Experiment Review Open Call 5:

Multi-Parametric QoS Predictions in Wireless Sensor Networks

MUHAMMAD ATEEQ

COMSATS University, Wah Campus

Online

IMEC, 01-April-2020

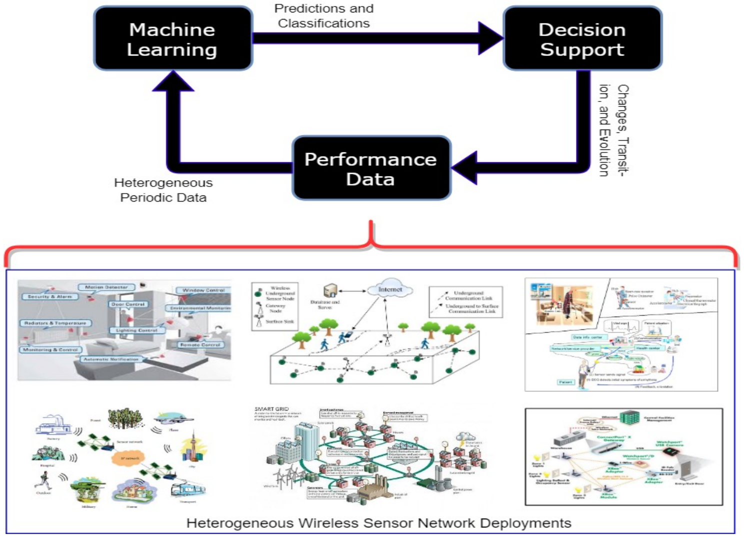
Outline

- Experiment description
- Project results
- Impact
- Feedback

Experiment Description

Concept

- Qos is growing concern as we move from WSNs to IoT
- Diverse application-specific requirements and metrics conflict
- Optimization is intractable
- Legacy solutions require hefty standardrization process, and are not adaptive

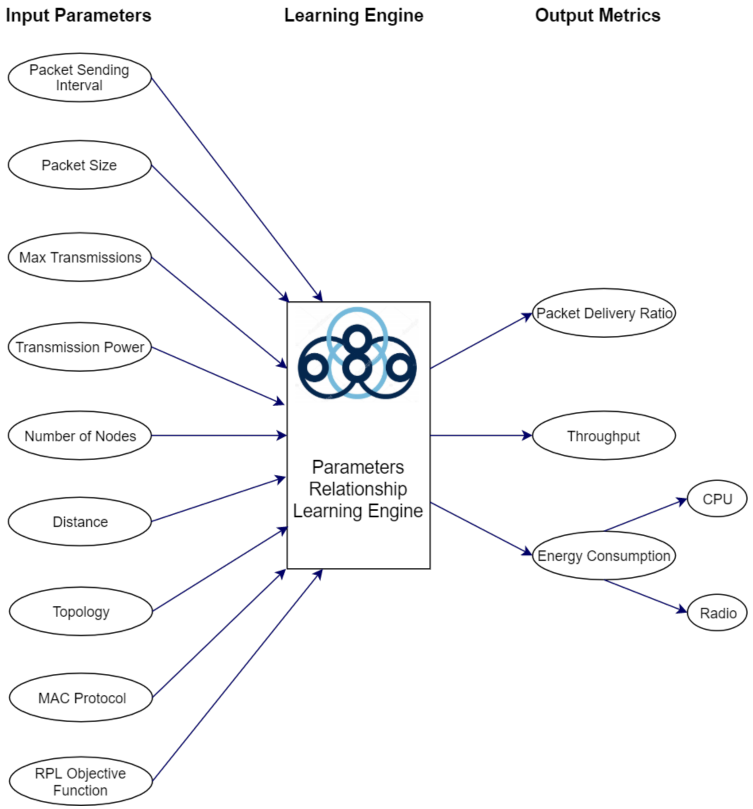


Objectives

- Devise a solution with
  - affordable complexity
  - sufficient accuracy, and
  - **adaptivity**
- To achieve this
  - design and run experiments covering diverse parameters/protocols and network settings, and collect comprehensive performance data.
  - analyze the relationships among various input parameters and output metrics
  - use ML to achieve practically acceptable prediction accuracy for QoS metrics.
  - avoid NP-hard optimizations.

