Computer Science Department Colloquium

CS Faculty Rapid-Fire Presentations of Current Research | Group A

Sanjay Rajopadhye, Louis-Noel Pouchet, Hamid Chitsaz, Yashwant Malaiya, Sudipto Ghosh, Shrideep Pallickara, and Sangmi Pallickara,

Department of Computer Science, Colorado State University

Presenters

- Sanjay Rajopadhye
- Louis-Noel Pouchet
- Hamid Chitsaz
- Yashwant Malaiya
- Sudipto Ghosh
- Sangmi Pallickara
- Shrideep Pallickara

Upcoming Talks by CS Faculty

Dr. Francisco Ortega
September 24th

Dr. Craig Partridge
November 26th

Dr. Vinayak Prabhu
TBA

Colorado State University

Rajopadhye research interests

MÉLANGE

Sanjay Rajopadhye
Colorado State University

Research Interests

Systems Research

- Automatic Parallelization & Algorithms
- Functional/Equational Programming Languages
- Embedded Systems/Architecture
- High Performance Computing

Also Applications

What is MÉLANGE

- Models
  - Performance models
  - Programming models
  - MDE
- Equations
  - Equational programming
- Languages
- Algorithms
- NGE: Next Generation Executables
  - GPUs accelerators, FPGAs
  - Multi- and many- core processors
- Weekly meeting Wednesday 10–11 CSB 315

Foundation: polyhedral model

- Mathematical framework for progrmas
  - Abstraction of loop programs
  - Compute-intensive applications
- Quantitative measures of cost
  - Leads to mathematical optimization
- Closure properties under program transformations
- Workshop on polyhedral compilation techniques http://impact.gforge.inria.fr/impact2016
- Spring Schools (2013, 2016) http://labexcompilation.ens-lyon.fr/polyhedral-school, https://mathsinfohpc.sciencesconf.org
- Keynote at WOLFHPC workshop at Supercomputing 2018

Killer App: PARRIC

- RNA-RNA Interaction (RRI) is an important scientific challenge
  - Potential cure for cancer
- Computational models of RRI (piRNA, BPMax, BPMaxW) are important but very expensive in
  - Time: $\theta(N^3M^3)$ for lengths M and N
  - Space: $\theta(N^2M^2)$
  - We want 100-million-fold speedup

Colorado State University 5

Need for Speed

- piRNA is slow and a memory hog
  - For two sequences of length 100 each, piRNA takes 3.5 hours on a 64-core machine with 512GB RAM
  - Cannot handle sequences longer than 200 (N*M > 40k)
    - Machine goes unresponsive must be rebooted
- For whole genome analysis:
  - 30,000 genes (~2k length)
  - 500 "interesting" small RNAs (length ~100)
    - i.e., 15 million calls to piRNA
  - Each call to piRNA would take 3.5*8000 hrs
  - On an 8 Terabyte machine
  - 15 million calls would take 50 million years
  - We will do it in six months on 100 department machines

Colorado State University 6

How to get there

- Easy parallelization (use 10^6 "large enough" machines on the cloud) is too expensive
- Make piRNA run efficiently on small RAM machines (e.g., 16GB)
- Speed it up on 100 machines in the department
  - Need 10^6-fold speedup on each machine
    - 1000-fold by using locality/parallelism/vectorization
    - 1000-fold by filtering on only 0.1% interesting pairs of sequences
    - Still needs 1000-fold speedup of filtering program

Colorado State University 7

WE can get there

Showed 100-fold speedup of "miniapp" called $(OSP)^2$

- 100-times simpler than piRNA
- On small (fits in RAM) problem sizes
- We have the expertise to do this on multi-cores & also GPUs

Colorado State University 8

Polyhedral Challenges

- piRNA is beyond current tools (few tens of lines to kLoC)
- Multilevel tiling: virtual memory, DRAM, caches, (and 2-levels of parallelism: cores &vector units)
- Legality of tiling: all six dimensions must be tiled: is that legal?
  - GKT: middle serialization (known since 1979, still not fully automatic)
- Raise the level of abstraction:
  - Sloppy Equations: Hamid should write ~100 eqns, not kLoC in C++
- Simultaneously schedule & tile reductions
  - On $(OSP)^2$ PLuTO slows down the program (mostly)

Compilers and High-Performance Computing Lab

Louis-Noël Pouchet
pouchet@colostate.edu
Colorado State University
August 27th, 2018

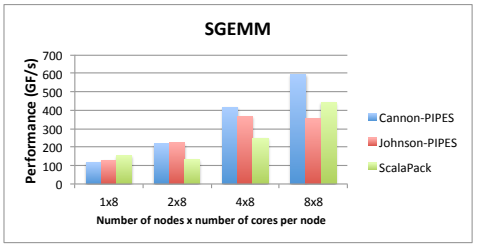
What is Optimizing Compilation?

