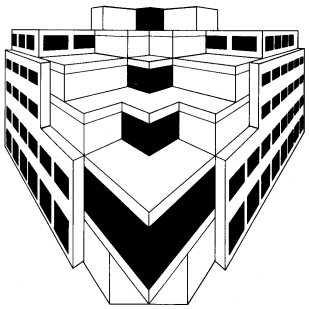


conduit!

Volume 6, Number 2 **Department of Computer Science** **Fall, 1997**
Brown University

COMPUTATIONAL BIOLOGY DEBUTS AT BROWN

Franco Preparata
An Wang Professor

In the relatively remote past, the science of life concerned itself with macroscopic features of living matter, which were used to explain both the morphology and the functions of living organisms. Such features were the basis also for taxonomy of species, in that similarities, both morphological and functional, were construed as evidence of relatively close common ancestry on the evolutionary trail. Although the empirical laws of heredity, as well as the physical sites of the heredity carriers, had been known for a long time, the detailed mechanisms of the process remained nearly inscrutable until well into this century, except for the realization that certain chemicals, the nucleic acids, were intimately connected with the process.

Scientific discoveries of about half a century ago were bound drastically to subvert this established mode of scientific inquiry. Crucial was the elucidation of the structure of DNA (deoxyribonucleic acid) and of its role as the fundamental carrier of hereditary information. The revolutionary discovery of the DNA double-helix structure by Watson and Crick in 1953 ushered in the era of *molecular biology*. They showed that DNA is a sequence of pairs of four structurally similar basic constituents called *bases* and denoted by the standard letters A, C, G, and T (as is well known, these are the initials of their respective chemical denominations). In fact, each base can be paired (has strong chemical affinity) with just another base, so only the pairs AT, CG, GC, and TA occur in the DNA sequence. While this view of DNA is perfectly adequate for its description, it is perhaps more significant to consider DNA as the pairing of two complementary strands, each carrying the same hereditary (genetic) information. This structure is essential for DNA replication, the archetypal phenomenon of reproduction: the two strands of a sequence are separated in the cell and each of them is copied into a complementary strand, giving rise to two replicas of the original sequence. This brief and very schematic digression is not intended to oversimplify marvelously complex biological phenomena, but simply to provide a glimpse into the emerging discrete structure of molecular biology.

In fact, the realization that the above four bases are the building blocks of the description of the genetic patrimony is appropriately viewed as the informatization of biology, in that it shifts the description into the conventional computer-science nomenclature of sequences over a finite alphabet. This feature is not exclusive to DNA, but recurs for other biomolecules, such as RNA (the other fundamental nucleic acid) and, with a larger alphabet, for proteins. This characterization establishes a natural link between the two domains, since they use analogous descriptive devices.

Contemporary with the beginning of molecular biology was the advent of the computer era. In its first decade, the rather rudimentary technology made the computer seem more a wondrous curiosity than a tool accessible to vast segments of users. The physical size of the installations, the associated physical plant, and the dismally poor reliability of the computers in no way let anyone suspect its ubiquitousness today. Thus it is not surprising that contacts between biology and computer science materialized somewhat later.

As the computer field was progressing rapidly (already in the sixties was computer science identified as an autonomous academic discipline), the informatization of biology revealed an entirely new host of problems. The notions of morphological or functional similarity evolved into the notion of similarity between sequences (polymers) of chemical constituents. For any class of homologous such sequences (we mean here just that two sequences are homologous if they can be meaningfully compared) this approach immediately poses two problems. The first is the *definition of the metrics*, i.e., a quantitative model for the measurement of sequence similarity (or distance). The second is the development of methods (*algorithms*) to carry out such quantitative assessment.

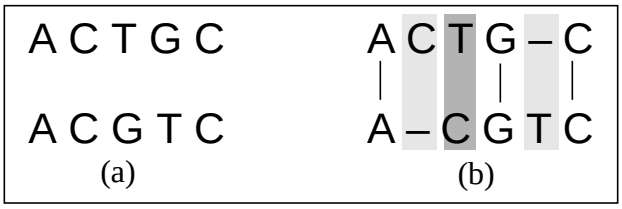


Figure 1. An example of sequence alignment

For example, consider the two DNA sequences displayed in Figure 1a. How similar are they? Obviously, two sequences are identical if no modification is needed to "transform" one to the other. Therefore, to quantify similarity we must first specify the types of primitive operations allowed to transform one sequence to the other and, second, we must assign a "penalty" weight to each of them. With this model in place, we must seek the least-weight sequence of operations that realizes the desired transformation. If the operations are substitution and insertion/deletion and have all penalty 1, then we find that the alignment of Figure 1b describes a least-weight transformation (of weight 3, one substitution and two insertions/deletions).

This trivial example illustrates the features of the alignment problem, which plays an important role in computational biology, both for its modeling difficulties (which are the biologically significant primitives and their weights?) and its algorithmic complexity (the collective alignment of several sequences of several hundred characters each). In addition to alignments, a vast collection of problems lies today at the intersection of computer science and biology: DNA fragment assembly, physical mapping, phylogeny, molecular structure prediction, genome rearrangement, and so on. Many of these problems have been stimulated by the human genome project (i.e., the mapping of the entire human DNA patrimony) but have also been fueled by rapidly growing industrial interest. The typical function of drugs has been identified as the key-to-lock fit of the drug to some "geometric" feature of the agent to be controlled, and since form is largely determined by structure (i.e., sequence), therapeutics is the emerging professional field drawing from computational biology. Another important aspect is to intervene in the genetic mechanisms of diseases. Quoting from a recent advertisement: "Our focus is to identify and characterize genes involved in common diseases and to translate our discoveries into therapeutic break-throughs...Challenging and rewarding opportunities exist..." The market is so active as to prompt a recent commentary (*Science*, 21 June 1996) entitled "Hot Property: Biologists Who Compute."

"...Many problems have been stimulated by the human genome project ...but have also been fueled by rapidly growing industrial interest"

"Hot Property: Biologists Who Compute" Science, 21 June 1996

Typically, computational biologists have been professionals from either field who have taken on the difficult task of retraining themselves in the other discipline. This approach has several shortcomings. First, individuals willing to undertake such an unconventional educational

path (self-teaching) are not the norm, and thus are not numerous enough to fill a clearly identified professional need. Second, a self-instruction plan may not be sufficiently systematic to meet the requirements. Third, one must take into consideration a subtle feature in the sociology of peer groups in research and professions, that is, "acceptance". A peer group accepts individuals with similar academic backgrounds, and such acceptance is rarely complete in the case of retraining. Indeed, it has been quite common for computer science to force (unrealistic) modelings for the benefit of algorithmic simplicity, and for biologists to content themselves with commercially available software tools, rather than undertaking original algorithmic development.

This state of affairs and the propitious opportunity offered by the excellent flexibility of Brown's undergraduate curriculum prompted the idea of proposing a new undergraduate concentration in computational biology. Whereas several graduate programs in computational biology exist at various universities in the country, we felt it essential to undertake the initiative at the earliest possible stage of college education. The motivation was to prepare graduates who would feel equally at ease with biochemistry and life sciences as with algorithms and software engineering, and be naturally recognized as peers by graduates in either camp. Whereas it may be observed that the instructional offerings were, to a large extent, already available in Brown's rich repertoire, a well thought-out plan can be an invaluable guide for the student oriented towards this new field and in addition provides the desired professional label. A proposal, originally conceived by David Rand, Bill Suggs, and myself, and fine-tuned by the advice of several other colleagues, was presented to the College Curriculum Council and approved in May 1997.

