Research Statement

Xin Jin

I design and build scalable, efficient and secure networked systems to manage physical and virtual networks and optimize data delivery for large-scale data-intensive computing and mobile applications. To balance between short-term and long-term impact, I take a problem-driven approach, and work on problems that are motivated by emerging technology trends, e.g., Software-Defined Networking (SDN), cloud computing and big data, but also tackle fundamental challenges in networked systems. I make real-world impact by releasing open-source software and transferring technology to industry.

Software is eating the world. A major trend in today's technology world is the rise of large-scale software systems to provide elastic computing resources on-demand and process massive datasets in a distributed fashion. The central theme of my research is the design of the networking component in these software systems, which interconnects hundreds of thousands of machines and provides end-to-end network support for emerging applications. I leverage SDN and virtualization to design new software-defined platforms for network management that can (i) efficiently and safely compose multiple network management services [1], (ii) quickly and consistently update distributed network configurations [2], (iii) securely support virtualized networking for virtual machines [3], and (iv) realize most network-control functions on end hosts to enable extremely simple switches [4]. Furthermore, new software systems rapidly emerge for new applications, from cloud computing, to big data, to mobile computing, to the Internet of Things. Besides the contributions to network management for these systems, I also design new solutions to provide end-to-end network support for emerging applications. These solutions are tailored for the unique characteristics of different network scenarios. Specifically, I have designed (i) an efficient big data transfer system for wide-area networks (WANs) [5], (ii) a scalable traffic engineering algorithm for large data centers [6], (iii) a scalable and flexible architecture for cellular core networks [7], and (iv) an efficient video streaming algorithm for mobile networks [8].

I am a firm believer in collaborations. My research combines my knowledge in networked systems with experience and expertise of my collaborators from both academia and industry, to solve big and important problems. During my PhD study, I have worked with professors and students in networking, systems and other areas (e.g., algorithms, programming languages) at Princeton and other universities (e.g., Yale University, University of Pennsylvania, Purdue University, ETH Zurich, Stony Brook University, the Chinese University of Hong Kong, Tsinghua University), researchers from industry labs (e.g., Microsoft Research, HP Labs, AT&T Labs, Bell Labs), and engineers from big companies (e.g., Microsoft Azure), nonprofit open-source software organizations (e.g., ON.LAB) and cutting-edge startups (e.g., Rockley Photonics, Sodero Networks, Barefoot Networks). These collaborations have produced research results in the form of publications in premiere venues, software and hardware prototypes, and open-source software releases. I will continue my collaborations with them and actively seek new collaborations in the future.

1 Software-Defined Platforms for Network Management

Computer networks play a vital rule in modern computer systems, from providing Internet access to end users, to building large-scale distributed systems for cloud computing and data analysis. Due to the fast-growing size and complexity of networks, network management is a challenging problem. Traditionally, network administrators have to configure network devices using vendor-specific interfaces and handle many low-level details of different network protocols. SDN has emerged in recent years to fundamentally change the way we build and manage networks. It decouples the control plane from the data plane, by having a logically centralized controller manage a distributed collection of switches. My research leverages this technology to build new network management platforms that provide high-level abstractions to simplify network management and uses efficient algorithms to make the platforms more scalable, flexible and secure. Furthermore, the maturity of cloud computing and the wide adoption of virtualization extends the network control boundary from physical switches to software virtual switches in servers. I also build new network management platforms that embrace the end hosts.

Efficient and secure composition of network-management services [1]: Modularity is an essential principle to build large-scale software systems. Modern computer networks run a wide diversity of management services, from routing, to monitoring, to load-balancing. Ideally, we would like to leverage the modularity principle to efficiently and securely compose best-of-breed network-management services that may be provided by different parties and developed with different programming languages. This would require a network hypervisor to host these services. However, existing network hypervisors restrict each service to a distinct slice of network traffic. We designed CoVisor, a new kind of network hypervisor that enables, in a single network, the deployment of multiple control services written in different programming languages and operating on different controller platforms. CoVisor provides a simple API that allows network administrators to define the composition of multiple controllers, to choose the virtual topology exposed to each controller, and to constrain the packet handling capability of each controller. We developed a new set of efficient algorithms for composing controller applications and for compiling configurations of virtual networks into those of the physical network. We built a CoVisor prototype and showed that it is several orders of magnitude faster than a vanilla implementation without our optimizations. We released CoVisor as open-source software, and worked with ON.LAB to transfer the core features of CoVisor into ONOS, a popular open-source SDN platform for next-generation computer networks.

