06021 Abstracts Collection
Reliable Implementation of Real Number Algorithms: Theory and Practice
— Dagstuhl Seminar —

Peter Hertling[1], Christoph M. Hoffmann[2], Wolfram Luther[3] and Nathalie Revol[4]

[1] Univ. der Bundeswehr - Neubiberg, DE
peter.hertling@unibw.de
[2] Purdue Univ., US
cmh@cs.purdue.edu
[3] Univ. Duisburg-Essen, DE
luther@informatik.uni-duisburg.de
[4] ENS - Lyon, FR
nathalie.revol@ens-lyon.fr

Abstract. From 08.01.06 to 13.01.06, the Dagstuhl Seminar 06021 "Reliable Implementation of Real Number Algorithms: Theory and Practice" was held in the International Conference and Research Center (IBFI), Schloss Dagstuhl. During the seminar, several participants presented their current research, and ongoing work and open problems were discussed. Abstracts of the presentations given during the seminar as well as abstracts of seminar results and ideas are put together in this paper. The first section describes the seminar topics and goals in general. Links to extended abstracts or full papers are provided, if available.

Keywords. Real number algorithms, reliable implementation

06021 Summary – Reliable Implementation of Real Number Algorithms: Theory and Practice

The seminar brought together researchers from many different disciplines concerned with the reliable implementation of real number algorithms either from a theoretical or from a practical point of view. In this summary we describe the topics, the goals, and the contributions of the seminar.

Keywords: Real number computability, real number algorithms, reliable computing, algorithms with result verification, interval arithmetic, geometric computing, robustness, solid modeling

Joint work of: Hertling, Peter; Hoffmann, Christoph M.; Luther, Wolfram; Revol, Nathalie

Extended Abstract: http://drops.dagstuhl.de/opus/volltexte/2006/711

Computable Errorbounds for Eigenpairs

Götz Alefeld (Universität Karlsruhe, D)

We report on a method which allows to improve an approximate eigenpair iteratively. In each step we compute intervals in which the eigenpair is contained. For sufficiently accurate initial approximations for an eigenpair with an algebraic simple eigenvalue convergence to the eigenpair can be proved. The method is also discussed with respect to rounding errors in a floating point system.

Validated Modeling of Multibody Systems: Reducing the Wrapping Effect

Ekaterina Auer (Universität Duisburg-Essen, D)

It is a well known fact that computer simulations of different real life processes can generate erroneous results, in many cases due to the use of the standard IEEE arithmetic. To ensure the correctness of the results obtained with the help of a computer, various kinds of validating arithmetics and algorithms were developed. Their purpose is to provide bounds in which the exact result is guaranteed to be contained. Verified modeling of kinematics and dynamics of multibody systems is a challenging application field for such methods, largely because of possible overestimations (the wrapping effect, cancellation, dependency) of the guaranteed bounds leading to meaningless results. In this talk, we discuss approaches to validated multibody modeling and present a recently developed template-based tool SmartMOBILE for verifying kinematical as well as dynamic properties of different types of multibody systems, which features the possibility to choose an appropriate kind of arithmetic according to the modeling task. We focus on the different strategies of the wrapping effect reduction in SmartMOBILE, including improvements in the underlying data types, modeling elements and solvers. In general, this talk provides information on enhancement of the already existing modeling software with validating techniques, extension of modeling possibilities through appropriate choices of validated arithmetics, and reduction of the wrapping effect. We conclude with an overview of the current tasks planned to increase the applicability of the tool.

Keywords: Validated modeling, multibody systems

Specifications via Realizability

Andrej Bauer (University of Ljubljana, SLO)

We present a system, called RZ, for automatic generation of program specifications from mathematical theories. We translate mathematical theories to specifications by computing their realizability interpretations in the ML language augmented with assertions (as comments).

While the system is best suited for descriptions of those data structures that can be easily described in mathematical language (e.g., finitely presented groups, real arithmetic, vector spaces, etc.), it also elucidates the relationship between data structures and constructive mathematics.

Keywords: Realizability, specifications, scientific computing

Joint work of: Bauer, Andrej; Stone, Christopher

Formal Proofs on Floating-Point Evaluation of Elementary Functions

Sylvie Boldo (INRIA Futurs - Orsay, F)

Elementary functions are widely used in floating-point arithmetic but their correctness is hardly specified due to the complexity of error evaluation concerning argument reduction and polynomial evaluation. I will present my latests related results (with M. Daumas, J.-M. Muller and C. Muñoz).

