RESEARCH STATEMENT

DAVID BINDEL

http://www.cims.nyu.edu/~dbindel

Introduction

Along with theory and experiment, computation is crucial to modern science and engineering. We can simulate models of problems for which hand analysis is intractable and experiments are infeasible. Such computational modeling is collaborative and interdisciplinary: to ask sensible questions and include relevant physics requires application expertise; to design and analyze solution algorithms requires mathematical expertise; and to build software to perform the computation requires expertise in software engineering. To analyze and validate computer experiments requires a combination of application knowledge, mathematical insight, and computer science. I have contributed at each of these stages.

As an instructor of mathematics at NYU and as a computer science graduate student at UC Berkeley, I have built and analyzed computational models of a wide variety of systems. Using these models, my collaborators and I hope to create smaller, cheaper cell phones with longer batter lifetimes; to help doctors understand patient bone strength from CT scan data; and to find and understand failure modes in computer networks. I have used mathematical and physical insight to create fast, accurate methods to analyze these models, and I have written new simulation software incorporating my methods. I have modeled several electromechanical devices, and my analyses helped engineers designing new devices. I also discovered a previously unsuspected physical effect which significantly changes the performance of some of these devices. This work combines my interests in physical modeling, computational mechanics, numerical analysis, and software engineering.

I describe my simulations of Micro-Electro-Mechanical Systems (MEMS) in Section 1 and my work on peer-to-peer networks in Section 2. In Section 3, I describe my work on simulation software engineering. In Section 4, I describe other contributions to numerical analysis, including numerical analysis of nonlinear eigenvalue problems, fast algorithms for finding polynomial roots and subspace continuation methods relevant to locating bifurcations of nonlinear PDEs. In Section 5, I outline my research plans.

1. MEMS simulation

I have worked extensively on computer-aided design (CAD) tools for Micro-Electro-Mechanical Systems (MEMS), micrometer-scale devices used in everything from digital projectors to car engines. MEMS and integrated circuits (ICs) are built using similar processes; and like integrated circuits, MEMS are used to make devices ever smaller, cheaper, and lower power. The market for MEMS is roughly 5-10 billion dollars per year: dominant applications include inertial sensors (accelerometers and gyroscopes), microfluidic devices (ink jet printers and biomedical devices), optical devices (optical data switches and digital projectors), and radio-frequency components (used in cell phones and in RFID tags for inventory tracking).

The MEMS industry inherited the IC industry's fabrication technology; but simulation technology for MEMS has lagged behind, so that much MEMS design still follows a "build-and-break" model. IC designers have CAD tools for all levels of design, from high-level hardware description languages like VHDL to analog circuit design tools like SPICE to the finite and boundary element tools used to compute parasitic capacitances and inductances. These tools support hierarchical design: simple components such as transistors combine to form logic gates, arithmetic units, and eventually computer chips. In contrast, only recently have MEMS engineers had commercially available tools for system-level design.

I am particularly interested in high-frequency resonant MEMS, which can be used as mechanical signal-processing elements. Cellular radios already use mechanical filters to process signals, since purely electrical filters have too much damping to be used at high MHz-GHz frequencies. In modern radios, these mechanical filters use waves traveling along a quartz or ceramic surface, or through the bulk of a piezoelectric film. The filter is separate from the signal processing circuitry, and a substantial fraction of the power budget for a cell phone goes into getting a signal through the mechanical filter and across a wire into the circuits. We have been working on mechanical signal processing elements which can be fabricated together with integrated circuits, so that an entire cellular radio might some day fit on one chip [51].

To build a radio system using RF MEMS, engineers need to know how each component behaves and how the components are coupled. To describe a component, we need at least the resonant frequencies and associated damping factors, but current finite element codes compute only resonant frequencies, and cannot compute such dominant damping effects as anchor loss and thermoelastic damping. Furthermore, current tools fail to predict the behavior of fully electromechanically-coupled resonator systems. I address such problems with current tools at several levels:

- By building system simulators that are fast enough for the early stages of MEMS design, and finite element tools that are useful in detailed design;
- By modeling damping mechanisms which matter for RF resonator design, and building fast algorithms and model reduction tools to study these damped systems; and
- By using my tools to model real devices, and validating results against experiments.

I describe my contributions in more detail below.

1.1. **Damping models for RF MEMS.** Two recognized damping mechanisms in high frequency MEMS are radiation of elastic waves through an anchor into the substrate (anchor loss), and heat diffusion across temperature gradients created by expansion and contraction of devices (thermoelastic damping). I built finite element models for both types of losses, and built fast algorithms to find the damped resonances in each case.

Anchor losses and perfectly matched layers: Compared to the acoustic wavelengths of RF MEMS vibrations, a silicon chip is typically very large. Detailed models of the entire chip volume are impractical, so we use transparent boundary conditions that mimic the behavior of an effectively infinite domain with a few auxiliary variables. In particular, we use a *perfectly matched layer* (PML) absorbing boundary. Perfectly matched layers were first introduced by Bérenger for problems in electromagnetic wave propagation [9], and were subsequently reinterpreted as a complex-valued change of coordinates applicable to any linear wave equation [38, 40, 57, 58]. In [8], Basu and Chopra showed how to implement perfectly matched layers for linear elasticity in a standard displacement-based finite element framework. I extended this work by using the notion of a coordinate transformation in the weak form to yield an even simpler finite element implementation which involves *no* modifications to the standard shape function routines, and which works in a simple way with axisymmetric, plane stress, plane strain, and three-dimensional elements. At the same time, I showed how the elastic PML can be re-interpreted in the context of ordinary elasticity by changing the material properties in an anisotropic, inhomogeneous way; this interpretation was previously known for scalar wave equations, but not for elasticity. The complex symmetry of the PML matrices was known and exploited in [5] to build a modified Jacobi-Davidson iteration with improved convergence behavior for PML eigenproblems; building on that approach, together with standard Krylov-subspace model reduction ideas [6], I derived a model reduction method which similarly uses the variational structure of the complex-symmetric eigenvalue problem to obtain better approximate eigenvalues and better reduced-order models from a given subspace [14, 19]. My algorithms obtain about twice as many correct digits as are obtained by standard model reduction methods using the same amount of work.

