UCSD
UNIVERSITY OF CALIFORNIA, SAN DIEGO
Jacobs School *of* Engineering

Automatic Event Detection and Ring Down Analysis & Mitigation of Grid Oscillations

Raymond de Callafon and Charles H. Wells

University of California, San Diego & OSIsoft

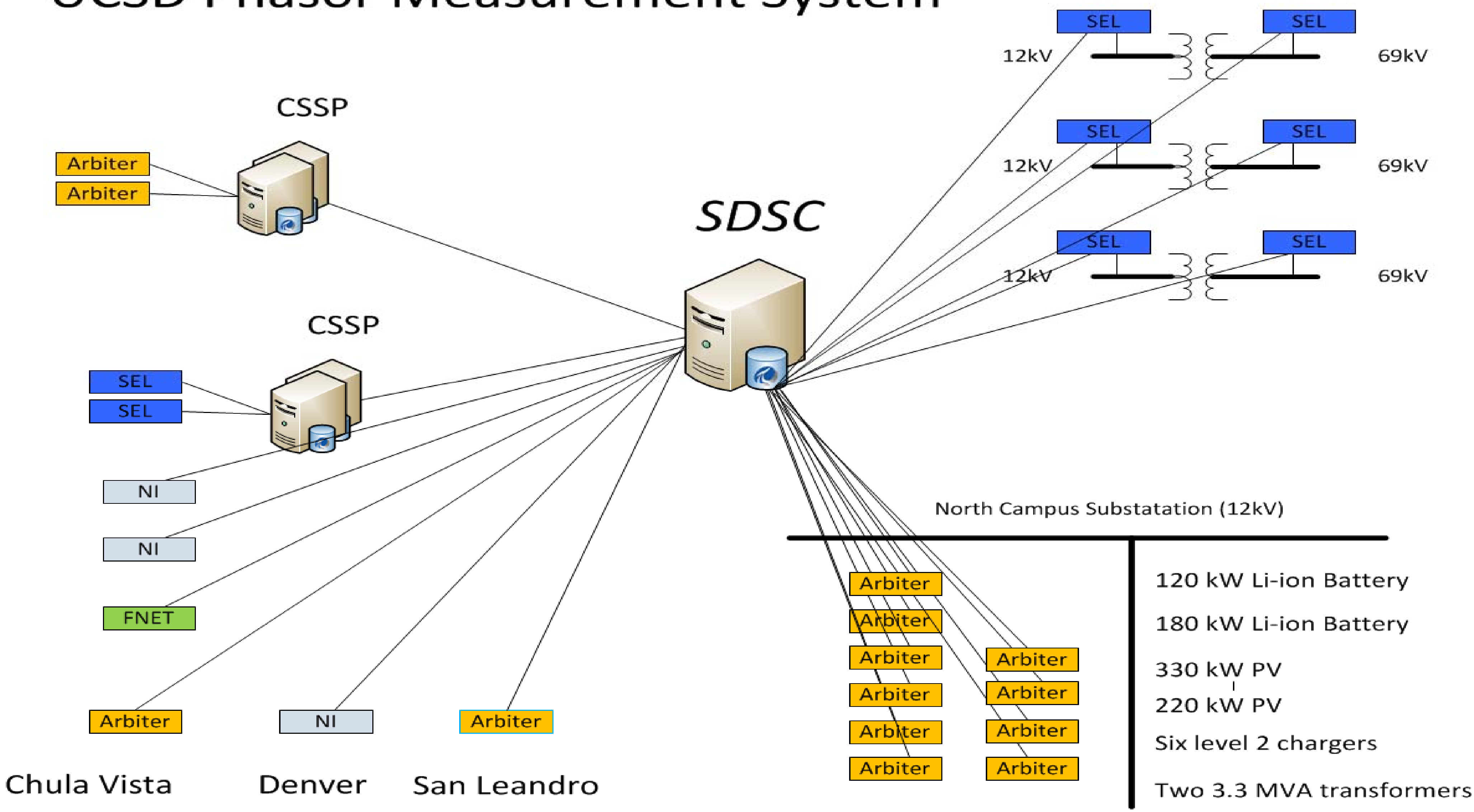
9th CMU Conference Pittsburgh, PA Feb 2, 2014

email: callafon@ucsd.edu, cwells@osisoft.com

UCSD Jacobs | Mechanical and Aerospace Engineering

UCSD Phasor Measurement System

UCSD Phasor Measurement System



Additional PMUs

- microPMUs from PSL (ARPA-E funded)

Building Name
EBU3A Biobuilding
Atkinson Hall
Pacific Hall
Natural Sciences
CMME&CMMW
SDSC
Sverdrup
P703
Jacobs
CUP A
CUP B
North Campus Housing
Rady School
RIMAC
Hospital CC Embergency A
Hospital CC Emergency B
SOM Pharm
SOM BSB
SIO Hubbs Hall
CPS WC -9

Goals

Identification/classification:

- Identify major modes of grid oscillation
- Identify their frequencies, damping and modal participation
- Develop dynamic model that can be used for future control/mitigation of disturbances

Analysis/control:

- Determine how well models correlate with modes
- Use models for automatic control for mitigation

Our main contributions

- Detection of Events via Filtered Rate of Change (FRoC) signal
  - Auto Regressive Moving Average (ARMA) filter of ambient data.
  - Definition of Filtered Rate-of-Change (Froc) signal for Event Detection
- Ring Down Analysis of Events via Realization Algorithm
  - Discrete-Time State Space Modeling of disturbance data.
  - Modeling of grid real power dynamics
- Mitigation of Events via Real-time Control
  - Use dynamic model from Realization Algorithm
  - Design low-order real-time (automatic) control with minimal control effort

- Illustration in this talk:
  - Part 1: Automatic event detection applied to May 30 WECC event
  - Part 2: UCSD Microgrid: analysis and control of Oct. 9 event

PART 1

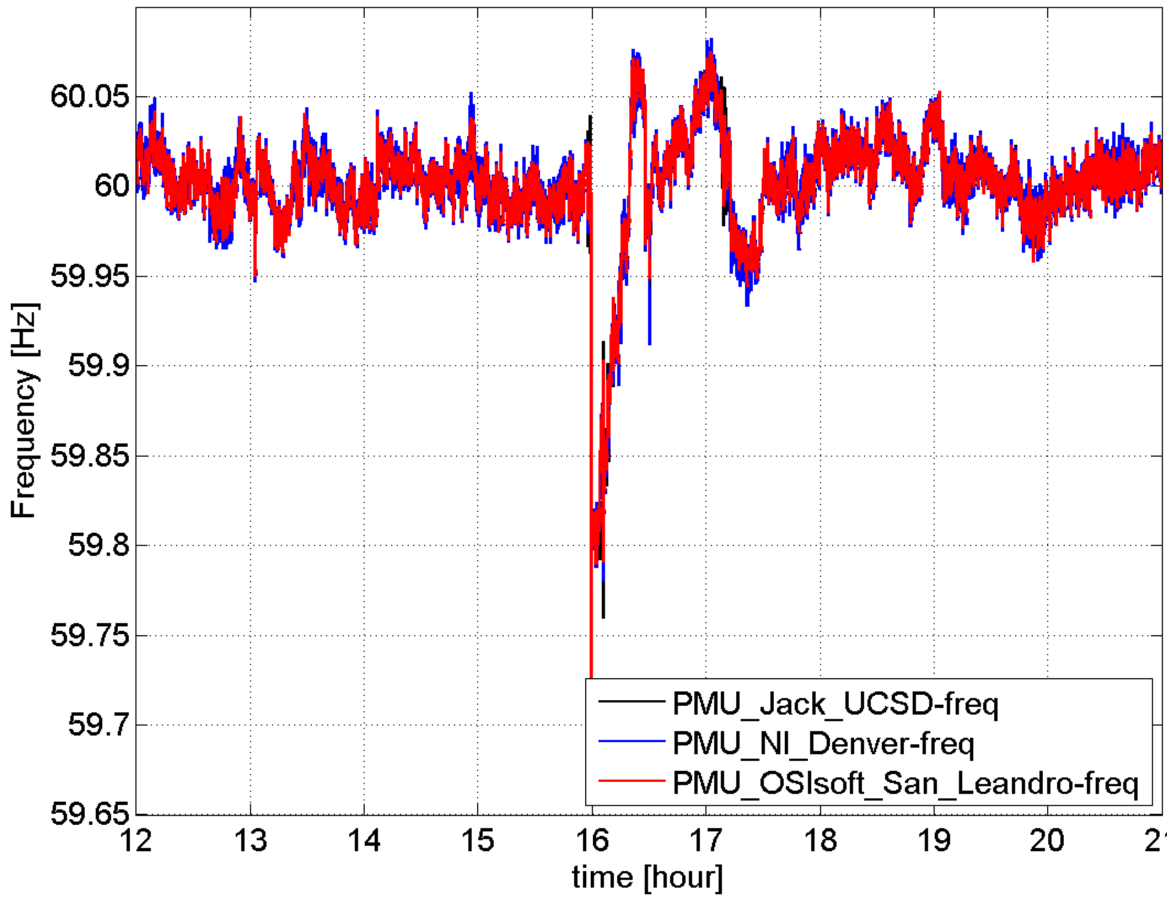
Automatic Event Detection

Application to May 30 WECC disturbance

Illustration on May 30th WECC data

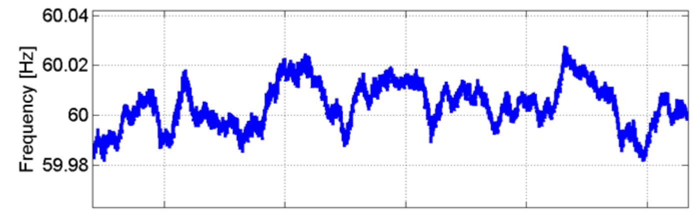
Grid events/oscillations (example: May 30 WECC event)

- PMU generated frequency signal
- How do we detect individual events?
- How can we quantify these events?
- What do these events tell us about our (micro)grid?



May 30 data: 972000 data points (30Hz sampling noon-9pm)

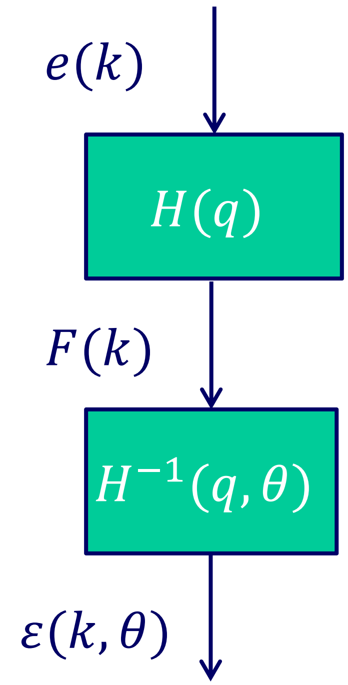
FRoC Signal - ARMA filter



- In ambient situation we may assume:
  - Fluctuations in frequency signal $F(k)$ assumed due to "random noise" on grid
  - $F(k)$ can be modeled as a "filtered white noise"

$$F(k) = H(q)e(k)$$

where $H(q)$ is an unknown filter and $e(k)$ is a white noise.



- Possible approximation for filter $H(q)$: ARMA filter

$$H(q,\theta) = \frac{b_0 + b_1 q^{-1} + \cdots + b_n q^{-n}}{1 + a_1 q^{-1} + \cdots + a_n q^{-n}}$$

  - Filter $H(q)$ is stable and stably invertible
  - We can compute

$$\varepsilon(k,\theta) = H(q,\theta)^{-1} F(k)$$

  - Parameters $\theta = [b_1 \quad \cdots \quad b_n \quad a_1 \quad \cdots \quad a_n]$ can be estimated via Least Squares (Prediction Error) to minimize variance of error $\varepsilon(k,\theta)$.

FRoC Signal – RoC filter

- With optimal value of θ we have "smallest possible"
$$\varepsilon(k,\theta) = H(q,\theta)^{-1}F(k)$$
during ambient behavior.
- To create FRoC: add additional filtering on $\varepsilon(k,\theta)$ to monitor Rate of Change in $F(k)$
- Typical Filter:
$$FRoC(k) = R(q)L(q)H(q,\theta)^{-1}F(k)$$
$$R(q) = \frac{q-1}{q-0.9}, \qquad L(q) = \frac{0.1367q + 0.1367}{q - 0.7265}$$

END RESULT: a real-time recursive formula to compute $FRoC(k)$:

$$FRoC(k) = b_0F(k) + b_1F(k-1) + \cdots + b_nF(k-n)$$
$$-a_1FRoC(k-1) - \cdots - a_nFRoC(k-n)$$

FRoC Signal – RoC filter

- In our case based on real-time PMU data we created the discrete-time filter equation to obtain FRoC(k):

$$
\begin{aligned}
FRoC(k) = {} & 0.12786 \cdot F(k) - 0.25412 \cdot F(k-1) - 0.00094 \cdot F(k-2) \\
& +0.25411 \cdot F(k-3) - 0.12694 \cdot F(k-4) \\
& +3.48506 \cdot FRoC(k-1) - 4.54036 \cdot FRoC(k-2) \\
& +2.61982 \cdot FRoC(k-3) - 0.56464 \cdot FRoC(k-4)
\end{aligned}
$$