Towards Computability of Elliptic Boundary Value Problems

Vasco Brattka (University of Cape Town, ZA)

We present computable versions of the Fréchet-Riesz Representation Theorem and the Lax-Milgram Theorem. The classical versions of these theorems play important rôles in various problems of mathematical analysis, including boundary value problems of elliptic equations. We demonstrate how their computable versions yield computable solutions of the Neumann and Dirichlet boundary value problems for a simple non-symmetric elliptic differential equation in the one-dimensional case. For the discussion of these elementary boundary value problems we also provide a computable version of the Theorem of Schauder, which shows that the adjoint of a computably compact operator on Hilbert spaces is computably compact again.

This is a joint work with Atsushi Yoshikawa, Kyushu University, Japan.

Keywords: Computable analysis

The SOFA toolbox for Evaluating Range of Outputs under Simulink

Jean-Marie Chesneaux (Université Paris VI, F)

When embedded functionalities are designed, an important information is the knowledge of the ranges of outputs from the ranges of the data.

From a new modelization and a new statistical test, we present the SOFA toolbox, which works under Simulink, to evaluate dynamically such ranges of outputs.

Joint work of: Chesneaux, Jean-Marie; Didier, Laurent-Stéphane; Rico, Fabien

Towards an Interval Subroutine Library

George F. Corliss (Marquette University, USA)

The interval/reliable/validated computing community has produced many packages for interval arithmetic and many problem-solving routines. For most topics in an elementary numerical analysis book, there is software that successfully solves many problems of modest size in a reliable manner.

Yet, if an engineer says, "I *get it.* What software do I use?" there is a long pause, because I have no CD or web site containing a portable, easy to use, comprehensive software library for reliable computations. A group of us are embarking on an long-term effort to assemble such a library from existing software where possible and by enlisting collaborators where new development is necessary.

The library is layered, beginning with a set of Basic Interval Arithmetic Subroutines (BIAS) and interval BLAS. We have a draft of a portion of the API of that layer. We have dreams of a comprehensive set of utilities such as Taylor models, automatic differentiation, and constraint propagation to support a library of problem-solving routines including linear and non-linear systems, eigenvalues, quadrature, optimization in several flavors, ordinary and partial differential equations, and more.

We seek your advice on the APIs for the BIAS and higher-level routines to follow. What can such a library provide that would make it easier for you to develop reliable engineering applications? We seek your contributions. What software from your work belongs in such a library?

Keywords: Subroutine library, problem-solving library, C++ interval standard

Interval Subroutine Library Mission

George F. Corliss (Marquette University, USA)

We propose the collection, standardization, and distribution of a full-featured production quality library for reliable scientific computing with routines using interval techniques for use by the wide community of applications developers.

Keywords: Subroutine library, problem-solving library, C++ interval standard

Joint work of: Corliss, George F.; Kearfott, R. Baker; Nedialkov, Ned; Pryce, John D.; Smith, Spencer

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/712

Convex Polyhedral Enclosures of Interval-Based Hierarchical Object Representations

Eva Dyllong (Universität Duisburg-Essen, D)

In this talk, we discuss approaches to construct convex polyhedral enclosures of interval-based hierarchical structures. Hierarchical object representations are the most frequently used data structures for the reconstruction of a real scene. This object modelling does not depend on the nature of a real solid but only on the chosen maximal level of the tree. This is a useful property for objects with complex structures that are difficult to describe via exact mathematical formulas. In this talk, we focus on reliable object modelling using the interval-based octree data structure. A feasible way to obtain a convex polyhedral enclosure of an octree is to use the concept of extreme vertices of the boxes forming the boundary. Accurate algorithms for constructing the convex hull of these vertices yield a convex polyhedron as an adaptive and reliable object enclosure at each level of the tree.