"The motivation was to prepare graduates who would feel equally at ease with biochemistry and life sciences as with algorithms and software engineering"

The core offerings of the concentration are designed to provide a balanced background in the interacting disciplines. This core is complemented by specialized tracks designed to differentiate among a number of related professions with identifiable expertise and skill. Thus the software track is for students interested in developing commercial software for biological applications; the molecular modeling track is for students interested in competence in molecular modeling and drug design; the biological sciences track is for students interested primarily in biological questions. In addition to core courses and electives, the program requires as its capstone experience the completion of a senior research project under close faculty supervision. The stewardship of the program is entrusted to three concentration advisors, respectively in Computer Science, Chemistry, and Biology and Medicine. Beside the usual advising responsibilities, the advisors have the task of supervising the evolution of the program.

Minimally, the proposed program identifies among the current offerings an instructional package that can be legitimately named Computational Biology. As the program evolves, we expect the establishment of a permanent lecture series and of graduate research and instructional initiatives, and, possibly, the addition of faculty clearly identifiable with the field.

David Rand, left, from the Graduate Program in Ecology and Evolutionary Biology, and Bill Suggs, Chemistry.

Artwork by CS undergraduate Michael LeGrand (mml)

Bacchandum est nobis !

THE VISION THING !

New chairman Tom Dean

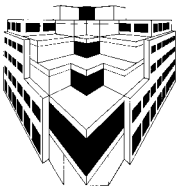
A talk given by new chairman Tom Dean at the fac-grad lunch 9/10/97.

I want to see the Brown Computer Science Department become the technological and sybaritic hub of the #1 party school in the United States. I've already coerced some overworked students in the graphics group to produce a new logo and motto for Brown. The new motto is—translating from the Latin—"Party down at Brown!" We're still rendering the logo, but to give you a preview, it involves a large, well-fed bear in a party hat with a full-view, surround-sound virtual reality helmet and one-million-point tactile-sensor data gloves. If we can't get the rest of Brown to loosen up, we will virtually secede and align ourselves with the University of West Virginia, using telepresence and our extensive experience in virtual collaboratories.

Of course, I'm being facetious, but I'm trying to make a point. The chairman of a department doesn't lead so much as listen, creatively combine, and facilitate. As the new CS chairman, I see one of my primary roles as trying to figure out what it is the faculty and students want and then trying to make it happen. I have to trust in your collective wisdom and good will to discuss issues openly and wherever possible reach a consensus. I'm lucky that at this particular juncture in the department's history I know exactly what it is that you want and need ... just kidding.

My job has been likened to "herding cats." Academic units are full of very bright, highly opinionated, and at times over-confident (at other times we are awash in self-doubt) prima donnas. We have to be this way in order to be successful in exploring uncharted research areas, pursuing hunches, and arguing for ideas that we believe hold promise. As I see it, the chair has to get people talking with one another, inspire collegiality, moderate factions at meetings, and create alliances and agreements that I liken to forging exotic composites out of only dimly understood alloys that are stable only in carefully maintained conditions. I won't even joke about our being of one mind on academic and departmental matters, but I also strongly believe that, compared with other institutions, we are remarkably collegial and willing to listen and seriously consider one another's ideas.

So what is it that we want and how do we go about realizing our collective vision for the future? Before we get into this, I want to reiterate that we are a diverse collection of minds with different agendas, and as a department we are allocated only finite funds to realize our dreams. Since I'm addressing the graduate students just now, it's worth reminding you that as a department we are also mandated to teach undergraduates and we take that mandate very seriously—our undergraduate concentrators are among the best in the world and we treasure them as students and often as colleagues and collaborators. So, in addition to the pushes and pulls from different research interests and



graduate and faculty needs, we have to integrate undergraduate teaching (and research) into our plan for what we want to be.

The needs of the graduate students and the faculty (depending on what hat they're wearing) are largely research-directed. Offer us the classes we need, provide us with the best research faculty in the world, provide critical mass in the areas we are interested in so that we have choices and differing perspectives, give us the machine resources and ample office space, provide funding for a colloquium series that brings the best people with the most exciting new ideas to Brown, and then leave us alone so that we can get our work done. Simple as this may sound, there are a myriad of choices. And it is important to be aware that almost everything about our field is competitive and the best students and the best faculty are a precious and limited resource. Trying to predict what CS might become is close to impossible, and apparently even more difficult for people close to the technology—don't ask any of the faculty for stock picks. Following the "next new thing" is a recipe for disaster. Instead, we have sought out the best people who represent a blend of "theory and practice," by which we mean that we look for people who are well trained in computer science, mathematically and technologically savvy, in touch with interesting real-world applications, but driven to subject their ideas to rigorous evaluation, whether using techniques from analysis of algorithms or experimental studies coupled with careful statistical analysis. Smart, well-trained people will seek out the interesting problems, the problems that need solution, and they will adapt and recreate themselves if need be to meet the challenges of the future.

"Smart, well-trained people will seek out the interesting problems, the problems that need solution, and they will adapt and recreate themselves if need be to meet the challenges of the future"

So that "solves" the problem of selecting faculty and students? Well, not quite. There are cultural issues that loom large in a small department. We want people who can get along, who recognize what needs to be done and do what it takes to get it done. We also want to encourage a sense of community and one particularly effective way of doing this is to bring in people who have something in common and recognize and encourage opportunities for synergy wherever possible. Students and faculty alike want a critical mass, a sense of purpose and directed activity that only colleagues can provide, and colleagues in your same institution or department are the ideal.

This brings up a resource issue. We are small and, though we want to grow, our growth will be limited; there are many areas that could serve as the basis for a critical mass of people—way too many for us to cover them all. What criteria do we use for areas to emphasize? In this case, we focus on our faculty, for they'll be here the longest, and in particular we focus on the youngest so that they can shape their futures and create an environment in which they can thrive, and so that the future of the department is secure. The basic recipe is a pretty familiar one in academia. We seek to build so-called "centers of excellence" around the interests of our junior faculty. In selecting people, excellence is still the most important criterion but we bias our selection so that we create synergy and provide the basis for collegiality. We try to work against the creation of factions or areas that become little fiefdoms unto themselves by being constantly aware of the negative effects of Balkanization.

There isn't time to go into a detailed argument why, but it is my belief and that of most (if not all) of the faculty that we need to grow to survive. We need to be larger in order to attract and retain the best students and faculty. Exactly how much bigger is a subject of heated debate (and of course something that requires the consent of the administration), but if we are to maintain our high ranking (and naturally we have higher aspirations than simply holding position) then we need to continue to compete with departments that are two and three times our size and have plans to grow in some cases by half again as much. These other departments, in their zeal to expand, look greedily at our faculty, and their parent institutions appear ready to back them with the positions, salaries, stipends, startup packages, etc., that make it easier for them to jump ship, with only small twinges of guilt or longing for the easy collegiality we all cherish at Brown.

So we will do our best to get administration approval to expand, taking into account our needs for area coverage at the graduate and undergraduate levels and our desire to maintain our culture and sense of community. Getting more faculty members will ultimately require getting more top-quality graduate students and (and this is a big "and") more office, lab and classroom space. We think we have a very strong argument to make to the administration about how the CS Department should figure more prominently in the university, what it can offer to the undergraduate population, and why we need more resources and more faculty in particular to meet these challenges. It's going to require a lot of effort to convince the administration, and that will be just the beginning. Once we have the administration behind us, we have to suffer the pain and distraction of growth. It will be exciting and demanding. From his writings it is apparent that our new president Gordon Gee is interested in graduate education. Brown is one of the best undergraduate institutions in the world, but its record is spotty on graduate departments. Yet Computer Science and Applied Math are world-class and among the very top departments in the university.

