

# **DESIGN, IMPLEMENTATION, AND EVALUATION OF NAPALI: A NOVEL DISTRIBUTED SENSOR NETWORK FOR IMPROVED POWER QUALITY MONITORING.**

Dr

Sergey Negrashov  
March 10, 2020,

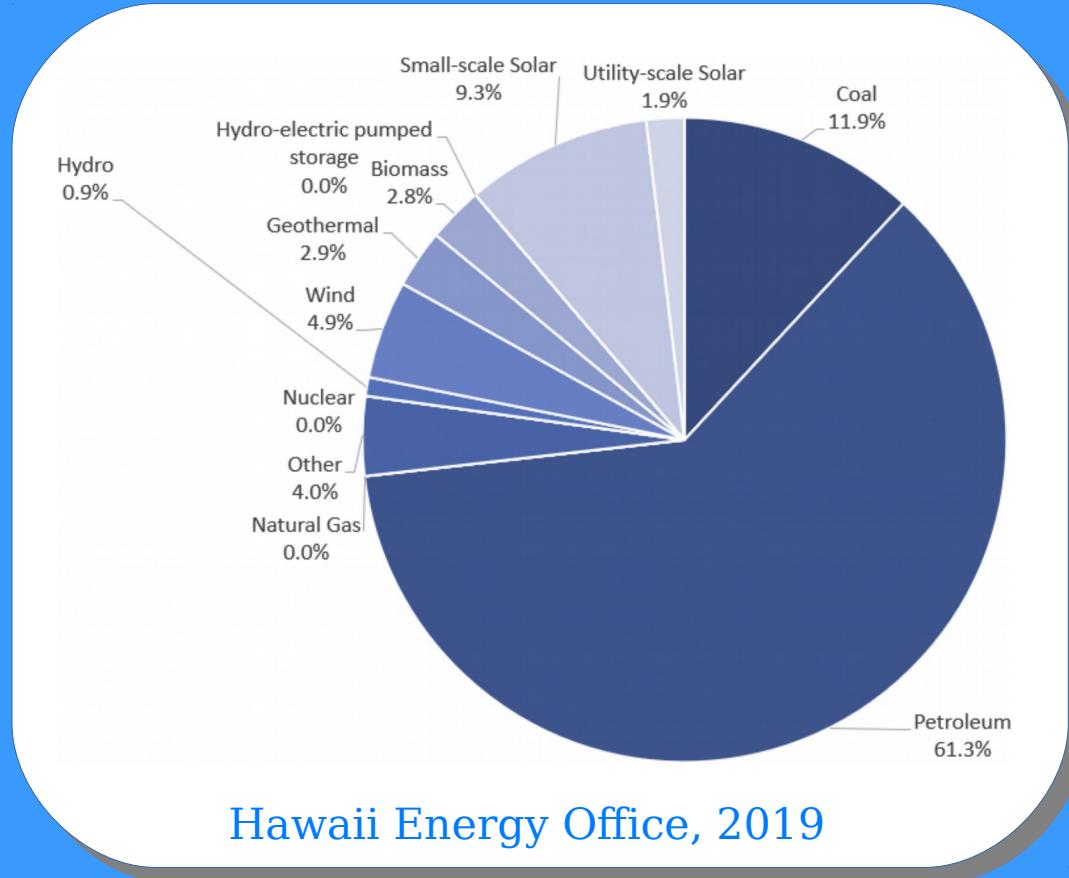


# Outline

- Power Quality
- Napali
- OPQ Network
- Evaluation
- Applications
- Future Work



# Climate Change

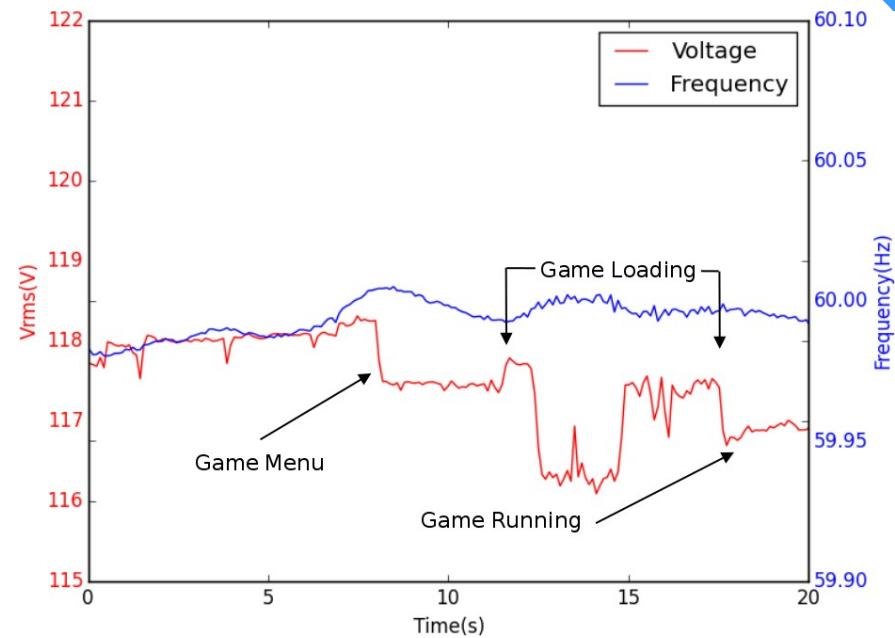
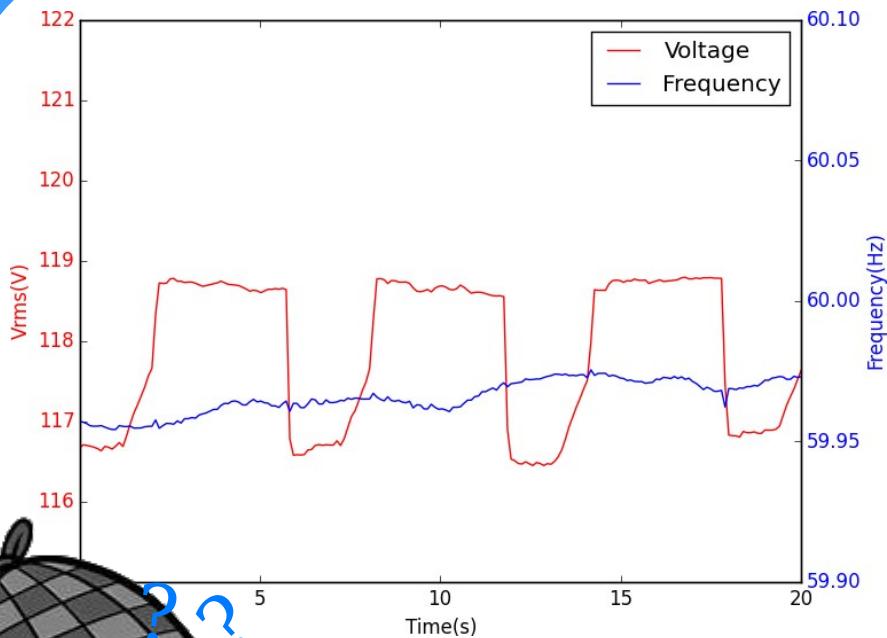


# Problem of Power Quality

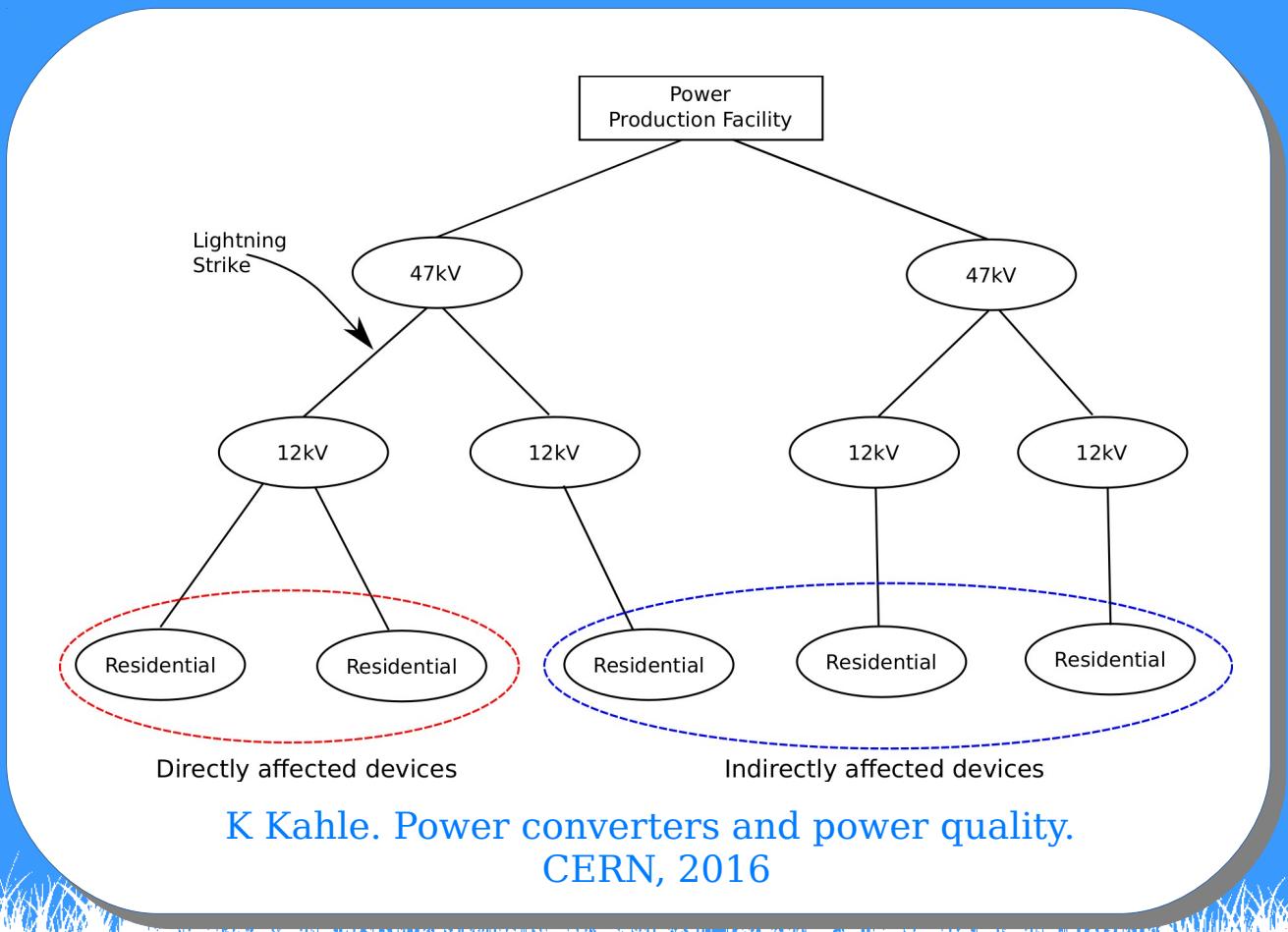
- 1. Amplitude**
- 2. Fundamental Frequency**
- 3. Harmonic Distortion**
- 4. Transient**



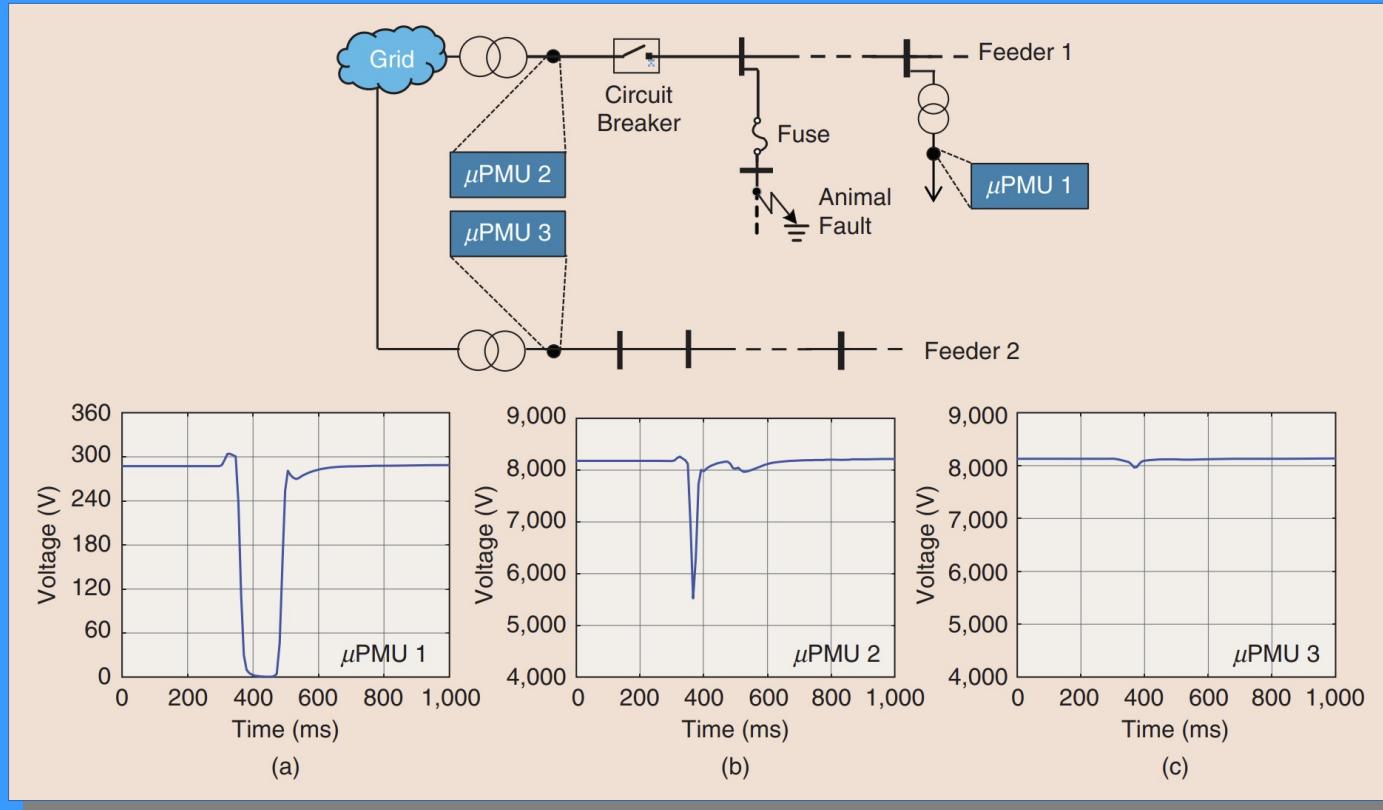
# Local Detection Issues



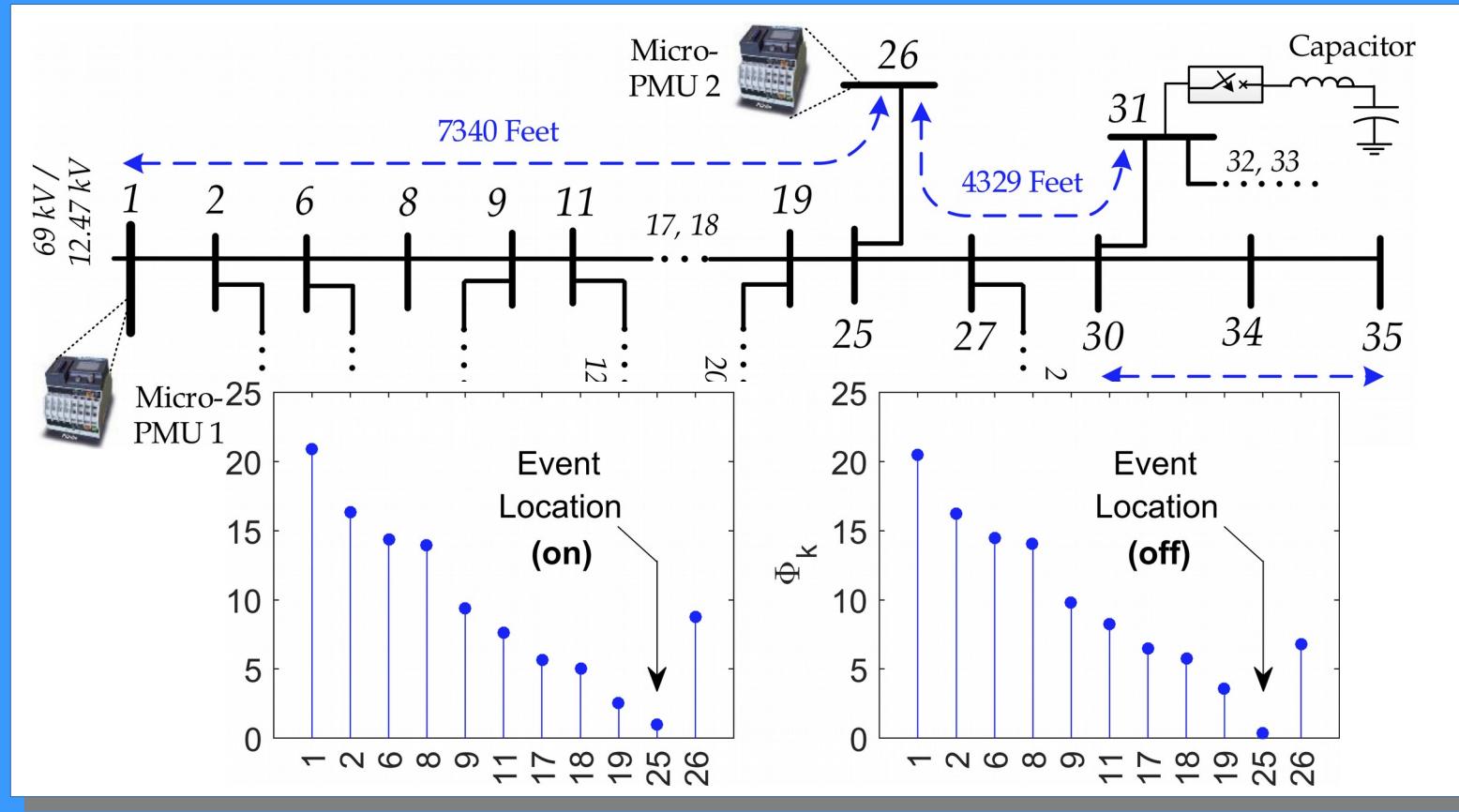
# Global Detection Issues



K Kahle. Power converters and power quality.  
CERN, 2016



Mohsenian-Rad, Hamed, Emma Stewart, and Ed Cortez.  
"Distribution synchrophasors: Pairing big data with analytics to create actionable information."  
IEEE Power and Energy Magazine (2018)



