



SCHLOSS DAGSTUHL

**INTERNATIONAL
CONFERENCE
AND RESEARCH CENTER
FOR COMPUTER SCIENCE**

Dagstuhl News

January - December 1998

**Volume 1
1999**

ISSN 1438-7581

Copyright © 1999, IBFI GmbH, Schloß Dagstuhl, 66687 Wadern, Germany

Period: January - December 1998

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

The Scientific Directorate is responsible for the program:

Prof. Dr. Thomas Beth, Karlsruhe
Prof. Dr. Oswald Drobnik, Frankfurt
Prof. Dr. Klaus Madlener, Kaiserslautern
Prof. Dr. Christoph Meinel, Trier
Prof. Dr. Horst Reichel, Dresden
Prof. Dr. Peter H. Schmitt, Karlsruhe
Prof. Dr. Otto Spaniol, Aachen
Prof. Dr. Ingo Wegener, Dortmund
Prof. Dr. Reinhard Wilhelm (Scientific Director)

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/>

Preface

You have in your hands the first edition of the “Dagstuhl News”, a publication for the members of the Foundation “Informatikzentrum Schloss Dagstuhl”, the *Dagstuhl Foundation* for short. The main part of this leaflet consists of collected resumes and other valuable information taken 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 1998 are on the Web under URL www.dagstuhl.de/DATA/Seminars/98.

The State and the Activities of the *Dagstuhl Foundation*

The foundation currently has 44 personal members and 6 institutional members.

In 1998, the foundation has supported 5 guests with travel grants and a reduction of the Seminar fees. According to German law only the interests earned can be used to support the aims of a foundation.

Future Events

There are several announcements which may be of interest to you. On August 11, 1999 Dagstuhl will experience the total solar eclipse. For that reason we invite to a Dagstuhl-Seminar on Computer Science in Astronomy. Check the following web page if you are interested:
www.dagstuhl.de/DATA/Seminars/99/sofi.html

During the week of August 28 til September 1, 2000, Dagstuhl will celebrate its 10th anniversary. We will organize a conference on the Perspectives of Computer Science and a big birthday party. At the conference, leading computer scientist from academia and industry will present their views of the state of our field and the vision they have for its future.

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 resumes collected in this volume.

Reinhard Wilhelm (Scientific Director)

Contents

1 New Media in (Computer) Science Teaching at University Level	7
2 Scenario Management	10
3 Information Systems as Reactive Systems	11
4 Dynamically Reconfigurable Architectures	15
5 The Fourth Dagstuhl Seminar on Data Structures	16
6 Continuous Engineering for Industrial Scale Software Systems	18
7 Program Comprehension and Software Reengineering	20
8 Theoretical Foundations of Computer Vision: Evaluation and Validation of Computer Vision Algorithms and Methods	21
9 Advanced Stochastic Modelling Applied to Telecommunication-Networks and Distributed Systems	28
10 Programs with Recursively Defined Data Structures Using Pointers	29
11 Generic Programming	34
12 Domain Theory and Its Applications	36
13 Quantum Algorithms	38
14 Algorithms and Complexity for Continuous Problems	41
15 Hierarchical Methods in Computer Graphics	44
16 Programs: Improvements, Complexity, and Meanings	45

17 Real Computation and Complexity	47
18 Modeling and Simulation of Gene and Cell Regulation and Metabolic Pathways	48
19 The Semantic Challenge of Object-Oriented Program- ming	53
20 Petri Nets and Business Process Management	55
21 Semi-Formal and Formal Specification Techniques for Soft- ware Systems	57
22 Graph Algorithms and Applications	58
23 Self-Stabilization	59
24 Tiling for Optimal Resource Utilization	61
25 Architectural and Arithmetic Support for Multimedia	63
26 Test Automation for Reactive Systems – Theory and Practice	64
27 Modeling and Planning for Sensor-Based Intelligent Robot Systems	66
28 Algorithms and Number Theory	71
29 Adaptation and Evolution in Embedded Information Sys- tems	72
30 CAD-Tools and Methods for Systems Development	74
31 Integrating Spatial and Temporal Databases	75

1 New Media in (Computer) Science Teaching at University Level

Seminar No. **98051** Report No. **198** Date **01.02.–06.02.1998**
Organizers: Peter A. Gloor, Robert D. Harding, Thomas Ottmann

This workshop brought together 25 researchers from various disciplines with considerable experience in the use of networked computers for teaching science topics at university level. They came from Germany, Austria, Switzerland, UK, and the US.

Currently there is a flurry of activities in promoting networked computers, in particular the World Wide Web as new media for teaching at university level. There are many scientists and teachers active in the field who develop interactive web pages, animations and simulations for educational purposes and contribute to the establishment of electronic libraries. The extension of data networks to information highways, providing high speed communication facilities to university teachers and students has the potential of triggering a revolution in the way that teaching and learning at universities will be done in the future. In particular, computer scientists have always been at the forefront of this development. This became obvious also during this workshop. There were quite a number of convincing examples for using computers for teaching computer science topics. Demonstration of a web-based course on Crypto Systems (Kaderali), the interactive manipulation of advanced geometric structures (R. Klein), and the visualisation of abstract concepts in compiler design (St. Diehl) are just a few examples. During this seminar it was also demonstrated that not only computer science can gain quite a lot from the new facilities. Similar experiences were also reported from Mathematics, Chemistry and even Humanities.

Participants of this workshop not only gave reports about their work and experience in the traditional way by a talk using blackboard, chalk, and transparencies. Almost everybody also gave at least a demo of software or courseware, respectively, illustrating the work. A few participants even participated in the experiment of getting their presentation recorded and post-processed for later replay by using new tools.

Besides the presentation of new multimedia content as CD-ROM or web based products or as a combination of both, the current status of the tools and their development played a major role. It became apparent that

we are still far away from the ideal situation that producing multimedia documents for teaching at university level is as easy as producing a purely paperbound document.

Experiences in using networks for teleteaching experiments were reported, the vision of how to extend the communication facilities provided by the web in order to establish a virtual university offering all and even more facilities traditional universities currently offer to their students was presented, and, finally, the political implications were discussed.

There was ample time for both formal and informal discussions and a final discussion round identified the following major issues.

1. Systems technology

One of the main problems in this area seems to be that people working in the field do not know enough about each other and about projects carried out at various places. That is, there is no established group communicating ideas and results such that all can benefit from each other. The impression of activities from web presentations seems to be quite different from what participants could see and experience during this workshop. Hence, even though it is important to be present in the web, workshops like this one bringing together people working in similar or related areas are highly welcome.

As one of the crucial points it was stated that establishing large repositories with multimedia teaching material would be desirable. However, it is time- and resource consuming to establish such a repository, to guarantee the quality of its content, and to make it accessible over a long time. This cannot be done without substantial support. It was discussed whether this might be a natural task for scientific organisations like the ACM or GI (in Germany) or whether it would be considered as a future service of publishing houses. For computer science and mathematics initiatives establishing those repositories have been established both in Germany and in the UK the Association for Learning Technology (ALT) is a centralized interest group which seeks to bring together all those with an interest in the use of learning technology in higher and further education.

Further information on the ALT can be found under <http://www.warwick.ac.uk/alt-E>

and the CTI centres for computers in teaching are found under <http://www.cti.ac.uk/centres/>.

It became very clear that we are still far away from the ideal situation that university teachers can rely on repositories of easy to use, high quality, well designed material which they can directly incorporate into their on-line or off-line courses.

It became furthermore apparent that authoring tools are extremely diverse and not integrated into a single useful system. What is necessary is to observe modularity, to be able to guarantee link consistency. Authoring tools must be suitable both for the web and for producing CD-ROM publications. They should be platform independent, scaleable and allow the use of existing modules.

2. A role of technology and its impact on teaching

Most participants said that the use of multimedia material and of networked computers certainly influences their teaching style. There was agreement that using such materials does not make traditional lectures obsolete. However, such use may change the role of university teachers such that they become information mediators more than presenters.

3. Differences between humanities and science (including mathematics)

In humanities lectures do not play such a central role as they do in science and mathematics. Humanities are much more interested in libraries. On the other hand, the use of video, photography, and other visual media seem not to play such a crucial role in computer science as they do in humanities, medicine, biological sciences and so on.

On the other hand there are many similarities between traditional teaching and teaching using new media. For example courses of the drill and kill type or following established didactic principles for teaching play a similar role in both worlds.

4. Political issues

Here it was mentioned that some kind of public awareness is necessary in order to support research in this area. As a very good example of how web-based material can be used for promoting a subject,

the on-line magazine developed in Cambridge was presented. Participants agreed that a similar on-line journal for computer science would help to give both students at high schools, and politicians a better insight into the field and may lead to the result that the currently observed decrease in enrolments can be overcome.

Finally it was considered as an important task of the community to provide politicians in particular with understandable information such that they can estimate the current state of development in networked multimedia and its use for education and thus get a better realistic view of this field.

2 Scenario Management

Seminar No. **98061** Report No. **199** Date **08.02.–13.02.1998**
Organizers: X. Tung Bui, John M. Carroll, Matthias Jarke

A scenario can be defined as a description of a possible set of events that might reasonably take place. The purpose of scenarios is to stimulate thinking about possible occurrences, assumptions relating these occurrences, possible opportunities and risks, and courses of action. Recent surveys of scenario research and practice undertaken by the European CREWS project show the enormous variety, but also the fragmentation of the field. For example, HCI researchers use scenarios to enhance user-designer communications and managers use scenarios to explore alternative futures and the impact of systems. Software engineers see scenarios as a promising means to discover user needs that are not obvious in analysis situations, to better describe the use of system in work processes, and to systematically explore normal-case and exceptional behavior of a system and its environment.

Researchers from other disciplines have used scenario analysis for a long time. Economists have successfully used scenarios for long-range planning. Management scientists use scenarios for strategic decision-making. Policy makers use scenarios to weigh the possible consequences of their actions. In an interdisciplinary perspective, scenarios are used to examine the interplay among economic, social, and technological issues.

This Dagstuhl Seminar convened twenty-four leading researchers and practitioners from various disciplines to cross-examine the effectiveness

and efficiency of using scenarios as a modeling, design, development, and implementation tool. A second issue of interest is the management of scenarios as complex artifacts throughout the planning and systems lifecycle.

The seminar was organized by the CREWS ESPRIT project in cooperation with the IFIP Working Group 2.9 (Requirements Engineering) and the RENOIR European Network of Excellence. It comprised plenary sessions as well as subgroup discussions. The results of the workshop discussions are summarized in an overview article by the organizers, titled “Scenario Management: An Interdisciplinary Approach” (in the Dagstuhl report). 13 individual research contributions by the participants and others, elicited through an open Call for Papers, are published in Special Issues of the IEEE Transactions on Software Engineering (vol. 24, no. 12, December 1998) and of the Requirements Engineering Journal (vol. 3, no. 3-4, December 1998).

3 Information Systems as Reactive Systems

Seminar No. **98071** Report No. **200** Date **15.02. – 20.02.1998**
Organizers: Hans-Dieter Ehrich, Ursula Goltz, José Meseguer

The specification, development and use of distributed information systems is an important research area with many practical applications. Examples for existing such systems are those owned by banks, airlines or governments. Moreover, the explosive growth of the Internet is evidence that a global information structure is developing, and expansion in the use of large distributed information systems can be predicted.

Information systems are *reactive* systems. Unlike transformational systems—that is, systems whose function is to transform some inputs into output results—reactive systems have an unlimited number of interactions with their environment. Basic notions of program correctness do not apply, and techniques for designing transformational systems do not easily carry over to reactive systems.

Non-distributed reactive systems, like conventional operating and database systems, are not easy to design, implement and restructure, a lot remains to be done to develop helpful high-level concepts, features, languages, methods and tools. However, existing products can already

cope successfully with reasonably complex systems in practice. Formal methods from logic, algebra and process theory have been successfully applied, for instance in query and transaction processing, but they are not in general standard engineering practice yet.

Distributed reactive systems are still an order of magnitude more complex and difficult to specify, analyse, implement and reconfigure. They are not well understood, and methods for designing, implementing and reengineering them have not yet reached a level of mature and sound engineering practice.

A great challenge is to cope with *concurrency, synchronization, and communication* at an appropriate level of abstraction. Process theory provides useful theoretical concepts for concurrency and interaction. But those providing high-level declarative specification concepts, like temporal logic, tend to be sequential in nature, while those capable of expressing full concurrency, like Petri nets, tend to be operational: the level of abstraction is low. A possible approach is to adopt concepts from process algebras which provide compositional system descriptions and formalize dynamic behaviour. However, many questions then arise. These questions have been a central issue of this seminar.

In particular, the following topics have been addressed:

Modeling and design methods. Information systems involve data, objects and global processes. In conceptual modeling, these aspects have been addressed in isolation and various combinations, and the same holds for the conceptual modeling perspectives: system architecture, object classes, object behaviour, and object communication. Diagrammatic modeling approaches like OOA&D, OMT, OOSE and their recent amalgamation UML employ a multitude of specification concepts and techniques that are not always well integrated and sometimes not well understood. The lack of semantic consistency among UML diagrams has been criticized, and approaches to giving formal semantics to UML have been discussed. Several contributions addressed the question of how to integrate formal specification techniques into pragmatic software development methods like UML or OOSE, respectively. Successful applications have been reported where the need arose to augment UML by formal specification techniques like pre- and postconditions or temporal-logic constraints. OOSE has been integrated with object-

oriented algebraic specification techniques by extending object and interaction diagrams with formal annotations.

Information systems are hardly ever built from scratch. One particular design issue that has been addressed is how to integrate existing databases into a federated database, concentrating on behaviour conflict issues.

Semantics of object-oriented specification languages. Describing the semantics of a formal language amounts to giving a map from syntactic constructs into a suitable semantic domain. For object-oriented specification languages, the domains under discussion were process algebras, Petri nets, event structures, the actor model, type algebra, and others. For example, it has been shown how an object-based language with intra-object concurrency can be translated into a process algebra.

