Network Analysis For Coalescing Binary

(or any analysis with Matched Filtering)

Benoit MOURS,
Caltech & LAPP-Annecy

March 2001, LSC Meeting
Baton Rouge, Louisiana
LSC Session: ASIS

LIGO-G010096-00-E

The basic issue

- What is the best way to analyze data from 2 or more detectors?

Combine the data streams and analyze it

or

Analyze each stream and combine the results?

- How this could be applied/tested on a single detector

The principles

- For one detector one has to compute:
  - » $$LLR(t_c, M) = \int_{f\,\min}^{f\,\max} c^{-1}(f)h(f)a(\Omega)t(M, t_c, f)df$$

    a = detector/source location
    h = data
    t = template
    c = noise
  - » The orientation part could be taken out of the integral:

$$LLR(t_c, M) = a(\Omega)\int_{f\,\min}^{f\,\max} c^{-1}(f)h(f)t(M, t_c, f)df$$

- For more than one detector
  - » Method 1: build a new h'(t) which is a linear combination of each h(t).
  - » Method 2: a,h,t,c become matrices.

Multi detector: Linear Combination

- Build a new ‘pseudo detector’: a linear combination of the detector outputs.
  - » Need proper whitening/weighting to not bias the data.
  - » Need new templates: linear combination of the single templates
- Problem:
  - » If the detector are at different locations, the location dependant part of the template could not be factorized.
  - » For LLH-LLO orientation is about the same, need to include the time delay: data flow increased by one or two orders of magnitude.
  - » For LIGO-GEO/VIRGO/TAMA the orientation is not the same. Many more orders of magnitude

Multi detector: Matrices

- Replace scalars by vectors (dimension: number of detectors).

$$LLR(t_c, M) = \int_{f\,\min}^{f\,\max} C^{-1}(f) H(f) A(\Omega) T(M, t_c, f) df$$

  - » Problem: If C is not diagonal (correlated noise between detectors), A and C do not commute and the integral could not be simplified
- Solution 0. Just compute the N_{det}^2 integrals
- Solution 1. Build a pseudo detector by mixing h(t)'s
  - » Then mix the templates $\Rightarrow$ still N_{det}^2 integrals to compute
  - » More complex templates, detectors problems mixed
- Solution 2. Correlation should be negligible

  (if not, the data contain a strong technical noise).

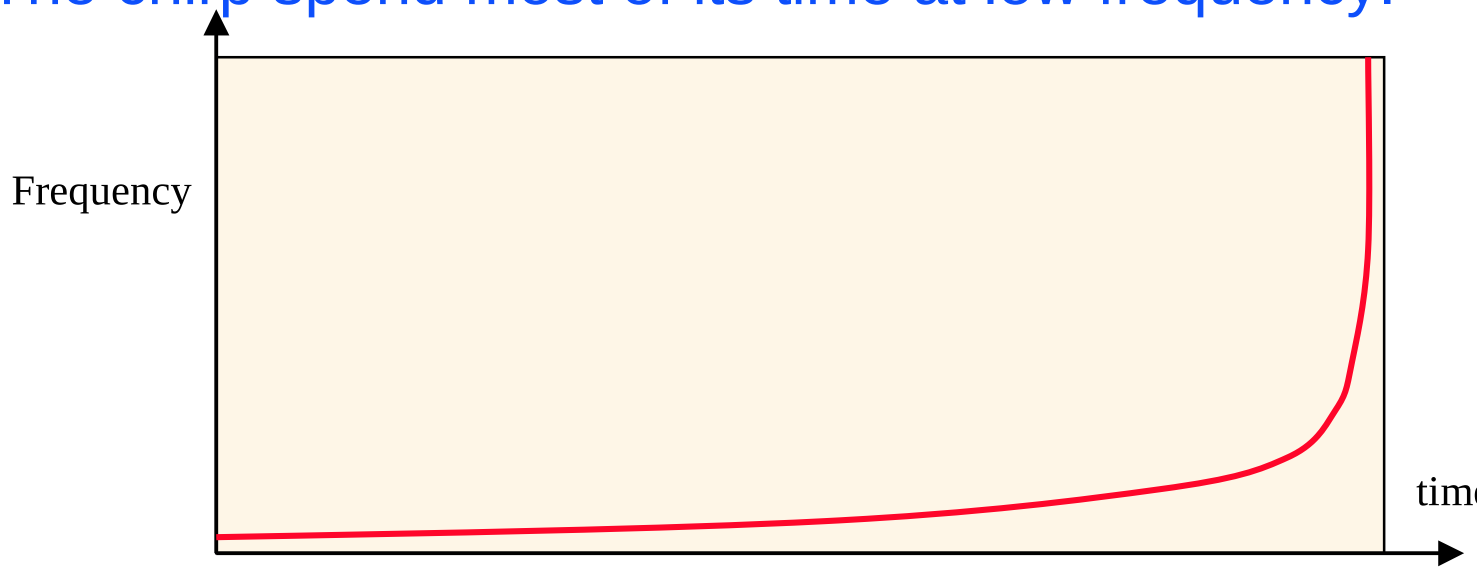
  N_{det} integrals to compute. The result is the sum of the individual analysis.