Farajollahi, Mohammad, et al. "Locating the source of events in power distribution systems using micro-PMU data." IEEE Transactions on Power Systems (2018)

# Event Detection



Animal



Detection



Bunny



Classification



Fancy  
Bunny



# Traditional Methods

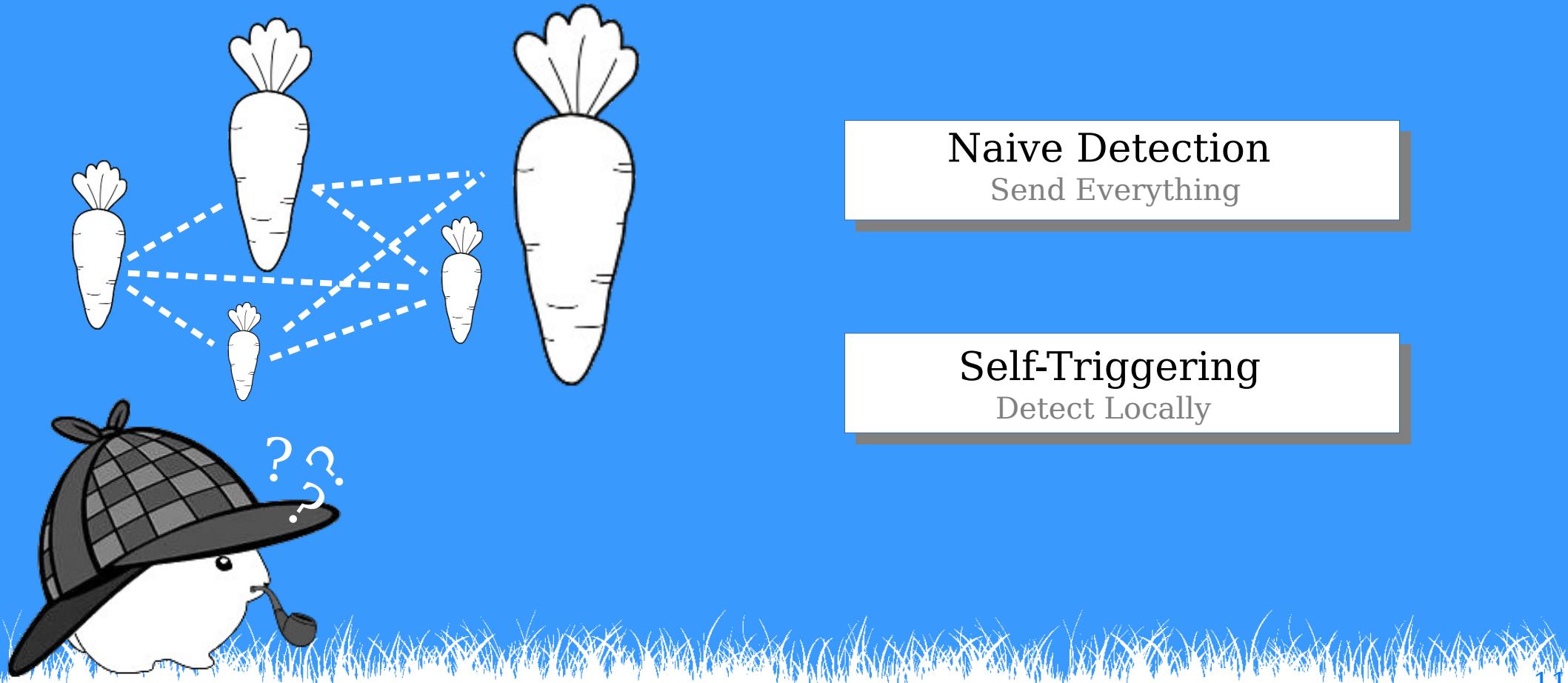
Metric Extraction

Threshold

Event Storage



# Distributed Event Detection



# Distributed Event Detection

Naive Detection  
Send Everything



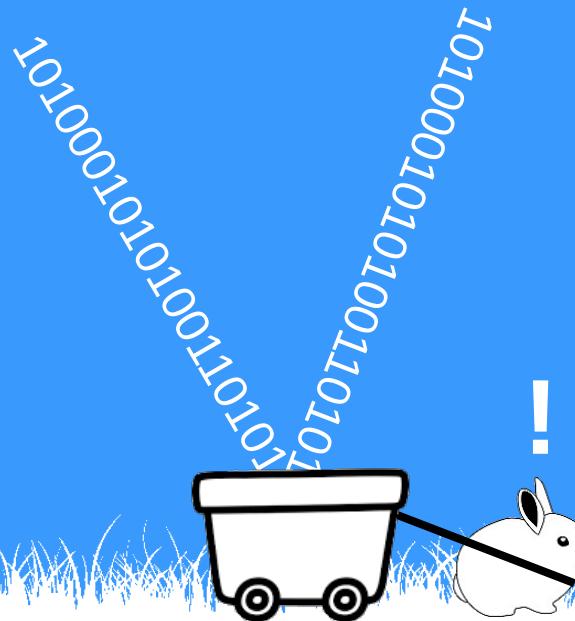
Self-Triggering  
Detect Locally



# Issues Current Methods

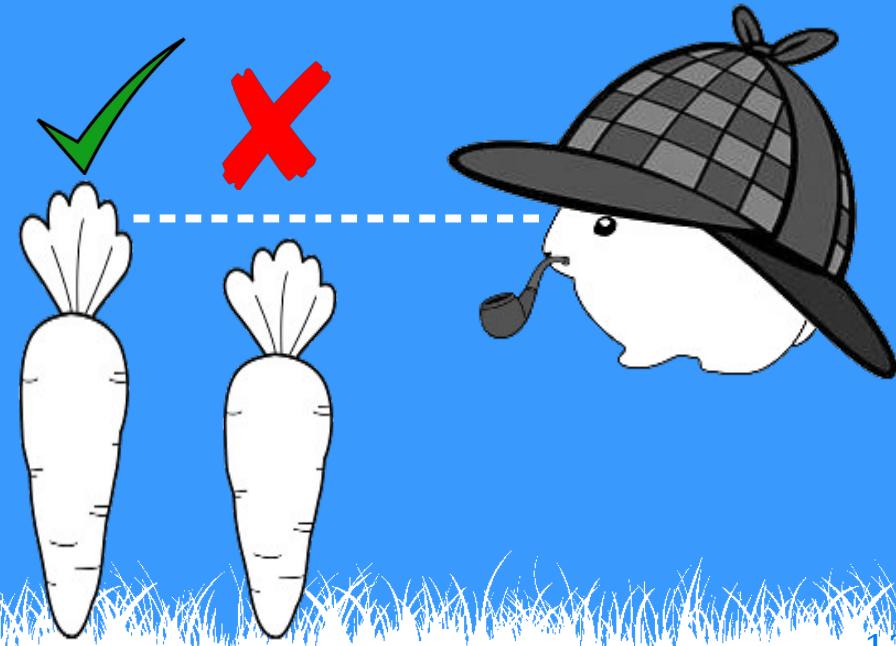
# Naive Detection

- Bandwidth
  - Memory



# Self-Triggered

- ### - Detection Efficiency



The main challenge is to go beyond manual methods based on the intuition and heuristics of human experts...

... it is crucial to develop the machine intelligence needed to automate and scale up the analytics on billions of  $\mu$ PMU measurements and terabytes of data on a daily basis and in real time.

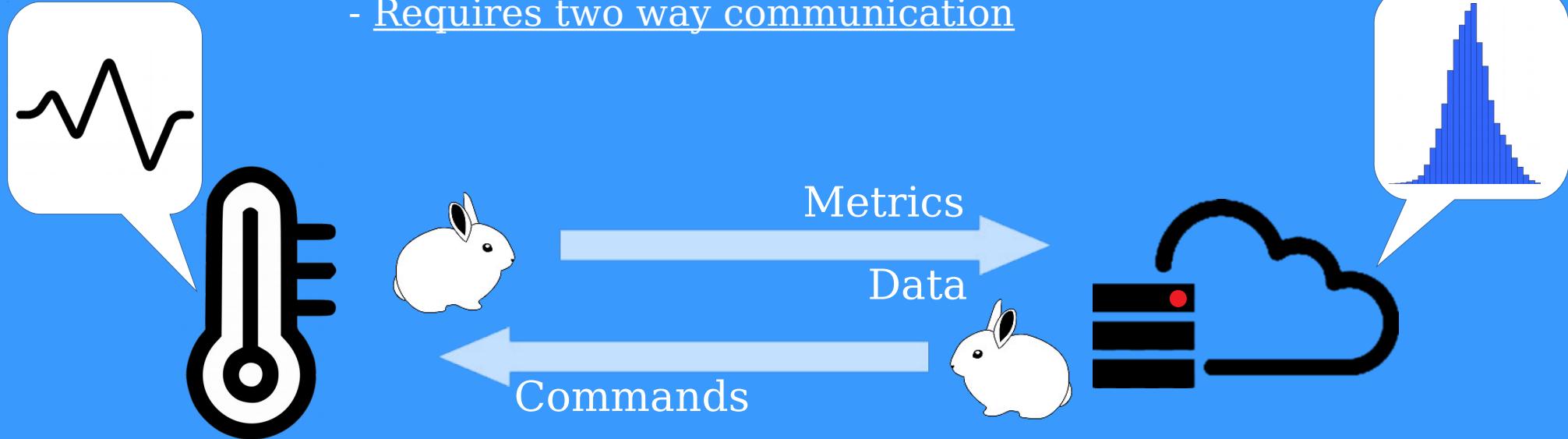
Mohsenian-Rad, Hamed, Emma Stewart, and Ed Cortez.  
"Distribution synchrophasors: Pairing big data with analytics to create actionable information."  
IEEE Power and Energy Magazine (2018)

I wonder where this is going?



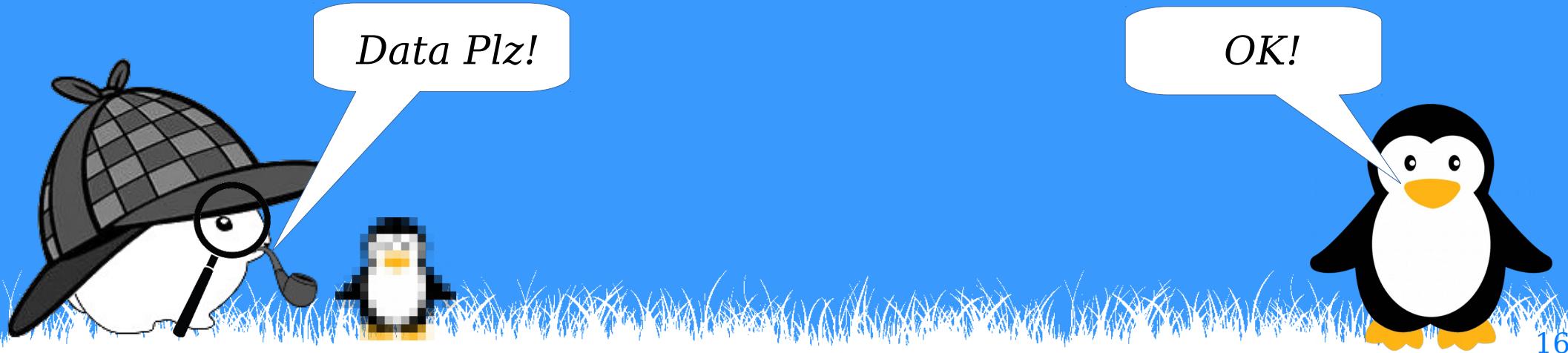
# Napali

- Sensor forward metrics to the sink
- Sensor stores raw data locally
- Sink builds a statistical model of the sensor
- Sink requests data from the sensor
- Requires two way communication



# Napali

- Sink maintains a coarse view of the entire system.
- Sink filters out local events.
- Sink determines which devices participate in event detection



# Claim



**Napali provides a novel architecture that is both a feasible solution to the problem of distributed power quality monitoring and provides significant benefits over the two standard alternative architectures.**



# Claim



**Napali architecture can, in principle,  
provide benefits for other domains beyond  
power quality.**



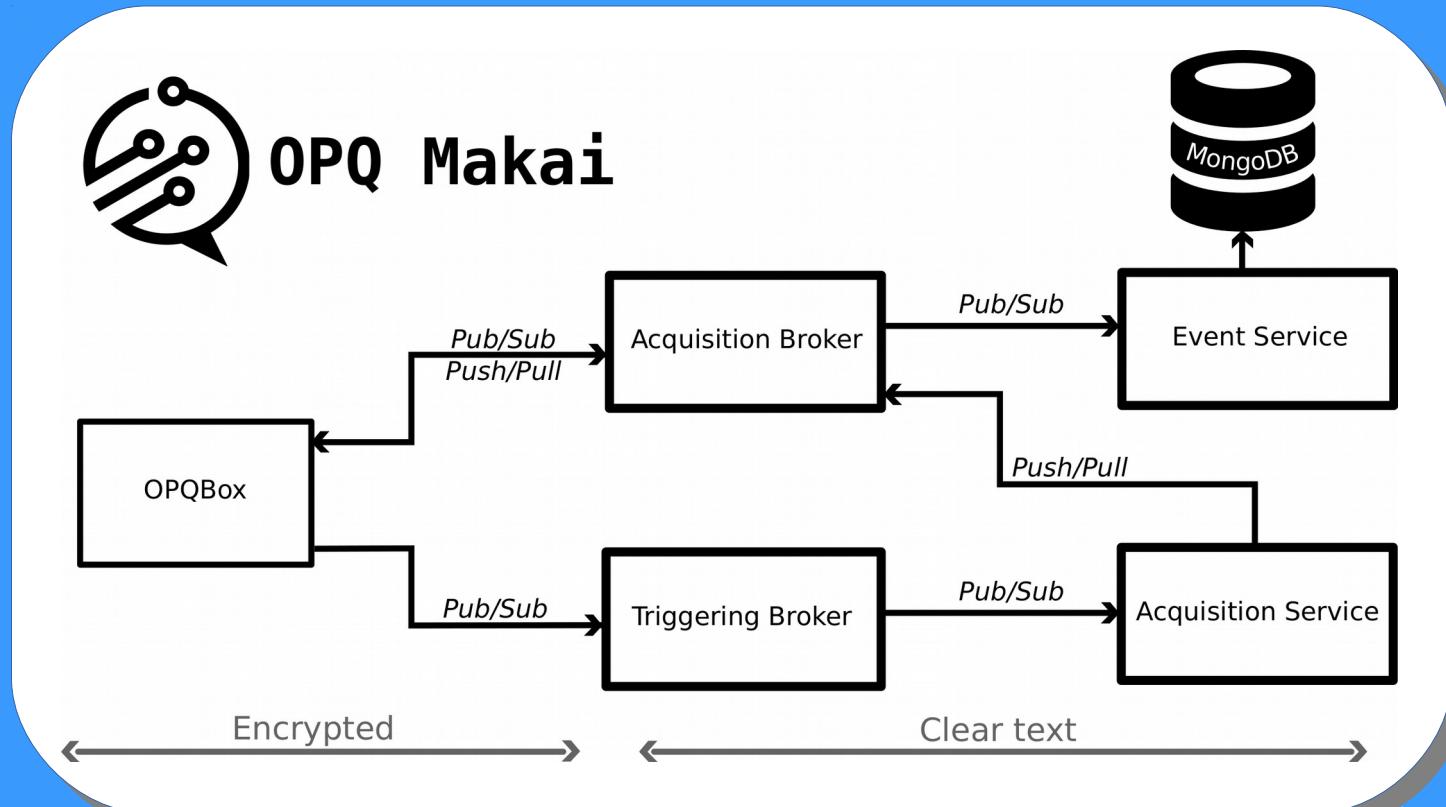
# **Subclaims of the Thesis**

- 1. Napali minimizes bandwidth**
- 2. Napali mitigates device latency effects**
- 3. Napali minimizes sink processing requirements**
- 4. Sub-threshold data acquisition is a viable event detection strategy**
- 5. Temporal locality triggering results in a low false negative detection**