Studies on how these semantic domains are interrelated have also been presented. The expressive power of process algebras and finite Petri nets has been compared by providing translations in both directions.

Among the language issues addressed were module composition, refinement and atomicity. The category-theoretic treatment of the parallel composition of objects with synchronous and asynchronous interaction has been discussed. Concerning refinement, it has been argued that a suitable notion of atomicity for method invocation allows the implementation of method invocation as action refinement in transition systems.

Logics for distributed systems. An attractive model for the execution of a distributed system is a *distributed process*, which has a time line for each location of the system and links the time lines by synchronization conditions like common events.

Rewriting logic, well known and quite successful in this field, has been augmented by adding synchronization conditions that are represented geometrically as the vertical sides of a tile. There is a multi-purpose specification language, Maude, that is based on rewriting techniques. Work is in progress to design an appropriate temporal logic which allows to formally verify the correctness of Maude programs.

Several contributions addressed the issue to bring distributed processes together with temporal logics which are widely accepted as an intuitive formalism for the description of the temporal order of events in the execution of a sequential program.

An overview was given of several generalisations of linear temporal logic for distributed processes and related models. A particular such logic has been presented which has the same expressive power as monadic second order logic or asynchronous automata, thus marking the boundary for automatic verification. Specific approaches were presented where local temporal logics are used for specifying object behaviour, and communication facilities are added for specifying interaction between concurrent sequential objects.

A notion of refinement for one of these logics has been presented and shown to be compatible with action refinement in process algebras.

A somewhat different approach was presented that is based on finite observations as communications between an observer and a system.

Application-oriented concepts in specification languages. Among the issues addressed were modules, real time, fault tolerance, software features, and internet applications.

For system development and design, concentration on objects as modules was criticized on the grounds that it only deals with system statics: parallel statements ought to be also considered as a kind of modules.

Real-time constraints set limits to when an action may or must occur or terminate, and how long it may take from a triggering event to the corresponding reaction. One real-time formalism was presented as it is used to specify systems in a Codesign approach. Another presentation gave a complete axiomatization of fully decidable propositional real-time linear temporal logics with past.

It has been shown that fault tolerant distributed services can be shown correct more easily by exploiting the causalities between actions. Related problems (database consistency, transaction protocols) can be treated similarly.

Features are software supplements providing additional functionality. The feature-interaction problem is the problem of detecting and resolving undesired interactions between features. It was argued that the feature-interaction problem is not solvable in general.

Mobility. For modelling large systems of communicating objects, it is indispensable to deal with a dynamically changing communication structure. In process algebras, this issue has been formally dealt with in the π -calculus. Results on characterising equivalences for the π -calculus have been presented. The fusion calculus, which was also presented at the seminar, has recently been developed starting from the π -calculus; it is at the same time conceptually simpler and more expressive. A kernel programming language for mobility, called KLAIM, was presented.

A central topic of concern in talks as well as in many informal discussions was the newly developed specification formalism UML (Unified Modelling Language). Both its usefulness for applications and the problems of giving semantics to various constructs of UML were intensively discussed.

Until this seminar, there was little interaction between the communities concerned with information systems and concurrency theory. One outcome of the seminar is that there is a definite need for interaction; new contacts have been established which will turn out to be fruitful in the future. 34 participants have accepted the invitation, among them scientists from China, the US, Israel and several European countries. The seminar was sponsored by the TMR program. This allowed to invite several young scientists giving very interesting input to the seminar; the seminar yielded valuable new contacts for them.

4 Dynamically Reconfigurable Architectures

Seminar No. **98081** Report No. **201** Date **22.02.–27.02.1998**
Organizers: Karl-Heinz Brenner, Hossam ElGindy, Hartmut Schreck,
Heiko Schröder

The Dagstuhl Seminar on “Dynamically Reconfigurable Architectures” brought together 40 participants from 8 different countries and from very diverse areas:

- “algorithms/RMesh people” designing efficient algorithms for processor arrays with reconfigurable bus systems,

- “FPGA people” using field programmable gate arrays for fast and flexible prototyping and designing tools supporting hardware/software codesign,
- “optical people” designing fascinating new microoptical systems for fast communication at incredible bandwidth.

All the 29 talks treated some aspect of dynamically reconfigurable architectures, some were survey talks giving an introduction into the different areas. “Dynamical reconfiguration” refers to reconfigurable interconnection systems, being used for “computing in space”, and to reconfigurable logic cells or processing elements which can be programmed by look up tables (LUTs).

The talks emphasized that technological advances have opened up new ways of implementing complex systems in a way that blurred the barriers between hardware and software components development, and that existing design tools do not seem to be adequate for the necessary new design styles. For example, dynamically reconfigurable hardware allows to swap ”soft cells” on demand in and out of hardware, requiring new 2-dimensional scheduling (or paging) strategies. It is even possible to let hardware evolve by itself, learning the required functions, or simulating biochemical processes. New advances in optical communication all of a sudden lead to feasible implementations of interconnection structures which so far had been seen as having only theoretical value.

5 The Fourth Dagstuhl Seminar on Data Structures

Seminar No. **98091** Report No. **202** Date **01.03.–06.03.1998**
Organizers: Ian Munro, Stefan Näher, Peter Widmayer

The design and analysis of algorithms is one of the fundamental areas in computer science. This also involves the development of suitable methods for structuring the data to be manipulated by these algorithms. In this way, algorithms and data structures form a unit and the right choice of algorithms and data structures is a crucial step in the solution

of many problems. For this reason, the design, analysis and implementation of data structures form a classical field of computer science both in research and teaching.

The development of the research in this area has been influenced by new application fields such as CAD, Geographic Information Systems, Molecular Biology and Genetics. Not only new methods and paradigms, such as randomization or competitive analysis of algorithms, have been developed, but there is also some shift of interest away from theory, e.g., the classical analysis of asymptotic behavior of algorithms, to more practical issues, such as implementation problems and the usefulness of algorithms in practical applications. One can observe that more and more researchers in computer science also want to make their results available in form of programs or software packages. This trend is also reflected in important international conferences.

Results

The Fourth Dagstuhl Seminar on Data Structures was attended by 41 people from 11 countries. The workshop brought together researchers from all over the world working on different areas of the field of efficient data structures and algorithms. Quite a few interest groups used the workshop to discuss and develop ongoing and new cooperations. Many interesting results and solutions for theoretical and practical problems were presented.

The explicit consideration of external memory is becoming a major concern. It is crucial for the practical value of algorithms and data structures, and it creates interesting theoretical problems. Accordingly, there have been several contributions of external memory algorithms and data structures for problems in Computational Geometry and Geographic Information Systems.

On the theoretical side, there have been interesting contributions concerning search & graph structures. A particular result on hashing and several results on special-purpose data & search structures and added features (such as distributed, fault-tolerant, concurrent ...) extended recent insights and limits in the area. Another remarkable result showed a (nearly) exact bound for planar point location.

The geometric data structure section saw new algorithms concerning mobile robots and unknown terrains. An algorithm that extends by

far the limits of known problem sizes has been presented for rectilinear Steiner trees.

An open problem session reviewed Knuth's 60 famous open problems. It turned out that the majority of the problems is still unsettled today.

Perspectives

The workshop helped to identify several tracks of research in data structures that play and will continue to play an important role in the area. The field of algorithms and data structures continues its expansion towards previously undiscovered fields. As computers influence more and more aspects of our lives, data structures are needed for an ever growing domain of applications, with questions of steadily increasing complexity. The general task of developing and analyzing simple and practical algorithms and data structures remains with us.

6 Continuous Engineering for Industrial Scale Software Systems

Seminar No. **98092** Report No. **203** Date **01.03.–06.03.1998**
Organizers: Hausi Müller, Herbert Weber

Coping with legacy systems has been a research issue for a number of years. Re-engineering and reverse engineering have become the keywords to address the issues involved. Re(verse) engineering and system maintenance are often not clearly separated. Moreover, as one would expect the two communities dealing with these two subjects intersect considerably.

Re(verse) engineering research has in the past primarily been instance-oriented, that is, a certain system or application needed re(verse) engineering. This was certainly the right approach at the beginning to get a better understanding of the problem. With the experience of a number of those instance-oriented research projects, the time seems right to generalize on the experiences gained.

One insight gained in recent years clearly suggests to look at re(verse) engineering not as something that is done once or infrequently but rather as something that has to be done continuously: operating environments changes, application requirements changes, technologies changes, and

business process changes demand for a more continuous change process of entire systems.

A second major insight relates to the research environment needed to conduct research that is of immediate use. It seems good research can be done only together with industry that is now plagued by the problems and has hence the best handle to the problem. Research gains relevance only if it addresses to problems of industrial dimension and must hence be done with systems and infrastructures of that dimension. This once again emphasizes the need to cooperate with industry.

Now questions arise as to whether the results and experience gained in previous projects can be condensed, whether principles of a Continuous Software Engineering (CSE), i.e. an integrated paradigm of forward, reverse and re-engineering, can be deduced, and whether methods can be formalized and successfully applied.

The central focus for discussion during the seminar is as follows:

- How is adaptive business process management including business process re-engineering enabled through Continuous Software Engineering and what is the impact of the many variants of process management techniques (i. e. work flow, group work, teleworking etc.) and process management tools on Continuous Software Engineering?
- How can Continuous Software Engineering be enabled through proper architecture concepts (i.e. architectural patterns, invariant platforms, software-componentry) for software that is meant to function in local or global communication infrastructures (i. e. intranets, extranets, internet)?
- How can adaptive information management be enabled through Continuous Software Engineering through techniques for data migration, data restructuring, and data transformation, through model integration and metamodelling and through the wrapping of components for their continuous use?
- How can program migration from procedurally oriented legacy software towards components in distributed information & communication infrastructures be enabled through Continuous Software Engineering, and how do particular re-engineering and reverse engineering techniques apply?

As an important result of this seminar, we expect to form a nucleus of a community interested in continuous engineering research and technology transfer. Hopefully this nucleus will be able to establish fast information exchanges between its members. One positive result could be a consolidation on the many workshops and conferences that seem to be very dispersed and attract only a part of the interested people.

7 Program Comprehension and Software Re-engineering

Seminar No. **98101** Report No. **204** Date **08.03–13.03.1998**
Organizers: Hausi Müller, Thomas Reps, Gregor Snelting

Analyzing old software systems has become an important topic in software technology. There are billions of lines of legacy code which constitute substantial corporate assets. Legacy systems have been subject to countless modifications and enhancements and, hence, software entropy has typically increased steadily over the years. If these systems are not refurbished they might die of old age—and the knowledge embodied in these systems will be lost forever.

As a first step in “software geriatrics” one usually tries to understand the old system using program understanding or program comprehension techniques. In a second step, one reconstructs abstract concepts (e.g., the system architecture, business rules) from the source code, the documentation, and corporate knowledge; this is called software reverse engineering. Given an abstract representation of the system, one can then re-implement the system. This forward engineering step ranges from fully automatic approaches to manual reimplementations including restructuring techniques, formal transformations, injecting component technologies, replacing old user interface or database technology. The process of moving from an old legacy system to a new implementation is called software reengineering.

It was the aim of this seminar to bring together researchers who are active in the areas of program comprehension and software reengineering regardless of their particular approaches and research avenues. However, one of the areas of concentration for this seminar was slicing technology which is an important technique for software understanding and maintenance activities. Another topic of increased interest have been empirical

studies for software reengineering. Mathematical concept analysis gained some attention as a new framework for program understanding.

Several talks were accompanied by system demonstrations, giving participants first-hand experience of new analysis and reengineering technology. A panel session compiled a list of open problems, both technical and methodological. The traditional Dagstuhl walk offered an opportunity for topological comprehension and map reengineering.

8 Theoretical Foundations of Computer Vision: Evaluation and Validation of Computer Vision Algorithms and Methods

Seminar No. **98111** Report No. **205** Date **15.03.–20.03.1998**
 Organizers: Robert M. Haralick, Reinhard Klette, H. Siegfried Stiehl,
 Max A. Viergever

The third TFCV meeting in Dagstuhl addressed a subject which has been under intensive (and partly controversial) discussion in the computer vision community for several years. The evaluation and validation of computer vision algorithms and methods is of key importance, particularly for the configuration of reliable and robust computer vision systems and as well as for the dissemination of reconfigurable systems in novel application domains. Although in certain areas of computer vision a plethora of literature is already available on this subject, the research community still faces a lack of a well-grounded, mutually accepted, and (possibly) standardized methodology.

The range of fundamental problems encompasses, e.g., the value of synthetic images in experimental computer vision, the selection of a representative set of real images related to specific domains and tasks, the definition of ground truth given different tasks and applications, the definition of experimental test-beds, the analysis of algorithms with respect to general features such as computation time, convergence, stability, or domains of admissible input data, the definition and analysis of performance measures for classes of algorithms, the role of statistics-based performance measures, the generation of data sheets with performance measures of algorithms supporting the system engineer in his configuration problem, etc. The workshop thus attempted to bring together ex-

perienced colleagues from the international computer vision community both to discuss the state-of-the-art and to formulate recommendations for future activities.

36 talks, grouped in several topical sessions, were given by 32 speakers from 14 countries. Out of a total of 41 participants, 11 were from Germany; 5 from The Netherlands and the U.S.A. each; 3 from Denmark and United Kingdom each; 2 from Canada, Czech Republic, Japan, New Zealand, and Slovenia each; and 1 participant from Hungary, Israel, Slovakia, and Taiwan each. In addition to the oral presentations, four working groups - partly working during wood-walking tours - were established to discuss issues of common interest in more detail (see appendix for summaries). The participation of a number of younger scientists from Denmark, Germany, The Netherlands, and United Kingdom was rendered possible through financial support from the TMR (Training and Mobility of Researchers) programme of the European Union which is gratefully acknowledged. Moreover we are pleased to state that the presentations at our meeting were of such a high quality that a refereed proceedings book is planned to be published soon by Kluwer Academic Publishers in the Computational Imaging and Vision series.