  $LLR = A_1 LLR_1 + A_2 LLR_2 + \ldots$

  - » Question: Could we easily sum single analysis??

Case of one single Detector

- The needed computing resources for binary search are large, especially :
  - » for low mass
  - » if we start at low frequency
- The chirp spend most of its time at low frequency.



- Waist of computing resources?

Single detector ?

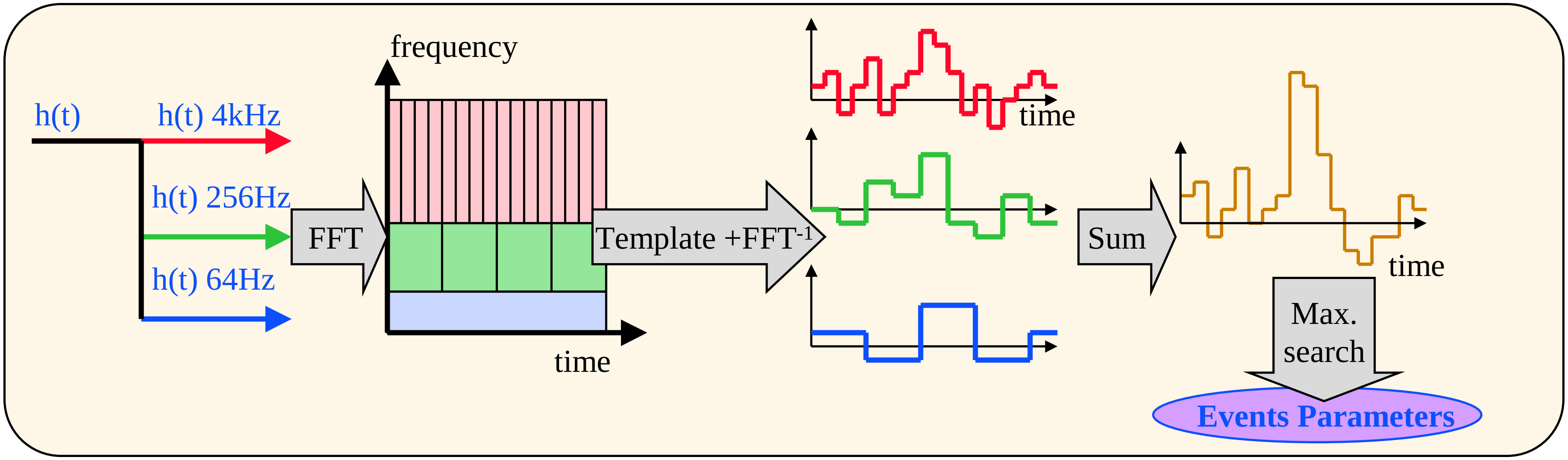
- Principle: Transform a single detector to a multiple one
  - » Split the analysis in a few frequency bands:

$$LLR(t,M) = \int_{f\,min}^{f\,max} h(f)T(M,f)df = \int_{f\,min}^{f1} h(f)T(M,f)df + \int_{f1}^{f\,max} h(f)T(M,f)df$$

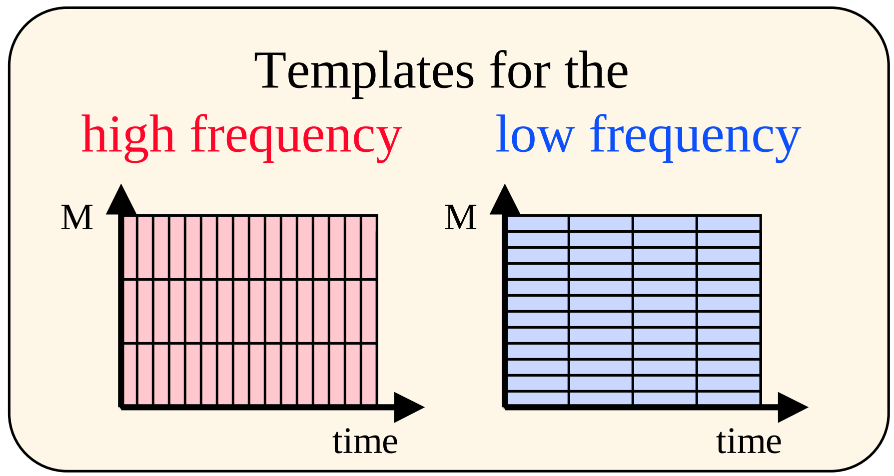
  - » Analyze independently each band
  - » Combine coherently the analysis result like for a network of detectors

- Remarks:
  - » The SNR should be unchanged.

