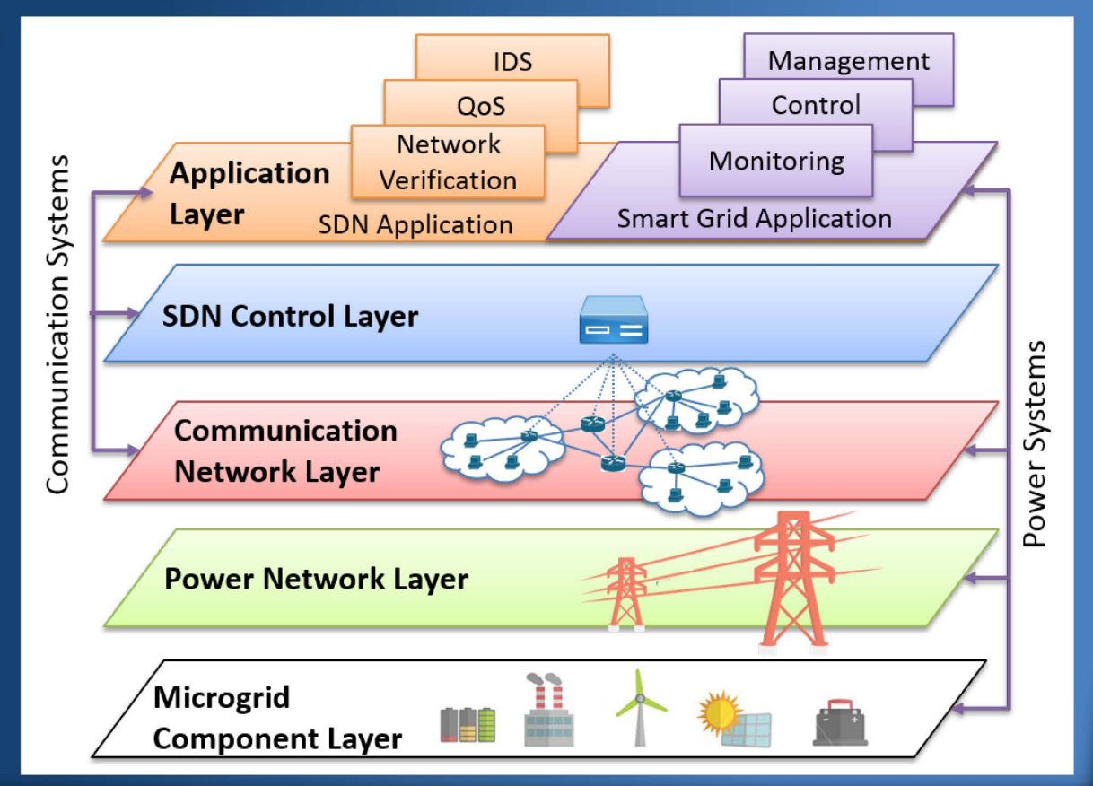
The successful operation of modern power grids is highly dependent on reliable and efficient underlying communication networks. Researchers and utilities have started to explore the opportunities and challenges of applying the emerging software-defined networking (SDN) technology to enhance efficiency and resilience of the Smart Grid. This trend calls for a simulation-based platform that provides sufficient flexibility and controllability for evaluating network application designs, and facilitating the transitions from in-house research ideas to real productions.

DSSnet: Smart Grid testbed
> Power System Simulator
> Communication Network Emulator

ILLINOIS INSTITUTE OF TECHNOLOGY

Combining Power and Communication Systems and the role of Software Defined Networking (SDN)



DSSnet: A Smart Grid Modeling Platform
Combining Electrical Power Distribution System Simulation and Software Defined Networking Emulation