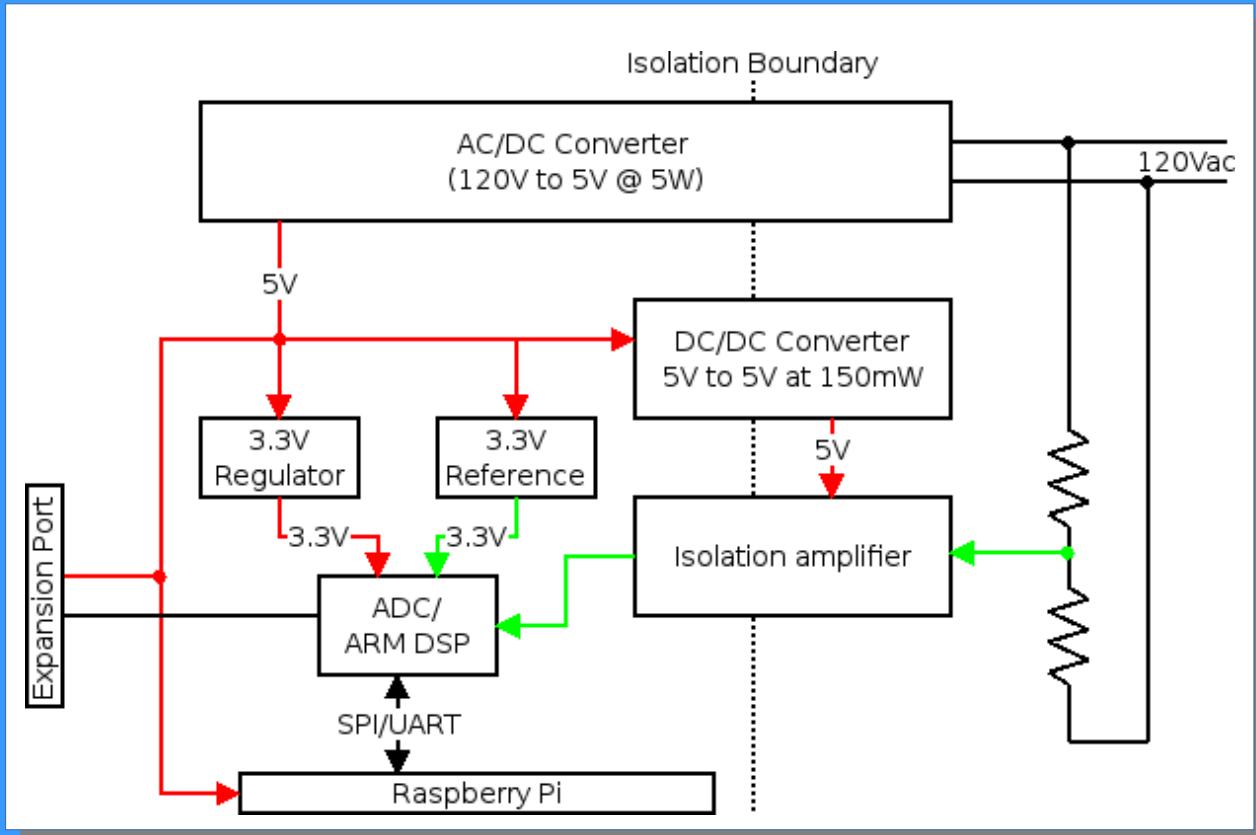
# Open Power Quality

## Gridwide event detection



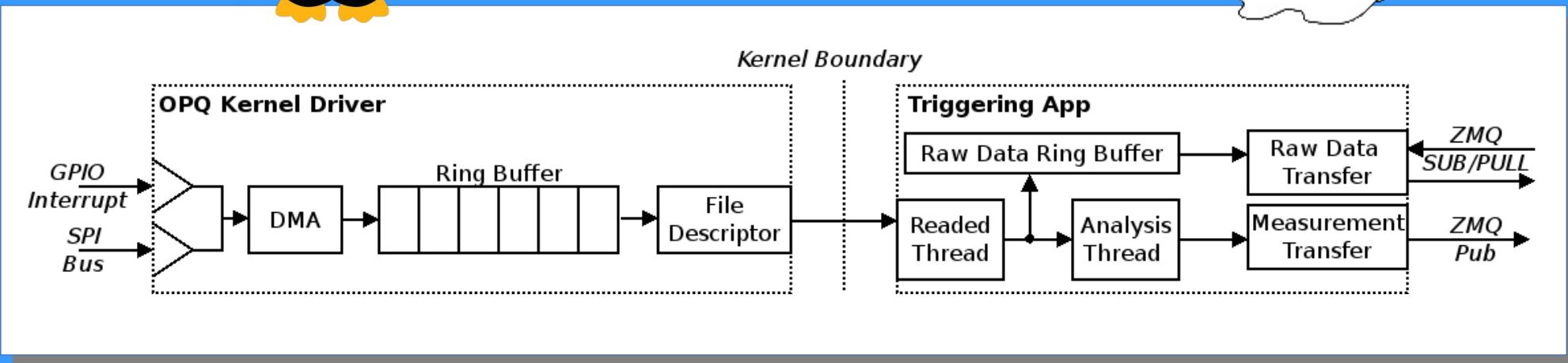
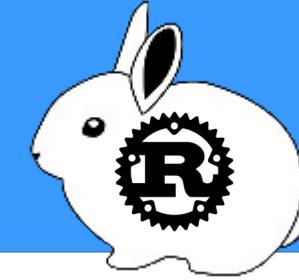
# OPQ Box

## Hardware



# OPQ Box

## Software



# OPQ Box

## Metrics

### Metrics:

- $V_{rms}$
- Fundamental Frequency
- THD
- Transient

### Transmission Rate:

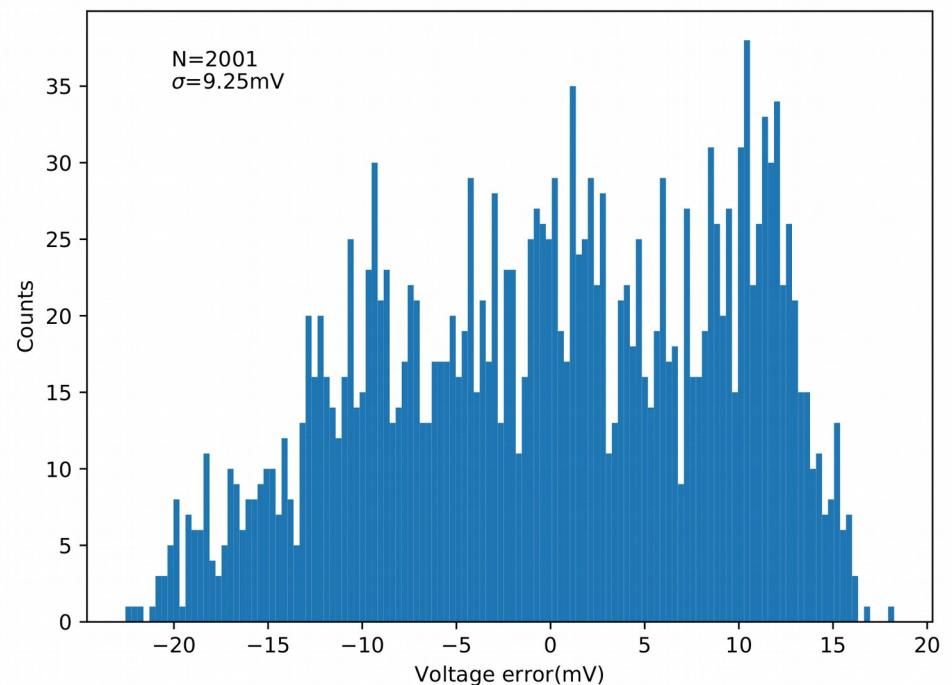
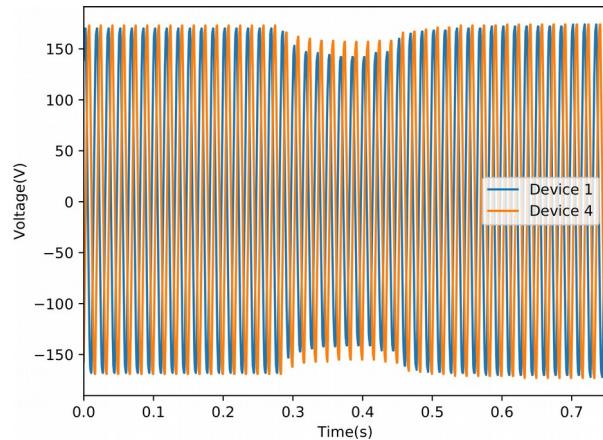
- 1s (Mean/Min/Max)

### RDRB:

- 1 Hour



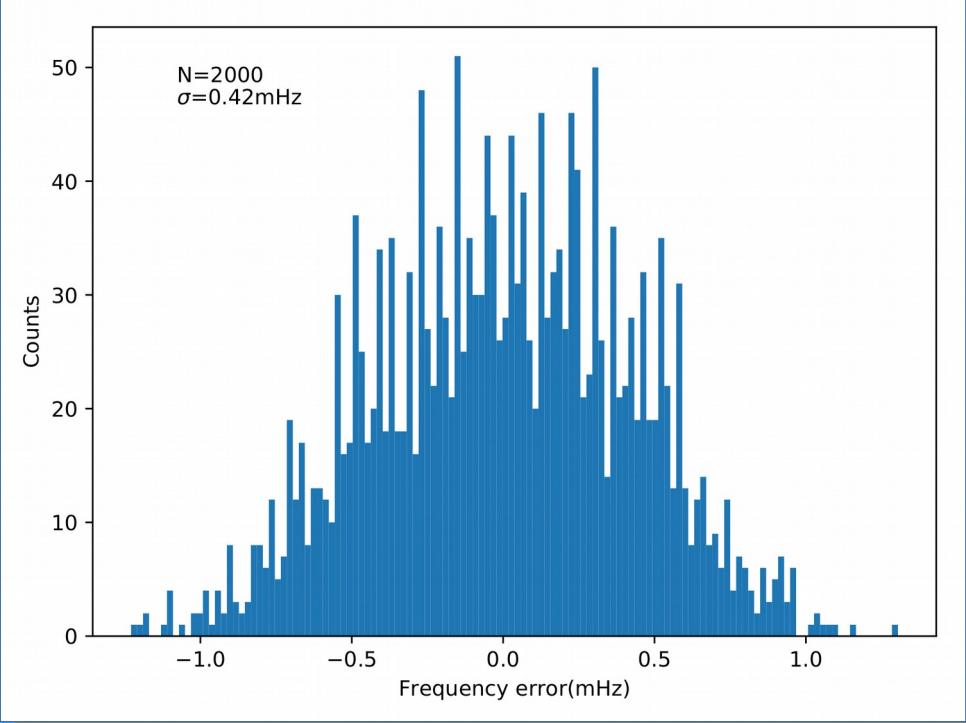
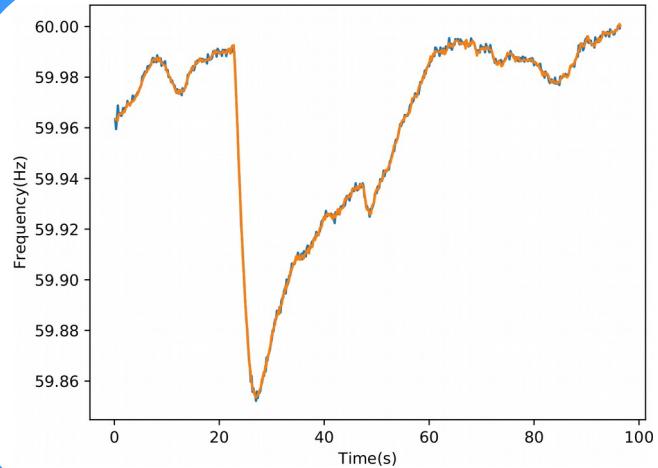
# Amplitude( $V_{rms}$ )



Validation against sdg1025



# Frequency $f_{\text{fundamental}}$

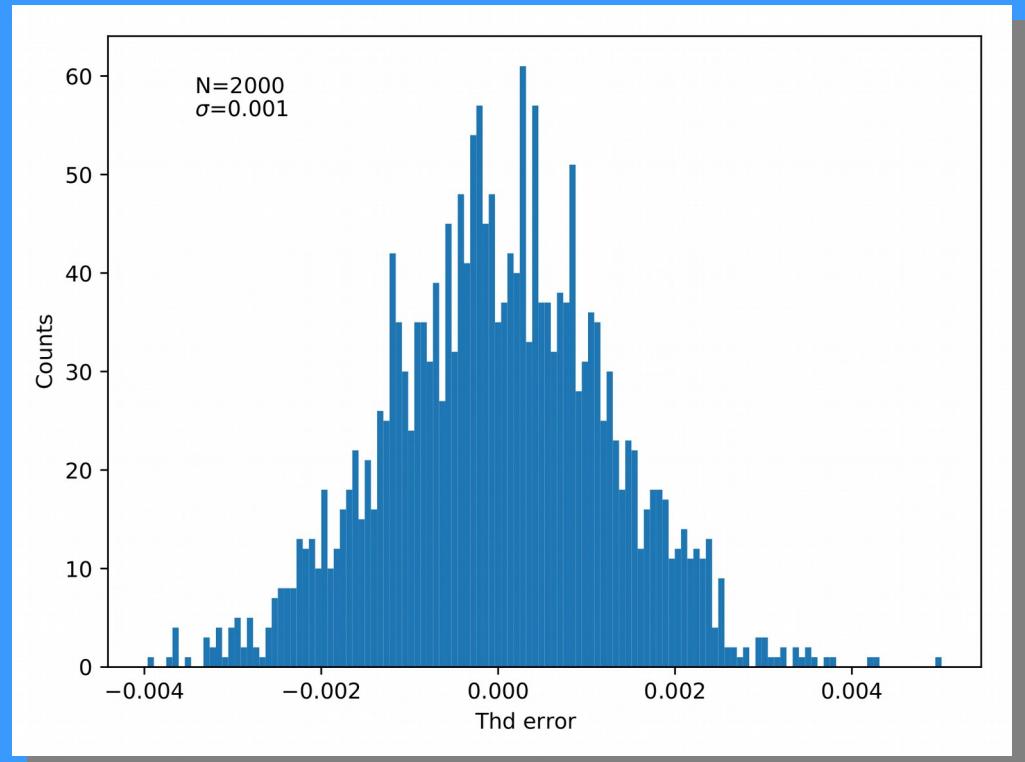
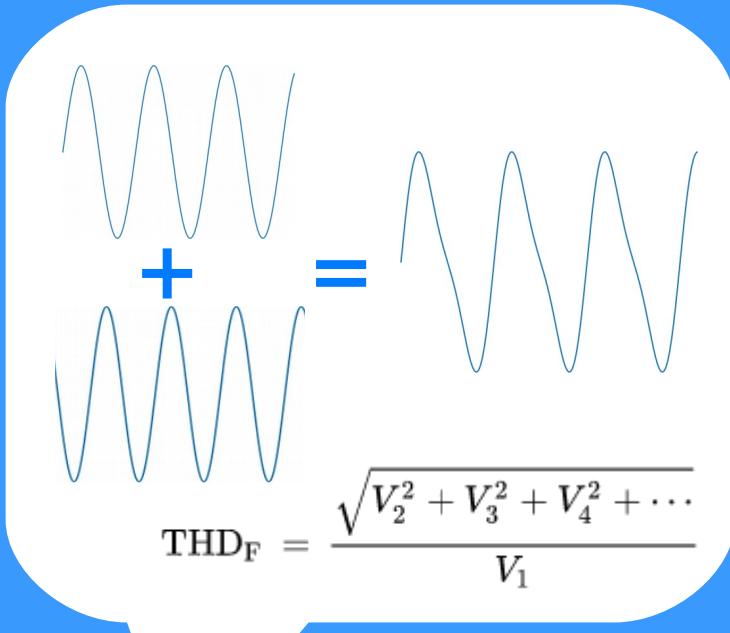


