Joint Laboratory for Petascale Computation

INRIA NCSA

Fault analysis framework

Ana Gainaru, Franck Cappello, Bill Kramer

INRIA

NCSA

Contents

- Introduction
- Framework architecture
- Signal analysis module
  - Extract signals
  - Aperiodic events – Correlations
  - Periodic events – Adaptive Filter

Framework overview

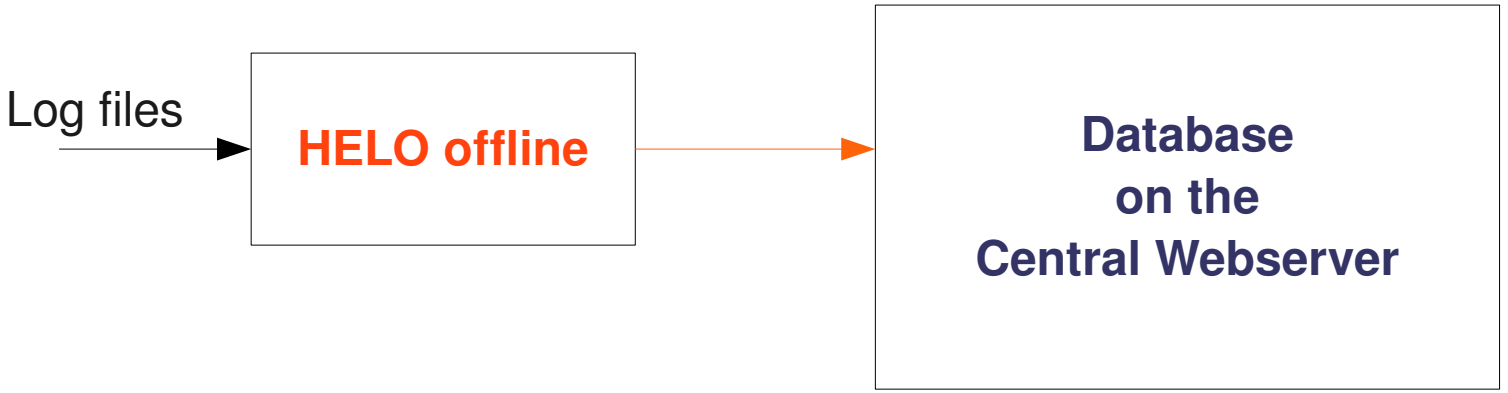
- Central database
- Different synchronized modules
  - HELO (Hierarchical Event Log Organizer) extracts patterns from logs generated by the system
  - Other modules in pipeline with HELO
    - Have the input received by the output from HELO

HELO Overview

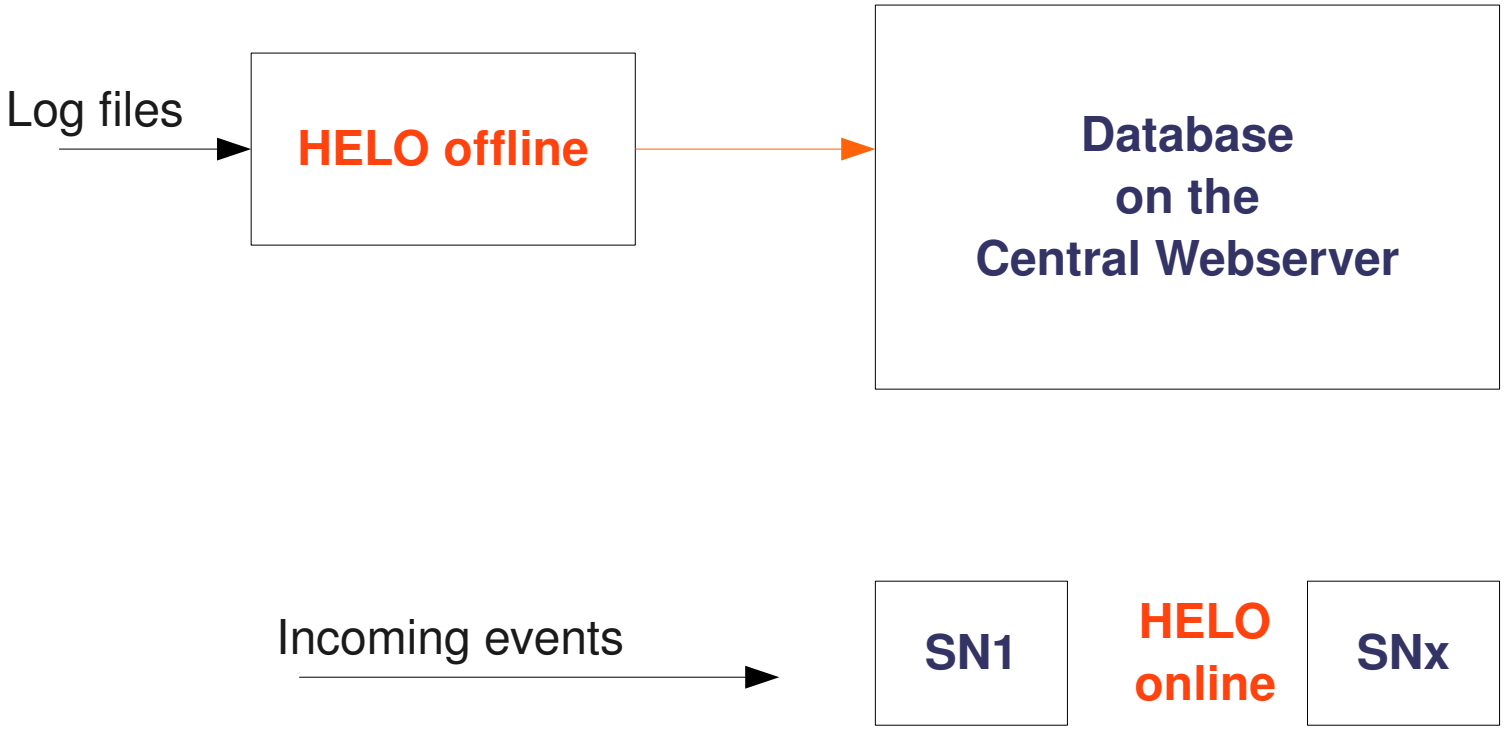
- Characterize events generated by large systems
- Extract patterns from historic log files
- Dynamically adapts templates as novel events appear in the system
    - Deals with configuration changes or system updates
- Analyze the normal behavior for every type of signal

