Correlative Monitoring for Detection of False Data Injection Attacks in Smart Grids

Michael Kallitsis (mgkallit@merit.edu)
joint with George Michailidis, Samir Tout

IEEE SmartGridComm 2015, Miami, Florida

Agenda

- Introduction - Problem Motivation

False Data Injection: a malicious actor injects "bad data" into the payload of a smart meter

False Data Injection

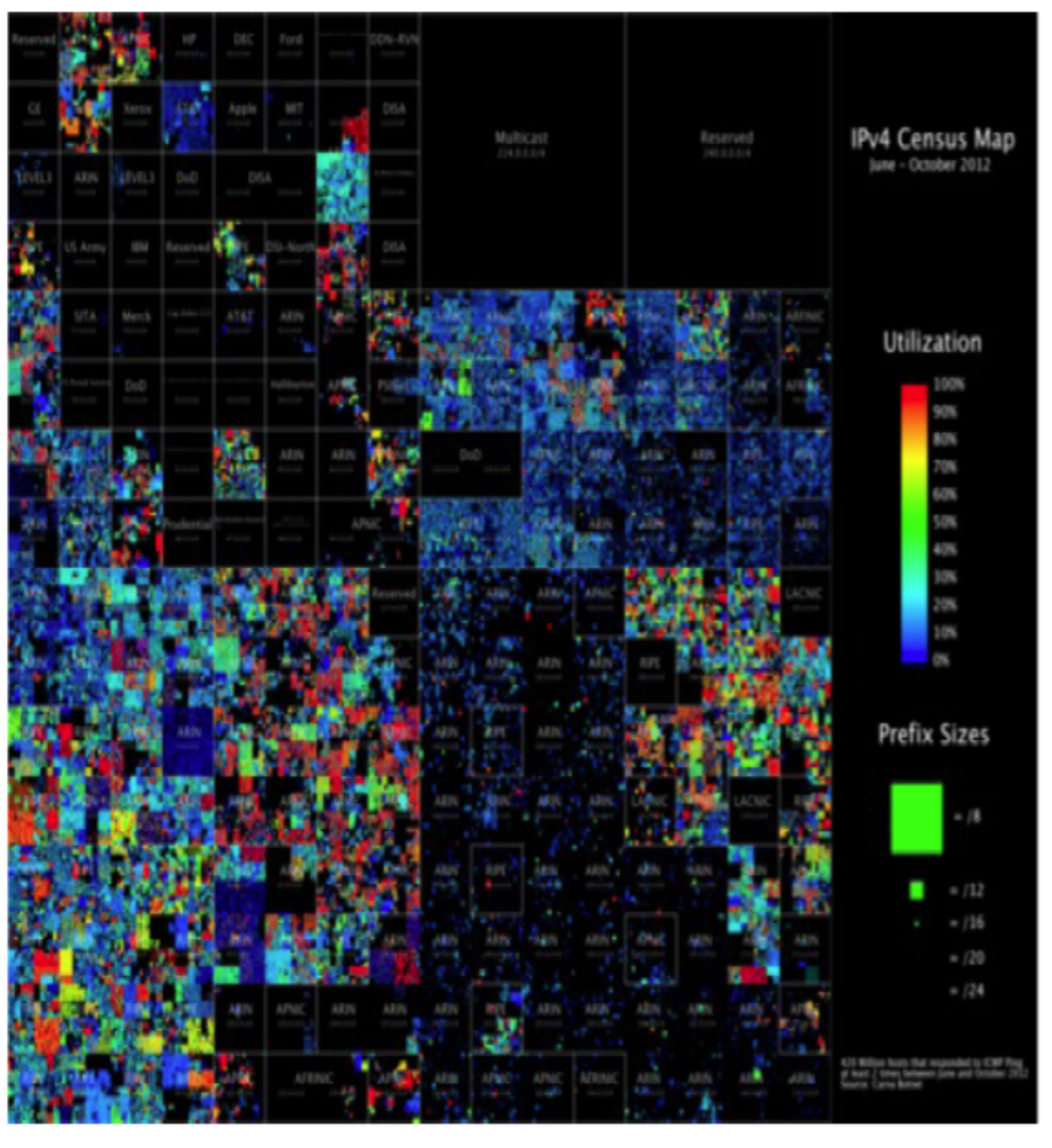
- Consequences
  - Destabilize grid (deteriorates grid's estimation process)
  - Endanger demand response schemes
  - Compromise operation of intelligent buildings
  - Energy theft and price manipulation
- Threat Model — Attack scenarios
  - Malware coordinating instantaneous demand drop
  - Nodes programmed to reduce and suddenly increase power demand

Smart Meter Vulnerabilities

- Rapid deployment of smart meters entails installing low-cost commodity embedded devices in physically insecure locations with a lengthy operational lifetime (several decades)

Attacks on Embedded Systems

- Stuxnet worm: damaging physical infrastructure
- DDoS report from Arbor Networks: most attacks spawned by embedded systems (e.g., home routers)
- Carna botnet: Internet census from compromised home routers!