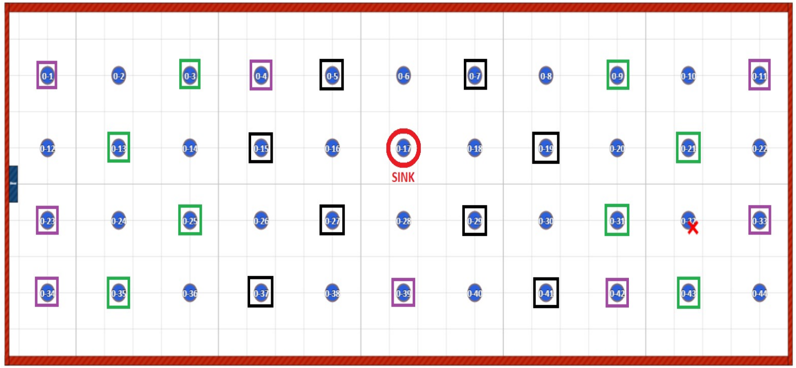
Background and Motivation

- QoS can be understood in the context of configurable parameters and settings
- Use of data-driven techniques to predict QoS is in focus recently
  - some metrics (e.g., PLR) predicted under limited variation of parameters
- Need for extensive study
  - multiple metrics and different parameters/protocols/topologies
  - need for comprehensive datasets (not a single dataset available for multihop WSNs)
  - reproducibility

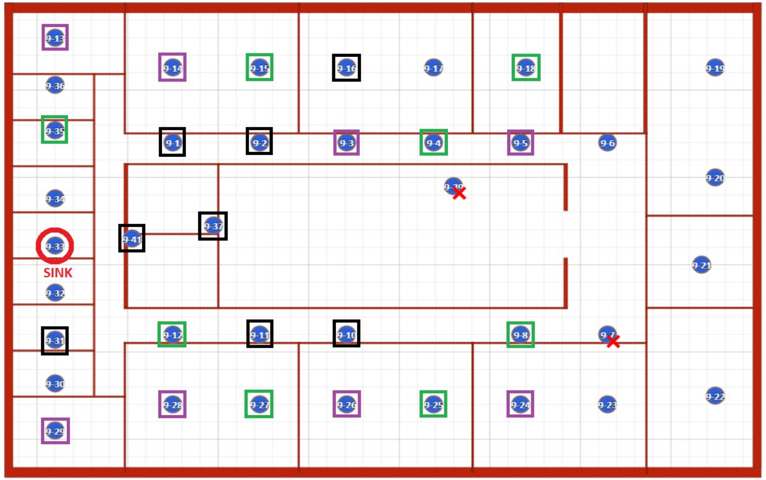


Experiment Setup (1)

Protocol Layer	Parameter	Num of Distinct Values	Proposed Values		Rationale
Physical	TP: Transmission Power (dBm)	4-5	Floor-DC: -7, -5, -3, -1, 1 Floor-9: -1, 1, 3, 5, 7		Average node-to-node distance on floor-9 is almost double compared to DC floor.
	NN: Number of Active Nodes	3	8, 16, 24		8, 16 and 24 sender nodes are used for transmission each in 2 different topologies
	NT: Number of Topologies	2	Data Centre (SC: Short avg internode distance, sink in Center) and Floor-9 (LD: Long avg internode distance, sink Distant)		
MAC	MACP: MAC Protocol	2	CSMA, TSCH		To observe the behavior of both MAC protocols
	MT: Maximum Transmissions	3	2, 4, 8		To characterize packet delivery behavior
Network	RPL-OF	3	OF-0, MRH (ETX, ETX^2)		To observe the behavior of different RPL-OF
Application	PS: Packet Size (Bytes)	3	25, 50, 75		802.15.4 accommodates small packet sizes
	IAT: Inter-Arrival Time (sec)	4	NAN=8	1, 2, 3, 4	The load ranges from very low to very high
			NAN=16	2, 4, 6, 8	
			NAN=24	3, 6, 9, 12	



Data Centre Floor



Floor 9

Experiment Setup (2)