Keywords: Octrees, convex hull, interval algorithms, reliable geometry

Structure and Problem Solving in ISL

Baker Kearfott (Univ. of Louisiana - Lafayette, USA)

Our Interval Subroutine Library project's overall goal is to facilitate validated computation within the community of experts, and also to foster wider use and acceptance within the scientific computing community in general. To achieve these goals, we propose to employ a combination of clear, simple organization, careful documentation, and standardization. The standardization can be accomplished through participation, acceptance, and wide use. Here, we

- briefly explain how we will build on previous work,
- describe our proposed hierarchical structure of ISL from the point of view of solving practical problems,
- give a case study of how we might incorporate previous work in problem-solving routines.

Keywords: Interval arithmetic, software libraries, numerical analysis

Upper and Lower Bounds on Sizes of Finite Bisimulations of Pfaffian Dynamical Systems

Margarita Korovina (A. P. Ershov Institute - Novosibirsk, RUS)

In this paper we study a class of dynamical systems defined by Pfaffian maps. It is a sub-class of o-minimal dynamical systems which capture rich continuous dynamics and yet can be studied using finite bisimulations.

The existence of finite bisimulations for o-minimal dynamical and hybrid systems has been shown by several authors; see e.g. Brihaye et al (2004), Davoren (1999), Lafferriere et al (2000).

The next natural question to investigate is how the sizes of such bisimulations can be bounded. The first step in this direction was done by Korovina et al (2004) where a double exponential upper bound was shown for Pfaffian dynamical and hybrid systems. In the present paper we improve this bound to a single exponential upper bound. Moreover we show that this bound is tight in general, by exhibiting a parameterized class of systems on which the exponential bound is attained.

The bounds provide a basis for designing efficient algorithms for computing bisimulations, solving reachability and motion planning problems.

Keywords: Hybrid systems, Pfaffian functions, bisimulation

Joint work of: Korovina, Margarita; Vorobjov, Nicolai

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/713

How to Take into Account Dependence Between the Inputs: From Interval Computations to Constraint-Related Set Computations

Vladik Kreinovich (University of Texas - El Paso, USA)

In the traditional interval computations approach to handling uncertainty, we assume that we know the intervals $\mathbf{x}_i$ of possible values of different parameters x_i, and we assume that an arbitrary combination of these values is possible. In geometric terms, in the traditional interval computations approach, the set of possible combinations $x = (x_1, \ldots, x_n)$ is a box $\mathbf{x} = \mathbf{x}_1 \times \ldots \times \mathbf{x}_n$.

In many real-life situations, in addition to knowing the intervals $\mathbf{x}_i$ of possible values of each variable x_i, we also know additional restrictions on the possible combinations of x_i; in this case, the set $\mathbf{x}$ of possible values of x is a subset of the original box. For example, in addition to knowing the bounds on x_1 and x_2, we may also know that the difference between x_1 and x_2 cannot exceed a certain amount. Informally speaking, the parameters x_i are no longer independent — in the sense that the set of possible values of x_i may depend on the values of other parameters.

In interval computations, we start with independent inputs; as we follow computations, we get dependent intermediate results: e.g., for $x_1 - x_1^2$, the values of x_1 and $x_2 = x_1^2$ are strongly dependent in the sense that only values (x_1, x_1^2) are possible within the box $\mathbf{x}_1 \times \mathbf{x}_2$. In interval computations, there are many techniques for handling similar dependence between the intermediate computational results. In this talk, we extend these techniques to handle a different type of dependence — dependence between the inputs.

To achieve this objective, at any given stage of the computations — when, in addition to the input values $x_1, \ldots, x_n$, we also have intermediate computation results $x_{n+1}, \ldots, x_N$ — we not only store (enclosures for) intervals $\mathbf{x}_i$ of possible values of all the variable x_i, $i = 1, \ldots, N$; we also store, for all pairs (i, j), (enclosures for) sets $\mathbf{x}_{ij}$ of possible values of pairs (x_i, x_j) – enclosures described as pavings (i.e., unions of 2-D boxes).

We also discuss how this technique can be extended to situations when we also have a partial information about the probabilities.

The talk is based on our joint work with Martine Ceberio, Scott Ferson, Gang Xiang, Adrian Murguia, and Jorge Santillan.