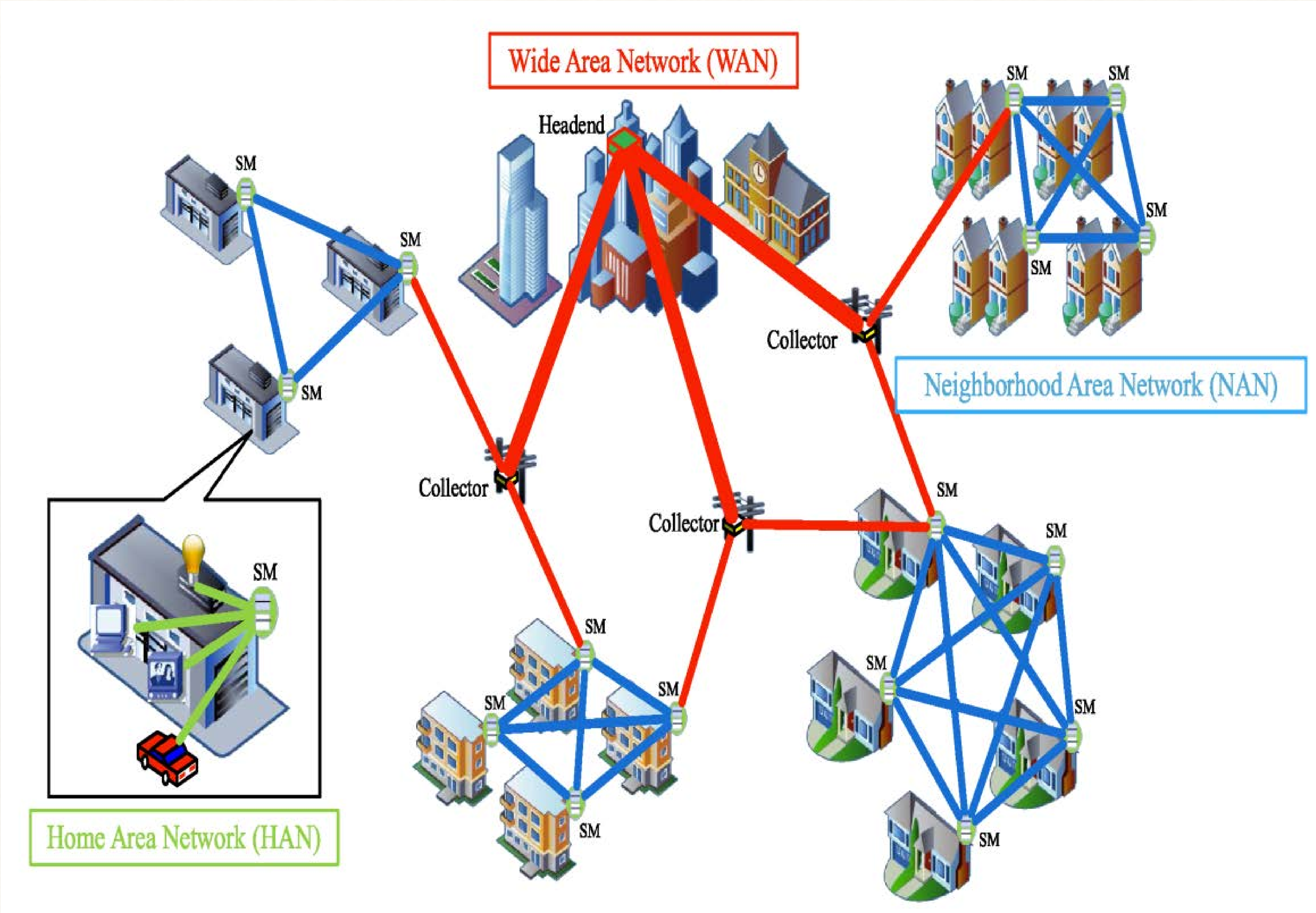
Related Work in Smart Grid Anomaly Detection

- Signature-based Methods (e.g., Snort, Bro)
  - Needs signatures, could miss polymorphic malware
- Specification-based Detection
  - Data validation, range checks: can be cumbersome to fine-tune
- Behavioral-based Techniques
  - Statistical based: classification/clustering
  - State Space techniques
  - Graphical based
  - Game theory methods (price manipulation)

Network-view perspective

AMI Architecture — Bottom-Up Approach

- Home Area Network (HAN)
  - Smart meter & appliances
  - Lightweight communication protocols (WiFi or ZWave)
- Neighborhood Area Network (NAN)
  - Aggregates data from HAN meters
  - Long-range wireless communications (e.g., cellular)
- Wide Area Network (WAN)
  - Connects the utility to NANs and data concentrators