Validation against sdg1025



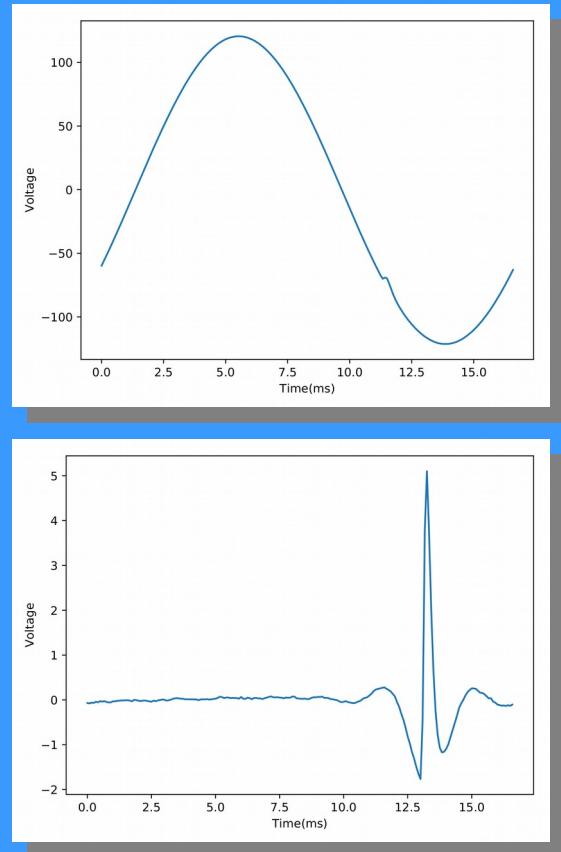
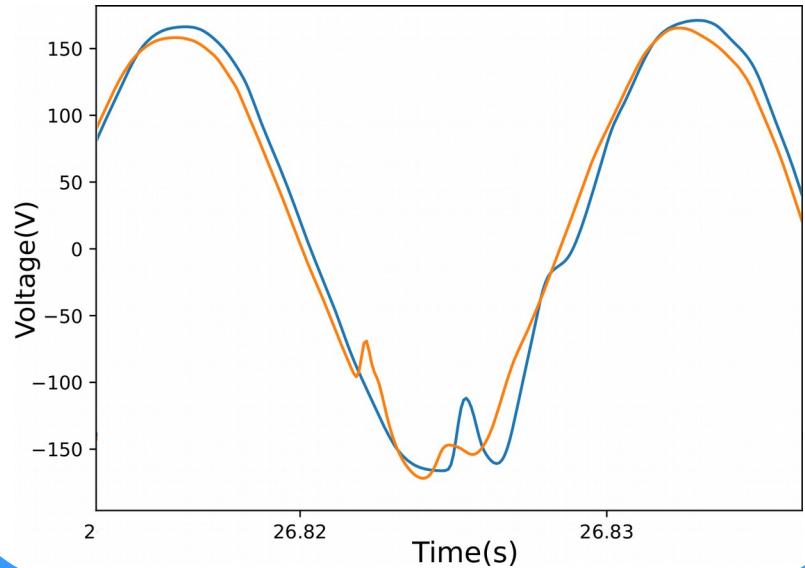
# Total Harmonic Distortion

## THD



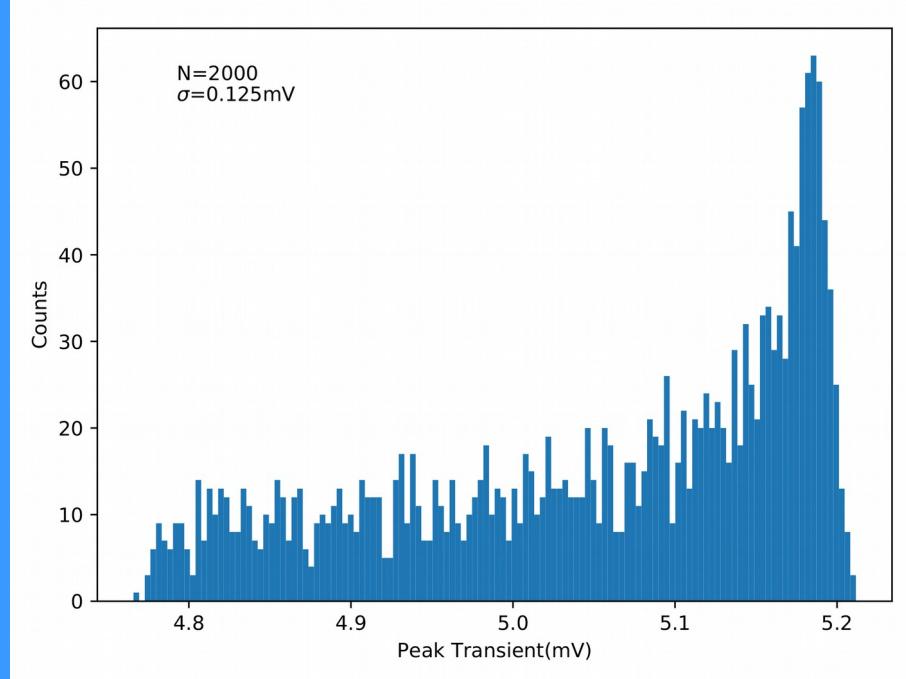
Validation against sdg1025

# Transients



# OPQ Box

## Evaluation of Transients.



Validation against sdg1025

# Napali Device Model

**For each device:**

- mean
- std

**Leaky mean:**

$$\mu_{n+1} = (1 - \alpha) * \mu_n + \alpha * m$$

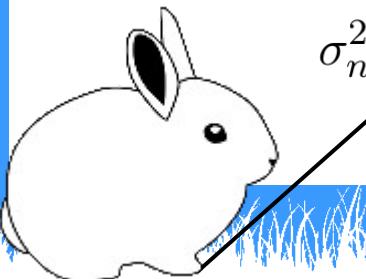
$$\mu_{n+1}^2 = (1 - \alpha) * \mu_n^2 + \alpha * m^2$$

$$\sigma_n^2 = \mu_n^2 - (\mu_n)^2$$

$$\sigma_n = \sqrt{\sigma_n^2}$$

**Event chronology:**

1. Any device passes threshold.
2. Record any device  $> 3*\text{std}$ .
3. Wait until all devices are below threshold.
4. If device count is  $> 1$  request data from
  - Devices which passed threshold.
  - Devices with a metric  $> 3*\text{std}$ .
5. If device count == 1:
  - No request is made.
6. Return to monitoring.

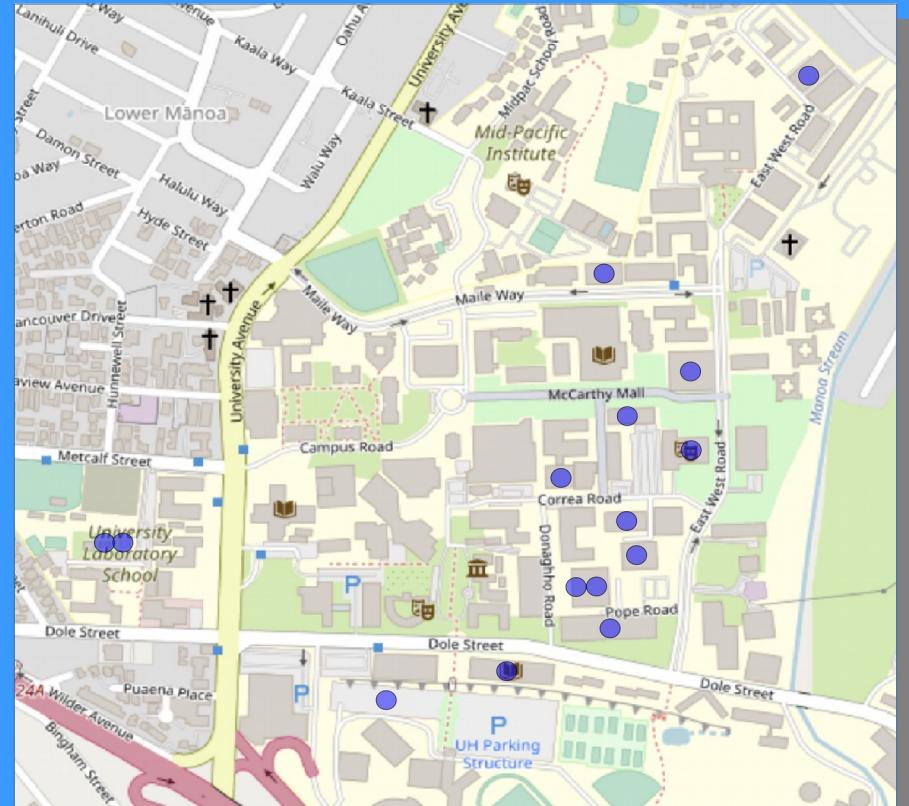


# UH Deployment

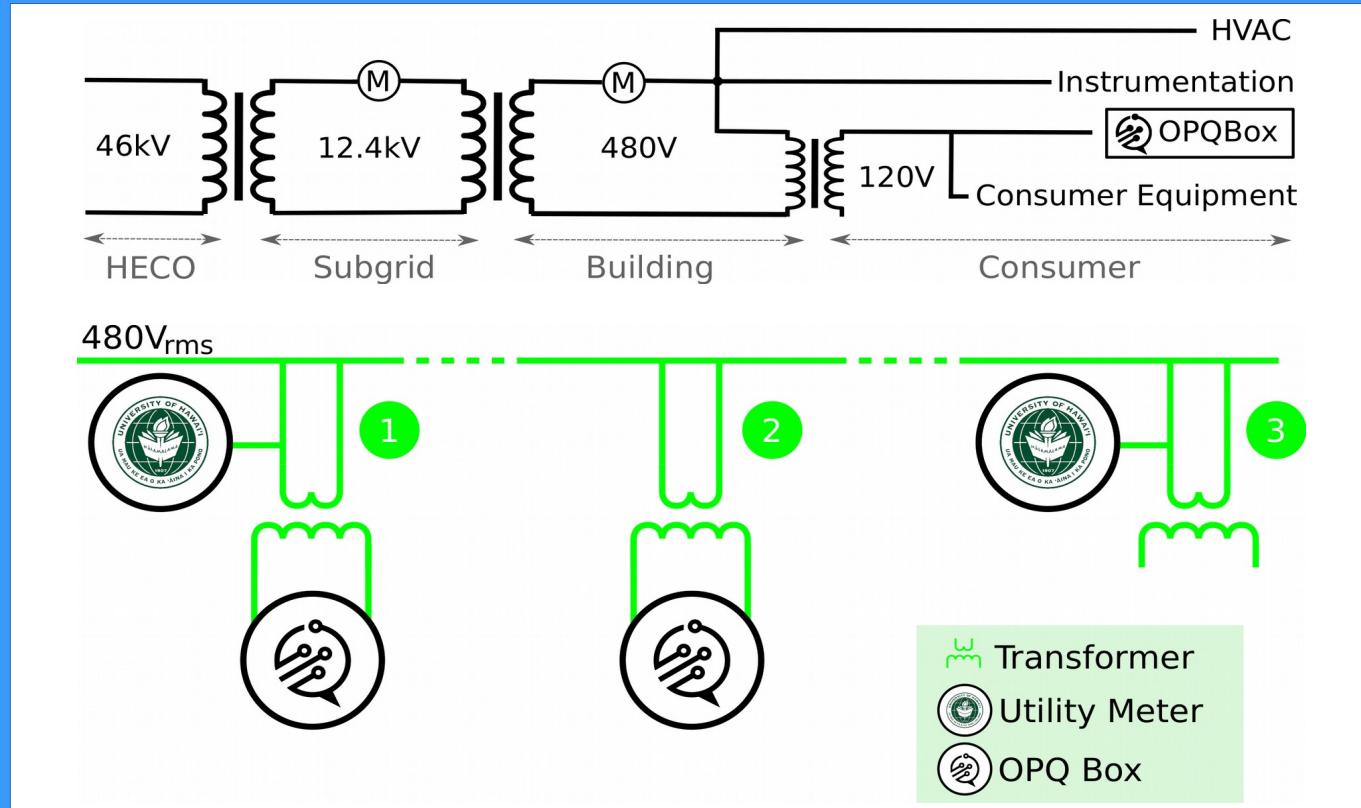
## in-situ validation

- 15 Devices.
- 3 months of data.
- Still running.

Anyone want a PQ network?

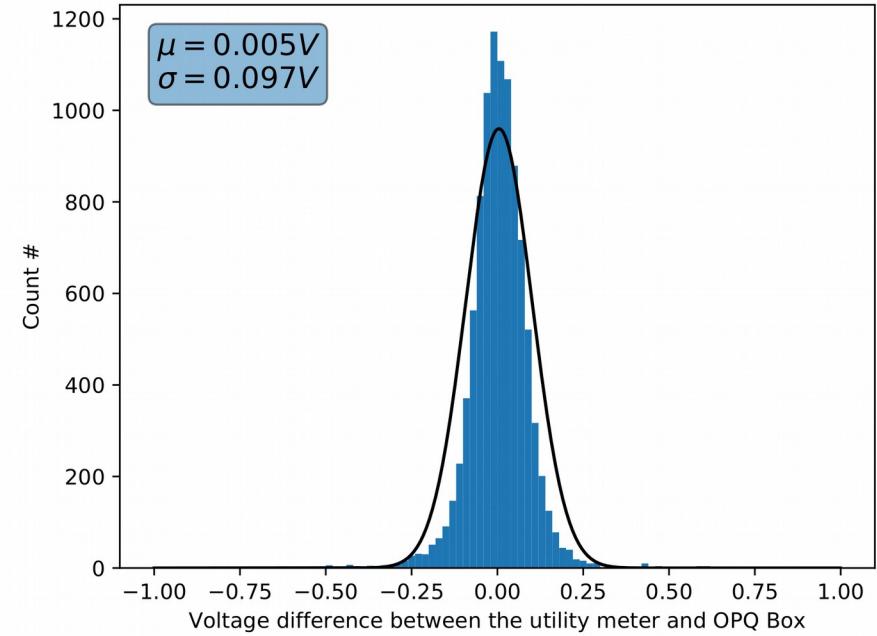
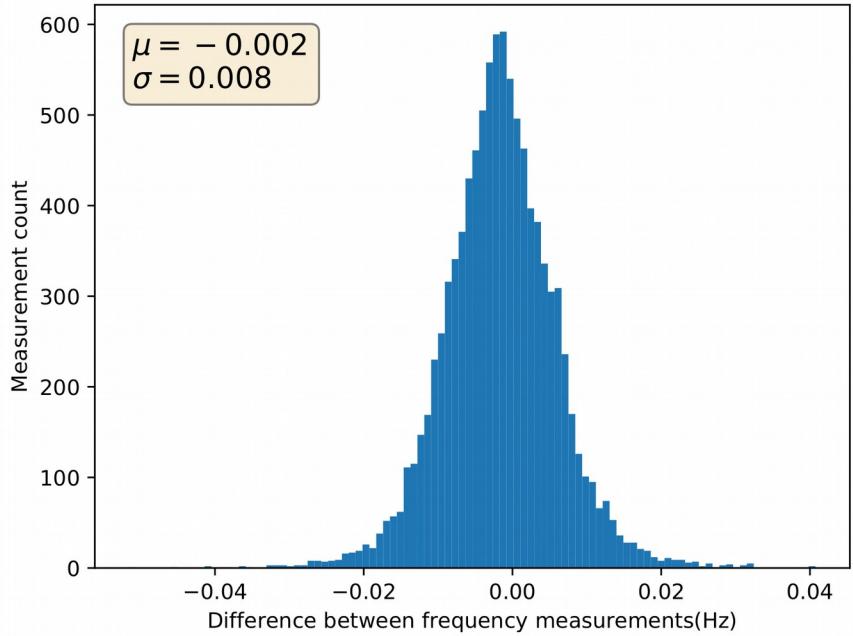


