources introduce new challenges on performance analysis systems and performance tuning techniques. To meet those challenges, experts in parallel computing have to work together with experts in distributed computing. Aspects such as network performance,

quality-of-service, heterogeneity, and middleware systems - just to mention a few - will have a big impact on the performance of grid computing.

Therefore, the workshop brought together people from high-performance and distributed computing to discuss the impact of the new aspects of grid environments on performance analysis tools and techniques.

The topics covered in the workshop came from six areas:

1. Grid Computing

   The presentations concentrated on programming aspects of Grids. In addition, UNICORE was presented as an representative Grid architectures as well as performance aspects of Web servers were introduced.

2. Parallel Architectures

   This area covered quite diverse aspects, such as mobile agents, cellular architectures in the context of the IBM Blue Gene project, and the Quadrics interconnection network being part of the ASCI Q machine.

3. Performance Analysis Tools and Techniques

   Two major aspects where covered in this area: scalability of performance analysis tools and tool automation. Other presentations covered performance analysis for Java, performance visualization, and performance data management.

4. Performance Modeling

   Performance prediction and its application in different contexts, such as DSM multiprocessors, large scale systems, task parallel programs and grid computing was the focus of the workshop in this area.

5. Performance Analysis and Grid Computing

   The presentations in this area gave an overview of the current approaches on monitoring and performance analysis in several grid projects, i.e. Crossgrid, Datagrid, and DAMIEN.

6. Performance Optimization

   Performance optimization is the major reason for performance analysis. In Grid environment, optimization mainly means optimizing scheduling decisions in dynamic and heterogeneous environments as well as online performance tuning or performance steering.

The presentations during the seminar led to two evening discussions on *Grid Computing* and on *Future Architectures*. Major open questions raised in the Grid Computing discussion were:

- Will QoS ("Quality of Service") become reality in the context of grids? The opinion of the group was that this depends fully on economic reasons. If people will pay for using the grid, performance guarantees are required.

- Does the analogy of the Grid and the electrical power grid hold? Two major differences were identified: users of resources pay for those resources and, second, users transfer information into the grid which raises major security problems.

- How will Grids be used? The majority of people favored the concept of virtual organizations as the main usage of the Grid instead of metacomputing applications. The major programming paradigm might be a component based approach.

In the area of future architectures many questions were raised:

- How will the available chip space be used? Several approaches were suggested as possible candidates, such as combining scalar and vector processing, processor in memory systems, as well as multiprocessor and multithreading architectures.

- When will Quantum Computing become reality? The prevaling opinion seemed to be that this technology will take at least another 50 years.

- What will be the role of reconfigurable architectures?

The workshop highlighted that multiple approaches are currently pursued for grid monitoring. The Global Grid Forum defined the Grid Monitoring Architecture (GMA) which is based on a directory service for detecting producers of monitoring data and consumers requiring specific data. The data transfer itself, between producer and consumer, is realized via individual connections.

In all three projects presented in the workshop, i.e. DAMIEN, Datagrid, and Crossgrid, system-level and application-level monitoring are not integrated. Only the Datagrid project uses a unified infrastructure for system-level and application-level monitoring, the Relational Grid Monitoring Architecture (R-GMA). But, the information on system-level is not taken into account in performance analysis with GRM/Prove. This integration is the only way to assess performance data measured on application-level, i.e. to answer the question whether bad performance results from application coding or from dynamic changes in the resource capabilities.

Only with the assumption of QoS on the target computers and the network performance, analysis for grid applications ignoring system-level information makes sense. The DAMIEN approach is based on QoS and thus pure application-level monitoring can be used to analyze the application and grid component information with the help of VAMPIR.

A closer integration of system-level and application-level monitoring as well as the integration of runtime performance analysis and performance optimization (performance steering) will be very important in grid environments and will certainly be the focus of the future work of the APART working group (www.fz-juelich.de/apart).

7.4 Formal Circuit Equivalence Verification

Seminar No. **02352** Report No. **351** Date **25.08.–29.08.2002**
Organizers: J. Moondanos, A. Kühlmann, K. Sakallah, W. Kunz

Recent advances in silicon fabrication technologies, clearly suggest that Moore's law will continue to hold for the next 10-15 years. During this timeframe we will progress from current microprocessors comprising of a couple hundred million transistors towards designs with 1 billion transistors. Consequently, we can safely expect to see tremendously more complicated designs, as we try to exploit the ever-increasing VLSI fabrication capabilities. To make the design of such higher performance microprocessors feasible, CAD tool flows are expected to be drastically changed to allow for more abstraction levels higher in the circuit representation hierarchy. This is necessitated by the fact that Design Validation can be performed faster at a higher level of abstraction. As a result, the microprocessor design process will become an even longer sequence of circuit model transformations.