Agenda

- Introduction - Problem Motivation
- Correlative Monitoring Approach

HAN Monitoring - Modular Approach

Data Collection

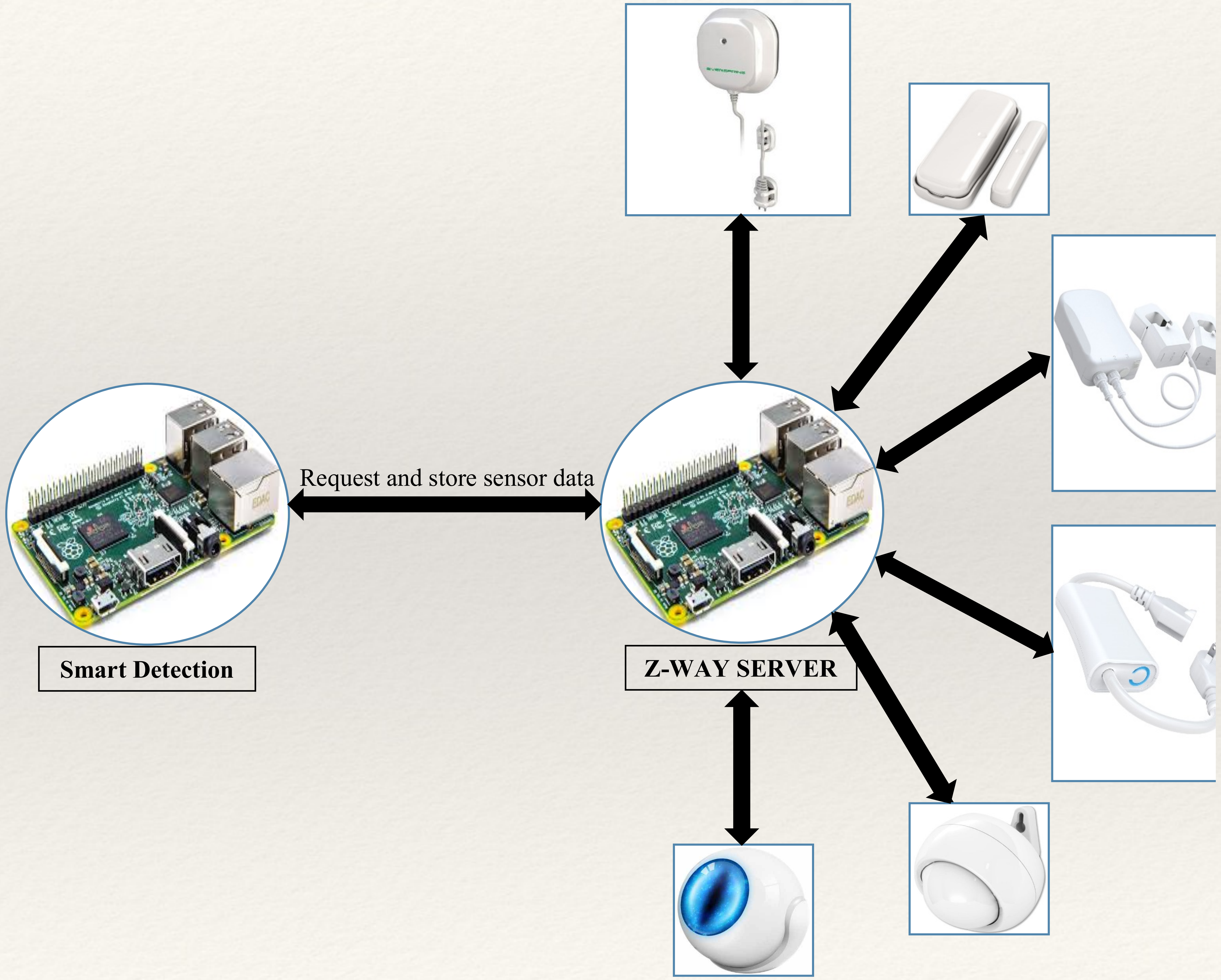
Forecasting

Hypothesis Testing

Dashboard / App

Correlative Monitoring Approach - Data Collection

- Data-driven methodology
- Associate AMI energy consumption with data from sensors
- Examples: motion, temperature, circuit information, characteristics of home appliances
- Off-the-shelf sensors for home automation