Thermoelastic damping analysis: Thermoelastic damping (TED) is mechanical damping from the differential heating and cooling generated by volumetric strain, which by irreversible heat diffusion leads to loss of coherent mechanical energy in resonators. Though MEMS designers have long believed that TED is a significant source of damping [29, 54], most published work refers to an analytical approximation developed by Zener which is valid only for long, thin beams [64, 65, 66]. With Sanjay Govindjee and Tsuyoshi Koyama, I wrote a finite element to solve the full coupled equations of thermoelasticity [47]. Based on scaling arguments, I also devised a fast perturbation approach to computing damped mechanical modes of the TED equation [17].

1.2. **Analysis of RF resonator models.** Beside building the numerical algorithms and simulation tools I described above, I also used these tools to model new high-frequency resonant MEMS as they were developed at Berkeley.

Shear ring resonators: I ran simulations that informed the design of a family of ring resonators proposed by Sunil Bhave [10]. These resonators are driven in a shearing motion: the outside rim rotates in one direction, while the inside rim rotates in the opposite direction. I analyzed the shapes and resonant frequencies of shearing modes for an ideal ring, and I built a more detailed parameterized finite element model of the device, which I used in parameter studies to determine the effects of different strategies for anchoring the resonator to the substrate and for driving and sensing the motion of the resonator.

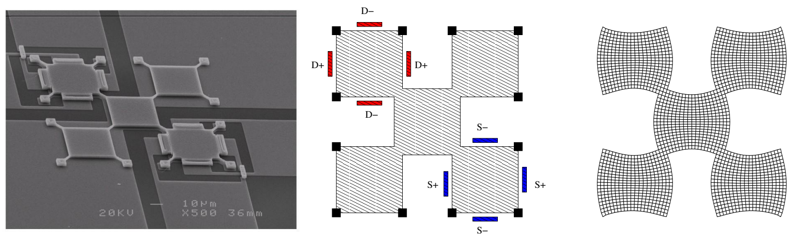

FIGURE 1. Illustration of a checkerboard resonator. The SEM picture (left) shows an actual device, while the simulation (right) shows one resonant mode excited during operation. The motion is capacitively excited at the northwest corner and sensed at the southeast corner (center). Because the system resists motions that are not close to a resonant frequency, the device acts as a bandpass filter. My simulation tools were used to design this device; the simulations were subsequently validated against experimental measurements.

Checkerboard filters: Again in collaboration with Sunil Bhave, I built finite element models, first in FEAPMEX and then in HiQLab (see Section 3), of a family of novel checkerboard-shaped electromechanical filters [11] (Figure 1). The device consists of a checkerboard of squares, which are linked at the corners. Using model reduction techniques, I was able to reduce the size of the model from 3231 variables (for a coarse mesh) to 150 variables. The reduced order model produced results which were visually indistinguishable from the full model; and because the results could be recomputed for each frequency within milliseconds, I was able to write a tool so that we could interactively scan through interesting frequencies to see a visualization of the motion of the system at each frequency [19].

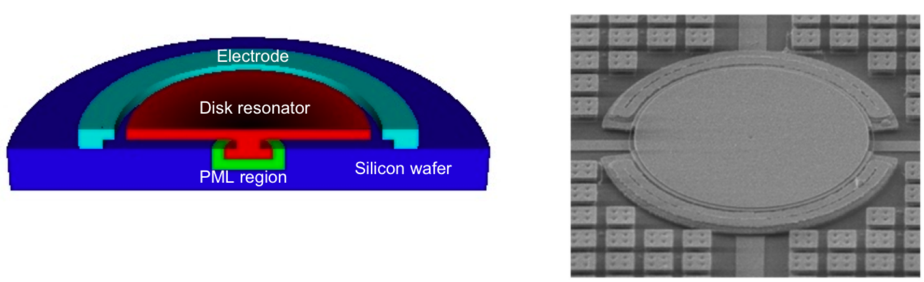

FIGURE 2. Cutaway schematic (left) and SEM picture (right) of a micromachined disk resonator. Through electrostatic attraction between the disk and a surrounding ring of electrodes, the disk is driven into mechanical resonance. Because the disk is anchored to a silicon wafer, energy leaks from the disk into the substrate, where it radiates away as elastic waves. No commercial tools are able to estimate this anchor loss, which is the dominant source of damping in these devices.

Disk resonators: I analyzed the behavior of several simple MEMS disk resonators [15, 61, 62] using the perfectly matched layer models in HiQLab (Figure 2). To our knowledge, this is the first time a detailed finite element model of the anchor damping of these devices has been constructed. My simulations showed a remarkable sensitivity of the amount of damping as a function of the thickness of the disks, which I was able to explain as the result of the interaction between bending and radial-extension modes at nearly the same frequency. This previously unexpected phenomenon, which I first predicted through computer experiments, was subsequently verified in experimental measurements performed by Emmanuel Quévy [14, 15].