Hence ascertaining that the microprocessor's functionality is kept unaltered throughout its representation hierarchy will remain a fundamental problem in the design process. Formal equivalence verification techniques have had tremendous success in solving this problem in the last decade. Such techniques are based on mathematical frameworks that conclusively guarantee the equivalence of circuit models, contrary to simulation based approaches. Nevertheless, formal equivalence verification techniques can be limited from space and time complexities that grow exponentially with circuit size in the worst case. The simplest of these formal equivalence approaches require that the corresponding number and placement of memory elements in the circuits under examination are identical. In this case we have an instance of the combinational circuit equivalence problem. This assumption of matching state encodings is applicable only to the comparison of circuit models whose levels of abstraction are not very different. To accommodate the future design methodologies we need to do away with this restriction for enabling the efficient comparison at significantly different levels of abstraction. This requires solving the more general problem of sequential circuit equivalence. In this proposed seminar, the theoretical and practical aspects of the most successful formal equivalence verification techniques will be examined for both the combinational and sequential equivalence checking problems.

The corresponding presentations and discussions will cover the new trends in formal equivalence techniques from many different fields. Equivalence Checking techniques found their way in the mainstream of circuit design CAD tool flows with the maturing of BDD based algorithms. BDDs are canonical representations that allow for extremely efficient comparison of logic functions, but they may suffer from exponential memory requirements. We will review the latest results in this area, including BDD based techniques that exploit structural similarities between circuits under verification to reduce the complexity of combinational equivalence. BDDs as compact representation of transition relations and output functions have enabled Symbolic Model Checking (SMC) techniques for sequential circuit equivalence checking. We will examine the state of the art in Model Checking techniques, where the state space of the product machine is traversed to establish the equivalence of the circuits under comparison. To overcome the worst-case exponential space require-

ments of BDDs (which also limit the applicability of SMC techniques), researchers have turned to Boolean Satisfiability (SAT) solvers to compare circuit logic functions. SAT solvers completely and exhaustively enumerate the input variable space of circuits to solve the problem of combinational equivalence verification with impressive results lately. In addition, SAT solvers are effectively used in Bounded Model Checking approaches to address the problem of sequential equivalence checking. BDD and SAT based algorithms are fundamentally functional analysis methods and to improve the effectiveness of formal equivalence checking tools researchers have also focused on structural based techniques. So in addition to our focus on SAT and BDD based solutions, we will go over equivalence techniques based on automatic test pattern generation (ATPG). These exhaustive methods operate directly on the circuit structure trying to establish equivalence, in a manner that has evolved from the algorithms used in the manufacturing testing of circuits. Finally, we will focus on integrated approaches that attempt to combine all these techniques. Due to the computational complexities in equivalence checking, no individual technique has been proven completely successful. As a result, researchers have been combining different technologies to make the problem of formal circuit equivalence checking tractable.

Given the above list of topics that will be covered, the goals of the seminar become evident. Initially we wish to review the recent advancements in the core algorithmic solutions for the problem of equivalence verification. Subsequently, we plan to evaluate their scalability in light of the experience accumulated by the design of the complex microprocessors of today. Finally, we would like to motivate further development and integration of formal equivalence techniques according to the emerging trends of future microprocessor designs. The presence of many leading researchers from academia and industry is expected to provide the collaborative framework necessary for capturing the state of the art and clarifying the key technology and methodology challenges that lie ahead in the field of formal circuit equivalence verification.

7.5 Quality of Service in Networks and Distributed Systems

Seminar No. **02441** Report No. **358** Date **27.10.–31.10.2002**
Organizers: A. Campbell, S. Fischer, K. Nahrstedt, L. Wolf

Scientific Highlights of the Event

Distributed multimedia systems are becoming more and more important in many situations of our daily life, for instance in office applications (video conferencing), learning environments (tele-teaching and tele-learning, virtual universities), or entertainment (online games, video-on-demand). Usually, some of the media types used in such an application have specific requirements on their transmission and presentation. The notion of Quality of Service (QoS) plays a central role when discussing about how to fulfil these requirements of multimedia applications. Distributed multimedia systems need QoS support in order to

function properly. Moreover, other applications such as certain simulation systems need QoS functionality as well.

For this reason, research in QoS has increased significantly during the past few years. For an end-to-end QoS, which is necessary in most applications (user to user), support has to be provided in all components of the participating systems, i.e., the endsystem components, the communication system and the application. Accordingly, there has been active QoS research in network hardware (switches, routers), protocol software (RSVP, RTP etc.), operating systems (CPU scheduling), user interfaces, etc. Today, some of the basic technical issues are understood, but a significant amount of work is still necessary. Furthermore, additional research is devoted to (partially) non-technical issues such as pricing for QoS, but also new technical developments such as Active Networks.