HAN Monitoring - Modular Approach

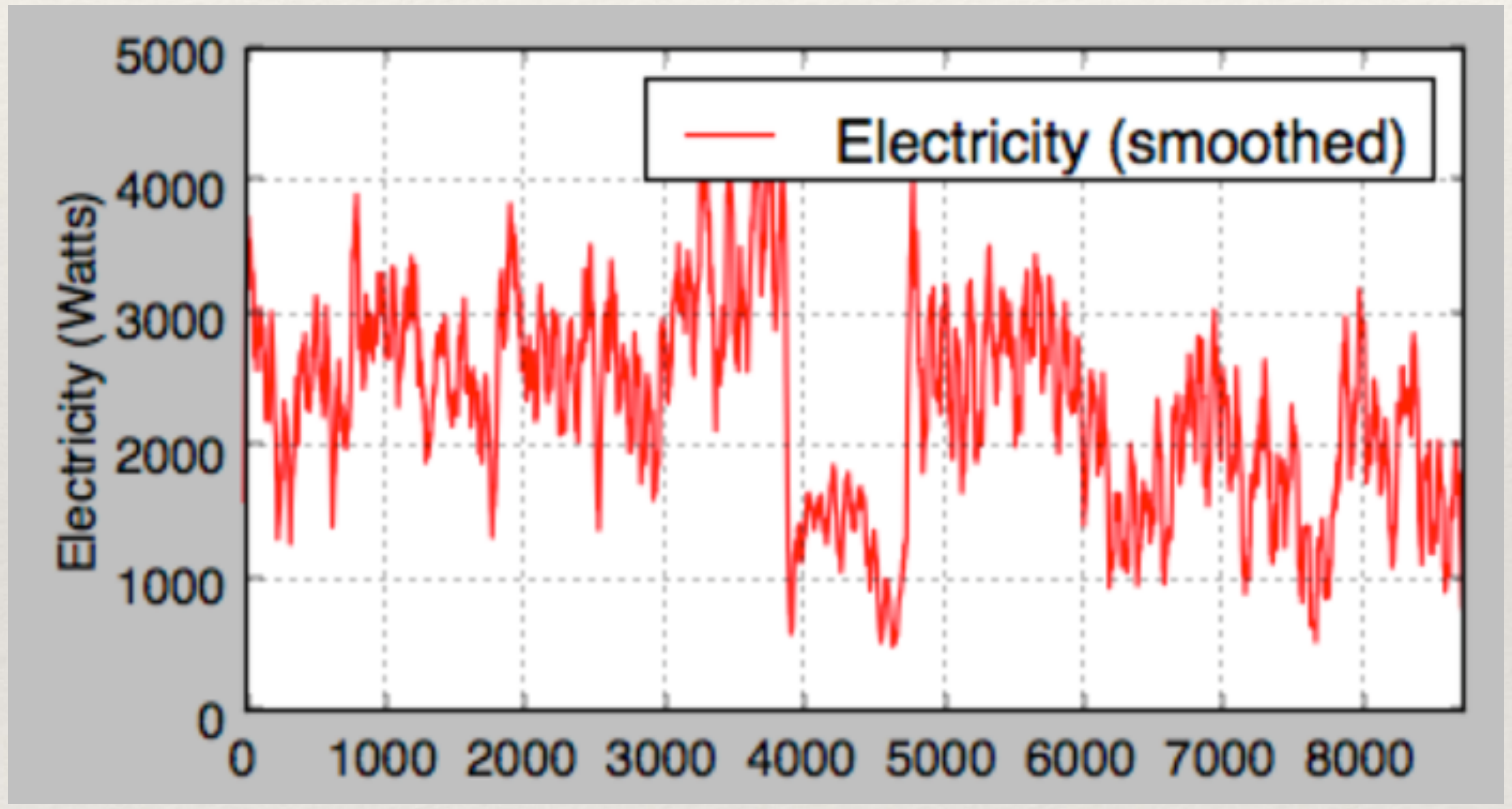
Data Collection

Forecasting

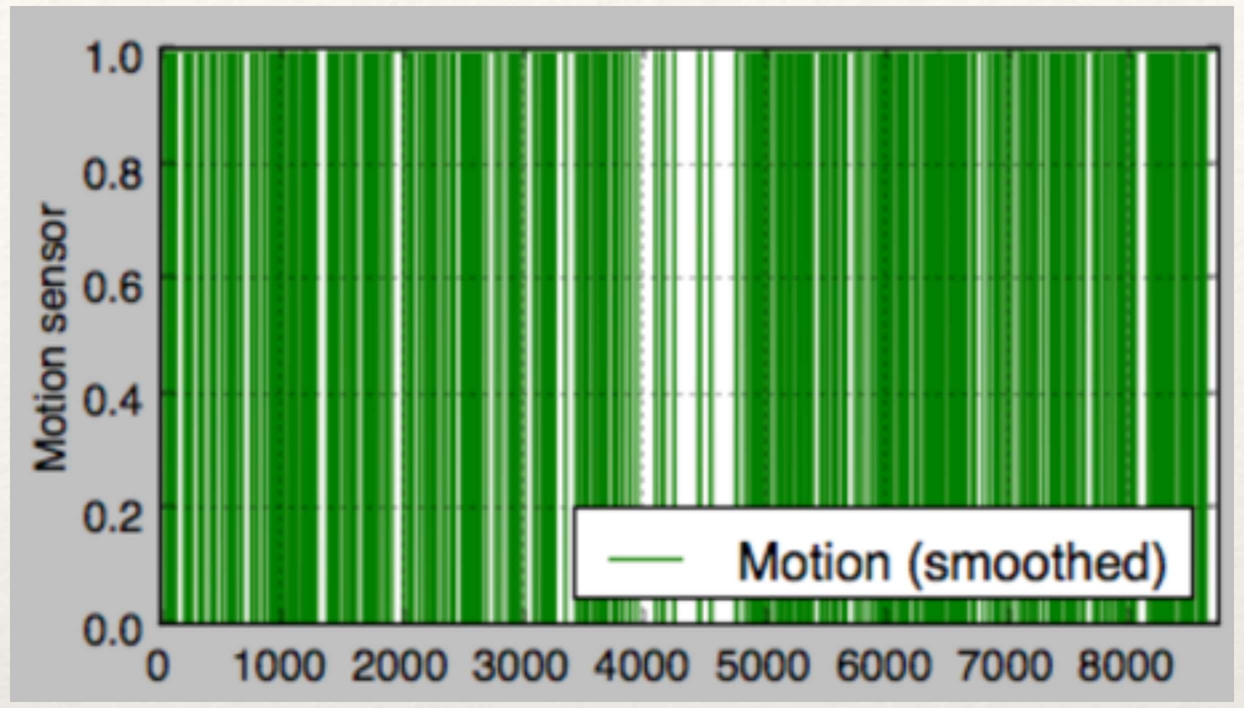