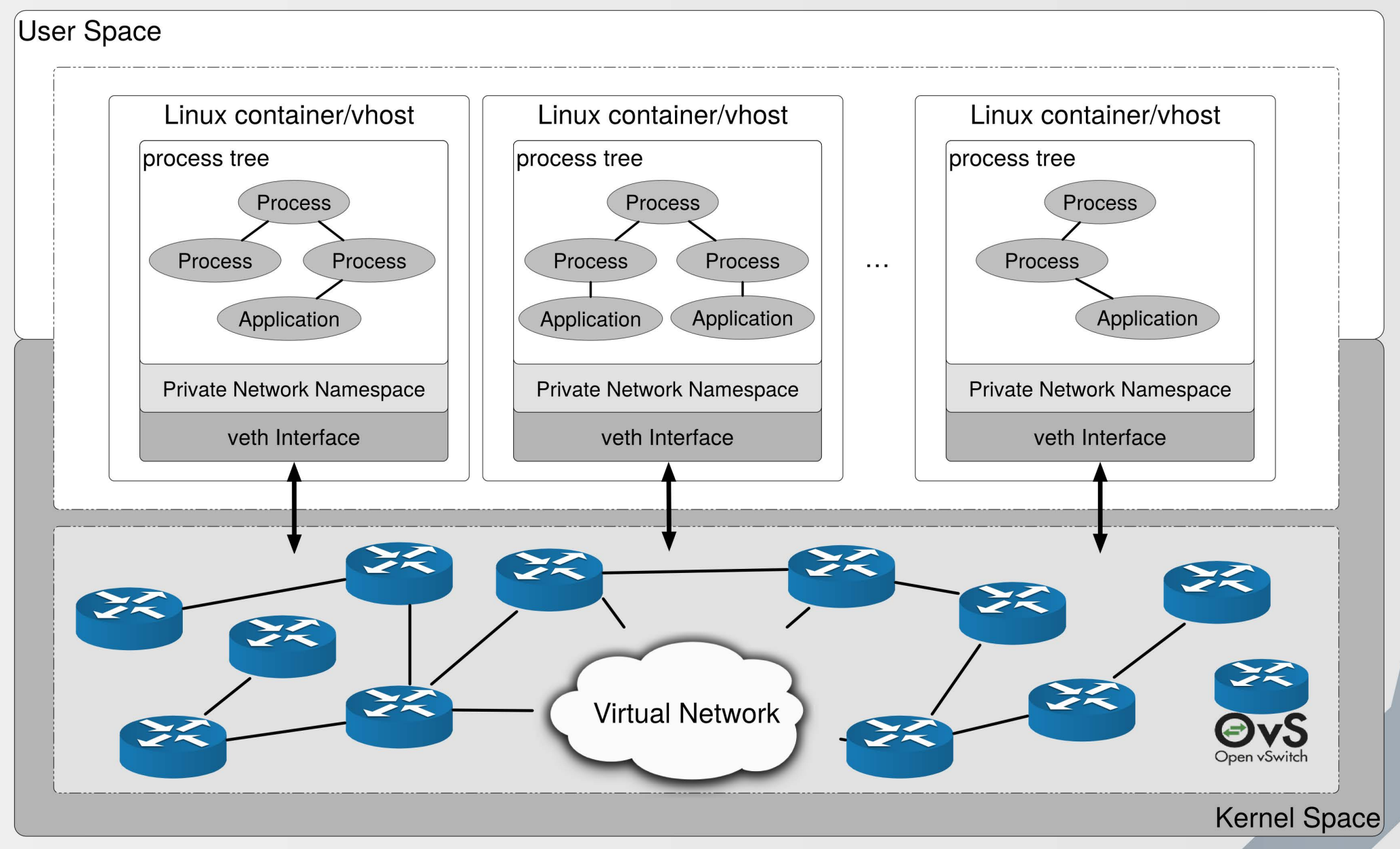
Christopher Hannon , Jiaqi Yan, Dong (Kevin) Jin

DSSnet
controllers
switches
IEDs

Communication Network Emulator

In DSSnet, hosts represent IEDs - Intelligent Electrical Devices - that exist in the power network, and each host has its own virtual network ports. Hosts have their own namespaces and can run real processes to model the IEDs.

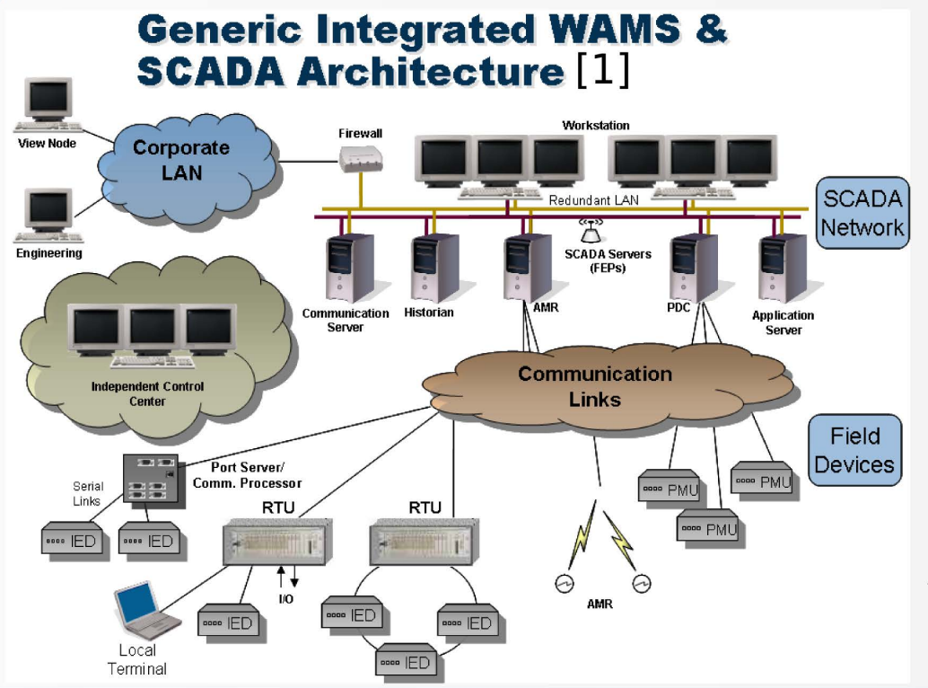
The switches used in the network emulator of DSSnet are software switches that emulate the function of real SDN switches.



Emulation does have drawbacks such as not being able to scale to thousands or hundreds of thousands of hosts due to the resource requirements of virtualization of hosts.