Eventually the workshop has stimulated different future activities, ranging from the establishment of an algorithmic web site for the international computer vision community, to the recommendation of organizing a similar meeting at Schloß Dagstuhl in Y2K on the subject of theory, methodology, and practice of performance measures.

WG I: Performance Analysis, Web Sites, and Teaching Computer Vision

Chair: ROBERT M. HARALICK
University of Washington, Seattle

Our group initiated an activity to bring together in a coordinated set of web pages material that would be useful in helping researchers and students to learn about characterizing and evaluating the performance of vision algorithms as well as material in helping students to

learn about computer vision. The web page material will be both tutorial in content by way of reports, papers, and viewgraphs as well as by way of software implementing algorithms and software characterizing and evaluating the algorithms implemented. The main site will be <http://george.ee.washington.edu>.

The software will have some web page front end so that it can be demonstrated over the internet.

Each algorithm web page will have places for the following:

1. A self-contained viewgraph tutorial of the algorithm, suitable for teaching
2. Reports or papers describing the algorithm and properties of the algorithm such as time complexity, convergence, numerical stability etc.
3. References to the literature
4. Code for the algorithm including a web interface to enable running of the algorithm by a web browser
5. Documentation for the running of the algorithm
6. Data sets and their ground truth, possibly including some negative examples on which the algorithm does not perform satisfactorily.
7. Software for generating synthetic data sets with ground truth
8. Software for evaluating the results of the algorithm against the ground truth
9. A self-contained viewgraph tutorial of how the characterization of the algorithm performance or the evaluation of the algorithm is done
10. A demo involving a run on real and/or synthetic data

The web pages will be started without necessarily having proper material for all the above places. The community of interested researchers will help fill in any empty place.

WG II: Task-Based Evaluation of Computer Vision Algorithms

Chair: MAX A. VIERGEVER
University Hospital Utrecht

The discussions in the working group on task-based evaluation of computer vision algorithms reflected the dichotomy of the workshop as a whole: analysis of performance characteristics of computer vision algorithms in a well-defined context versus evaluation of methods to carry out an image analysis task in a complex environment.

Most of the research on performance characterization is carried out in the framework of a highly constrained application as, e.g., document analysis or photogrammetry. In such areas, the model underlying the computer vision algorithms generally reflects physical reality well. In complex real world problems as, e.g., in seismic imaging or medical imaging, hard model constraints can be imposed only to a limited extent. In medical imaging, for instance, the biological variability between patients as well as the occurrence of unknown pathologies contest the validity of case-independent model assumptions. This creates a situation in which the model error may be orders of magnitude larger than pure algorithmic inaccuracies.

The first round of the working group discussions was taken up by recognizing and analyzing the above differences. The very establishment of the dichotomy between algorithmic performance characterization and application dependent system evaluation has been the main result of this working group.

The second discussion round focussed on computer vision in medical imaging, the primary field of interest of most of the working group participants. In medical imaging, the crucial issue for acceptance of a computer vision algorithm is whether it improves the diagnosis or the treatment, or at least the diagnostic/therapeutic procedure (similar results, but faster and/or more objective). Evaluation of algorithms needs to be carried out in this perspective.

The major point of discussion was how to reconcile ground truth and realism. Computer-generated data are a perfect gold standard, but provide little realism. They may serve to establish a consistency check of computer vision algorithms, but cannot predict how well the application

will benefit. On the other hand, real world images (patient data) lack the objectivity needed for a gold standard. For example, image segmentation may be evaluated on patient data that have been segmented manually by experts, but the evaluation will then have to deal with issues as intra- and inter-operator dependency and inter-patient variations. This will require very large and thereby expensive studies, which will - if at all - be performed only on entire software procedures, certainly not on individual algorithmic components.

While the discussions did not result in practical suggestions for evaluation of algorithms in ill-defined environments, the working group considered the recognition and analysis of the complexity of this type of evaluation quite valuable. In consequence, the establishment of the working group has well served its purpose.

The working group consisted of the following participants: L.M.J. Flo-rack, S. Frantz, M.H. Loew, P. Meer, M. Nielsen, W. Niessen, I.T. Phillips, H.S. Stiehl, M.A. Viergever, and P. Zamperoni.

WG III: Specification and Propagation of Uncertainty

Chair: WOLFGANG FÖRSTNER
University of Bonn

The scope of the working group was to discuss the two questions:

1. How to specify the required performance of an algorithm?
2. How to perform propagation of uncertainty within an algorithm?

We proceeded in a top down way to attack the problem. We hypothesize each algorithm can be specified by its input (x, Q_x) , its output (y, Q_y) and the theory A of the transition between input and output. The input and output values x and y are uncertain, their quality is described by Q_x and Q_y . This allows to link different atomic algorithms to form an aggregated algorithm. Specification then is to pose constraints on the range of y and Q_y , which are task specific.

The discussion gave the following results:¹

¹(cf. also <http://www.ipb.uni-bonn.de/ipb/lit/abstracts98/foerstner98.dagstuhl-wg.html>)

1. Proof of the hypothesis: This was done by investigating a complicated enough task (3d-object recognition) which contains enough representative subtasks.
2. Discussion of the possible structures of input and output: The discussion showed that these may be simple or composed structures with subparts being: integers, reals, labels sets, lists, vectors, matrices, tensors, graphs, trees, relational structures etc. The *problem* therefore is *to establish a theory of quality measures for the different types of data*.
3. Discussion of the representation and propagation of uncertainty: It appears that statistics provides the appropriate tools, including covariance matrices, confusion matrices, Markoff-Chains, Markoff-Random-Fields, Bayesian networks, graphical models. However, *the task of modelling the uncertainty, of estimating the free parameters, of propagating the uncertainty and of testing the validity of the chosen models is by far not worked out for the basic tasks in image analysis*. The numerous examples from literature however show, that the tools are available and suggest the statistical framework to be applicable.

The working group consisted of the following participants: S. Bailey, C. de Boer, P. Faber, W. Förstner, R.M. Haralick, H. Haussecker, N. Ohta, and J. Sporring.

WG IV: Modular Design

Chair: ARNOLD W.M. SMEULDERS
University of Amsterdam

We consider the methods of performance evaluation as a useful technique to enhance the usability and the life time of techniques of computer vision, in our case by splitting them up in basic building blocks: the modules. We are of the opinion that proper evaluation is against what the module claims it does. So composed modules have to be tested against their intended purpose and with the data corresponding to that purpose. A description of its purpose useful for a general public (“this module finds edges”) may not nearly be precise enough for evaluation. A central

thesis to investigate is the following. When the inputs and outputs of two modules each are of the same type, the purpose of the performance evaluation is equal, the performance evaluation shows great similarity.

As we have restricted ourselves to low-level image processing, the first focus of attention is to evaluate known intensity patterns. We can restrict ourselves to synthetic images where the ground truth is known. This opens the way for automation of the performance evaluation process, and sampling the space of all images at will. The second focus of attention must be on images with a real world meaning. They have a semantic interpretation. Such will change the character of performance evaluation as here synthetic images with a computational ground truth are not available. It may be even so that the ground truth is only available as a best-consensus opinion. For one thing, this limits the number of images to consider as well as it changes the measure of quality.

The purpose of the working group is to establish patterns in performance analysis methods for basic building blocks. The study will be continued on after the workshop. The case of segmentation reads as an operation of “image” \rightarrow “map”. The ground truth in this case is also of the type “map”. All essentially different data configurations are listed as point (blobs, corners), line (edges, curves) or regions. Each of these possibilities requires a specific detector with a specific metric of performance. The metric has as input: $\text{metric}(\text{map}, \text{map})$. For a line segmentation algorithm a line–line metric is needed. In the metric, there should be included: under- and overdetection (has it been detected at all) and locality (is the result on the right place). The alternative is that there is a separate metric for detection as well as for locality. A secondary performance evaluation is the decay of the distance measure when the noise is increased until the point where the algorithm loses track. This is a measure of robustness against noise and other perturbations of the computational model. Other robustnesses, e.g against different noise models, may also be considered.

In the composition of modules, an intriguing question is performance evaluation for composed modules as a product of the performance evaluation of each of the atoms. If such a scheme exists it would be of great help as modules can be evaluated each in their own right and do not need to be evaluated again when put into a bigger system. Such orthogonality would serve performance evaluation considerably as evaluation can be done once. There would be no need to repeat the evaluation any time the module is embedded in a bigger system. However, there is doubt

whether this orthogonality can exist.

The working group consisted of the following participants: M. Petrou, R. Klette, J. Puzicha, and D. Richter.

9 Advanced Stochastic Modelling Applied to Telecommunication-Networks and Distributed Systems

Seminar No. **98131** Report No. **206** Date **29.03.–03.04.1998**
Organizers: Ulrich Herzog, Guy Latouche, Vaidyanat Ramaswami, Phuoc Tran-Gia

In the field of telecommunications and distributed systems we observe a vivid technological and methodological development; the integration of services and new fields of application require demands of up to now unknown quality and quantity.

Stochastic modeling methods being necessary for the conception, planning, realization and operation have always to be adapted to this development. In many cases the traditional models are no more sufficient; then totally new ways have to be found. Furthermore, shortened cycles of development require the highly efficient and extensively automatized support by tool environments.

The aim of the planned seminar is to unite European and foreign scientists working in industry and university, in the field of telecommunications and computer science. New ideas, concepts and results of modeling will be presented, discussed and evaluated.

The subjects we are dealing with can be classified with respect to technology, application, and modeling. Some important problems are for example the following:

1. The information and control flow as well as the management of slotted high-speed networks (e.g. ATM-asynchronous transfer mode) require discrete models with highly efficient evaluation techniques.
2. Modeling processes for mobile radio nets varying by location and time must theoretically be further developed, tested and evaluated.

3. Synchronizing and coordinating processes within the nets require an adequate modeling of causal dependencies.
4. Distributed applications require a simultaneous modeling of both the functional and temporal behavior and mutual influencing.
5. Beside the classical ways of solving stochastic modeling, integral concepts, especially Stochastic Petri Nets, Stochastic Automata and Stochastic Process Algebras are gaining more importance.
6. Beside the exact component modeling, modeling hierarchies for partial systems and for complex nets are playing a more and more important role. The correct description of the interfaces between modeling levels is thereby very important.

Efficient Stochastic Modeling of large telecommunications nets and distributed systems is only possible, if the modeling, the evaluation technique and presentation of results are harmonically coordinated and if they realistically comprise the field of application.

We will systematically invite international experts for all problem fields. Beside summarizing of all results in a Dagstuhl seminar report, we intend to publish the most interesting contributions in a special issue of a recommended journal.

10 Programs with Recursively Defined Data Structures Using Pointers

Seminar No. **98161** Report No. **207** Date **19.04.–24.04.1998**
Organizers: Mooly Sagiv, Michael Schwartzbach, Karsten Weihe

The theme of this seminar was to study programs that manipulate dynamically allocated data. The attendees were researchers from three different areas:

- Design and implementation of combinatorial (e.g., graph) algorithms with a heavy usage of dynamically allocated memory. Many of these algorithms are currently implemented in C++, e.g. [MN95].

- Verification of partial correctness of such implementations, e.g., proving that a program only refers to allocated memory cells. Proving that a program does not create memory leaks. By convention, we call these *cleanness* checks since they must hold for every reasonable program as opposed to correctness, which is program specific. The hope is that some of these tests can be carried out by future compilers.
- Compiler optimization to speed up the execution time of such programs. For example prefetching of linked data structures can improve performance by 45%, see [LM96].

Three panels of open problems were held (summarized below). Few of the open problems already initiated research, e.g., in the area of improving locality of programs that manipulate dynamically allocated memory and in the area of automatically eliminating “checking code” that validates that certain invariants are maintained after a library operation, e.g., that the insertion of a new edge into a directed graph, maintains the Euler equation.

33 talks were given presenting the state of art techniques in these areas. The talks were very interesting and accessible to all groups. Some of the talks presented solutions to open problems. One of the most interesting subject studied is the treatment of memory hierarchy in general and cache in particular. Finally, one talk by John L. Ross presented a solution to a problem posed by Luddy Harrison at the Dagstuhl seminar No. 9535, held in August 1995.

The schedule of the workshop was not fixed on the usual daily basis. Instead, we did an experiment on Monday afternoon: the schedule was fixed in an additional “problem session.” In this session speakers from different fields were encouraged to discover relations between their works (the preceding “five minutes madness” session helped a lot) and to build a session. Moreover, we included time for questions at the end of these and other sessions to review all talks in summary. In our opinion, some of the most interesting insights came about this way.

There are numbers of interesting conclusions that we concluded from the seminar:

- Despite of the difference in background and interest the different groups interact very well.

- At the organizational level, it may be better to present research in groups to save time and make discussions more interesting. Also, talks can be shorter and not everybody needs to give a talk.
- Memory hierarchy will continue to be a challenging issue both to algorithm designers and compiler implementors.
- Cleanness checking is a very important and difficult problem.
- Many of the pointer analysis algorithms and in particular the flow insensitive ones are mature enough to be implemented in industrial compilers.

Open Problems

We spent three evenings in understanding the open problems in this field. There are numbers of reasons why dynamically allocated memory will be more important in the research community and in the industry in the near future:

- Stack and statically allocated storage cannot be used for many of the combinatorial algorithms.
- It is well known that programs that manipulate dynamically allocated memory are hard to debug, prove correct, and optimize.
- Many of the pragmatic open compiler research problems related to stack and statically allocated memory were already solved. For example, Michael Hind presented in the seminar an empirical study which indicates that in many cases, flow insensitive analysis of stack allocated pointers provides information that is as precise as the flow sensitive approaches.
- There are many opportunities to speed up the execution of programs that manipulate dynamically allocated memory both on existing and on future architectures.
- Programming languages like Java create many challenges by opening many of the research ideas such as safe pointer dereferences and garbage collection to a wide audience.