The Dagstuhl seminar on "Quality of Service in Networks and Distributed Systems" gave an excellent overview on the state of the art in QoS research. It featured 23 talks which dealt with most of the above-mentionned research topics. Included were talks on QoS right on the network level, especially in wireless networks, such as those from Stefano Basagni from Northeastern University on QoS in Bluetooth networks or from Jörg Diederich, TU Braunschweig, on a simple and scalable handoff prioritization scheme in mobile networks. On the other end of the spectrum, a number of application-oriented approaches was presented, such as the one given by Torsten Braun, University of Berne, on IP Telephony over Differentiated Services or by Ralf Steinmetz, TU Darmstadt, on media semantics. And in between these extremes, many other topics were covered, such as middleware issues, ad-hoc networks and many more.

European Added Value

The European way of standardization has proven successfully, for example, for second generation mobile networks such as GSM, which is the world's most successful system. This will certainly hold for the currently built-up third generation UMTS networks, as well. However, this development has to be continued, for example, since the current release of UMTS networks still does not incorporate true Quality of Service mechanisms. End-to-end mechanisms even involve not only mobile networks, but also fixed networks. Therefore, it is of great value if researchers in the area Quality of Service from all over Europe and, additionally, from further countries like the USA exchange their ideas so that a common notion of Quality of Service, using common mechansims to implement them, is available in the final stage. This Dagstuhl seminar has been a small, but possibly important step into this direction, providing a communication platform and a creative environment to gather a broad spectrum of ideas about Quality of Service and its future.

Public outreach

To enable new applications such as Video-on-demand, telephony over the Internet etc., it is fundamental to provide QoS support. Although QoS has been a research topic over the last years, there is currently no approach, which fulfills all requirements and which has

attracted interest from the industry. Many solutions are simply to complex to implement in a real-world scenario. Hence, many of these applications which may become the 'killer application' of tomorrow, have still not become widespread. For this reason, it is still highly important to deal with QoS intensively in future research.

Chapter 8

Modelling, Simulation, Scheduling

8.1 Grand Challenges for Modeling and Simulation

Seminar No. **02351** Report No. **350** Date **25.08.–30.08.2002**
Organizers: R. Fujimoto, W.H. Lunceford, E.H. Page, A. Uhrmacher

The identification and pursuit of Grand Challenges has been a hallmark of the high performance computing arena for well over a decade. In recent years, many other technical communities, including the modeling and simulation (M&S) community, have begun defining Grand Challenge problems for their disciplines. While Grand Challenges themselves provide a useful focal point for research and development activities within a discipline, perhaps more important is the community dialogue that surrounds the formulation of Grand Challenge problems.

Within the M&S community, the dialogue surrounding the notion of Grand Challenges began with the First International Conference on Grand Challenges for Modeling and Simulation, which was held 27-31 January 2002 in San Antonio, TX, USA as part of Society for Computer Simulation (SCS) 2002 Western Multiconference. The conference program consisted of 15 papers and a panel.

The Dagstuhl seminar on Grand Challenges for M&S was dedicated to continuing this dialogue, with the goal of condensing ideas into a set of Grand Challenge problem statements that might serve to guide strategic research initiatives in modeling and simulation for the next decade.

The seminar was structured around various application and methodological areas of modeling and simulation:

- Simulation of cellular systems
- Simulation of air traffic
- Simulation large scale computer networks
- Simulation as part of agent-oriented software engineering

- Simulation in virtual manufacturing
- Simulation in military applications
- Parallel and distributed simulation
- Modeling and simulation methods

While the groups had unique perspectives derived from their particular application domain, they also shared a commonality derived from the modeling and simulation life cycle (i.e. understand a system, represent a system as a model, execute the model, analyze the results). Cognitive models of human actors, their decision processes, and their behavior are important in military applications, and in testing autonomous agent software. However, cognition processes are still little understood and "of the shelf" cognitive models that can be re-used in different settings do not exist. The same is true if we are looking at cellular, biological systems. The successful completion of the ambitious endeavor of the human genom project depends to a large degree on a better understanding of the behavior of cellular systems.

In dealing with complex systems, like cellular or cognitive systems, modeling and simulation has often played a role to support the development of theories and understanding of systems rather than predicting the systems' behavior. Efforts of the application area have to be combined with developing simulation systems that support an explorative approach to modeling and simulation more effectively. Whereas many techniques, e.g. hierarchical decomposition, object-oriented modeling and programming, graphical depiction of system behavior, visual modeling and programming, or agent based modeling, have enhanced our ability to build and use complex models, despite efforts like HLA, still the challenge of re-usability of models seems largely unresolved, particularly if we are approaching the realm of multi-paradigm, multi-resolution modeling. Supporting multi-paradigm, multi-resolution modeling is arguably a central prerequisite to significantly advancing modeling and simulation in such diverse application areas like manufacturing, military, air traffic, biology, software development, and networks.