Smart Grid Models:

- Power grid devices with communication components
- Power applications that influence those devices



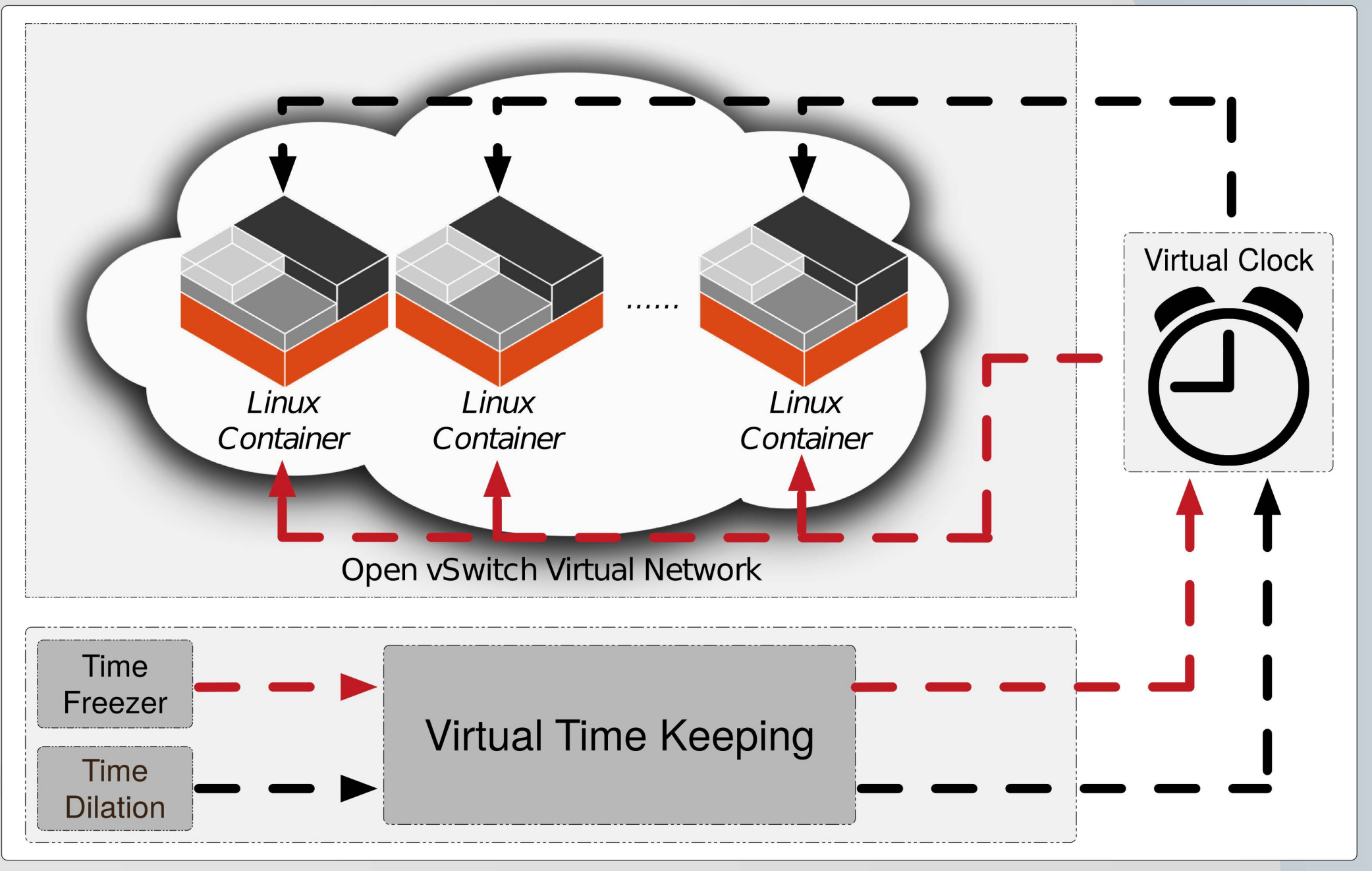
Models in DSSnet run **real** programs enabling high fidelity. Models are language independent and user defined.

Existing models include PMU, PDC, Smart Meter, Data Aggregator, Demand Response Server, and more to come.

Virtual Time Module

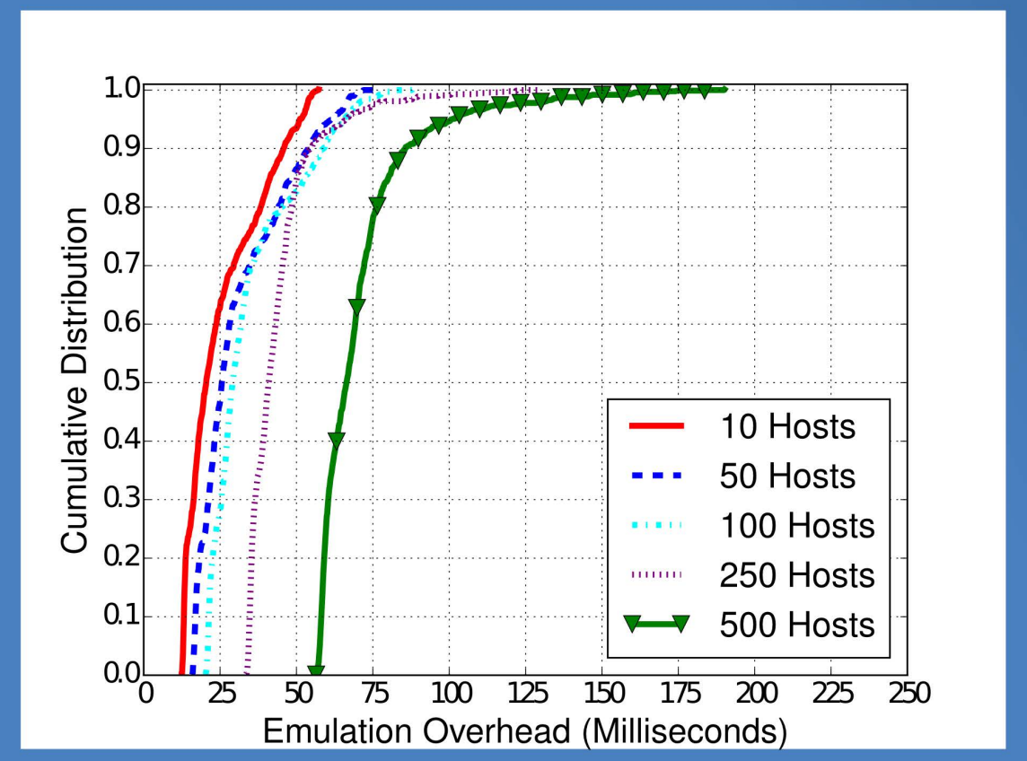
Emulation clock elapses with the real wall clock. Therefore, pausing the emulation requires more than just stopping the execution of the emulated entities, but also pausing their clocks.

