1600 South Eads Street

Apartment 316N

Arlington, Virginia 22202, USA

Summary

Xiaohui Liu

Phone: +1 313-318-8421 [xiaohui@wayne.edu](mailto:xiaohui@wayne.edu) <http://www.cs.wayne.edu/xliu>

o 6+ years experience in designing, implementing, deploying, and evaluating distributed, embedded, and real-time wireless networking protocols/software

o Extreme tenacity, perseverance, and reasoning faculty gained through 6+ years of de- bugging complex distributed protocols in large-scale multi-hop dynamic networks. Bugs are often non-repeatable and timing-dependent, caused by, e.g., race conditions and inconsistent distributed states. Worse still, no gdb.

o 2 years experience in web development

Education

2008–2014 **Wayne State University**, *Ph. D. in Computer Science*, GPA 3.93/4.0.

2004–2008 **Wuhan University, China**, *B. S. in Computer Science*, GPA 3.60/4.0.

Patent

H. Zhang, X. Liu, C. Li, “PRK-Based Scheduling for Predictable Link Reliability", U.S. Provisional Application #61/788,445, International Application #PCT/US2014/27055

Experience

4/12 – 4/14 **Physical-Ratio-K based Scheduling Protocol (PRKS)**, *Research Assistant*, Wayne

State University, MI.

PRKS is the first-ever practical distributed scheduler to ensure reliability of wireless communication while maximizing throughput, eliminating the fundamental decade-old hidden-terminal issue.

o Designed the architecture of PRKS: split code into reusable encapsulated components, grouped components wth coupled functionalities, and organized groups in a tree hierarchy.

o Implemented, single-handedly, PRKS, its two variants, and four state-of-the-art protocols from scratch, on sensors with only 48kB ROM, 10kB RAM, and 8MHz 16-bit MCU using TinyOS. The source code (∼**58,000** lines) is available at <https://github.com/xhliu/prks>.

o Deployed these protocols in two sensor network testbeds of 127+ nodes.

o Increased link reliability from as low as 0% in the state of the art to 95% in PRKS.

9/09 – 3/12 **Multi-Timescale Adaptation (MTA) Routing Protocol**, *Research Assistant*, Wayne

State University, MI.

MTA is the best distributed routing protocol for delivering probabilistic real-time traffic.

o Implemented, independently, MTA, its seven variants, and four other protocols from the ground up in TinyOS. Code (∼**16,000** lines) is at <https://github.com/xhliu/mta>.

o Modified TinyOS kernals systematically to make code run concurrently and enable real-time

computing, including radio stack, time synchronization, and resource arbitration.

o Increased deadline success ratio by 89% and reduced transmission cost by a factor of 9.7, shown by measurements in two testbeds of 127+ nodes.

2/09 – 5/09 **LifeCode**, *Technical Team Member*, LifeCode Health, MI.

LifeCode is a remote health monitoring system to compete in Microsoft Imagine Cup.

o Built a Windows mobile phone application in C# to receive and display real-time heartbeat rates collected by wearable sensors and transmitted to the phone via Bluetooth.

11/08 – 5/09 **CSC1000 Ticketing System**, *System Administrator*, Wayne State University, MI.

Students in lab course CSC1000 report machine breakdowns by filing tickets in this web-based system, where IT personnel can track and update the status of each ticket.

o Modified database schema and wrote PHP code to enable users to insert, update, delete, search, and dump tickets online using the Linux, Apache, MySQL, and PHP (LAMP) stack.

12/07 – 3/08 **Microsoft Forefront Security (MFS)**, *Software Engineer Intern*, Wicresoft Company, China.

MFS is a business antivirus software product that can be controlled over the network.

o Wrote test cases and tested MFS on different Windows families, architectures, and languages.

o Automated tests using MS-DOS scripting to run MFS on remote machines with various above configurations.

9/06 – 11/07 **Websites**, *Chief Development Officer*, Trinity Studio, China.

Trinity Studio is a studio I co-founded with three classmates, building websites for small businesses, government agencies, and universities.

o Designed database schema using SQL Server and Access.

o Developed back end using ASP, IIS, and ODBC.

Open Source Community Participation

TinyOS is the de facto operating system for low-power wireless devices, such as those used in sensor networks and personal area networks. [http://www.tinyos.net](http://www.tinyos.net/). I have reported several bugs, including:

o Patch accepted: fixing bug to set default tx power in cc2420x. [https://github.com/](https://github.com/sallai/tinyos-main/commit/974ff870551d6fcc86f44e311dcbfd0fb71dbc94)

[sallai/tinyos-main/commit/974ff870551d6fcc86f44e311dcbfd0fb71dbc94](https://github.com/sallai/tinyos-main/commit/974ff870551d6fcc86f44e311dcbfd0fb71dbc94)

o Patch accepted: fixing bug in duplicate detection in CTP. [https://www.millennium. berkeley.edu/pipermail/tinyos-help/2010-March/045095.html](https://www.millennium.berkeley.edu/pipermail/tinyos-help/2010-March/045095.html)

Technical Skills

**Languages**: expert in C, TinyOS/nesC, and Matlab; proficient in C++ and LATEX; prior experience in Java, C#, Objective-C, ASP, PHP, Javascript, and HTML

**Databases**: MS SQL Server and MySQL

**Operating systems**: Linux, Mac OS X, and Windows

**Wireless Standards**: 802.11 and 802.15.4

Awards

2009 Microsoft Imagine Cup US Software Design Top 15 Finalist

2012 Outstanding Graduate Research Assistant (GRA) Award, Wayne State University

2005 National Scholarship, China

Selected Publications

*Adaptive Instantiation of the Protocol Interference Model in Wireless Networked Sensing and Control*. Hongwei Zhang, Xin Che, Xiaohui Liu, Xi Ju. In *ACM Transactions on Sensor Networks (ToSN), 2014.*

*Taming Uncertainties in Real-Time Routing for Wireless Networked Sensing and Control*. Xiaohui Liu, Hongwei Zhang, Qiao Xiang, Xin Che, Xi Ju. In *IEEE Transactions on Smart Grid (TSG), 2013.*

*When In-Network Processing Meets Time: Complexity and Effects of Joint Optimization in Wireless Sensor Networks*. Qiao Xiang, Jinhong Xu, Xiaohui Liu, Hongwei Zhang, Loren J. Rittle. In *IEEE Transactions on Mobile Computing (TMC), 2011.*