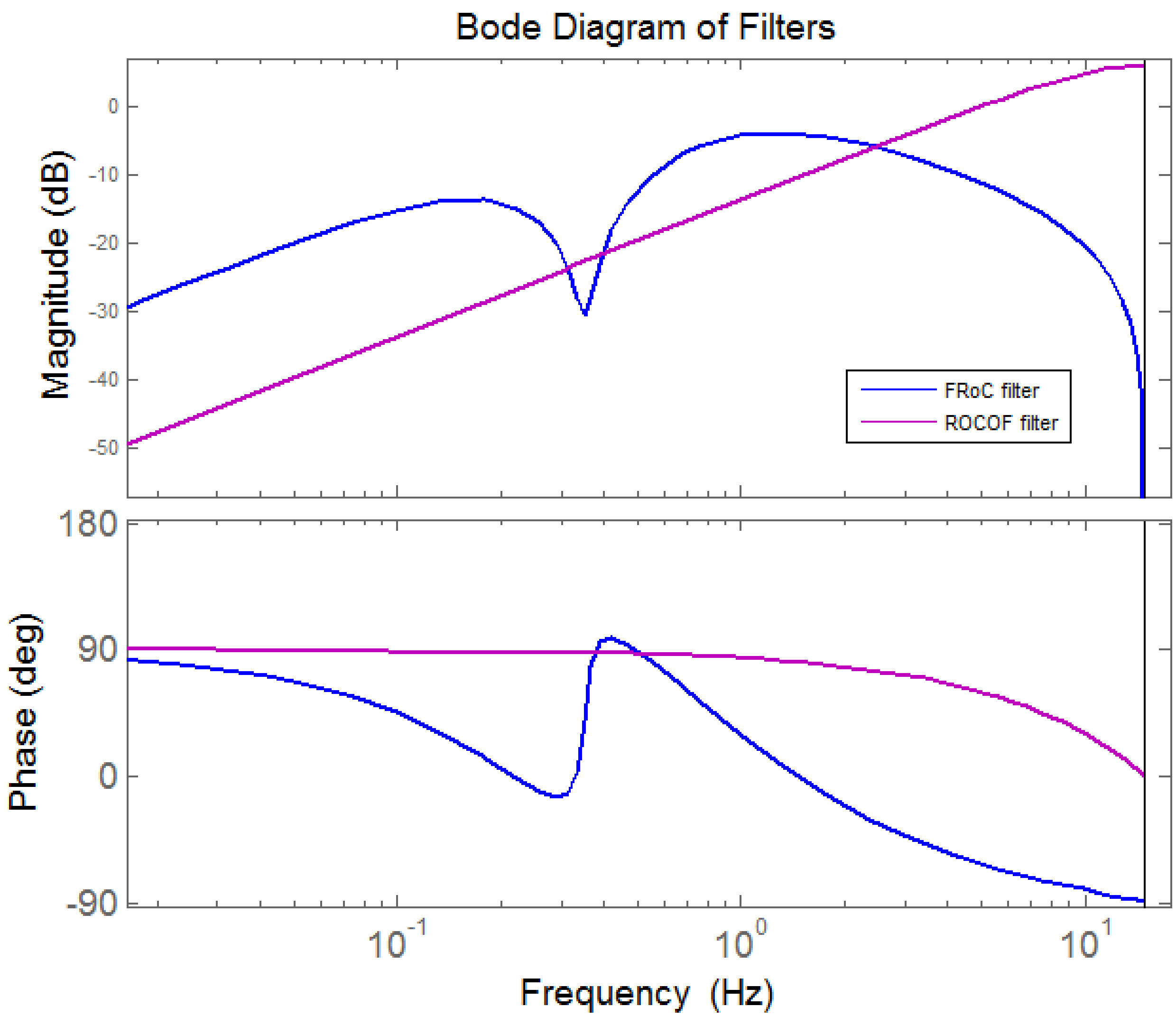
- Compared with ROCOF(k):

$$ROCOF(k) = 30(F(k) - F(k-1)$$

(dirty discrete-time derivative)

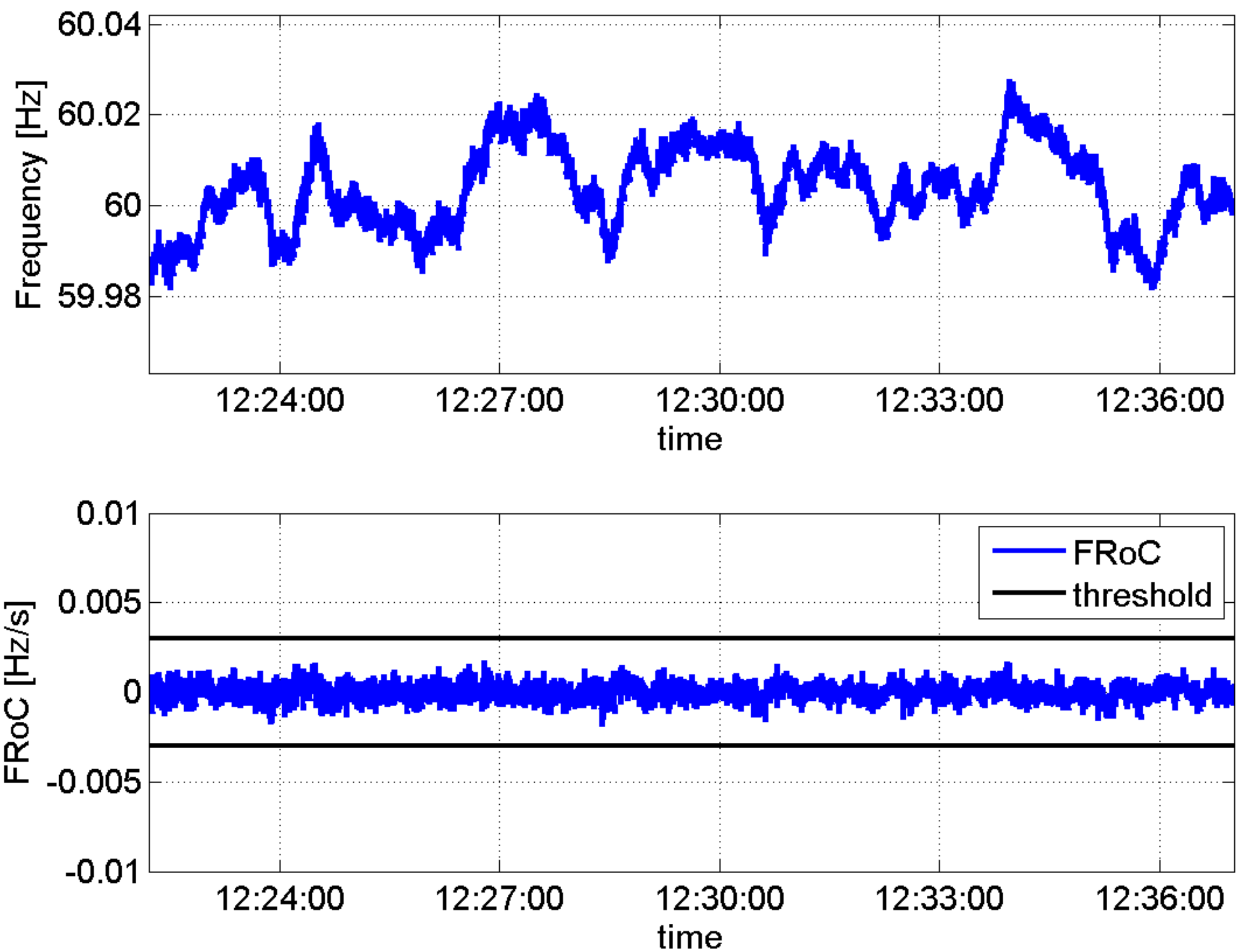
FRoC Signal – RoC filter

- Bode plot of filters used to create FRoC(k) and ROCOF(k) illustrates the benefits:
  - Filter looks like a 'differentiator'
  - Additional filtering of harmonic disturbances ambient data at 0.35Hz
  - Additional low pass filter to reduce noise



FRoC Signal – what's the big deal?

- Small $FRoC(k)$ during ambient behavior
- Even for "noisy" NI PMU

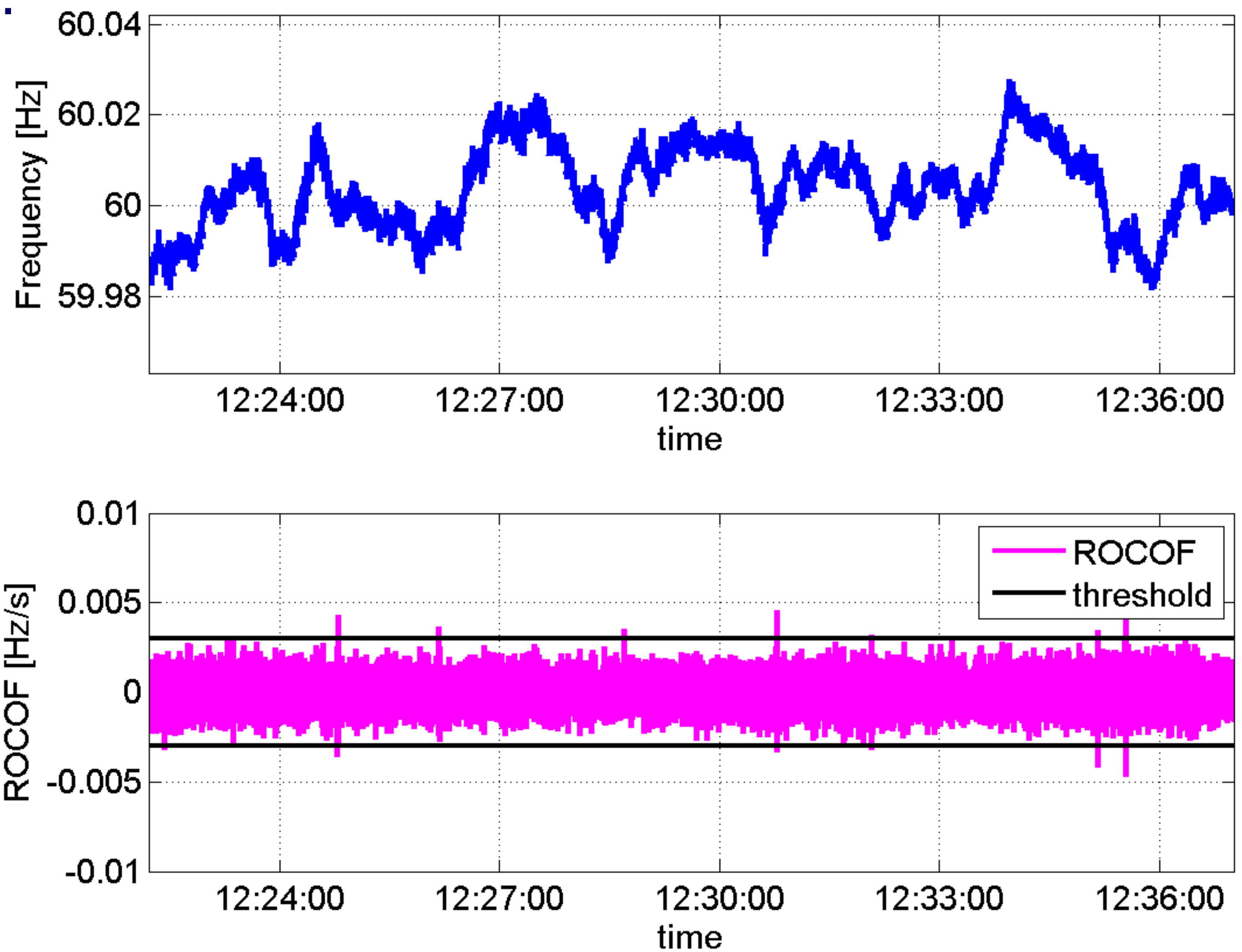


FRoC Signal – what's the big deal?

Compare with ROCOF:

$$ROCOF(k) = \frac{F(k) - F(k-1)}{\Delta t}$$

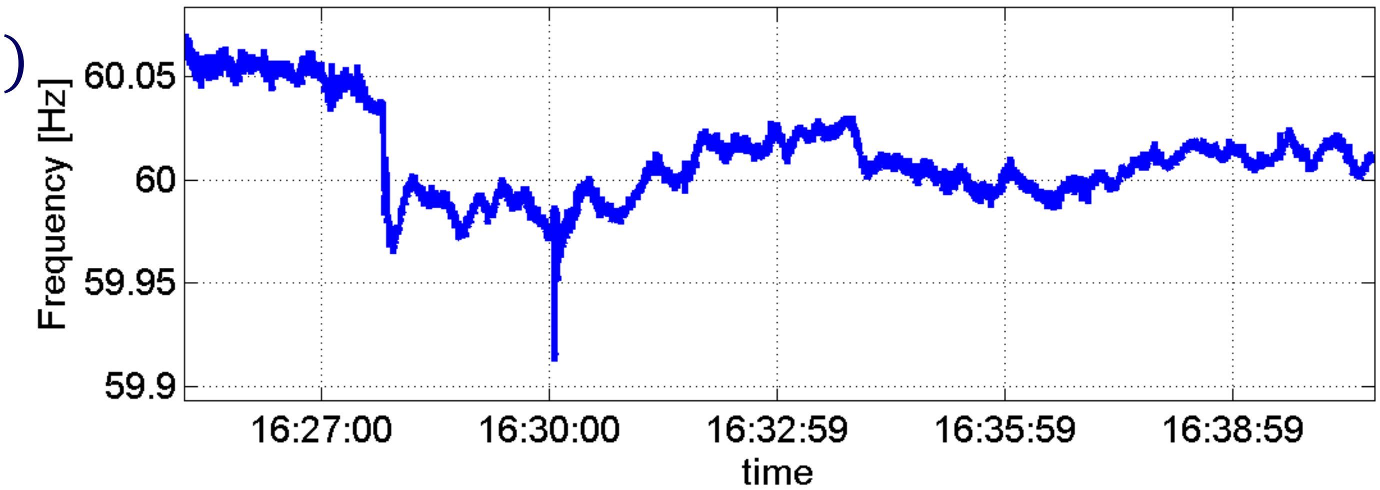
(dirty discrete-time derivative)



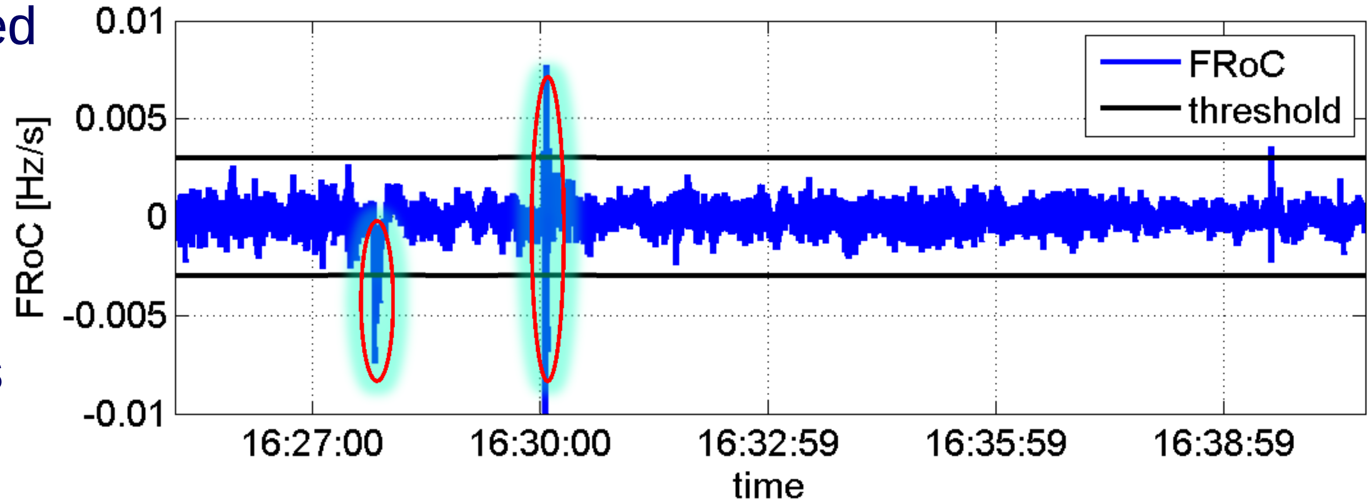
- Much larger than *FRoC(k)*
- Would require larger thresholds

FRoC Signal – what's the big deal?