Each container has a private clock, instead of using Linux OS. The containers have the flexibility to slow down, speed up or stop entirely.



Pausing the Emulation has 2 major requirements:
- All hosts should be paused or resumed when we stop or restart the emulation
- All processes inside a container should be paused or resumed when we stop or restart the emulation

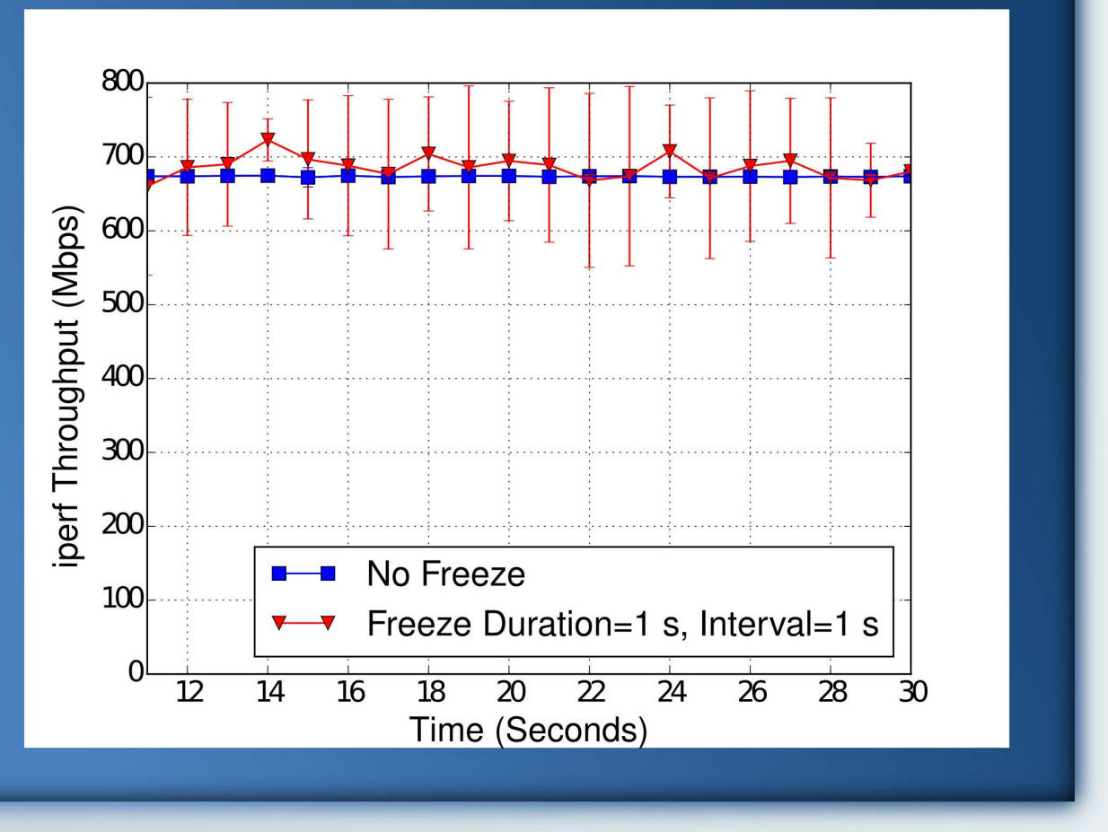
Performance Analysis



The overhead for freeze is within acceptable bounds. The amount of time required to freeze the emulation is proportional to the number of emulated hosts.

We have also shown that the accuracy of DSSnet falls within acceptable ranges compared to the network emulator alone.

The most important metrics to network performance are latency and throughput. Here we compare the results of iperf - a network utility for measuring throughput between 2 hosts - under regular operating conditons and under freeze and unfreeze routine.



Synchronization

Synchronization enforces that the system maintains temporal accuracy when events require the passing information between network and power systems.