Hypothesis Testing

Dashboard / App

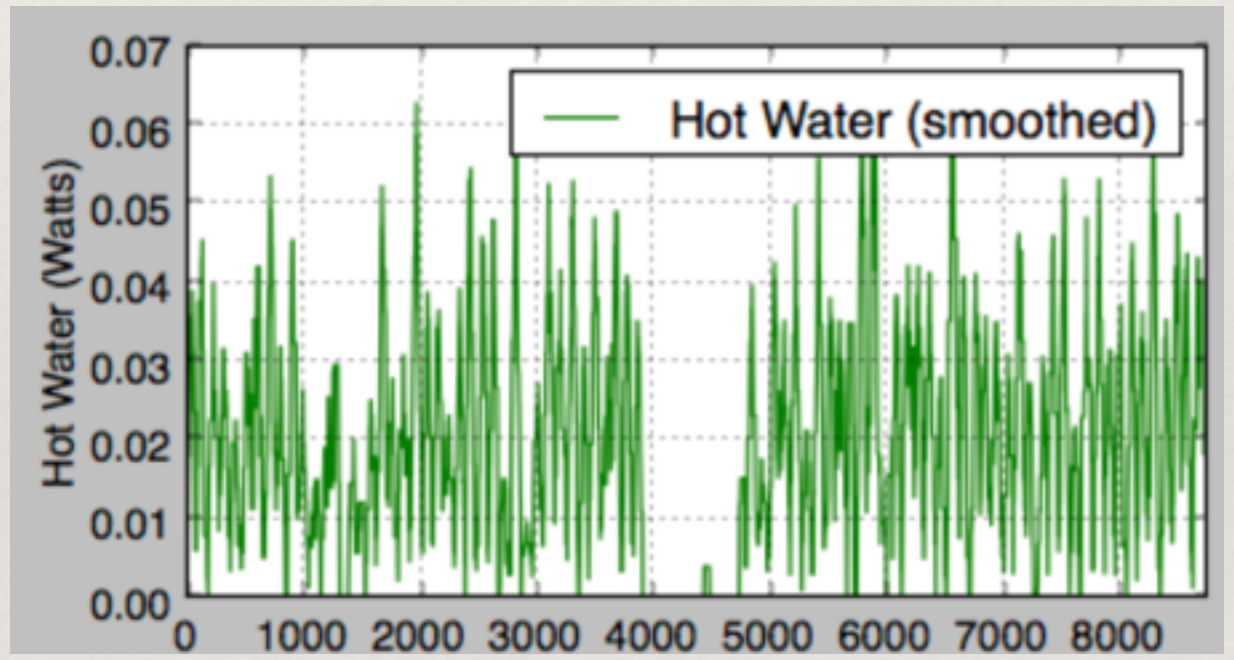
Correlative Monitoring - Forecasting Module