**Database
on the
Central Webserver**

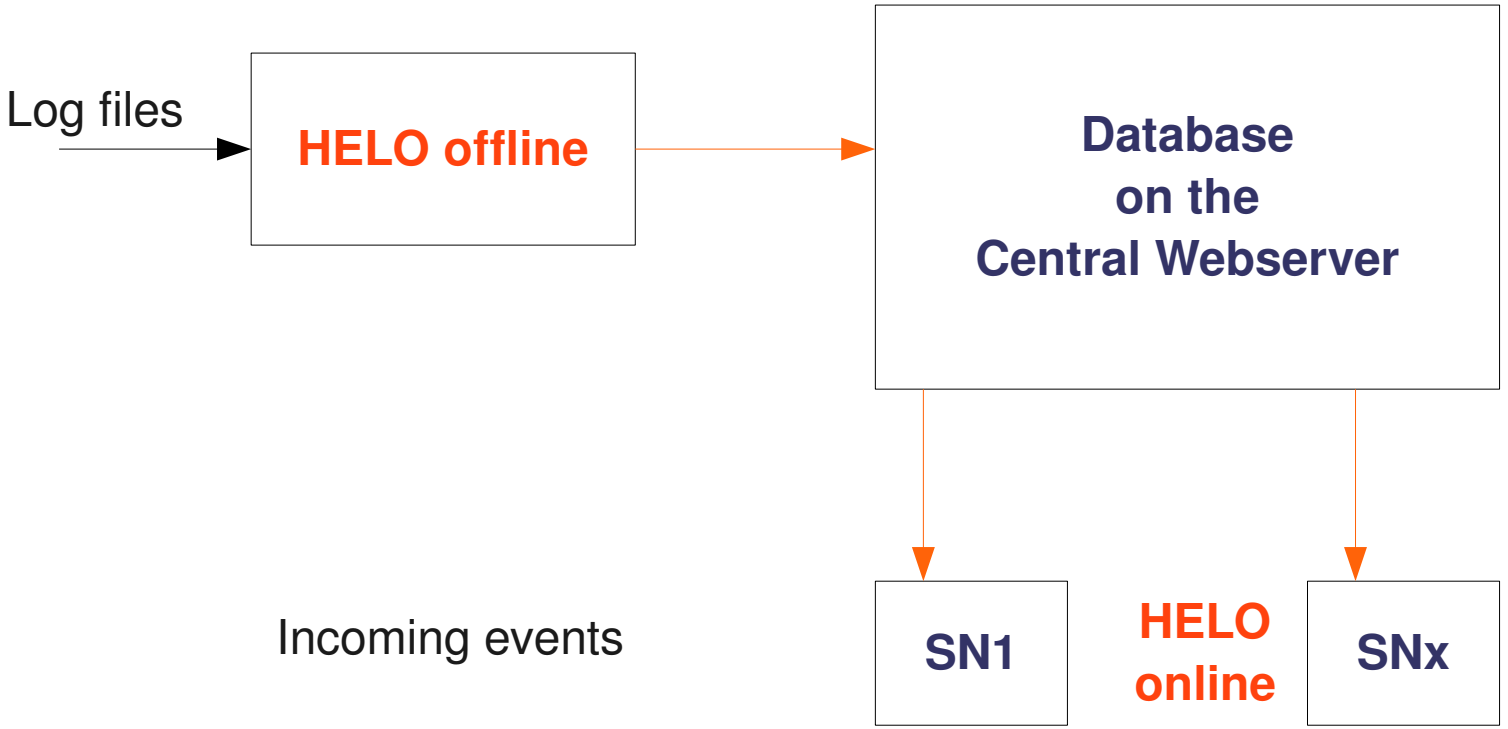
- Central database to keep synchronized templates