2. Analysis of peer-to-peer networks

Peer-to-peer systems have become a major topic in computer networking, and peer-to-peer applications such as BitTorrent and its competitors are widely used. The same open network architecture that makes these applications possible also makes them subject to failures from malicious attacks or natural network problems. These failures are hard to diagnose and reproduce, so that computational models of the network behavior are necessary. I am involved in analyzing such models.

Security: OceanStore is a wide-scale storage system project at Berkeley. I was one of the graduate students who worked with John Kubiatowicz on the original OceanStore paper [48]; my contributions mainly involved system security. In particular, Yan Chen, Adam Bargteil, and I ran computational experiments to quantify how different network organizations might contribute to making distributed denial-of-service attacks more or less successful [33].

Network path monitoring: Overlay networks allow distributed network applications to quickly detect and route around slow or failing network links. In order to route around problems, though, the hosts in an overlay network must first find which paths between them have problems. In collaboration with Yan Chen and Randy Katz, I have studied linear algebraic approaches to computing estimates of path loss rates and latencies in overlay networks [34, 35, 36, 37]. Our techniques depend only on end-to-end measurements of routing paths, and so require no cooperation from routers inside the network. Moreover, our model is *deterministic*, and requires no statistical assumptions about the distribution of lossy links or other such properties. We represent the system topology through a highly-structured sparse matrix which relates link properties (such as link loss rates) to path properties. Because of the hierarchical structure of large scale networks, this matrix has low rank; consequently, we can typically predict the behavior of all $O(n^2)$ paths through an overlay network of n nodes based on roughly $O(n \log n)$ measurements. Moreover, I have worked on fast algorithms for updating predictions when the network routing behavior changes dynamically, and when nodes are allowed to leave or enter the overlay network.

In recent work with Jiexun Xu, an undergraduate researcher at NYU, I have been studying fast algorithms to compute specially structured outer product factorizations of the network path matrix used in our earlier work. These factorizations are based on our understanding of the ways in which routing paths typically overlap in real networks. Unlike the factorizations we used in earlier work, these factorizations maintain sparsity, and they are an order of magnitude faster than our earlier algorithms (about a minute to factor the path matrix for a model network of 300 nodes, as opposed to 45 minutes). We also hope this work will lead to distributed versions of our earlier initialization and inference algorithms.

Tomography and fault diagnosis: In more recent work with Yan Chen, we have turned to the problem of *locating* failing links within a network based on end-to-end overlay measurements. This location task is complicated because it may be impossible to compute the loss rates of individual links from end-to-end measurements. The Deterministic Overlay Diagnosis system (DOD) [67, 68] and Least-biased End-to-end Network Diagnosis (LEND) system [69] find the shortest sequences of links within the network, or "virtual links," with properties which can be determined from end-to-end measurements. Using these virtual link properties, we are able to localize problems to within a small number of links. To our knowledge, this is the first deterministic network diagnosis system based solely on end-to-end measurements.

Multi-radio mesh network protocols: Wireless mesh networks forward traffic from node to node over radio links. Such networks can achieve much higher throughput when there are multiple radios at each node rather than a single radio. With Jinyang Li and Lakshmi Subramanian, I have been looking at joint routing and channel assignment protocols for wireless mesh networks optimized under the assumption that most traffic goes to a few gateway nodes.

3. Simulation software engineering

I have written or rewritten four simulation codes: SUGAR, a system-level simulator for MEMS; MATFEAP and FEAPMEX, MATLAB interfaces to the FEAP finite element code; HiQLab, a new finite element code for simulating damping in radio-frequency MEMS; and BoneFEA, a new finite element code for simulating failure in bones. These codes are written in a combination of C/C++, Fortran 77, MATLAB, and other languages; not only are they useful in their own right, but they also illustrate useful patterns for designing interfaces between codes which are not always designed to work together. In addition, I wrote several tools and libraries that have proven useful in their own right.

SUGAR: SUGAR is a system-level MEMS simulator whose name and philosophy come from the influential SPICE family of circuit simulators. Using a high level, hierarchical language, designers describe MEMS to SUGAR in terms of primitive circuit and structural components, such as capacitors, resistors, beams, and anchors. SUGAR was created by Kris Pister, Jason Clark, and Ningning Zhou; I joined the project to help make the code faster and more numerically robust. I rewrote the parser and graphics subsystem, improving the overall performance four-fold for a simple gyroscope model; with Zhaojun Bai, I added model reduction to make transfer function calculations orders of magnitude faster for large devices; and with Patrick Eaton, I devised M&MEMS, a SUGAR-based web service hosted on the Millennium computer cluster [7].

For version 2, I rewrote SUGAR in a combination of MATLAB and C. Six months later, I rewrote SUGAR again for version 3. The core of SUGAR 3 is an object-oriented C library (about 11000 lines) with support code and interfaces written in MATLAB (4800 lines) and Lua (350 lines) and supporting utilities written in C (1800 lines). SUGAR 3 features an input language based on the Lua scripting language, and is written in a literate programming style which combines documentation and source code. SUGAR 3 is faster than earlier versions, and computations which took minutes in SUGAR 1 take only seconds in SUGAR 3.