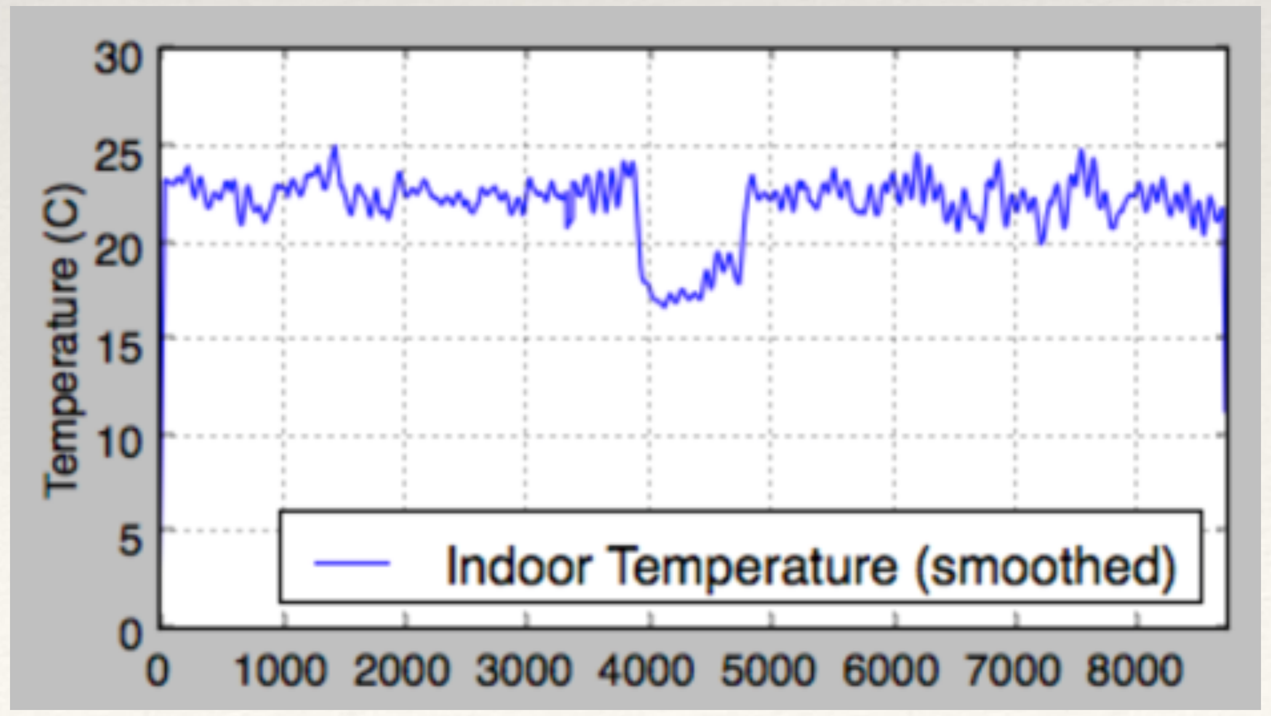
target variable (t)
Total Electricity



Motion



Hot Water



Indoor
Temperature

Main steps of the detection algorithm

- **Build predictive model** that forecasts energy consumption in the next time period(s) based on past consumption (over a trailing window) and other sensor readings
  - Various choices: linear, kernel, GP regression, support vector regression

$$\text{Upon observing } (t_n, \mathbf{x}_n), \text{ compute } y(\mathbf{x}_n, \mathbf{w})$$

Main steps of the detection algorithm

- Build predictive model that forecasts energy consumption in the next time period(s) based on past consumption (over a trailing window) and other sensor readings
  - Various choices: linear, kernel, GP regression, support vector regression
- Obtain forecasting error: (prediction - actual reading)

$$\text{Compute error } e_n = t_n - y(\mathbf{x}_n, \mathbf{w})$$

HAN Monitoring - Modular Approach

Data Collection

Forecasting

Hypothesis Testing

Dashboard / App

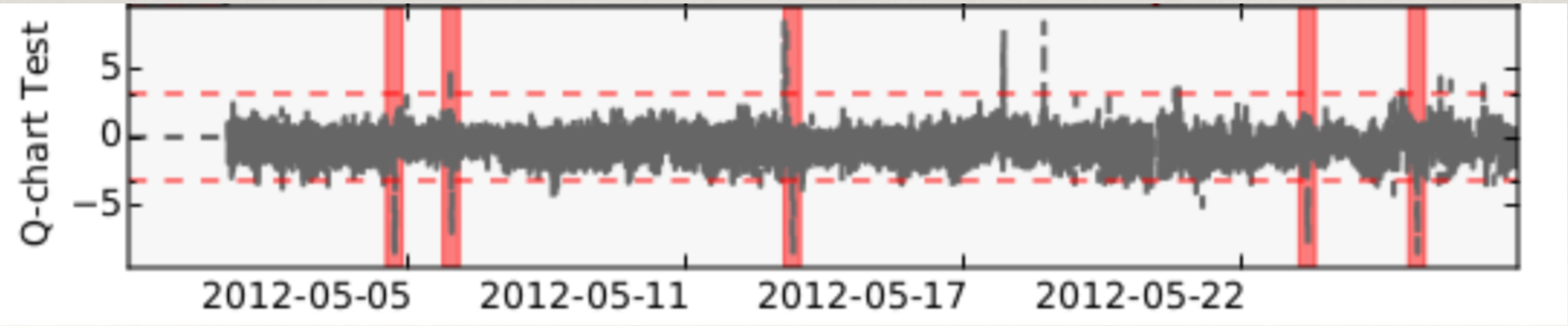
Main steps of the detection algorithm

- **Build predictive model** that forecasts energy consumption in the next time period(s) based on past consumption (over a trailing window) and other sensor readings
  - Various choices: linear, kernel, GP regression, support vector regression
- **Obtain forecasting error:** (prediction - actual reading)
- Use **predictive distribution** to calculate the **tail (p-value)** of the error

$$p(e_n|\mathbf{x}_n, \mathbf{t}, \alpha, \beta) = \mathcal{N}(e_n|0, \sigma_N^2(\mathbf{x}_n))$$

Main steps of the detection algorithm

- Build predictive model that forecasts energy consumption in the next time period(s) based on past consumption (over a trailing window) and other sensor readings
  - Various choices: linear, kernel, GP regression, support vector regression
- Obtain forecasting error: (prediction - actual reading)
- Use Exponentially Weighted Moving Average charts to identify out-of-control