Complex systems, e.g. the world wide web, do not only require new techniques for a more effective representation of systems. The efficient execution of these models poses unsolved problems as well. New parallel distributed simulation methods are needed not only to support an efficient simulation but to adapt themselves flexibly to the changing demands of a multi-resolution and multi-paradigm modeling.

During the seminar, a set of Grand Challenge problems statements from each of the application areas was formulated, and in some cases, possibilities for research agendas were sketched. While the results of the seminar offer a good starting point, and illustrate a number of intersections of interest across M&S application domains, more thought and effort is required to develop concrete research agendas in the multi-disciplinary arena of modeling and simulation.

Organization

Dagstuhl is dedicated to working groups. In contrast to traditional conference settings, the schedule offered plenty of time for working groups, discussions, and spontaneous activities.

The week was divided into two parts (1-4, and 5-8 respectively) and allowed everybody to participate in two working groups during the seminar. To give an overview about the different areas, state-of-the-art plenary talks were given. Short presentations provided the opportunity for each participant to present his or her work, and ideas on Grand Challenges for Modeling and Simulation before the parallel working groups started. In plenary sessions the results of the working groups were presented. Intertwining working groups and plenary sessions helped to work on concrete challenges in the different groups and to support a cross fertilization among them. The seminar was a truly interdisciplinary event and all participants played an active role in driving the progress and content of the workshop.

8.2 Scheduling in Computer and Manufacturing Systems

Seminar No. **02231** Report No. **343** Date **02.06.–07.06.2002**
Organizers: J. Blazewicz, E. Coffman, K. Ecker, D. Trystram

The objective of the seminar was to provide a forum for the discussion of current and proposed research in scheduling problems. It covered the entire spectrum from case studies of real applications to recent advances in mathematical foundations.

The seminar did not address only classical application areas such as distributed processing, operating systems, dependable systems, flexible manufacturing, etc. but also exciting new areas such as those in modern communications, examples being wireless networks, multimedia networks, and the internet.

The seminar proceeded along three broad fronts:

(i) *applications*, which includes empirical studies of existing systems as well as numerical studies of the analyses or simulations of system models;

(ii) *algorithmics*, which includes the design and analysis of perhaps randomized algorithms ranging from simple and tractable on-line and greedy rules to methods based on semi-enumerative approaches, branch and bound, local neighborhood search, LP formulations, etc.;

(iii) *theory*, which includes recent results in complexity classification, approximability, approximation schemes, analysis of classical problems under novel (or multiple) criteria, etc.

(i) *Applications.* This topic includes both, empirical studies of existing systems in various application areas as well as numerical studies of the analyses and simulations of system models, and the study of the new problems arising in actual applications on new systems like cluster computing and grid computing. New characteristics like heterogeneity of the resources, large communication delays and hierarchy of scheduling have been investigated.

In particular, the areas of application cover *parallel and distributed system.* This component deals with methods of analysis and modeling of distributed and parallel systems. Questions such as communication in MIMD systems, mapping program graphs and task

systems onto various processor topologies, and load balancing are covered by the contributions.

Manufacturing and production systems. This is the second broad area of applications where scheduling theory contributes important methods of modeling, analysis and algorithms. Shop problems in their most general form concern the manufacturing of a set of jobs on a set of machines where each job of the production process is characterized by its specific machine order. Contributions dealing e.g. with the optimization of production in flexible manufacturing systems and production centers with given numbers of parallel machines and as well optimization of robot control are welcome.

(ii) *Algorithmics.* This subtopic is mostly concerned with an algorithmic approach to scheduling problems. The unified framework for the presentations is the concept of computational complexity of combinatorial problems. The analysis of scheduling problems arising in computer systems and computer controlled manufacturing systems, and the adequacy of heuristic algorithms for solving these problems as well, have been discussed at the seminar. Additionally, methods employing artificial intelligence for solving some of the applications are covered by the seminar. This techniques are important to create a general tool for solving broad classes of practical problems.

(iii) *Theory.* This topic includes recent results in complexity classification, approximability, approximation schemes and heuristic approaches, and the analysis of classical problems under novel (and multiple) criteria.

Chapter 9

Mathematics, Cryptography

9.1 Mathematical Structures for Computable Topology and Geometry

Seminar No. **02221** Report No. **342** Date **26.05–31.05.2002**
Organizers: Ralph Kopperman, Mike Smyth, Dieter Spreen

Topological notions and methods have successfully been applied in various areas of computer science. Computerized geometrical constructions have many applications in engineering. The seminar we propose will concentrate on mathematical structures underlying both computable topology and geometry.

Due to the digital nature of most applications in computer science these structures have to be different from the mathematical structures which are classically used in applications of topology and geometry in physics and engineering and which are based on the continuum. The new areas of digital topology and digital geometry take into account that in computer applications we have to deal with discrete sets of pixels.