Keywords: Interval computations, constraints

Interval Arithmetic Using SSE-2

Branimir Lambov (Univ. of Aarhus, DK)

We present an implementation of double precision interval arithmetic using the single-instruction-multiple-data SSE-2 instruction and register set extensions. The implementation is part of a package for exact real arithmetic, which defines the interval arithmetic variation that must be used: incorrect operations such as division by zero cause exceptions, loose evaluation of the operations is in effect, and performance is more important than tightness of the produced bounds. The SSE2 extensions are suitable for the job, because they can be used to operate on a pair of double precision numbers and include separate rounding mode control and detection of the exceptional conditions. The paper describes the ideas we use to fit interval arithmetic to this set of instructions, shows a performance comparison with other freely available interval arithmetic packages, and discusses possible very simple hardware extensions that can significantly increase the performance of interval arithmetic.

Keywords: Interval Arithmetic, SSE2

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/714

Fused Multiply and Add Implementations of the Compensated Horner Scheme

Philippe Langlois (Université de Perpignan, F)

The Fused Multiply and Add operator (FMA) that provides $x * y + z$ with only one final rounding error can be used to improve the accuracy and the speed efficiency of the Horner scheme for polynomial evaluation.

The compensated Horner scheme (CHS) only uses the working precision to yield a polynomial evaluation as accurate as if computed with (at least) twice the current precision.

This compensated Horner scheme can also benefit from the FMA. We present two different CHS with FMA and associated error analysis. We conclude with numerical results and identifying which CHS with FMA is the best one in terms of running time efficiency.

Keywords: Horner scheme, accuracy, FMA, compensated algorithms, polynomial evaluation, error free transformations

Joint work of: Langlois, Philippe; Graillat, Stef; Louvet, Nicolas

Real Number Calculations and Theorem Proving

David Lester (Manchester University, GB)

In this work we present a formally verified interval arithmetic package.

In addition the theorem prover has been augmented to automatically deduce the validity of (some) real number inequalities. This work has been carried out by NASA/NIA in order to validate avionics and air-traffic control systems.

The novelty lies in two areas. We believe that this is the first machine-checked formal proof for the validity of real number calculations that includes the transcendental functions sin, cos, atan, log (base e) and exp. Secondly it represents a first attempt at a partial strategy for proving real number inequalities.

Joint work of: Lester, David; Muñoz, César

Parameter Estimation for Exponential Sums

Wolfram Luther (Universität Duisburg-Essen, D)

This talk reports on a recent project “Traffic modeling in multiservice networks and analysis of resource requirements for quality of service support”.

The classical approach in queueing and service systems is to consider random variables for the interarrival times of events corresponding to the arrivals of packets, flows, connections or other units relevant for network elements. Two basic characteristics of the stochastic behavior of traffic are the distribution function of considered random variables and the autocorrelation of the process. To model the distribution function of arriving data as well as the autocorrelation function, semi-Markov processes (SMPs) can be used. Special emphasis is given to the accurate computation of workload distributions of service systems and the modeling of the autocorrelation function of a SMP(m) with m states as a superposition of $m-1$ geometrical terms including complex coefficients. Traffic data are used to recover the coefficients of the exponential sums. Stochastic methods or interval arithmetic can be applied to validate the parameter estimation and to guarantee the results of the analysis. We want to develop an interval Prony algorithm and to analyze its relation to the expectation maximization algorithm and other approaches to solve the parameter estimation problem.

Another interesting work in progress is the computation of transient workload distributions and of the time until the system reaches the equilibrium. This work completes a recently proposed algorithm to compute accurately the verified stationary workload distributions of GI/GI/1 and SMP/GI/1 service systems.

Keywords: Reliable algorithms, stochastic traffic modeling

Joint work of: Luther, Wolfram; Haßlinger, Gerhard; Kempken, Sebastian

Root Isolation of Polynomials with Bitstream Coefficients

Kurt Mehlhorn (MPI für Informatik - Saarbrücken, D)

The Descartes method is an algorithm for isolating the real roots of square-free polynomials with real coefficients. We assume that coefficients are given as (potentially infinite) bit-streams. In other words, coefficients can be approximated to any desired accuracy, but are not known exactly. We show that a variant of the Descartes algorithm can cope with bit-stream coefficients. To isolate the real roots of a square-free real polynomial $q(x) = q_n x^n + \ldots + q_0$ with root separation ρ, coefficients $|q_n| \geq 1$ and $|q_i| \leq 2^\tau$, it needs coefficient approximations to $O(n(\log(1/\rho) + \tau))$ bits after the binary point and has an expected cost of $O(n^4(\log(1/\rho) + \tau)^2)$ bit operations.