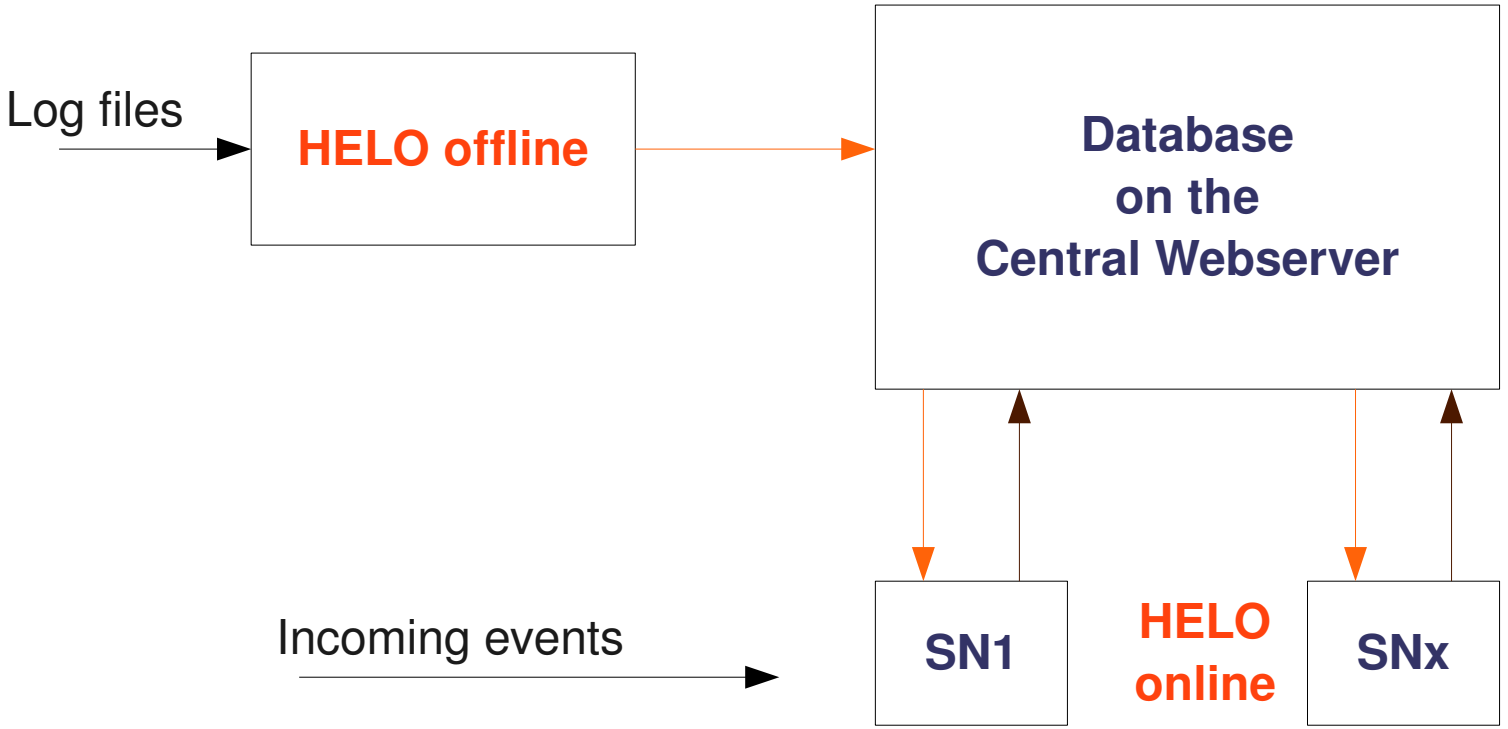
- Hierarchical Event Log Organizer (HELO)
  - extracts patterns from the log file - offline manner



- For incoming events
  - Each service node runs HELO online independently



- Each Service Node (SN)
  - Downloads templates from the database
  - Inspects each incoming event
    - If the new event is a match – classify (parallel for each SN)



- For a new event type
  - SN - Send the event to the database
  - Database – Updates the templates
  - SN - Receive the new template list

Analysis modules

- Different modules can be added in the framework
  - Synchronized by the database
  - In a pipeline manner
- Each time a SN modifies the database
  - The modules update their data

Signal analysis

- The system generates signals
  - Daemons
  - Monitoring information
- Errors generate noise
  - Non periodic
  - Influence other events
- Module aim
  - Extract the normal behavior of a system
  - How error change the shape / propagate in the system

Extracting the signal

- Nyquist theorem

  If a function x(t) contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced 1/(2B) seconds apart.

