Erdal Mutlu

P.O. Box 999
Richland, WA 99352

erdal.mutlu@pnnl.gov
www.erdalmutlu.com
+1 (509) 460-2869

Research Interests

My research focuses on solving programmability and correctness challenges in "High Performance Computing" software developed for modern architectures. My main expertise is on designing high level representations and optimization frameworks for scientific computations, i.e., developing new mathematical notation for tensor algebra, designing domain specific languages and compiler systems for modern HPC architectures.

Programming Languages: compiler design and implementation, domain specific languages.

Software Engineering and Reliability: runtime systems, dynamic/static verification techniques and tools for asynchronous, parallel and distributed programming models.

Education

Koç University, İstanbul, TURKEY
Ph.D. in Computer Science and Engineering, September 2016
Dissertation title: Race Detection Techniques for Applications using Asynchronous Programming Models
Advisor: Serdar Taşıran
Description: We worked on developing race detection techniques for parallel programming models specifically asynchronous JavaScript applications and task based high performance computing (HPC) applications using dataflow constructs along with shared memory programming models. For testing our detection techniques, we manually instrumented Firefox web browser for collecting execution information from real-world applications. We also implemented a randomized exploration scheduler for Atomic DataFlow framework from Barcelona Supercomputing Center.

Sabancı University, İstanbul, TURKEY
M.Sc. in Computer Science and Engineering, February 2011
Thesis title: High Level Rule Modeling Language for Airline Crew Pairing
Advisor: Hüsnü Yenigün
Description: We designed a domain specific programming language (DSL) for representing rule based feasibility controls and cost calculations in crew pairing systems. Later, we designed and implemented a compiler for this DSL using Flex(scanner), Bison(parser) and C++ programming language as the backend code generation. Our framework generates C++ libraries for each rule specified with our DSL which then used by crew scheduling optimization framework at runtime.

Sabancı University, İstanbul, TURKEY
B.Sc. in Computer Science and Engineering, February 2008

Research Experience

Tensor Algebra for many-body methods (TAMM) **2018–present**

We are developing a runtime system specialized for efficient tensor operations heavily used in computational chemistry applications for exascale high performance computing (HPC) systems. Framework is developed as part of NWChemEx project which focuses on tackling the challenges in methods developed for exascale systems.

Race Detection for Hybrid HPC Applications **2015–2016**

We developed race detection techniques for applications using hybrid programming models combining data-flow constructs with shared memory programming models. We implemented a prototype tool for Atomic Dataflow programming model by instrumenting the framework using C++ programming language.

Race Detection for JavaScript Web Applications **2014–2016**

We introduced a race detection technique for JavaScript web applications using asynchronous constructs(user interaction, server requests). We instrumented Firefox web browser (over 20M lines of codes) for collecting traces using C++ programming language and applied our race detection technique on real world web applications detecting various harmful races.

Behavior Exploration Techniques for Distributed Data Types **2014**

We investigated different data types used in distributed systems (Conflict-free Replicated Data Types (CRDTs), Cloud Types etc.) and formalized techniques for exploring s possible behaviors of different interactions between client and server systematically.

Randomized Schedule Exploration for Hybrid Applications **2013**

We presented a dynamic verification technique for a class of concurrent programming models that combine data-flow and shared memory programming models. We identified and illustrated a novel category of bugs in these hybrid concurrency programming models and provide a technique for randomized exploration of program behaviors in this setting. Implemented a randomized exploration technique into ADF framework using C++ programming language.

Face Detection Optimization for System-on-Chip **2012**

We developed and implemented an optimized scheme for calculating the integral image and applying the face detection which optimizes the memory usage so that it can be used in SecSoC, which targets to achieve video surveillance application on energy optimized microprocessors.

High Level Rule Modeling Language for Airline Crew Pairing **2009–2010**

We designed a domain specific language and implemented a compiler generating C++ libraries for rule based feasibility controls and cost calculations in crew pairing systems. We generate run time methods for the crew pairing systems where end-users are able to change the feasibility rules without interacting with the crew pairing engine itself.

Work Experience

Computer Scientist — Pacific Northwest National Laboratory
Richland, WA, USA — Dec 2019–present

Post Doctorate Research Associate — Pacific Northwest National Laboratory
Richland, WA, USA — Jan 2018–Dec 2019
Worked in development of Tensor Algebra for Many-body Methods (TAMM) runtime systems for efficient tensor operations heavily used in computational chemistry applications.

Visiting Researcher — Barcelona Supercomputing Center
Barcelona, SPAIN — Sept 2013–Dec 2013
Developed behavior exploration techniques for a hybrid programming model "Atomic DataFlow(ADF)" which combines "Transactional Memory" concurrent programming models with data-flow constructs. Implemented a testing scheduler into ADF framework using C++ programming language.