# UH Deployment



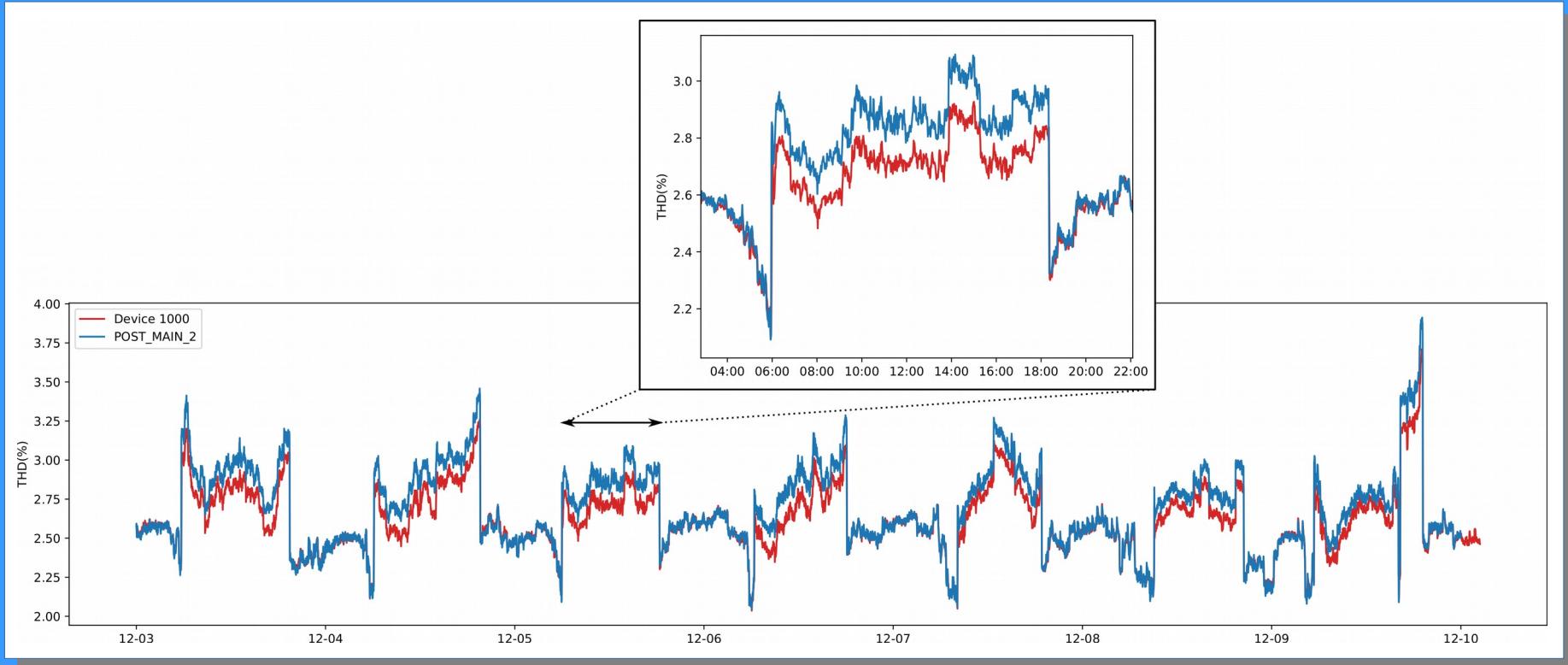
# OPQ Box Validation

Comparison with UH meters(frequency and  $V_{rms}$ )



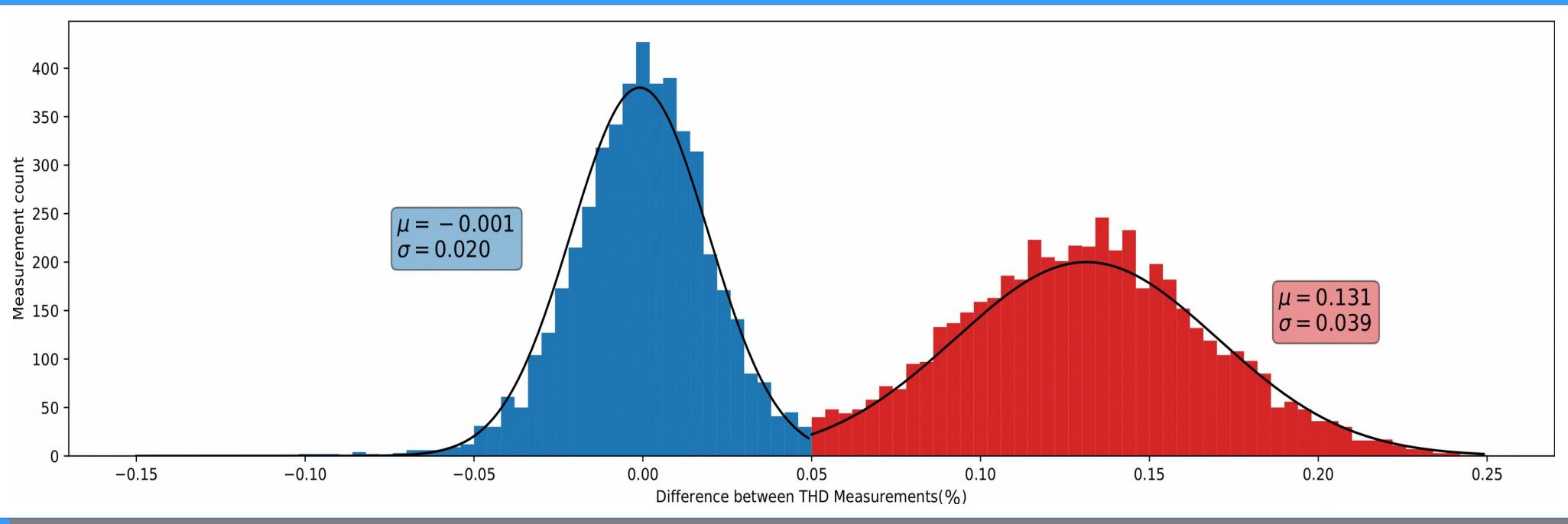
# OPQ Box Validation

## Comparison with UH meters(THD)



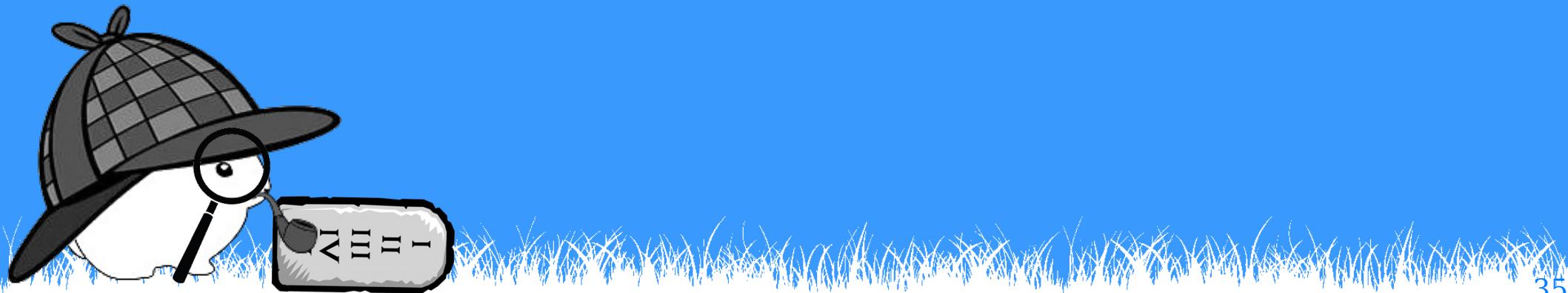
# OPQ Box Validation

## Comparison with UH meters(THD)

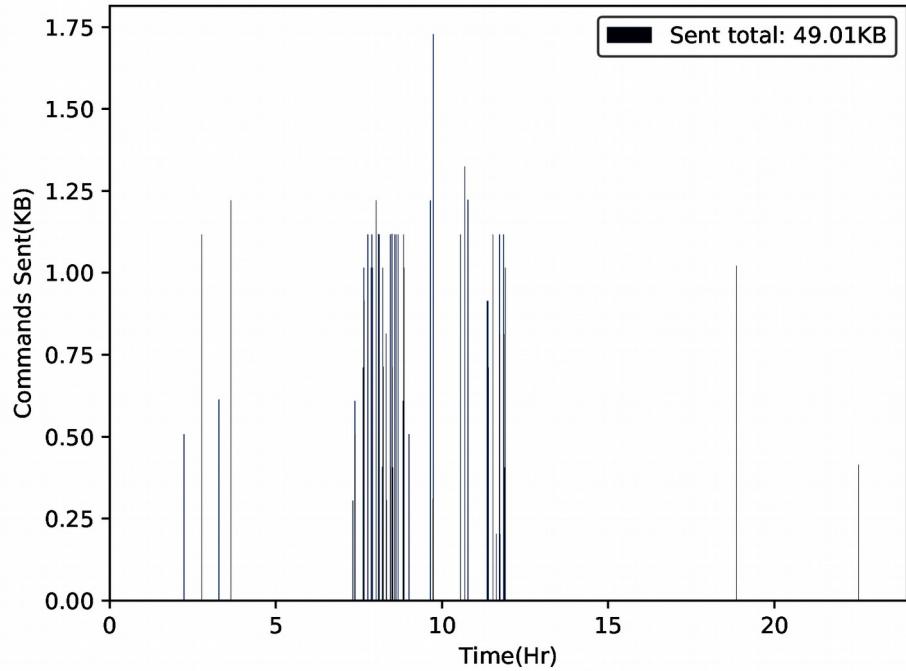
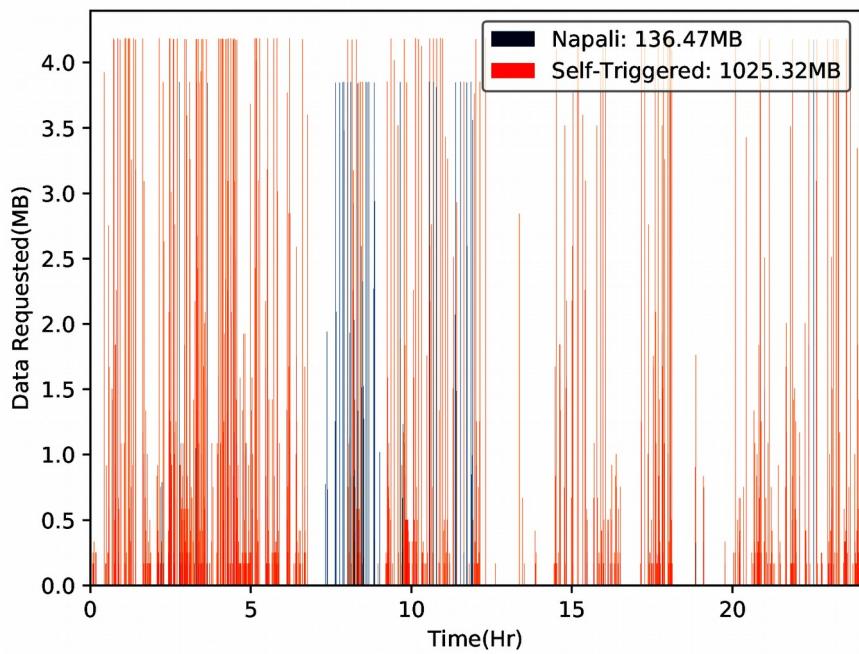


# Testing Claims

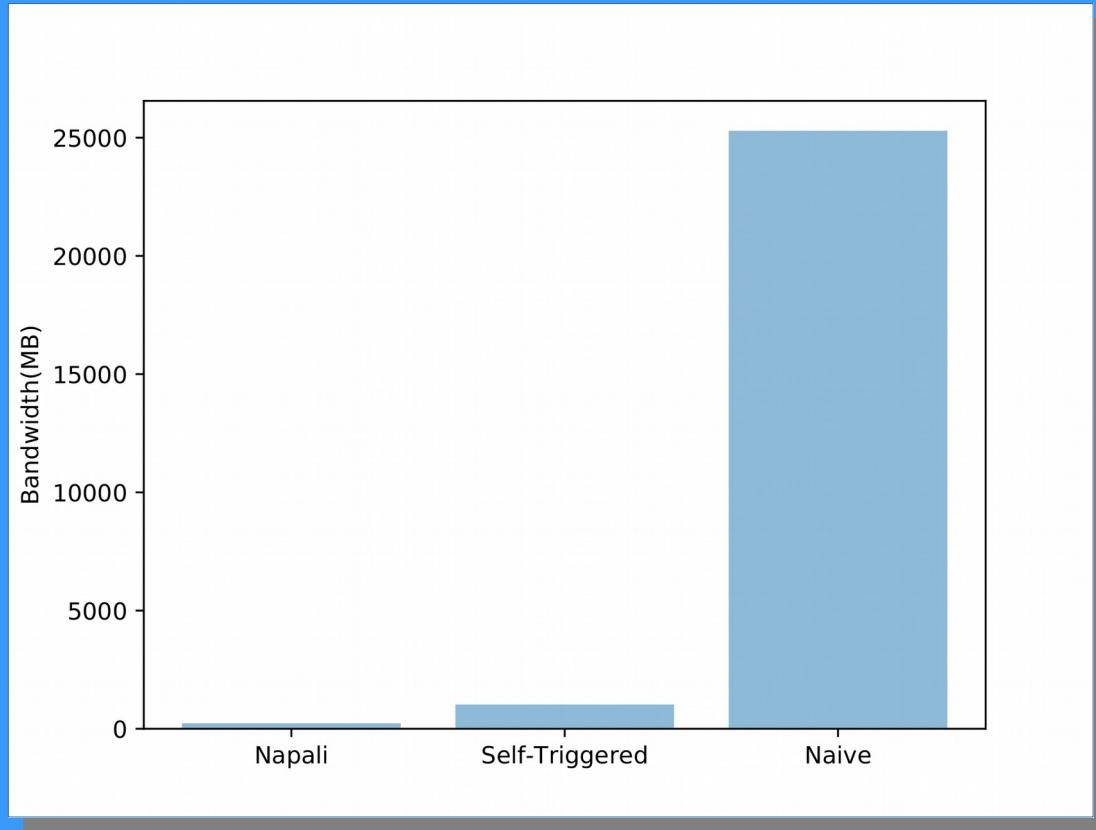
1. Napali minimizes bandwidth usage
2. Napali mitigates device latency effects
3. Napali minimizes sink processing requirements
4. Sub-threshold data acquisition is a viable event detection strategy
5. Temporal locality triggering results in a low false negative detection



