Dept. of Computer Science - MS 132
Rice University
P.O. Box 1892 Houston, TX 77251
☎ *+1 (214) 872 0458*
✉ *shams@rice.edu*
🖱 *http://shams.web.rice.edu/*
`http://www.linkedin.com/in/shamsm`

Shams Imam

Objective Obtaining a research or advanced technology position that leverages my expertise in parallel programming models and runtime systems, while addressing new challenges in these areas in an effective manner.

Education

2009–Current **Master of Computer Science + Doctor of Philosophy**, *Rice University*, Houston, TX, USA.
Major: Computer Science. GPA: 4.0/4.0. Master's defended in December 2011. PhD defended in May 2015.

PhD title **Cooperative Scheduling Techniques for Execution of Parallel Tasks with Synchronization Constraints**

committee Vivek Sarkar (Chair), John Mellor-Crummey, Swarat Chaudhuri, Lin Zhong

thesis Many modern synchronization and coordination constructs in parallel programs can incur significant performance overheads on current runtime systems, or significant productivity overheads when the programmer is forced to complicate their code to mitigate these performance overheads. Our thesis is that that the use of cooperative scheduling techniques can help address the performance and productivity challenges of using modern synchronization and coordination constructs. These techniques can be implemented using extensions to programming models, compilers, and runtime systems, depending on the desired trade-offs among performance and productivity.

Master's title **Habanero-Scala: A Hybrid Programming model integrating Fork/Join and Actor models**

committee Vivek Sarkar (Chair), Robert S. Cartwright Jr, Swarat Chaudhuri

thesis This work introduced a hybrid parallel programming model that integrates previously developed Fork Join Model and Actor Model. The new model helps solve certain class of problems more efficiently and productively than either of the aforementioned models individually by enabling safe intra-actor parallelism.

2002–2005 **Bachelor of Science**, *North South University*, Dhaka, Bangladesh.
Major: Computer Science. Minor: Mathematics. GPA: 3.99/4.0. Valedictorian - 2006 Convocation.

Publications

Imam, Shams and Vivek Sarkar. **Load Balancing Prioritized Tasks via Work-Stealing**. In *21st Int'l European Conf. on Parallel and Distributed Computing*, Euro-Par'15.
This work develops a decentralized work-stealing scheduler that dynamically schedules fixed-priority level tasks in a non-preemptive manner .

Imam, Shams, Jisheng Zhao, and Vivek Sarkar. **A Composable Deadlock-free Approach to Object-based Isolation**. In *21st Int'l European Conf. on Parallel and Distributed Computing*, Euro-Par'15.
This work introduces a new approach to object-based isolation that is guaranteed to be deadlock-free, while still retaining the rollback benefits of transactions. Our approach also differentiates between read and write accesses in its concurrency control mechanisms .

Imam, Shams and Vivek Sarkar. **The Eureka Programming Model for Speculative Task Parallelism**. In *29th European Conf. on Object-Oriented Programming*, ECOOP '15.
This work introduces the Eureka Programming Model (EuPM), an extension of the `async-finish` task parallel programming model that simplifies the expression of parallel speculative *eureka-style* computations (EuSCs). A pattern common to EuSCs is a *eureka* event, such an event curtails computation time by avoiding further exploration of a solution space or by causing the successful termination of the entire computation. The constructs in our EuPM are expressive enough to encode many parallel programming patterns common to EuSCs .

Imam, Shams and Vivek Sarkar. **Selectors: Actors with Multiple Guarded Mailboxes**. In *4th*

International Workshop on Programming Based on Actors Agents & Decentralized Control, AGERE! '14.
This work introduces our extension to actors: *selectors* which can have multiple guarded mailboxes. We believe that this is a powerful extension, analogous to that of adding condition variables to semaphores. Our experimental results show that using selector-based solutions simplify programmability and deliver significant performance improvements compared to other actor-based solutions .

Imam, Shams and Vivek Sarkar. **Savina - An Actor Benchmark Suite: Enabling Empirical Evaluation of Actor Libraries**. In *Proceedings of the 4th International Workshop on Programming Based on Actors Agents & Decentralized Control*, AGERE! '14.
This Suite present 30 Actor benchmarks that includes microbenchmarks and programs that display concurrency as well as parallelism .

