CS101
What do Professors do?

Prof. Patrick H. Madden
pmadden@acm.org
www.cs.binghamton.edu/~pmadden
A quick advertisement

- The local ACM chapter is at http://acm.cs.binghamton.edu
- They run local programming contests about once a month
- A few years ago, our team went to the world finals (one of about 80 teams from 3500 worldwide).
  - We beat Harvard and Brown on our way there. We beat some good schools too.
  - Our team was flown to the finals by Google (they paid for all the teams). And then they got a trip to Google HQ. And then they all got offers from Microsoft that were too good to turn down. And now two of the three have been lured away -- one to Facebook, one to a Facebook app builder.
A quick advertisement

- Bloomberg sponsors our team -- they pay travel expenses, provide contest prizes, and their recruiters show up at contests. Many past ACM team members are now at Bloomberg.

- Want to have a successful career? Put in the effort. There’s an abundance of people who want to be computer scientists, but a shortage of people who are willing to put in the work required to become great.
A Bit of History

• Born in Detroit, Michigan
• Grew up in Albuquerque, New Mexico
• Undergrad: New Mexico Institute of Mining and Technology
• Startup company doing oil field automation
• MS: New Mexico Institute of Mining and Technology
• Startup company doing oil field automation, part 2
New Mexico Tech

Current enrollment: 1800
Socorro population: 9000
A Bit of History

- After going out of business a few times.... PhD at UCLA
  - I got in to UCLA based on research I did while at NMT.
  - The education I got at the super-cheap school was every bit as good as anything at UCLA.

- While at UCLA
  
  - Part-time work at Activision making games
  
  - MechWarrior, Muppet Treasure Island, and a really great game that never got released. This payed very well....
Current Stuff

- Teaching CS120, CS333
- Teach CS575 at the grad level
- Chair of ACM/SIGDA -- a professional society for integrated circuit design researchers
- Bash parallel computing "experts." Some research on circuit design. Also some work on protein identification.
So, what do these professors actually do?

- Drink coffee and look like we’re thinking
- Teach classes
- Research
- Professional Service
- Start companies
Teaching

• “You can lead a horse to water, but you can’t make him drink”

• Education is the sleeping pill that makes dreams happen. (King of the Hill)

• If you’re sufficiently motivated, you will learn the material no matter what

• If you’re not motivated, you will not learn the material, no matter what

• Don’t kid yourself. It’s all up to you. The professor will have only a minor impact on what you learn.

• Grades don’t matter.
Research

• The bartender says “we don’t serve superluminary particles due to the possibility of causality inversion.”

• A neutrino walks into a bar.
Research

• There’s lots of things we don’t know, or that has not been invented yet
  • The web is “new.” So is the computer. And penicillin, the internal combustion engine, and zippers.

• Solve the energy crisis?
  • Is it possible to make efficient solar panels? Wind? Geothermal? Biofuels?
  • Reclaim the hot air generated by Politicians?

• Can we cure cancer?
  • Or at least the common cold?
  • Or eliminate viruses from Microsoft operating systems?
My Research Focus (one of them at least)

• Right now, we can’t make chips go faster (they burst into flames)

• So... Everyone and their dog is trying to design parallel computers

• This is a horrific mistake. I’m trying to beat some common sense into the rest of the research community, and come up with a solution.
Data Structure Co-Processing

• Many of the best algorithms rely on data structures
  • In particular, priority queues, trees, hash tables, and so on, require some compute time for insert, delete, ….
  • Opportunity to extract some parallelism; when inserting an item, the main thread doesn’t need to wait for the data structure to reorganize
  • Operations on the data structure may take tens or hundreds of cycles
  • Clean break between main algorithm and data structure simplifies programmer adoption
  • Possible to hide all the complications behind the data structure API. One might obtain parallel performance gains by simply linking an alternate data structure library.
DSCP Approach

(a) Conventional serial processing

(b) Data structure co-processing
Data Structures Considered

- STL Container Classes:
  - Set, stack, vector, map, queue, deque, list, slist, priority queue, multiset, multimap, hash map, hash m-map, hash sets

- Access times for insert, delete, search vary; in general significant (ranging from hundreds to several thousand cycles)

- Opportunities to exploit overlap?
  - Depends on the data structure, and the algorithm
  - Encouraging results on the “hard serial bottleneck” of Dijkstra’s algorithm

- Extensive experiments with different types of objects, data structure sizes, to gain insight into possible application areas
How does this fit in the landscape?

• Many groups are looking at heterogeneous processors (with many cores)

• Why not a core dedicated to common DS tasks?