Total combinations =

TP × NN × Topology × MACP × MT × RPL-OF × PS × IAT

= 4-5 × 3 × 2 × 2 × 3 × 3 × 3 × 4

~ 6000 combinations

Total time for experiments =

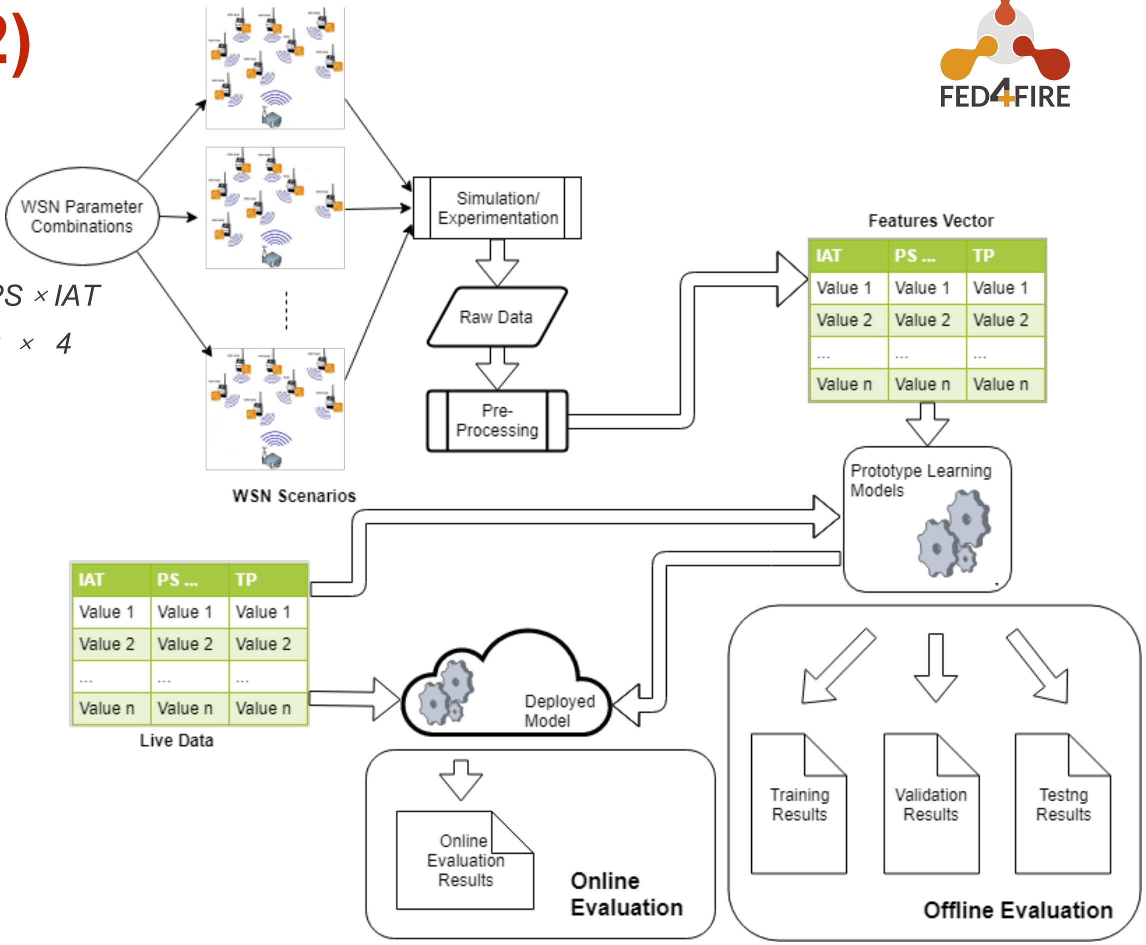
Number of parameter configurations × Time +

Overhead (1/5th of actual time)