- Small thresholds with small $FRoC(k)$ during ambient behavior



- Detection of events via:
  - Set threshold based on ambient data
  - $FRoC(k)$ outside threshold for *m* consecutive points

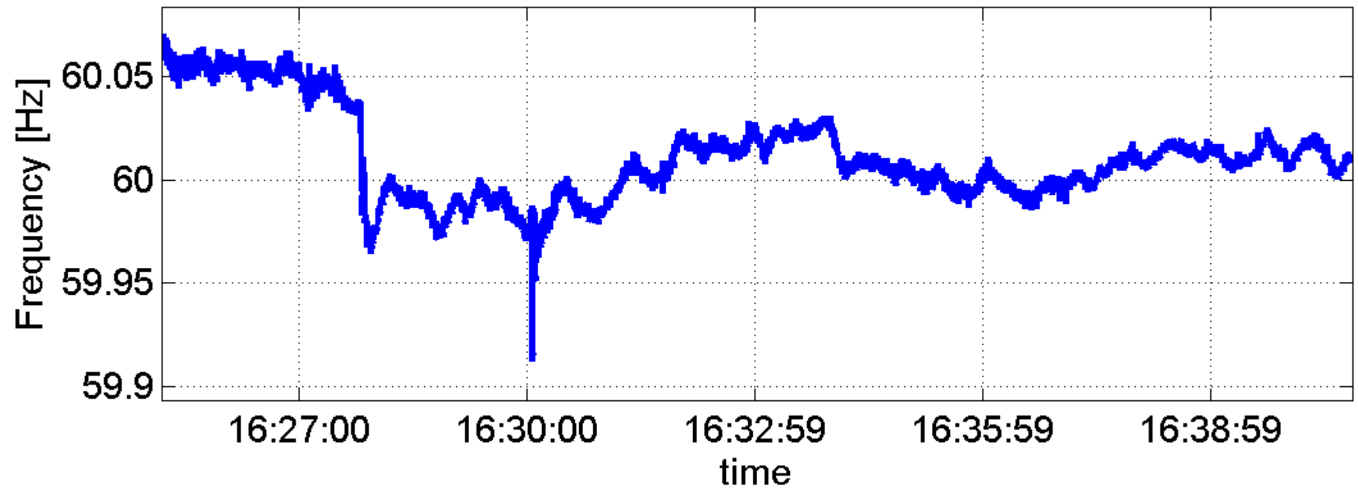


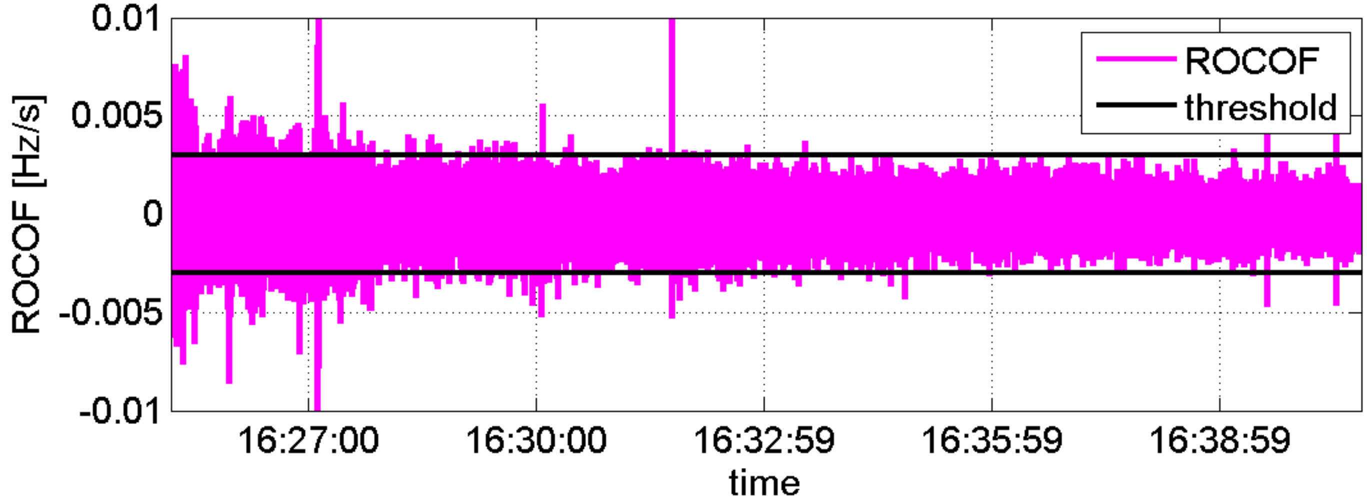
  - Classify event by saving/analyzing *N* data points

FRoC Signal – what's the big deal?

Compare with ROCOF

- Much larger than *FRoC(k)*
- More false alarms

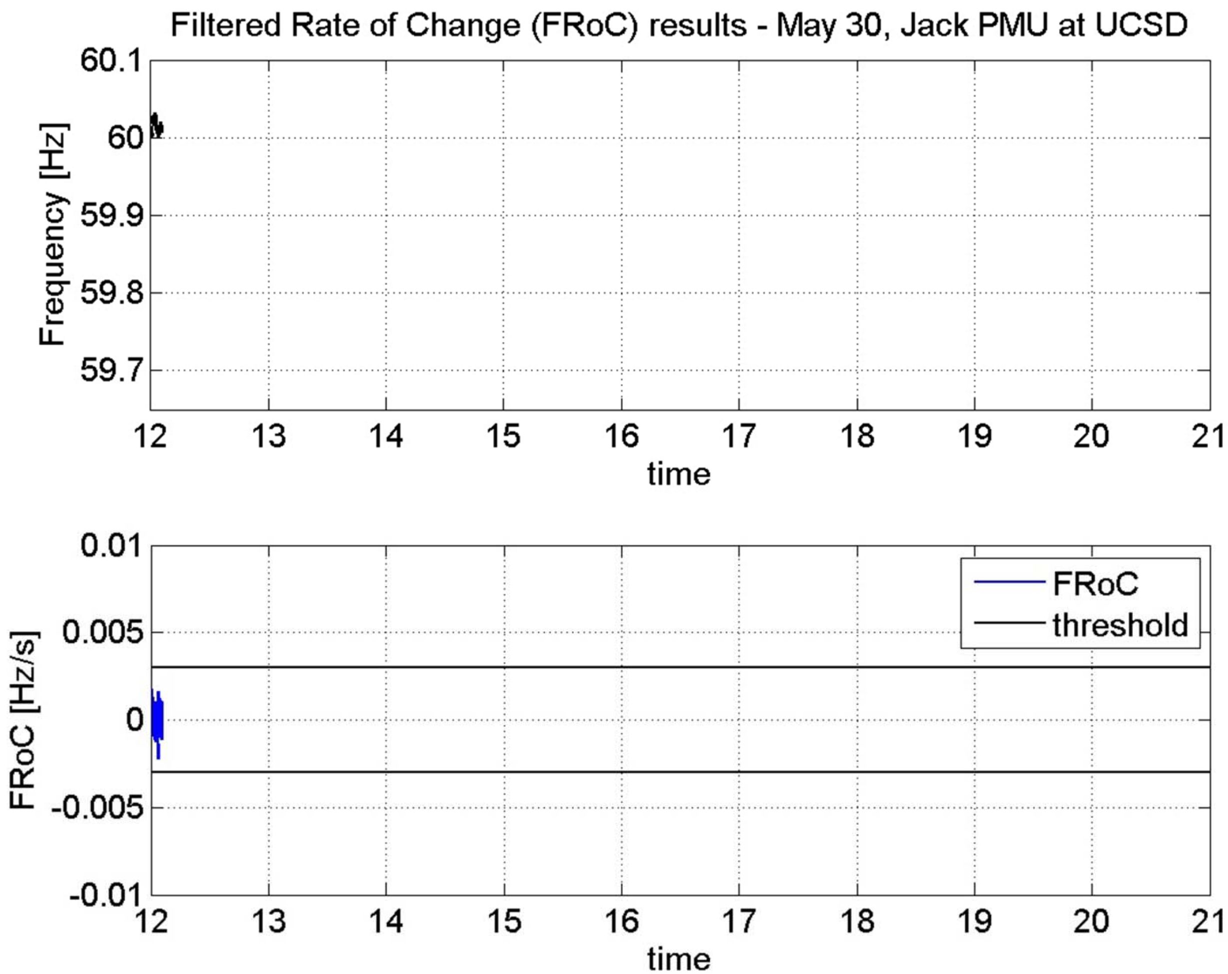




Automatic Detection Results

Automatically:

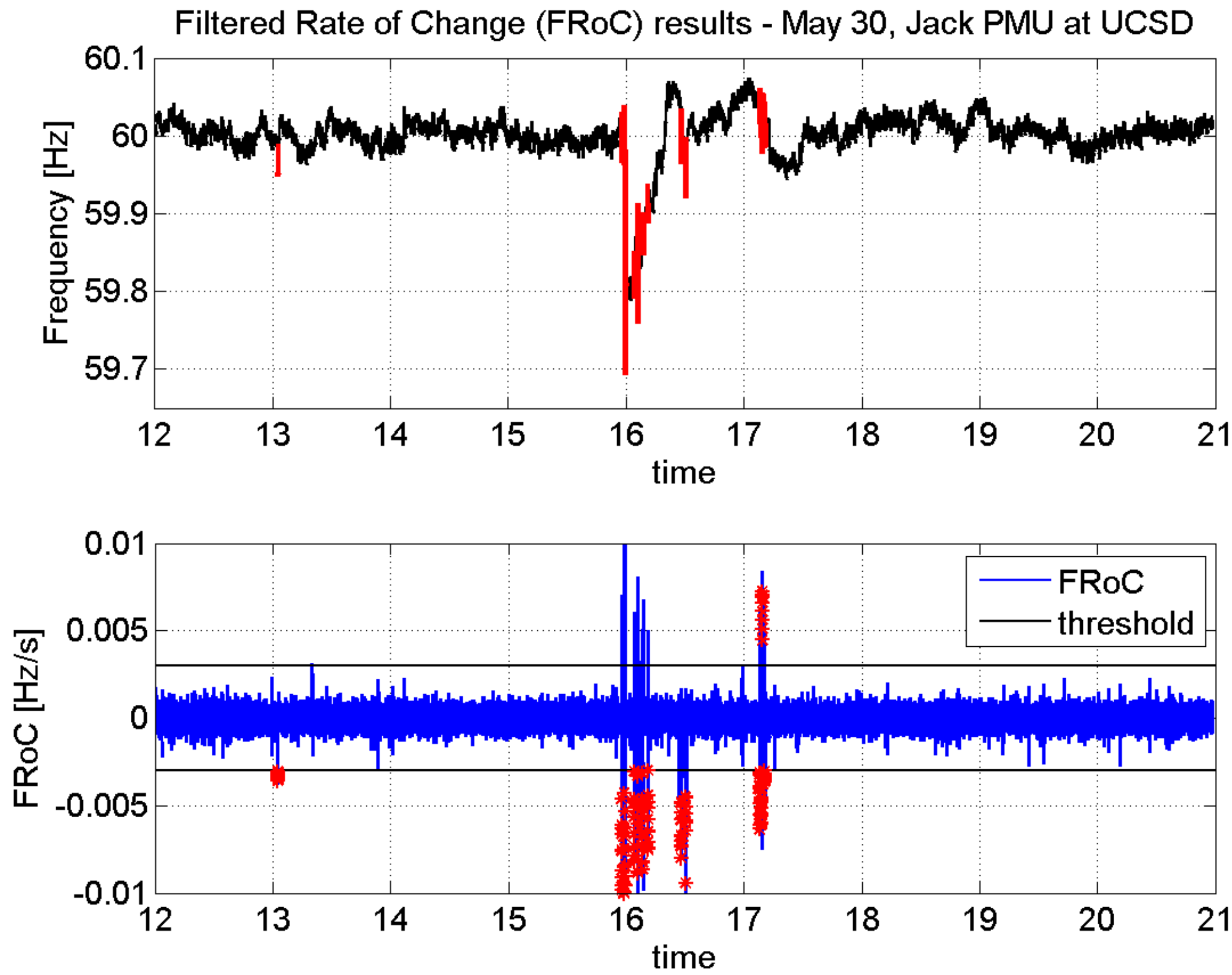
- Detect event. (via threshold on Filtered Rate of Change signal)



