Ian Bertolacci

SOFTWARE AND PERFORMANCE GRADUATE RESEARCH ENGINEER

✉ ian.bertolacci@gmail.com | 🏠 ian-bertolacci.github.io | ⓞ ian-bertolacci | in ianbertolacci | 🐦 @ianbertolacci

Education

University of Arizona   *Tucson, AZ*
- MASTERS OF SCIENCE IN COMPUTER SCIENCE   *May 2020*

GPA: 3.8 / 4.0

Colorado State University   *Ft. Collins, CS*
- BACHELOR OF SCIENCE IN COMPUTER SCIENCE
- BACHELOR OF SCIENCE IN PSYCHOLOGY: MIND, BRAIN, AND BEHAVIOR   *May 2016*
- BACHELOR OF SCIENCE IN APPLIED COMPUTING TECHNOLOGY: HUMAN CENTERED COMPUTING

GPA: 3.337 / 4.0

Skills

Developer Skills	Git and GitHub, Linux, Parallel programming, Legacy applications, Performance profiling and analysis, Code transformation and generation, Parsers and regular expressions, Agile development
Languages	Python, Bash, C, Chapel, C++, Java, CUDA
Libraries and Frameworks	MultiProcessing, OpenMP, OpenCL, MPI, ROSE, LLVM, ZeroMQ, GTest
Build and Packaging Systems	Make, CMake, AutoTools, Module, Docker
Non-Technical Skills	Public speaking, Technical and scientific writing, Teaching, Statistical analysis

Experience

Workday   *Pleasanton, CA / Remote*

SENIOR ASSOCIATE SOFTWARE DEVELOPMENT ENGINEER- DISTRIBUTED SYSTEMS   *September 2020 - Current*

University of Arizona   *Tucson, AZ*

SCIENTIFIC ANALYST 1   *May 2020 - August 2020*
- Conduct performance experiments and analysis of scientific applications
- Assist in the development and improvement of scientific applications
- Assist in the development of scientific data-processing infrastructure

University of Arizona   *Tucson, AZ*

GRADUATE RESEARCH ASSISTANT   *August 2016 - May 2020*
- Conducted and published research on methods in inter-loop optimization specified using extensions to OpenMP language
- Investigated performance of existing scientific application and propose changes to improved efficiency
- Planed API changes to provide path for shared-memory parallelization and automated inter-loop optimization of existing application
- Collaborated with interdisciplinary and cross-institutional research team
- Worked with team exploring methods of extracting parallelizable loops in Python applications using dynamic analysis
- Mentored undergraduate student in developing parallel scientific benchmarks as part of their undergraduate honors thesis

Cray Incorporated   *Seattle, WA*

SOFTWARE ENGINEER INTERN   *June 2018 - August 2018*
- Extended Chapel's compressed sparse-array data structure
- Explored a developer-friendly refactor to Chapel's Domain Standard Interface
- Developed distributed matrix toposorting benchmark exploring different distributed work queuing strategies

Colorado State University   *Ft. Collins, CO*

UNDERGRADUATE RESEARCH ASSISTANT   *May 2014 - August 2016*
- Conducted and published research on methods of hiding time-tiling loop-optimizations using existing programming language features
- Developed benchmarks testing loop-optimizations in C and Chapel
- Conducted performance experiments

Cray Incorporated   *Seattle, WA*

SOFTWARE ENGINEER INTERN   *June 2015 - August 2015*

- Implemented Chapel Linear Algebra Package interface module using custom automated C/Fortran-to-Chapel interface translation tool
- Developed Chapel programming language tutorial (`learnxinyminutes.com/docs/chapel`)

Colorado State University   *Ft. Collins, CO*

UNDERGRADUATE TEACHING ASSISTANT   *August 2012 - May 2014*

- Developed and lead hands-on instructional sessions in computer-lab setting

Projects

LowFlow Mini-App

`github.com/ian-bertolacci/LowFlow_MiniApp`

- Scientific "mini-app" using computationally intensive loops extracted from ParFlow watershed hydrodynamics application
- Explores methods of parallelizing loops using different frameworks, including OpenMP, OpenCL, and CUDA
- Explores methods of hiding details of parallelization using custom preprocessor macro programming language
- Custom CMake framework for integrating different source components and enabling different profiling code at compile-time

LoopChain

`github.com/CompOpt4Apps/LoopChainToolDemo`

`github.com/CompOpt4Apps/LoopChainIR`

`github.com/ian-bertolacci/ISL_To_Sage`

- Source-to-source compiler based on ROSE implementing extensions to OpenMP providing inter-loop optimizations
- C++ library for representing loop sequences and data accesses, potimizing with inter-loop transformations, and generating new C/C++ code
- C++ library for converting from Integer Set Library's C AST format to ROSE's SAGE AST format
- Uses integer linear programming method of determining smallest shift extents to enable legal loop fusion
- Custom test framework for testing legality and correctness of loops generated by the transformation library

Cellular Automata Simulator

`github.com/ian-bertolacci/Cellular-Automata-Simulator`

- Personal summer project in Java diving into compilers, virtual machines, programming languages, graphics, and cellular automata
- Developed programming language and compiler to describe cellular automata rules
- Custom virtual machine and byte-code
- Capable of executing arbitrary cellular automata rules of any dimensionality

Publications

- I. Bertolacci, M. M. Strout, B. R. de Supinski, T. R. W. Scogland, E. C. Davis, and C. Olschanowsky. Extending OpenMP to Facilitate Loop Optimization. In B. R. de Supinski, P. Valero-Lara, X. Martorell, S. Mateo Bellido, and J. Labarta, editors, *Evolving OpenMP for Evolving Architectures*, volume 11128, pages 53–65. Springer International Publishing, 2018.

- I. J. Bertolacci, C. Olschanowsky, B. Harshbarger, B. L. Chamberlain, D. G. Wonnacott, and M. M. Strout. Parameterized Diamond Tiling for Stencil Computations with Chapel Parallel Iterators. In *Proceedings of the 29th ACM on International Conference on Supercomputing*, ICS '15, pages 197–206. ACM, 2015.

- I. J. Bertolacci, M. M. Strout, S. Guzik, J. Riley, and C. Olschanowsky. Identifying and Scheduling Loop Chains Using Directives. In *2016 Third Workshop on Accelerator Programming Using Directives (WACCPD)*, pages 57–67, 2016.

- I. J. Bertolacci, M. M. Strout, J. Riley, S. M. Guzik, E. C. Davis, and C. Olschanowsky. Using the loop chain abstraction to schedule across loops in existing code. *International Journal of High Performance Computing and Networking*, 13(1):86–104, 2018.

Awards

Spring 2019	**Graduate Service Award**, College of Science and Department of Computer Science	*University of Arizona*
Fall 2015	**Deans List**, College of Natural Science	*Colorado State University*
November 2014	**3rd Place Undergraduate Research Poster**, ACM Research Competition	*SuperComputing 2014*
October 2014	**Best Undergraduate Research Poster**	*Rocky Mountain Celebration of Women in Computing*