A further aspect in which topological structures used in computer science differ from the classical ones is partiality. Classical spaces contain only the ideal elements that are the result of a computation (approximation) process. Since we want to reason on such processes in a formal (automated) way the structures also have to contain the partial (and finite) objects appearing during a computation. Only these finite objects can be observed in finite time.

At least three types of computationally convenient structures for topology have been studied, and all of them may be developed in the direction of geometry. The first is domains, the second locales (and formal topology), and the third cell complexes.

Domains, originally introduced by Dana Scott for the formal definition of programming language semantics, have recently found a broader field of applications. Domain theory provides interesting possibilities for exact infinitary computation. There are the "maximal point models". The interval domain in which the real numbers are embedded as maximal elements is an example of this. Escardó has used it for his development of a programming

language that allows computing with intervals. But there are also domain models for convexity and intended applications in computer-aided design (Edalat et al.).

Closely related to domain theory is the theory of locales (Coquand, Resende, Vickers, ...) and its logical counterpart: formal topology (Martin-Löf, Sambin, ...). Here, one takes a constructive attitude and starts from the already mentioned fact that only the finitely describable properties of the ideal mathematical entities we are interested in are observable. Thus, these properties are the primary object of study. The ideal entities are obtained as derived objects. Formal topology is an open system based on Martin-Löf's type theory that allows the derivation of topological and (at the moment only to a certain extent) geometrical statements.

In geometry a similar approach, bypassing mainstream mathematics, tries to develop a system suitable for "commonsense" spatial reasoning, by taking regions rather than points as the basic entities. Developed initially by philosophers under the name mereology (and later, mereotopology), this viewpoint has been taken up by researchers interested in applications in AI, robotics, and GIS. The best-known product of this recent work in computer science is the so-called RCC (region connection calculus), also named for its originators Randell, Cohn and Cui.

This region-based topology/geometry has remained rather isolated from those mathematical disciplines which might be expected to interact fruitfully with it. In particular, there is an obvious analogy with point-free topology (as above), but there has been relatively little interaction so far. (It is one intention of the seminar to help changing this.) Again, the region-based theories have typically had the infinite divisibility of space built in; attempts at a discrete version are few. Here one may expect that cell complex theory in which, after all, the cells are usually thought of as convex regions could help.

Combinatorial topology offers us discrete (or finitary) structures which have long played a part in image processing: cell complexes. These may be either "concrete" (derived explicitly from Euclidean space, or more generally from a manifold), or "abstract". The concrete complexes do not provide us with the autonomous theory we are looking for; the abstract complexes permit the computation of various topological invariants, but do not support specifically geometric features such as convexity and linearity. To make progress, it seems that we need either to endow the abstract complexes with suitable extra structure, or else to ground them in some richer combinatorial structures (not classical manifolds).

In the latter connection, it is worth mentioning oriented matroids. Despite pioneering work by Knuth (1991), these have been almost completely ignored by computer scientists. Briefly, a matroid can be described as a simplicial complex with just enough extra structure to handle linear dependence. An oriented matroid then admits just enough further structure so that one can deal with convexity as well as linearity. It seems likely that oriented matroid theory will have a significant input to the (eventual) foundations of digital geometry, even if this has been little recognized so far.

The aim of the workshop was to bring together people working in fields like domain theory, computer science oriented topology and geometry, formal topology, ... and to foster interaction between them. 57 top scientists and promising young researchers accepted the invitation to participate in the challenging experience. They came from 16 countries,

mostly European countries and the USA, but also China, Japan, Mexico, New Zealand, Russia, South Africa and Turkey. The 45 talks covered all of the areas mentioned above.

The workshop was a great success. Many new cooperations were started. The particpants expressed high appreciation of this gathering and praised the extraordinary Dagstuhl.

9.2 Cryptography

Seminar No. **02391** Report No. **355** Date **22.09.–27.09.2002**
Organizers: Ueli Maurer, Adi Shamir, Jacques Stern, Moti Yung

Since the advent of public key cryptography, about twenty-five years ago, the field of cryptography has been developing very rapidly. Constantly, there are new areas and new issues to investigate. The advance of the Internet as the major computing paradigm has increased the applicability and diversity of the field.

The aim of the 2002 Cryptography seminar was to provide an opportunity to focus on the scientific foundation of cryptography, to spot the emerging new areas based on recent advances in theory and technology needs, and to work on them.

The emphasis of the seminar was on the conceptual framework that allows the use of appropriate models, amenable to mathematical reasoning. Applications are natural in this field and were covered as well.

We note that earlier cryptography seminars at Dagstuhl were held in the fall of 1992 and 1997. Similar workshops were also held at Luminy, France, and Monte-Verita, Switzerland. Previous meetings have led to valuable exchanges and to various investigations that advanced the field. The present seminar continued this tradition, with renewed topics, suitable to the current state of the art, and with concentration on a number of subjects that are being developed nowadays.