• Our focus is on the hard serial bottlenecks

• Small gains on the serial sections lead to greatly increased overall scalability
An Example Application

- Navigation (path finding) is a key feature in many portable devices
- The main underlying algorithm: Dijkstra
  - $O(E + V \log V)$
  - Effectively $O(n \log n)$ for sparse graphs (road maps)
  - Even better complexity with specialized priority queues. Very close to $O(n)$ in practice.
- Dijkstra’s Algorithm is serial in nature
Dijkstra’s Algorithm

- Priority Queue of vertices, ordered by distance
  - Implementation as a binary heap in this study; Fibonacci heaps, etc., are also possible

- Dijkstra’s main loop:
  - Pop a vertex from the queue
  - Relax edges to neighboring vertices
  - Update positions of vertices in the heap
A Simple Example

A Simple Example

A
B
C
D

Pop A
Reconfigure Heap
Compute distance to B
Update heap Position of B
Compute distance to D
Update heap Position of D

Monday, September 26, 2011
A Simple Example

A

B

C

D

5

4

0

3

9

Pop A

Reconfigure Heap

Compute distance to B

Update heap Position of B

Compute distance to D

Update heap Position of D

Monday, September 26, 2011
A Simple Example

A

B

C

D

5

3

4

9

5

3

Compute distance to B
Update heap Position of B
Compute distance to D
Update heap Position of D

Pop A
Reconfigure Heap
The Basic Observation

A

B 5

C

D 3

5

4

9

Pop A
Compute distance to B
Reconfigure Heap
Compute distance to D
Update heap
Position of B
Update heap
Position of D

Monday, September 26, 2011
Keys to Making this Practical

- The main Dijkstra loop does only a few tens of instructions for each edge relaxation
- A heap update also requires only a handful of instructions (log n at worst, typically in the tens for graphs such as road maps)
- Any method of work distribution must be very low cost
Possible Implementations of DSCP

- Multithreaded (SMT): data structure related processing is performed on a separate thread context within the same core
- Multicore: additional core is used for heap-related processing
  - Commodity multicores: threads communicate through shared memory
  - Hardware support: threads communicate through inter-core registers
Experimental Results

Reduction in CPU Cycles Normalized to Baseline

L2 Cache Size

512 KB 1 MB 2 MB 4 MB

Baseline
Multicore DSCP
Instantaneous Heap Access
Experimental Results

![Bar chart showing experimental results for different L2 cache sizes and conditions.](image-url)
Observations on Dijkstra Experiments

- Dijkstra’s algorithm is stubbornly sequential, with tight coupling between the algorithm and the data structure
- On real-world examples, 25% performance gain through DSCP
  - No significant algorithmic or code changes
  - 25% is not a huge amount – but this comes in the “serial” portion of an application, and thus directly attacks the barrier of Amdahl’s Law
Pitfalls for Accelerating the Shortest Path Problem

- Bellman-Ford can be made massively parallel
- The gotcha – Bellman-Ford is $O(VE)$
- Effectively $O(n^2)$ for sparse graphs
- No matter how much parallel resource is available, the serial Dijkstra algorithm outpaces it for large graphs
The Research Process

• Read technical literature
• Think
• Think some more
• Read some more
• Think
• Get a good idea (if you’re lucky)
• Write it down, publish
Note two important steps

• Reading. Sorry, you’ve got to be able to read.

• Writing. Sorry, you’ve got to be able to write.

• Not everything you need to know comes with a YouTube video. Some ideas take more than 140 characters to explain. You can’t cut-and-paste a new idea from Wikipedia.
And part of the research time...

- Goes into writing proposals to get funding to do more research
- You have to explain what you want to do, and convince the folks with the money to pay you to do it (the success rate is around 7%, and there’s a LOT of politics).
- Tonight, I’ll be working on a research proposal, with a deadline of October 4th*
Professional Service

- Mentioned earlier: reading and writing technical literature
- ACM sponsors conferences, and runs many journals and magazines
  - Chock-full of research from universities (and industry) from around the world
- Someone has to organize all this stuff
  - That would be me, at least for research on integrated circuit design
Welcome to the Association for Computing Machinery (ACM). ACM is the world's largest educational and scientific computing society, delivering resources that advance computing as a science and a profession. ACM provides the computing field's premier Digital Library and serves its members and the computing profession with leading-edge publications, conferences, and career resources.

Advancing Computing as a Science and a Profession

ACM has resources for everyone

With our leading Digital Library of computer science literature, world-class publications, online books and courses, professional and student member benefits, and institutional packages — ACM will enrich your professional, educational, and research life.