Keywords: Root isolation, interval arithmetic, Descartes algorithm

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/715

Certifying Numerical Applications: Gappa's Approach

Guillaume Melquiond (ENS - Lyon, F)

In a computer program, numerical computations do not behave as if they were made on real numbers. Because of the limited precision, the computed values slowly drift away from the exact values. Moreover, exceptional behaviors may occur (overflow, square root of negative numbers, etc). These "details" cannot be set aside when designing safety-critical applications. In order to correctly certify such an application, the arithmetic features of the targeted architecture have to be taken into account.

Certifying a numerical program by hand then becomes tedious and error-prone. In order to be able to complete such works, automatized tools are indispensable. Gappa is specially designed to verify arithmetic properties that usually appear when certifying numerical applications. It relies on interval arithmetic and forward error analysis to bound mathematical expressions involving floating-point rounded operators. In addition, the tool generates proof scripts of the numerical results. These scripts are formal certificates and the validity of the results can then be automatically verified by an independent proof-checker like Coq.

Gappa is used for the development of CRlibm: a mathematical library (libm) that aims to provide elementary functions with correct rounding in double precision for the four IEEE-754 rounding modes. It has also been used to develop robust floating-point semi-static filters for CGAL's geometric predicates. Some operators of FLIP, a floating-point library for integer processors, were also proven thanks to Gappa.

Keywords: Floating-point computations, rounding errors, interval arithmetic, formal certification

Full Paper: http://lipforge.ens-lyon.fr/www/gappa/

Robustness and Randomness

Dominique Michelucci (Universite de Bourgogne, F)

Robustness problems of computational geometry algorithms is a topic that has been subject to intensive research efforts from both computer science and mathematics communities. Robustness problems are caused by the lack of precision in computations involving floating-point instead of real numbers. This paper reviews methods dealing with robustness and inaccuracy problems. It discussed approaches based on exact arithmetic, interval arithmetic and probabilistic methods. The paper investigates the possibility to use randomness at certain levels of reasoning to make geometric constructions more robust.

Keywords: Robustness, interval, randomness, inaccuracy, geometric computation

Joint work of: Michelucci, Dominique; Moreau, Jean Michel; Foufou, Sebti

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/716

Pythagore's Dilemma, Symbolic-Numeric Methods, and Geometric Modeling with Semi-Algebraic Curves and Surfaces

Bernard Mourrain (INRIA - Sophia Antipolis, F)

The treatment of shapes on a computer leads to a large domain of investigation, connected to topological, differential, numeric, symbolic questions and with many applications.

We are interested in shapes represented by semi-algebraic models, in particular by parameterized and implicit representations, which allow compact and powerful shape representations. Performing geometric operations on these models requires on one hand topological computation and on the other hand numerical approximation. In particular, this leads to the manipulation of roots of polynomial equations. We are going to detail this duality between exact implicit and approximate computation.

First, we will describe some techniques to deal effectively with real algebraic numbers, how they can be isolated and manipulated in geometric computation, what is the complexity of such operations and some experiments in the library SYNAPS.

Next, we will consider multivariate problems, in particular subdivision methods, and show how that they yield approximate, control or certified computation, depending on the context and application. Their complexity and practical behavior will also be discussed.

A special attention will be given to the problem of computing the topology of curves and surfaces, which might be singular. We will illustrate our approach on planar or three dimensional curves and surfaces, and on some operation like the detection of self-intersecting points or computing silhouette, intersection, or characteristic curves on surfaces.

Keywords: Symbolic-numeric computation, curve, surface, polynomial equation

On Quadrature in Exact Real Arithmetic

Norbert Müller (Universität Trier, D)

The numerical determination of integrals of real functions is still a challenging task, esp. if we consider it from the viewpoint of reliable high-precision (or "exact") real arithmetic, where we are forced to guarantee the accuracy of the results.

In the talk we address the well-known quadrature schemes Newton-Cotes and Gauss-Legendre, which both lead to surprisingly comparable results concerning complexity measured in the accuracy of the computed integrals.