Peter Elmers, Hongyu Li, **Imam, Shams**, and Vivek Sarkar. **HJ-Viz: A New Tool for Visualizing, Debugging and Optimizing Parallel Programs**. In *Proceedings of the Companion Publication of the 2014 ACM SIGPLAN Conference on Systems, Programming, and Applications: Software for Humanity*, SPLASH '14.
We present a tool that generates interactive computation graphs for parallel programs. The visual feedback is valuable for a programmer to eliminate the presence of potential bugs which may otherwise be difficult to detect .

Imam, Shams and Vivek Sarkar. **Habanero-Java Library: a Java 8 Framework for Multicore Programming**. In *Proceedings of the 11th International Conference on the Principles and Practice of Programming on the Java Platform*, PPPJ '14.
Introduces the Habanero-Java library .

Imam, Shams, Vivek Sarkar, David Leibs, and Peter B. Kessler. **Exploiting Implicit Parallelism in Dynamic Array Programming Languages**. In *ACM SIGPLAN International Workshop on Libraries, Languages and Compilers for Array Programming*, ARRAY '14.
This work describes our Java-based implicitly parallelizing interpreter for the array programming language J. The interpreter exploits implicit data parallelism in the language to achieve good parallel speedups on a variety of benchmark applications, including near perfect linear speed-up on inherently parallel benchmarks .

Imam, Shams and Vivek Sarkar. **Cooperative Scheduling of Parallel Tasks with General Synchronization Patterns**. In *28th European Conf. on Object-Oriented Programming*, ECOOP '14.
This work addresses the problem of efficient cooperative scheduling of parallel tasks with general synchronization patterns. Our solution is founded on the use of *one-shot delimited continuations* and *single-assignment event-driven controls* to schedule tasks cooperatively in the presence of different synchronization patterns. We believe that any task-parallel synchronization construct can be supported by this cooperative runtime .

Imam, Shams and Vivek Sarkar. **Integrating Task Parallelism with Actors**. In *27th Conf. on Object Oriented Programming Systems Languages and Applications*, OOPSLA '12.
This work was the basis of my Master's thesis .

Adrian Prantl, Thomas Epperly, **Shams Imam**, and Vivek Sarkar. **Interfacing Chapel with Traditional HPC Programming Languages**. In *Partitioned Global Address Space Programming Models Conf.*, 2011.
This work presented a language interoperability tool that lets Chapel programs call functions and instantiate objects written in C, C++, Fortran 77–2008, Java and Python. My particular contribution to this work was to expose Chapel's distributed data types through our interfaces and make them accessible to external functions implemented in traditional serial programming languages .

Courses Taken (selected)

COMP 422: Parallel Computing, *by John Mellor-Crummey*, The course introduces the foundations of parallel computing including the principles of parallel algorithm design, analytical modeling of parallel programs, programming models for shared- and distributed-memory systems, etc.

COMP 512: Advanced Compiler Construction, *by Keith Cooper*, The course focuses on advanced topics in scalar optimization and code generation for an optimizing compiler. The course provides a working knowledge of best practice algorithms and techniques in the area of code optimization.

COMP 515: Advanced Compilation for Vector and Parallel Processors, *by Vivek Sarkar*, The course covers the foundations of advanced compilation techniques for vector and parallel computer systems, including the analysis of program dependence, program transformations to enhance parallelism, compiler management of the memory hierarchy, and inter-procedural data flow analysis.

COMP 311: Principles of Programming Languages, *by Robert "Corky" Cartwright*, The course focuses on identifying the conceptual building blocks from which languages are assembled and specifying the semantics, including common type systems, of programming languages.

COMP 425: Computer Systems Architecture, *by Peter Varman*, The course covers fundamental concepts underlying the architecture of modern computer systems. Computer architecture provides the conceptual framework for designing computers that meet the needs of diverse computing applications while dealing with the constraints and opportunities afforded by constantly evolving technology.

Work Experience

2009–present **Research Assistant**, *Rice University*, Houston, TX.
Teaching assistant for the introductory computer science course (COMP140) and consultant for the introductory parallel programming course (COMP322). Gave multiple guest lectures in COMP140 labs and COMP322. Research focused on Parallel Programming Models and Runtime Systems on multicore machines. Sole author of the Habanero-Java library used for the last two years in COMP322. Worked on projects such as CnC-Python, CnC-Babel, and Chapel Futures.