- Sampling rate
  - Respecting Nyquist theorem
  - Make computations fast but with a small delay

Extracting the signal

- Algorithm
  - Extract the mean time between two events from the same type
    - Errors could create more messages from one type
    - Errors could make some messages dissapear
  - Start with the mean time until max time
    - Use autocorrelation function for time sample of t/2 until we get a periodic signal

Autocorrelation

- Cross-correlation
  - A measure of similarity of two waveforms as a function of a time-lag applied to one of them
- Autocorrelation
  - Cross-correlation of a signal with itself
- Thresholds
  - R coefficient of correlation – between -1 and 1
  - Testing for the significance of the correlation coefficient

Results

- 50% periodic signals
  - For all periodic signals the sampling rate was correct
  - Using the constant threshold 10% of the periodic signals are not found
  - Using adaptive threshold 30% of the found signals are not periodic
  - Non periodic events are detected in the filtering stage

Aperiodic signals

- Time window of 1 sec – offline analysis
- Correlations
  - Similarity value – the same as autocorrelation
  - Extract the time difference between signals
  - Graph of correlated events

Example

INFO starting systemcontroller *

WARNING endserviceaction is restarting the nodecards in midplane * as part of service action d+

WARNING node card is not fully functional

SEVERE node card vpd check: * node in processor card slot * do not match. vpd ecid d+ found d+

ERROR can not get assembly information for node card

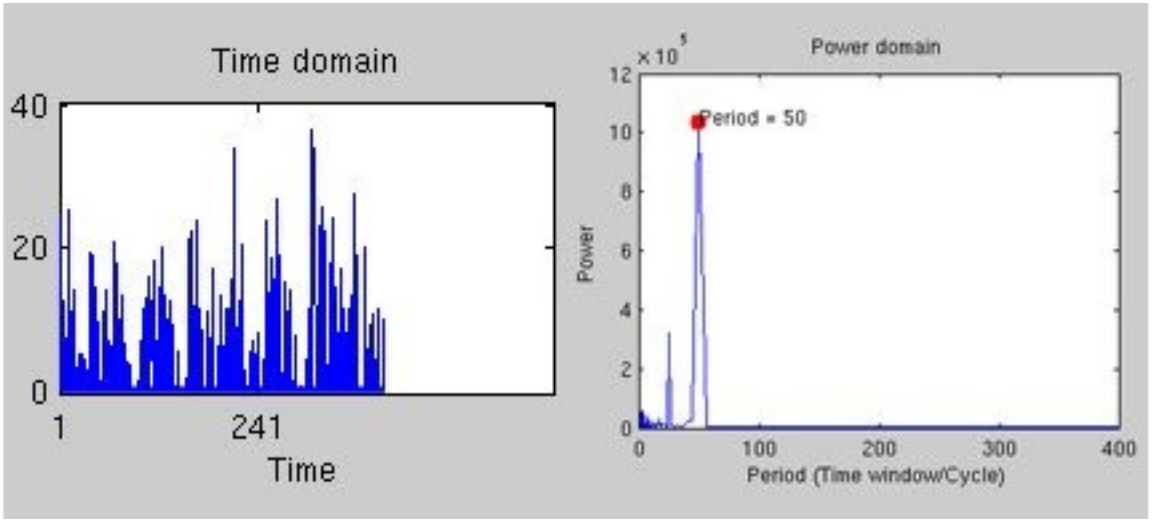
SEVERE link card power module * is not accessible

