**Fast, Accurate Simulation for SDN Prototyping**

**Mukta Gupta**  
University of Wisconsin

**Joel Sommers**  
Colgate University

**Paul Barford**  
University of Wisconsin

### Motivation
- Prototyping, evaluating and debugging SDN is hard because:
  - Increasing scale, diversity, and complexity of apps
  - Will my SDN app behave as expected when deployed in the wild?
  - Does it operate correctly and efficiently at scale?

### Objectives
- Develop an SDN simulation capability that complements existing development and debugging tools
  - A controller API environment to facilitate transition to live environments
  - Ability to generate realistic application traffic flows
  - Capability to scale up to large networks
  - Facilities for detailed debugging and tracing

### Design
- Integrate POX controller and library code via monkeypatching
  - Key aspects: calls that get or set external state (time, network) and packet/flowlet translation
- Upshot: POX controller modules can be used without modification in fs
  - Discovery, spanning tree, l2 learning, hub, l2 pairs, etc., all work out of the box

### Results: Accuracy

#### UDP CBR traffic

<table>
<thead>
<tr>
<th>Load</th>
<th>Tiny</th>
<th>Small</th>
<th>Medium</th>
<th>Large</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>6</td>
<td>8</td>
<td>33</td>
<td>72</td>
</tr>
<tr>
<td>High</td>
<td>4</td>
<td>8</td>
<td>31</td>
<td>76</td>
</tr>
</tbody>
</table>

#### Harpoon traffic (Pareto distr. flow sizes)

<table>
<thead>
<tr>
<th>Load</th>
<th>Tiny</th>
<th>Small</th>
<th>Medium</th>
<th>Large</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>16</td>
<td>33</td>
<td>104</td>
<td>193</td>
</tr>
<tr>
<td>High</td>
<td>30</td>
<td>62</td>
<td>194</td>
<td>337</td>
</tr>
</tbody>
</table>

### Results: Speedup

**Timeline**

- “Fast, Accurate Simulation for SDN Prototyping”, In HotSDN workshop of SIGCOMM ’13
- Complete packet/flowlet translations to truly make the environment seamless
- Better tracing and debugging capabilities
- Improve scalability through parallelizing fs
- Is it possible to bridge other (including non-Python) controller platforms?

### Evaluation
- Evaluate accuracy and scalability of fs-sdn
- Set up congruent experiments in fs-sdn and Mininet
  - Background traffic: CBR stream or Harpoon flows at two different loads each
  - Linear topologies in 4 configurations of increasing size (up to 100 switches)
  - Simple layer-3 shortest paths controller module

### Results
- Plots above show byte counts per second collected in fs-sdn and an equivalent setup in Mininet
- As topology and/or traffic increase, measurements collected in Mininet degrade
- Tables above show fs-sdn execution times for scenarios with 900 simulated seconds
  - Mininet takes 900 seconds for each experiment
  - pypy interpreter with JIT compiler was used for experiments