President Gee has a history of rewarding excellence and building on strength. But we have to make clear our vision for the future. Computer Science is not just an arcane discipline. It isn't just hacking code or building systems; it constitutes a very different way of looking at the world, a perspective rich in conceptual content and promise for society. We need to convey this perspective to all of our students and provide in our undergraduate concentrations a foundation that will serve as a basis for a wide range of exciting careers, from molecular engineering and medical informatics to commerce and collaboration and distance learning on the World Wide Web. The world is changing as it wakes up to the promise and consequences of the information age.

In a recent article President Gee quoted the CEO of General Electric, Jack Welch, as saying, "When the rate of change outside exceeds the rate of change inside, the end is in sight." Welch seems to be a source for all sorts of pithy inspirational sayings—on his top-ten list of rules for success he also has

> Face reality as it is, not as it was or as you wish it to be.
>
> Control your own destiny, or someone else will.
>
> Change before you have to.
>
> Be candid with everyone.

The first three seem particularly relevant to us in considering the future of the department. The last I just like. By the way, I hope that I'm not reduced to spouting similar power-of-positive-thinking sayings like the above—I find it a little alarming that I found the above interesting and worth including. I may be losing my cynical edge.

LETTERS & ALUMNI EMAIL

John Crawford '75

JOHN CRAWFORD '75

Since winning the prestigious Eckert-Mauchly award, noted in the spring '94 issue of ***conduit!***, John, an Intel Fellow and Director of Microprocessor Architecture, has again been cited for engineering leadership. In June he received the IEEE Ernst Weber Engineering Leadership Recognition award for his "leadership in the development of microprocessors for the personal computer industry." In a recent *Wall Street Journal* article John was lauded as "Intel's Top Chip Architect" about to unveil his latest creation—the IA-64, a next-generation microprocessor architecture that will be implemented in a chip code-named Merced.

E. GORDON GEE, President-elect

Dear Professor Kaelbling: I did have an opportunity to read the comments presented by you at a college convocation which were reprinted in your departmental newsletter (***conduit!***, V6, N1). I would be remiss if I did not tell you how much I enjoyed your speech and, more important, the obvious commitment you have to your work and the university. It is obvious that Brown is fortunate to have you as a member of its faculty.

I look forward to an opportunity when we can meet. As you well know, the transition from one university to another is never easy. This one is particularly difficult, given the size and complexity of Ohio State. Nonetheless, I look forward to joining you and my other faculty colleagues as soon as possible. Do know of my best personal wishes.

SWAMI MANOHAR, PhD '89

It was fun visiting the department after a break of about five years (I peeped in last in '92 during commencement), close to a decade (that long!) after getting out with my PhD (and the rubber chicken, which incidentally isn't doing too well, got squished by a row of books in my cupboard for a couple of years before I retrieved it in two parts). Quietly checked out room 555 which was my office for the few months I was in the (then) new building: John Stasko, the housing czar, had worked out some formula by which grad rooms were allotted, and I, Rob Ravenscroft and Cheryl Harkness had combined our seniority to grab this lovely office. I think another weighting factor was our willingness to move from Gould to a converted house next to the Gould building the previous year. I went over to Gould and found a nice little open space where the house-converted-to-office used to be!

Swami Manohar and antiquities

And I remembered missing by a week the privilege of presenting the first PhD defense in the new building: I am not sure who beat me to it, it was either Eric Golin or Rob Rubin. It was good to see Jeff and Max still holding the systems fort, Trina the department treasury, and Jennet the chairman's office. I pushed my status as an ex-grad student to the limit by barging into the Chairman's office to wish Tom a successful tenure as chairman. Andy was the perfect host even while communicating (over cell phone/video conference/ fax/email/..) with a zillion people coordinating a huge project review. Had a nice chat with John (Savage). Walked around the various labs, the ex-pingpong room, the copy room, said hello to a few others who made the mistake of recognizing me, and was generally a nuisance till I was forced to leave to catch my bus back to New York City.

President-elect E. Gordon Gee

I am currently here at the University of Missouri at Columbia as a visiting associate professor for this academic year. I am on sabbatical leave from my permanent faculty position at the Indian Institute of Science, Bangalore. My seven-year-old triplets and my wife Sathya are with me here, and who knows, we may inflict ourselves collectively on the department before we leave for India next year. You have been warned! In answer to Suzi's question about our children, I have two girls and a boy. The boy's name is Adithya (can mean either the First One or the Sun). The girls are Yamini (again, can mean either a Heavenly Maiden or a Star) and Indumathi (Moon-Moon or Sweet Moon). And thanks, Suzi, for taking my photo, especially the (appropriate) one with all the ancient machines! Best regards

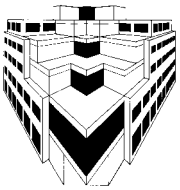
manohar@rapid.cecs.missouri.edu

JONATHAN MONSARRAT

I took a leave of absence in 1995 to start a computer games company now called Turbine Entertainment *(www.turbinegames.com)*. Turbine makes MUDs (Internet-based multiplayer role-playing games) with full 3D graphics and plenty of bells and whistles. The company is doing well; hopefully the first game will finally ship by spring 1998, as part of a publishing deal with Microsoft. Other Brown CS folk involved with the company include Tim Miller, Jeremy Gaffney, Tim Brennan, Andy Reiff, and Mike Rubin. We've been grateful for advice and assistance from Professors Andy van Dam, Rob Netzer, Tom Doeppner, and John Hughes.

Through curious circumstances, including wanting to contribute more to society than just making games, I'm now off in the biotech field at Genome Therapeutics, a genomics company in Waltham, MA. They read the DNA of disease-causing bacteria and viruses, as well as the DNA of humans who have genetically related diseases. Computer people here do a lot of software engineering to organize the labwork and store the DNA information in a database. There's a lot of interesting algorithmic work in analyzing the DNA—comparing similar DNA strands, using AI to model gaps in the biologists' knowledge of molecular pathways, and molecular model-ing of proteins (the chemicals in the body that DNA creates).

In the last few months, I've given a couple of talks at Brown on how to start a company and about life in the real world—pitfalls, common mistakes, tips for success, and so forth. There's a standing offer to anyone in the community who'd like some advice in that regard. I would welcome a hello at *jonmon@genome corp.com*.



JOE PATO, ScB '81

Andy, I read two articles last night that related to you, so I decided to drop a note to say hello. One was a reprint of a recent interview with the ProJo that was carried in a suburban New York paper that a friend mailed on to me. The other was this Sunday's *New York Times* magazine section that quoted Ted Nelson on hypertext and Frank McCourt (also a former teacher of mine) on multi-threaded novels.

I'm still here at HP working in Ira Goldstein's Internet Technology Group as the lead security architect (far afield from my Brown days, but the area I've been working in for the past 10 years). Hope all is well with you—your picture in the paper betrays few changes. Jeri is well and continues at home with our two sons, Isaac 7 and Dylan 3.5.

pato@apollo.hp.co

JANNE SAHADY, ScM '79

Please give my regards to Andy van Dam. I remember his dynamic lectures and amazing course load very well as excellent preparation for survival in the technical real world. Andy set a teaching standard I haven't seen matched yet.... I am currently working for Analysis and Technology, Inc. in Middle-town, RI, as the Project Director of Information Systems at Newport Hospital, now in transition to Lifespan. My daughter Nicky, born in 1979, attended my graduation from Brown and is now graduating from high school. I also have a second daughter, Freida, three years younger than Nicky.

jsahady@atinc.com

SCOTT RAPOSA, ScB '94

Well, it's been three eventful years for me out in the 'real world' so I thought it was time to check back in with my family from the CIT. After graduating from Brown in '94, I decided to spend some time up in Boston, so I accepted a software engineering position at Lotus. There, I spent my days helping the team build the first Windows '95 version of our product, Freelance Graphics. The goal was to beat Microsoft to the punch by releasing the first Windows '95 office suite—I'm not sure the plan ever worked. In any case, just before I left Lotus in July '95, I got to sit in a large auditorium and watch Lou Gerstner and Jim Manzi walk around the stage together trying to convince us that the new IBM/Lotus team was going to provide the world with one final chance at uprooting Microsoft. Last I heard, Lotus is going to be bundling Microsoft's IE 4.0 with every new copy of Lotus Notes. Hmmm.