Below we summarize the open problems that we have identified. Each of these problems is characterized as follows:

C A conceptual problem

P A pragmatic problem

A An algorithmic problem

1. (**C**) Can (explicit) pointers (sometimes) be avoided by introducing stronger language concepts?
 - (a) Eliminate pointer arithmetic and explicit addressing (*C* → *Java*)
 - (b) Eliminate Java style pointers (references)
2. (**C+P**) Can verification of cleanness or even correctness of programs with pointers be feasible?
 - (a) Automatically
 - (b) With user information + how much?
 - i. Interactively
 - ii. Annotations as comments (a-la-LCLint [Eva96a, Eva96b])
 - iii. Program checking, e.g., using ANSI C assert — can the compiler optimize a way or reorder these checks?
 - iv. Programming language styles, e.g., isolate pointer stuff in a separate function and use smart pointers in the rest of the code — can the compiler eliminate the cost via optimizations?
3. (**C**) How to do run-time prediction., e.g., of cache models?
4. (**C**) Can templates be used to verify libraries in a “generic way” without investigating the application programs?
5. (**C**) Can program analysis be used to verify that reasonable graph operations preserve Euler equations?
6. (**P**) How much can pointer analysis buy for classical machine (in)dependent optimizations? (Amer Diwan answered that for a particular optimization of redundant loads)

- (a) Which pointer analysis is best for a given application?
7. **(C)** Is there a way to improve the process of testing using information obtained by program analysis/verification?
8. **(A+P)** How can the memory efficiency of programs with pointers be improved?
9. **(C+A+P)** How can locality of references be expressed, proved or analyzed?
10. **(P)** Can flow sensitive algorithms for analyzing dynamic allocated data structure scale up for large programs?
11. **(P)** Is there a benchmark that is preferable for pointer verification and/or analysis (Spec [SPE92], Olden [RCRH95], LEDA, Todd Austin's pointer intensive benchmark)?
12. **(C)** User interface to allow tuning the garbage collection performance (w/o changing program semantics).
13. **(P)** Use cheap analysis (e.g., linear time point-to) in order to improve the running time of a more expensive, e.g., flow sensitive analysis?
14. **(C)** Can code idioms that identify special data structure manipulations, e.g., can insertions into a single linked list be identified via program analysis (in order to be replaced by a more efficient implementation)?
15. **(A)** Demand-driven pointer analysis.
16. **(C)** Pointer analysis in "open" programming environments (languages), e.g., Java dynamic loading.
17. Can domain specific information be used to improve efficiency of the memory subsystem performance (by a user or compiler)
18. Is there a formula that predicts the number of cache misses for a given program. Trishul Chilimbi partially answered this question.

References

- [Eva96a] D. Evans. LCLint User's Guide, 1996. Available at <http://larch-www.lcs.mit.edu:8001/larch/lclint/guide/> .
- [Eva96b] D. Evans. Static detection of dynamic memory errors. In SIGPLAN Conference on Programming Languages Design and Implementation, 1996. Available at <http://larch-www.lcs.mit.edu:8001/~evs/pldi96-abstract.html> .
- [LM96] Chi-Keung Luk and Todd C. Mowry. Compiler-based prefetching for recursive data structures. In Proceedings of the Seventh International Conference on Architectural Support for Programming Languages and Operating Systems, pages 222–233, October 1996.
- [MN95] K. Mehlhorn and S. Näher. LEDA: A platform for combinatorial and geometric computing. *Communications of the ACM*, 38(1):96–102, 1995.
- [RCRH95] Anne Rogers, Martin C. Carlisle, John Reppy, and Laurie Hendren. Supporting dynamic data structures on distributed memory machines. *ACM Transactions on Programming Languages and Systems*, 17(2):233–263, March 1995. Article and benchmark suit available at <http://www.cs.princeton.edu/~mcc/olden.html> .
- [SPE92] SPEC Component CPU Integer Release 2/1992 (Cint92). Standard Performance Evaluation Corporation (SPEC), Fairfax, VA, 1992.

11 Generic Programming

Seminar No. **98171** Report No. **208** Date **26.04.–01.05.1998**
Organizers: Mehdi Jazayeri, Rüdiger Loos, David Musser, Alexander Stepanov

Generic programming is a sub-discipline of computer science that deals with finding abstract representations of efficient algorithms, data

structures, and other software concepts, and with their systematic organization. The goal of generic programming is to express algorithms and data structures in a broadly adaptable, interoperable form that allows their direct use in software construction. Key ideas include:

- Expressing algorithms with minimal assumptions about data abstractions, and vice versa, thus making them as interoperable as possible.
- Lifting of a concrete algorithm to as general a level as possible without losing efficiency; i.e., the most abstract form such that when specialized back to the concrete case the result is just as efficient as the original algorithm.
- When the result of lifting is not general enough to cover all uses of an algorithm, additionally providing a more general form, but ensuring that the most efficient specialized form is automatically chosen when applicable.
- Providing more than one generic algorithm for the same purpose and at the same level of abstraction, when none dominates the others in efficiency for all inputs. This introduces the necessity to provide sufficiently precise characterizations of the domain for which each algorithm is the most efficient.

The intention in this seminar is to focus on generic programming techniques that can be used in practice, rather than to discuss purely theoretical issues. By the end of the seminar we would like to come up with the following results:

1. A list of problems in generic programming. These include new components, new kinds of abstractions, language extensions, tools. (This list is maintained at www.cs.rpi.edu/~musser/gp/pop .)
2. A process for extending the existing body of generic components, as well as methods for their specification and verification and for establishing their efficiency in actual programs.

We think that to accomplish these goals we need to share a common vocabulary. Therefore, we will use the vocabulary established by the C++ Standard Template Library (STL) of fundamental data structures and

algorithms. This is not intended to preclude discussion of generic programming issues that occur in other areas and that might be more easily illustrated with other libraries and languages. For example, topics might include language extensions to support generic programming in more recent languages such as Haskell or Java, or how generic programming goals intersect with design patterns or frameworks research.

12 Domain Theory and Its Applications

Seminar No. **98181** Report No. **209** Date **04.05.–08.05.1998**
Organizers: Stephen Brookes, Manfred Droste, Michael Mislove

This Dagstuhl seminar focused on Domain Theory and Its Applications. Domains were first discovered by DANA SCOTT in the mid-1960s, and quickly became of central importance in the area of programming language semantics, which remains the primary area of their application. Recently, more diverse applications of domain theory to other areas have been discovered, and this has led to a flourishing of activity in the area. The seminar focused on this diverse range of applications, as well as on the role of domain theory in its more established areas of application.

The Dagstuhl Seminar also was coordinated with the Spring 1998 meeting of the series, the Mathematical Foundations of Programming Semantics. This series meets annually, mainly in the U.S., and is among the most popular meetings for researchers working in programming semantics and its mathematical underpinnings. Coordinating the Dagstuhl Seminar on Domain Theory with the MFPS meeting allowed us to attract some of the most active researchers in domain theory to both meetings. More importantly, the Dagstuhl meeting was used to assess the state of domain theory, and important insights were gained through the seminar into the status and future possibilities for domain theory and its applications.

Among the various areas within domain theory, we mention a few that were highlighted during the Dagstuhl Seminar:

- *Structure theory of domains*: This “basic” area of domains has had recent impetus through the work of ACHIM JUNG (*Birmingham*) in which the approach of *abstract bases*, originally proposed by M. B. Smyth (*Imperial*), was used to devise a uniform approach

to the theory of continuous domains in analogy to the traditional approach to algebraic domains.

- *Domain theory and topology:* Topological aspects have played a fundamental role in domain theory since its founding. In recent years, efforts have been made by M. B. Smyth (*Imperial*), Jan Rutten (*CWI*), Bob Flagg (*Southern Maine*) and others to generalize aspects of domain theory to more general settings. In particular, the basic question of the relationship of domain theory and (quasi-)metric spaces has recently received considerable attention.
- *Domain theory and category theory:* Categorical considerations are of primary importance in domain theory. Domains provided the first (and, as far as we know, the only) categories in which one can solve the equation $X \simeq [X \rightarrow X]$ non-trivially. More recently, the work of KIM WAGNER (*Cambridge*) has demonstrated that some of these categorical results and related ones in categories of metric spaces have an intimate connection.
- *Synthetic Domain Theory:* The work of MARTIN HYLAND (*Cambridge*) and of GIUSEPPE ROSOLINI (*Genova*) and others on synthetic domain theory – the attempt to establish a theory of sets in which all sets are domains – has had a major influence on the development of the theory and on its relationship to other areas, most notably category theory.
- *Axiomatic Domain Theory:* One of the tantalizing aspects of domain theory is the fact that the only solutions of the equation $X \simeq [X \rightarrow X]$ that are known in any cartesian closed category all reside within categories of domains. The work of PETER FREYD (*Pennsylvania*) and of MARCELO FIORE and GORDON PLOTKIN (*Edinburgh*) has led to a better understanding of this issue.
- *Domain theory and logic:* Domain theory has served as a fertile ground for models for many of the logics that are fundamental to theoretical computer science. The seminal work of SAMSON ABRAMSKY (*Edinburgh*) has shown how domains are closely linked to program logics. Recent attempts to extend Abramsky’s approach has appeared in the work of MARTA KWIATKOWSKA (*Birmingham*) on the modal mu calculus, and MARCELLO BONSAUGUE

(*Vrije Universiteit Amsterdam*) has provided an extension with the notion of observation frames.

- *Domain theory and concurrency*: Domain theory has been the traditional area in which to devise models for concurrency and process algebra. Recent work by PAUL GASTIN (*LITP, Paris*) and VOLKER DIEKERT (*Stuttgart*) has used domains to provide models for true concurrency, in which parallel composition is taken as primitive, rather than being resolved in terms of sequential composition and nondeterministic choice. These areas were amply represented at the seminar.
- *The Comprox Group*: One of the pioneers of applications of domain theory to other areas in the last few years has been ABBAS EDALAT (*Imperial College*). Along with such colleagues as REINHOLD HECKMANN (*Saarbrücken*), he has shown how domains and their topological analogues provide unexpected and remarkable insights into phenomena ranging from fractals to neural networks and learning automata to such established areas as real analysis and the Riemann integral.

An additional important accomplishment of the seminar was the drafting of a *List of Open Problems* on domain theory. This list, which is currently being edited and refined, will be the template for a number of such lists in several areas of theoretical computer science.

13 Quantum Algorithms

Seminar No. **98191** Report No. **210** Date **10.05–15.05.1998**
Organizers: Thomas Beth, Gilles Brassard

Quantum algorithms — a new topic in both informatics and physics has become a central theme of one of the most challenging areas of interdisciplinary research of modern science. This seemingly esoteric area of research which has been stimulated by Feynman in 1980 and Benioff, who was the first to suggest quantum-mechanical evolution for computation in 1982, has become a serious challenge after Shor’s publication of the “Algorithms for quantum computation: Discrete logarithms and factoring”

in 1994. This breakthrough in theoretical computer science has stimulated a new field of physics, especially on the experimental side, namely the investigation of controlled quantum systems which is motivated by the promise that algorithmic processing of quantum information may provide an exponential gain in speed and space over classical computers. As a consequence of this joint research there has been a surprising progress during the last years towards new algorithms and especially complexity bounds for certain problems. However, up to today there are no exact theorems available on the relation between quantum complexity classes and classical complexity classes.

Quantum algorithms rely on three effects:

- superposition,
- entanglement,
- interference.

If several qubits are combined in a quantum register by the laws of quantum mechanics, the state space of such a processor allows the handling of exponentially many data by superposition of entangled states. Owing to the linearity of quantum mechanics each operation of so-called quantum gates will act on all states simultaneously which have non-zero population in the superposition. This phenomenon is the basis for quantum parallelism which leads to a completely new model of computation: While a classical probabilistic algorithm can be well described through the tree of all possible computations weighed with the respective probabilities, the sum of probabilities of all positive results are added to give the total probability of a successful computation, the use of quantum states based on complex amplitudes instead of probabilistic weights will allow the enhancement or deletion of amplitudes. This, in a nutshell, is the essential advantage of quantum algorithms. Each desirable computational path can be designed to “absorb” the probability amplitude on the account of the amplitudes of other paths. This principle is known from physics as constructive and destructive interference.

Physical realizations of quantum computers are envisaged as hybrid systems consisting of a classical computer and a quantum register controlled by classical electromagnetic fields. The control of this system at runtime especially and the design of program loops are performed by classical computers on the basis of measurements carried out on the quantum

register. For obvious physical reasons such programming languages are rather restricted even though it has been proved by Deutsch in 1985 that the class of quantum Turing machines encompasses the class of classical Turing machines, following a proposal by Benioff who was the first to think that computation can be done entirely in a quantum mechanical unitary manner.

The Dagstuhl seminar 98191 has brought together scientists from computer science, mathematics, theoretical physics, and experimental physics to discuss the most recent developments of this very new and possibly revolutionizing concept of modern computing.

As opposed to most “classical” computer science conferences, the unusual concept of this conference can be described by the fact that the theoretical aspects of algorithm design are, at least at the present state of the field, not to be seen device-independent: In other words, in contrast to classical computer science approaches to algorithms and their application, where the actual physical realization on the bit level is not taken into account, it is the feature of this field that in quantum algorithms the physical realization is intrinsically connected with the design of algorithms: Thus, everyone working in the area has to be relatively well acquainted with both, the computer science sides and the physics sides where the modeling of both physics and computer science on the basis of quantum theory and their applications relies on rather strong mathematical foundations such as Hilbert space theory, group theory, combinatorics, information theory, coding theory, and signal processing.

The organizers have judiciously combined the topics to be addressed at this conference to bring together those experts in the fields which can contribute to each of the questions from the mentioned side, ranging from pure theory to actual experiment. Owing to the relative youth of the field and the demanding requirements for a successful work in this area, this Dagstuhl seminar did not only bring together a considerable set of the world experts in the area but also a relatively dominating majority of young scientists which have been attracted by this area. The success of this workshop was not only noted by the computer scientists who have been able to learn from fundamental physical developments of the last years, but also especially physicists have been attracted by the methods of algorithm design and theoretical computer science to be applied to design new physical processes.