SUGAR has been used in introductory MEMS classes at Berkeley, Cornell, and Johns Hopkins University; the M&MEMS service was used for Berkeley's MEMS course. I gave tutorials on SUGAR to students at Berkeley, to visiting industrial members at the Industrial Advisory Board meeting for the Berkeley Sensor and Actuator Center (BSAC), and to attendees of the 2001 International Conference on Modeling and Simulation of Microsystems. Ideas from SUGAR were used heavily in the development of the SYNPLE, a similar system-level simulation code in the commercial Intellisuite package [3]. SUGAR was also used by Berkeley researchers supervised by Alice Agogino and Carlo Sequin in their work on MEMS design synthesis and optimization [1].

FEAP, FEAPMEX, and MATFEAP: While SUGAR is useful for constructing models of systems, it is not appropriate for constructing continuum-level models of individual devices. In order to perform parameter studies of models of RF MEMS devices, I wrote FEAPMEX, a MATLAB interface to the FEAP academic finite element code [56]. FEAPMEX consists of about 5000 lines of C, Fortran, and MATLAB, not including automatically generated code; there are also about 2000 lines of example scripts included with the program. FEAPMEX was useful for my own work (see Section 1.2), but it proved to have broader appeal as well. The FEAPMEX page has been visited 4149 times since February 2005, and the code has been used for applications in instrument modeling [27], stochastic structural analysis [2], ultrasonic nondestructive evaluation [4], and material parameter identification and inverse problems [59].

MATFEAP, a successor to FEAPMEX, consists of about 2800 lines of C and FORTRAN and 800 lines of MATLAB. Though still in a pre-release version, MATFEAP has been used by Arka Prabha and Panos Papadopoulos at Berkeley for scripting simulations of textured materials.

Beside writing FEAPMEX, I have, with Sanjay Govindjee and Robert Taylor, written several extensions to FEAP, including the memory management subsystem which became a standard feature starting with FEAP 7.0. In spring 2006, I also worked with them on a parallel version of FEAP.

HiQLab: HiQLab is a new finite element code that I wrote to model high-frequency MEMS resonators [12, 47]. By writing a new code, I was able to add special support for my new elements and solvers (see Section 1.1), and to combine features I liked from other simulators: a highly flexible description language and object-oriented architecture like those in SUGAR, and mature finite element data structures like those in FEAP. HiQLab consists of a core library written in C++ (14000 lines), automatically-generated interfaces to MATLAB and Lua (7000 lines of high-level source), and MATLAB solvers and visualization routines (1500 lines). The interface code generators are written with lex and yacc, and the standard numerical libraries I use are written in Fortran and C. HiQLab can be used from MATLAB or using a Lua interface; the latter user interface has fewer features, but also has less overhead.

HiQLab was first publicly released at the end of 2004; since February 2005, the HiQLab page has been visited 2900 times. The code has been used by device designers, and in published work from Cornell [52], and from the VTT Technical Research Centre in Finland [46]. While I was a graduate student, it was used by our local research group as a development platform for problems in design optimization and model reduction. We also collaborated with Roger Howe to use HiQLab to model new types of RF MEMS. As an early result of this collaboration, we were able to predict a previously unknown mode-interference phenomenon which causes substantial variation in the quality of certain high-frequency resonators (see Section 1.2).

BoneFEA: BoneFEA is a finite element tool for simulating the deformation and failure of bones, written together with Panos Papadopoulos for ON Diagnostics. From finite element models derived from CT scan data, BoneFEA predicts how a patient's bones might fail under loading conditions. BoneFEA serves as a replacement for the ABAQUS software previously used by ON Diagnostics, but with application-specific solvers. I contributed to this project in writing the software itself (BoneFEA consists of about 12000 lines of C++ and 1400 lines of support scripts), and also in designing the geometric multigrid solver used in the system.

Support tools: In the most recent versions of HiQLab and BoneFEA, I make extensive use of two translators that I wrote. `MWrap` is a translation tool that generates MEX files from augmented MATLAB scripts; it consists of about 3000 lines of C++ and lex/yacc files. `matexpr` is a source-to-source translator that generates optimized C code from a MATLAB-like matrix expression language embedded in C; it consists of about 2000 lines of C++ and lex/yacc. `MWrap` has been used by researchers at Columbia and is being used in a semiconductor manufacturing process optimization project at IBM.

CLAPACK: I generated CLAPACK 3.0, a C translation of the widely-used LAPACK linear algebra library. CLAPACK 3.0 was released in September 2000; and between August 2001 and November 2005 there were over 2.2 million accesses to the CLAPACK pages.

4. Numerical analysis

A main goal of numerical analysis is to create fast, robust algorithms for problems of continuous mathematics. Part of this activity is the synthesis of algorithms and analysis of their performance; part is about the mathematical analysis of numerical approximations. These pieces mutually reinforce each other: new algorithms require new error analyses, and new approximation results suggest new algorithms. Exploring from a base of specific problems and areas of curiosity, I both write general-purpose software and develop the corresponding approximation theory. Though much of the numerical analysis I do is directed toward computational models of MEMS (Section 1) and of computer networks (Section 2), I have also studied fundamental questions about the structure of linear and nonlinear eigenvalue problems and the mechanics of floating point arithmetic.

Nonlinear eigenvalue problems and resonance computations: Nonlinear eigenvalue problems occur naturally in the frequency-domain analysis of damped linear vibrations, such as the ones that arose in my work on resonant MEMS. Similar problems arise in photonic crystal and fiber optic cable design, in the design of particle accelerators, and in various other problems involving transmission behavior and resonance for acoustic, elastic, electromagnetic, and quantum mechanical scattering.