= 6000 × 10 min + 12000 min

~ 60000 min + 12000 min = 72000 min

~ 1200 hours = ***50 days***



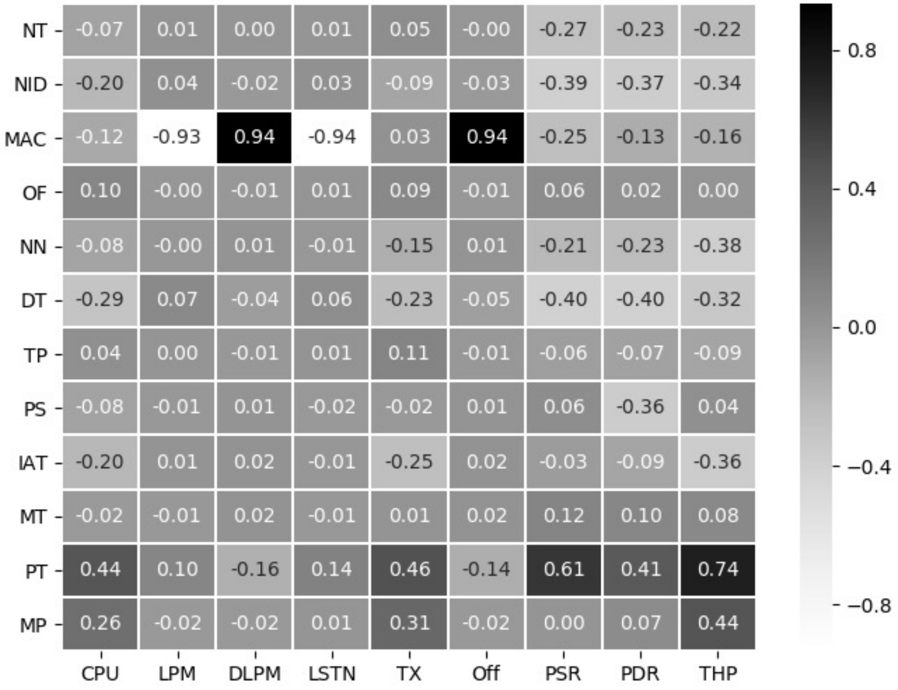
Project Results

Measurements

COLLECTED/COMPUTED INFORMATION

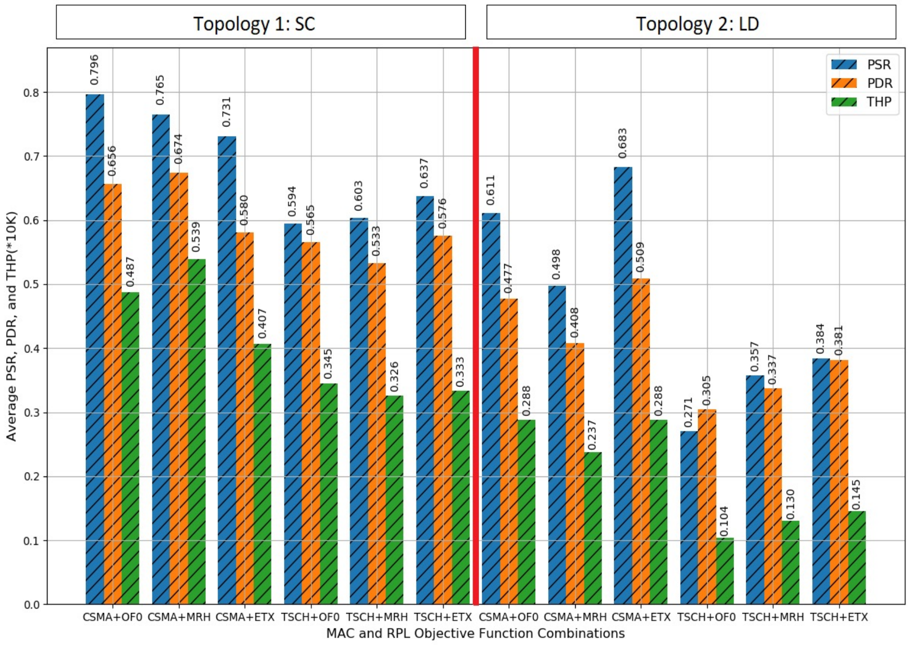
- Packet delivery
- Enenrgy consumed
  - Radio (TX, Listen, Off)
  - CPU (RCPU, LPM, DLPM)
- Throughput
- MAC (CSMS/TSCH) logs
- Routing (RPL) logs
- Delay (TSCH)