14 Algorithms and Complexity for Continuous Problems

Seminar No. **98201** Report No. **211** Date **17.05.–22.05.1998**
Organizers: Peter Mathé, Klaus Ritter, Joseph F. Traub

The main topic of the Seminar was the complexity of high-dimensional continuous problems. In 17 out of 37 talks the dimension played a crucial role. Among the problems addressed during the Seminar were partial differential equations, integral equations, linear optimization, and verification for Boolean functions. Most of the talks on high-dimensional problems were related to the computation of d -dimensional integrals $\int_{[0,1]^d} f(x) dx$. Therefore from now on we focus on the integration problem.

For a given class F of integrands f , the ε -complexity is the minimal computational cost that is needed to compute integrals with an error at most ε for every $f \in F$. Since floating-point arithmetic is commonly used in practice, its abstraction, the real number model of computation, is used in the study of the ε -complexity of integration. In addition to sharp bounds on the complexity one is also interested in algorithms that yield error ε with cost close to the ε -complexity.

High dimensional integrals arise, for instance, in statistical mechanics and mathematical finance. In the latter case, we might wish to compute the value of a financial derivative, e.g., a collateralized mortgage obligation (CMO). A typical CMO is a pool of 30-year mortgages, whose value depends on the proportion of borrowers who refinance their loans and on future interest rates. A stochastic model (geometric Brownian motion) is used for the interest rates, and therefore the value of the CMO is an expectation. If monthly refinancing is permitted, then we end up with a 360-dimensional integral, which cannot be determined analytically. Moreover, the evaluation of the integrand at a single point may require some 10^5 floating point operations. The fast valuation of financial derivatives is of considerable interest to the financial community and is a fascinating and challenging problem for the scientific computing community. Several of the participants (e.g. from Columbia University, NY, and from IBM Japan) have developed new software for the evaluation of financial derivatives.

In the past the main computational tool for integrals in very high

dimensions was the Monte Carlo method, whose basic version yields an error bound $\|f\|_2 \cdot n^{-1/2}$ at cost proportional to n . Observe that there is no explicit dependence on the dimension d . On the other hand, we have only got a stochastic assurance on the error and $n^{-1/2}$ tends to zero rather slowly as the number of sample points tends to infinity.

Can we do better in high dimensions? The Seminar provided answers and implications for the computational practice. We survey some of the new results and trends, and we point out important open problems.

Although numerical integration is a classical computational problem, the classical results do not really help to answer the above question since they deal with fixed dimension d and large n . Such results are applicable only for small values of ε . For instance, a famous theorem says that for integrands of bounded variation an error $c_d \cdot n^{-1} \cdot (\ln n)^{d-1}$ can be achieved at cost n . Corresponding sample points are given as low discrepancy points, and several number theoretical constructions for such points are available. The resulting algorithms are called quasi Monte Carlo methods. The factor c_d is usually unknown. Observe, however, that the function $n^{-1} \cdot (\ln n)^{d-1}$ is increasing for $n \leq \exp(d-1)$. Therefore such asymptotic results are useless for every practical value of n if d is large, say $d \geq 50$.

A new point of view towards the complexity of high dimensional problems is needed. Suppose that a sequence of function classes F_1, F_2, \dots is given, where F_d consists of functions of d variables. Then it makes sense to study the complexity as a function of ε and d . One may ask, for instance, how the complexity grows for fixed accuracy ε as the dimension d gets large. In most applications the limits of practical computability are in fact due to the huge dimension, while only a moderate accuracy is needed. This new approach is studied with a great success by several of the participants.

A ‘negative’ new result holds for many classical spaces F_d . There exists a constant $c > 1$ such that every cubature formula with positive weights and error less than ε needs at least $(1 - \varepsilon^2) \cdot c^d$ sample points. It is conjectured that this is actually a complexity bound, i.e., a similar lower bound holds for the cost of every method with error at most ε . Assuming that this conjecture is true, integration is intractable for classical smoothness spaces.

A related open problem for functions with bounded variation reads as follows. Is there a constant $c < 1$ such that for every dimension d

there exist 2^d sample points with star-discrepancy less than c ?

To break intractability of integration for classical spaces, weighted function classes F_d have been introduced. For quasi Monte Carlo methods we know necessary and sufficient conditions on weights to get tractability of integration. In some situations only nonconstructive proofs of tractability are available, and the search for good algorithms is still open. In a Hilbert space setting newly developed weighted tensor product algorithms yield tractability with the proper decay of the weights.

In the last few years, quasi Monte Carlo methods have been applied with a great success in derivative pricing, value at risk computations, and financial models in insurance industries, causing a significant impact also on the computer software business. It is believed that the results for weighted function spaces can explain the success of quasi Monte Carlo methods. So far, however, the structural properties of integrands in derivative pricing are not completely understood, and membership to certain weighted function classes is an open problem. We also believe that new theoretical insight will lead to algorithmic improvements.

A general open problem in this area is the characterization of classes F_d for which quasi Monte Carlo methods significantly beat the classical Monte Carlo method.

In some applications, e.g., in mathematical finance and in several branches of physics, even integrals over infinite-dimensional function spaces occur. Moreover, functional or path integration is closely related to the theory of partial differential equations. An important example is the Feynman-Kac integral for the heat equation. New approaches and complexity results for Wiener integrals and Feynman-Kac integrals were presented at the Seminar.

Further major topics of the Seminar included Monte Carlo and quasi Monte Carlo methods for operator equations with applications to image processing, as well as adaptive algorithms for partial and stochastic differential equations.

The Seminar included a meeting of IFIP Working Group 1.1 (Continuous Algorithms and Complexity). Selected papers from the Seminar will be published in a special issue of the Journal of Complexity.

15 Hierarchical Methods in Computer Graphics

Seminar No. **98211** Report No. **212** Date **24.05.–29.05.1998**
Organizers: Markus Gross, Heinrich Müller, Peter Schröder, Hans-Peter Seidel

Over the last decade hierarchical methods, multiresolution representations and wavelets have become an exceedingly powerful and flexible tool for computations and data reduction within computer graphics. Their power lies in the fact that they only require a small number of coefficients to represent general functions and large data sets accurately. This allows compression and efficient computations. They offer both theoretical characterization of smoothness and coherence, insights into the structure of functions, and operators, and practical numerical tools which often lead to asymptotically faster computational algorithms. Examples of their use in computer graphics include

- curve, surface, and volume modeling,
- efficient triangle meshes, mesh simplification, subdivision surfaces,
- multiresolution surface viewing and automatic level of detail control,
- image and video editing, compression and querying,
- efficient solution of operators such as global illumination and PDEs as they occur in finite element modeling for animation and surgery simulation,
- flow and volume visualization.

There is strong evidence that hierarchical methods, multiresolution representations, and wavelets will become a core technique in computer graphics in the future.

This Dagstuhl Seminar has provided a forum for some of the leading researchers in this area to present their ideas and to bring together applications and basic research in order to exchange the requirements of systems, interfaces, and efficient algorithmic solutions to be developed. The seminar has been attended by 52 participants from 11 countries.

16 Programs: Improvements, Complexity, and Meanings

Seminar No. **98231** Report No. **213** Date **07.06.–12.05.1998**
Organizers: Neil D. Jones, Oege de Moor, James S. Royer

The general topic of this seminar was the interface between Programming Languages and Complexity Theory. A model question for this area is:

Are functional or logic programs necessarily less efficient than equivalent imperative programs?

This has been the subject of endless hallway debates which have generated “much heat, but little light.” Such questions need to be made more precise in order to be answered and solution can be tricky indeed. A model answer to the above question was given by Nick Pippenger (POPL 1996):

“Pure LISP” is *provably* less efficient, by a logarithmic time factor, than “impure LISP” with list-modifying assignments, when applied to solve a problem involving a series of online permutations.

This result together with a result relating the efficiency of “pointers versus addresses” (Ben-Amram and Galil) and a proof that “constant time factors do matter” (Jones) have inspired considerable interest in this topic. However, the area is still in a nascent stage.

The intent of this workshop was to bring together:

Semanticists interested in operational models of programming languages, expressiveness, intensional properties, etc.

Program transformers and analysts interested in program execution speed, limits to implementation efficiency, etc.

Complexity theorists interested in precisely stated problems about computation time and space, language expressiveness, etc.

The goal was to promote fruitful interactions, leading to formulation and perhaps addressing new questions of both theoretical and practical interest.

There was a great breadth to the topics discussed and areas represented. A considerable amount of bridge building took place, but there were clearly some chasms no one yet knows how to cross (e.g., how to deal with all the issues involved in rigorous complexity analyses of non-trivial lazy programs). The discussions were lively, wide ranging, and stimulating.

The Interface Between Complexity and Programming Languages (by Neil D. Jones)

Comparisons are made between research in complexity theory and research in programming languages, with an eye to gaining the good features of both on subjects of common interest.

Complexity is an extensional theory, focusing on *what* is to be computed rather than how. High points include the existence of complexity classes PTIME, etc., that are *robust* with respect to machine model and choice of problem representation; and notions of problem equivalence via reductions, leading to *complete problems*. Low points: a tendency to ignore constants and to concentrate only on asymptotic factors; messy and sometimes cryptic algorithm descriptions.

Language research is intensional, focusing on algorithms more than problems. High points include elegance, generality, giving a basis for correctness proofs and program synthesis and transformation. Low points: imprecise cost measures, lack of robustness, few common assumptions.

A discussion was made of the lack of open problems in programming languages—the tendency to set up new frameworks rather than work on already-existing problems.

The talk ended with an overview of the contributions by the speaker to the interface. Results include complexity bounds on program analysis, characterizations of LOGSPACE and PTIME by simple programming languages, a theorem that for a natural language I and a constant b , $\text{TIME}(abn) \not\equiv \text{TIME}(an)$ for all $a \geq 1$, and an analysis of Levin's Theorem, that r.e. search problems *all* have near-optimal search programs. Both of the last two results depend critically on the existence of “effi-

cient” self-interpreters, whose overhead is *independent* of the program being interpreted.

17 Real Computation and Complexity

Seminar No. **98241** Report No. **214** Date **15.06.–19.06.1998**
Organizers: Felipe Cucker, Thomas Lickteig, Marie-Françoise Roy,
Michael Shub

The field of algorithmic complexity of real computational problems has received much attention in recent years. This topic with geometrical, algebraic, analytic, and numerical aspects encompasses the foundational area of scientific computing and has a wide range of relevant applications. The main object is the computation with polynomials, the better understanding of what makes these computations difficult or easy in order to design faster computer algebra algorithms. Due to their omnipresence, the case of real polynomials evidently plays a predominant role. Many new algorithms have been designed during the last decade for deciding the existence of solutions of equations and inequalities, or for computing solutions.

The seminar was intended to be broad and open. While clearly focussed on real computational problems, scientists with rather different backgrounds such as computer science, numerical analysis, algebraic geometry, logic, and abstract real algebraic geometry could be brought together to know and discuss main problems from different perspectives. This is important as scientific questions often don't find their answers within their domain of origin. One aim of the seminar was to support the collaboration of people of different origin.

We had 33 participants coming from Belgium, France, Germany, Great Britain, Italy, Poland, Russia, Spain, and the United States. During the meeting 30 lectures have been given that have been continued by informal evening discussions in smaller groups. The talks focussed on new concrete algorithmic results, others introduced to main ongoing work or to the way of looking at things in important neighboring fields. Main topics addressed include

- quantitative bounds in real algebraic geometry related to lower and upper complexity bounds,

- computation schemes (straight line programs) as basic data structure to cope with multivariate polynomial computation,
- translation of theoretical knowledge into fast implementations,
- univariate and multivariate root finding and root isolation,
- Blum-Shub-Smale discussion of differential equations, other ground fields, and counting problems,
- exact real computation, and
- approximation methods in the bit model.

Both senior researchers as well as young researchers contributed to the seminar. A JOURNAL OF COMPLEXITY (<http://www.apnet.com/www/journal/cm.htm>) special issue is dedicated to the workshop with selected papers addressing topics covered by this seminar.

The field has two major challenges. The first one, on the theoretical level, is the mathematical understanding of obstructions to the existence of fast polynomial computations, that is to say, lower bounds via an intrinsic *inherent geometric complexity* of a computational task. By knowing what must be avoided knowledge of lower bounds can furnish a guide-line in the design of algorithms. The second one, the design, is just now going in the direction of putting into effective use the theoretical knowledge of algebraic and arithmetic complexity, that is to say, translation into *really real* real algorithms that are fast in practice with new basic organizational designs.

18 Modeling and Simulation of Gene and Cell Regulation and Metabolic Pathways

Seminar No. **98251** Report No. **215** Date **21.06.–26.06.1998**
Organizers: Julio Collado-Vides, Ralf Hofestädt, Michael Mavrovouniotis, Gerhard Michal

The second Dagstuhl seminar for *Modeling and Simulation of Gene Regulation and Metabolic Pathways* was held from June, 21 to 26, 1998. It was a multidisciplinary seminar with 59 participants from 15 countries.

Schloss Dagstuhl workshops in general emphasize computer science, and we are delighted to focus on the rapidly developing links between biosciences and computer sciences. The 1998 meeting is a sequel to the 1995 Dagstuhl seminar on the same topic. Both were generously supported by grants from the Volkswagen Stiftung and the European Community (TMR Grant).

The availability of a rapidly increasing volume of molecular data enhances our capability to study cell behavior. In order to exploit molecular data, one must investigate the link between genes and proteins; the link between protein structure and protein function; and the concerted effects of many proteins acting on, and interacting with, the mixture of small and large molecules within a cell. This last step is the study of gene regulation and metabolic pathways which was the topic of the Dagstuhl seminar.