Nonlinear eigenvalue problems also arise naturally when computing the point spectrum and resonance poles (poles of a meromorphic continuation of the resolvent across the continuous spectrum) for linear operators on unbounded domains, such as the Schrödinger operators of quantum mechanics. In joint work with Maciej Zworski, I developed `Matscat`, a code to compute resonances for Schrödinger operators with compactly supported potentials on the real line. Using this code, we discovered a previously unknown asymptotic symmetry between the energies of bound states for such Schrödinger problems and the anti-bound resonant states [26].

Together with Iva Vukicevic, a summer undergraduate researcher at NYU, I studied a related nonlinear eigenvalue problem that occurs in analyzing the stability of stationary solutions and traveling-wave solutions to nonlinear differential equations on unbounded domains. In particular, we analyzed the error in numerical methods for computing kink soliton solutions and linearized spectra for a discrete sine-Gordon model of coupled discrete nonlinear pendula [25, 60].

A central idea in my work on nonlinear eigenvalue problems has been to extend perturbation results and spectral inclusion bounds from the linear case to the case of analytic operator-valued functions of a single complex variable [18]. The MEMS damping problems, Schrödinger resonance problems, and wave stability problems I have considered all share this structure, and one of the ultimate goals of this work is to give a rigorous error analysis of the eigenvalue computations for each of these problems. I have given recent talks on this work at the 2008 Householder meeting and at a meeting in Banff on the mathematical theory of resonances.

Continuation of invariant subspaces: Parameter-dependent matrices occur naturally in engineering parametric design studies like those described in Section 1.2, and also in the study of dynamical systems, where they provide important information about the behavior of branches of dynamic equilibria. The Continuation of Invariant Subspace (CIS) algorithm was introduced in [41], where it was used in computing connecting orbits. The idea of the algorithm is to track a smoothly-varying invariant subspace of a parameterized matrix using a predictor-corrector iteration. In joint work with Mark Friedman, I have made both theoretical and practical contributions to further development of the CIS algorithm and related schemes for tracking eigenvalues of parameterized matrices [21, 22]. First, by extending and refining constructive proofs of the existence of continuous invariant subspaces [55], I proved a theorem that amounts to a posterior test to ensure that invariant subspaces computed at two parameter values of a parameterized matrix are samples of a single continuously-defined parameterized invariant subspace. Such theory is useful to ensure the robustness of subspace continuation algorithms. I also developed the first sparse algorithms for predicting, adapting, and normalizing invariant subspaces [13]. Using sparsity, we are able to solve problems an order of magnitude larger than those solved by dense methods. I co-developed code to implement these algorithms, and to integrate them into Matcont, a bifurcation analysis tool [22, 42]. In addition, I have developed both code and theory to exploit variational structure – both for symmetric and nonsymmetric problems – in order to obtain high-accuracy predictions and reliable step control for eigenvalue continuation. I have presented this work in talks and proceedings [13, 23]. Recently, I have also been working on analysis which provides a practical test of when a given invariant subspace contains all the information needed to diagnose instability or near-instability.

Rank-symmetric structure and companion matrix eigenvalues: In joint work with Ming Gu and Shivkumar Chandrasekharan, I have shown how to convert the standard workhorse algorithm for computing all the roots of a polynomial – the QR iteration applied to a companion matrix [50] – from an $O(n^3)$ algorithm to an $O(n^2)$ algorithm [20]. Our modification works for any Hessenberg eigenvalue problem coming from a matrix which is a low rank modification of a symmetric, skew-symmetric, or orthogonal matrix, and exploits the semi-separable structure [31] present in related matrices. Matrices in this class include companion and block companion matrices, arrow matrices, matrices corresponding to vibrations with localized damping, and matrices from the discretization of PDEs which fail to be self-adjoint only at a boundary. I have proven theorems which describe such matrices in terms of eigenvalue structures, which leads directly to perturbation theorems that quantify what it means to be "close" to a low-rank modification of an orthogonal, symmetric, or skew matrix. This theory is critical to understanding the numerical error analysis of this class of algorithms. I have also described a limited-memory variant of Arnoldi for this class of matrices which generalizes the Lanczos [53] and isometric Arnoldi [28] algorithms. I have addressed practical issues of balancing, showing how infinity-norm balancing [32] can be applied to the companion matrix in order to make the eigencomputation more accurate while retaining the required structure. In addition, I have constructed a practical implementation of the algorithm for the specific problem of polynomial root-finding.

Floating point computation: Through my interactions with William Kahan, I have become interested in the mechanics of floating point computation. I have contributed both through work done on numerical analysis of the behavior of Givens rotations in floating point [24], and through two years of service as the secretary of the committee for revision of the IEEE 754 floating point standard. My annotated bibliography on computer support for floating point computation [16] is widely cited in online encyclopedias (e.g. [63]). The standard was accepted in August 2008, and my work as part of the 754 revision committee was recognized in October 2008 by a certificate of recognition from the IEEE Microprocessor Standards Committee.

5. Research plans

Beside continued collaboration with scientists and engineers working on applications, my plans include:

RF MEMS simulation: I plan to extend HiQLab to include more physical effects relevant to resonant MEMS designers. In particular, I am interested in the deliberate use of *nonlinear* effects in RF MEMS design. We have already done preliminary work to study how quasi-static nonlinear loading can be used to tune linear oscillations in RF MEMS, and to see if such tuning can compensate for fabrication variations.

