COMPUTER SCIENCE COLLOQUIUM

FORTY-FOURTH SERIES SPRING 2016

THURSDAYS AT NOON SALAZAR 2016

FEB.11

Nora Ayanian, Gabilan Assistant Professor of Computer Science, USC Viterbi School of Engineering

MULTIROBOT COORDINATION: FROM HIGH-LEVEL SPECIFICATION TO CORRECT EXECUTION

Using a group of robots in place of a single complex robot to accomplish a task has many benefits, including simplified system repair, less down time, and lower cost. Combining heterogeneous groups of these multi-robot systems allows addressing multiple subtasks in parallel, reducing the time it takes to address many problems, such as search and rescue, reconnaissance, and mine detection. These missions demand different roles for robots, necessitating a strategy for coordinated autonomy while respecting any constraints the environment may impose. Synthesis of control policies for heterogeneous multirobot systems is particularly challenging because of inter-robot constraints such as communication maintenance and collision avoidance, the need to coordinate robots within groups, and the dynamics of individual robots.

FEB.18

John Aycock, University of Calgary, Alberta, Canada

CLASSIC GAMES

Even the simplest old computer games may have required technical miracles to get running at all -- retrogame programmers were constrained by both hardware and software in ways that are unimaginable now. Retrogame archeology looks under the hood of old games to uncover the clever tricks that make them tick. Learn about what retrogame archeology is (and isn't) and how old games are studied today.

FEB.25

Jennifer Chubb Reimann, University of San Francisco

SURVEY OF QUANTUM COMPUTATION & APPLICATIONS

The notion of quantum computing is older than you might think. In 1981, Richard Feynman put forth the idea that computers designed to exploit the principles of quantum physics should be fundamentally different and have more capabilities than classical computers. It was in 1994, when Peter Shor unveiled his algorithm for factoring numbers in polynomial time with a quantum computer that the subject attracted the attention of a large community of researchers. Since then, scores of computer scientists, mathematicians, and physicists have worked on developing the theories of quantum computing, information, logic, and cryptography, and research teams all over the world are racing to implement their discoveries. In this talk, we will survey the history and some applications of the subject, in addition to seeing something about what quantum computing is, and what makes quantum algorithms different from the programs we write every day for our classical machines.

MAR.03

Andru Luvisi, ISO, SSU

THE STORED PROGRAM COMPUTER AND SOME EARLY COMPUTATIONAL ABSTRACTIONS

The stored program computer held instructions and data in the same memory. This enabled programs to manipulate other programs (or even themselves!) and made it possible to decouple the external representation of a program from the actual instructions executed by the machine. This in turn allowed new forms of abstraction to be created and used by implementing them in software. I will discuss the creation of the stored program computer and some early software and hardware ideas related to computational abstraction.

MAR.10

Peter Pacheco, University of San Francisco

OPTIMIZING THE PERFORMANCE OF A CUDA KERNEL

General purpose programs for Graphics Processing Units (GPGPU) have become an extremely important tool in High-Performance Computing. However, because of the novelty of the architecture of GPU's, optimizing the performance of GPGPU's can be very challenging, and it may involve some subtle changes to the "obvious" code. We give a brief introduction to the CUDA API for programming Nvidia GPU's, and then discuss optimization techniques for a simple dot product.

MAR.17

SPRING BREAK (No Colloquium)

MAR.24

Rebecca Hartman-Baker, NERSC Division, Lawrence Berkeley Laboratory

PAST, PRESENT, AND FUTURE PARALLEL PROGRAMMING PARADIGMS

Parallel programming paradigms have been adapted to ever-changing computing resources, needs and goals over the course of history. The evolution of high-end computer architecture (from organic to mechanical to digital and from single-core computers to vector-based, multicore-based, and hybrid CPU/GPGPU-based machines) has necessitated the continuing development of new parallel programming paradigms and numerical algorithms. In this talk, I discuss the development of numerical algorithms throughout the history of numerical computing placed within their historical and architectural contexts, and the implications of future architectures on numerical methods.

MAR.31

CESAR CHAVEZ DAY (No Colloquium)

APR.07

Jason Shankel, CTO, Wildstop

TECHNICAL CONSIDERATIONS FOR VR

Virtual reality headsets are poised to become the next breakthrough in human/computer interface technology. I will outline the unique challenges involved in supporting VR rendering and headset input.