The molecular data must be stored and analyzed. Database systems for genes and proteins (EMBL, GENBANK, PIR, SWISS-PROT) offer access via internet. In the research field of molecular biology this technique allows the analysis of metabolic processes. To understand the molecular logic of cells we must be able to analyze metabolic processes in qualitative and quantitative terms. Therefore, modeling and simulation are important methods. They influence the domain of medicine and (human) genetics - the microscopic level. Today integrative molecular information systems which represent different molecular knowledge (data) are available. The state of the art is shown by P. Karp's system EcoCyc, which represents the metabolic pathways of *E. coli*. For every gene or protein within a specific metabolic pathway, EcoCyc presents the access to all corresponding genes and/or proteins. Moreover, the electronic information system KEGG represents all biochemical networks and allows the access to the protein and gene database systems via metabolic pathways. However, both systems are based on the idea of the statical representation of the molecular data and knowledge. The next important step is to implement and integrate powerful interactive simulation environments which allow the access to different molecular database systems and the simulation of complex biochemical reactions. Molecular information systems for gene regulation and metabolic pathways were one topic of the Dagstuhl seminar. The idea was to discuss the progress of this research field and the integration of the molecular database systems in combination with simulation tools. The organisers of the seminar invited colleagues, who presented their ideas through 42

talks and computer demos. More than 30 years ago Gerhard Michal started to collect all biochemical reactions. His classification is presented by the Boehringer pathway chart. This data collection was extended by the KEGG research group, which implemented the first electronical representation of this data in 1996. Nowadays all biochemical reactions are available via internet using the KEGG system. KEGG represents links to molecular database systems for genes, proteins, and enzymes, which are elements of metabolic pathways. Thus a link to the EMBL database system represents more information about a specific gene, and a link to the SWISS-PROT system represents more information about the protein (enzyme). Regarding the KEGG system the representation of quantitative data and kinetic data is not available today. Furthermore, additional to the molecular data (genes, proteins, and pathways) the first molecular information systems are available which represent data of the cell signals. Besides the Japanese Cell Transduction Database the GENENET database system is available. Taking regard to both molecular information systems this can be interpreted as the first scientific step in which cell reaction processes are surveyed from the gene regulation process to the cell communication.

For molecular biology the phenomenon of gene regulation is the main question. The systematic discussion of this question is based on the electronical representation of the molecular knowledge, which allows the complex analysis of this data. For that reason specific database systems are implemented (OperonDB, TRANSFAC and TRRD). These database systems represent all known operons and the transcriptional factors for *E. coli* (OperonDB) and eukaryotic cells. Today, two research fields based on this data are supported: The prediction of promoter sequences and the modeling of gene regulation. The prediction of promoter sequences is of importance, because the promoter is the starting signal for a structure gene which represents the genetic information. The human genome project will sequence the whole genome until the year 2004 ($64 * 10^9$ base pairs). The next step is to calculate the corresponding genetic map. Therefore, sequence pattern matching algorithms must be developed and implemented. In addition modeling and simulation of gene regulation processes will support the systematic analysis of the metabolic pathways.

John Reinitz opened the seminar. He presented ideas about modeling of genetic factors and analyzed the process of segment determination in *Drosophila* through numerically inverting a chemical kinetic equation which describes the regulatory circuitry and accounts for the synthesis

rate, diffusion, and decay of gene products. The molecular mechanisms of gene regulation were presented by **Edgar Wingender**. During the last decade he has been analyzing the molecular mechanisms of eukaryotic gene regulation and has been collecting all transcriptional factors which can be found using his database system TRANSFAC. The prediction of promoter sequences based on this data was one important topic of the gene regulation session. **Julio Collado-Vides**, **Gary Stormo**, and **Thomas Werner** showed algorithms for the detection of promoter sequences for *E. coli* and eukaryotic cells. The molecular mechanisms of the cell death were discussed by **Dominique Bergeron**, and **Luiz Mendoza** talked about complex metabolic networks.

The modeling of regulatory networks belongs to the topic of Biophysics and Biomathematics. Moreover, discrete models are developed using methods of Bioinformatics. At the beginning of that session **Jay Mittenthal** presented the metabolic pathway of the Pentose Phosphate Cycle. **Gerhard Michal** is the creator of the Boehringer pathway chart which inspired many of us to pursue databases and integrative methods for the study of the metabolism. In his talk he discussed a brief overview of the issues surrounding the development of graphical representations and displays of metabolic pathways and other biological information. In the case of analytic models **Michael Savageau** introduced a model which allows the simulation of complex kinetic effects. Using graph theoretical methods **Michael Kohn** discussed his model for the simulation of metabolic networks. **Stefan Schuster** outlined several powerful methods for determining key features of a metabolic pathway or network. He showed how conservation relations may be identified and how elementary biochemical routes (and hence the spectrum of behaviors of the biochemical network) may be determined. Further he outlined the principles of metabolic control analysis and its extensions. A new grammatical model for the analysis of complex metabolic processes was presented by **Simone Bontolila**.

Another topic of the seminar were molecular database systems. At the beginning of this session **Thomas Mück** discussed new topics in the research field of database systems and **Vladimir Babenko** introduced new techniques for the integration of molecular database systems. **Minor Kanehisa** showed the pathway database system KEGG and discussed further applications. **Fedor Kolpakov** demonstrated the database system GENENET, which is similar organized to the Japanese database system for Cellular Signal Transduction, which was presented

by **Takako Takai-Igarashi**. **Rolf Apweiler** talked about the SWISS-PROT database, and **Daniel Kahn** demonstrated a new database system for the integration of protein knowledge. One important application of this molecular data is the diagnosis of metabolic diseases. In the case of inborn errors **Manuela Prüss** introduced the database system MDDDB.

The final topic of the seminar was the integration and simulation of metabolic networks. The first generation of powerful simulation environments for the metabolic network control was discussed. These tools work using the biochemical data and diverse models which were presented in the sessions mentioned before. **Pedro Mendes** demonstrated his simulation environment GEPASI, which allows the analytical modeling of the metabolic processes. A first information system based on the integration of molecular databases and a grammatical simulation environment was introduced by **Uwe Scholz** and **Ralf Hofestädt**. Finally, an expert system for the modeling of metabolic processes was presented by **Jaime Lagunez**.

Concluding remarks

It is not sufficient to know what each protein or gene does in the cell (it usually catalyzes or regulates a biochemical reaction), but one must also decipher what they are all doing together (they form pathways of elaborate transformations and regulatory networks). In order to decipher the metabolic pathways that define the behavior of the cell as a whole, one must use information on single-protein activity. But there is also information flow in the reverse direction: The position and role of an enzyme in the metabolic network provides crucial insights and hypotheses for its genetic regulation and its relationship to other proteins. Genes and proteins are routinely sequenced and stored in database systems. Data on biochemical pathways has been systematically collected for the last three decades (in pictorial and text form), and the accumulation of such data has increased dramatically in recent years (and shifted to computational representations). The systematic use of collected data is also continually making advances. Methods for computational modeling and simulation are made feasible by the availability of data and are driven by the need to understand the behavior of complex biological systems. The integration of information, especially combinations of genes, enzymes, and metabolic pathways will be necessary in the study of biological regulatory structures, which usually involve multiple facets, components, and scales of action. Database systems and powerful models are already available, and the first practical simulation tools are implemented based on

powerful theoretical methods. These information-integrative activities will increasingly shed light on the biochemical mechanism of life.

The actual questions of the seminar were focused by the final discussion which concluded that: The number of molecular database systems is increasing. Moreover, these systems are available via internet. The now available accessing techniques are www links to the relevant molecular database systems, which support the navigation through the molecular data. However, this data must be available for further analysis processes. The detection of promoter structures is one actual example, which shows also the algorithmic problems of this research field. Besides the algorithmic analysis, modeling and simulation based on this molecular data are of importance. Different tools are developed and implemented. However, the selection of the model depends on the actual question. The main task for the next years is the integration of the database systems and the simulation environments, which will allow the simulation of complex metabolic networks.

Further information about the Dagstuhl seminar:

http://www.witi.cs.uni-magdeburg.de/iti_bm/dagstuhl/

19 The Semantic Challenge of Object-Oriented Programming

Seminar No. **98261** Report No. **216** Date **29.06.–03.07.1998**

Organizers: Luca Cardelli, Achim Jung, Peter O’Hearn, Jens Palsberg

Object-oriented programming is based on an informal concept of object as an entity or thing whose identity persists over time. The object concept is immediately meaningful to programmers, and has proven to be a useful and flexible organisational device in the analysis, design, and maintenance of complex systems. But though objects are attractively simple and intuitive in their initial conception, programming languages that support object-orientation are subtle and pose significant challenges for researchers.

Research on the Foundations of OOP began in the mid eighties. Many of the first inspirations came from denotation models, but it was quickly realised that existing semantic technology was not adequate to meet the demands of the object paradigm. For this reason, a number of researchers

adopted a more ad-hoc (but more effective in the short term) operational and type-theoretical approach. This has resulted in significant achievements, the most visible of which are calculi and safe type systems for OOP, which simplify and deal soundly with intricate issues involving inheritance. This lead to important semantics work on polymorphism and subtyping.

In the meantime there have been substantial advances in more “mainstream” semantics. Some of this has been directly relevant to or inspired by OOP, especially work on modelling subtypes and polymorphism. But there have also been substantial new advances using game semantics, functor categories, and process calculi. This has lead, for example, to a better understanding of local state and interaction, both of which are integral to the essence of the object concept.

The purpose of this workshop was to bring researchers from the two camps together. On one hand, OOP provides a great challenge for current semantic methods, and attempting to apply them will likely bring up new problems and give new insight on the methods themselves. On the other hand, a deeper semantic analysis of object-oriented languages can potentially impact program specification, type systems, and static analysis.

In addition to the talks presented at the workshop, there were numerous lively “corridor discussions”, and a wrap-up session, where the main themes and problems were discussed. Some of the central problems or issues that repeatedly arose included the following.

Objects versus Functions There are a number of translations from object-oriented to functional programming. Some researchers stressed, however, that these translations are somewhat indirect, and (especially) have difficulty explaining the typing concerns prevalent in OOP. For this reason, and also because translations go both ways, primitive object calculi are gaining currency, where translations to (pure or impure) functional programming are viewed as complementing the calculi, not replacing them.

State, Extrusion, and Identity Extrusion occurs when an object passes a newly created object to the outside; this occurs frequently in OOP (and even forms the basis of Hewitt’s Actor model). Purely functional object calculi do not give an accurate account of extrusion, however, because when the extruded entity is changed by the

receiver, this change needs to be reflected in the object itself: the distinction between sharing and copying is crucial here. But while “adequate” or sound models of extrusion can easily be given using traditional methods, more accurate semantics, accounting for locality, presents a challenge.

Specification Specification methods for objects remain a challenge. Particular problems include abstraction and inheritance, and imperative behaviour.

Also, while progress in type systems represents perhaps the most significant advances in Foundations of OOP, and has formed the core of prior work, interest continues and further work is progressing on a number of fronts.

20 Petri Nets and Business Process Management

Seminar No. **98271** Report No. **217** Date **05.07.–10.07.1998**
 Organizers: Jörg Desel, Andreas Oberweis, Wolfgang Reisig, Grzegorz Rozenberg

The management of business processes comprises their planning, specification, modeling, analysis and computer-supported execution. Current trends in information technology such as workflow management systems or packaged software systems indicate the increasing importance of a systematic approach to business process management. Many graphical notations are in everyday use for business modeling purposes, usually resembling some version of data flow diagrams. Often, the lack of formal semantics and methodological support limits their computer-based analysis and execution. In most cases, the design and the improvement of business processes in industrial applications does with intuition and experience of experts only.

Petri nets have been in use as a graphical modeling language for more than 30 years. Mathematical analysis techniques allow for analytical verification of many relevant properties of systems’ behaviour. Petri nets nevertheless are seldom used in business applications. Even the few commercially available Petri net based software development tools exploit

Petri net theory to a very limited extent only. There is an apparent gap between practical needs of business process management in industry and theoretical investigations of Petri nets in the academic sector.

Recently, different variants of Petri nets have been suggested for various purposes in the context of business processes: As a reference language for the semantics of other, formal or semi-formal modeling languages, or as a means to compare and relate different modeling languages, or as a tool to execute process models, or in order to check reachability of distinguished states.

Central problems of this area include:

- Which specific requirements for modeling languages arise in the area of business process management and workflow management (related to office documents, organizational dependencies of tasks, cost, time and quality aspects)?
- How can the analytical and methodological potential of Petri nets be exploited for business process management?
- Which requirements are not met by classical Petri net models? Which extensions of the model are sensible or necessary?
- Will Petri nets eventually play a role for business process modeling which compares to today's role of the relational model for data modeling? Will there develop a normalizing theory for business processes similar to the normalization theory of relational data bases?
- May Petri nets formally link application oriented modeling languages to particular workflow programming languages?

The seminar had more than 30 participants from various research and application areas, including Petri nets, business process modeling, workflow management, software and information system engineering.

21 Semi-Formal and Formal Specification Techniques for Software Systems

Seminar No. **98281** Report No. **218** Date **12.07.–17.07.1998**
Organizers: Hartmut Ehrig, Gregor Engels, Fernando Orejas, Martin Wirsing

Participants meeting at Dagstuhl
Considered the weather as too cool!
The idea of a pool
Would attract just a fool;
A firm rule is: wear things of good wool!
Bernd Krieg-Brückner

During the last 20 years several different formal and semi-formal specification techniques have been successfully developed and used. Applications comprise the specification of simple programs, data types and processes as well as complex hardware and software systems. The variety of specification techniques ranges from formal set theoretical, algebraic, and logic approaches for specifying sequential systems and from Petri nets, process algebras, automata, and graph grammars for specifying concurrent and distributed behaviors to semi-formal software engineering methods for developing complex systems.