CORRELATION MATRIX

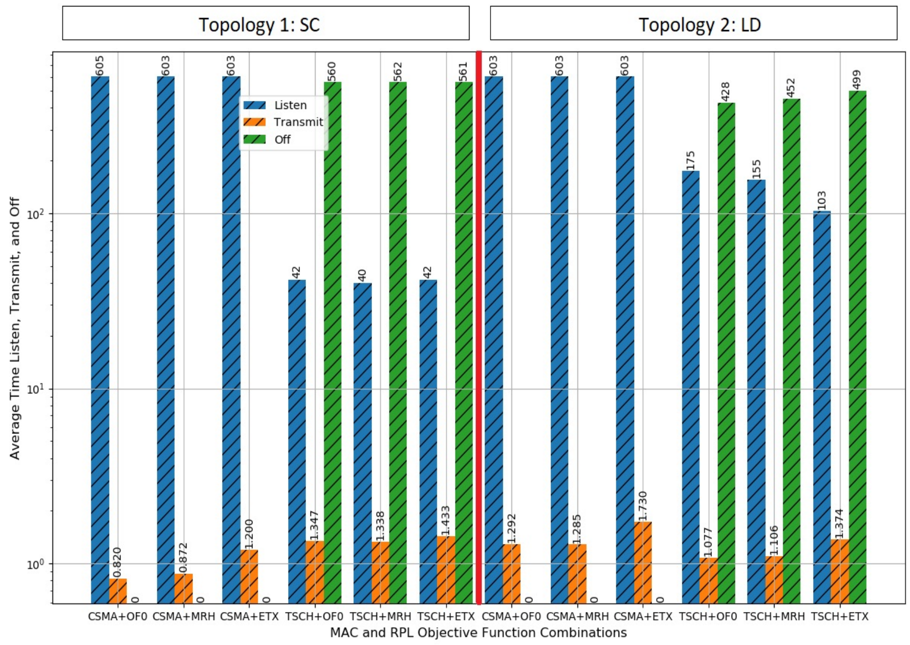


Relationships

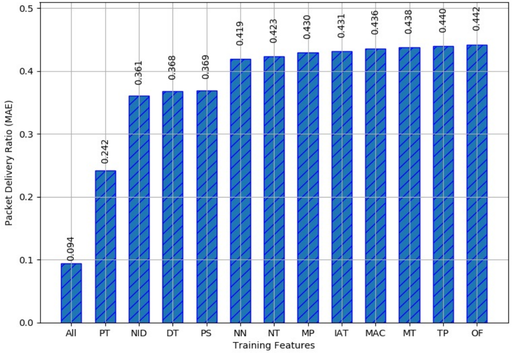
PSR, PDR, THP vs MAC, RPL-OF



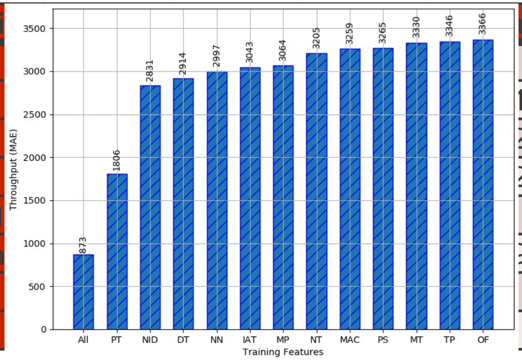
LSTN, TX, OFF vs MAC, RPL-OF

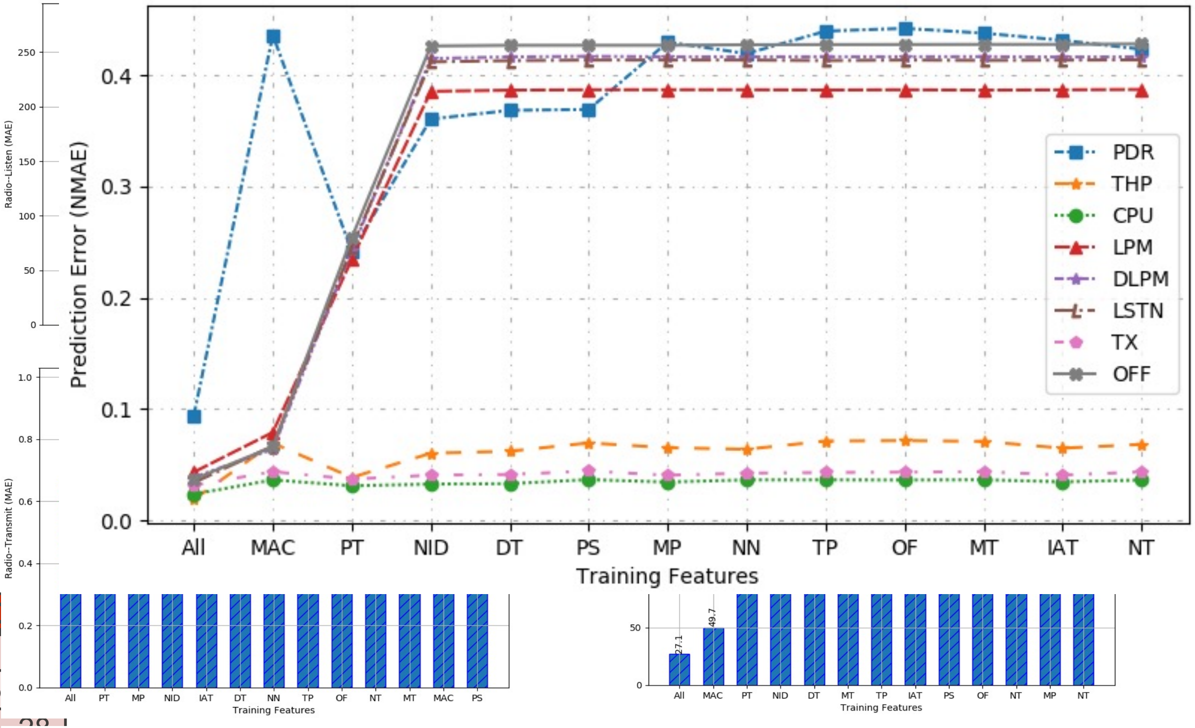


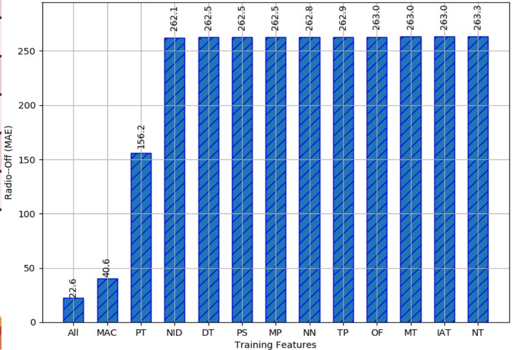
Prediction Results

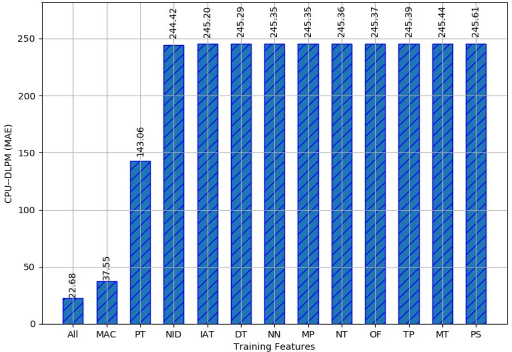


Metri	n	StD
PDR	.6	0.
THP	10	5
CPU	28	
LPM	29	
DLPM	0	
Listen	99	
TX	1	1.
Off	0	









Impact

Research Impact

- An adaptive and data-driven design in important for realizing QoS in IoT at scale
- Lays foundation for a working prototype and potentially a patent
- Opens research avenue for such initiatives in other technologies like WiFi, LTE, 5G, cognitive radio etc.
- Inclusion of heterogeneous technologies can potentially yield in a cognitive design for QoS in IoT

Value

- In the process to complete a couple of articles focusing measurement and prediction results, respectively.
- Had another project funded by NRPU, HEC Pakistan
    - Focused QoS in single-hop/multi-node scenarios
- Working on a proposal for a working prototype of adaptive QoS design in IoT
    - Included WiFi, and 5G