# Claim: Bandwidth Usage



# Claim: Bandwidth Usage



*Exquisite!*



# Claim: Computational Cost cost at the sink

$$C_{total} = C_{metric\_extraction} + C_{detection}$$

---

Napali

0

10uS

Naivie

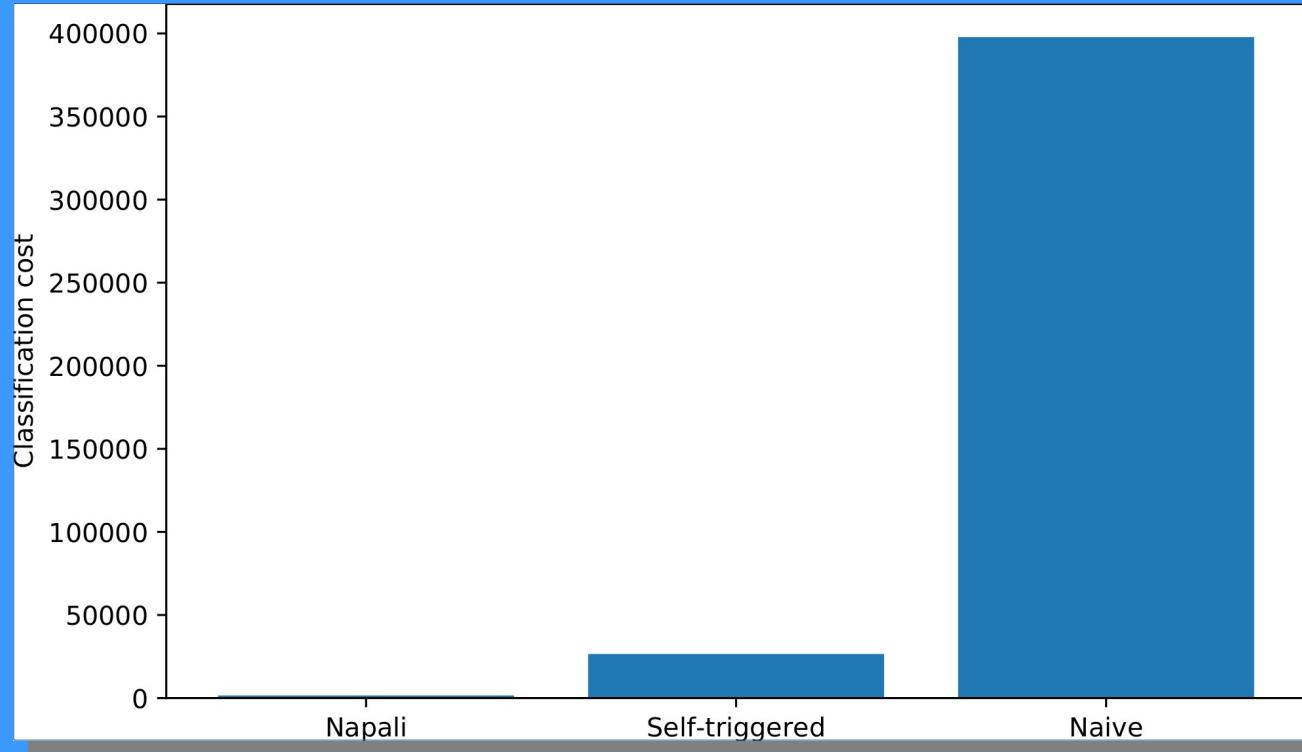
800uS

10uS

**1 device 1 second**

# Claim: Computational Cost

## classification cost



*Flawless!*

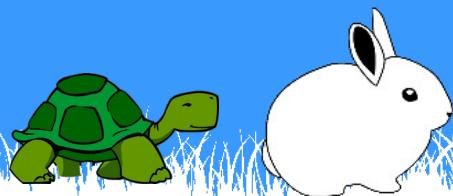


# Claim: Latency

Late metrics over 24 hours

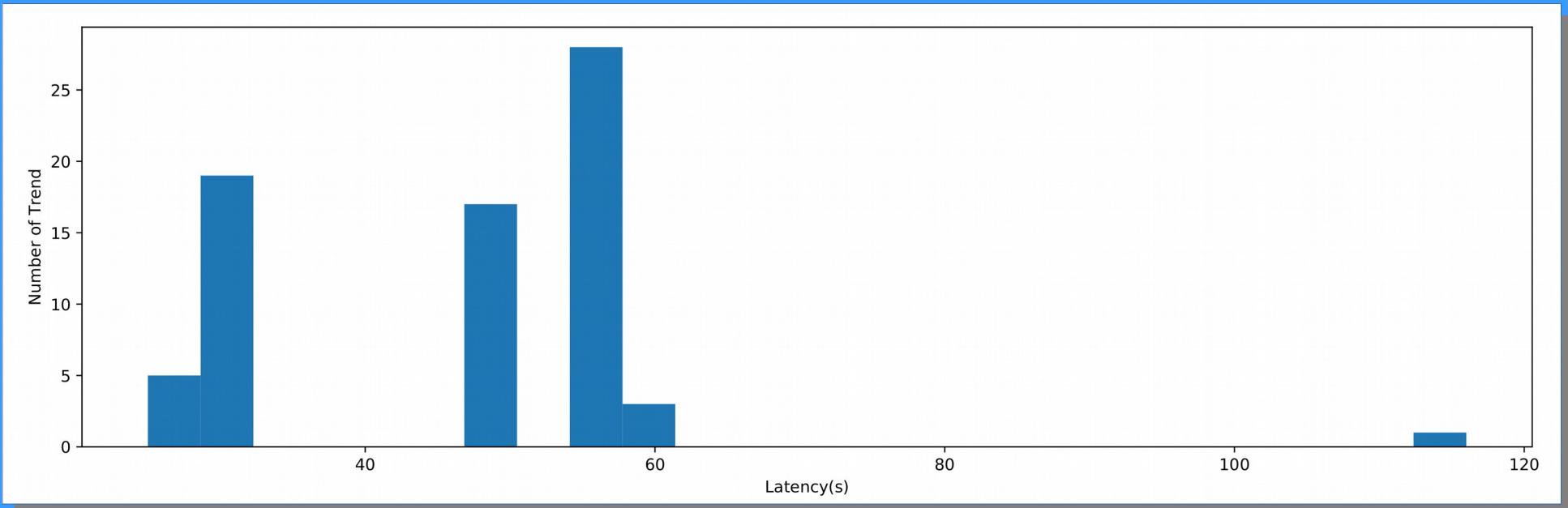
$$Buffer_{naive} = 2 * latency_{max} * (Datarate) * (\#ofdevice)$$

$$Buffer_{napali} = 2 * latency_{max} * (Metric\ rate) * (\#ofdevice)$$

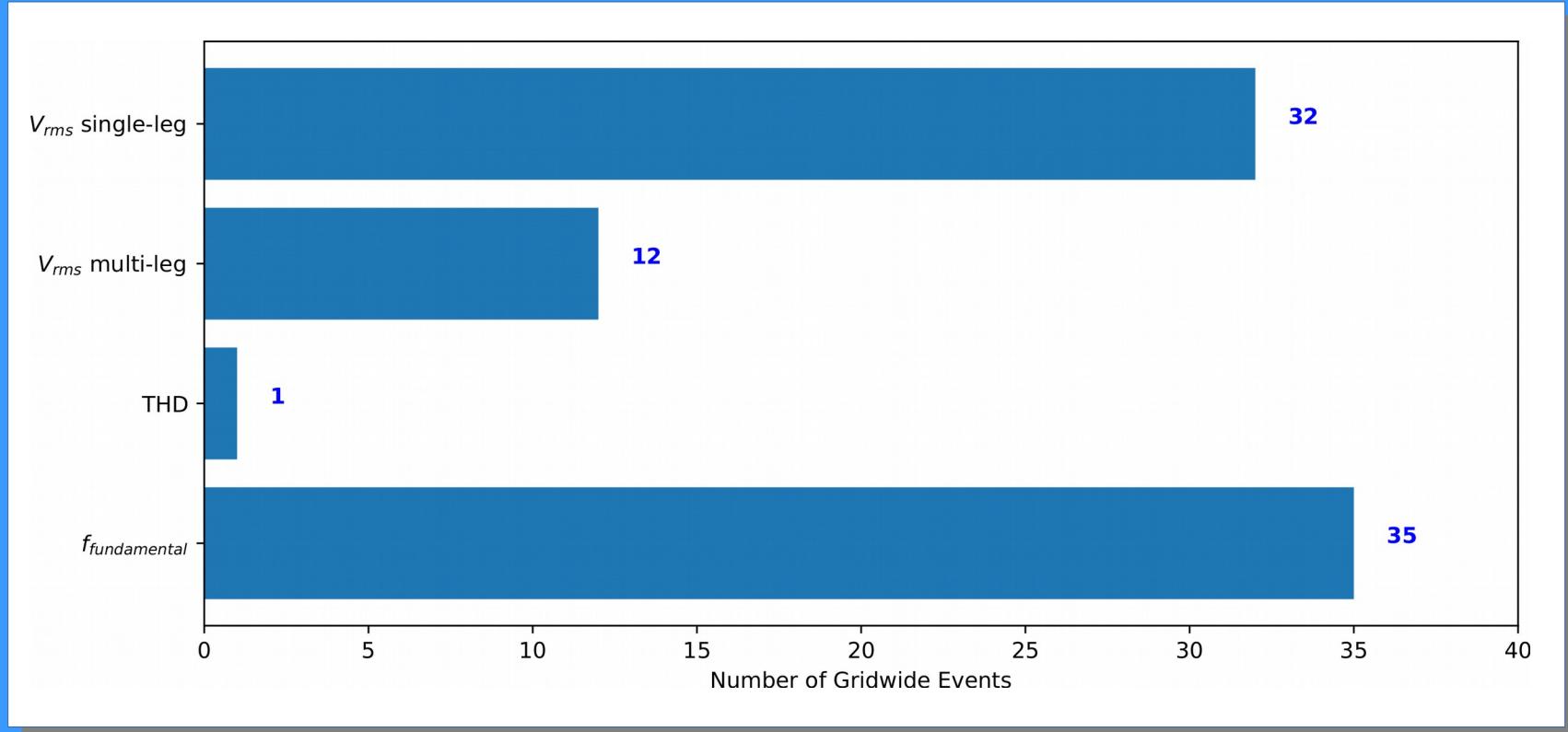


# Claim: Latency

Late metrics over 24 hours

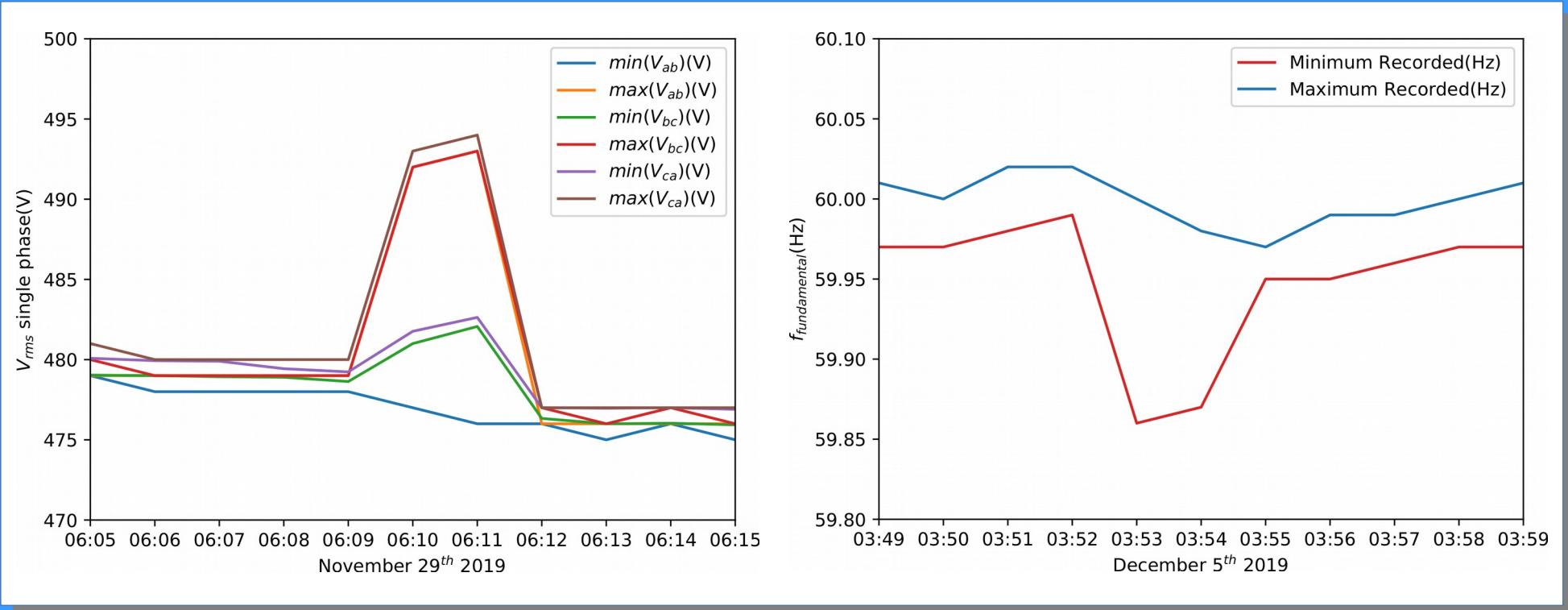


# Claim: Temporal locality



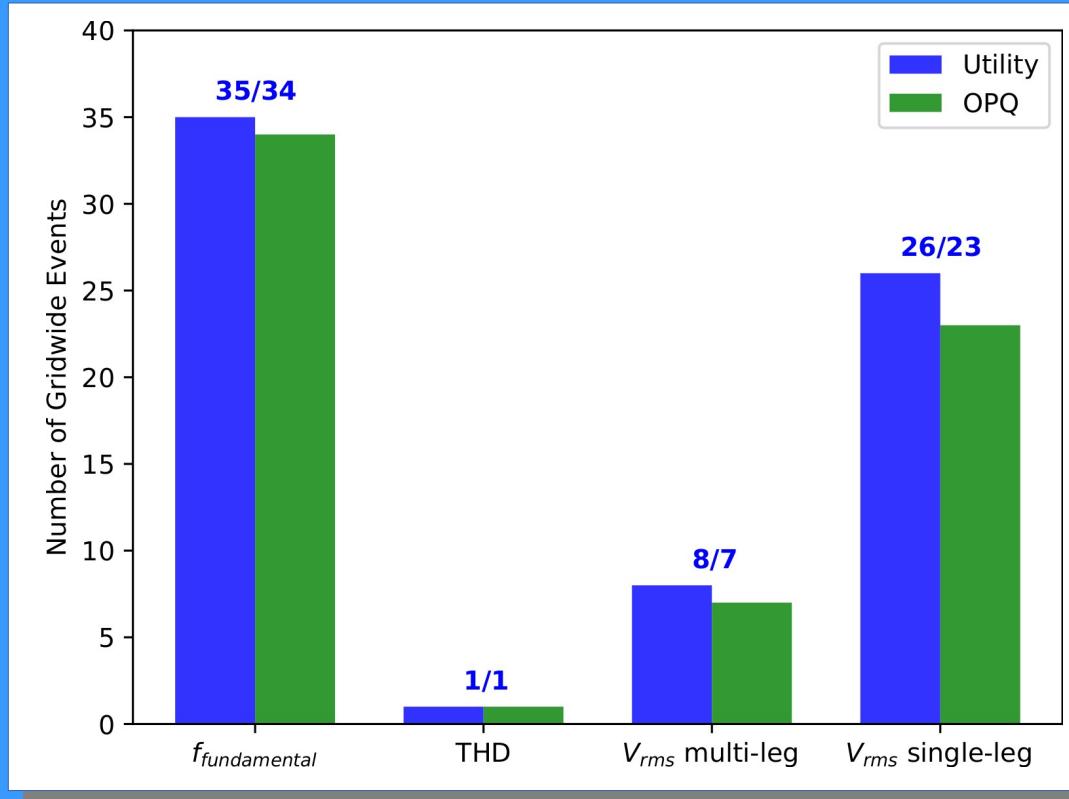
# Claim: Temporal locality

Ground truth event examples.



# Temporal locality triggering

## Detected events



*Marvelous!*



# Claim: Sub-threshold Events

1. Select events where:

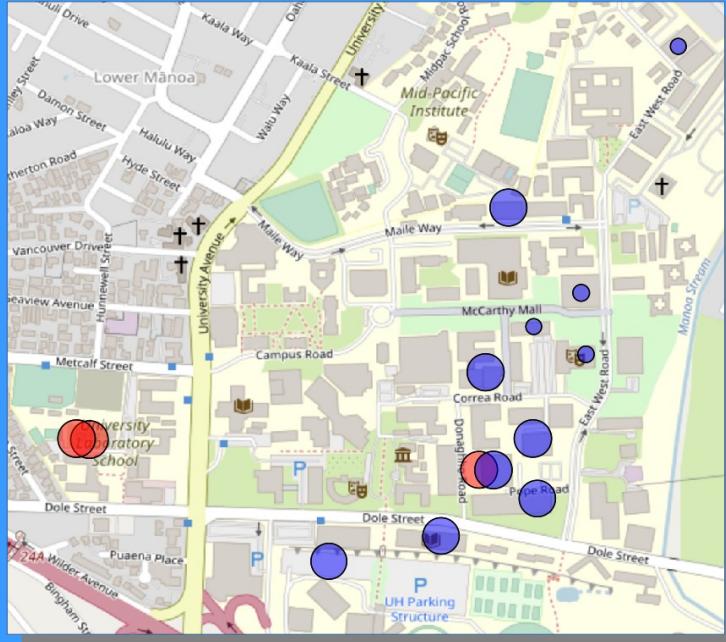
- single collocated utility meter passes threshold.
- one or more non-collocated utility meters passes threshold.