After leaving Lotus to move to Pittsburgh (long story), I ended up managing a development team at a software startup called Danet, Inc. We had a single product—a client/server billing and customer care system—that we sold to the first PCS (Personal Communication Services) providers in North America. It was an extremely interesting time to be in an extremely interesting industry. I'm not sure I learned much about software, but I did get a crash course in the cutting-edge of wireless telecommunications.

Finally, just a few months ago, I decided I had waited long enough, so I jumped ship to start my own business. In April of '97, I went out and registered 'tigerleaf.com' so that Tiger Leaf Communications, Inc. would have a home on the Internet. Today, there are six of us full-time (that is, 9 am 'til midnight, Mon-Sun). What do we do? It really depends on the week. The best part about living inside an Internet startup is that the world changes drastically about once every month. In any case, we're giving the public a small taste of what we're up to at *www.tigerleaf.com*. But all of the exciting stuff is going on behind-the-scenes. Oh, and if you live in or ever plan to visit Pittsburgh, you're invited to our Food Festival! at *www.foodfestival.com*.

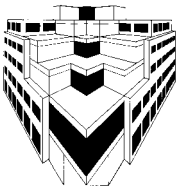
Well, I can't wait until I have the time and resources to begin recruiting at Brown. I'm afraid that if I don't start soon, Tiger Leaf will eventually be known as a Carnegie-Mellon company—and we can't let that happen.

Take care, and hopefully we can all meet up in Providence sometime soon.

scott@tigerleaf.com

DAVID WADHWANI, ScB '93

Andy, Mike Horn ('97) started working at Actioneer a couple of days ago. I immediately noticed something and could not help but grow concerned about your influence on students. *My experience:* After four years of working with you I cannot even call my parents without a notepad and pen in hand! *Mike's experience:* I had my first meeting with Mike to give him an overview of our



Dave Wadhwani

architecture. He brought neither pen nor notepad—I nearly fainted! You are certain that you know Mike and that he worked for you?

I just wanted to thank the Brown Computer Science Department for all it has done and continues to do for Brown CS graduates. Every day it becomes more and more obvious how a Brown computer science degree has helped prepare graduates in so many ways. Fellow Brown graduates that I have worked with continue to impress me technically and personally. Brown taught us not only to be technical today, but it showed us how to stay on top of the fastest evolving industry in the world! It prepared us not only to engineer, but went the extra step to teach us how to work with engineers. And most importantly, it instilled the confidence needed not to just follow but rather to lead. For all this and more I thank the professors, the administrative staff, the teaching assistants and the students!

On a personal note, I have been at a small company called Actioneer for just over a year. While at Actioneer, Dennis Buchheim '92 and I led a team of developers in creating a peer-to-peer collaboration framework. This has been a great experience, but I plan to leave Actioneer and start a company, possibly with some other Brown graduates. If anyone out there is interested in learning more about the idea or might be interested in joining us, please email me at *wadhwani@non-sequitur.com*. Ever True ...

http://www.non-sequitur.com

BRIAN KNEP ScB '90, MS '92

I just checked the latest ***conduit!*** online—good job. Unfortunately there's an error that needs to be corrected in the next issue: I didn't work on a 3D paint program while at Brown. Some other folks there developed a 3D paint program but the work done at ILM wasn't based on it and in fact took a very different approach. Thanks.

THE PARADIGM SHIFT FROM ALGORITHMS TO INTERACTION

Peter Wegner

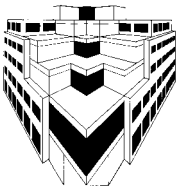
The paradigm shift from algorithms to interaction captures the evolution from procedure-oriented to object-oriented languages, from number-crunching to embedded software systems, and from logic-based to agent-oriented artificial intelligence. Objects, agents, and embedded systems require radically new models of computation because they cannot be modeled by algorithms. The proof that algorithms cannot model interaction turns out to be surprisingly simple. The "impossibility result" that algorithms cannot express interaction allows us to escape from the Turing tarpit and frees us from the obligation of expressing interactive models of real-world applications in algorithmic terms.

Interaction Machines

Interaction machines extend Turing machines by adding direct input and output actions that interact dynamically with the external environment. It is easy to show that interaction machines cannot be modeled by Turing machines with a finite initial input tape because any finite input stream can always be extended. They are at least as powerful as Turing machines with infinite initial input tapes, which are known to be more expressive than Turing machines. This proof reinforces informal evidence that the behavior of operating systems, robots, and other computing agents is not algorithmic.

Computer science is an experimental discipline whose models are validated by their usefulness in the development and analysis of applications. Interactive models sufficiently expressive for practical applications supplement tractable algorithmic models that cannot completely specify actual software systems. Interaction machines are a canonical model for interaction comparable to Turing machines for algorithms.

Interactive models extend the intuitive notion of computability. Church's thesis that the intuitive notion of computing is formally captured by Turing machines, accepted as obviously true for over 50 years, breaks down for interactive computing. Though Church's thesis is valid in the narrow sense that Turing machines do express algorithmic computing, the broader view that algorithms capture interactive computing is not. Algorithms capture "automatic computing" in which the sequence of execution steps is completely controlled by the algorithm, while interaction machines express nonautomatic computing whose interaction steps are externally controlled. The intuitive notion of computing clearly includes objects, agents, and embedded systems, and must therefore be interactive.



Modes of Interaction

Modes of interaction provide an outside-in view of components that complements the inside-out view of algorithms. The analysis of systems by their mode of interaction is an interactive analog of the analysis of algorithms by their mode of execution. However, modes of interaction provide only a qualitative framework for analysis, since quantitative complexity analysis is not applicable to systems.

Software engineering (SE) and artificial intelligence (AI) can be distinguished by their characteristic modes of interaction. SE focuses on reactive systems that provide services by reacting to the requests of clients, while AI focuses on proactive agents that act on their environment to realize external goals. At a lower level of granularity, types classify values by their modes of interaction, while interfaces specify systems by their modes of interaction.

Reactive systems that passively supply services have simpler environment models than proactive agents that try to understand and change the world. If interactive expressiveness is defined by external modeling power, then proactive AI systems are more expressive than reactive SE systems. The reactive services provided by components to clients can be very complex, but they are generally algorithmic. Component interfaces provide both an abstraction of the external world to proactive agents looking outward and an abstraction of the inner worlds of reactive components to clients looking inward.

Modes of interaction provide a unifying descriptive framework for SE. Software architecture deals with alternative modes of interaction like pipes, client-server, and blackboard models. Interoperability examines interaction among heterogeneous components that differ in platform and interface definition. Object-oriented design models specify interaction among objects differently from computation within objects. Design patterns and frameworks determine modes of interaction that can be classified and reused but cannot be proved correct, formally specified, or formally composed. Modes of interaction also provide a unifying descriptive framework for AI. Learning, planning, and acting have characteristic modes of interaction. Proactive agents actively learn about the external world, build complex internal models of the world, and perform actions to change it. Agents use incremental data to update their model of the world as a basis for action. Planning systems combine updating their world model with the execution of policies that maximize their expected reward over finite or infinite time horizons, while learning systems explore the world and build models for later action.