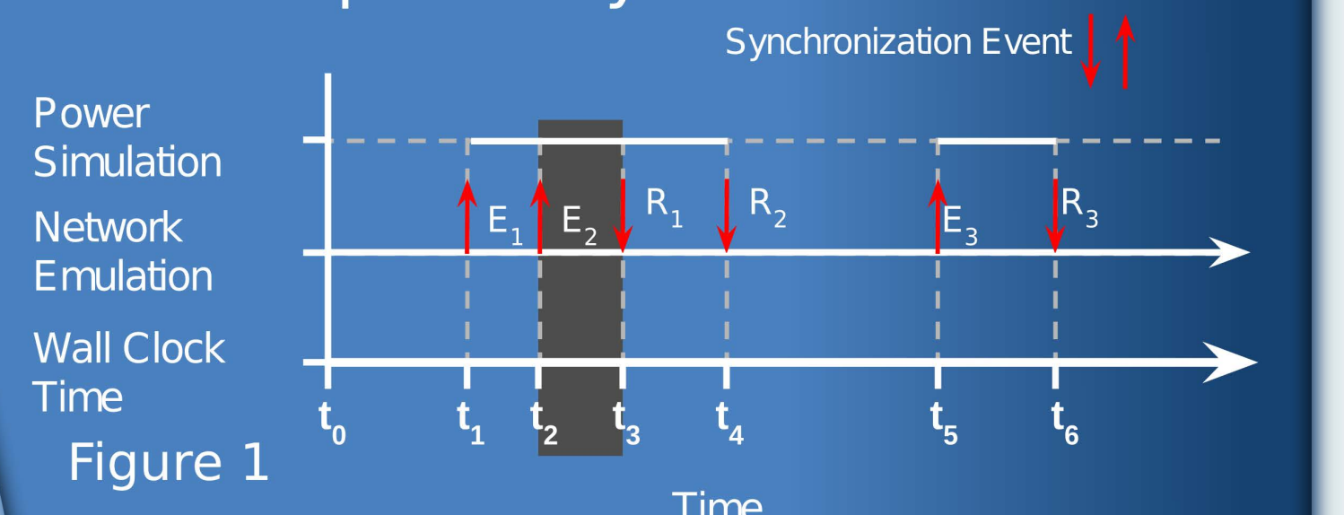


Figure 1

Synchonization errors occur when new requests arrive before existing ones are sent.

Solution: freeze (pause) the network emulator

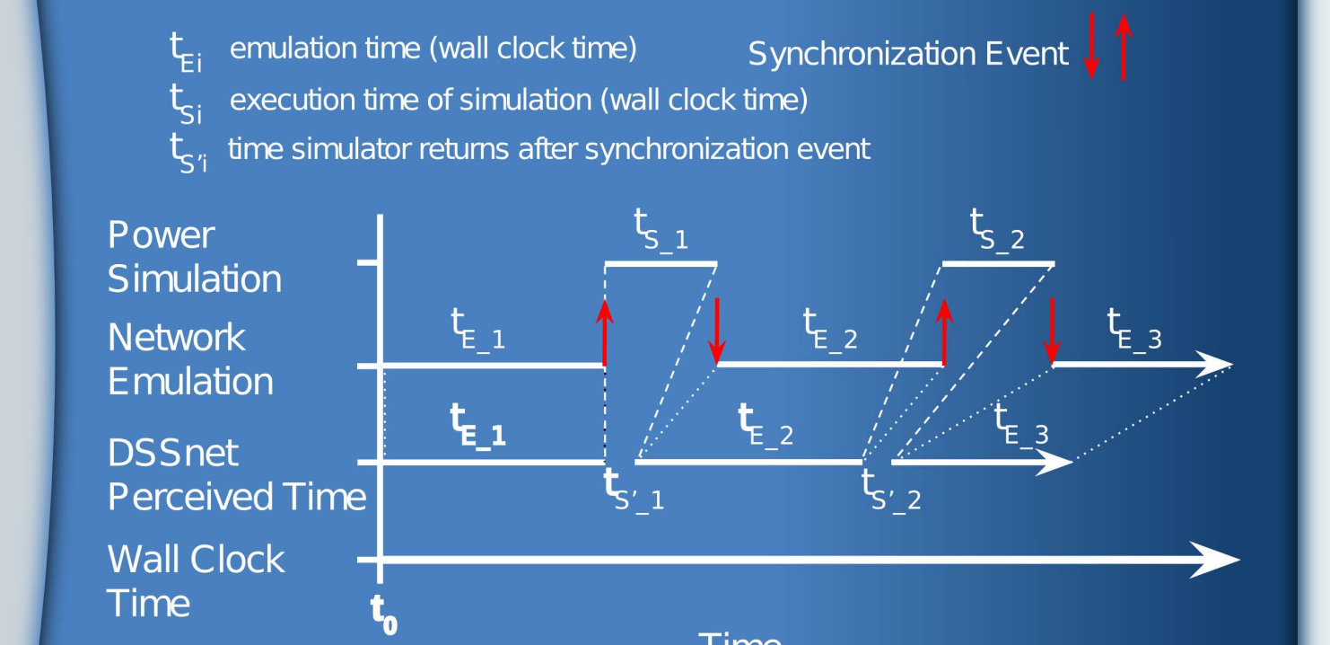

t_Ei emulation time (wall clock time)
t_Si execution time of simulation (wall clock time)
t_S'i time simulator returns after synchronization event
Synchronization Event
Power Simulation
Network Emulation
DSSnet Perceived Time
Wall Clock Time
t_S_1
t_S_2
t_E_1
t_E_2
t_E_3
t_S'_1
t_S'_2
t0
Time

Figure 2

By pausing the network emulation, DSSnet's perceived time during synchronization can be between 0 and infinity to maintain temoral precision.

