



## **Dong Zhou**

### **Data Structure Engineering for High Performance Software Packet Processing**

**Friday, January 25, 2019 – 1:00 p.m. – NSH 4305**

Compared with using specialized hardware, software packet processing on general-purpose hardware provides extensibility and programmability. From software routers to virtual switches to Network Function Virtualization, we are seeing increasing applications of software-based packet processing. However, software-based solutions often face performance challenges, primarily because general-purpose CPUs are not optimized for processing network packets.

We observed that for a wide range of packet processing applications, performance is bottlenecked by one or more data structures. Therefore, this thesis tackles the performance of software packet processing by optimizing the main data structures of the application. To demonstrate the effectiveness of our approach, we examined three applications: Ethernet forwarding, LTE-to-Internet gateway and virtual switches. For each application, we propose algorithmic refinements and engineering principles to improve its main data structures, including:

- **A concurrent, read-optimized hash table for x86 platform**
- **An extremely compact data structure for set separation**
- **A new cache design that offers both low cache miss rate and high lookup throughput**

In all three applications, we are able to achieve higher performance than existing solutions. For example, our Ethernet switch can saturate the maximum number of packets achievable by the underlying hardware, even with one billion FIB entries in the forwarding table.

**Thesis Committee:**  
**David G. Andersen, Chair**  
**Michael Kaminsky**  
**Justine Sherry**  
**Sylvia Ratnasamy, UC Berkeley**