SHASHI GOWDA

Office: 2-390D, Massachusetts Institute of Technology
Home: 1 Banks St, Somerville, MA 02144
gowda@mit.edu +1 (617) 899 4295

I love building well-rounded software that is simple and pleasurable to use, yet powerful and fast. I have diverse expertise in performance engineering, programming languages and compilers, large scale symbolic systems, distributed computing, analytical databases, and UI development.

EDUCATION

2018-current

***PhD Candidate* in Computational Science and Engineering**
Massachusetts Institute of Technology, Cambridge, MA

Thesis working title: "Staged symbolic-numeric programming". I am building high-performance Symbolic computing capabilities and domain-specific compilers.
Advisor: Prof. Alan Edelman | The Julia Lab GPA: 4.7/5

2010-2014

***B. Tech.*, Information Technology**
National Institute of Technology, Surathkal, India

EXPERIENCE

2016-2018 ***Principal Software Engineer,* Julia Computing Inc.**

Led development of high-performance data science tools at various levels of detail – from text processing to distributed statistics. Built JuliaDB – a pure-julia distributed-memory alternative to pandas(+dask), spark and KDB. Supported clients in deployment of the same in high-throughput production environments.

2014-2016 ***Research Software Engineer,* (Remote) CSAIL, MIT**

Worked on a distributed computing scheduler and a distributed array implementation on top of it. Also developed many tools for pedagogical interactive visualizations which were used in the famous MIT 18.06 (Linear Algebra) course.

Summers 2010, 2011, 2012, 2014 ***Intern,* Google Summer of Code**

Worked on various web-technologies and federated-computing projects under different open source organizations – StatusNet (2010, 2011 *PHP and JS*), Sahana Eden (2012 *Python*), The Julia Language (2014 *Julia and JS*)

PUBLICATIONS

2021 High-performance symbolic-numerics via multiple dispatch

S. Gowda, Y. Ma, A. Cheli, M. Gwóźdź, V.B. Shah, A. Edelman, C. Rackauckas.
ACM Communications in Computer Algebra Vol. 55 (dl.acm.org)

2021 ModelingToolkit: A Composable Graph Transformation System For Equation-Based Modeling

Y. Ma, S. Gowda, R. Anantharaman, C. Laughman, V. Shah, C. Rackauckas.
preprint arXiv:2103.05244

2019 Sparsity Programming: Automated Sparsity-Aware Optimizations in Differentiable Programming

S. Gowda, Y. Ma, V. Churavy, A. Edelman, C. Rackauckas.
NeurIPS Program Transformations for Machine Learning Workshop. (pdf)

SOFTWARE PORTFOLIO

I have created and maintained a number of open source software projects over the years, a longer list of notable projects can be found on my website, shashi.biz

2019-current ***Symbolics.jl* (at MIT)** Computer algebra library with a focus on ease of use, speed, extensibility, multidimensional tensor operations, and high-performance numerical code generation. Centerpiece of my thesis work. It is heavily used in Julia's modeling and simulation ecosystems, and currently has **14 direct and 74 indirect dependents** (juliahub). It is used in fields from drug discovery to ocean climatology.

2016-2018 ***JuliaDB*** A distributed-memory analytical database. Competitive with Spark, pandas (+dask) (more details in our PyData NYC 2017 presentation), and kdb. I built about 80% of the project working at every level, including indexed relational tables (IndexedTables.jl), text parser compiler (TextParse.jl), compression, distributed blob storage (MemPool.jl), parallel scheduling (Dagger.jl), distributed table operations and statistics (JuliaDB.jl). At Julia Computing, I also helped clients deploy JuliaDB in machine learning applications and achieve orders of magnitudes of speedups over systems we replaced.

2014-2016 ***Escher.jl/Interactive UIs*** I built Interact.jl – a domain specific language for interactive explanations in jupyter. Later created a publishable Functional-reactive UI framework for non-UI experts in Julia. Since maintain WebIO – a minimal framework for making UI widgets that work in Jupyter, VS Code, and over the web standalone.

SKILLS

Preferred Tools Julia, Python, C, Scheme, Unix, vim+tmux, git

Engineering skills Software design, performance engineering, programming languages–interpretation and compilation, distributed systems, computer algebra, reactive progamming.

GRADUATE-LEVEL COURSES

Theory of Computation, Numerical Methods, Large Scale Symbolic Systems, Computational Classical Mechanics,Statistical Learning Theory, Autonomy and Decision Making.

TEACHING EXPERIENCE

Spring 2019 & Fall 2020 Introduction to computational thinking (18.S191)

Fall 2018 High-performance computing (18.337)

NOTABLE TALKS

2022 Strange Loop, St Louis. "Symbolic-numeric programming in Julia"

2022 SciMLCon, Boston, MA "Symbolic arrays: past, present and future"

2018 JuliaCon, London, UK "How JuliaDB works"

2019 NeurIPS, Workshop on Programming languages for ML, Vancouver BC "Sparsity aware optimizations in differentiable programming"

2017 PyData NYC, NYC "JuliaDB: A data system for Julia (with Jeff Bezanson and Josh Day)"

2016 Microsoft, Bangalore "Keynote: A Functional Algebra of UIs"

2015 JuliaCon, Boston, MA "Escher.jl-a new way to make and deploy UIs"

February 10, 2023