Fast and consistent updates of network configurations [2]: Network administrators continuously change network configurations to alleviate congestions by routing more efficiently, to detect and block cyber-attacks, to perform network hardware and software upgrades, and so on. Updates of network configurations are painful because administrators unfortunately have to balance the tradeoff between the disruption caused by the problem (e.g., congestion and cyber-attacks), and the disruption introduced in fixing the problem. Because of the difficulty of synchronizing all switches to complete their updates at the same time in large-scale networks, serious problems like loops, routing blackholes and security policy violations can happen during the update period. Existing solutions carefully order update operations to update the network consistently (i.e., preventing these problems during the updates). However, these solutions are slow because they do not consider the variations in the update speeds of different switches. A slow operation would unnecessarily delay many other operations. We designed Dionysus, a new network update system that can quickly and consistently update network configurations. Dionysus uses a novel tool called a dependency graph to capture the dependencies between individual update operations. Based on the dependency graph, Dionysus dynamically schedules operations according to runtime conditions, and guarantees the consistency of the updates. We built a prototype of Dionysus and extensively evaluated it with testbed experiments and large-scale simulations. Results showed that Dionysus improves the median update speed by 53–88% as compared to prior methods. We collaborated with Microsoft on this project. We have filed a patent application on Dionysus, and Dionysus is part of their design for their next-generation SDN network.

Secure virtual networking for virtual environments [3]: Cloud computing leverages virtualization to offer resources on demand to multiple tenants. While this frees enterprises from building and maintaining their own infrastructures, sharing the cloud creates new vulnerabilities, where one tenant can attack another by compromising the underling hypervisor. We designed a novel system that supports virtualized networking using software switches without a hypervisor to eliminate the attack surface. In our system, the software switch runs in a Switch Domain (DomS) that is separate from other hypervisor components. Both the guest VMs and DomS run directly on the server hardware, with processing and memory resources allocated in advance. Each guest VM interacts with the software switch through a shared memory region using periodic polling to detect network packets. The communication does not involve the hypervisor or the control VM. In addition, any software bugs that crash the software switch do not crash the rest of the system, and a crashed switch can be easily rebooted. We built a prototype using Xen and Open vSwitch, and evaluations showed that the combination of shared pages and polling offers reasonable performance compared to conventional hypervisor-based solutions.

Clean-state design of host-based network control [4]: Cloud operators invest heavily in the cloud infrastructure (e.g., Google spent 11 billion dollars on its cloud infrastructure), with an estimated 15% of that spent on networking. While there have been tremendous advances for data-center networking technology, modern data-center switches are still very complicated and take a lot of responsibility. We designed Sourcey, a clean-slate design that pushes almost all network-control funtions to the end hosts and makes switches extremely simple. A Sourcey switch has no CPUs, no software, no forwarding tables, no state, and require no switch configuration. It supports only source-based routing. The end hosts push the path into each packet before sending them to the network. At each hop, the Sourcey switch pops the top label on the path stack and uses the label value as the switch output port number. The major technical challenge for Sourcey is to realize the network-control functions, like topology discovery and failure detection, with host-only mechanisms. We designed novel algorithms that use host-based end-to-end measurements to efficiently implement the network-control functions. We view Sourcey as an extreme form of SDN that fully embraces the end hosts and makes the network unprecedentedly simple.

2 End-to-End Network Support for Emerging Applications

Besides software-defined platforms to manage network devices, I also design new solutions that makes better use of network devices to provide better network support for applications. Since applications run in different types of networks that have their own unique characteristics, I design domain-specific solutions that are tailored for different networks, including wide-area networks, data center networks, cellular core networks, and mobile networks.

Optical WAN (optimizing big data transfers) [5]: Big data transfer on the WAN is a fundamental service to many globally-distributed applications. It is challenging to efficiently utilize expensive WAN bandwidth to achieve small transfer completion times and meet mission-critical deadlines. Existing solutions only adjust routing and rate allocation to optimize big data transfers, assuming a static network-layer topology. The cost is extra hops in the route. Advances in SDN and optical hardware make it feasible and beneficial to quickly reconfigure the optical devices in the optical layer. We designed Owan, a novel traffic management system that optimizes wide-area big data transfers with centralized joint control of the optical and network layers. Owan can dynamically change the network-layer topology by reconfiguring the optical devices. We developed efficient algorithms that dynamically reconfigure the optical circuits in the optical layer so that most transfers can use one or few hops in the network layer to reach their destinations. We built a prototype of Owan with commodity optical and electrical hardware. Testbed experiments and large-scale simulations showed that Owan completes big data transfers up to $4\times$ faster on average, and up to $1.35\times$ more transfers meet their deadlines, as compared to prior methods that only control the network layer.

