a few random

IMPORTANT QUESTIONS

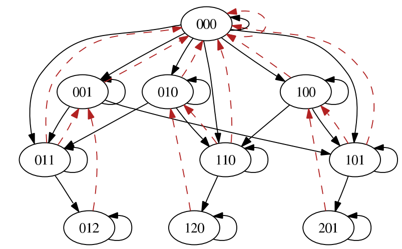
Or, at least, Questions.

whatever

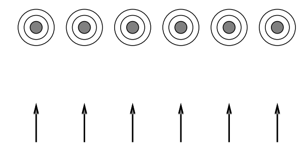
See Prof. Quentin Stout
3605 BBB Building
qstout@umich.edu
www.eecs.umich.edu/~qstout

Can I go now?

Julia Lipman looked into the idle time wasted by processors as they wait for others — this is important in parallel computing. If there are repetitive tasks that each has to do, but variability in the time needed to do them, the idle time can be significant. For example, if the task times are exponentially distributed then p processors all waiting for each other take $\Theta(\log p)$ time per step, but only $\Theta(1)$ if each only depends on a fixed number of others. Here's her Markov chain for analyzing this:



Will we run out of ammo?



n shooters, n targets, every target must be hit. Shots have a fixed probability of missing, independent of others. They shoot in rounds: in each round, the shooters evenly divide up the unhit targets and then shoot.

What is the expected number of rounds before every target has been hit?

How can they do this faster than they can count?

What does this have to do with parallel algorithms for Hamiltonian cycles?

For the answers see the work by Phil MacKenzie. He won the U. Michigan *Best Thesis* award.

Will parallel algorithms melt computers?

There is a supercomputer that has > 1,000,000 cores, and in 7–8 years there will be some with > 100,000,000. Unfortunately, using all of the cores at full power for very long will generate far too much heat.

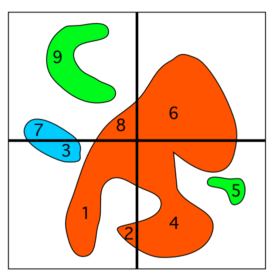
Perhaps the computers will be saved by the fact that people haven't learned how to use so many cores on a single problem, but of course you'll change that.

Can you also solve the energy problem?

Note: some ACAL faculty work on reducing power, as do people building supercomputers and zebra networks

Rats to the rescue?

We have some new parallel algorithms that reduce power by turning off most of the processors at any one time, yet solve the problem nearly as fast as if all were on. They can be thought of as following the movement of rats as they work together to solve a problem.



Naming rights are still available: curiously, many people don't like to think about rats doing algorithms. "Ant algorithms" is already taken. Fluffy bunny algorithms?

Do we have to stay in flatland?

People have considered massively parallel systems of tiny processors in a 2-dimensional grid, perhaps with 100,000 on a single chip and many chips per board. Many problems can be solved in $\Theta(\sqrt{n}) = \Theta($max distance between processors) time, i.e., time governed by the limits of classical physics.
This is faster than is possible on quantum computers.

Moving to a 3-d grid reduces worst-case distance, with the possibility of algorithms taking $\Theta(n^{1/3})$ time. Meanwhile, some people are developing on-chip optical pathways, which you could think of as providing worm-holes to a 2-d grid.

2-d + optics, 3-d, 3-d + optics: Do they really help? Can they be used to solve problems significantly faster and/or use less energy?

Ask Patrick Poon.

The red pill, or the blue one?

We do extensive work in adaptive learning, especially for clinical trials. Rather than standard trials that put half the patients on one treatment and half on the other, you can learn as you go and put more on the better treatment.

This saved many lives in a famous clinical trial here at UM. However, they couldn't do full analysis of the procedures, so Harvard repeated the study using the standard 50/50 allocation, killing several babies unnecessarily.

We help by inventing new algorithms and approaches, implemented on serial and parallel computers. Bob Oehmke was finalist for best paper at Supercomputing.

Note: FDA recently agreed to use more adaptive trials, but noted the need for greater ability to design and analyze the trials.

The halting problem is undecidable?

Not if the Turing machine is very small and the input is a completely blank tape. Rona Machlin determined how long 4-state Busy Beavers could compute and then halt, as opposed to their colleagues that were too exuberant and went on forever. It's the largest Busy Beaver number known. See *The On-Line Encyclopedia of Integer Sequences* to find out what Busy Beavers are, and a reference to her paper.

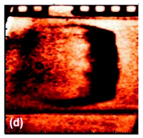
Will that be on the test?

I was an expert witness in a case where a French cellphone company was shut down nationwide for 18 hours because some programmers decided that those fussy rotations on deletions in red-black trees weren't important.

In my expert opinion, that was a dumb thing to do.

Should you believe me?

Our DOE *Center of Excellence in Predictive Science* is modeling an experiment: the world's most powerful civilian laser pushes a small pellet through a tube filled with ionized gas. Where will the pellet will be in 15 nanoseconds? What will the shock wave look like?

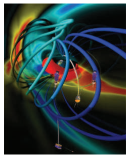


Predicting is quite hard, but there are much harder questions: how uncertain is the prediction? Why do we believe we know the uncertainty? What should we do to reduce the uncertainty?

The computer science aspects include massive parallelization, computational geometry approaches to determining which parameter settings to evaluate, image analysis, ...

What's the climate if the Sun explodes?

We've developed systems that could help determine this. Our Space Weather Modeling Framework is used by NASA and the Air Force Space Command to predict Sun-Earth space weather. It was also used to produce a movie for the American Museum of Natural History.



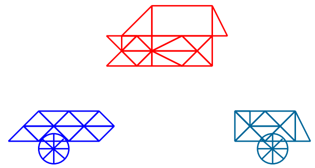
ESMF

The Earth Systems Modeling Framework links together complex models, using hundreds of processors to couple models of the atmosphere, ocean, land, and sea ice into a climate model. NOAA (the national weather forecasting agency) uses ESMF to link together components to make their 4 $\times$ daily forecasts. All commercial weather forecasts are based on the NOAA forecast.

Why does a mustang need a sandwich?

A mathematician gives you two pieces of bread and a slice of ham, but they are only density distributions that might overlap. There is a straight cut that simultaneously slices both pieces of bread, and the ham, in half, so the mathematician and you get equal shares. Unfortunately, the mathematician doesn't know how to find that cut.

Ford's car crash simulations use parallel computers, and need to divide the problem into pieces taking the same amount of time on several aspects. Andy Poe used the Ham Sandwich Theorem to do this better.



With designs like this, it's clear why the auto industry is having problems.

Is Hawaii better than Kansas?

Reiko Tanese studied a parallel distributed memory version of the genetic algorithm. It optimizes faster than the standard serial version. This corresponds to evolution on islands, with occasional migration, vs. evolution in a large homogenous region. The former evolves faster, quickly adapting to changes. Her thesis work has been cited $>$ 1000 times.