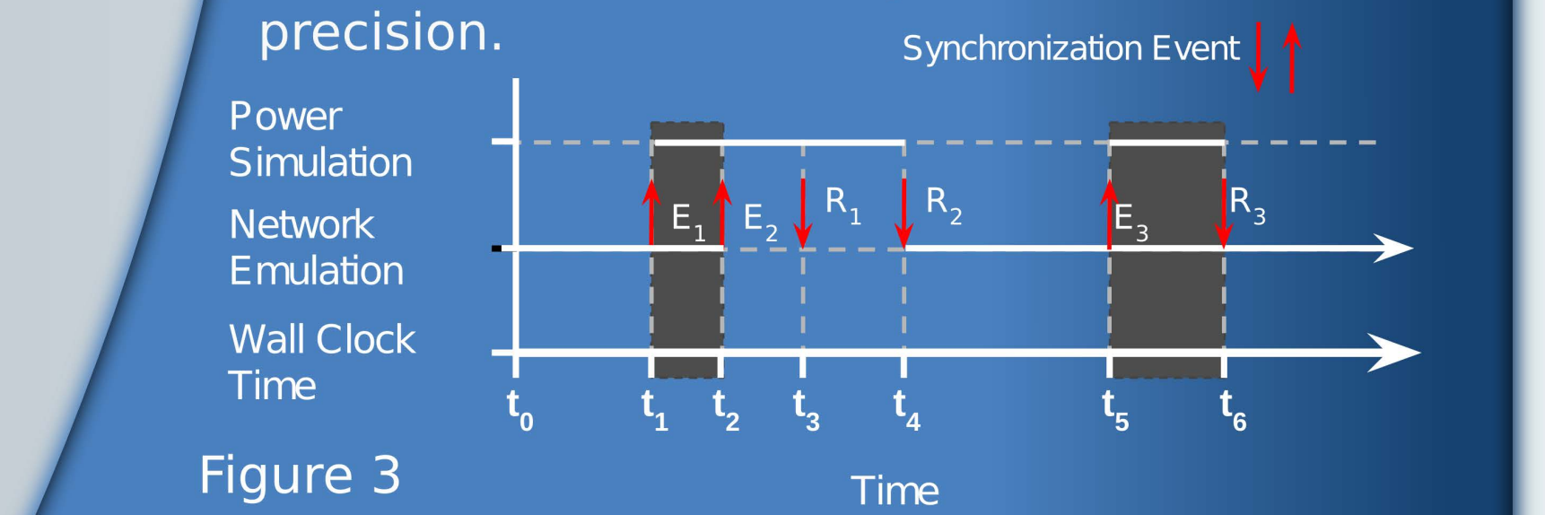


Figure 3

Through the use of blocking and non-blocking (E1 and E3) events, the execution of DSSnet can be sped up through parallelism while still maintaining temporal accuracy.