Main steps of the detection algorithm

- Build predictive model that forecasts energy consumption in the next time period(s) based on past consumption (over a trailing window) and other sensor readings
  - Various choices: linear, kernel, GP regression, support vector regression
- Obtain forecasting error: (prediction - actual reading)
- Use Exponentially Weighted Moving Average charts to identify out-of-control:
  - forecasting errors
  - Or their p-values, if reference distribution exists for normal operations (see Lambert and Liu, JASA, 2006)
  - Extensive literature on selecting adaptive thresholds and memory parameters for the EWMA chart (book by Qiu, 2014)
  - Employ the two-in-a-row rule to robustly the out-of-control calls (Lucas and Santucci, Technometrics, 1993)

Agenda

- Introduction - Problem Motivation
- Correlative Monitoring Approach
- Performance Evaluation

Evaluation: Smart* dataset

- Measurement period: May - July 2012
- Granularity of 1-minute
- Training window size: 24 hours
- Forecasting period: 30, 60 minutes
- Inject random data attacks

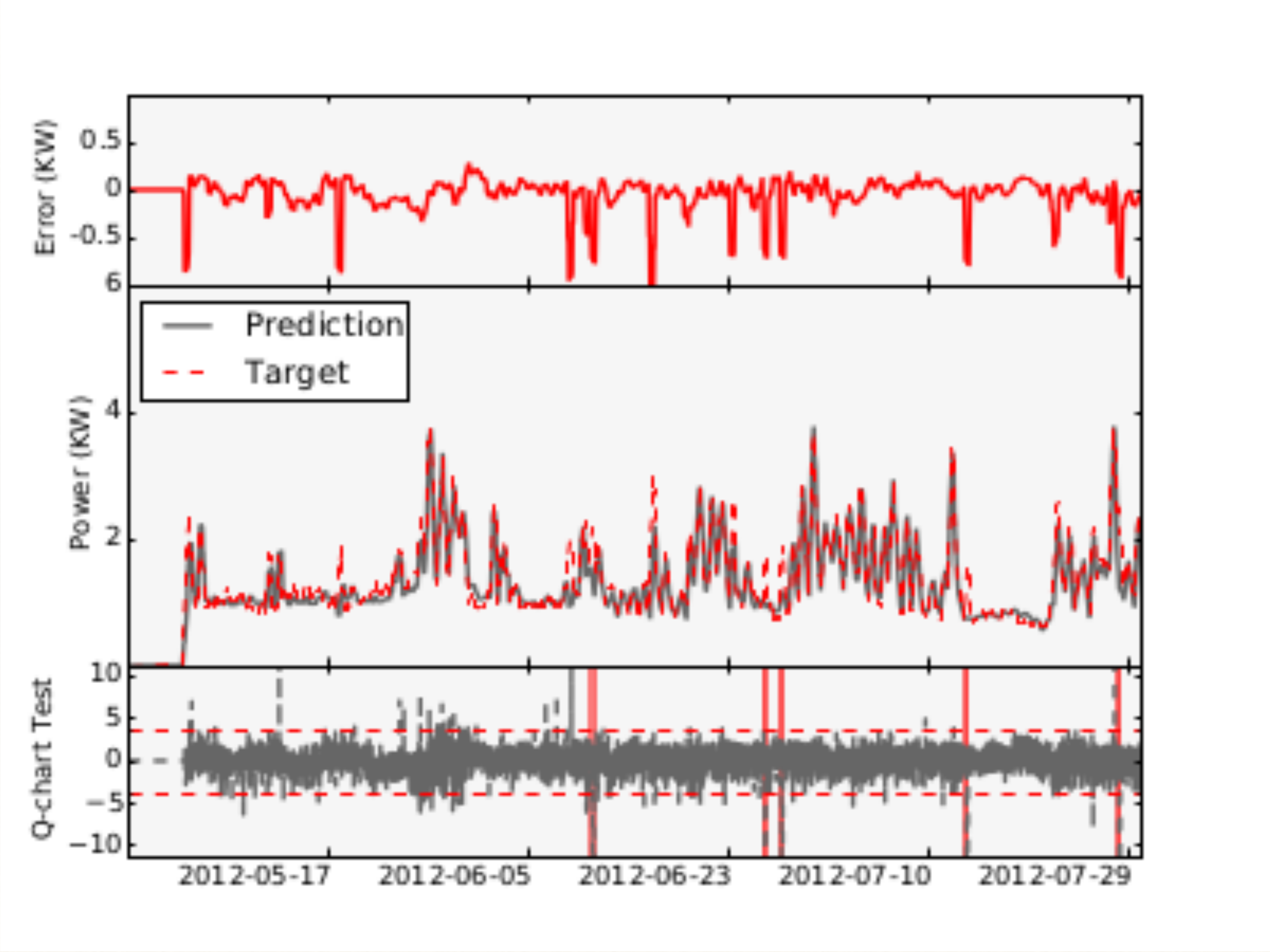


TABLE I: Evaluation of detection performance on the Smart* dataset. Values in parenthesis signify standard deviations.