Ambassadors for ACM

Tell your colleagues about ACM, and share the benefits of membership!

ACM Digital Library Helps Users Customize and Share Computing Resources

Site simplifies usability, extends connections, expands content.

Announcements

Call for Nominations: ACM Transactions on Graphics
ACM is seeking an Editor-in-Chief for its Transactions on Graphics (TOG). The deadline for nominations is November 30. Visit the EIC search page for more information.

Call for Nominations: ACM Selected Readings Series
ACM is seeking an Editor-in-Chief for its ACM Selected Readings Series, a new publication series of readings on specific topics. The deadline for nominations is October 31. Visit the EIC search page for more information.

Software Category Editor Needed for Computing Reviews
Computing Reviews, the post-publication review and comment journal of ACM, is seeking a volunteer interested in serving as a category editor for a segment of the software area. Please see the Software Category Editor search page for more information.

Featured Items at acm.org

ACM CEO John White Applauds Computer Science Education Act
Key legislative initiative addresses crisis in K-12 CS education

Call for High-Impact Research IT Proposals
Intent is to generate interest from funding agencies

ACM Releases Enterprise Architecture Tech Pack
Annotated resources provide insights into fast-growing EA discipline and profession

More FCRC 11 Plenary Talks Available on Video
View Talks by Knuth Prize Winner Ravi Kannan and Google’s Luiz Andre Barroso
About SIGDA

About ACM's Special Interest Group on Design Automation

SIGDA is committed to advancing the skills and knowledge of electronic design automation professionals and students throughout the world. We do so in a variety of ways: sponsoring and organizing international workshops, symposia and conferences; leading the way in capturing archival electronic design automation publications on CD-ROM; providing travel grants to sponsored workshops, symposia and conferences; pioneering the maintenance and distribution of electronic design automation benchmarks; hosting university and government researchers for software demonstrations at the University Booth at DAC and ICCAD; publishing the SIGDA Newsletter; maintaining a World Wide Web access site on Internet; and by initiating the new ACM Transactions on Design Automation of Electronic Systems. Highlights of our recent activities include:

- The ACM Transactions on the Design Automation of Electronic Systems. The journal provides comprehensive coverage of innovative research and work concerning the creation and evaluation of VLSI electronic systems. The journal emphasizes a computer science and engineering orientation. Topics include system design, high-level synthesis, logic synthesis, physical layout, design verification, system reliability, and high-performance circuits. The journal is actively seeking research papers, tutorial and survey papers, and short technical notes. The journal is distributed in hard-copy, electronic, and CD-ROM formats. SIGDA members receive a significantly subsidized subscription rate.

- SIGDA sponsors or is in cooperation with major design automation conferences and workshops. We average more than two such events per month. A SIGDA member receives a significantly reduced registration rate for these meetings.

- Our travel grant program provides matching support for SIGDA members in good standing to attend sponsored meetings. For an online application, view the SIGDA Travel Grant page.

- Annual Design Automation Conference (DAC)
  - PhD Forum
  - University Booth
The International Conference on Computer-Aided Design (ICCAD) continues to be the premier and most selective conference devoted to technical innovations in design automation.

Register Now

Technical Program is now available

Conference Schedule
Sunday, November 6:
Monday, November 7:
Tuesday, November 8:
Wednesday, November 9:
Thursday, November 10:

Advance Program PDF ~6.5MB

ACM/SIGDA CADathlon Contest
Awards Presentation/Opening Session
Keynote Address
Technical Sessions
ACM/SIGDA Member Meeting
Technical Sessions
Technical Sessions
Workshops

Keynote Address
Design of Secure Systems - Where are the EDA Tools?
Georg Sigl - Technische Univ. München
As SIGDA Chair...

• I oversee about 16 conferences and symposia, and make sure things get done
• Total cost to put these things on: around $6 million a year
• Somewhere on the order of 20k people attend
• Lots of research papers and publications, awards to give, politics to wrangle
ACM also runs the programming contests

- I’ve been a member of ACM since 1986....
- The year I went to the world finals with my team....
And one last thing professors do -- start companies

• Lots of thinking can result in ideas that are so new that industry really doesn’t understand them. To get the ideas out, many professors start companies.

• Google was a grad student project; faculty members and the University funded it, and got it off the ground

• Many of the faculty here consult, have startup companies, etc., -- the university has a tech transfer office for exactly this sort of thing
To summarize

• A small chunk of my day goes to teaching.

• When I’m drinking coffee and acting like I’m busy, I’m probably actually thinking.