Multiscale finite element simulation for textured materials: Materials with texture often have two relevant physical scales: a macroscopic scale at which we would like to simulate, and a microscopic scale associated with the texture. For example, in a metal sample, the texture is at the scale of the metal grains. A two-level finite element simulation uses fine-scale simulations to evaluate the stress-strain relationship at quadrature points in a coarse-scale macro-model. I have recently started collaborating with Panos Papadopoulos and Robert Taylor on parallel software for such two-scale simulations. Because the fine-scale calculations are so time-consuming, I plan to study model reduction techniques to accelerate them.

Fast solvers for rank-structured matrices: Matrices with low-rank blocks occur naturally in discretizations of integral equations, in pseudospectral discretizations of PDEs, and in Schur complements that occur in the course of Gaussian elimination of finite element matrices. This structure is the basis for methods like the fast multipole method [44] and panel clustering methods [45], which are used to accelerate iterative linear solvers by accelerating multiplication by these matrices. More recent work has focused on fast algorithms for directly solving linear systems with such rank-structured matrices [30, 43, 49]. One approach to deriving such algorithms involves generating an extended sparse linear system in which the original rank-structured matrix appears as a Schur complement [30] . This approach appears to be more generally applicable than is recognized; for example, matrices in sequentially semi-separable form [31] may be written as a Schur complement in a slightly extended matrix with band structure. This extended system approach is useful because the extended systems can be solved with conventional sparse and banded matrix factorization routines. I plan to explore this connection further, with the ultimate goal of devising fast direct solver software for the inhomogeneous elasticity problems that occur in my models of bones and MEMS.

Fast solvers for distributed network tomography: In the network tomography work with Yan Chen, we used dense direct factorization algorithms to compute path loss rates from a minimal number of end-to-end network measurements. I have recently worked on sparse factorization algorithms that use the structure of the computer network. These algorithms are necessary because standard sparse LU and QR factorizations suffer tremendous fill-in on these matrices, so that for even modest size networks they become more expensive than dense methods. I believe that my new algorithms provide an alternative which is attractive not only because it is faster, but also because it potentially provides the basis for a fully distributed algorithm. I also believe we may be able to combine our methods with the "network kriging" approach proposed by Chua, Kolaczyk, and Crovella, in which our scheme is extended by observing that the matrices relating link properties to path properties can be approximated by low-rank matrices [39].

In addition, I plan to study methods to automatically detect and adapt our network tomography schemes when there are routing changes in the network.

Analysis of resonances and nonlinear eigenvalue problems: Nonlinear eigenvalue problems occur in the spectral analysis of damped linear vibrations, and also in methods to compute eigenvalues and resonances for operators on noncompact domains. I am working on using nonlinear eigenvalue analysis for stability analysis of kink and traveling wave soliton solutions to nonlinear PDEs, as well as for calculation of resonances for Schrödinger operators with compactly supported potentials in one and two dimensions. I am also interested in extending these techniques to compute resonances for the Laplacian on non-compact hyperbolic surfaces; such resonances convey geometrically meaningful information, but they have only been computed for a small number of special cases.

As part of my work on nonlinear eigenvalue problems, I want error estimates and provable bounds to check that my codes accurately compute all eigenvalues in some region of the complex plane. Currently, resonance and point spectrum calculations for operators on noncompact domains are often posed in terms of determinantal functions (Evans functions in the traveling wave case and zeta functions in the resonance case). These play a role analogous to the characteristic polynomial of a matrix in the finite-dimensional case. In the finite-dimensional case, eigenvalue algorithms that directly manipulate the characteristic polynomial are typically much less stable than algorithms that work with the full matrix, and I want to apply my error analysis to see if something similar holds in the more general context of computing point spectra and resonances for operators.

References

[1] BEST lab MEMS synthesis design project page, November 2005. http://best.me.berkeley.edu/research/mems/info.php.

[2] Sondipon Adhikari, November 2005. Personal communication.

[3] Sandeep Akkajuru, November 2005. Personal communication.

[4] John Aldrin, November 2005. Personal communication.

[5] P. Arbenz and M. E. Hochstenbach. A Jacobi-Davidson method for solving complex symmetric eigenvalue problems. *SIAM J. Sci. Comp.*, 25:1655–1673, 2004.

[6] Z. Bai. Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Applied Numerical Mathematics*, 43:9–44, 2002.

[7] Z. Bai, D. Bindel, J. Clark, N. Zhou, J. Demmel, and K. Pister. New numerical techniques and tools in SUGAR for 3D MEMS simulation. In *Tech. Proc. of the 4th Intern. Conf. on Modeling and Simulation of Microsystems*, Hilton Head Island, SC, March 2001.

[8] U. Basu and A. Chopra. Perfectly matched layers for time-harmonic elastodynamics of unbounded domains: theory and finite-element implementation. *Computer Methods in Applied Mechanics and Engineering*, 192:1337–1375, 2003.

[9] J.-P. Bérenger. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of Computational Physics*, 114:185–200, 1994.

[10] S. Bhave. *Electrostatic Transduction for MEMS Resonators*. PhD thesis, University of California at Berkeley, September 2004.

[11] Sunil A. Bhave, Di Gao, Roya Maboudian, and Roger T. Howe. Fully differential poly-SiC Lamé mode resonator and checkerboard filter. In *Proceedings of MEMS 05*, Miami, FL, January 2005.

[12] D. Bindel. HiQLab project page.

[13] D. Bindel, J. Demmel, and M. Friedman. Continuation of invariant subspaces for large bifurcation problems. In *Proceedings of the SIAM Conference on Linear Algebra*, Williamsburg, VA, 2003.

[14] D. S. Bindel and S. Govindjee. Elastic PMLs for resonator anchor loss simulation. *International Journal for Numerical Methods in Engineering*, 64:789–818, 2005.