2. Compare the OPQ data with utility meters.

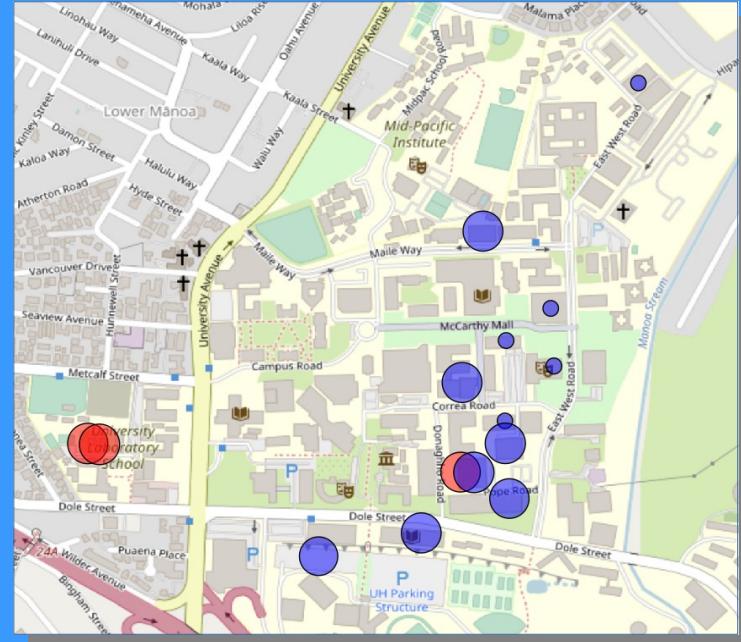


Only 3 events!

# Event #1 and #2

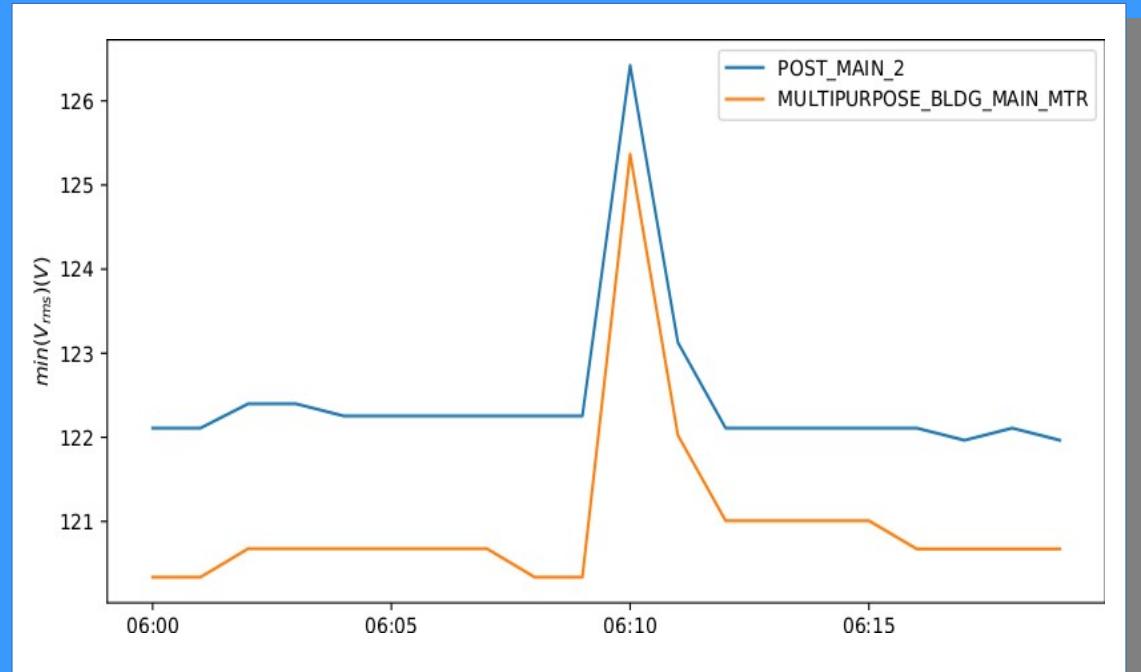
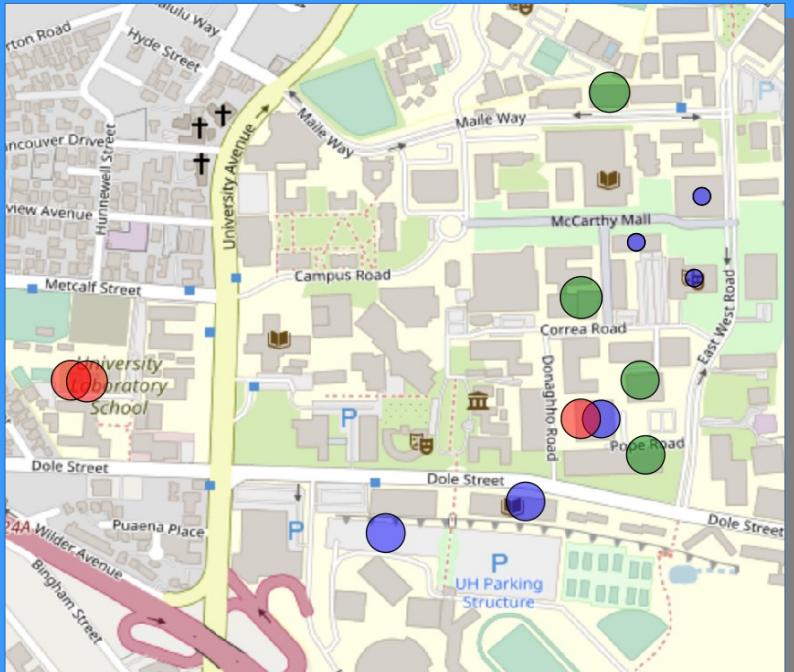


5 Utility Meters



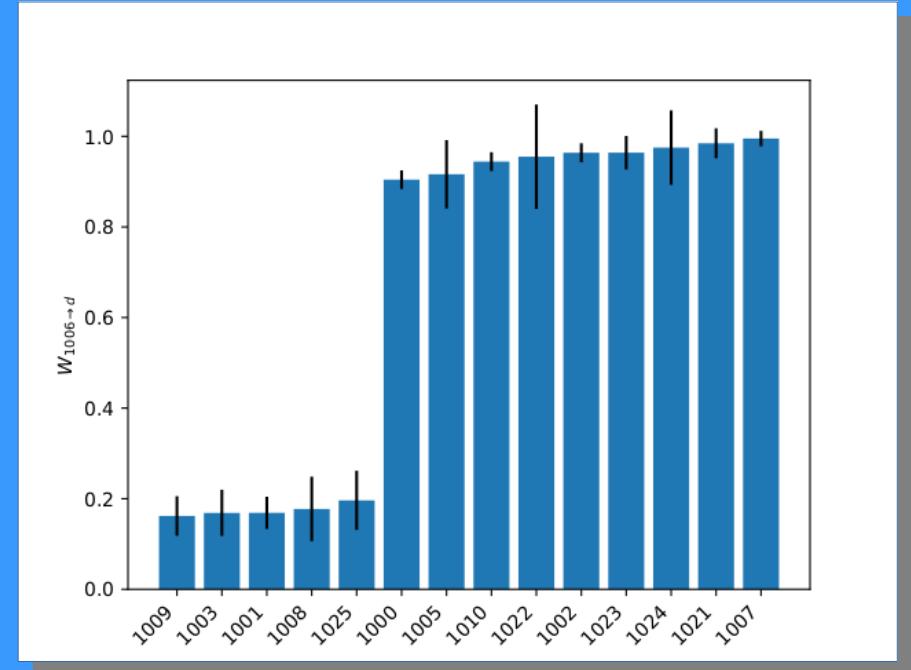
4 Utility Meters

# Event #3

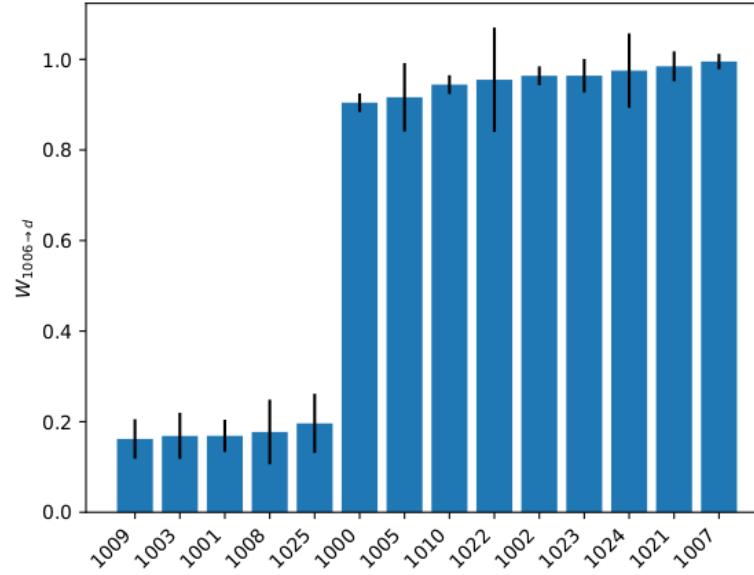
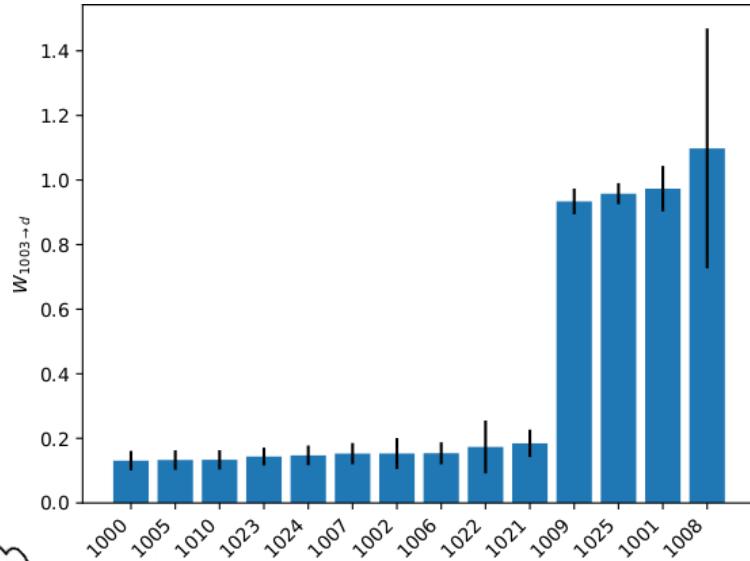


2 Utility Meters

# Event Clustering



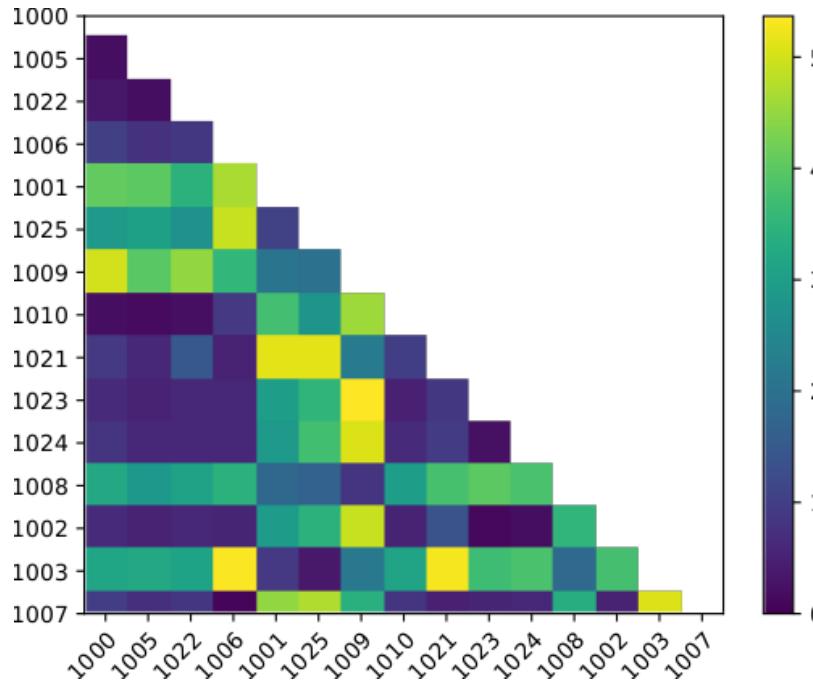
# Common Event



# Pairwise Dissimilarity

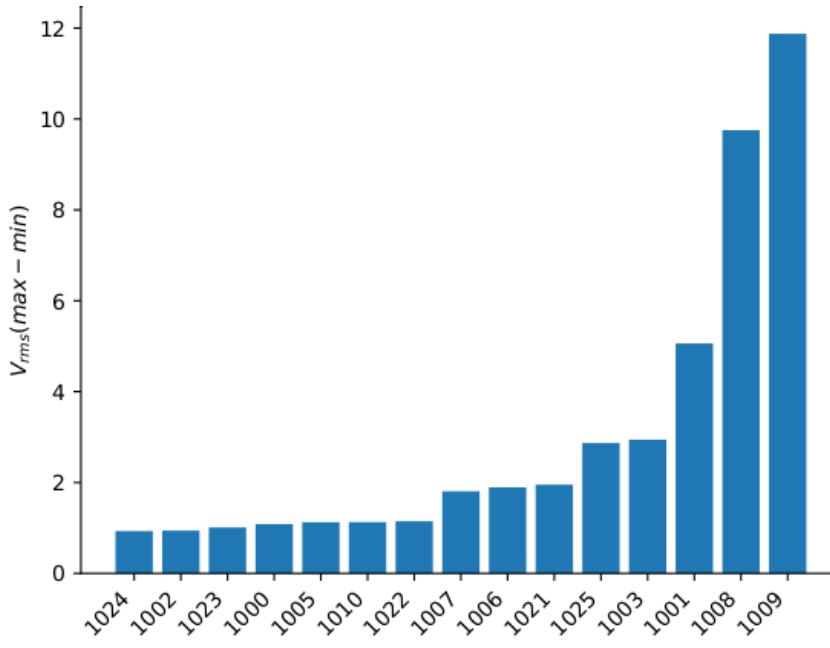
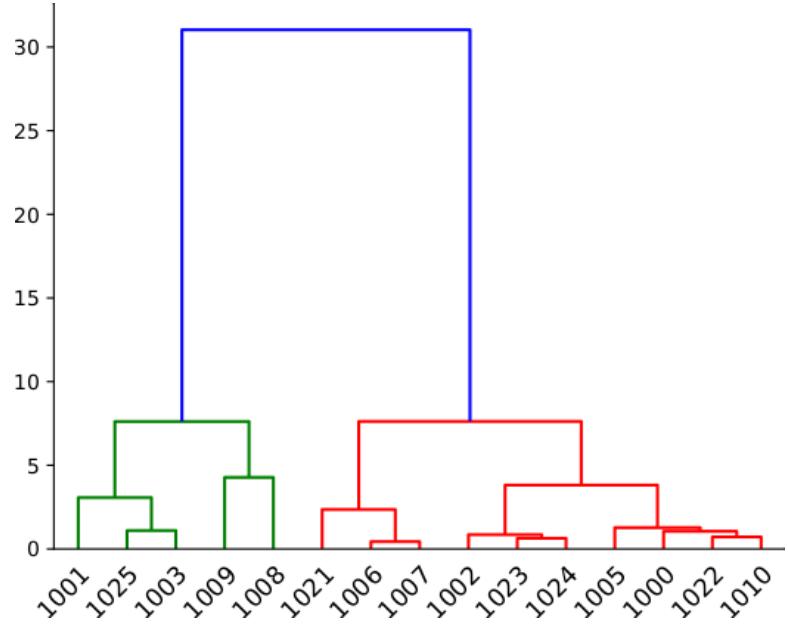
$V_{rms}$

$$D_{p \rightarrow d}^{rms} = \frac{1}{n} \sum_n |(\max(V_{rms}) - \min(V_{rms}))_p - (\max(V_{rms}) - \min(V_{rms}))_d|$$