Main idea: ask a computer to find an equivalent program which executes faster than your own program

- Must preserve the program semantics, must exploit parallel/distributed architectures, etc.
- Multiple disciplines are leveraged: algorithmic, programming, architecture, mathematics, machine learning, experimental computer science, etc.

- **A compelling example: programming distributed systems (PIPES)**

```
1  Parameter N, P;
2  // Define data collections
3  [float* A:1..N,1..N];
4  ...
5  // Task prescriptions
6  env :: (MM:1..N,1..N,1..N);
7  // Input/Output:
8  env -> [A:1..N,1..N];
9  ...
10 [C:1..N,1..N,N] -> env;
11 // Task dataflow
12 [A:i,k],[B:k,j],[C:i,j,k] -> (MM:i,j,k) -> [C:i,j,k+1];
13 Topology Proc = Topo2D(P,P);
14 // Place the N tasks (i,j,*) to Proc((i/8)%P,(j/8)%P)
15 (MM:i,j,1..N)@Proc((i/8)%P,(j/8)%P);
16 // Circular communication pattern for Cannon algorithm
17 [A:i,k]@(MM:i,j,k) => (MM:i,(j-1)%P,k+1);
18 [B:k,j]@(MM:i,j,k) => (MM:(i-1)%P,j,k+1);
```



- Input: 20 lines, nearly identical to textbook, compiler generates 2000+ lines of code!

M. Kong, L.-N. Pouchet, P. Sadayappan, V. Sarkar. "PIPES: A Language and Compiler for Task-based Programming on Distributed-Memory Clusters", to appear in IEEE/ACM Supercomputing (SC'16), Nov. 2016.

2

Opportunities in Computing

- **Many applications are driven by massive computation needs**
  - Modern physics, computer-aided design (CAD), deep learning, etc.
  - Healthcare tomorrow, especially with genomics-related work?
- **Compilers are one of the cornerstone of modern computing**
  - The pace of new architecture development, and the complexity and specificity of the code needed makes programming-by-computers necessary
  - High impact by enabling new applications
    - Examples: in-situ lung tumor detection, deep learning
- **So, why study compilers?**
  - **A great way to learn many different topics: programming, algorithmic, hardware architecture, mathematics, etc.**
  - **A special opportunity to interact with pluri-disciplinary teams, and learn a domain**
  - **Compiler researchers/engineers are in high demand in industry**

3

Some Research Opportunities [1/2]

Many topics possible, come talk with me! Office: CS346

Projects/research can be tailored to your skills / expectations, several funding opportunities

1. **Polyhedral compilation** 
   - Design program equivalence classes, build new optimization algorithms
   - Multi-platform optimization (CPU/GPU/FPGA/SoC, collaborations with Intel and UCLA)
2. **Machine learning (ML)** 
   - Use ML inside compilers to find better implementations of algorithms
   - Optimize ML applications (Tensors for deep learning, collaboration with Facebook AI)
3. **Hardware/software co-design** 
   - Hardware/software partitioning: choosing whether to use the CPU, GPU, or programmable IP
   - Algorithms for efficient FPGA synthesis (using Vivado HLS, collaboration with UCLA)
4. **High-performance computing for Exascale science** 
   - Optimization framework for in-node computing, deploy PolyOpt on DoE applications
   - Analyze and optimize RAJA applications (collaboration with LLNL)

4

Some Research Opportunities [2/2]

Many topics possible, come talk with me! Office: CS346

Projects/research can be tailored to your skills / expectations, several funding opportunities

5. **GPU acceleration of graph analytics** 
   - Performance and energy optimization of sparse linear algebra computations
   - ML models to select best implementation for a particular graph (collab. with OSU and UCSD)
6. **Performance modeling** 
   - Build analytical models to predict the performance of a program, without running it
   - Use ML to build more accurate performance models (collaboration with INRIA, France)
7. **Macro-dataflow and task-based programming** 
   - Design new compilation algorithms for shared-memory and distributed-memory clusters
   - Runtime resource allocation and task placement (collaboration with Rice University)
8. **- - - Put your idea here ☺ - - -**
   - If you have a research idea, come discuss it with me!

5

Yashwant K. Malaiya

- Professor Computer Science
  - Teach Fault Tolerant Computing CS 530 on-campus/on-line
- Research areas
  - Fault modeling
  - Reliability/Risk
  - Testing and testable design
  - Quantitative security risk evaluation
  - Human/economic/social factors
- Field contributions
  - IEEE Third Millennium Medal, IEEE Computer Society Golden Core award.
  - 200 publications, advised 65 graduate students, served on 250 graduate committees.