Formal and semi-formal approaches have their advantages and disadvantages: the informal diagrammatic methods are easier to understand and to apply but they can be ambiguous. Due to the different nature of the employed diagrams and descriptions it is often difficult to get a comprehensive view of all functional and dynamic properties. On the other hand, the formal approaches are more difficult to learn and require mathematical training. But they provide mathematical rigor for analysis and prototyping of designs. Verification is possible only with formal techniques.

Since a few years many researchers and research groups are putting more and more effort in closing this gap by integrating semi-formal and formal specification techniques. Their studies and experiences show the added value of combining semi-formal and formal techniques and at the same time open a whole range of new problems and questions which

cannot be asked when studying formalisms in isolation.

In this seminar more than 40 scientists came together in 28 talks and two panel discussions to study possibilities and solutions for integrating and validating different formal and semi-formal specification techniques. Similarities and differences of formal and semi-formal specification formalisms as well as possibilities for combining such techniques were discussed. Most talks of this seminar analysed, compared, or integrated at least two such methods.

22 Graph Algorithms and Applications

Seminar No. **98301** Report No. **219** Date **26.07.–31.07.1998**
Organizers: Takao Nishizeki, Roberto Tamassia, Dorothea Wagner

Algorithmic graph theory is a classical area of research by now and has been rapidly expanding during the last three decades. In many different contexts of computer science and applications, modelling problems by graphs is a natural and canonical process. Graph-theoretic concepts and algorithms play an important role in many fields of application, e.g. in communication network design, VLSI-design, CAD, traffic optimization or network visualization.

Apart from the design and analysis of algorithms for solving fundamental graph problems, the application of these methods to real world problems is an interesting task for researchers in algorithmic graph theory. Recently, researchers also started developing software systems for graph algorithms to provide effective computational tools to support applications prototyping, algorithm animation or further algorithmic research. Several algorithm libraries, algorithm animation tools or special purpose software packages, e.g. graph editors and graph drawing software, have been developed within the last five to ten years.

This seminar was intended to bring together researchers from different areas in algorithmic graph theory and from application. One aim was to support the collaboration between computer scientists, mathematicians, and applied researchers, both from academia and industry in the field.

Main topics of interest were on the one hand classical problems from graph theory such as connectivity and cuts, paths and flows, coloring problems and theoretical aspects of graph drawing. On the other hand,

problems from application where those concepts are of special importance were discussed. Particular emphasis was placed on experimental research and aspects of the implementation of graph algorithms. One of the central topics was “graph drawing”, which addresses the problem of visualizing structural information. The automatic generation of drawings of graphs has important applications in key computer science technologies such as database design, software engineering, VLSI and network design and visual interfaces. Applications in other sciences concern all fields of visual data mining, e.g. in engineering, chemistry and biology, archeology or sociology and political science. The interaction between theoretical improvements and implemented solutions is an important part of the area of graph drawing.

We had 48 participants from Austria, Germany, France, Italy, Poland, Hungary, Slovenia, Israel, Australia, USA, Canada, Japan and Korea. During the workshop, 38 lectures, some including also software demonstrations, were given. Most of the talks presented very recent research results. The informal character of the workshop made it possible to have intensive discussions. There was an open-problem-session and a lively discussion on new directions in graph drawing. Both, senior researchers as well as young researchers, contributed to this seminar.

The organizers plan to edit a special volume of the *Journal of Graph Algorithms and Applications* with selected papers addressing areas covered by the seminar.

23 Self-Stabilization

Seminar No. **98331** Report No. **220** Date **17.08.–21.08.1998**
Organizers: Anish Arora, Shlomi Dolev, Willem-Paul de Roever

Distributed systems substantially improve our ability to compute and exchange information, as is evidenced by the dramatic success of the so-called World Wide Web. At the same time, distributed systems – and computer networks in particular – are hard to design, control, and maintain, as they consist of a variety of complex hardware and software components that are subject to faults and dynamic changes.

Self-stabilization has emerged as a promising paradigm for the design, control, and maintenance of fault-tolerant distributed systems. As its

name suggests, self-stabilization enables systems to automatically recover from the occurrence of faults. Its essential idea is this: Regardless of what state a system is placed in, by virtue of being self-stabilizing, the system converges to desired behavior. Thus, even if faults cause the system to be placed in an arbitrary state, the system can eventually resume its desired behavior.

The field of self-stabilization is young and rapidly growing. To facilitate research in this field, experts in this field and in allied fields were invited to share their research interests and work with each other. The Dagstuhl Seminar on "Self-Stabilization" brought together thirty five researchers from seven different countries. The opening talk was given by Edsger W. Dijkstra, then an overview of the state-of-the-art and future directions was provided by Shmuel Katz (see below). The talks that followed presented new results and directions:

- Formal methods for verification and specification of self-stabilizing algorithms,
- Use of the self-stabilization concept in the context of security and privacy,
- Integration with other fault models,
- Transient fault detectors,
- Design frameworks for achieving self-stabilization and other fault tolerances,
- Self-stabilizing algorithms and their time/space efficiency; impossibility results

Evaluating Self-Stabilization (by Shmuel Katz)

The goals, problems, achievements, and remaining challenges of self-stabilization were analyzed in this talk—obviously a personal perspective. The overriding goal in this area is seen as gaining acceptance of self-stabilization as a standard form of fault-tolerance. To achieve this, developing algorithms, transformations, complexity measures, and clearly stated models of computation are crucial. To these should be added the goals of compositionality and modularity of both algorithms and correctness reasoning.

Among the problems in this area are the common perception of self-stabilization as too strong a fault-model for distributed systems. Both data and control are arbitrary in an initial state, with nothing safe from corruption, a stable state cannot be reliably identified from within the system, and the algorithms are inherently non-terminating. Other problems with self-stabilization are self-inflicted by the research community: papers that ignore significant differences in computational models, use unstated limiting assumptions, and advocate problematic complexity measures. Beyond these obstacles to acceptance, self-stabilization is seen as ‘different’ from other types of faults, orthogonal to crash, send-receive, or Byzantine faults.

After considering alternative definitions of the term, and the variety and sensitivity of related models, the value of general techniques was emphasized. Among these are *prodding*—having some process able to send a message without receiving one first, *stamping*—adding process ids of those processes that receive information before passing it on, and *flushing*—removing error-filled initial messages by guaranteeing that fresh round numbers are generated. Especially valuable are techniques for *composing* self-stabilizing algorithms. One way to verify a parallel composition is to show that each component does not interfere with the proof of correctness of the other component.

Such techniques should be reused and further developed, precise proof schemas and robust complexity measures should be used, and appropriate application areas should be identified to allow fully integrating self-stabilization into general fault-tolerance.

24 Tiling for Optimal Resource Utilization

Seminar No. **98341** Report No. **221** Date **23.08.–28.08.1998**
Organizers: Jeanne Ferrante, Wolfgang Giloi, Sanjay Rajopadhye, Lothar Thiele

The following statement, written by the organizers, was included with the invitations to the participants. Its scope was deliberately somewhat narrow, and had the useful effect of forcing the participants to view a large number of compiler optimizations through “tiling colored” glasses.

Tiling is a regular partitioning of a uniform index space representing either computations (e.g., the iteration space of a loop program), or

data (e.g., arrays distributed over the processors of a parallel machine). Tiling can be used to achieve many different performance goals, such as exploiting data locality in hierarchical memory machines, communication optimization by message aggregation, communication-computation overlap, and latency avoidance.

Being such a common paradigm, tiling is used by many different communities in computer science, each with slightly different perspectives. Application writers tune a given program for performance by hand using multiple instances of tiling, sometimes sacrificing portability and ease of programming and debugging. Compiler writers have the same performance goals, but apply tiling automatically to a wide class of programs. Hence their considerations include feasibility of an automatic solution, reasonable computation times, and efficiency and ease of automatic code generation. VLSI processor array and embedded system designers have other constraints in the global context of the application (real-time throughputs, power consumption), but are willing to accept slow compilers (or design assistant tools). And certain problems may be best resolved at run time, since all parameters may not be available at compile time.

Underlying all of these applications of tiling is the issue of optimality in the presence of limited resources, embodied as a non-linear discrete optimization question.

- Is it possible to accurately model the machine behavior with a few cost parameters for a wide variety of machines?
- Is it possible to use such a model to predict the cost of a given program?
- How tractable is the resulting optimization problem (is it preferable to retain a less accurate model in the interests of tractability)?
- What are the consequences of separating multiple applications of tiling into separate optimization problems?
- Is a more global approach needed?

This seminar brought together researchers from these diverse groups to foster cross fertilization between them. We examined the differences in the problems, the models, and techniques they use, and the quality of their solutions, and how they interact with ultimate performance of an ENTIRE system.

The outcomes of the meeting include

- Fruitful and ongoing discussion on all aspects of tiling (through the creation of a web site <http://www.irisa.fr/api/Rajopadhye/tiling> and a mailing list, tiling@irisa.fr).
- A collection of abstracts of all the lectures, made publicly available at the web site.
- A panel discussion and followup activities on the creation of a common set of benchmarks for researchers in the community.

25 Architectural and Arithmetic Support for Multimedia

Seminar No. **98351** Report No. **222** Date **30.08.–04.09.1998**
Organizers: Guy Even, Peter Kornerup, Wolfgang Paul

With the widespread dissemination of PCs and workstations with Internet access in homes and offices, there has been an increasing quest for computational resources to support ever more demanding multimedia applications, like audio, video, virtual reality and games employing advanced graphics. Beyond increasing the performance of CPUs through pipelining and superscalar execution, in the past few years a number of CPU manufacturers have enhanced their instruction sets with special instructions to support such applications. In particular SIMD-types of basic arithmetic operations have been introduced, exploiting parallelism on new and typically shorter data types, to support video decompression and other compute intensive applications. But also special arithmetic instructions like fast reciprocal and root-reciprocal operations are emerging to support special graphics applications. The increasing CPU processing power places extra demands on the overall system architecture, on cache, bus, memory and I/O bandwidth, problems exaggerated by this development.

The purpose of this seminar was to bring together researchers and practitioners from three different topic areas: Multimedia architectures, the general purpose systems and processor architecture area, and the

arithmetic design and implementation area. Our invitations were well received, many expressed their enthusiasm about this particular combination of topics. Unfortunately some had to decline participation due to other obligations, in particular some people from industry could not attend due to urgent work, but expressed their hope that this type of event could be repeated later. A total of 29 people attended the seminar, 27 formal talks were given, together with an “open problem session” and numerous informal discussions in smaller groups. The facilities of Schloss Dagstuhl were extensively utilized, even such that a formal conference submission was conceived, prepared and submitted during the week, meeting an imminent deadline.

The talks presented covered a wide spectrum of problems from the three areas, ranging from problem identifications to specific algorithmic or implementation designs, from MPEG video decompression to multimedia and hypertext books (with publication issues), basic arithmetic to implementation of standard functions, floating point arithmetic to signal- and image processing, and issues like pipelining, hardware scheduling, precise interrupts, multi threading, encryption and compiler optimizations.

26 Test Automation for Reactive Systems – Theory and Practice

Seminar No. **98361** Report No. **223** Date **06.09.–11.09.1998**
Organizers: Ed Brinksma, Jan Peleska, Michael Siegel

The design and implementation of correct reactive systems is one of the major challenges of information technology. More than ever before modern society has become critically dependent upon applications of such systems. Important application areas include telematic systems, such as communication protocols and services, process control systems, and embedded-software systems, ranging from aircraft control to consumer products like television sets. Most applications are critical in one or more aspects, e.g. in the context of safety, economy, ecology etc.

Testing is an indispensable technique for the validation of reactive systems. Although other tool supported methods, such as simulation, model checking and verification by theorem proving, are increasingly be-

ing applied to assess the correctness of system designs, testing remains the practically most widely used validation technique. Although testing is generally too weak to guarantee correctness, its advantages are that it is more readily applicable to large industrial systems, and that it is in fact the only validation technique that can link the functionality of a (physical) realization and its formal specification.

On the one hand, the great growth of reactive system applications and with that the growing need to assess their correctness, has brought about a considerable practical interest in the improvement of their testing procedures. On the other hand, academic researchers, who formerly regarded testing as an inferior and superfluous validation method, have begun to investigate the use of formal methods and tools to specify, generate and implement tests. This is of great practical interest, as most test suites for reactive systems are still being produced on the basis of ill-understood ad hoc procedures. As a result the production and maintenance of test suites have become very expensive activities, presenting a considerable practical bottleneck. Even minor improvements on existing working procedures are economically attractive. Major improvements are expected from algorithms for the (semi-)automated derivation and selection of test suites from formal specifications of the implementations under test, as well as the machine-assisted evaluation of test results.

In spite of these bright prospects for the application of testing theory and related tools, still much work is needed to turn the existing theories and tools into an effective testing workbench fit for industrial application.

The Dagstuhl workshop on test automation for reactive systems was the first of its kind, and was very well attended (more than 40 participants). It brought together people from different backgrounds with a strong interest in testing, such as protocol testing, real-time testing, performance testing, statistical testing, etc., to present and discuss the state-of-the-art in testing and test automation. There was a strong representation from the formal methods community, but here again a wide range of models and formalisms was represented: Mealy automata, labeled transition systems, abstract data types, I/O automata, timed automata, Z etc.

The presentations and discussions made clear that although there are quite a few research groups active in the area now, we are only at the beginning of things and major contributions to the (semi-)automation of practical software testing are still to come. A bottleneck is the limited

availability of formal system specifications that can be used to derive tests from. In certain areas, however, such as e.g. safety-critical and embedded systems, the willingness of industry to go more formal is growing, which is among others motivated by the potential gains for testing implementations.

A discussion session that was held to talk about future research themes in test automation suggested at least the following topics: relating test methods to common design and implementation methodologies (e.g. objected oriented design), the issue of design for testability, which has been very successful in the area of hardware testing, and measures to quantify the coverage of tests. The general opinion was that the seminar was a success and it is planned to organize another Dagstuhl meeting on testing in the future.