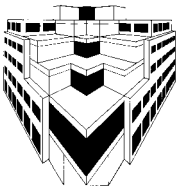
"Virtual reality achieves its realism by integration of spatial, stereo and temporal perspectives: footprints in space are combined with footprints in time to create cognitively realistic virtual worlds"

Modes of interaction can be modeled mathematically as projections of the world on the input sensors of an agent. Stimuli S from an external world W are projections $S = P(W)$ onto input sensors such that the inverse cannot be completely known (W cannot be reconstructed from S). Projection mappings P provide a mathematical tool for analyzing modes of interaction. Incompleteness of S in specifying the world W is related to Godel incompleteness.

An agent's knowledge of the world is expressed by Plato's cave metaphor, which compares humans to cave dwellers who can observe only shadows on the walls of their cave (retina) but not the external world. S is the shadow cast by W on the walls of the agent's cave. However, S can include stereo inputs from multiple sensors (eyes and/or ears) and temporal inputs at successive points of time, and indeed can be a stereo-spatio-temporal interaction pattern more complex than a two-dimensional image on the walls of a cave. Virtual reality achieves its realism by integration of spatial, stereo and temporal perspectives: footprints in space are combined with footprints in time to create cognitively realistic virtual worlds.

Models of Interaction

Grammars and first-order logic are non-interactive models whose extension to interaction parallels that of Turing to



interaction machines. Generative grammars specify languages by noninteractive generating rules, while logics derive theorems by noninteractive rules of inference. The expressive equivalence of Turing machines, grammars, and logic derives from their common assumption of noninteractiveness.

Grammars for interaction extend off-line generating rules for strings to on-line "listening" rules with temporal semantics as in speech. The extension from logic models to models of external worlds parallels the extension from algorithms to interaction. The use of the term "model" for both logical and empirical models reflects their common goal of expressing semantics by syntactic representations. These parallel, though conceptually very different, extensions from algorithms to interaction suggest that interaction is a robust form of computable behavior that can be described in many alternative ways, just like algorithms:

"Dijkstra's view of testing as an inadequate approximation to correctness must be modified because correctness of systems is in principle unspecifiable"

- *Turing machines (state-transition steps)* → *interaction machines (interaction steps)*
- *generative grammars (off-line strings)* → *interaction grammars (on-line streams)*
- *first-order logic (true inferences)* → *empirical models (external modeled worlds)*

The perspective gained by escaping from the Turing tarpit shows that algorithms are in fact quite limited in their modeling power. Noninteractive problem solving in both humans and computers is autistic: algorithms solve problems by autistically following rules and cannot ask for or receive external help while computing. The pejorative statement that a person is acting like an automaton means he or she is executing an algorithm rather than being interactive.

The irreducibility of interaction to algorithms captures the essential difference between experimental science and mathematics, enhancing the intellectual legitimacy of computer science as an experimental discipline distinct from mathematics. Interaction machines provide a unifying framework for talking precisely about the conceptual foundations of model building, so that philosophical distinctions between rationalism and empiricism can be concretely expressed by "algorithms versus interaction." The belief that empiricism is more expressive than rationalism can be precisely stated and proved by reducing philosophical to computational arguments. Fred Brooks' intuition that there is no silver bullet for specifying complex systems follows from the irreducibility of interaction to algorithms.

Tradeoffs Between Formalization and Expressiveness

Tradeoffs between formalizability and expressiveness arise in many disciplines but are especially significant in computer models. Overemphasis on formalizability at the expense of expressiveness in early models of computing led to principles like "go to considered harmful" and the more sweeping "assignment considered harmful" of functional programming. Functional and gotoless programming, though beneficial to formalizability, are harmful to expressiveness. However, they merely make certain kinds of programs more difficult to write without reducing problem-solving power. The restriction to Turing machines is a more serious harmful consequence of formalization, since it reduces problem-solving power to that of algorithms so that objects, personal computers and networks cannot be adequately modeled.

Logic specifies complete models of noninteractive behavior while interactive models provide incomplete models of real and conceptual external worlds. Logic is too weak to capture completely the true statements or behavior of interactive systems (a form of Godel incompleteness). Incompleteness plays havoc with formalizability, but interactive models of practical applications are inherently incomplete.

Though completeness and correctness must be abandoned, many forms of incomplete reasoning are useful in interactive analysis. Notions of conditional and probable correctness sufficient for practical purposes can play a role similar to testing and symbolic execution. Dijkstra's view of testing as an inadequate approximation to correctness must be modified because correctness of systems is in principle unspecifiable. Interactive testing to show partial, conditional, or probable correctness is a form of incomplete reasoning for gathering evidence of system reliability.

Algorithmic parts of a system can be proved correct, but correct interactive coordination among algorithmic parts cannot be guaranteed. Software engineers already know that algorithm correctness is insufficient to specify nonfunctional software qualities like reliability, and interactive models corroborate this well-known fact. Result checking is another useful technique that replaces static checking prior to execution by dynamic checking of the result: the execution-time binding of result checking is analogous to dynamic binding of interactive inputs.

The first version of these ideas was developed in 1992 when I was invited to participate in the closing conference of the Japanese Fifth-Generation Computing Project. Many participants felt that this project had been terminated prematurely and could have achieved its goal of "computing by logic" with more time and money. I had noticed that logical completeness (by backtracking) was incompatible with reactiveness (timely responsiveness) in logic programming, and argued that logic was inherently weaker than computing. During five years of further research, informal as well as formal evidence for this thesis has accumulated and become overwhelming.

Increasing evidence from technology confirms that algorithms can no longer serve as a complete foundation for computer science and that a broader intuitive notion of computing is needed. More details can be found in an article in the *Communications of the ACM*, May 1997, and in papers available on my home page at *www.cs.brown.edu/people/pw/*.

Since the spring issue of ***conduit!*** we have had four thesis defenses.

Shieu-Hong Lin's "Exploiting Structure for Planning and Control" was directed by Tom Dean. Shieu-Hong is attending the Logos Evangelical Seminary in California.

Swarup Acharya's dissertation, advised by Stan Zdonik, was "Broadcast Disks: Dissemination-Based Data Management for Asymmetric Communica-tion Environments." Swarup is working for Lucent Technologies, Bell Labs Innovations in New Jersey.

Anthony Cassandra's topic was "Exact and Approximate Algorithms for Partially Observable Markov Decision Processes," directed by Leslie Kaelbling. Tony is heading to Austin, Texas, where he'll work for MCC (Microelectronics and Computer Technology Corporation).

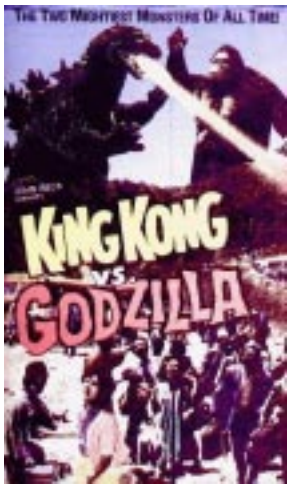
Pascal Van Hentenryck's advisee **Viswanath Ramachandran** defended his thesis on the topic "An Optimizing Compiler for CLP(R_Lin)." He has a position with Netscape Communications in California.

Salvaged from the rubble of Richardson House, the University crest is restored to glory on the north face of the new W. Duncan MacMillan science building

Symposium speakers from l to r: back row: host Tom Doeppner, Hal Stern, Sun; host Steve Reiss, Bill Blake, Digital; front row: David Korn, AT&T Labs; Sean Quinlan, Bell Labs; Jon Hall, Digital and LINUX International. Richard Draves from Microsoft Research is not pictured.