Colorado State University
Knowledge to Go Places

Research Approach

- Multiple perspectives, new and unusual approaches
- Contributions: concepts introduced include
  - Alhazmi-Malaiya Logistic Vulnerability Discovery Model
    - Used actual data for OSs, browsers, web-servers etc
    - Modeling long-term trends, cyclicity, underlying causes
  - Test coverage – defects relationship model
    - Relationship among test effectiveness, defects and reliability
  - Antirandom testing
    - Apply tests as different as possible ("*cannot be done*")
  - Detectability Profile
    - No such thing as an average fault

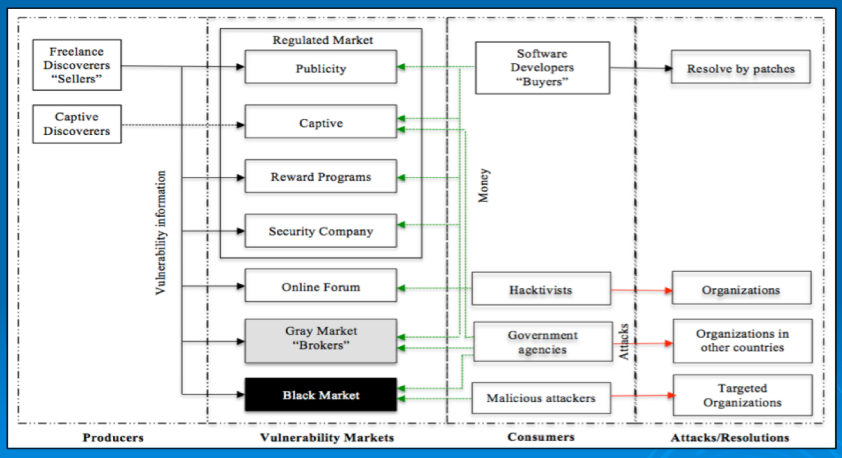
Colorado State University
Knowledge to Go Places

Some of the Research Interests

- Vulnerability discovery in evolving software
  - Modeling evolution and inheritance of defects
  - Zero day vulnerabilities
- Predictability of exploitation & Risk evaluation
  - Breach probability: regularly occurring or rare events?
- Money flow in vulnerability markets
  - Modeling participants and prices/rewards
- Assessing cost of a breach:
  - corrective/preventive actions, price paid by organizations and society (insurance)

Colorado State University
Knowledge to Go Places

Vulnerability Markets



Colorado State University
Knowledge to Go Places

Information/Contact

- Contact:
  - 356 CSB,
  - email Malaiya@cs.colostate.edu
- Recent research results:
  - Website: Yashwant Malaiya > Publications,
  - Google Scholar
- Recent students
  - A. M. Algarni, A. Younis, O.H. Alhazmi, H. Joh, S.-W. Woo ….

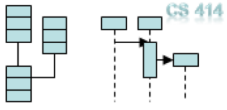
Colorado State University
Knowledge to Go Places

My Teaching and Research

SUDIPTO GHOSH
PROFESSOR OF COMPUTER SCIENCE
Sudipto.Ghosh@ColoState.EDU
http://www.cs.colostate.edu/~ghosh

Recent Courses for Grad Students

Spring 2018	CS414 – Object Oriented Design
Fall 2017	CS514 – Software Product and Process Evaluation
Spring 2017	CS580A5 – Software Testing and Analysis
Various	CS793 – Software Engineering Group



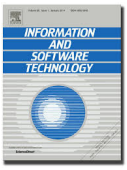
Recent Activities

Associate Editor:

- IEEE Transactions on Reliability
- Journal of Software Testing, Verification, and Reliability
- Software Quality Journal

Editorial Board:

- Information and Software Technology



Conference Program Co-Chair

IEEE International Symposium on Software Reliability Engineering, 2018

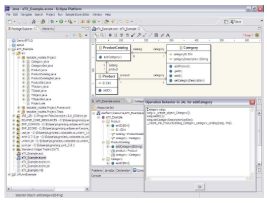
Core Research in Software Engineering

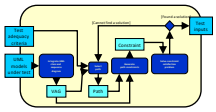
Modeling software

- Precise specification of structural and behavioral properties
- Composition of models
- International repository of software models (ReMoDD)
- Funding source: NSF

Testing and verification

- Regression testing
- Model-based test generation
- Mutation testing
- Fault localization
- Past funding: IBM, Qwest, NSF





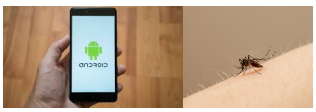
Interdisciplinary Research

Chemical and Biological Engineering

- Population Physiologically based Pharmacokinetic modeling
- Funding: FDA

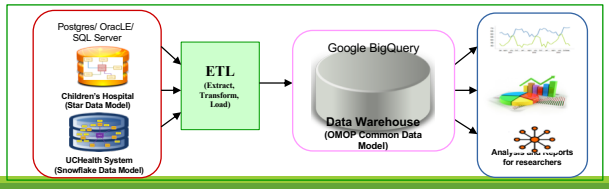
Infectious Diseases

- Mobile data collection for Dengue Decision Support System
- Funding: NIH



Anschutz Medical Center

- Testing ETL transforms
- Funding: CU Denver

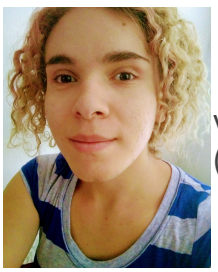


Current Students

Mohammed Al-Refai
(PhD Candidate)