FAILURE no power module * found found on link card

FAILURE temperature Over Limit on link card

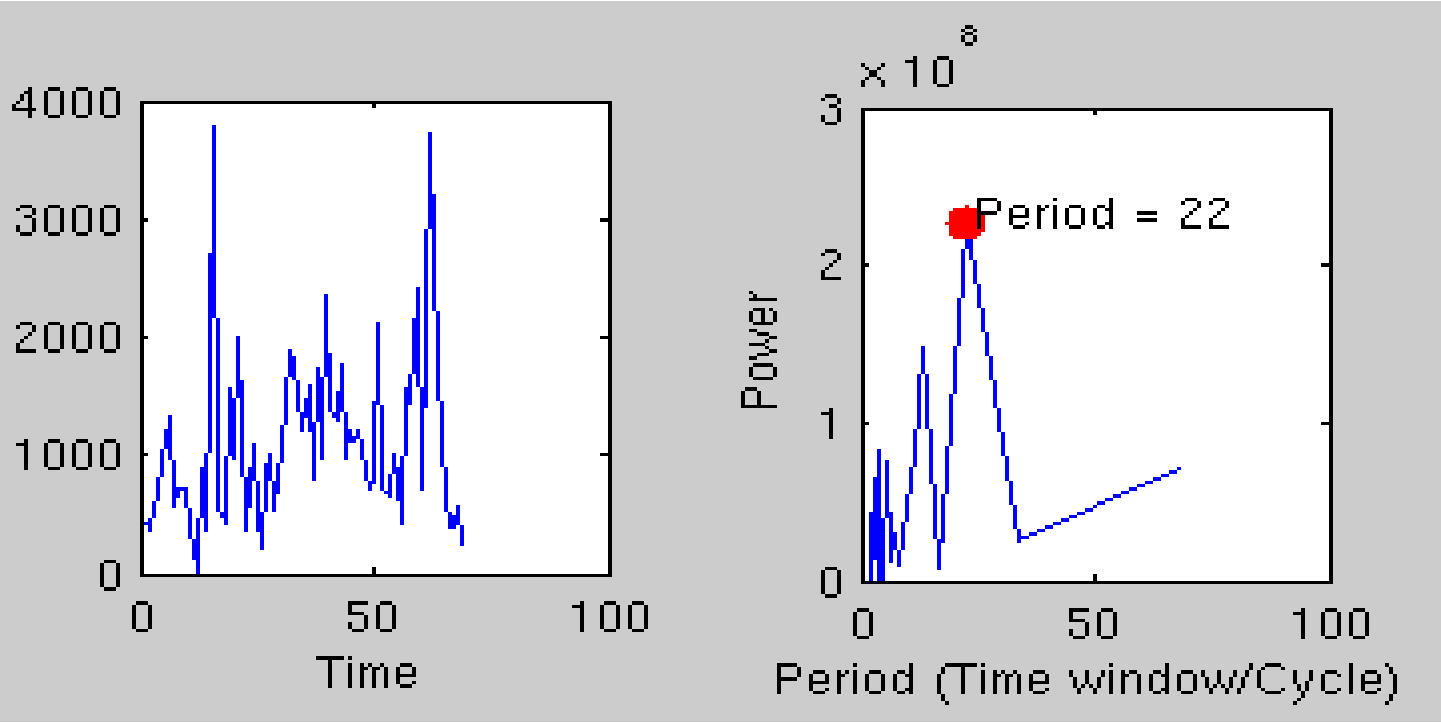
Period signals

- Periodogram
  - If a time series has a strong sinusoidal signal for some frequency, then there will be a peak in the periodogram at that frequency