THE 19TH IPP SYMPOSIUM

Steve Reiss



The 19th Industrial Partners Program technical symposium, held April 3, was on the topic "NT vs. UNIX: Whither the Future?" This topic was chosen to coincide with the near-finish of our two-year study on replacing our aging Sparcstation workstations, a study that involved extensive and sometimes acrimonious debates on the relative merits of UNIX and NT in an educational environment. The IPP symposium was scheduled to explore others' experience with the two operating systems and to share this information with our partners along with what we had learned from our study. The speakers included representatives from Sun and LINUX International talking about the future of UNIX, a speaker from Microsoft discussing the future of NT, and several speakers who have used both systems and could relate their experiences.

After the usual introductions, the day opened with Hal Stern from Sun Microsystems. Hal, an old friend of the department (he has helped us extensively with technical aspects of our Sun configuration in the past), talked about the future of computing. His emphasis was on the upcoming revolution toward ubiquitous computing, the use of distributed objects, and how current systems, in particular operating systems, will need to adapt to the new environments. Much of his talk explained Sun's Java strategy and how it will fit into this vision of the future. He ended by prognosticating that while the big player twenty years ago was IBM with its mainframes and today's big player is Microsoft with PCs, twenty years from now the big player will be Disney with its multi-faceted content and that the underlying machine, operating system, and languages will not be important.

The second speaker was John Hall from Digital. John came wearing his hat as president of LINUX International, the organization that is driving the development of the freeware UNIX operating system LINUX. His talk provided a rationale for a freeware UNIX clone and described its current capabilities, its unique distributed development strategy, and where he saw it going in the future. Then John put his Digital hat on top of his LINUX hat and described Digital's role in LINUX development and support. John was invited back to the department to provide additional details about LINUX about a month later.

Sean Quinlan, a researcher from Lucent Technologies (Bell Labs), followed. He related his experiences porting UNIX code to the windows environment. He found windows overly complex and complained that it was "not UNIX." To aid porting he had developed a common library, 9pm, that allowed applications to be developed simultaneously for both UNIX and NT. He briefly described this library and his experiences with it.

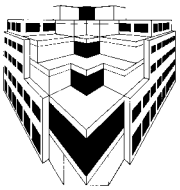
The final speaker before lunch was Digital's Bill Blake. Bill is the director of the HPTC/Core technology group, which is in charge of developing most of Digital's programming tools. Bill's hypothesis was that NT was for programmers while UNIX was for programs, i.e. that programmers want to sit in front of an NT box because of cost and the overall environment, but that many of the support tools needed, such as configuration-management tools, tools for distributed building and debugging, and tools to manage large software systems, tend to run better in a UNIX environment. Moreover, he predicted that while programmers prefer a development environment based on Microsoft's Visual Studio under NT, there will continue to be a demand for server applications that run under UNIX, since UNIX will continue to be the operating system of choice for high-end platforms. He described Digital's strategy of providing an integrated NT/UNIX environment that allows and supports cross-platform development by integrating features from both systems.

Richard Draves, from the operating systems research group at Microsoft Research, was the first speaker after lunch. His talk concentrated on his work with the future of networking. He discussed the future of networking support for NT based on IPv6, an updated version of the current Internet protocol, and described how Microsoft is rethinking its overall approach to distributed systems in order to raise the level of abstraction and make large-scale distributed objects practical. The solution he outlined, Millennium, will be incorporated into future releases of the NT operating system.

The final speaker of the day was David Korn from AT&T Labs, the developer of the Korn Shell and many other pieces of software. David described his experiences in using both NT and UNIX, emphasizing the problems that face an experienced UNIX programmer when attempting to develop windows programs. He outlined the various tools available commercially and publicly to aid in porting to NT, with emphasis on their various drawbacks. He then described tU/WIN, the UNIX library he is developing that lets applications be ported with a minimum of effort, and the various problems he had to solve.

The day concluded with a panel discussion emphasizing the pros and cons of NT versus UNIX. The audience raised lots of concerns regarding the capabilities and future of NT. Most of these were addressed by the various speakers and the result was a lively discussion contrasting the two operating systems. While there were no overall conclusions, it was generally agreed among the panelists that NT was here to stay for the desktop and that it had the potential to be as good or better than UNIX, but that it wasn't quite there yet.

***Computing Surveys* is pleased to announce that Scott Lewandowski of Brown University will receive its first award for Best Student Paper for his "Frameworks for Component-Based Client-Server Computing." Peyman Zehtab-Fard of Umea University, Sweden, will receive honorable mention for his "Still Image Compression and File Formats." Peter Wegner, the Editor-in-Chief, wishes to thank the student committee, headed by John Cavazos of the University of Massachusetts, and the members of the editorial committee, Marvin Israel of Bellcore, Eliot Moss of the University of Massachusetts and Alan Tucker of Bowdoin College, who agreed unanimously with the recommendations of the student committee. Papers from half a dozen countries were submitted to the 1997 competition. The submission deadline for the 1998 award is June 30, 1998.**



Eugene Charniak. Eugene's paper at the American Association for Artificial Intelligence conference received a Best Paper award (though interestingly, there were four best papers this year). He also was invited to give the after-dinner talk at the Uncertainty in Artificial Intelligence Conference's banquet. This latter was a slightly daunting event. Said Eugene, "I have had a fear of after-dinner talks ever since a conference about 15 years ago when an academic gave a traditional academic talk as the after-dinner talk. Our table had had a fair amount of wine, and one of my colleagues did a *sotto voce* parody of the talk that had us all laughing so hard that we created quite a spectacle. Wanting to avoid this happening to me, I asked some colleagues for advice, and one said that making fun of everyone else at the conference always worked. I decided that that was not my style, but I could make fun of myself, and the bulk of my talk was exactly that. This had a beneficial side effect: since there is so little about me to make fun of, my talk was quite short."

Tom Dean. Together with research scientist Bob Givan, Tom coauthored papers presented at AAAI in Providence, CUAI in Providence, and IJCAI in Nagoya, Japan, this summer. At ECP in Toulouse in September he gave an invited talk in addition to presenting their joint paper.

Tom Doeppner. Tom was given funding by Sun for research in multithreaded programming and in widespread dissemination of information on the Internet.

John Hughes. Spike co-authored three papers at SIGGRAPH this summer, thus tying with two others for the largest number of technical papers. One of his SIGGRAPH 97 paper's images was selected for the front cover of the conference proceedings book and CD-ROM.

Leslie Kaelbling. Leslie won the Computers and Thought Award presented in August at IJCAI in Nagoya, Japan, where she made a plenary address in conjunction with the award.

Philip Klein. Philip served on the Program Committee for the IEEE Symposium on Foundations of Computer Science. He gave an invited talk at the Dagstuhl Seminar on Approximation Algorithms in Germany and will give one in December at the Princeton DIMACS workshop on Randomized Methods in Algorithm Design.

Franco Preparata. At the beginning of the year a series of videotaped lectures presented the year before at the University of Kyoto was edited by Franco's Japanese colleagues into a volume entitled *Professor Preparata's Lectures on Parallel Computation* and published in Japanese. At the same time, Franco received an honorary doctorate from the University of Padova, Italy (the third oldest university in Europe). In March he was invited to Kyoto to be the keynote speaker at a technical symposium honoring Professor S. Yajima. In June he visited INRIA in Sophia-Antipolis, France, and co-chaired a workshop on robust geometric computing.