The following is a non-exhaustive list of topics discussed during the seminar:

1. Provable security of encryption and signature schemes.
2. New functions for cryptographic applications: the mathematics behind the Weil and Tate pairings over supersingular elliptic curves.
3. Novel cryptographic applications based on the bilinearity of pairings.
4. The applicability of various proof methodologies (random oracle proofs, generic model) to validation of cryptographic constructions.
5. New paradigms for cryptographic primitives (neural cryptography, quantum cryptography).
6. Methods for trust distribution.

7. Security models for multi-party protocols for function evaluation over private inputs and for commitment schemes: universal composability, non-malleability, etc.

8. New multi-party and commitment protocols.

9. Public key infrastructure and key distribution protocols.

10. Distributed cryptography over the Internet and in mobile networks (Byzantine agreement, threshold cryptography, fair exchange in ad-hoc mobile networks, etc.).

11. Relations between cryptographic primitives.

12. New methods in electronic voting.

13. Security in data retrieval.

14. New methods in content protection.

15. New notions in security: formal steganography, anonymity mechanisms.

16. New algebraic methods of cryptanalysis and their applications.

17. The effect of emerging developments in quantum computing on cryptographic primitives.

18. New design and analysis methodologies for, and experience with block ciphers (including the recent AES standard) and stream ciphers.

19. Improved efficiency of cryptographic mechanisms.

20. The influence of emerging computing environments and modern computer networks on cryptography.

21. The global implications of the "trusted computing platform" environments, recently proposed by the industry.

During the seminar, new directions for theoretical and applied research have been identified in numerous areas. The formal lectures, as well as the informal discussions, the moderated discussion session, and the informal session on recent results, were all inspiring and highly productive.

We feel that the subjects we have covered and worked on are most likely to influence cryptographic research in the coming few years. Furthermore, they have the potential to have impact on future applications of cryptographic techniques in computing systems.

Abstracts selected by the Dagstuhl News editor:

Lifting Part of the Veil on the Murder Of Patrice Lumumba; or Breaking 1961 Hagelin Ciphertexts

Bart Preneel

In this talk we discuss some cryptanalytic work carried out for the Belgian Parlement in the Fall of 2001. The motivation for this work was the investigation carried out by the Parliament on the circumstances of the murder on Patrice Lumumba. We were provided with 15 enciphered telexes sent between December 1960 and February 1961 between Brussels and Elisabethville and Brazzaville. In addition, 5 likely plaintexts (with errors) were available. We describe how we were able to identify that the PRINTEX variant used was the Hagelin C-38. We also succeeded in recovering the key settings by improving the Morris algorithm published in 1978. We also identified and cryptanalyzed the mechanism to encrypt session keys (Playfair). One of the telexes, dating from a few days before the murder, revealed some interesting new information.

Chapter 10

Data Bases

10.1 Information Integration

Seminar No. **02181** Report No. **340** Date **29.04.–03.05.2002**
Organizers: V. Krishnamurthy, F. Leymann, N. Mattos, B. Mitschang

Information Integration subsumes all technologies needed to provide for manipulation of information scattered over many data stores while supporting a single system image. The data stores to be integrated are inherently heterogeneous in nature, owned by different organizations, and distributed over the whole world. Data can be structured (e.g. relational data), semi-structured (e.g. XML documents or hyper-linked HTML pages), or unstructured (e.g. opaque flat files, multi-media streams). Access to the data can be based on standardized interfaces (e.g. SQL) or via proprietary APIs (e.g. RYO solutions).

Information integration is expected to become a key technology in many application areas like product data management, business process management, enterprise application integration, life science (including drug design, health care management), or entertainment (e.g. media on demand) to name but a few. Software vendors begin to deliver first products, currently focusing on a particular application area. Research in Information Integration is currently done in different disciplines.

The major goal of the seminar is to bring representatives from the different communities (from research as well as from software vendors and from users) together for a first stock-taking, a joint in-depth understanding of the issues, to identify and prioritize the main research items, identify standardization needs, and to discuss demanding questions and open problems in detail. The areas to discuss include:

- How to get access to the various data stores?
  Different technologies like SQL/MED wrappers, J2EE connectors, EAI adapters, and Web Services can be used for these purposes. When should either of these technologies be used? Can they be unified?
- What are possible system structures?
  Which role will database systems, application server, workflow systems, messaging systems, portal servers, etc. play? How do they relate and cooperate?

- Does "Web Database Technology" suffice?
  Can XML be used as the language for describing the integrated information base? How to capture "navigational access" based on hyper-linked HTML pages performed today in many application areas? How to combine search and query functionality? How is XML stored - sliced/diced, as whole document as file in file system, as whole document but combined with other documents in file system? How do you index these effectively? How do you combine SQL and an XML-based query over the same data (i.e., XML query against SQL data and SQL against XML)? Is a pure XML database the way to go or will an extended relational engine be the right solution?