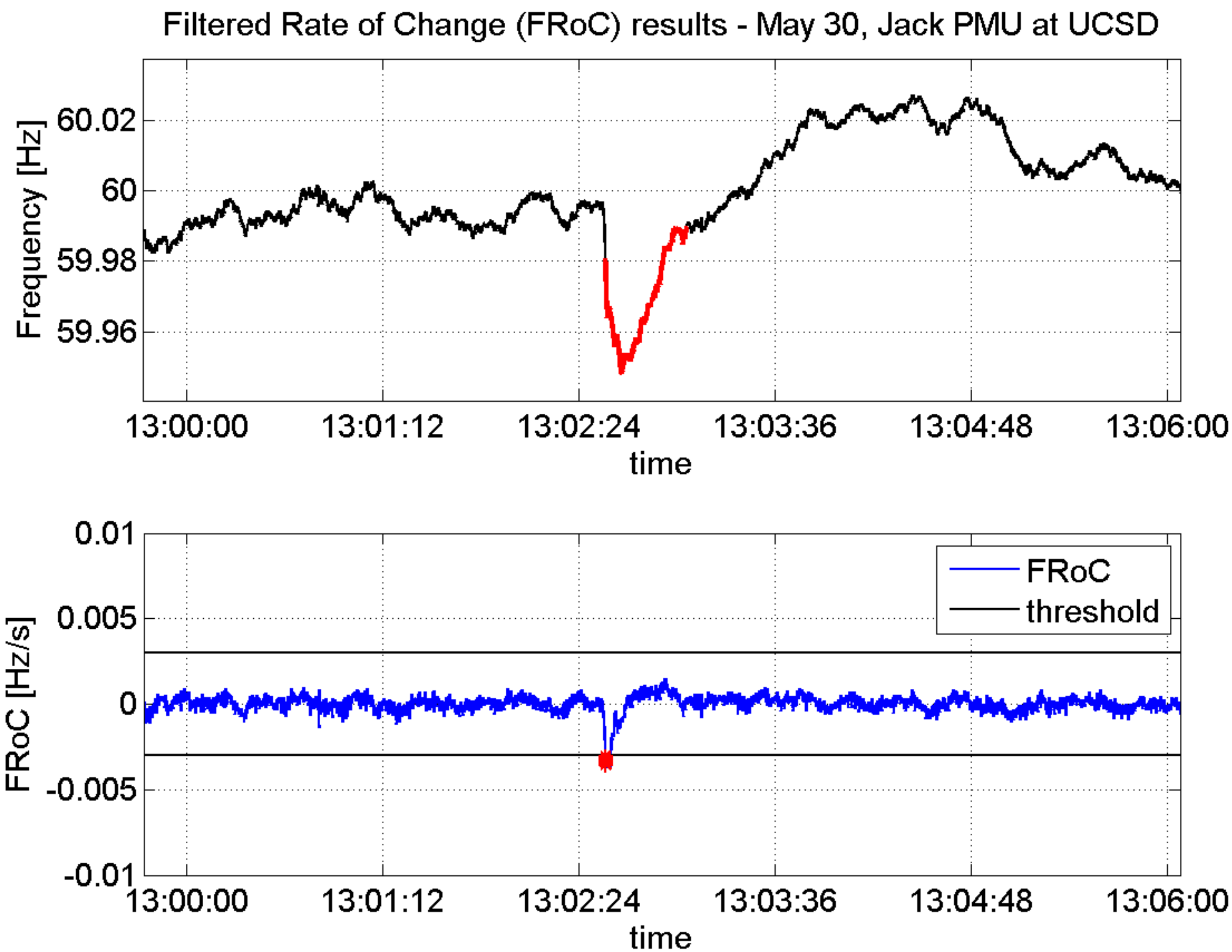
Automatic Detection Results

Automatically:

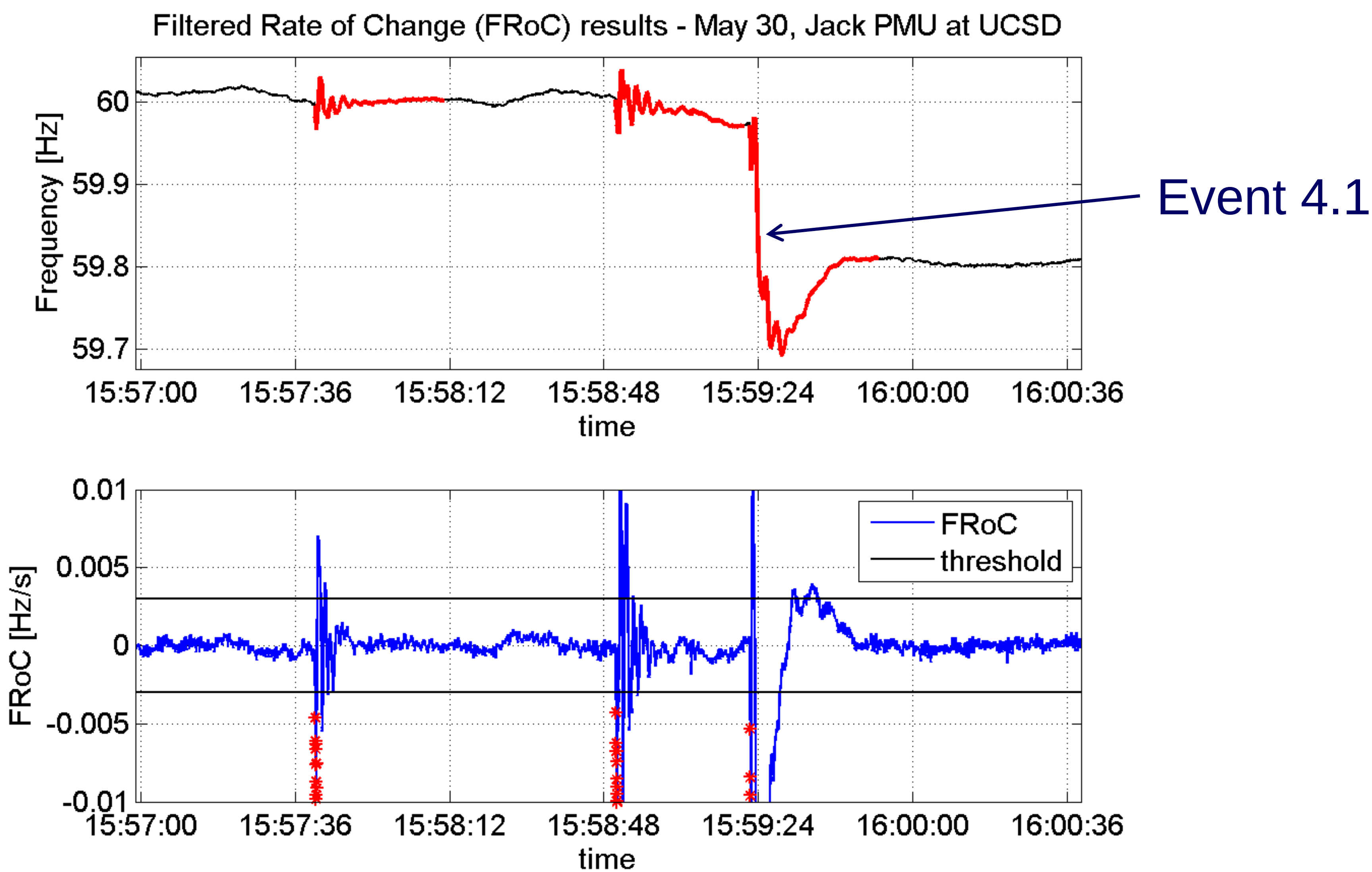
- Detect event. (via threshold on Filtered Rate of Change signal)
- Able to distinguish 14 separate events over 9 hour data



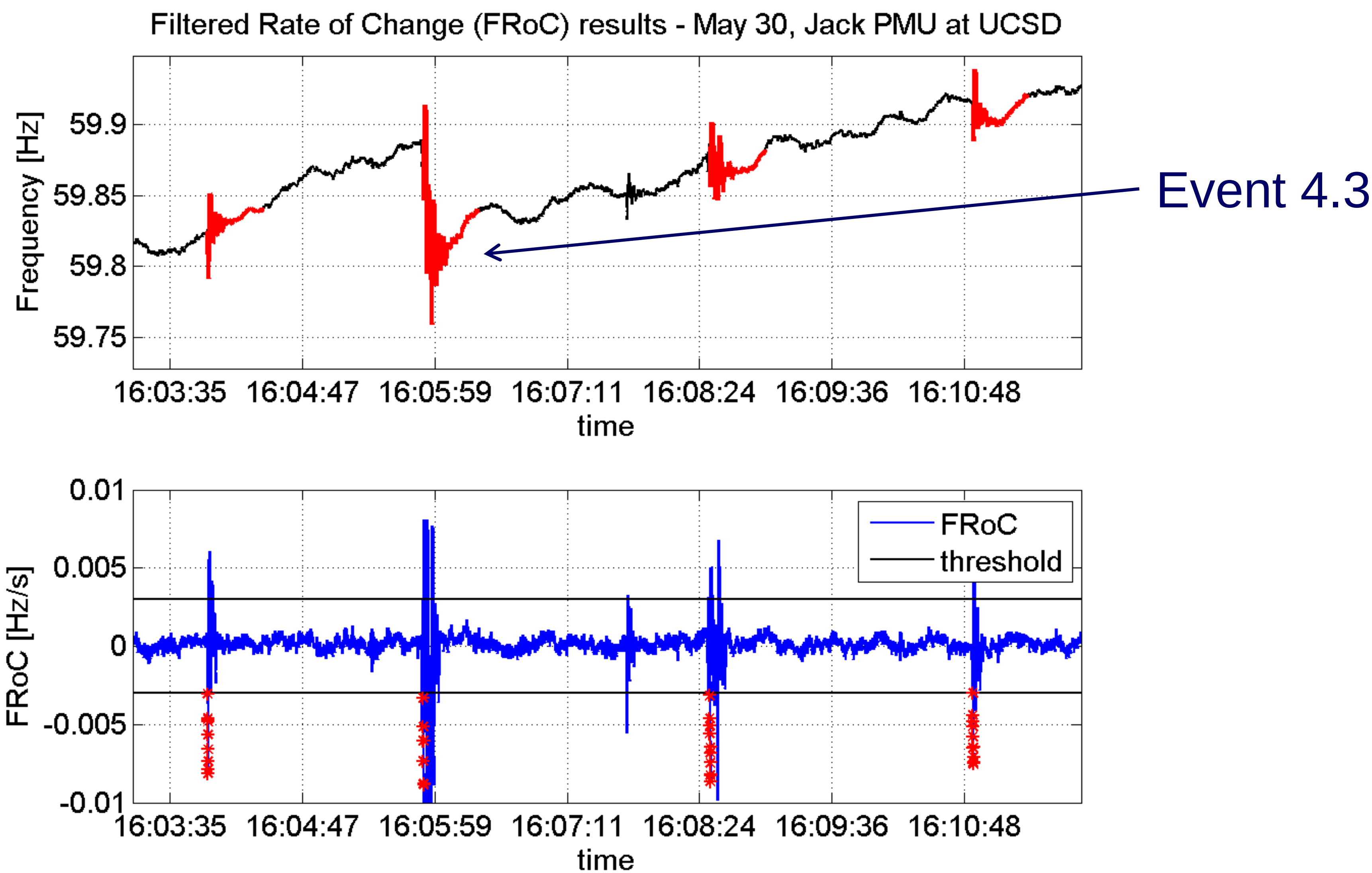
FRoC Signal – application to May 30 data



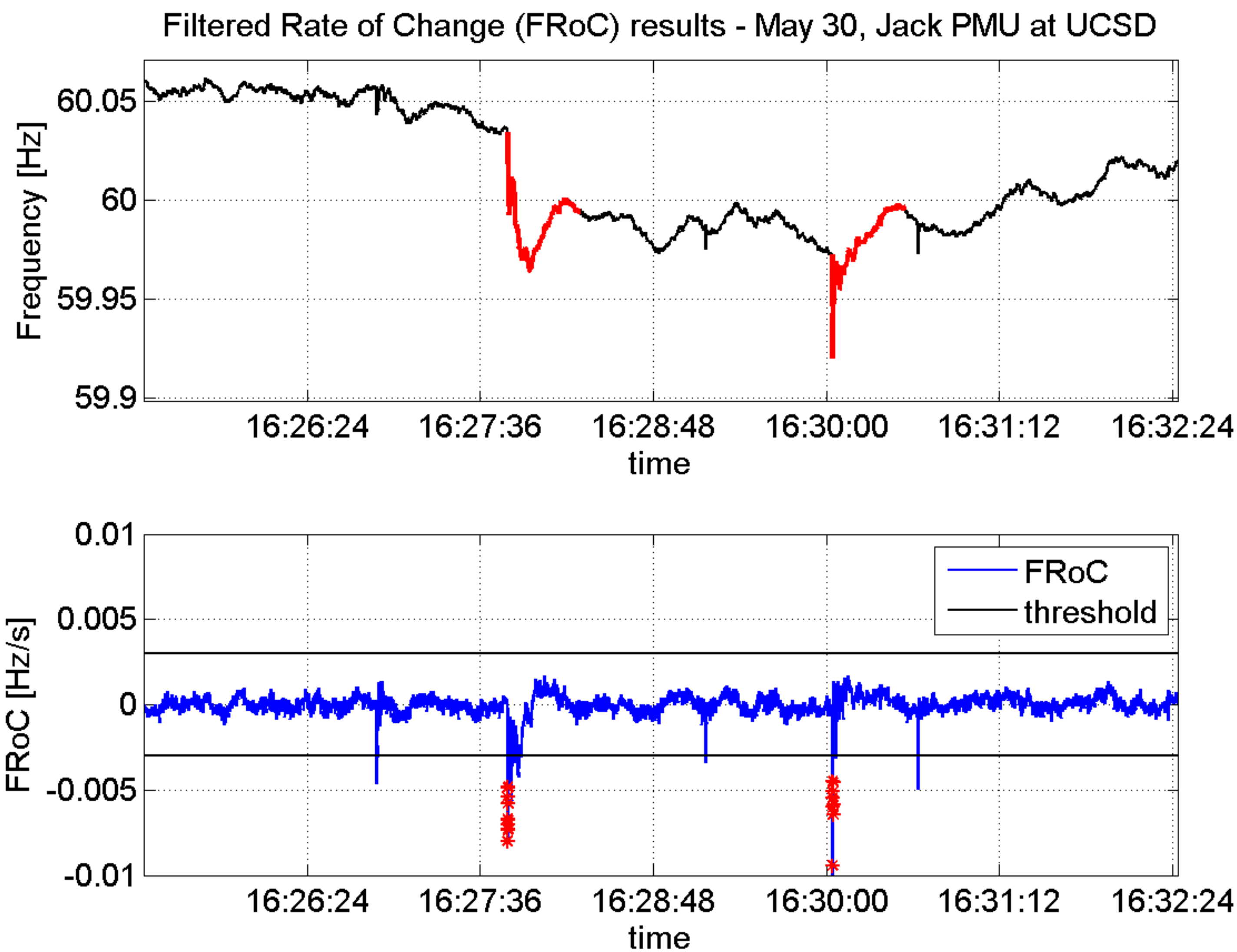
FRoC Signal – application to May 30 data