John Savage. John has completed his third book, *Models of Computation: Exploring the Power of Computing*, for Addison Wesley Longman. It can be used for an undergraduate theory course as well as advanced courses and contains material on a large variety of concrete models of computation. Several Brown people have lent their expertise to this tome—Jose Castanos (jgc) and Dimitrios Michailidis (dm) read John's manuscript thoroughly several times, offering many constructive suggestions; the cover was designed by Michael LeGrand (mml) and Scott Klemmer (srk), both juniors;

layout of the book designed by doctoral candidate Dimitrios Michailidis; copyediting by Katrina Avery (kha) and a great index by Rosemary Simpson (rms). John has been elected a Fellow of the American Association for the Advancement of Science. In his capacity as Director of the Industrial Partners Program, John welcomed MERL and EMC as new Industrial Partners.

John Savage and Dr. Lubchenco, President of AAAS, during his induction as an AAAS Fellow

Andries van Dam. Brown has received a substantial grant from NSF for a scientific/visual supercomputing center. It will likely be housed in the old computer center at 180 George, where some readers got their start with computing in general and graphics in particular! Andy was one of the co-PIs (in conjunction with faculty from the Divisions of Applied Math and Engineering and the Department of Physics). The grant will buy us a multiprocessor and a virtual reality four-wall cave. Andy was honored by being chosen a Senior Class Marshal at commencement this year—"...you get to hug a lot of women and even a few men!" said he.

Pascal Van Hentenryck. Pascal's new book *Numerica: A Modeling Language for Global Optimization* (co-authored with Laurent Michel and Yves Deville) was published last April by MIT Press. Pascal is program chair of the International Symposium on Static Analysis (SAS '97) and gave an invited talk at IJCAI in Nagoya.

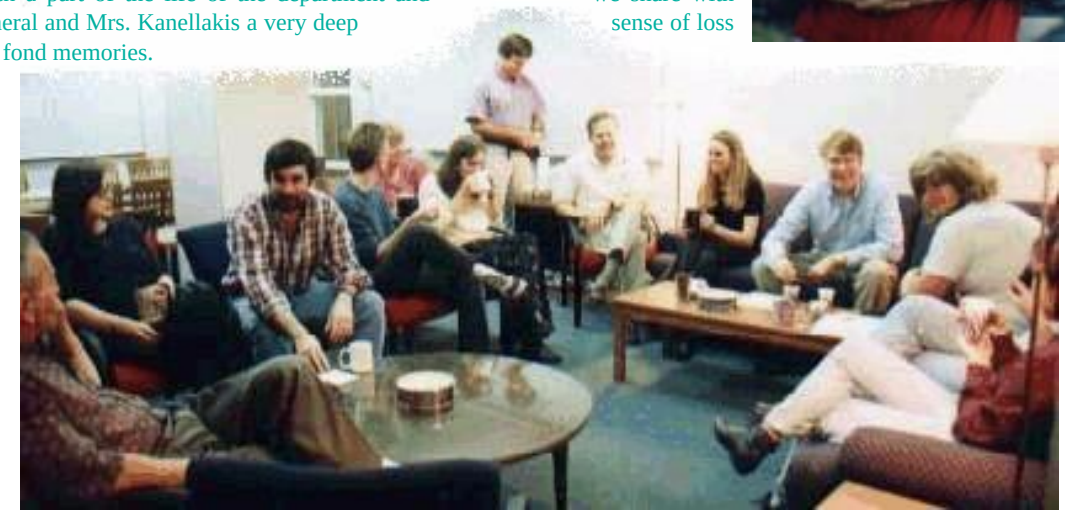
Andy in full regalia with Jennet Kirschenbaum, Assistant to the Chairman

Peter Wegner. Peter lectured in Norway, Finland, Austria, Germany, England, Israel and Brazil during the summer of '97. He chaired the Kanellakis Award committee, which gave its first award in March for a theoretical contribution with a significant impact on practice to Diffie, Hellman, Merkle, Shamir, Rivest and Adelman for their work in public-key cryptography. He coordinated the *Computing Surveys* symposium on Strategic Directions in Computing, was an associate editor of the 2500-page *Handbook of Computer Science and Engineering*, and continues his oversight of the ACM transition to electronic publication as a member of the ACM Publications Board. His research on interactive computing (*CACM*, May 1997) is described in this issue of ***conduit!***

The contributions to **ACM's Kanellakis Award** for the impact of theory on practice have exceeded their goal of $100,000, so that annual giving of this award is now assured.

Emmanuel (Manos) Renieris, the first recipient of Brown's Kanellakis graduate fellowship, visited General and Mrs. Kanellakis in Athens during the summer, together with Christos Papadimitriou. He said he felt honored and privileged to be the first fellowship recipient. Born and raised in Athens, Manos attended the National Technical University, the oldest technical insti-tution in Greece and the same university that Paris attended. Paris's advisor, Emmanual Protonotarios, is still teaching there.

The department library here in CS has been renamed in Paris's honor and has been refurnished with money contributed to Brown's Kanellakis Memorial Fund to serve as a combination lounge and reading room. Paris is still very much a part of the life of the department and we share with General and Mrs. Kanellakis a very deep sense of loss and fond memories.



THE GRACE HOPPER CONFERENCE

With the strong support of department chairman Tom Dean and funding from the Motorola grant presented last spring in conjunction with our Industrial Partners Program, three CS students attended the 1997 Grace Hopper Celebration of Women in Computing, held September 19-21 in San Jose.

Tom had this to say:

"The first Grace Murray Hopper Celebration held in Philadephia in 1995 was limited to 400 and many potential attendees had to be turned away. The 1997 event in San Jose had over 600 attendees, and prospects for future events are bright. The networks forged in the halls and conference rooms of the Hopper Celebration are a powerful tool that benefits both women computer professionals and computer science as a whole: women educating and mentoring one another, celebrating the achievements of female computer scientists, and creating a web of communication and collaboration. The Hopper Celebrations have provided and will continue to provide a unique opportunity for woman in computer science."

All three students found it to be a challenging and enriching event. Following are their comments:

Ilana Frankel (email)

Of all the items of interest and pieces of wisdom garnered over the course of the weekend, one phrase alone effectively sums up the Grace Hopper Conference experience: Anita Borg, her Systers' Keeper and consulting engineer at Digital, sent us scholarship students into laughter with her anecdotal words: "Imagine having to wait in line for the bathroom at a computer conference!"

Several months ago, news of the Grace Hopper Celebration of Women in Computing had me applying for what I anticipated would be, at the very least, a forum for stimulating discussion among exciting women—in retrospect, an understatement. I heard technical lectures delivered by female scientists on the cutting edge of their fields: biomedical imaging, operating systems, computational theory; provocative and engaging panels on subjects

ranging from the validity of affirmative-action-based pedagogical methods to the implications of interdisciplinary research. And me in the middle of it all. Adding my own voice: as a student-as-yet-unsure of her academic career path, questioning and commenting; as a student-with-experience, sharing with the high-school girls in attendance. Learning and being inspired and waiting on line for the bathroom—and loving it.

from l to r: Jennifer Stewart, Valerie Green, Ilana Frankel

Valerie Green (email)

There was a good mix of technical talks and talks specifically related to women's issues. In particular, there were some informative discussions on the differences in the ways men and women communicate that Jennifer Stewart and I will present to students here at Brown.

An important aspect of the conference was the ability to meet so many women in the field, whether in industry or academia. While many of the women in computer science at Brown are mentors for first-year CS students, it is important to give those mentors some support themselves. I was able to talk to many Brown alumnae as well as other women in industry and academia, which gave me connections I can use in the future. It also was so inspiring that I feel I have more energy to devote to similar local activities related to women in the sciences. Many ideas emerged about how to encourage women to explore computer science and how to keep them in the field, including a proposal by Anita Borg for a new institute.