Implementation Of Highly Accurate Complex Inclusion Functions in the CoStLy Library

Markus Neher (Universität Karlsruhe, D)

The practical calculation of highly accurate range bounds for some complex standard functions is addressed in our talk. The functions under consideration are root and power functions, the exponential, trigonometric and hyperbolic functions, and their inverse functions. For such a function f and a given rectangular complex interval **z**, some interval **w** is computed that contains all function values of f in **z**. This is done by expressing the real and the imaginary part of f as compositions of real standard functions and then estimating the ranges of these compositions. In many cases, the inclusions are optimal in exact arithmetic, such that **w** is the smallest rectangular interval containing the range of f.

The algorithms presented in our talk have been implemented in a C++ class library called CoStLy (Complex Standard Functions Library), which is distributed under the conditions of the GNU General Public License. Great care has been taken to eliminate over/underflow exceptions and to avoid cancellation in intermediate expressions, which would impair the accuracy of the range bounds in floating point calculations. Numerical examples are presented to demonstrate that high accuracy in computation has been achieved with our implementation.

Keywords: Complex standard functions, range bounds, interval arithmetic

Floating Point Geometric Algorithms for Topologically Correct Scientific Visualization

Thomas J. Peters (Univ. of Connecticut, USA)

The unresolved subtleties of floating point computations in geometric modeling become considerably more difficult in animations and scientific visualizations. Some emerging solutions based upon topological considerations will be presented. A novel geometric seeding algorithm for Newton's method was used in experiments to determine feasible support for these visualization applications.

Keywords: Geometry, algorithm, visualization

Joint work of: Peters, Thomas J.; Moore, Edward L. F.

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/717

Proposing Interval Arithmetic for the C++ Standard

Sylvain Pion (INRIA - Sophia Antipolis, F)

I will report on a recent effort by Guillaume Melquiond, Hervé Brönnimann and myself to push forward a proposal to include interval arithmetic in the next C++ ISO standard. The goals of the standardization are to produce a unified specification which will serve as many uses of intervals as possible, together with hoping for very efficient implementations, closer to the compilers. I will describe how the standardization process works, explain some of the design choices made, and list some of the other questions arising in the process. We welcome any comment on the proposal.

Keywords: Interval arithmetic, C++, ISO standard

Joint work of: Pion, Sylvain; Brönnimann, Hervé; Melquiond, Guillaume

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/718

Interval Arithmetic with Containment Sets

John D. Pryce (Cranfield University, GB)

The idea of containment sets (csets) is due to Walster and Hansen, and the theory is mainly due to the author. Now that floating point computation with infinities is widely accepted, it is necessary to achieve the same for interval computation. The cset of a function over a set in its domain space is the set of all limits of normal function values over that set. It forms a sound basis for defining a number of practical models for interval arithmetic that handle division by zero and related operations in an intuitive and consistent way. Cset-based systems offer new opportunities for compiler optimization.

The talk presents basic theory, implementation issues, and some numerical examples using a trial Matlab implementation.

Keywords: Intervals, containment sets, exception-free execution

Static Analysis for the Validation of Numerical Software

Sylvie Putot (CEA - Gif sur Yvette, F)

We present a static analysis of the errors introduced by the approximation by floating-point arithmetic of real number computations. The analysis follows the floating-point computation, and bounds at each operation, the error committed between the floating-point and the real result. This error is decomposed in terms coming from the control points (or groups of points) in the source code, in order to provide some information about the provenance of the imprecision in the program.

The natural analysis relying on this idea consists in using intervals to bound the floating-point value and errors for each variable. However, this does not allow to take into account correlations between variables.

We present here a relational analysis using affine interval arithmetic to compute the floating-point value of a variable. But, affine arithmetic uses some properties that are true only on real numbers and not on floating-point numbers. Thus, it can not be used directly for bounding the floating-point value: we add a rounding error term to the estimation of the real number with affine arithmetic. We use in fact three error terms for each variable: the error corresponding to the bounds of the result of a computation in real numbers, and the maximum error for all possible values of this result. We show how these complementary terms can be used to reduce the overapproximation of the error.

Finally, we present some applications in the analysis of safety critical programs.

Joint work of: Putot, Sylvie; Goubault, Eric

Accurate Summation and Dot Product

Siegfried M. Rump (TU Hamburg-Harburg, D)

We present a variety of algorithms for computing sum and dot product of floating-point vectors with 1) specified precision, and 2) with specified accuracy. All algorithms use only standard floating-point operations in one working precision. They are especially fast because no branches are necessary in the inner loop.