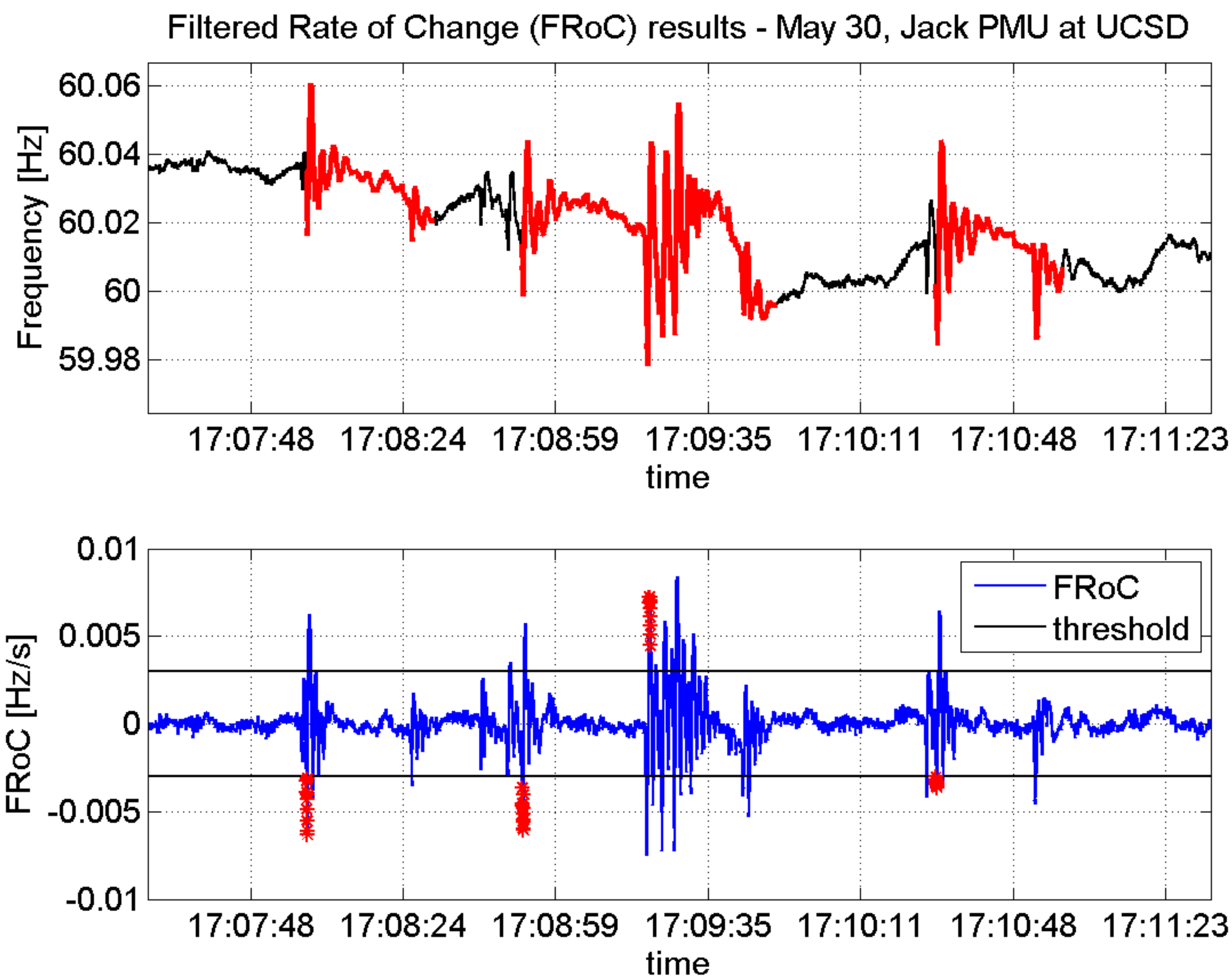
FRoC Signal – application to May 30 data



FRoC Signal – application to May 30 data



FRoC Signal – application to May 30 data



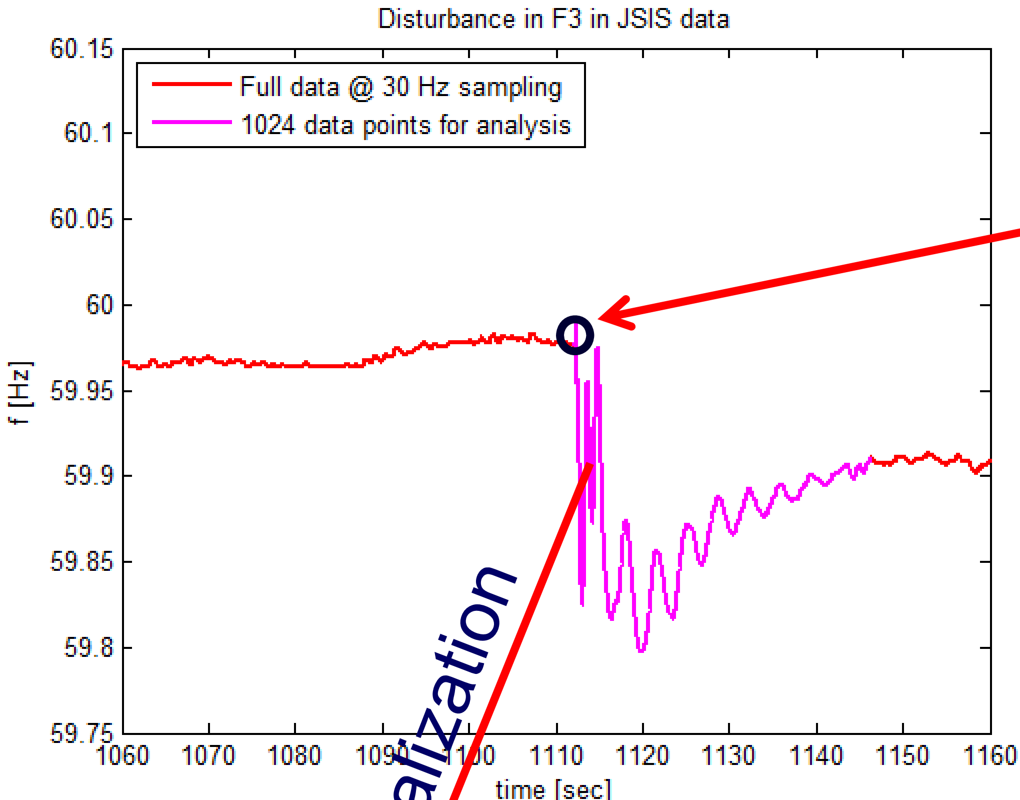
PART2

UCSD Microgrid

Ring Down Analysis of Oct. 9 event

Mitigation of events via real-time control

Analysis of Events - Realization Algorithm



detect beginning of event

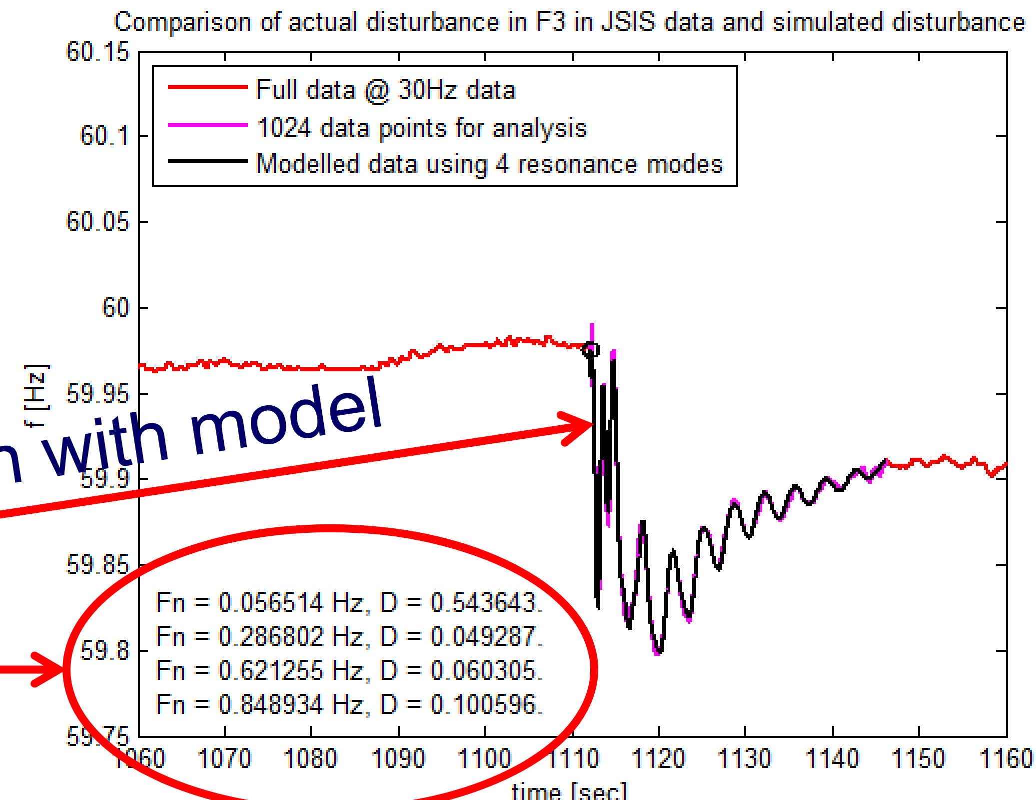
realization

$$x(t+1) = Ax(t) + Bd(t)$$

$$F(t) = Cx(t)$$

simulation with model

analysis



Analysis of Events - Realization Algorithm

Approach:

- Assume observed event in frequency F(t) is due to a deterministic system

$$x(k+1) = Ax(k) + Bd(k)$$
$$F(k) = Cx(k)$$

Discrete-time model

  where (unknown) input d(t) can be \`impulse' or \`step' or \`known shape'
- Store a finite number of data points of F(t) in a special data matrix **H**
- Inspect rank of (null projection on) **H**: determines # modes
- Compute matrices A, B and C via Realization Algorithm.
- Extension of Ho-Kalman, Kung algorithm. Miller, de Callafon (2010)
- Applicable to multiple time-synchronized measurements! (multiple PMUs)

End Result:

- Dynamic model (state space model) can be used for
  - Simulation: simulate the disturbance data
  - Analysis: Compute resonance modes and damping (from eigenvalues of A)

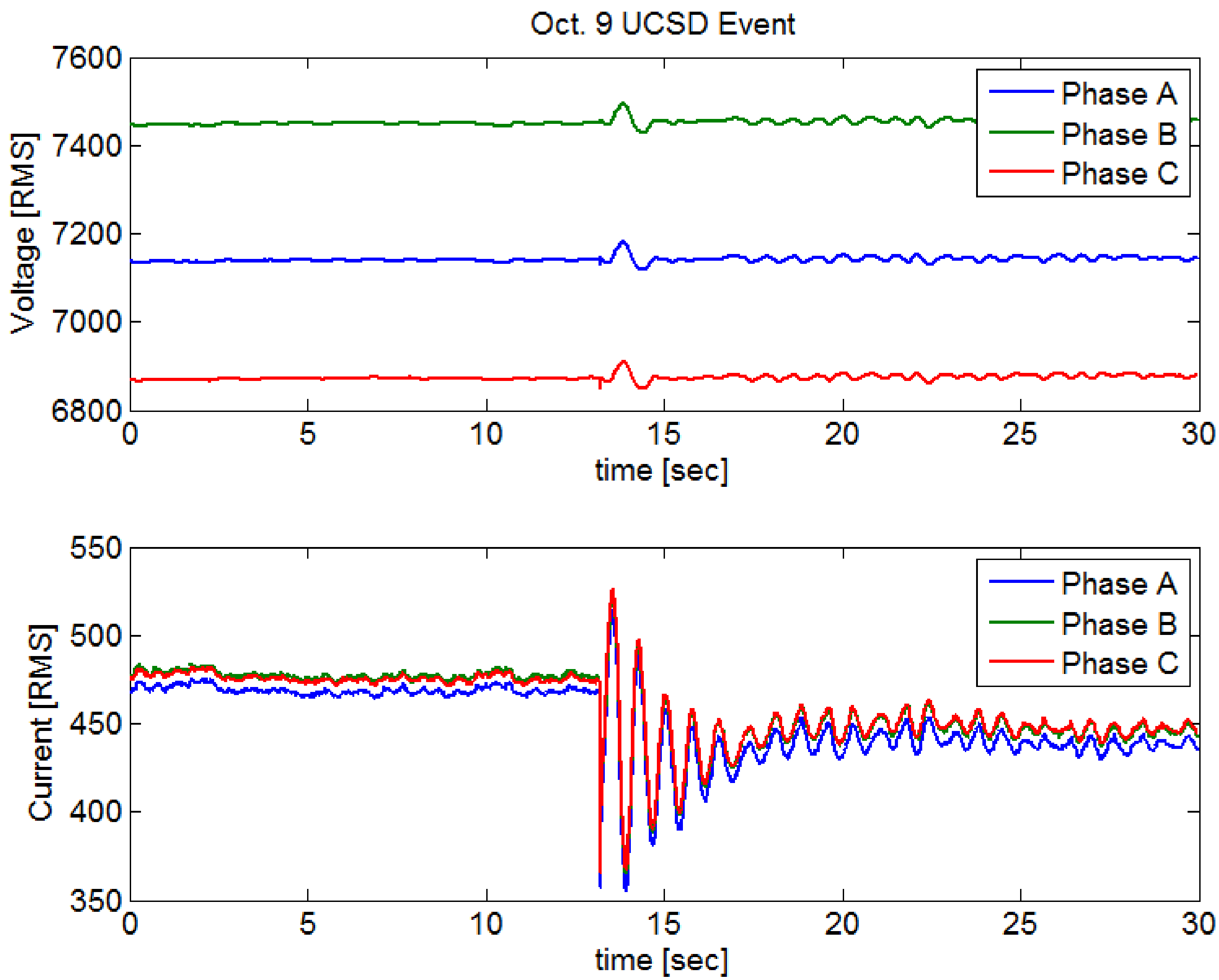
Oct. 9 UCSD microgrid event

Measurements from SEL breaker at 12kV

3 phase line (6.9kV phase to phase)