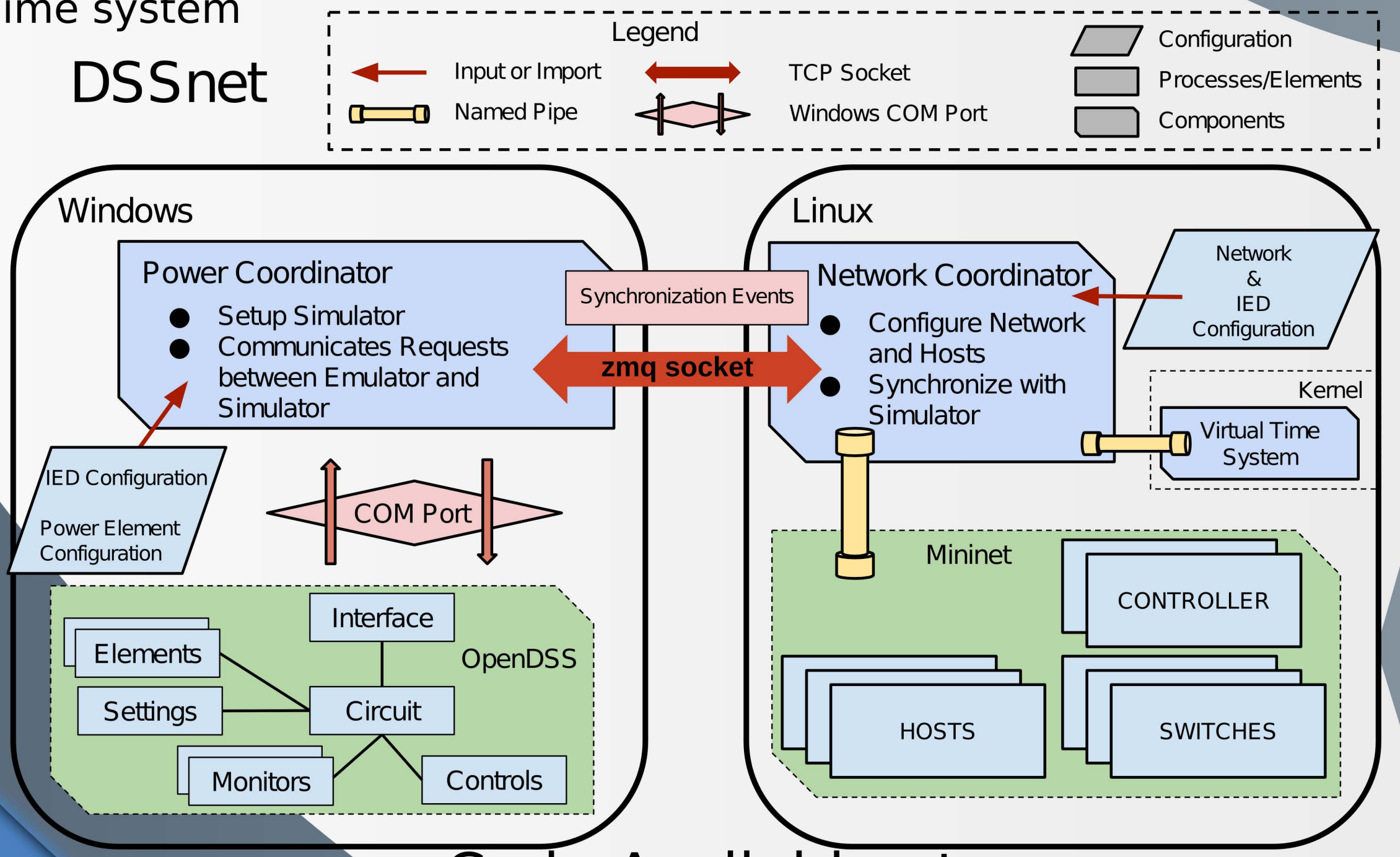
Motivation and Challenges

- How do communication delays affect power system operation?
- Identifying Cyber security solutions in Smart Grid infrastructure.
- How can we secure the Microgrid with the use of SDN?

Putting it All Together

The Detailed architecture of DSSnet is shown below. The 5 main components are:

- Network Coordinator
- Power Simulator
- Virtual Time system
- Power Coordinator
- Network Emulator



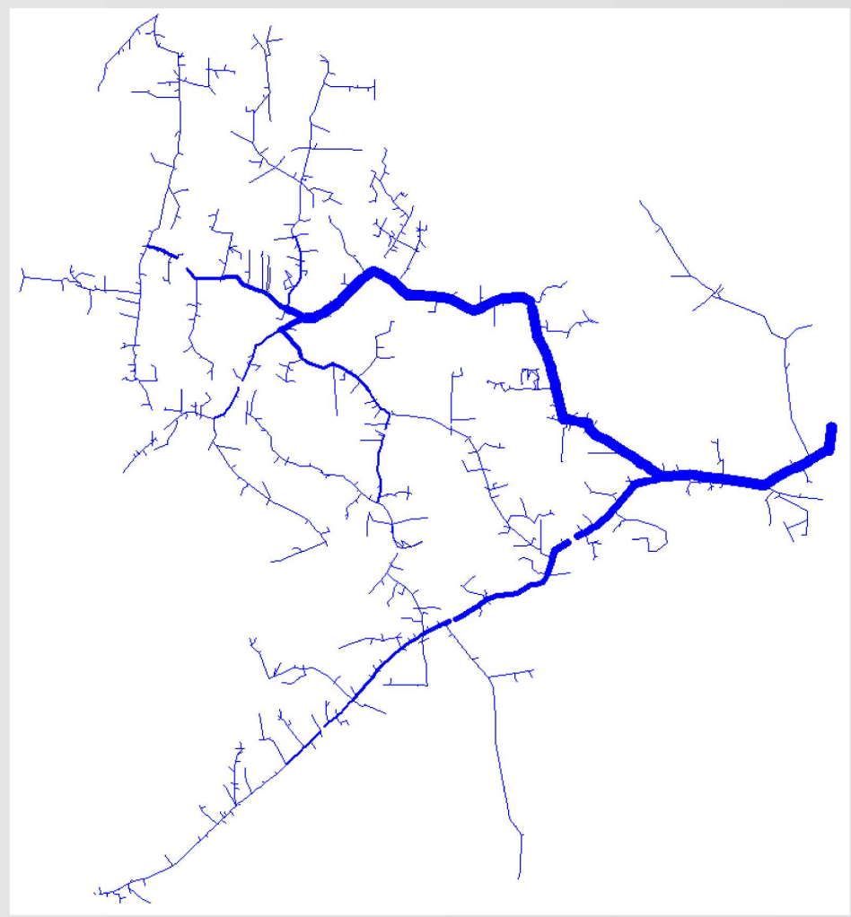
Code Available at:
https://github.com/annonch/DSSnet

Christopher Hannon: channon@iit.edu
Jiaqi Yan: jyan31@hawk.iit.edu
Dong (Kevin) Jin: dong.jin@iit.edu

Electrical Power Distribution System Simulator

OpenDSS [2] - the power simulator used in DSSnet has the following capabilities:

- Snapshot Power Flow Studies
- Daily/Yearly Power Flows
- Harmonics
- Faultstudy
- Monte Carlo Fault study



Power Flow in 8500 node system in OpenDSS [2]