With the second type of algorithm a result accurate to the last bit is computed independent of the condition number. Nevertheless this algorithm is some 40 % faster than XBLAS, which gives 'only' a result as if computed in quadruple precision for double as working precision.

Computation Visualized - A Microscopic View on the Results of Sign Computations with Floating-Point Arithmetic

Stefan Schirra (Universität Magdeburg, D)

In this talk we show a number of pictures illustrating shortcomings of (naive) floating point computation with respect to sign computations in geometric predicates, especially the orientation predicate for three points in the plane. The tools use to produce the pictures use lazy arithmetic with expression dags and separation bounds for zero testing in order to identify point configurations that lead to incorrect results when evaluated with floating-point arithmetic. We briefly describe the underlying theory of such number types. Such a tool for visualizing computation problems can be used for error analysis, visual debugging, teaching, and the construction of problematic examples.

Keywords: Computational geometry, exact geometric computation, visualization

Topological Models for Tolerant Modeling

Vadim Shapiro (University of Wisconsin - Madison, USA)

Interpretations of imprecise data and approximate geometric computations require some notion and a model of validity. Unfortunately, most formal models adopted in computer-aided design presuppose ability to compute exactly. For example, a solid model is defined to be a closed regular bounded and semi-analytic subset of Euclidean space. In this talk, I will consider possible formulations of valid solids and their representations in a presence of errors in geometric data and/or algorithms.

Quality Assurance in Developing an Interval Subroutine Library

Spencer Smith (McMaster University, CDN)

This talk motivates the use of software engineering methodologies to improve the quality of an Interval Subroutine Library (ISL). The qualities of interest include reliability, performance, usability, maintainability and reusability. The software engineering methodologies that are proposed for ISL have been successfully applied to improve the quality of safety critical systems and business systems. The methodologies include an iterative waterfall software development process model, a commonality analysis, a requirements analysis, a module interface specification and verification and validation techniques. In addition, the talk will mention ideas to assist with implementation, such as configuration management software.

Keywords: Interval subroutine library, software engineering

Joint work of: Smith, Spencer; Corliss, George; Kearfott, Baker; Nedialkov, Ned; Pryce, John

Transfinite Interpolation for Well-Definition in Error Analysis in Solid Modelling

Neil Stewart (Université de Montréal, CDN)

An overall approach to the problem of error analysis in the context of solid modelling, analogous to the standard forward/backward error analysis of Numerical Analysis, was described in a recent paper by Hoffmann and Stewart. An important subproblem within this overall approach is the well-definition of the sets specified by inconsistent data. These inconsistencies may come from the use of finite-precision real-number arithmetic, from the use of low-degree curves to approximate boundaries, or from terminating an infinite convergent (subdivision) process after only a finite number of steps.

An earlier paper, by Andersson and the present authors, showed how to resolve this problem of well-definition, in the context of standard trimmed-NURBS representations, by using the Whitney Extension Theorem. In this paper we will show how an analogous approach can be used in the context of trimmed surfaces based on combined-subdivision representations, such as those proposed by Litke, Levin and Schröder.

A further component of the problem of well-definition is ensuring that adjacent patches in a representation do not have extraneous intersections. (Here, "extraneous intersections" refers to intersections, between two patches forming part of the boundary, other than prescribed intersections along a common edge or at a common vertex.) The paper also describes the derivation of a bound for normal vectors that can be used for this purpose. This bound is relevant both in the case of trimmed-NURBS representations, and in the case of combined subdivision with trimming.

Keywords: Forward/backward error analysis, robustness, well-definition, trimmed NURBS, combined subdivision, trimming, bounds on normals

Joint work of: Stewart, Neil; Zidani, Malika

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/719

Interval Analysis and Computation Without Intervals

Paul Taylor (Manchester University, GB)

Abstract Stone Duality (ASD) is a new theory of general topology that axiomatises it directly in terms of its natural concepts (continuous functions, open subspaces, etc) — completely eliminating set theory. In particular, Dedekind completeness and the Heine–Borel property are as much part of the syntactic calculus for the real line as is its arithmetic. Some aspects of real analysis have already been developed using it.

I will show how this calculus (of "single" points) can be translated formally into Interval Analysis, interpreting the arithmetic operations a la Moore and compactness as optimisation under constraints. Notice that interval computation is the conclusion and not the starting point. In future this will provide the framework in which to develop interval analogues of ideas in, for example, differential calculus — in a conceptual way, rather than by guesswork.