- RMS Voltage and Current of 3 phases
- Real Power
- Apparent Power

Disturbance on 3 phase network

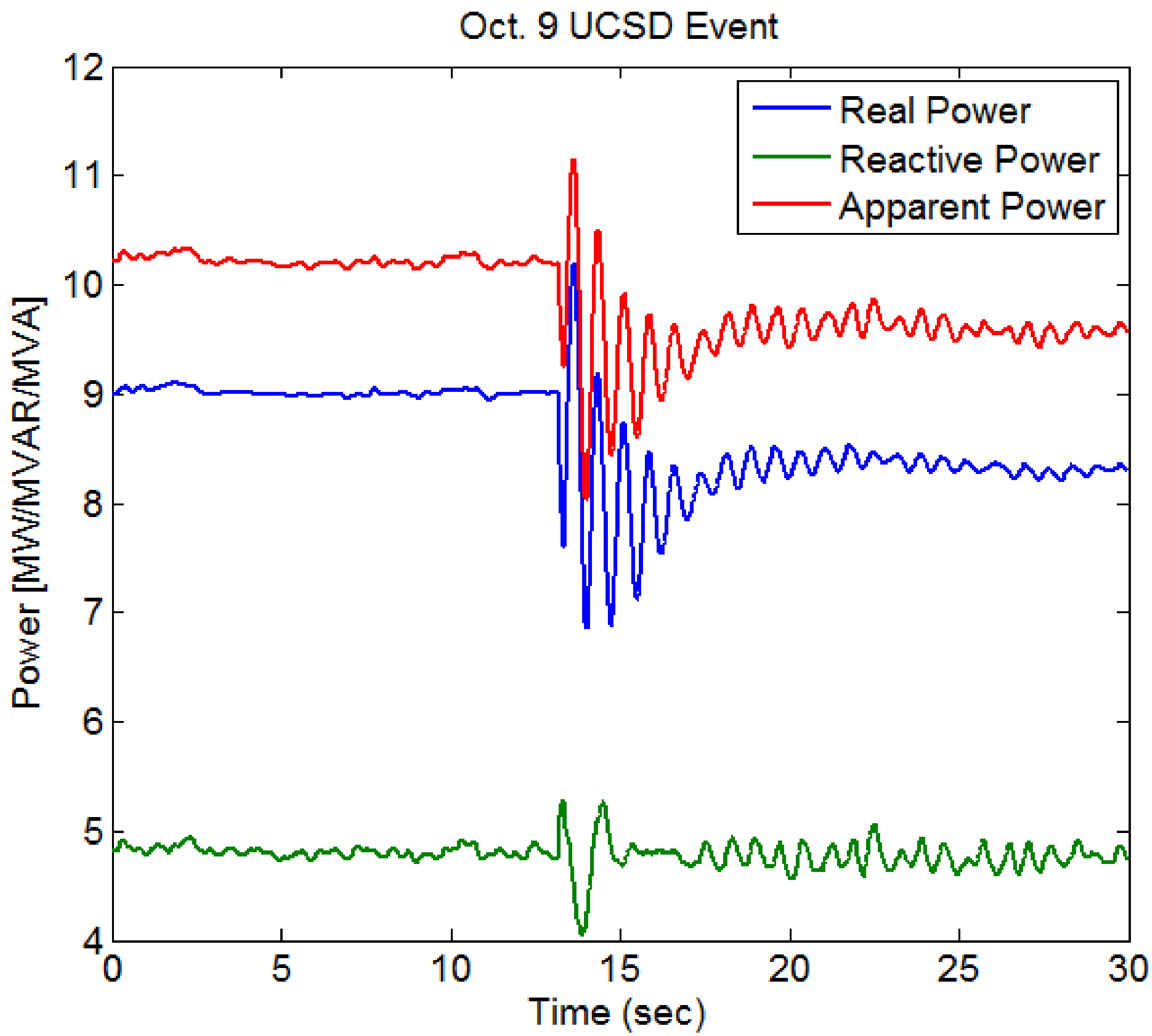


Oct. 9 UCSD microgrid event

Measurements from SEL breaker at 12kV

3 phase line (6.9kV phase to phase)

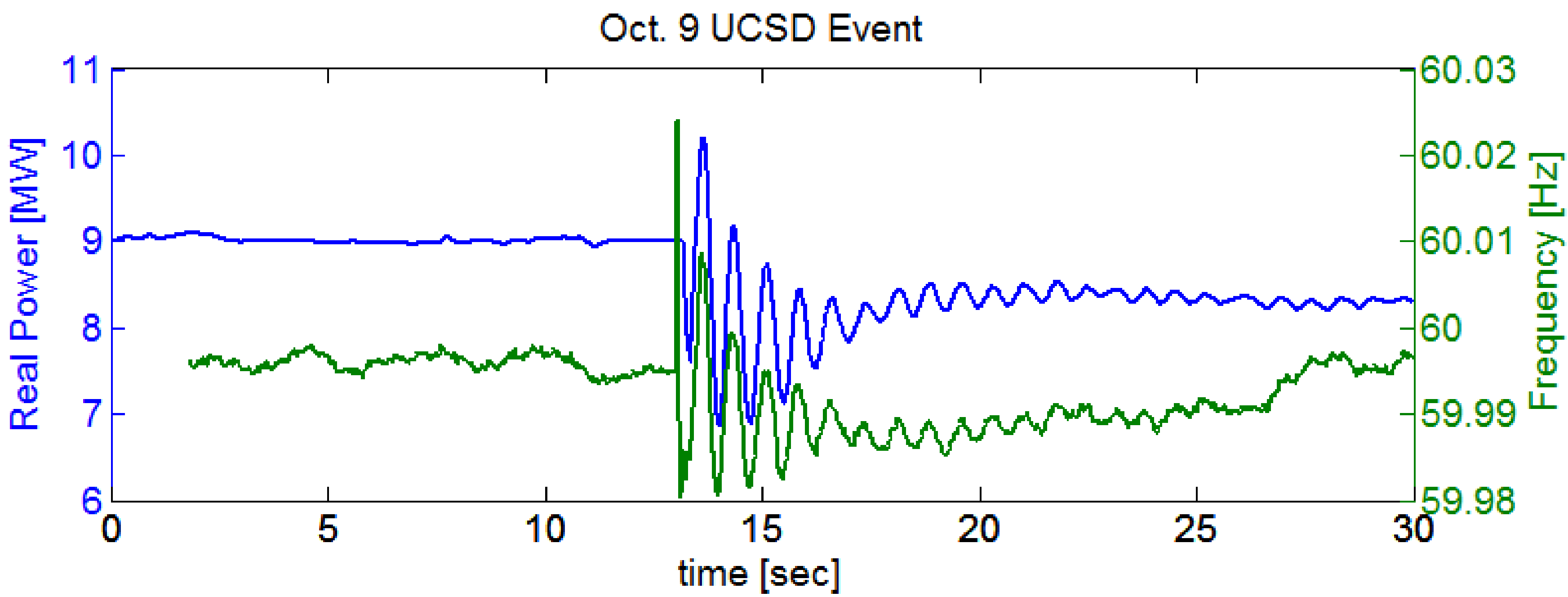
- RMS Voltage and Current of 3 phases
- Real Power
- Apparent Power



Oct. 9 UCSD microgrid event

Main conclusions from Measurements from SEL breaker:

- Sustained oscillations in 3 phase V and I mostly due to reactive power.
- Real power oscillations dampen out faster
- (time adjusted) Frequency show similar dynamics as Real Power:

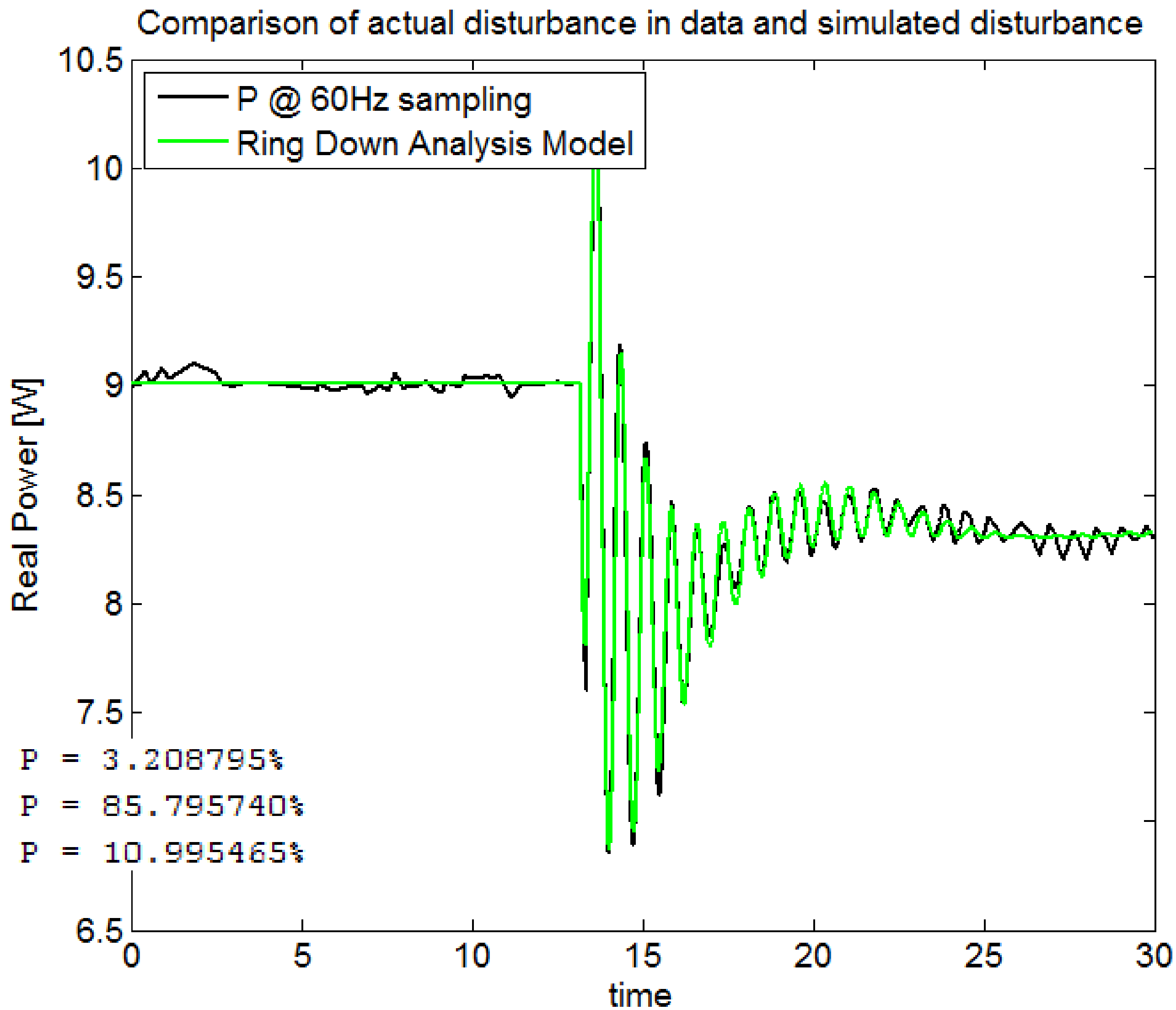


Analysis of UCSD microgrid dynamics

Realization Algorithm:

excellent fit of oscillation/damping

Modeled frequencies Fn, damping D and model participation P:




```
Fn = 0.094653 Hz, D = 0.450955, P = 3.208795%
Fn = 1.353568 Hz, D = 0.044507, P = 85.795740%
Fn = 1.461354 Hz, D = 0.026519, P = 10.995465%
```

Mode around 1.4Hz
less than 5% damping, 85% participation

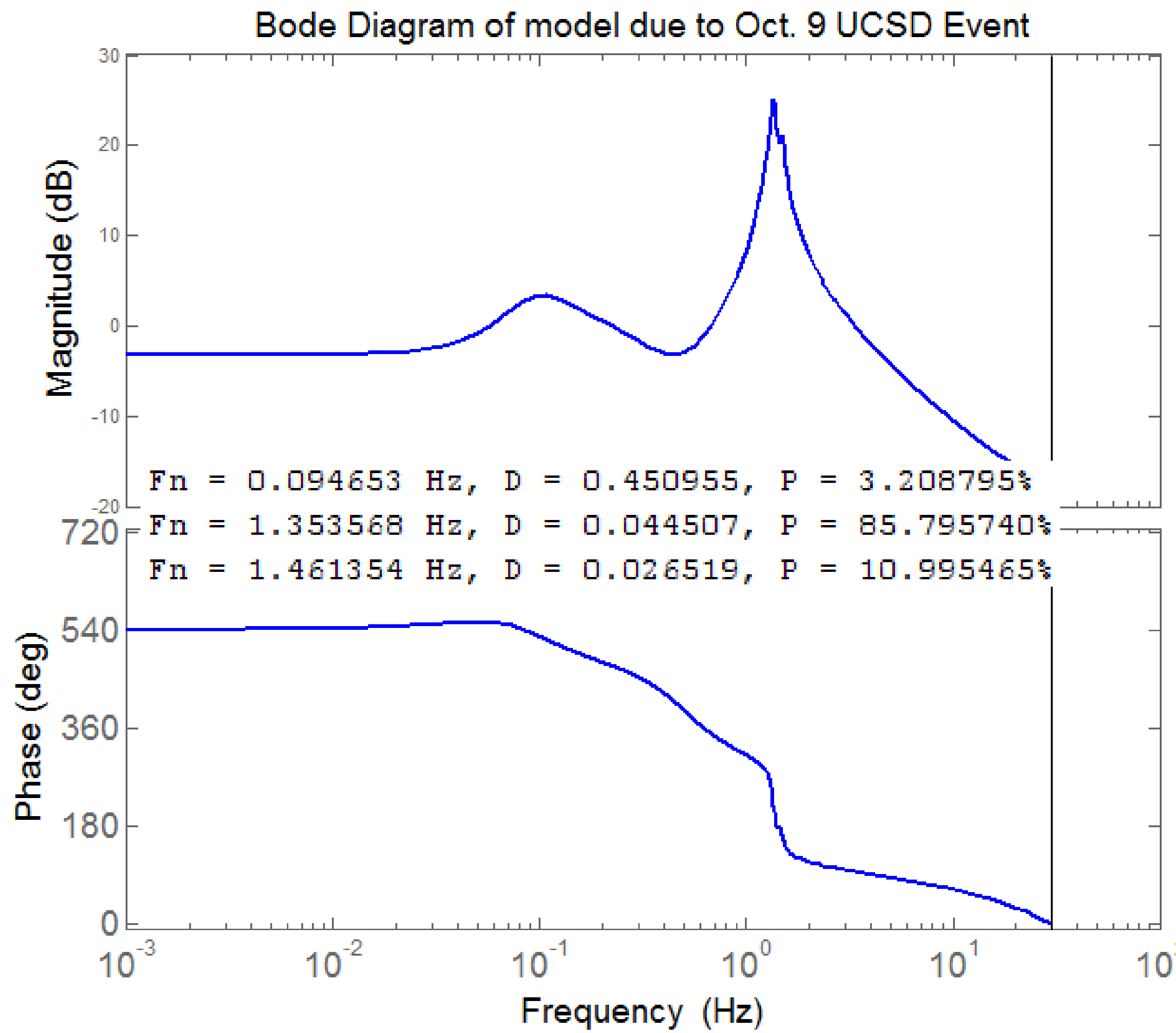
Analysis of UCSD microgrid dynamics

Dynamic model found by realization in Bode plot (frequency domain)