Shift (KW)	**Weight λ**	**Delay (in mins)**	**Precision**	**Recall**	**F1-score**
-1	1	$9.7_{(7.2)}$	$.29_{(.45)}$	$.07_{(.11)}$	.11
-1	.53	$8.1_{(4.6)}$	$.76_{(.41)}$	$.29_{(.21)}$	.42
-1	.84	$10.4_{(5.5)}$	$.48_{(.50)}$	$.12_{(.14)}$	.19
1	1	$8.0_{(4.5)}$	$.75_{(.43)}$	$.26_{(.19)}$	.38
1	.53	$3.4_{(1.7)}$	$.95_{(.17)}$	$.50_{(.22)}$	.66
1	.84	$6.6_{(3.4)}$	$.86_{(.31)}$	$.31_{(.19)}$	.46
3	1	$1.1_{(.5)}$	$.98_{(.05)}$	$1.00_{(.03)}$	.99
3	.53	$1.0_{(.0)}$	$.98_{(.06)}$	$1.00_{(.03)}$	.99
3	.84	$1.0_{(.2)}$	$.98_{(.06)}$	$1.00_{(.03)}$	.99
6	1	$1.2_{(.9)}$	$.96_{(.11)}$	$.99_{(.05)}$	.97
6	.53	$1.0_{(.0)}$	$.97_{(.08)}$	$1.00_{(.05)}$	.98
6	.84	$1.0_{(0.0)}$	$.96_{(.09)}$	$.99_{(.05)}$	.98

TABLE I: Evaluation of detection performance on the Smart* dataset. Values in parenthesis signify standard deviations.

Shift (KW)	Weight λ	Delay (in mins)	Precision	Recall	F1-score
-1	1	$9.7_{(7.2)}$	$.29_{(.45)}$	$.07_{(.11)}$	.11
-1	.53	$8.1_{(4.6)}$	$.76_{(.41)}$	$.29_{(.21)}$	.42
-1	.84	$10.4_{(5.5)}$	$.48_{(.50)}$	$.12_{(.14)}$	.19
1	1	$8.0_{(4.5)}$	$.75_{(.43)}$	$.26_{(.19)}$	.38
1	.53	$3.4_{(1.7)}$	$.95_{(.17)}$	$.50_{(.22)}$	.66
1	.84	$6.6_{(3.4)}$	$.86_{(.31)}$	$.31_{(.19)}$	.46
3	1	$1.1_{(.5)}$	$.98_{(.05)}$	$1.00_{(.03)}$	.99
3	.53	$1.0_{(.0)}$	$.98_{(.06)}$	$1.00_{(.03)}$	.99
3	.84	$1.0_{(.2)}$	$.98_{(.06)}$	$1.00_{(.03)}$	.99
6	1	$1.2_{(.9)}$	$.96_{(.11)}$	$.99_{(.05)}$	.97
6	.53	$1.0_{(.0)}$	$.97_{(.08)}$	$1.00_{(.05)}$	.98
6	.84	$1.0_{(0.0)}$	$.96_{(.09)}$	$.99_{(.05)}$	.98

Summary & Future Directions

- Correlative monitoring in HANs, bottom-up approach
- Proof-of-concept implementation with Raspberry Pi's and Z-Wave sensors - partnership with NextEnergy!
- Incorporate energy harvesting sensing!
- Acknowledgements: Joe Adams, Yeabsera Kebede, Max Morgan, Davis Vorva (UM/Merit), Atman Fozdar (EMU), Wayne Snyder (NextEnergy)
- Supported by NSF SATC CNS-1422078

“If we have data, let’s look at data.
If all we have are opinions, let’s go with mine.”

–Jim Barksdale, former Netscape CEO

Supplementary Material

Bayesian Linear Regression

- Avoid the need for cross-validation and model selection
- Provides a predictive distribution
- Linear: good choice when data from HAN circuits are available. In addition, with appropriate basis functions non-linearity may not be an issue

Framework 1 Measurement-based False Data Detection

Require: For each forecasting period: new training set $\mathbf{X}$ and $\mathbf{t}$.
Require: Control chart parameters λ and L.
Require: Robust threshold θ_r and period ν.

1: [Start] Fit the model and begin data monitoring.
2: [Forecast] Upon observing $(t_n, \mathbf{x}_n)$, compute $y(\mathbf{x}_n, \mathbf{w})$.
3: [Update] Compute error $e_n = t_n - y(\mathbf{x}_n, \mathbf{w})$.
4: [Control Chart] Compute $S_n = f(\lambda, L, e_n)$.
5: [Robust EWMA] Apply two-in-a-row rule on S_n (see section III-B).
6: [Robust Filter] Update $A = \{k : \; |S_k| > L\sigma_\lambda, k = n - \nu, \ldots, n\}$.
7: [Decision] Raise alarm if $|A| > \theta_r$, else system is in-control.

DTE Energy Bridge - App access to meter

Precision & Recall