Importance of FED4FIRE+

- There was apparently no other way to get to experiment with real devices
    - Public repos (scarce) and simulators were the options
- FED4FIRE+ offered large-scale hardware resources and very effective interface to design and control the experiments through tools
    - Used nodes from wilab1 and virtual wall through jFed
- Brought learning, opportunity, and credibility

Follow-up Experiments

- Activities beyond the current experiment:
  - The state-of-the-art facilities from FED4FIRE+ encourage us to extend the notion of experimenting with different features and traits including QoS in diverse wireless domains. We are particularly building a strong case for experimenting with the cognitive QoS in advanced version of 802.11 (e.g., n, and ac), and 5G.
  - We also intend to develop a working prototype of data-driven framework using facilities like Tengu
- Activities in line with the current experiment:
  - TSCH: we used a minimal implementation of TSCH. We intend to evaluate the performance of full-scale TSCH.
  - Per-device Queuing: The variations in queue/buffer size will help us better understand how delay and other QoS metrics relate to it.
- Within the context of data collected using the current experiment, we have a lot of extra information in the form of MAC and routing logs. In addition, there is a task of devising a model for delay.

Feedback

Resources

- Following resources were used:
  - Sensor nodes from datacentre and floor-9 of wilab1 were used in large numbers simultaneously.
  - Virtual wall facilitated to initiate and control of experiment
  - GPU Lab was used to perform predictions of QoS metrics using deep learning
  - jFed was used as the main tool to initiate and control the experiments
  - opentestbed (cli.py in particular) to control the sensor nodes
- Most of the resources remained available as needed
- We were able to run the experiments 24/7 as we felt no significant interference
- Results were repeatable as we repeated a few runs to perform this check

Feedback

- Start-up was time consuming as we took a while to grasp the overall context of the setup.
- Opentestbed turned out to be a key tool for setting up and controlling the experiment
- Interfaces provided by jFed as well as JupyterHub were very easy to use and effective
- Data storage and download facility was adequate as we needed to collect a lot of performance data
- The overall environment appeared trustworthy

Suggestions

- Adding reader oriented assumptions (e.g., the reader is assumed to be aware of following concepts/tools:) and pointers (e.g., if you do are not aware of concept/tool “A”, please follow the link: “link”) may add a lot of value.
- The compiler installed through the command used in the tutorial: “sudo apt-get install -y srecord gcc-arm-none-eabi”
- Out of almost 1800 node runs (each run lasting 6 hours and on average 16 nodes running at a time), there were almost 5% occasions where a node resets during execution and starts running the firmware from the beginning.
- Added multi/selective-node logging and programming capability in “cli.py”.
- The sensor nodes, once programmed, have to be freed manually using “discover” command of “cli.py”. Otherwise, it is possible that even in a new reservation, an apparently idle node may be acting according to the old firmware.
- The default firmware that echoes node EUID every second is apparently compiled with defaults (transmitting at 3 dBm with CSMA). We built the same firmware with NullMAC. This helped avoid generating any default signals from the sensor nodes. This matter has been shared with Mr. Pieter Becue and Mr. Thijs Walcarius.
- There was a storage limit problem using JupyterHub. However, it was of transient nature and we were told that it would be resolved soon.
- Reservation page needs to be a bit more responsive

Added Value

- Since our experiment used w-iLab.1 and JupyterHub, the components ranking in order of importance (high to low) is:
  - Tools offered (JFed GUI and opentestbed)
  - Easy setup/control of experiments (both JFed and opentestbed facilitated this very well)
  - Support and documentation (to get a look and feel of the things) including tutorials
  - Diversity of available resources (enables research and experimentation across multiple state-of-the-art technologies)
  - Combining infrastructures (is of primary importance for researchers focusing on the integration of diverse/heterogeneous technologies)

FED4FIRE

Co-funded by the European Union

Co-funded by the Swiss Confederation

This project has received funding from the European Union's Horizon 2020 research and innovation programme, which is co-funded by the European Commission and the Swiss State Secretariat for Education, Research and Innovation, under grant agreement No 732638.

WWW.FED4FIRE.EU