Filtering signals

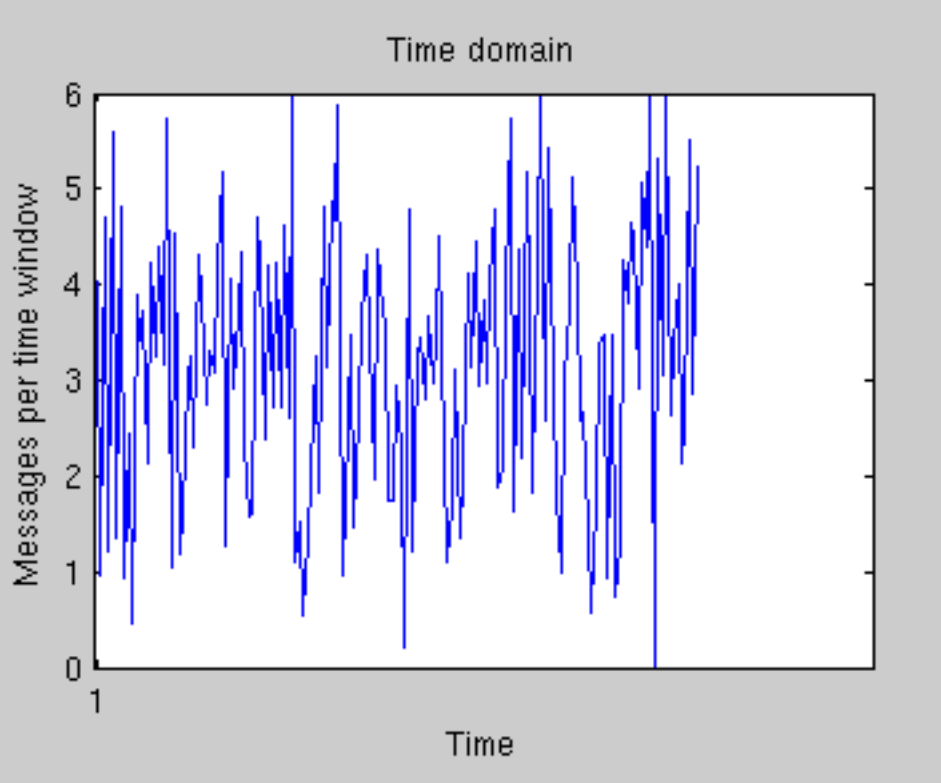
- In case of errors



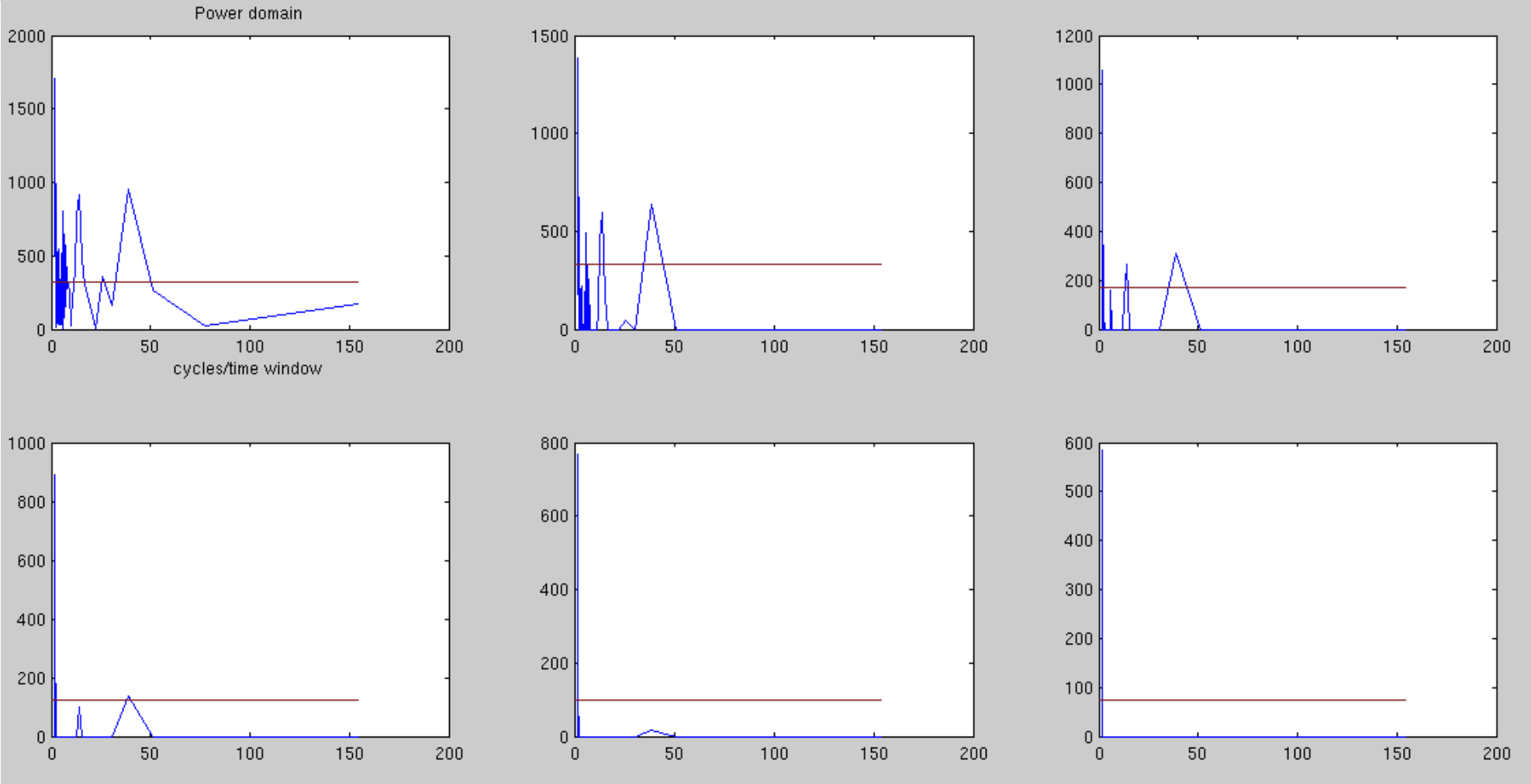
Filtering signals

- Algorithm
  - The frequencies that are prominent
  - Iterative
    - Find the horizontal line that divides the frequencies values into equal sets
    - Eliminate all under the middle + 2 standard deviation
      - Only keep 5% of all values
    - If nothing changes in two iterative steps -> stop
  - Local maximum → frequency