Observe large resonance frequency around 1.4Hz

MITIGATION

Control/damping of 1.4Hz oscillation

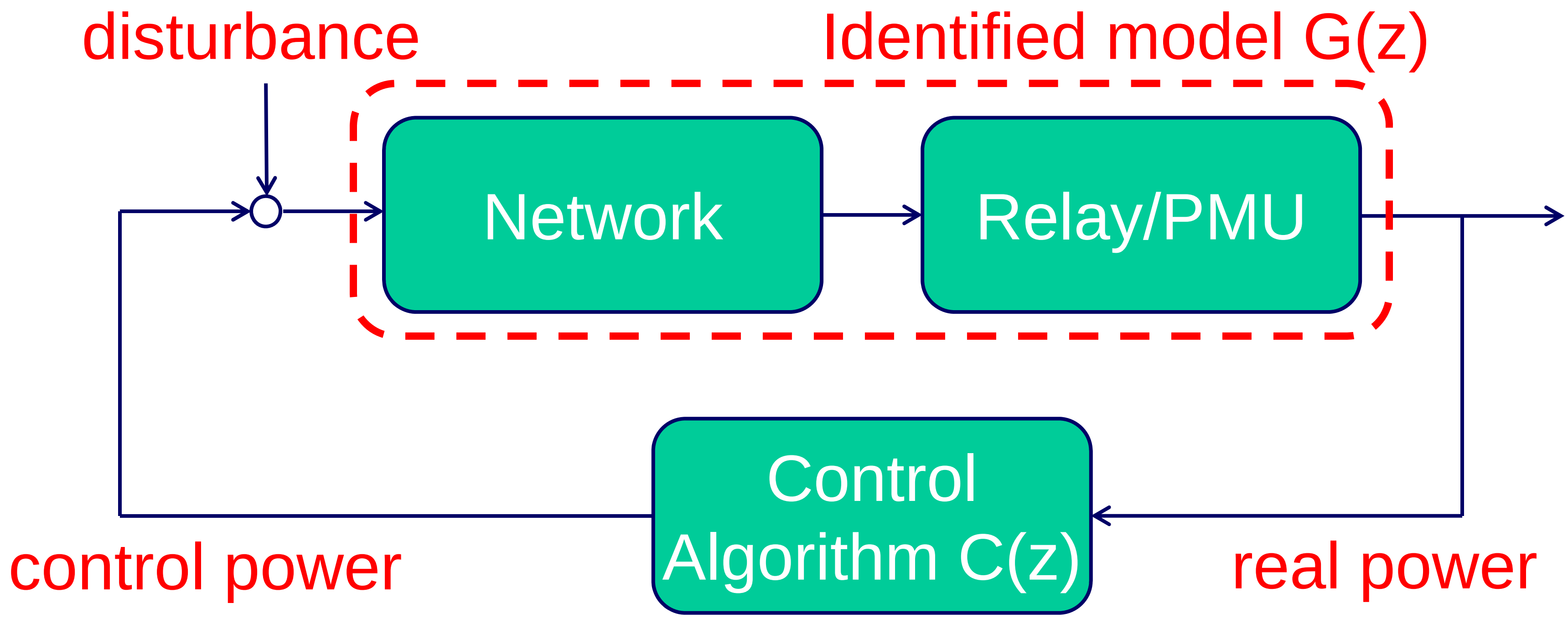


UCSD Jacobs | Mechanical and Aerospace Engineering

Mitigation of UCSD microgrid dynamics

MITIGATION

Control/damping of 1.4Hz oscillation via Real Power control:



- What is the control algorithm?
- How much control power is needed to dampen oscillation?

Mitigation of UCSD microgrid dynamics

Identified Discrete-Time Model G(z):

$$G(z) = \frac{-0.2791\, z^{\wedge}6 + 1.677\, z^{\wedge}5 - 4.204\, z^{\wedge}4 + 5.63\, z^{\wedge}3 - 4.249\, z^{\wedge}2 + 1.713\, z - 0.2882}{z^{\wedge}7 - 6.89\, z^{\wedge}6 + 20.39\, z^{\wedge}5 - 33.58\, z^{\wedge}4 + 33.26\, z^{\wedge}3 - 19.8\, z^{\wedge}2 + 6.564\, z - 0.9344}$$

Proposed control algorithm C(z) that has the following shape:

$$C(z) = K\frac{z-1}{(z-a)(z-b)}$$

- Discrete-time differentiator (to add damping + reduce low frequency control)
- Two poles (a,b) to limit bandwidth
- Gain K to adjust power gain

Mitigation of UCSD microgrid dynamics

Choice of control parameters K, and b in

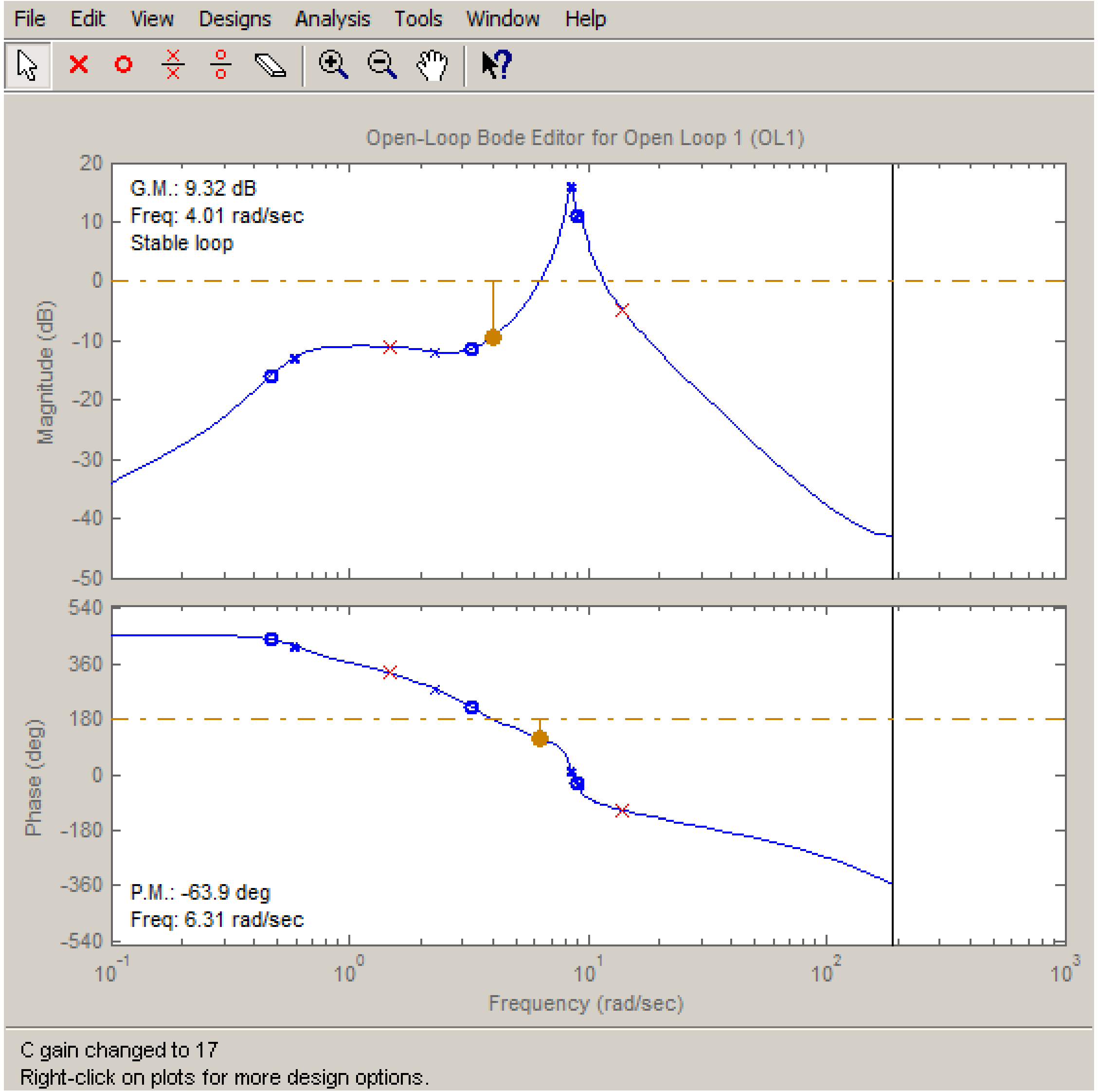
$$C(z) = K\frac{z-1}{(z-a)(z-b)}$$

via loop shaping tool

Shape Bode plot of L(z)=G(z)C(z)

See direct effect of:

- Damping
- Stability
- Control signal



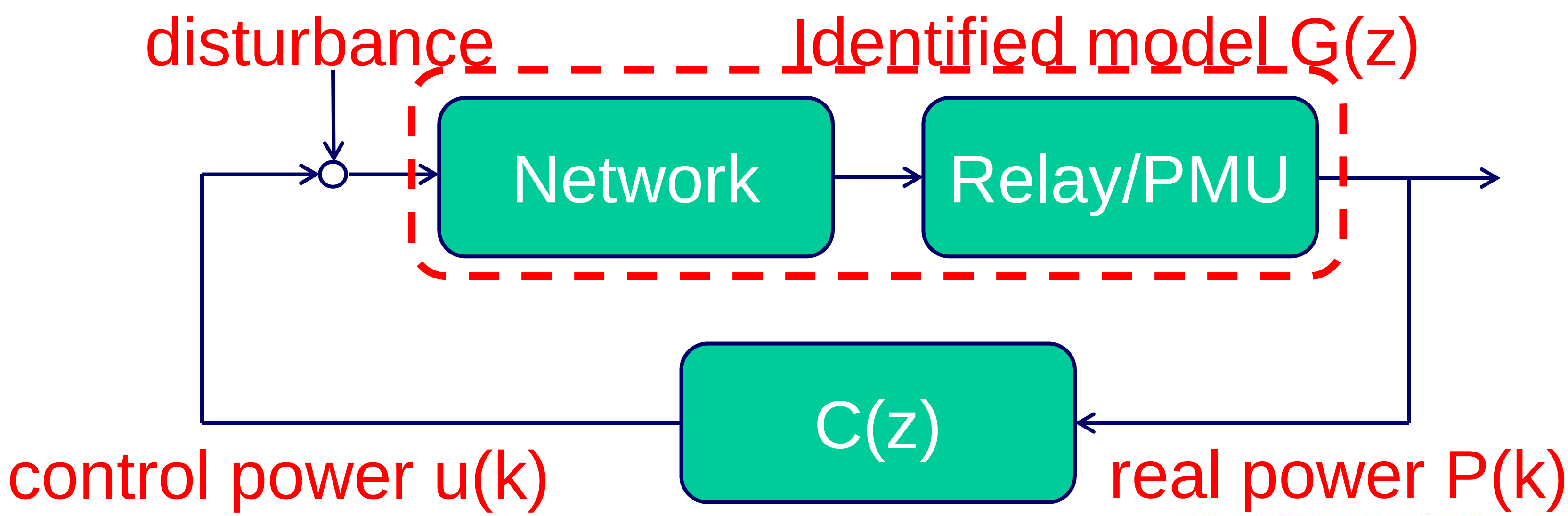
Mitigation of UCSD microgrid dynamics

End result of control design:

$$C(z) = K\frac{z-1}{(z-a)(z-b)}, K = 0.085211, a = 0.9757, b = 0.7933$$

Resulting discrete control algorithm:

$$u(k) = 0.0852 \cdot P(k-1) - 0.0852 \cdot P(k-2) + 1.7690 \cdot u(k-1) - 0.7740 \cdot u(k-2)$$



Mitigation of UCSD microgrid dynamics

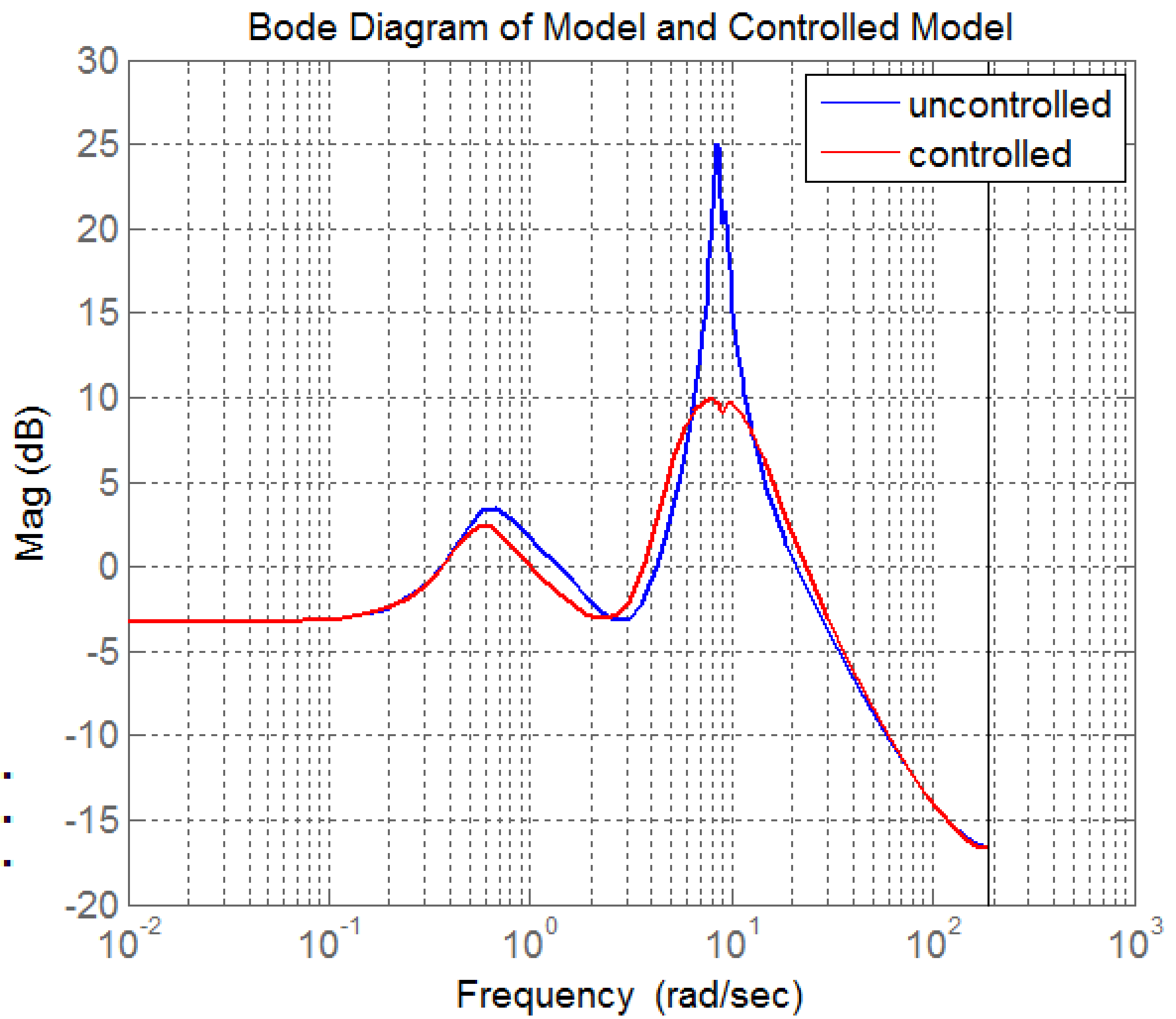
Effect of Control Algorithm:

Damping of UCSD microgrid:

```
Fn = 0.094653 Hz, D = 0.450955.
Fn = 1.353568 Hz, D = 0.044507.
Fn = 1.461354 Hz, D = 0.026519.
```

Damping of controlled UCSD microgrid:

```
Fn = 0.089560 Hz, D = 0.445131.
Fn = 0.904540 Hz, D = 0.415226.
Fn = 1.771599 Hz, D = 0.502977.
```

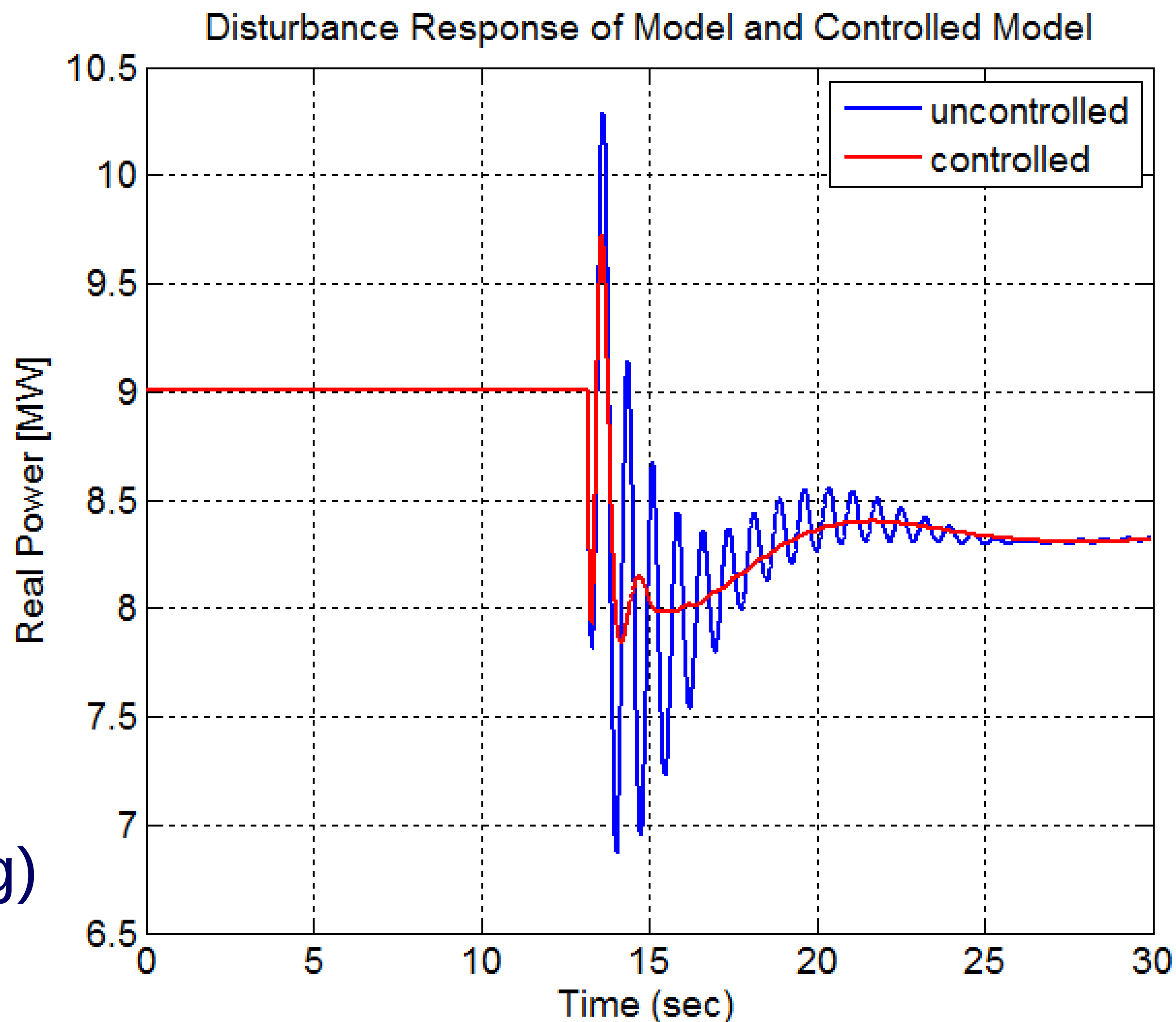


Slight change in resonance modes, ten-fold increase in damping!

Mitigation of UCSD microgrid dynamics

Effect of Control Algorithm:

- Disturbance effect still present (unavoidable)
- Control algorithm does mitigate disturbance faster!
- Less oscillations in microgrid (better damping)

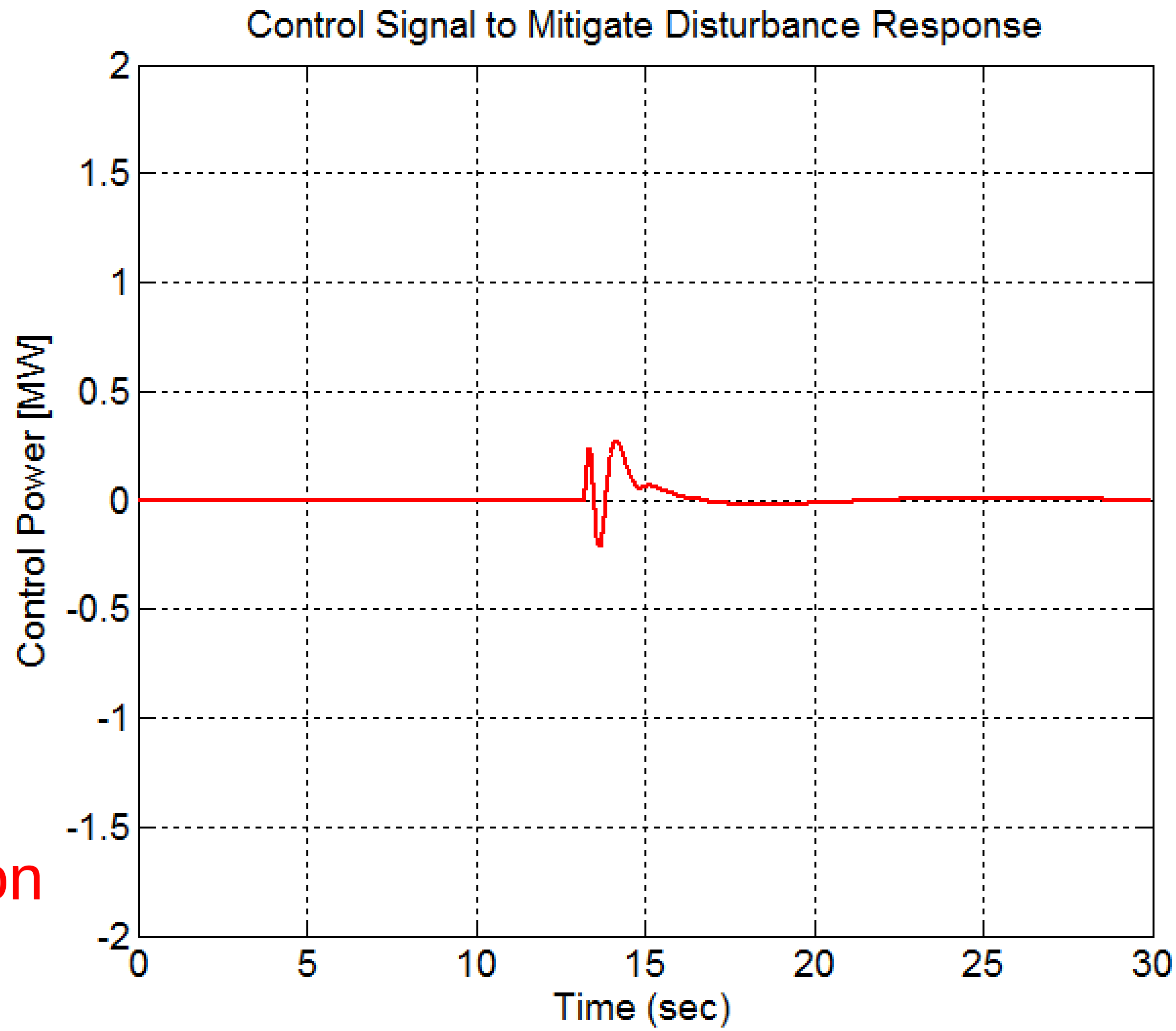


- How much control power needed?

Mitigation of UCSD microgrid dynamics

Effect of Control Algorithm:

- For comparison, control power plotted at same scale a disturbance in real power
- Disturbance almost +/- 2MW
- Control power only +/- 0.25MW for mitigation



- Results scale with size of disturbance and increase of damping

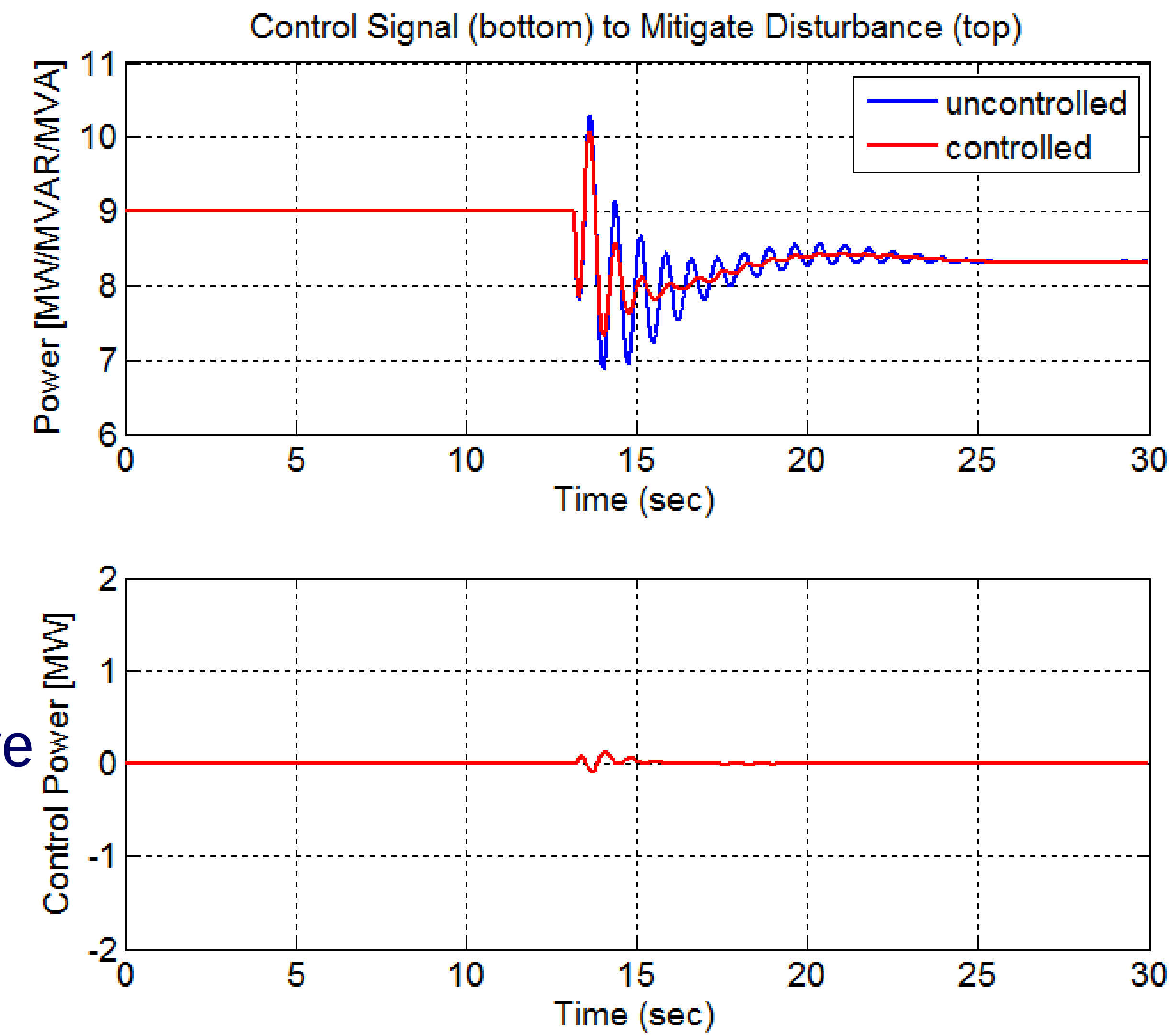
Mitigation of UCSD microgrid dynamics

Reducing control effort to +/- 125KW still works, but:

- Damping cannot be influenced that much

```
Fn = 0.092977 Hz, D = 0.448233.
Fn = 1.349573 Hz, D = 0.132450.
```

- Still acceptable to improve dynamics of microgrid
- Control power only +/- 125KW for mitigation



Summary on Detection and Analysis

- Automatically detect when a disturbance/transient event occurs
- Automatically estimate Frequency, Damping and Dynamic Model.

Main Features:

- Automatically detect event:
  - Predict ambient Frequency signal "one-sample" ahead
  - Observe when prediction deviates for event detection via FRoC signal
- Automatically estimate:
  - # of modes of oscillations in measured disturbance
  - Estimate frequency and damping of the modes
  - Put results in dynamic mode
- All done in real-time!
- Note: resulting dynamic model can be used for feedback control design to mitigate event!

- Thank you