Hajar Homayoni
(PhD Student)

Erica Shin
(Graduate Student)



Vidya Gaddy
(Undergraduate Student)

1

BIG DATA & CLOUD COMPUTING

SANGMI LEE PALLICKARA
SHRIDEEP PALLICKARA
Computer Science Department
Colorado State University

August 26, 2018

Research Overview

2

- Voluminous data management
  - Number of files and data packets can be very high
- Real time: analytics, stream processing, and query evaluations
- **DOMAINS:** Epidemiology, geosciences, environmental science, healthcare, and IoT
- **CURRENT FUNDING SOURCES:**
  - National Science Foundation [2]
  - Department of Homeland Security
  - Advanced Research Projects Agency-E (ARPA-E)

August 26, 2018

GALILEO: Managing multidimensional time series data

3

- High throughput storage and retrieval of observations
  - Support for large number (~10^{11}) of small files
  - Petascale datasets
- Query support: Range queries, analytic queries, approximate queries, and probabilistic queries

M. Malensek, S. L. Pallickara, and S. Pallickara. Fast, Ad Hoc Query Evaluations over Multidimensional Geospatial Datasets. *IEEE Transactions on Cloud Computing*. Vol. 5(1) pp 28-42. 2017.

M. Malensek, S. L. Pallickara, and S. Pallickara. Analytic Queries over Geospatial Time-Series Data using Distributed Hash Tables. *IEEE Transactions on Knowledge and Data Engineering*. Vol 28(6) pp 1408-1422. 2016.

Tracking Methane Gas Leaks using Google Streetview Cars

4

- Joint effort with Environmental Defense Fund, Google, and Biology
- Process spatiotemporal mobile sensing data
  - Hosted by Google Street view cars
- Collaboration with Google EarthEngine

J. Arulswamy and S. L. Pallickara, Columbus: Enabling Scalable Scientific Workflows for Fast Evolving Spatio-Temporal Sensor Data. Proceedings of the 14th IEEE I nternational Conference of Service Computing (IEEE SCC), Honolulu, Hawaii, USA

August 26, 2018

NEPTUNE: High throughput stream processing for Internet of Things

5

- Online scheduling of streams in the presence of resource uncertainty
- Refinements for high-throughput
  - Interference alleviation
  - Application buffering
  - Backpressure for flow control
  - Entropy-based dynamic message compactions

T. Buddhika, R. Stern, K. Lindburg, K. Ericson, and S. Pallickara. Online Scheduling and Interference Alleviation for Low-latency, High-throughput Processing of Data Streams. *IEEE Transactions on Parallel and Distributed Systems*. Vol. 28(12) pp 3553-3569. 2017.

August 26, 2018

SYMPHONY: Exploring consequences of disease outbreaks and vaccination strategies

6

- Analytics of voluminous epidemiological data
  - Statistical, machine learning, and ensemble methods to build analytical models
- Economic consequences
- Planning exercises
  - Real-time analytics and visualizations
- Scale: Manage over a trillion files

W. Budgaga, M. Malensek, S. L. Pallickara, N. Harvey, J. Breidt, and S. Pallickara. Predictive Analytics Using Statistical, Learning, and Ensemble Methods to Support Real-Time Exploration of Discrete Event Simulations. Future Generation Computer Systems. Elsevier. Vol 56, Pages 360–374. 2016.

From the Center to the Edges, Like Ripples in a Pond

- Cloud Computing

  W. Lloyd, S. Pallickara, et al. Demystifying the Clouds: Harnessing Resource Utilization Models for Cost Effective Infrastructure Alternatives. *IEEE Transactions on Cloud Computing*. Vol. 5(4) pp 667-680. 2017.

- Edge Computing

  M. Malensek, S. L. Pallickara, and S. Pallickara. Hermes: Federating Fog and Cloud Nodes to Support Query Evaluations in Continuous Sensing Environments. *IEEE Cloud Computing*. Vol. 4(2) pp 54-62. 2017.

- Distributed Generation of Sketches

  T. Buddhika, M. Malensek, S. L. Pallickara, and S. Pallickara. Synopsis: A Distributed Sketch over Voluminous Spatiotemporal Observational Streams. *IEEE Transactions on Knowledge and Data Engineering*. Vol. 29(11) pp 2552-2566. 2017.