Ph.D. Intern — ST-Microelectronics
Milan, ITALY — Aug 2012–Dec 2012
Worked on optimizing and parallelizing Viola-Jones face detection algorithm for a new prototype system-on-chip, SecSoC, which targets to achieve video surveillance application on energy optimized microprocessors. Implemented the algorithm using C programming language over FPGA simulator.

Researcher - Software Engineer — The Scientific and Technological Research Council of Turkey
Kocaeli, TURKEY — May 2010–Sept 2011
Worked in IYON (Forensic Image Enhancement and Restoration Software) project on developing plug-in manager and custom filter generator modules. Implemented various modules using C++ programming language with Qt and Boost libraries.

GSM/UMTS Global Product Support Engineer — Nortel-Netas
İstanbul, TURKEY — Feb 2008–Aug 2008
Worked as part time product support engineer for Nortel's GSM/UMTS products on investigating and solving reported issues on the software stack.

Computer Skills

Languages: Advanced knowledge of C/C++(11-14-1z)/C#, Java, JavaScript; Basic knowledge of OCaml, Python.
Parallel Programming Models: OpenMP, MPI, CUDA, Intel - TBB, Microsoft-TPL
Development Tools: Eclipse, Visual Studio, WebStorm, Git

Honors and Awards

Microsoft Research invitation to Summer School in Cambridge, UK	**2015**
EuroTM Short Term Scientific Mission (STSM) Scholarship	**2013**
EuroTM travel award to attend DMTM	**2013**
EuroTM financial support award to attend HTDC Winter School	**2013**
HiPEAC Industrial Ph.D. Internship Scholarship	**2012**

Publications

LCPC'20 – Erdal Mutlu, Ruiqin Tian, Bin Ren, Sriram Krishnamoorthy, Roberto Gioiosa, Jacques Pienaar, and Gokcen Kestor. COMET: A domain-specific compilation of high-performance computational chemistry. Workshop on Languages and Compilers for Parallel Computing (LCPC), 2020

ARRAY'19 – Erdal Mutlu, Karol Kowalski, and Sriram Krishnamoorthy. Toward generalized tensor algebra for ab initio quantum chemistry methods. In *Proceedings of the 6th ACM SIGPLAN International Workshop on Libraries, Languages and Compilers for Array Programming*, ARRAY 2019, page 46–56, New York, NY, USA, 2019. Association for Computing Machinery

Correctness'18 – E. Mutlu, A. Panyala, and S. Krishnamoorthy. HPC software verification in action: A case study with tensor transposition. In *2018 IEEE/ACM 2nd International Workshop on Software Correctness for HPC Applications (Correctness)*, pages 9–16, Nov 2018

PARCO'18 – Hassan Salehe Matar, Erdal Mutlu, Serdar Tasiran, and Didem Unat. Output nondeterminism detection for programming models combining dataflow with shared memory. *Parallel Computing*, 71:42 – 57, 2018

JPF'17 – Maryam Dabaghchian, Zvonimir Rakamarić, Burcu K. Ozkan, Erdal Mutlu, and Serdar Tasiran. Consistency-aware scheduling for weakly consistent programs. *ACM SIGSOFT Software Engineering Notes*, 42(4):1–5, October 2017. Proceedings of the 2017 Java Pathfinder Workshop (JPF)

FSE'15 – **Erdal Mutlu**, Serdar Tasiran, and Benjamin Livshits. Detecting JavaScript races that matter. In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*, ESEC/FSE 2015, pages 381–392, New York, NY, USA, 2015. ACM

RV'14 – **Erdal Mutlu**, Vladimir Gajinov, Adrián Cristal, Serdar Tasiran, and Osman S. Unsal. *Runtime Verification: 5th International Conference, RV 2014, Toronto, ON, Canada, September 22-25, 2014. Proceedings*, chapter Dynamic Verification for Hybrid Concurrent Programming Models, pages 156–161. Springer International Publishing, Cham, 2014

DYLA@PLDI'14 – **Erdal Mutlu**, Serdar Tasiran, and Benjamin Livshits. I know it when I see it: Observable races in javascript applications. In *Proceedings of the Workshop on Dynamic Languages and Applications*, Dyla'14, pages 1:1–1:7, New York, NY, USA, 2014. ACM

PaPEC'14 – Burcu Kulahcioglu Ozkan, **Erdal Mutlu**, and Serdar Tasiran. Towards verifying eventually consistent applications. In *Proceedings of the First Workshop on Principles and Practice of Eventual Consistency*, PaPEC '14, pages 11:1–11:4, New York, NY, USA, 2014. ACM

SCLIT'11 – **Erdal Mutlu**, Ilker Birbil, Kerem Bulbul, and Husnu Yenigun. High level rule modeling language for airline crew pairing. In *International Conference on Numerical Analysis and Applied Mathematics*, 2011