- How is information described?
  As different data stores are combined in a dynamic manner the quality of the information available in a data store becomes key. Which information qualities are needed? How are they described? How can qualities be compared, assessed, measured? Which metadata is relevant (schema, ontologies)?

- Which federated database technologies can be used?
  What is a federated schema if structured and unstructured data are brought together? Which schema integration techniques, federated query and search technologies are applicable?

- Which transaction model is appropriate?
  Some of the underlying data stores support classical transactions, others don't. Collective manipulation of data stores demands transactional guarantees. Which guarantees are needed? Data stores are owned by different legal entities and are often accessed via the Internet. Which concurrency models, recovery models are applicable?

With this seminar we would like to bring together, for the first time ever, people from different areas that all work on the broad topic of Information Integration. We can see the topic of Information Integration to range from application-oriented areas like geographic information systems or product management systems to generic areas in computer science like repository technology, database federation, or data exchange. It is assumed that the discussions in this seminar will provide a first step in the process of finding the needed solutions to the various forms of Information Integration. The participant list covers various well-known people as well as young scientists from both industry and academics. It is our hope that the seminar will improve the understanding of this field, and stimulate new collaborations between the different communities.

Chapter 11

Evolutionary Algorithms

11.1 Theory of Evolutionary Algorithms

Seminar No. **02031** Report No. **330** Date **13.01.–18.01.2002**
Organizers: H.-G. Beyer, K. De Jong, C. Reeves, I. Wegener

The previous Dagstuhl workshop on the "Theory of Evolutionary Algorithms" held in February of 2000 had a great influence on the development of this field and provided a unique opportunity for the people working in this area to interact with each other. Therefore, we had many people who were interested in the new workshop and we could not invite all who asked us.

The idea was to discuss the different approaches to a theory of evolutionary algorithms. The participants were researchers with quite different scientific background. People influenced by computer science, mathematics, physics, biology, or engineering came together which led to vivid and fruitful discussions. The organizers are happy to report that 40 researchers accepted an invitation to Dagstuhl. They came from Germany (13), USA (9), England (6), Belgium (2), Austria (2), India (1), Japan (1), Mexico (1),Netherlands (1), Romania (1), Russia (1), Spain (1), and Switzerland (1). The 32 talks captured all the aspects of a theory of evolutionary algorithms, among them EA-dynamics, non-static fitness and robustness, algorithmic aspects of EAs, recombination, fitness landscapes, global performance of EAs, and schema approaches. The schedule included an evening session showing "evolution strategies in action".

Chapter 12

Other Work

12.1 Rule Markup Techniques for the Semantic Web

Seminar No. **02061** Report No. **332** Date **03.02.–08.02.2002**
Organizers: H. Boley, B. Grosof, S. Tabet, G. Wagner

Rules have traditionally been used in theoretical computer science, compiler technology, databases, logic programming, and AI. The Semantic Web is a new W3C Activity trying to represent information in the World Wide Web such that it can be used by machines not just for display purposes, but for automation, integration, and reuse across applications. Rule markup in the Web has become a hot topic since rules were identified as a design issue of the Semantic Web. However, rule markup for the Semantic Web has not been studied as systematically as the corresponding ontology markup. This Dagstuhl Seminar was an attempt to fill the gap by bringing together researchers exploring rule systems suitable for the Web, their (XML and RDF) syntax, semantics, tractability/efficiency, and transformation/compilation. Both *derivation rules* (also called "inference rules") and state-changing *reaction rules* (also called "active" or "event-condition-action" rules), as well as any combinations, have been of interest to this effort.

This seminar has succeeded in bringing together leading researchers from the classical logic programming and knowledge representation community and from the Semantic Web community. The discussions at the seminar have been very productive, both scientifically and in terms of triggering new research activities such as a EU FP6 Network of Excellence initiative.

12.2 Electronic Market Design

Seminar No. **02241** Report No. **344** Date **09.06.–14.06.2002**
Organizers: D. Lehmann, R. Müller, T. Sandholm, R. Vohra

Introduction

During the week of June 9–14 the Dagstuhl Seminar on Electronic Market Design was held. The aim of the seminar was to provide researchers and practitioners in economics, mathematics and computer science working on topics in electronic trading, auction design, mechanism design and artificial intelligence with a platform where they could interchange results and ideas and profit from the achievements of each other's field of expertise.

Electronic market design is a new field of research that builds on theories from various established fields. The challenges of market design are to create rules for trading interaction, in particular for auctions, that lead to economically desired allocations of items and payments, and that are immune against manipulation by strategic behavior of the participants. When market design is implemented electronically, in principle more complex market designs are realizable because of computerized transactions. However, computational complexity increases rapidly and excludes therefore certain designs. These problems have led to plenty of research in computational issues of market design, which was widely presented at the seminar. At the same time, most electronic markets will still have human participants interacting with them. The impact of design on their behavior cannot completely be captured by theoretical models, but requires an empirical or experimental investigation. Finally, every (electronic) market design has to be embedded into a broader set of issues, for example the industrial environment in which it takes place, asking for careful economic considerations of the impact of design on market outcome in the short and the long term.