Technology Java, Scala, Python, Matlab, C++, LaTeX.

Summer 2012 and 2013 **Research Intern**, *Oracle Labs*, Redwood City, CA.
Mentors: Christian Wimmer (2012) and Peter Kessler (2013)
Implemented an implicitly parallelizing AST interpreter for the array programming language J using the Truffle framework in the Alphabet Soup project. Presented at the ARRAY'14 workshop.

Summer 2010 and 2011 **Student Intern**, *Lawrence Livermore National Laboratory*, Livermore, CA.
Mentors: Tom Epperly and Adrian Prantl
Extended support for Chapel into Babel, a language interoperability tool that previously supported C, C++ Fortran 77/90/2003, Python, and Java. Presented at the PGAS11 conference.

2009–2011 **UI Technologist**, *Travelocity*, Southlake, TX.
Senior developer for Hotel and Flight landing pages. UI Developer for configuration management system.

Technology J2EE, Groovy-Grails, Spring, jQuery, Maven, Tomcat Server.

2007–2008 **Software Developer**, *Escenic Bangladesh*, Dhaka, Bangladesh + Oslo, Norway.
Project Lead and Scrum Master for Dhaka projects, Mentor for junior Dhaka developers. Developed core service and client modules for custom Escenic plugins.

Technology J2EE, JSP, Struts, Java Swing, iBatis, Maven, Glassfish, Tomcat Server, Scrumworks, Perforce, JIRA.

2006–2007 **Software Developer (promoted from Junior Software Developer)**,
United IT Global Net, Dhaka, Bangladesh.
Develop custom web applications. Firsthand experience interacting with the customer to manage requirements.

Technology J2EE, Wicket, JSP, Servlets, Spring, Hibernate, JPA, SWT, Ant, Apache Tomcat, MySQL, Subversion, MS Project, MS Office, XPlanner.

Recommendations (selected)

Recommendations from academic advisors and committee members available on request.

David McCarty, *Manager UIT/Frontend Development*, Travelocity.
"During the time I worked with Shams he was instrumental in the creation of entirely new frameworks and tools that were paired with these. He always looked for creative solutions to business issues, and strived to make these simple in support, yet highly flexible in use. We definitely couldn't have done it without his significant contributions. I would highly recommend Shams."

Tory Smith, *Product Manager Principal*, Travelocity.
"Shams is one of the most talented developers I've ever had the opportunity to work with. He has a gift of solving even the most complex business requirements and he does so with grace and agility. Shams is also a teacher: I've had seasoned developers tell me how much they have learned from Shams and how he figures things out where others sometimes struggle."

Md. Momenul Islam Milton, *Chief Operating Officer*, Escenic Bangladesh Ltd..
"Shams is a *natural* programmer. He has natural talents to solve complex programming issues – very fast learner and can do work perfectly. He is a reliable developer – has very good commitment, communication skill and professionalism. He also understands situation (project and client) well. Bottom line is, he is a valuable professional for any software development company :-)."

More recommendations available at my LinkedIn profile.
`http://www.linkedin.com/in/shamsm`

Academic/Personal Projects (selected)

2012–2013 **Habanero-Java - Cooperative runtime**, *http://habanero-java.rice.edu/*, Rice University.
The runtime uses one-shot delimited continuations internally to avoid thread blocking operations in parallel tasks. The EventDrivenControl API can be used to build any task-parallel synchronization construct.

2013 **Habanero-Java - Library**, *http://wiki.rice.edu/confluence/display/PARPROG/HJ+Library*, Rice University.
This project exposes the Habanero-Java parallel constructs as a library for use in Java 8 compliant projects.

2011–2012 **CnC-Python**, *http://cnc-python.rice.edu/*, Rice University.
CnC-Python is an implementation of the Intel Concurrent Collections (CnC) programming model.

2009–2011 **Habanero-Scala**, *http://habanero-scala.rice.edu/*, Rice University.
Habanero-Scala integrates the async-finish and Actor programming models in the Scala ecosystem.

Personal Information

US Citizen.

Extensive international travel background, comfortable working with and in different cultures.

Extra-curricular interests: Table tennis (Rice University Champion), Cricket.