27 Modeling and Planning for Sensor-Based Intelligent Robot Systems

Seminar No. **98391** Report No. **224** Date **27.09.–02.10.1998**
Organizers: Robert Bolles, Horst Bunke, Henrik I. Christensen, Hartmut Noltemeier

Intelligent Robot Systems are very complex soft- and hardware systems. These robots include facilities for action selection based on sensory input and prior knowledge. Intelligent algorithms are interface between sensors and actions. The research area includes significant theoretical and practical problems, especially high-level planning (using AI techniques), geometric methods, real-time software systems and methods for interpretation and fusion of sensory information.

A significant problem is modelling of dynamically changing environments, and robust methods for sensory perception. The gap between theory and practice is a wide one and a lot of research must be done for example for building a human-like robot.

Results

Many researchers from Europe, Asia and the USA met in an ideal setting and discussed in depth the following issues:

- Path and Motion Planning, Navigation
The problem of exploring a simple unknown environment and searching for the best competitive ratio.
- Localization and Pose Tracking
Relocalization especially using partial map matching or polygonal distances.
- Vision, Shape and Object Recognition
Theoretical and practical shape and object recognition using 2D or 3D sensing is the main objective.
- Distributed Systems
Using multiple robots interacting and cooperating together and solving distributed problems is an important question.
- Applications
Intelligent, helpful and social missions of an intelligent robot are demonstrated.
- Human Robot interaction
The interface between human and robot is an important research topic. Here some aspects are showed.
- Current state of the art discussions
How to design working and climbing machines, discussion about telerobotics, and an open problem discussion session.

Perspectives

It is now recognised that the ultimate flexible robot is a fantasy. The trend is towards specialised robots that are designed for a particular application. The manufacturing of specialised robots must however be based on standardised components to make the final product economically tractable. Overall trends in end-user products and the corresponding needs for basic research were discussed in the open problem session.

Sessions

How to design walking machines

Rüdiger Dillmann, University of Karlsruhe, Germany

Henrik I. Christensen, Royal Institute of Technologie Stockholm,
Sweden

Ewald von Puttkamer, University of Kaiserslautern, Germany

The commercial applications for climbing and walking robots were initially recognised by the nuclear and space industries which have had to address clear and identifiable problems of performing maintenance tasks within hazardous and unstructured environments intrinsically hostile to man. Discussions with industrialists and researchers have highlighted the following:

- Wall climbing vehicles are needed for remote inspection and maintenance especially in the nuclear industry.
- Underwater applications for mobile machines including inspection of bridges, clearing intake pipes of hydroelectric plants, inspection of fouled drains, cooling water ducts and sewage outfalls. Robots able to accomplish these types of tasks would be capable of adaptation for ship cleaning whilst the vessel is at sea.
- The use of climbers and walkers by the emergency services to enter dangerous areas devastated by fire or earthquake to collect samples, and to search for survivors would be invaluable.
- The clearance of anti-personnel land mines within the context of humanitarian missions could be usefully carried out by low cost legged robots with specifically designed sensors.

Research to develop such climbers and walkers continues to focus on these applications. The locomotion can be based on articulated legs if rough ground needs to be negotiated, gripping feet if vertical structures have to be climbed or some special motion capability in certain circumstances.

Telerobotics

Åke Wernersson, Lulea University of Technology, Sweden

This discussion gives a short overview about telerobotics. The first question is, what do we mean? Telerobotics is a master-slave telemanipulation of a robot. What are the problems? We summarize a lot of it without solving them:

- What is the theory for this problem?
- Definition of telecommands for telerobotics. We propose gripping, generating of maps, docking and loading, detecting changes in workspace.
- How can we interpret scenes?
- We need tools for building telecommands.
- We need new ways for transfer of results so we propose a telerobotics lab on the internet.
- We need self monitoring onboard the robot.
- How can we handle the telecommunication time delay?

Most of the problems are unsolved and undefined, so much work must be done to build up a theoretic framework for telerobotics.

Open Problems and wine-/cheese session

The Dagstuhl-Seminar gave a deep view into actual research areas of robotics. But there are a lot of open problems:

- Benchmarking
How can we compare different robot-systems? Why do we only measure our systems with worst-case scenarios and not with average-case scenarios? How do we define measures, scenarios, rooms that can be used as benchmarks for all robots?

- Human-Robot-Interaction
How must an interface between a human and a robot appear? Is there a dialogue necessary? Can an interface be designed that is very simple and intuitive for humans?
- Energy
Mobile robots do have a battery with limited time. We need better batteries with much longer operating time for more complex robots.
- Learning, Adaption and Introspection
Can robots learn several tasks? Can they do things alone without human control? Can they explain things they do?
- Challenges
There are a lot of challenges, to be solved:
 - Distributed Robot Systems
Robots can interact in simple problems. How can they interact to solve large problems? Must they communicate among themselves?
 - Reliable object recognition
Object recognition is highly non-robust today. How can this be performed in a better way?
 - Low cost robots
How can we reduce the cost to design a robot?
 - Self monitoring
Can robots repair themselves? Can they recognize their own failures?
 - Large c-space mobile manipulation
Robot motion nowadays is a discrete thing and not an continuous thing. How can we design smooth motion?
 - Theoretical vs. experimental use
There is a gap between theoretical and experimental research. How can we overcome this problem?
 - Telerobotics
We must define a model of telerobots. What does it mean?
 - Emotions
Does a robot need emotions?

- Wine and Cheese
Does a robot like wine and cheese, too?

28 Algorithms and Number Theory

Seminar No. **98431** Report No. **225** Date **25.10.–30.10.1998**
Organizers: Harald Niederreiter, Andrew Odlyzko, Michael E. Pohst

This seminar was the third one on number theoretical algorithms at Dagstuhl over the past 7 years. A major goal was always to bring together number theorists who develop the theory for efficient algorithms and people writing the corresponding software for applications. This year we had 42 participants from 13 countries.

In the last few years number theoretical applications to Coding Theory and Cryptography have become more and more important. Hence, it was no surprise that the majority of talks was on topics related to these applications. We would like to mention:

- computations with elliptic curves over finite fields; several new and efficient methods were presented; elliptic curve methods are currently under consideration for becoming part of the new standard for public key cryptosystems;
- primality testing and proving, large primes being of importance for quite a few cryptosystems;
- finite field algorithms, factorization of polynomials over finite fields; the ability to do efficient computations in and with finite fields is a basis for almost all algorithms applied in practice in the areas mentioned; factoring methods for polynomials over finite fields were tremendously improved over the last years;
- class group computations in global fields; since the usefulness of class groups of quadratic number fields for cryptographical applications was demonstrated, this has become a new area of research on a class of basic objects from pure mathematics; as for now the constructive approach is still limited to global fields of small degree.

In the other talks given a large variety of problems in algorithmic (algebraic) number theory was treated.

29 Adaptation and Evolution in Embedded Information Systems

Seminar No. **98441** Report No. **226** Date **01.11.–06.11.1998**
Organizers: Hubertus Franke, Bernd Kleinjohann, Janos Sztipanovits

Among the most significant technological developments of the last two decades has been the proliferation of embedded information systems (EISs). In EISs, functional, performance and reliability requirements mandate a tight integration of information processing and physical processes. EISs include a wide range of applications, such as computer-integrated manufacturing (CIM) systems, aerospace systems, computerized vehicles, appliances, consumer electronics equipment, and a wide variety of systems in health care, transportation, defense, communication, power generation and distribution. The rapid evolution of EISs has triggered paradigm shifts in industry and exerted a profound impact on engineering processes throughout the system life cycle, from design through manufacturing, operation and maintenance. This trend is clearly demonstrated by the dramatic increase of the role and size of software in products. For example, currently, over 60% of the development cost of aerospace systems is software, but even an electric shaver has over 16K bytes software.

The ultimate driver of this trend is the fact that incorporation of information processing as an integral part of physical systems increases the potential interactions among physical components and processes, generates complex dynamics, and establishes component interdependencies unknown in previous-generation systems. The tight integration of "physical" and "information" processes represents major challenges for the software technology. First of all, using Brook's terminology, the "conceptual construct" of the software which represents its "essential complexity" is inextricably combined with the conceptual construct of its "external environment", i.e. with the structure of physical processes. Consequently, the overall system behavior can only be understood if information, material and energy transfer processes are modeled and analyzed together. It means that software artifacts need to be modeled together "with their context", using a modeling language – or modeling paradigm – which is meaningful for the design, analysis and operation of the whole system. Another well-known challenge in the design and implementation of EIS

software stems from the fact that it serves as a component in a larger, changing, heterogeneous system. As a result of this, the EIS software cannot be static, it must change, evolve together with its embedding environment. An additional challenge of EIS software technology is criticality. EIS software directly impacts the operation of physical processes and failure may cause unacceptable social or economic damage. Thus the software technology must offer methods and tools for the formal verification and validation of system level requirements, such as dependability, safety and reliability.

This seminar intended to bring together leading researchers from academia and industry in the areas of embedded information systems, adaptive software architectures and evolutionary design environments. The goal is to review the state-of-the-art and map future research directions that help to answer challenges in EIS, and to investigate the potential of applying recent advances in the aforementioned fields. The seminar included the following topics:

1. Definition and characterization of adaptive and evolutionary embedded systems:
 - needs
 - characteristics
 - case studies
2. Critical technology components
 - self-adaptive software architectures
 - architecture-based evolution
 - incremental, embeddable generators
 - transient management
3. Design environments
 - design abstractions, hybrid/heterogeneous modeling
 - verification, correctness by construction
 - incremental design
 - design optimization

Further information and presentation material are made available under <http://www.isis.vanderbilt.edu/dagstuhl98>

30 CAD-Tools and Methods for Systems Development

Seminar No. **98461** Report No. **227** Date **16.11.–20.11.1998**
Organizers: P. Brunet, C. Hoffmann, D. Roller

Continuously shrinking innovation cycles for technical products enforce highly sophisticated computer-based development tools. These tools need to go far beyond conventional CAD systems. Significant progress steps in research and development of CAD tools for the design of new products therefore are playing a key role in respect to the competitive power of complete industries.

This seminar at the International Meeting and Research Center for Computer Science at Schloss Dagstuhl Germany focuses on this important domain. Of particular interest are hereby methods for solid modelling, constraint and feature-based design of object families, product design, efficient integration of tools for calculation, simulation and analysis, design tools for very complex systems and assemblies, as well as rapid prototyping.

During the seminar, new solutions for these important problem areas will be discussed among leading experts. Besides the research presentations, this seminar is intended to provide a forum for discussion and exchange of ideas and participant experiences. In this seminar, we focus on new developments in this dynamic research areas:

- Solid and surface modelling.
- Constraint and feature-based design of object families.
- Feature-based product design.
- Design tools for very complex systems and assemblies.
- Efficient integration of design, simulation and analysis.
- Future CAD architectures.

31 Integrating Spatial and Temporal Databases

Seminar No. **98471** Report No. **228** Date **22.11.–27.11.1998**
Organizers: Oliver Günther, Timos Sellis, Babis Theodoulidis

Spatial databases incorporate the notion of space in order to accommodate the requirements for databases that allow reasoning about 2D and 3D such as geographical applications (GIS). Their study exists for more than twenty years. Lately, it is triggered even more by the progress achieved in the power of computers which permits them to accommodate graphics and easily perform geometrical calculations. Spatial databases form an autonomous, active research community and a series of International Conferences are regularly organised (series of Symposium on Spatial Databases and Symposium on Spatial Data Handling). A number of journals is concerned with spatial databases as well (Cartographica, International Journal of Geographic Information Systems). The National Center for Geographic Information and Analysis (NCGIA, USA) is an established body coordinating research in Spatial Databases and their beneficial application in geography. OpenGIS is an International Consortium trying to bring interoperability into Geographic Information Systems.

Temporal databases incorporate the concept of time to create high-level abstractions useful in database applications. This has been an active area of research for about twenty years. In the last few years the importance of the temporal database area has been recognised by the international scientific community. This recognition came in part in the form of the ARPA/NSF sponsored International Workshop on Temporal Database Infrastructure in 1993, a VLDB-affiliated temporal workshop in 1995, a special section of the IEEE Transactions on Knowledge and Data Engineering on temporal and real time databases published in August 1995, and the incorporation of temporal constructs, proposed by the temporal database community, in the soon-to-be standardised SQL3 language.

The main objective of the seminar was to bring together researchers from the two areas that have been working independently from each other and only recently have started to talk to each other. For example, research work on integration has started appearing on the main conferences

and publications of each discipline.

One of the main issues discussed was whether it is feasible and if yes, how the research should be further integrated and if possible, what the mechanisms should be that the community can define so as to accelerate the process of developing a spatiotemporal infrastructure.

The “Integrating Spatial and Temporal Databases” seminar focused on establishing the foundations of a new discipline and also the future directions of that discipline, with respect to both research issues and the means to incorporate spatiotemporal databases into main-stream application development. A list of topics discussed at this seminar follows:

- Strategic discussions about the future of spatiotemporal databases as a discipline. Evaluation of the current state of the art with respect to the current trends in the DBMS tools and standards.
- Research Issues in Spatial and Temporal Databases: What is important?
- Spatiotemporal data models: relational, object-oriented, deductive and hybrid models. Where do the spatial and temporal capabilities fit in?
- Spatiotemporal user interfaces and languages. Update and retrieval languages for various types of temporal data models.
- Implementation issues in spatiotemporal databases. Issues that arise from experience of implementors and users and the agenda for research into these areas and transition to use in practice.
- Issue a “call for action” to the community (academia and vendors alike).

This seminar brought together over sixty researchers from fifteen countries that have dealt with different disciplines (spatial and temporal), as well as developers of databases and users, to conduct a fruitful discussion and evaluation of the activities thus far with a view on establishing the foundations of a new discipline: that of spatiotemporal databases. There was a general agreement that there is still work to be done in Spatiotemporal Design, Data Models, Query Languages and Indexing while areas such as Temporal Data Models and Algebras are almost complete. Spatiotemporal Data Mining, Query Processing and Optimisation will produce significant results in the next ten years.