Data center network (scalable traffic engineering) [6]: Data centers have massive scale. A major challenge for optimizing their network performance is scalability. We designed a new scalable traffic engineering solution for data centers called ensemble routing. Ensemble routing achieves scalability by operating on the granularity of flow ensembles, i.e., group of flows, instead of flows. It leverages Virtual Local Area Networks (VLANs) to divide a data center network into several sub networks, and assign flow ensembles to these VLANs. We formulated the VLAN assignment problem, i.e., optimally assigning flow ensembles to VLANs to achieve load balancing and low network cost, as a combinatorial optimization problem, and designed new approximation algorithms based on the Markov approximation framework with close-to-optimal performance guarantees. We studied several properties of our algorithms, including performance optimality, perturbation bound, convergence of algorithms and impacts of algorithmic parameter choices. We also extended the algorithms to consider Quality of Service (QoS) and interactions with TCP congestion control. We conducted extensive numerical simulations to validate our analytical results, and showed that our algorithms are able to meet different temporal constraints and tolerate imprecise and incomplete traffic matrices.

Cellular core network (scalable and flexible architecture) [7]: Cellular core networks suffer from inflexible and expensive equipment, as well as from complex control-plane protocols. To address these challenges, we designed SoftCell, a scalable architecture that supports fine-grained policies for mobile devices in cellular core networks, using commodity switches and servers. SoftCell enables operators to realize high-level service policies that direct traffic through sequences of middleboxes based on subscriber attributes and applications. To minimize the size of the forwarding tables, SoftCell aggregates traffic along multiple dimensions—the service policy, the base station, and the mobile device—at different switches in the network. Since most traffic originates from mobile devices, SoftCell performs fine-grained packet classification at the access switches, where software switches can easily handle the state and bandwidth requirements. SoftCell guarantees that packets belonging to the same connection traverse the same sequence of middleboxes in both directions. We collaborated with Bell Labs on this project. We demonstrated that SoftCell improves the scalability and flexibility of cellular core networks by analyzing real LTE workloads from one of their large customers, performing micro-benchmarks on our prototype controller, and large-scale simulations.

Mobile network (optimizing video streaming) [8]: Video streaming accounts for over 40% of mobile traffic. Although various efforts have been made, achieving reliable video streaming over cellular networks has proven to be difficult. We found that existing video streaming algorithms achieve only 69%-86% of the optimal quality, as they use indirect indicators of available bandwidth (e.g., buffer occupancy) to adjust video rates. We collaborated with AT&T to explore how to leverage better predictions by having the carrier cooperate with the content provider. We designed a new algorithm that combines the short-term bandwidth prediction (e.g., one chunk duration), buffer occupancy and rate stability to adapt video rates. Evaluations on traces from large US cellular providers showed that our algorithm can achieve nearly 96% of the optimal quality, thereby outperforming existing algorithms by up to 40%. These results lead us to believe that cellular operators and content providers can tremendously improve video Quality of Experience (QoE) by predicting available bandwidth and sharing it through APIs.

3 Future Research

In the future, I plan to build on my expertise in networked systems to develop new abstractions, algorithms, and systems for the following problems.

Networked systems for the Internet of Things: The Internet of Things (IoT) is going to connect billions of new devices to the Internet, from mobile phones, to wearable devices, to home appliances, to automobiles. These devices are becoming indispensable tools for human beings to interact with each other and with the physical world. There are many challenges to be solved for the wide adoption of IoT. (i) Scalability: IoT will connect several orders of magnitude more devices to the Internet. The Internet needs to properly assign network addresses to each of these devices and set up paths to transmit their data. This would put excessive pressure on the scalability of the Internet. I would like to design new scalable network architectures and systems to support IoT applications. (ii) Multi-tenancy: IoT applications would generate traffic with very different characteristics. Some may be delay-sensitive, some be throughput-sensitive, some transmit continuously throughout the day, and some transmit spontaneously. Furthermore, some IoT applications may require guaranteed services, e.g., medical devices that monitor seniors' health need to send altering messages to doctors right away. I would like to draw on my experience to create network virtualization platforms for IoT applications that can support these diverse requirements on network resources. (iii) Security: Many IoT devices, like automobiles, drones and implanted medical devices, directly interact with human beings and the physical world. It would cause serious problems if these devices were compromised. To make things worse, these devices need to be always connected to perform their jobs (e.g., navigation, health alerting), which makes them easier to be located and attacked. Based on my experience in designing secure networked systems for cloud computing, I would like to design new network security solutions that can detect, mitigate, and prevent attacks to IoT.