APR.14

Kelly A. Shaw, University of Richmond, Virginia

HELPING SOFTWARE EXPLOIT HARDWARE

In order to gain improved performance and reduced power consumption, computer architects create specialized hardware geared for specific computations. Graphics Processing Units (GPUs) are specialized processors designed for applications with large amounts of regular and parallel computation. While these specialized processors offer performance and power benefits, these advantages come at a cost. These devices are frequently less well understood and more difficult to program than general-purpose processors. In this talk, I present two approaches we created to help GPU software developers. Starchart provides developers with a tool that enables them to systematically and quickly understand how to tune important characteristics of their applications. The second approach, called MRPB, automatically prioritizes and reorders memory accesses to increase the benefits obtained from data caching in GPUs.

APR.21

Landon Curt Noll

HOW TO FIND A NEW LARGEST KNOWN PRIME

The quest to discover a new largest known prime has been ongoing for centuries. Those seeking to break the record for the largest known prime have pushed the bounds of computing. We have come a long way since 1978 when Landon's record breaking 6533-digit prime was discovered (www.isthe.com/chongo/tech/math/prime/m21701.html). Today's largest known prime (www.isthe.com/chongo/tech/math/prime/mersenne.html#largest) is almost 13 million digits long! To encourage the discovery of ever-larger primes, awards of $150,000 and $250,000 are offered (https://www.eff.org/awards/coop) to the first published proof of a discovery of a prime of at least 100 million and 1 billion digits respectively. The search for the largest known prime requires writing and running code that must run to completion, without any errors. Because it takes a very long time to run to completion (several thousand hours in many cases), the code MUST RUN CORRECTLY the very first time! A significant QA effort is required to write 100% error-free code. Moreover considerable effort must be put into fault tolerant coding and recovery from the eventual operating system and hardware errors that will arise. The record goes neither to the fastest coder nor to the person with the fastest hardware but rather to the first result that is proven to be correct. How are these large primes discovered? What are some of the best ways to find a new world record-sized prime number? These and other prime questions will be explored. We will examine software and hardware based approaches and will look at code fragments and hardware machine state diagrams. NOTE: Knowledge of advanced mathematics is NOT required for this talk.

APR.28

Anagha Kulkarni, Computer Science Department, San Francisco State University

EFFICIENT AND EFFECTIVE LARGE-SCALE TEXTUAL SEARCH

The traditional search solutions for large datasets assume access to practically unlimited computational resources, and thus cannot be employed by small-scale organizations. Our work introduces Selective Search, a new retrieval approach that processes large volumes of data efficiently and effectively in computationally constrained environments. To achieve this, Selective Search, partitions the dataset into subsets (shards) in such a way that at query execution time only a few selected shards need to be searched for a query. The dataset is divided into shards based on the similarity of the documents, thus creating topically homogenous partitions (e.g. politics, sports, technology, and finance). This topic-based organization of the dataset concentrates the relevant documents for a query into a few shards. During query evaluation a few shards, that are likely to contain the relevant documents for the query, are identified and searched. Empirical evaluation using some of the largest available datasets (e.g. half a billion web pages) demonstrates that Selective Search reduces search costs dramatically without degrading search effectiveness, and operationalizes this using very few computational resources.

MAY.05

PIZZA DURING PRESENTATIONS IN SALAZAR 2016

STUDENT PRESENTATIONS / SHORT PRESENTATIONS OF RESEARCH CARRIED OUT BY SONOMA STATE COMPUTER SCIENCE STUDENTS

MAY.12

PIZZA DURING CELEBRATION IN SALAZAR 2016

END OF SEMESTER CELEBRATION / AWARDS PRESENTED TO SONOMA STATE COMPUTER SCIENCE MAJORS

Special guest: Eric Levinson, Distinguished SSU CS alumnus, will describe his high-tech career.

SONOMA STATE UNIVERSITY

Supported by the SSU Instructionally Related Activities Fund
and the generous donations of friends of SSU Computer Science Department.

Computer Science Department, Sonoma State University, Rohnert Park, CA 94928
(707) 664-2667
http://www.cs.sonoma.edu

Parking is usually available in lots "E" and "F" and costs $5.00