[15] D. S. Bindel, E. Quévy, T. Koyama, S. Govindjee, J. W. Demmel, and R. T. Howe. Anchor loss simulation in resonators. In *MEMS 2005, IEEE international conference*. IEEE, January 2005.

[16] David S. Bindel. Annotated bibliography on computer support for scientific computation, 2002. `http://www.cims.nyu.edu/~dbindel/class/cs279/dsb-bib.pdf`.

[17] David S. Bindel. *Structured and parameter-dependent eigensolvers for simulation-based design of resonant MEMS*. PhD thesis, University of California, Berkeley, August 2006.

[18] David S. Bindel. Generalizing spectral and pseudospectral theorems to nonlinear eigenvalue problems. Preprint., 2008.

[19] David S. Bindel, Zhaojun Bai, and James W. Demmel. Model reduction for RF MEMS simulation. In *Proceedings of PARA 04*, volume 3732/2006 of *Lecture Notes in Computer Science*. Springer, June 2004.

[20] David S. Bindel, Shivkumar Chandresekaran, James W. Demmel, David Garmire, and Ming Gu. A fast and stable non-symmetric eigensolver for certain structured matrices. Technical report in progress, 2005.

[21] David S. Bindel, James Demmel, and Mark Friedman. Continuation of invariant subspaces in large bifurcation problems. *SIAM Journal on Scientific Computing*, 30(2):637–656, March 2008.

[22] David S. Bindel, James W. Demmel, and Mark Friedman. Continuation of invariant subspaces for large bifurcation problems. Technical Report UCB/EECS-2006-13, EECS Department, University of California, Berkeley, February 2006.

[23] David S. Bindel, James W. Demmel, Mark J. Friedman, Willy J.F. Govaerts, and Yuri A. Kuznetsov. Bifurcation analysis of large equilibrium systems in MATLAB. In *Proceedings of ICCS 2005*, volume 3514, pages 50–57. Springer-Verlag, April 2005.

[24] David S. Bindel, James W. Demmel, William Kahan, and Osni Marques. On computing Givens rotations reliably and efficiently. *ACM TOMS*, 28(2):206–238, June 2002.

[25] David S. Bindel and Iva Vukicevic. Error bounds and estimates for a discrete sine-Gordon model. Preprint., 2008.

[26] David S. Bindel and Maciej Zworski. Symmetry of bound and antibound states in the semiclassical limit. *Letters in Math Physics*, 81(2):107–117, August 20007.

[27] Cynthia Bruyns, Robert Taylor, and Carlo Séquin. Virtual instrument design and animation. In *SIGGRAPH 2004 Visual Proceedings (sketch)*, August 2004.

[28] Angelika Bunse-Gerstner and Heike Fassbender. Error bounds in the isometric Arnoldi process. *J. Comput. Appl. Math.*, 86:53–72, 1997.

[29] R. N. Candler, H. Li, M. Lutz, W.-T. Park, A. Partridge, G. Yama, and T. W. Kenny. Investigation of energy loss mechanisms in micromechanical resonators. In *Proceedings of the 12th International Conference on Solid State Sensors, Actuators and Microsystems (Transducers 03)*, pages 332–335, Boston, June 2003.

[30] S. Chandrasekaran, P. Dewilde, M. Gu, W. Lyons, and T. Pals. A fast solver for hss representations via sparse matrices. *SIAM J. Matrix Anal. Appl.*, 29(1):67–81, 2006.

[31] S. Chandrasekaran, P. Dewilde, M. Gu, T. Pals, X. Sun, A.-J. vander Veen, and D. White. Some fast algorithms for sequentially semiseparable representations. *SIAM J. Matrix Anal. Appl.*, 27(2):341–364, 2005.

[32] T.-Y. Chen and J. W. Demmel. Balancing sparse matrices for computing eigenvalues. *Linear Algebra and its Applications*, 309:261–287, April 2000.

[33] Y. Chen, A. Bargteil, and D. Bindel. Quantifying network denial of service: A location service case study. In S. Qing, T. Okamoto, and J. Zhou, editors, *Proceedings of the Third International Conference on Information and Communications*

Security (ICICS 01), volume 2229/2001 of *Lecture Notes in Computer Science*, Xian, China, November 2001. Springer-Verlag.

[34] Y. Chen, D. Bindel, and R. H. Katz. Tomography-based overlay network monitoring. *ACM SIGCOMM 2003*, 2003. Poster abstract to appear in ACM Computer Communication Review.

[35] Y. Chen, D. Bindel, H. Song, and R. H. Katz. An algebraic approach to practical and scalable overlay network monitoring. In *Proceedings of ACM SIGCOMM 2004*. ACM, 2004.

[36] Y. Chen, D. Bindel, H. Song, and R.H. Katz. Tomography-based overlay network monitoring. Technical Report CSD-03-1252, University of California at Berkeley, CS Division, June 2003.

[37] Yan Chen, David S. Bindel, Hanhee Song, Brian Chavez, and Randy H. Katz. Algebra-based scalable overlay network monitoring: algorithms, evaluation, and applications. *ACM Transactions on Networks*, 15(5):1084–1097, October 2007.

[38] W. C. Chew and Q. H. Liu. Perfectly matched layers for elastodynamics: a new absorbing boundary condition. *J. Comp. Acoust.*, 4(4):341–359, 1996.

[39] D. Chua, E.D. Kolaczykk, and M. Crovella. A statistical framework for efficient monitoring of end-to-end network properties. In *Proceedings of ACM SIGCOMM 2005 (poster)*. ACM, 2005.