The ASD calculus is not a programming language, and in particular there is a major challenge to devise an order of execution. As in PROLOG, a logical term is considered as the program that searches for its proof and witnesses. However, it may be possible to confine the non-determinism and back-tracking to specific processes that are similar to those already used in exact real arithmetic packages such as iRRAM. These processes suggest possible extensions of these packages. More complex situations resolve into optimisation problems subject to non-strict polynomial constraints.

My conclusion is therefore an open-ended question to the respective experts: how much of this language could be implemented?

Keywords: ASD

Full Paper: http://www.cs.man.ac.uk/∼pt/ASD

Towards Expression Defined Accuracy

Jürgen Wolff von Gudenberg (Universität Würzburg, D)

In numerical computations the accuracy of the result quite often depends on a few expressions. In the talk we show how evaluation semantics of expressions can be controlled in C++ using expression templates. The design of an appropriate library is discussed.

Keywords: C++, expression templates

Theory of Real Computation according to EGC

Chee K. Yap (Courant Institute - New York, USA)

The theory of discrete computation has a widely accepted basis rooted in Turing's model of computation or other equivalent models.

In contrast, the theory of real computation has turned out to be considerably more subtle, with debates over its very foundation after 50 years of investigation. The two dominant views here might be called the Analytic Approach (a.k.a. computable analysis) and the Algebraic Approach (a.k.a. BSS model, real RAM model).

In this talk, we discuss another view arising from our work in Exact Geometric Computaton (EGC). Here, the Zero Problem is central, and it informs us that a proper theory must allow for a range of complexity for the zero problem. Consequently, unlike the analytic or algebraic schools, we disavow the introduction of all real numbers into the model, but focus on the approximation of real numbers and their functions (similar to the analytic approach).

Others have tried to reconcile the algebraic and the analytic models (e.g., Braverman). We prefer to leave each theory in its own domain, but seek instead transfer theorems to relate them.

Keywords: Theory of real computation, BSS model, computable analysis, exact geometric computation

Fast Evaluation of Sine and Cosine for Real Argument

Paul Zimmermann (INRIA Lorraine, F)

We present an extension to the sine and cosine functions of Brent's algorithm for evaluating $\exp(x)$. The algorithm uses the binary splitting method and works in $O(M(n)\log(n)^2)$ for n-bit output. It computes simultaneously $\sin(x)$ and $\cos(x)$. We also give computer experiments within the MPFR library.

Worst Cases for the Exponential Function in the IEEE 754r decimal64 Format

Paul Zimmermann (INRIA Lorraine, F)

We searched for the worst cases for correct rounding of the exponential function in the IEEE 754r decimal64 format, and computed all the bad cases whose distance from a breakpoint (for all rounding modes) is less than 10^{-15} ulp, and we give the worst ones. In particular, the worst case for $|x| \geq 3 \times 10^{-11}$ is $\exp(9.407822313572878 \times 10^{-2}) = 1.098645682066338\,5\,0000000000000000\,278\ldots$. This work can be extended to other elementary functions in the decimal64 format and allows the design of reasonably fast routines that will evaluate these functions with correct rounding, at least in some domains.

Keywords: Floating-point arithmetic, decimal arithmetic, table maker's dilemma, correct rounding, elementary functions

Joint work of: Lefèvre, Vincent; Stehlé, Damien; Zimmermann, Paul

Full Paper: http://drops.dagstuhl.de/opus/volltexte/2006/748

Computations with Effective Real Numbers

Joris van der Hoeven (Université Paris Sud, F)

In our talk we will present a C++ library for the computation with effective real numbers (i.e. numbers x which come with an algorithm that takes ϵ on input and compute a dyadic approximation x' with $|x' - x| < \epsilon$) included in Mmxlib.

This library consists of three parts: MPFR multiprecision arithmetic, an extension for interval/ball arithmetic, and the high-level real number interface. We will present several strategies for computing with intervals and effective real numbers and analyze them from a complexity point of view. If time permits it, we will also present work in progress to extend the Mmxlib library with a class for "effective analytic functions".

Keywords: Effecive real numbers, certified computations, interval analysis, effective analytic functions