Software-defined security: As the industry is moving to build large-scale software systems for cloud computing and big data, new vulnerabilities emerge and security is a big concern for the wide adoption of these systems. While this is a vast research area, I plan to leverage my expertise in SDN and cloud security to tackle the following problems. (i) Taming security vulnerabilities in SDN: SDN is a new paradigm for building and managing computer networks. As it is still a rapidly-evolving technology, many new vulnerabilities emerge. For example, depending on the SDN applications deployed in the network, certain packets can trigger switches to send messages to the SDN controller. A malicious host can exploit this vulnerability to flood the channel between switches and the controller. I would like to develop techniques to protect SDN from emerging attacks. (ii) Security applications with next-generation programmable switches: Hardware security appliances are usually very expensive, not scalable and not flexible. While software-based solutions can easily scale out by adding more machines and onboard new features by upgrading software, they suffer from low per-server throughput. The next-generation switching ASICs will offer more programmability and functionality on switches. I would like to leverage this opportunity to design new security applications that combines the high-performance of hardware and the flexibility of software.

Bridging the gap between optics and networking: Traditionally, the optical communications and the computer networking communities are separate, with one in EE and the other in CS. In the era of big data and IoT, the switching hardware and control software need to be redesigned from the ground up, to meet the increasing need for scalable and high-bandwidth networks. I plan to draw on my knowledge and experience in optics and networking to design next-generation networking technologies and bridge the gap between these two communities. (i) Large-radix high-performance switches for data centers: Data centers provide high network bandwidth by interconnecting a vast number of switches in CLOS-like topologies. Due to the limitations on CMOS technology, today's commodity switches have small radix, e.g., 48 10G ports. Large-radix (e.g., 96 or more) high-performance (e.g., 100G ports or higher) switches can remarkably reduce the total number of switches and thus makes the network easier to build, maintain and trouble-shoot. A possible direction to create such switches is to design new hybrid switching hardware that combines the strength of the CMOS technology (on packet processing) and that of the optics technology (on switching and IO). I am excited to continue my collaboration with Rockley Photonics on this direction. (ii) Cross-layer management of the optical and network layers: A modern WAN network consists of two layers, an optical layer and a network layer. Unlike data center networks, routers on the WAN are not connected by point-to-point fibers, but by optical circuits in the optical layer. Conventionally, operators manage these two layers separately. I would like to leverage SDN to create cross-layer management systems for WANs. My research on Owan is just the first step in this direction. There are many unsolved research problems in this direction, e.g., the bandwidth guarantee problem in optical WANs. Furthermore, as most optical devices today are not standard and use vendor-specific interfaces, I would like to design open platforms and interfaces for optical networks, i.e., the "OpenFlow" for optics.

References

[1] **Xin Jin**, Jennifer Gossels, Jennifer Rexford, and David Walker, "CoVisor: A compositional hypervisor for software-defined networks," in *USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, May 2015.

[2] **Xin Jin**, Hongqiang Harry Liu, Rohan Gandhi, Srikanth Kandula, Ratul Mahajan, Ming Zhang, Jennifer Rexford, and Roger Wattenhofer, "Dynamic scheduling of network updates," in *ACM SIGCOMM*, August 2014.

[3] **Xin Jin**, Eric Keller, and Jennifer Rexford, "Virtual switching without a hypervisor for a more secure cloud," in *USENIX Workshop on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services (Hot-ICE)*, April 2012.

[4] **Xin Jin**, Nathan Farrington, and Jennifer Rexford, "Your data center switch is trying too hard," in *Symposium on SDN Research (SOSR)*, March 2016.

[5] **Xin Jin**, Yiran Li, Da Wei, Siming Li, Jie Gao, Lei Xu, Guangzhi Li, Wei Xu, and Jennifer Rexford, "Optimizing bulk transfers with software-defined optical WAN," in *submission*, September 2015.

[6] Ziyu Shao, **Xin Jin**, Wenjie Jiang, Minghua Chen, and Mung Chiang, "Intra-data-center traffic engineering with ensemble routing," in *IEEE International Conference on Computer Communications (INFOCOM)*, April 2013.

[7] **Xin Jin**, Li Erran Li, Laurent Vanbever, and Jennifer Rexford, "SoftCell: Scalable and flexible cellular core network architecture," in *ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, December 2013.

[8] Xuan Kelvin Zou, Jeffrey Erman, Vijay Gopalakrishnan, Emir Halepovic, Rittwik Jana, **Xin Jin**, Jennifer Rexford, and Rakesh K Sinha, "Can accurate predictions improve video streaming in cellular networks?," in *ACM International Workshop on Mobile Computing Systems and Applications (HotMobile)*, February 2015.