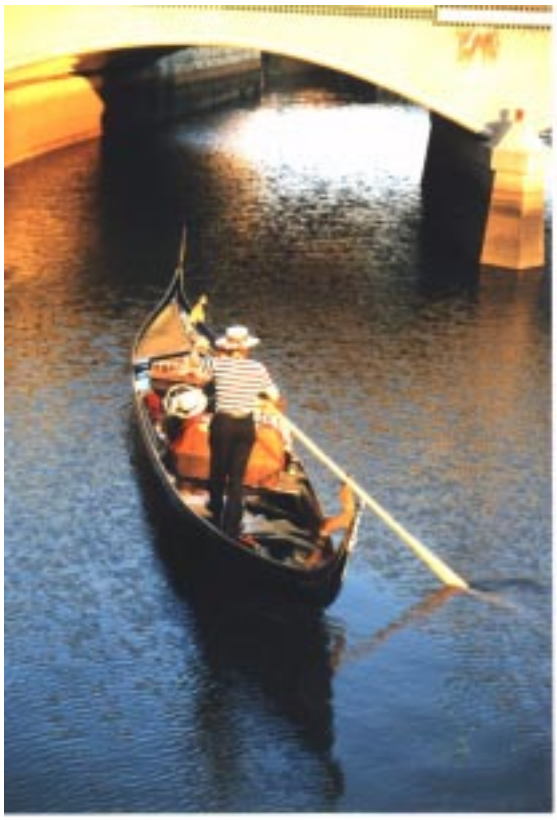
Many efforts by WiCS at Brown will be aided by the contacts we made at the conference, in areas such as the alumnae network and on-line mentoring programs. One fact seems clear, however: if the Computer Science Department at Brown truly wishes to diversify, the department should allocate some staff and funding for a full-time position to administer a program. As it is, WiCS is run by busy students who do not have the time necessary to deal with the administrative effort of creating and maintaining a program to attract women and minorities. Women in Science and Engineering (WiSE) at Brown has just received a university liaison and increased administrative support; our department needs to take a step like this if WiCS is expected to grow and succeed in drawing women into the department.

Jen Stewart (from her web page)

Some highlights of the Grace Hopper Celebration were a talk on how advances in image processing are helping doctors detect breast cancer and a talk called "Windows in the Glass Ceiling" that gave suggestions for getting to the top based on interviews with women CEOs from Fortune-1000 companies. It was wonderful to be in an environment filled with women who are accomplished in this field, as well as those just starting out.

We also met some great folks from The Backyard Project, which is an initiative of Katrina Garnett, founder and CEO of Crossroads Software, whose goal of the project is to encourage highschool girls to go into computer science. We met with them in small groups and discussed their goals for the future and their reactions to the conference. We hope to be involved with the Backyard Project as it expands in the future.

Steve Reiss and his 130lb gigapumpkin !



Providence, Venice of the east coast!

CHARNIAK UNPLUGGED

Eugene Charniak

"I am happy to announce that the university has selected Tom Dean as the next chair of the department, and that he has accepted the position. I am also happy to announce that Steve Reiss has accepted the position of Associate Chair. Their terms will start as of July 1. I would like to wish both of them the luck to live in dull times, though I am sure that many things will happen to liven them up."

When my six years as department chair were up last June 30, I sent out the above e-mail, moved my mug collection back to my old office, and shed the demands of the chairmanship faster than you can say "" (the empty string). However, bowing to overwhelming popular demand that I keep up my ***conduit!*** column (well, nobody objected out loud), here goes.

The American Association for Artificial Intelligence (AAAI) 1997 national conference was held here in Providence in August. As AAAI conferences go it was typical, with one exception. I found it very interesting to get a view of Providence from a conference-goer's perspective, and I was surprisingly impressed. I hardly ever see downtown Providence except after dark, on the way to Trinity Rep (Providence's award-winning professional theater company), when it is empty and somewhat depressing. Now I got a chance to see downtown in the daytime as a conference attendee. It looked clean and remarkably inviting. Also, the weather was terrific. Several of my colleagues said they had no idea Providence was such a great city, and one said she was so impressed that if Brown had any openings we should let her know. I am sure our mayor would be delighted to hear of someone who would like to come to Brown because of Providence, and not vice versa.

Our mayor, Buddy Cianci (a man of checkered past but relentless good cheer, and an amazing Providence booster) was also on my mind that week the two times I walked home from AAAI through Waterplace Park, a new park built along the Woonasquatucket and Moshassuck Rivers, near the convention center. About four months ago the mayor was the force behind someone's buying a gondola (built in Manchester, MA, according to specifications mailed from Venice) so that people could take rides along the river. I and all my friends just put this down as another of Buddy's crazy ideas, but on both occasions I found that the gondola was in use, and on one there were actually a few people in line waiting for the next trip. It seems to be a success: I just read in the paper that the demand has been such that someone has brought in a second one (actually made in Venice). Now that I think of it, perhaps I should not have been so sceptical. Buddy is also known locally for selling bottles of marinara sauce made to his own recipe (proceeds go to a scholarship fund). My wife and I saw it in the supermarket a while ago and bought some as a joke. The joke was on us. It was very good.

A follow-up on atrium whiteboard usage (***conduit!*** V6, N2): two CS students using it to study for a Math 10 (introductory calculus) final wrote on the side of the board "Math 10—Do not mock us please—yeah, you!"

I suppose it is obvious (and hackneyed) that people are multidimensional, but two recent incidents brought it particularly to my attention. The commonplace distinction between personal and professional came up when my family accompanied me to a conference this summer at a resort in the Columbia River Gorge. At checkin I starting talking with another conference participant. Lynette, my wife, speaking to both of us, noted that I had not bothered to introduce myself. My ego

received a boost when the other conference attendee answered, “It isn’t necessary. He’s famous.” Reality returned when Lynette responded, “From my perspective he’s still the guy who needs to be reminded to take out the garbage on Sunday night.” (How does the saying go, “No programmer is a hero to his computer”?)

Faculty books and a collage of book covers artfully displayed for a recent visit from members of the Corporation

A more interesting clash of dimensions concerns Professor Stan Zdonik. In the ***conduit!*** (V2, N1) article on faculty hobbies we noted that two of Stan’s major interests are auto maintenance and bluegrass music. (Stan is also known around the department for his fountain pen collection, mentioned in the spring ***conduit!*** piece about Jeff ‘Ice Man’ Coady.) Recently one of Stan’s friends, Ken Irwin, the president of Rounder Records (a record company specializing in country and bluegrass), came to see Stan here in the department. His visit made personal a fact he had previously known only intellectually: the person he identified as a bluegrass master of ceremonies and occasional musician also had a life as a professor of computer science, and this seemed odd to him indeed.

In late June the power failed in the CIT. The screens (and many offices in the department) went black and in the shocked silence we heard a jubilant shout: “It worked!” (The voice was that of Mark Oribello, whose oddball sense of humor has already been fuel for this column.)

conduit!

A publication of
The Computer Science Department
Brown University

Inquiries to: *conduit!*
Department of Computer Science
Box 1910, Brown University
Providence, RI 02912
FAX: 401-863-7657
PHONE: 401-863-7610
EMAIL: sjh@cs.brown.edu
WWW: http://www.cs.brown.edu/publications/conduit/

Suzi Howe
Editor-in-Chief

Katrina Avery
Editor

Jeff Coady
Technical Support

Department of Computer Science
Brown University
Box 1910, Providence, RI 02912, USA

conduit!

NON-PROFIT
U.S. Postage
PAID
Providence, RI
Permit #202

Printed on recyled paper

Address changes welcomed