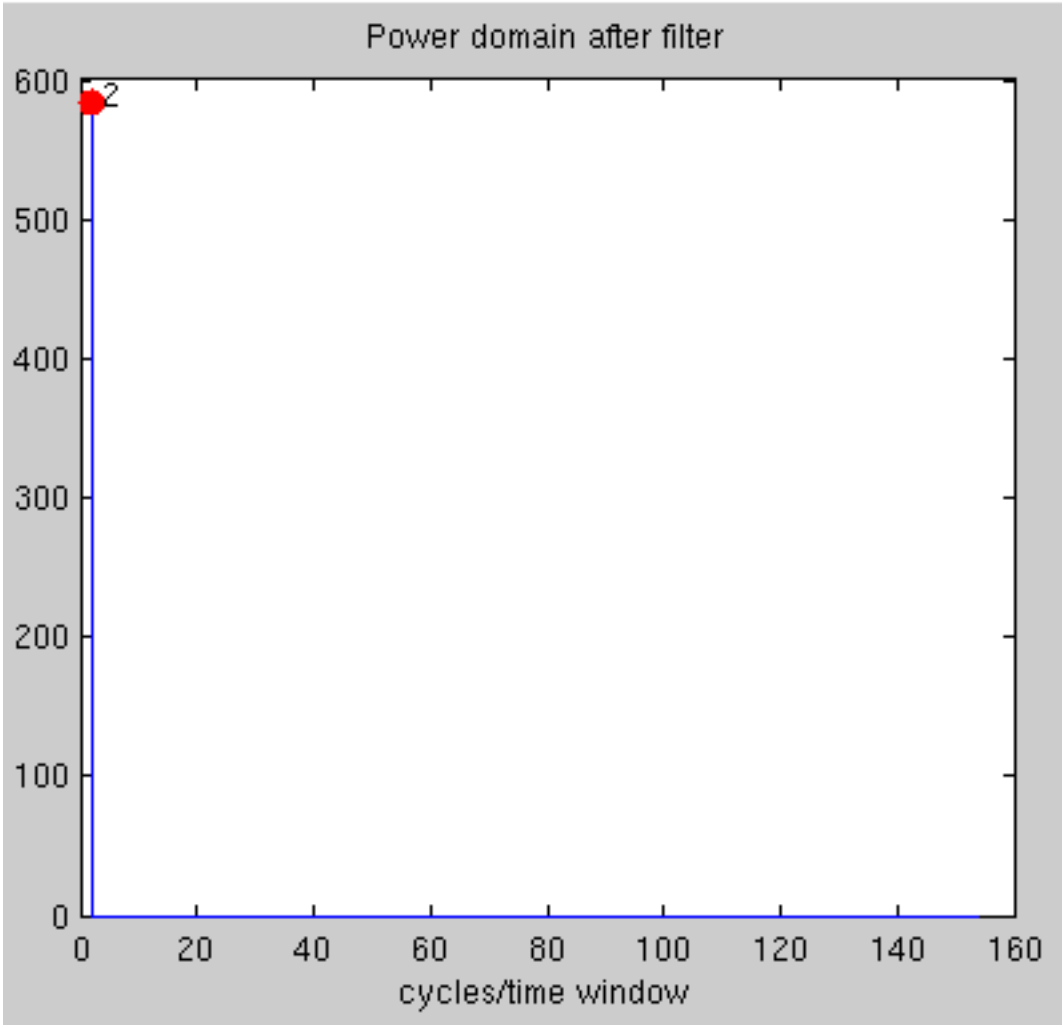
Filtering signals



Filtering signals



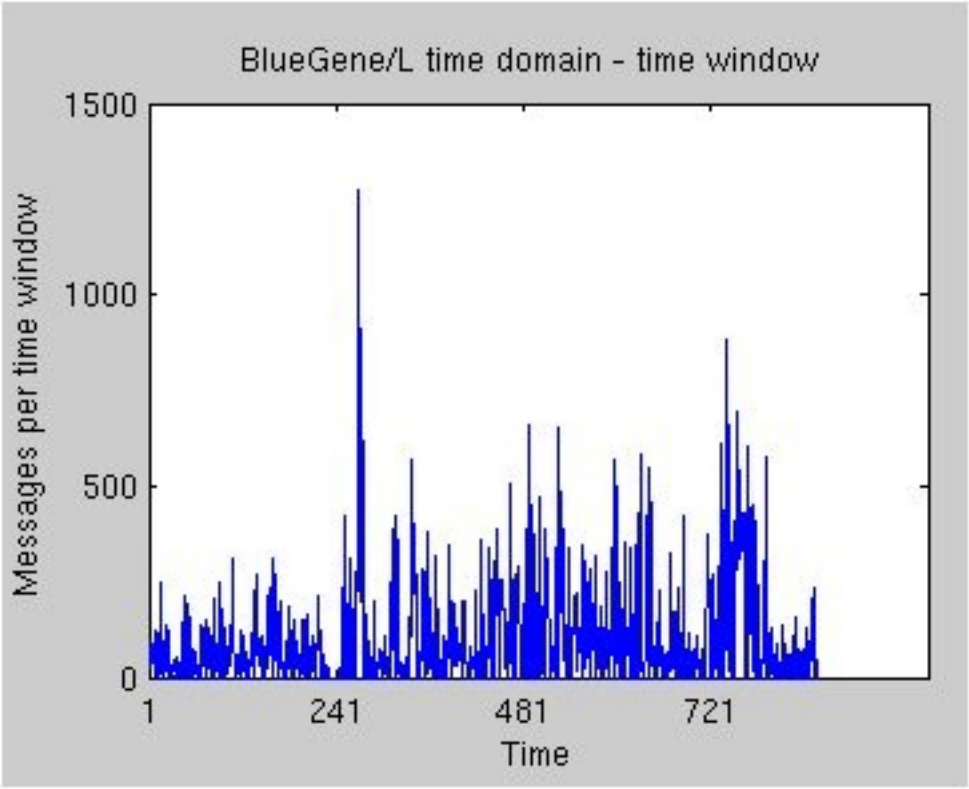
Filtering signals



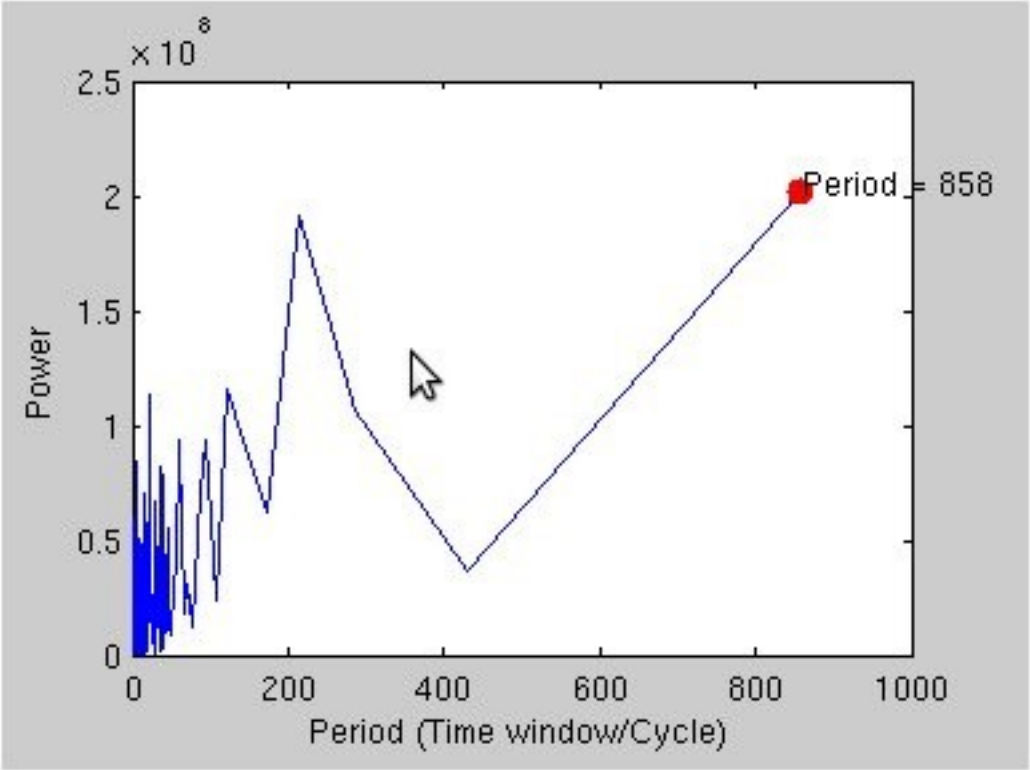
Results

- If the noise is:
  - 5 times more than the signal – new frequencies
- Analysis:
  - 3 weeks without errors
    - Extracted the right signal 100% correct
  - One month with errors
    - 10 periodic signals – 2 for which the noise error was too high to find the right frequencies

Results


BlueGene/L time domain - time window
Messages per time window
1500
1000
500
0
1
241
481
721
Time

Results


x 10^8
2.5
2
1.5
1
0.5
0
Power
Period = 858
0
200
400
600
800
1000
Period (Time window/Cycle)

Conclusion & Future work

- Offline analysis
  - Extracting the normal behavior for periodic signals
  - Correlated events
- Online analysis
  - How does an error change the normal behavior
    - Same frequencies but higher values
    - Other frequencies appear
    - Error propagation in the correlated chain

Q&A

Thank you