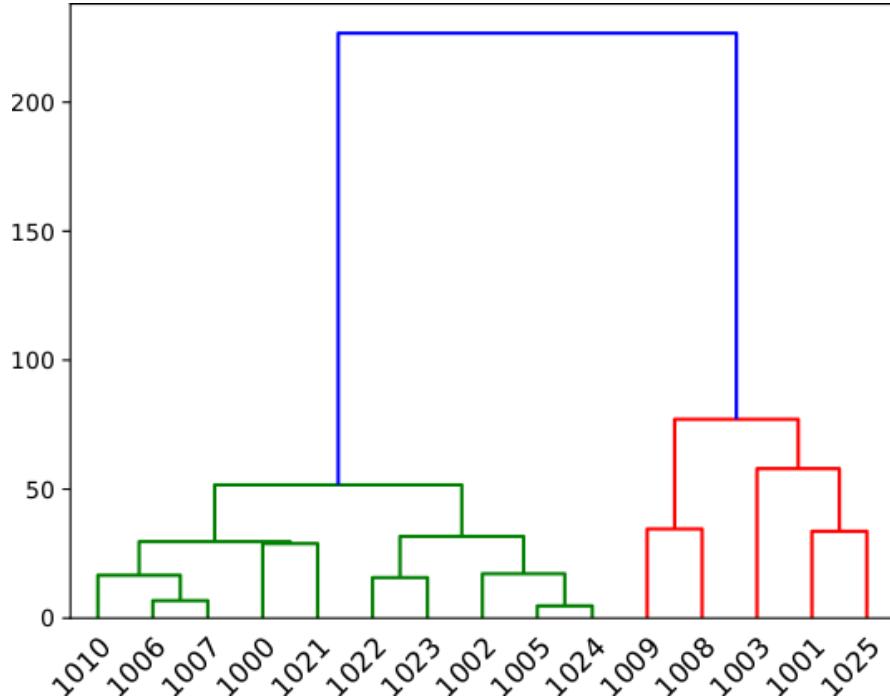
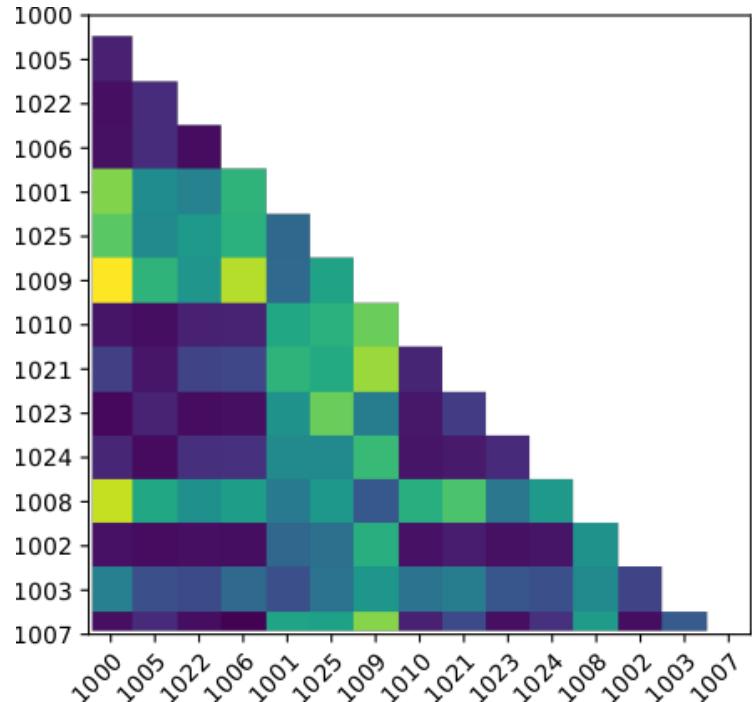
# Clustering

$V_{rms}$



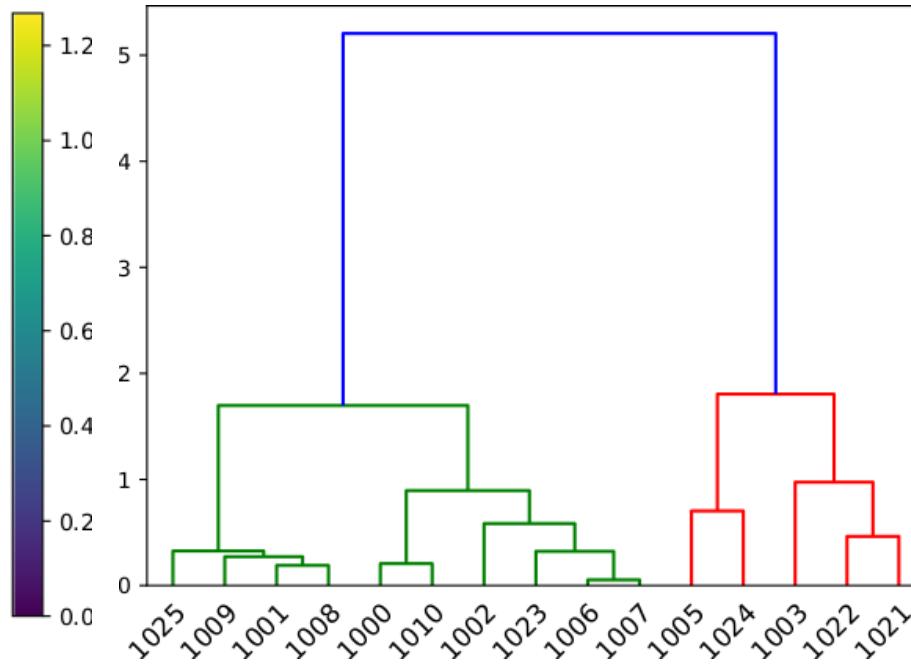
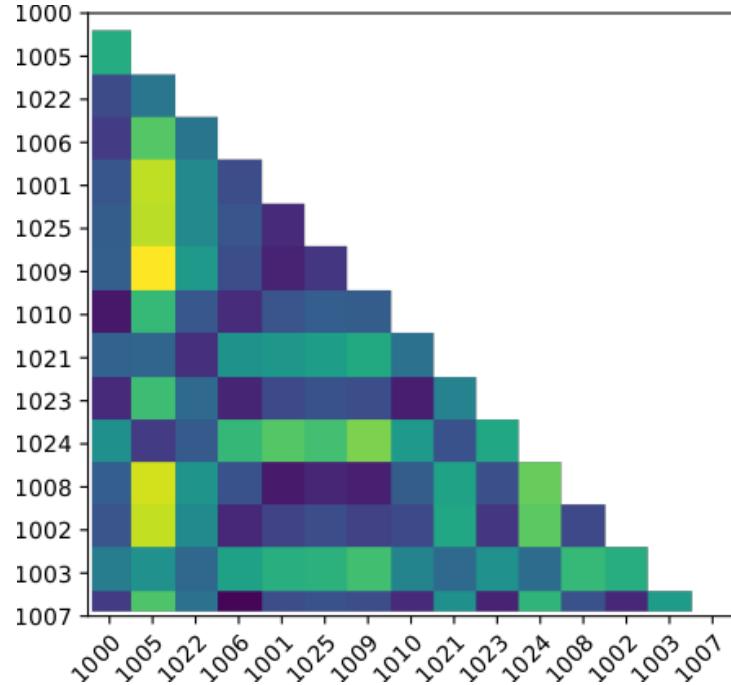
# Clustering

## Transients



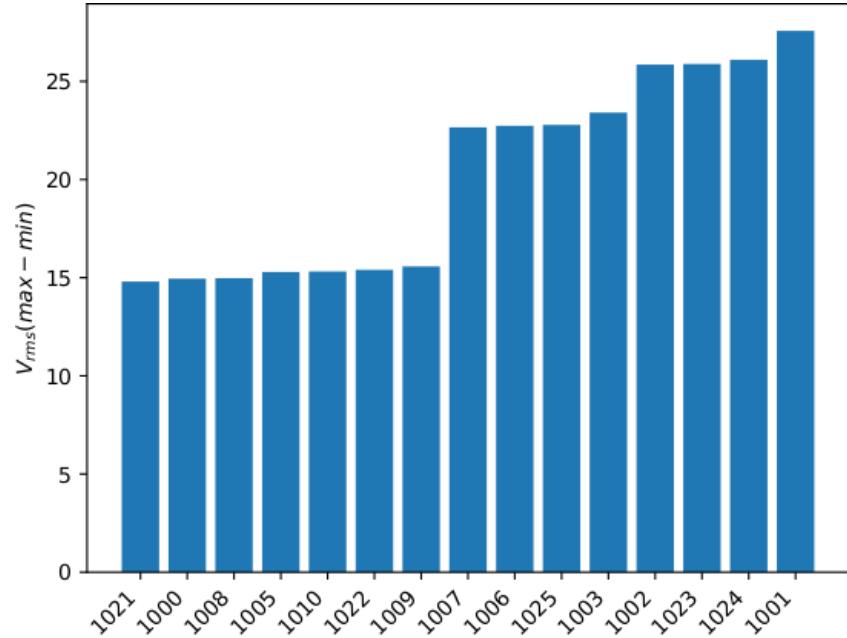
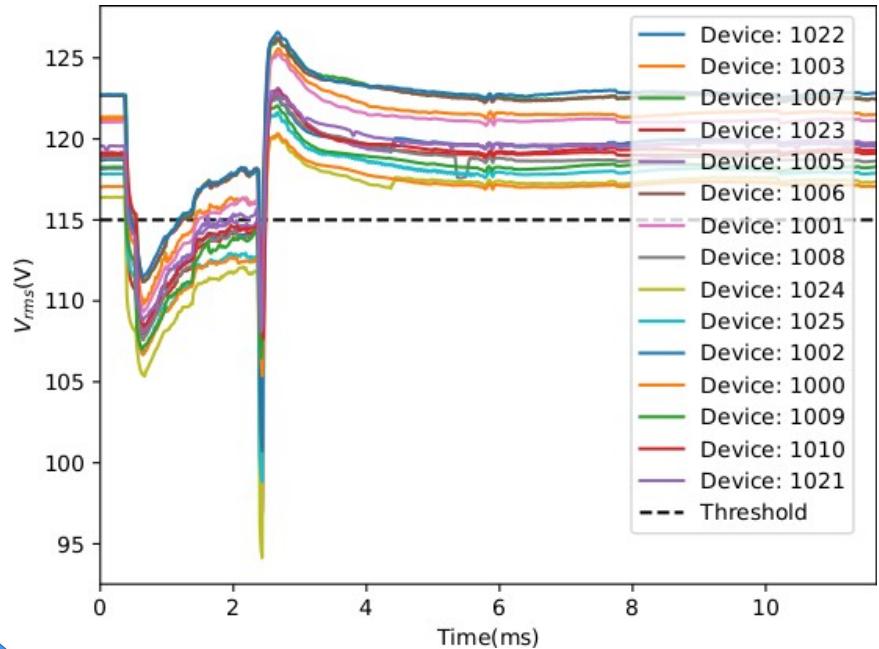
# Clustering

## THD



# True Gridwide Events

## THD



# Sub-threshold Triggering

1. Partitioning of the power grid.
2. Localization of anomalies.
3. Napali scalability improvements.



# Napali in other domains

## Lightning detection

### Current System:

- Remote area deployment
- Satellite communication

### With Napali:

- Metropolitan Area
- Abundant communication



# Napali in other domains

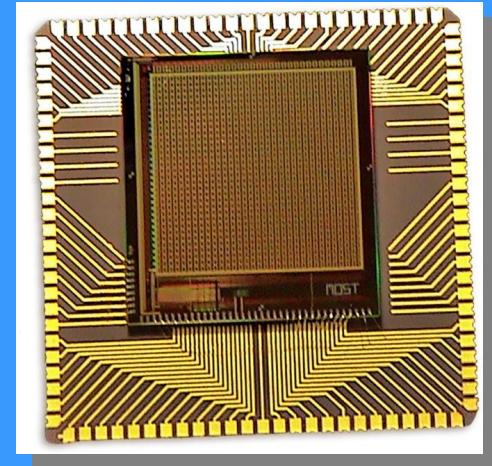
## Gunshot Detection

### Current System:

- No geographically distributed system in literature.

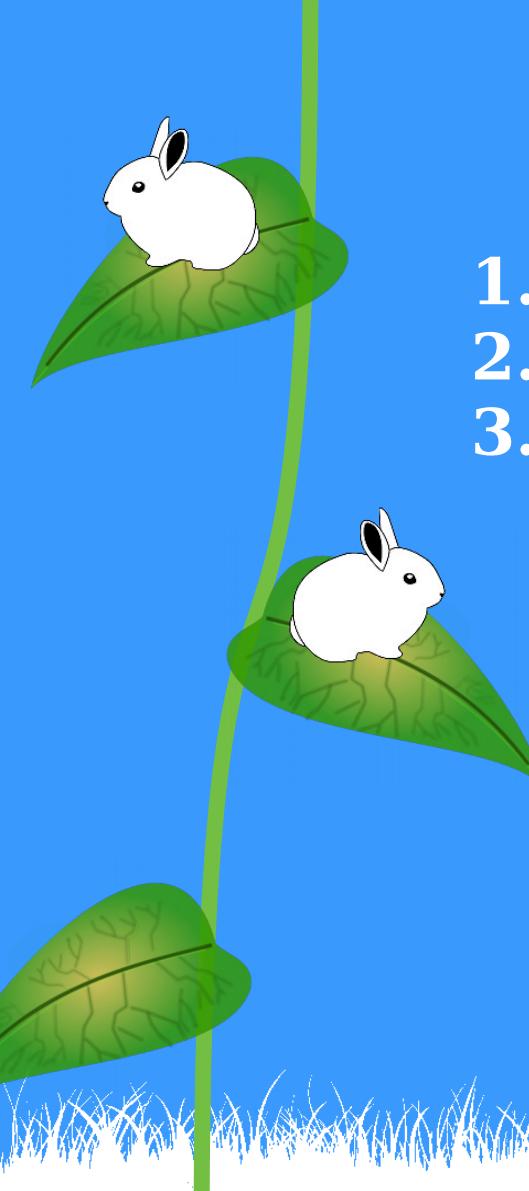
### With Napali:

- Metropolitan area coverage.



MOST Sensor  
Oceanit 2017

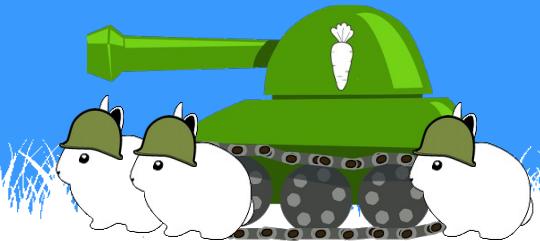
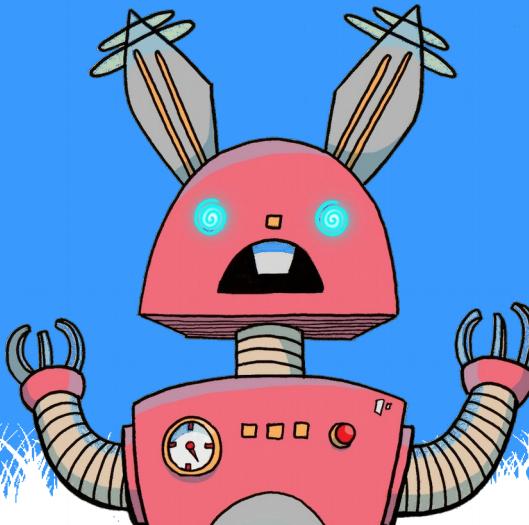
# Contributions

- 
- 1. OPQ Box:** low cost distributed PQ meter
  - 2. Napali:** Scalable performant event detection
  - 3. Sub-threshold Events:**
    - Grid partitioning
    - Event localization

**Wide potential for other fields.**

# Future Work

- 1. AI instead of statistical model**
- 2. Scalability study**
- 3. Privacy study**
- 4. Power outage resiliency**
- 5. Integration with Utility**





Turtle and Fishy



Tiger  
2012-2019

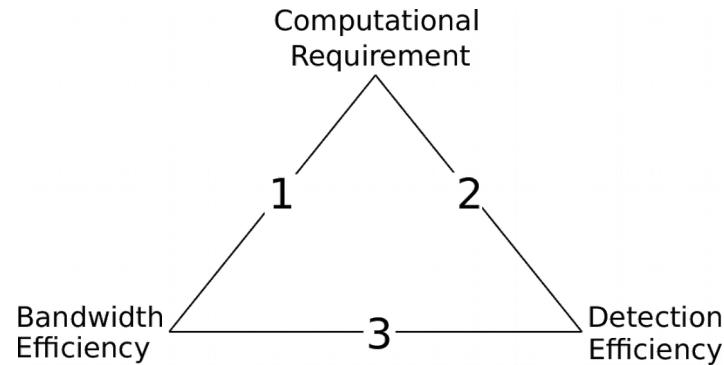


Thanks!



# Comparison From Sensors Perspective

*Pick Two!*

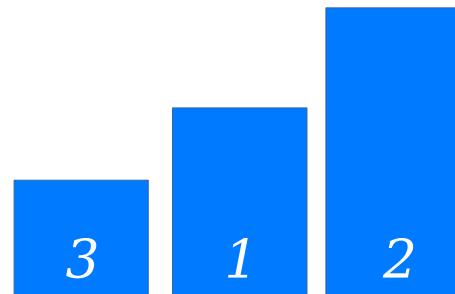


1. Self-Triggered
2. Naive
3. Napali



# Comparison From Sink Perspective

*Computational Cost:*



1. Self-Triggered
2. Naive
3. Napali



# **Additional Benefits of Napali**

## **Outside the scope of this thesis**

- 1. Increased flexibility with respect to privacy protection.**
- 2. Increased resiliency with respect to power failure**