The Dagstuhl seminar on electronic market design has successfully provided a forum, in which world-wide leading researchers from these fields could exchange their ideas. The working atmosphere was excellent, in particular thanks to the perfect local organization in Dagstuhl. In the following we will give a short overview of the main topics addressed during the seminar and the impact the workshop has on the development of the field.

Scientific Overview

The seminar included 37 lectures as well as a rump session on diverse problems such as winner determination for combinatorial auctions, the trade-off between the informational and economic efficiency of markets, implementation of incentive compatible mechanisms as well as analyses of the strategic consequences of the design of real-life auctions on and off the Internet as they currently exist, in particular the UMTS spectrum auctions, eBay, and the electricity auctions.

Because of the diverse background of the participants the central issues in electronic market design were approached from many different angles. Specifically the following four different aspects of market design were discussed extensively.

1. *economic efficiency* From an economic perspective auctions and markets are instruments that can be used to allocate scarce goods in an efficient way, meaning that the goods are to be divided among the agents participating in the auction or the economy in such a way that the overall welfare of the agents is maximized. In for example combinatorial auctions or markets for perishable goods (with severe time constraints) enforcing economic efficiency may be a complicated or sometimes even impossible task.

2. *strategic behavior* Market design relies on the assumption that the agents engaged in trade within the market behave according to the ideas envisaged by the designer. Thus it is of crucial importance that the design is immune to manipulation by the participants in the market. Especially in electronic markets, where buyers and sellers engage in anonymity, shill bidding, sniping, and false-name bidding do often occur. It is an important task for designers to develop trading mechanisms that discourage manipulation of this type.

3. *computational complexity* Trade that takes place in a complex environment, such as a combinatorial auction or market for heterogeneous goods, puts a heavy computational burden on the buyers and sellers in such an environment. One of the problems market designers face is to develop trading mechanisms for these complex situations that are still transparent from the trader's perspective and nevertheless guarantee outcomes that also are sufficiently close to efficiency in the economic sense of the word.

4. *evidence from the field* Case studies from the day-to-day ongoing practice of electronic trade such as the sales on eBay, the FCC and UMTS auctions and the auctions for surplus electricity indicate that electronic market design is an area of research that is and still very much needs to be developed further. The call for faster, simpler and more robust designs is heard everyday and everywhere throughout the internet!

Related to each of these topics, most recent research results were presented. In comparison to a previous seminar, organized by the same organizers at the International Institute of Infonomics in the Netherlands (see www.etrade.infonomics.nl/workshop), several scientific breakthroughs were presented. For example the development of fast algorithms for combinatorial auctions has reached a strength such that these auctions can be used by the FCC in future spectrum auctions in the US. Experimental and empirical results let us better understand the strategic behavior in online auctions. Several presentations illustrated new iterative auctions with bundle bids. Furthermore, new insights in the interplay of complexity of communication, computation, and bidders decision making were presented, for example by Daniel Lehmann and Noam Nisan and their students from Hebrew University, researchers from Universiteit Maastricht, and from Tuomas Sandholm and his team from Carnegie Mellon University. Notably, most of these results strongly benefit from the interplay between economics, game theory and computer science. Without the fruitful exchange of ideas between disciplines, like it has been facilitated by this Dagstuhl seminar, many of them would not have been possible. Finally, this seminar contributed to lay out a research agenda on electronic market design for the following years. For example, it remains still a big puzzle how integer programming theory can be successfully applied to understand auction markets with budget constraints.

Impact

Very remarkable for the seminar was the intensity of communication between very different fields, and between junior and senior researchers. This communication is even more remarkable given the heterogeneous scientific background of the participants. Almost half of the lectures were given by young researchers, many of them still PhD students. For

many young colleagues, this was likely the first time that they got the opportunity to meet the senior colleagues from the field. All presentations enjoyed a large audience, and presenters got plenty of feedback, despite a dense schedule. At the same time it was observable that the tutorial speakers as well as other senior speakers had put a big effort into their presentations, such that PhD students could get a maximum out of it.

It has to be said that the field is still not very well developed inside Europe, at least when it comes to the theoretical foundation of electronic markets. European research seems to focus more on the adoption of technologies in business settings. Nevertheless several research groups were present (e.g., Maastricht, Karlsruhe, Kiel, München, Cambridge), including PhD students from these groups. For these groups, and for others in Europe hopefully too, the seminar provided certainly a large stimulation to catch-up with the international research agenda.

Partly inspired by the seminar, a European consortium on the field of market design is currently emerging, and preparing project proposals for the sixth framework program (see www.etrading-europe.org).