[40] F. Collino and P. Monk. The perfectly matched layer in curvilinear coordinates. *SIAM Journal of Scientific Computing*, 19:2061–2090, 1998.

[41] J. W. Demmel, L. Dieci, and M. J. Freidman. Computing connecting orbits via an improved algorithm for continuing invariant subspaces. *SIAM J. Sci. Comput.*, 22(1):81–94, 2001.

[42] A. Dhooge, W. Govaerts, Yu.A. Kuznetsov, W. Mestrom, and A. M. Riet. MATLAB continuation software package CL_MATCONT, January 2003. `http://www.math.uu.nl/people/kuznet/cm/`.

[43] Lars Grasedyck and Wolfgang Hackbusch. Construction and arithmetics of h-matrices. *Computing*, 70(4):295–334, 2003.

[44] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.

[45] W. Hackbusch and Z. P. Nowak. On the fast matrix multiplication in the boundary element method by panel clustering. *Numerische Mathematik*, 54:463–491, 1989.

[46] Antti Jaakola, Piia Rosenberg, Arto Nurmela, Tuomas Pensala, Tommi Ricekkinen, James Dekker, Tomi Mattila, and Ari Astalo. Piezotransduced single-crystal silicon BAW resonators. In *Proceedigs of the 2007 IEEE Ultrasonics Symposium*, 2007.

[47] Tsuyoshi Koyama, David S. Bindel, Wei He, Emmanuel Quevy, James W. Demmel, Sanjay Govindjee, and Roger T. Howe. Simulation tools for damping in high frequency resonators. In *12th International Conference on Solid-State Sensors, Actuators, and Microsystems (Transducers '03)*, November 2005.

[48] J. Kubiatowicz et al. OceanStore: An architecture for global-scale persistent storage. In *Proceedings of ASPLOS 2000*, November 2000.

[49] P. G. Martinsson and V. Rokhlin. A fast direct solver for boundary integral equations in two dimensions. *J. Comput. Phys.*, 205(1):1–23, 2005.

[50] Cleve Moler. ROOTS – of polynomials, that is. *The Mathworks Newsletter*, 5(1):8–9, 1991.

[51] C. T.-C. Nguyen. Vibrating RF MEMS for low power wireless communications. In *Proceedings of the 2001 International MEMS Workshop (iMEMS01)*, pages 21–34, Singapore, July 2001.

[52] Manoj Pandey. *Analysis of entrainment and clamping loss in an optically actuated MEMS*. PhD thesis, Cornell University, August 2007.

[53] B. N. Parlett. *The Symmetric Eigenvalue Problem*. SIAM, 1997.

[54] V. T. Srikar and Stephen D. Senturia. Thermoelastic damping in fine-grained polysilicon flexural beam resonators. *Journal of Microelectromechanical Systems*, 11:499–504, 2002.

[55] G. W. Stewart. Error and perturbation bounds for subspaces associated with certain eigenvalue problems. *SIAM Review*, 4:727–764, 1973.

[56] R. L. Taylor. FEAP. `http://www.ce.berkeley.edu/~rlt/feap`, 2007.

[57] F. Teixeira and W. Chew. Complex space approach to perfectly matched layers: a review and some new developments. *International Journal of Numerical Modelling*, 13:441–455, 2000.

[58] E. Turkel and A. Yefet. Absorbing PML boundary layers for wave-like equations. *Applied Numerical Mathematics*, 27:533–557, 1998.

[59] Jaan Unger, November 2005. Personal communication.

[60] Iva Vukicevic. Error bounds and estimates for a discrete sine-Gordon model. Courant Summer Undergraduate Research Experiences (SURE) final report, October 2008. Available at `http://www.cims.nyu.edu/vigrenew/ug_research/`.

[61] J. Wang, J. E. Butler, T. Feygelson, and C. T.-C. Nguyen. 1.51-GHz nanocrystalline diamond micromechanical disk resonator with material-mismatched isolating support. In *Proceedings MEMS 2004*, pages 641–644, Boston, 2004.

[62] J. Wang, Z. Ren, and C. T.-C. Nguyen. Self-aligned 1.14 GHz vibrating radial-mode disk resonators. In *12th International Conference on Solid-State Sensors, Actuators, and Microsystems (Transducers '03)*, pages 947–950, 2003.

[63] Wikipedia. Floating point — wikipedia, the free encyclopedia, 2005. [Online; accessed November 2005].

[64] C. Zener. Internal friction in solids I: Theory of internal friction in reeds. *Physical Review*, 52:230–235, 1937.

[65] C. Zener. Internal friction in solids II: General theory of thermoelastic internal friction. *Physical Review*, 53:90–99, 1938.

[66] C. Zener, W. Otis, and R. Nuckolls. Internal friction in solids III: Experimental demonstration of thermoelastic internal friction. *Physical Review*, 53:100–101, 1938.

[67] Yao Zhao, Yan Chen, and David Bindel. Towards deterministic network diagnosis. *SIGMETRICS Perform. Eval. Rev.*, 34(1):387–388, 2006.

[68] Yao Zhao, Yan Chen, and David S. Bindel. Toward deterministic overlay diagnosis (poster). In *ACM SIGMETRICS/Performance 2006*, pages 387–388, 2006.

[69] Yao Zhao, Yan Chen, and David S. Bindel. Toward unbiased end-to-end network diagnosis. In *Proceedings of SIGCOMM 2006*, pages 219–230, 2006.