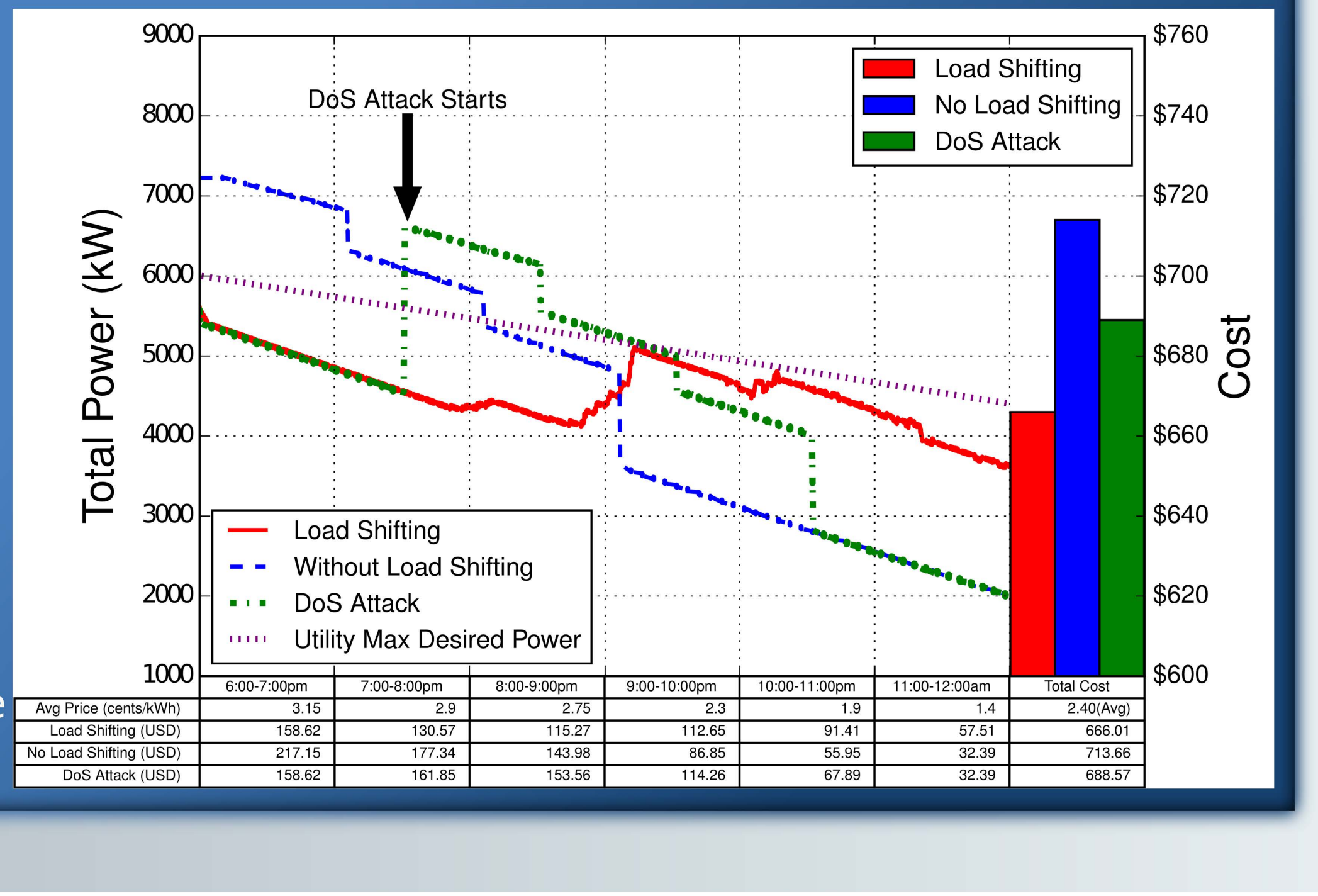
In Simulation, continuous variables are modeled with differencial equations supporting sequential-time simulation solved at discrete time intervals

The time step used in the simulation depends on the nature of the request. For protection studies, time steps at the scale of milliseconds may be required while load and generation studies may be at large scales such as seconds or minutes.

Use case

Load Shifting Application: In this use case we consider a distribution system that contains shiftable loads - loads that can have delayed start time. The loads are represented in DSSnet as a communication and power device.

Using a scheduling algorithm we can shift the consumed power to meet the utility maximum desired power and therefore lower the cost for the utility and the consumer. Through the DoS attack simulated, we show motivation for future work of securing communication network in the distribution and microgrids.



	6:00-7:00pm	7:00-8:00pm	8:00-9:00pm	9:00-10:00pm	10:00-11:00pm	11:00-12:00am	Total Cost
Avg Price (cents/kWh)	3.15	2.9	2.75	2.3	1.9	1.4	2.40(Avg)
Load Shifting (USD)	158.62	130.57	115.27	112.65	91.41	57.51	666.01
No Load Shifting (USD)	217.15	177.34	143.98	86.85	55.95	32.39	713.66
DoS Attack (USD)	158.62	161.85	153.56	114.26	67.89	32.39	688.57

Future Works

Advanced Synchonization:
- (dynamic) synchronization window
- application specific lookahead

to improve parallelism between the systems.

Distributed Emulation: to achieve better scalability in the network emulator

SDN Applications:
- network-wide configuration verification
- context-aware intrusion detection
- self-healing network layer for power applicaitons
- QoS enforcing policies in microgrid
- Network Function Virtualization for control of distributed applications

Microgrid Security and Resilience Evaluation and Modeling

References:
[1] Wide Area Measurement Systems http://energy.gov/sites/prod/files/oeprod/DocumentsandMedia/8-Securing_WAMS.pdf
[2] OpenDSS https://sourceforge.net/projects/electricdss/
[3] Mininet http://mininet.org/
[4] Linux Containers https://linuxcontainers.org/