The Multi Band Analysis Method



- Remarks:
  - » All sub analysis cover the same parameter space ***BUT*** may have different grids.
  - » Need interpolation to combine the results and search for the maximum.
  - » All FFT are small FFT.



Estimation of CPU resources

$CPU = K\, f_{min}^{-8/3}\, f_s\, \log_2(f_{min}\, f_{max})$ (simplify model: $N_{template}$.FFT cost)

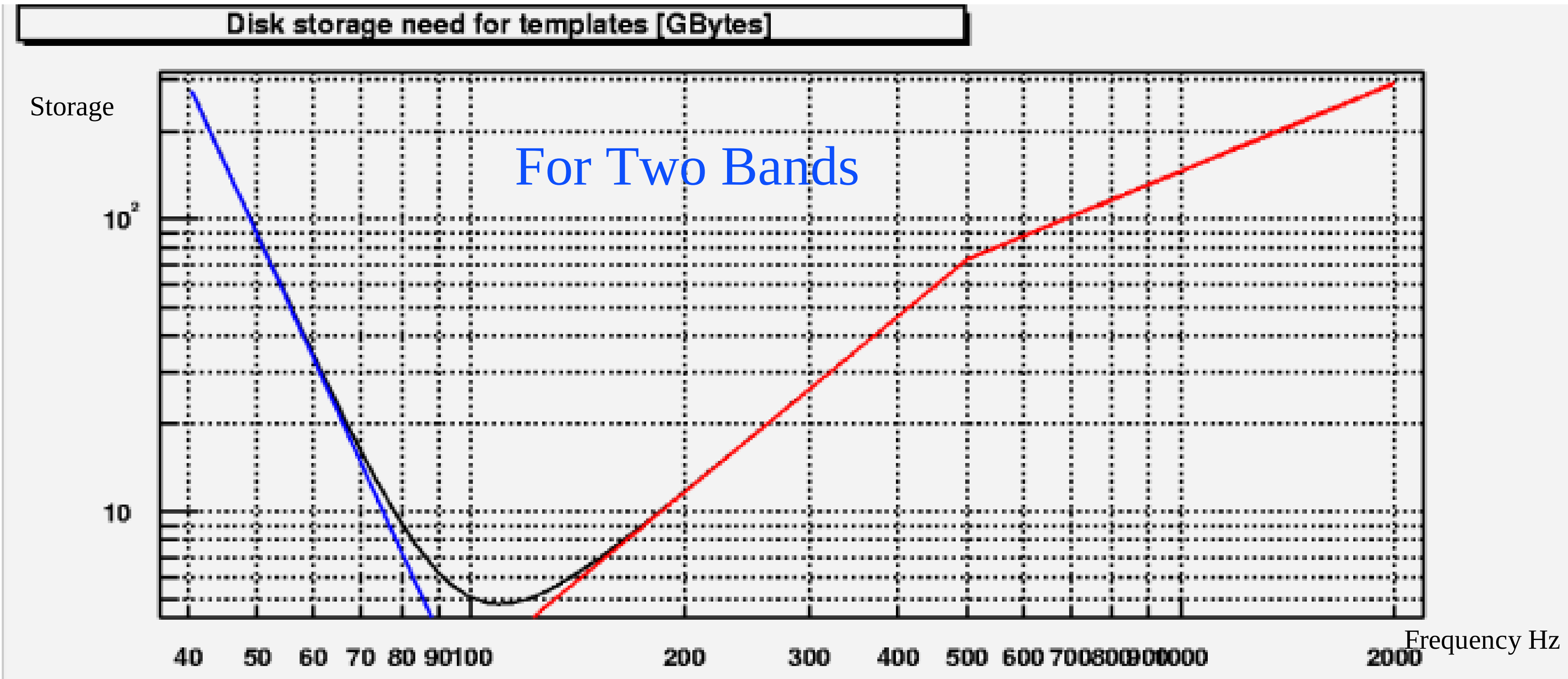
T = Template length (seconds). $= T_0 f_{min}^{-8/3}$

$N_{template}$= T/template spacing

$N_{sample} = 2T\, f_{max}$

$CPU = N_{template}\, 6 N_{sample} \log_2(N_{sample})/T = K\, f_{min}^{-8/3}\, f_{max}\, \log_2(f_{min}\, f_{max})$

Estimation Template Storage



$Storage = K\, f_{min}^{-16/3}\, f_s\, \log_2(f_{min}\, f_{max})$ (simplify model: $N_{template}$.tempSize)

T = Template length (seconds). $= T_0 f_{min}^{-8/3}$

$N_{template}$= T/template spacing

$N_{sample} = 2T\, f_{max}$

$Storage = 2\, N_{template} N_{sample} = K\, f_{min}^{-16/3}\, f_{max}\, \log_2(f_{min}\, f_{max})$

Estimation of computing resources

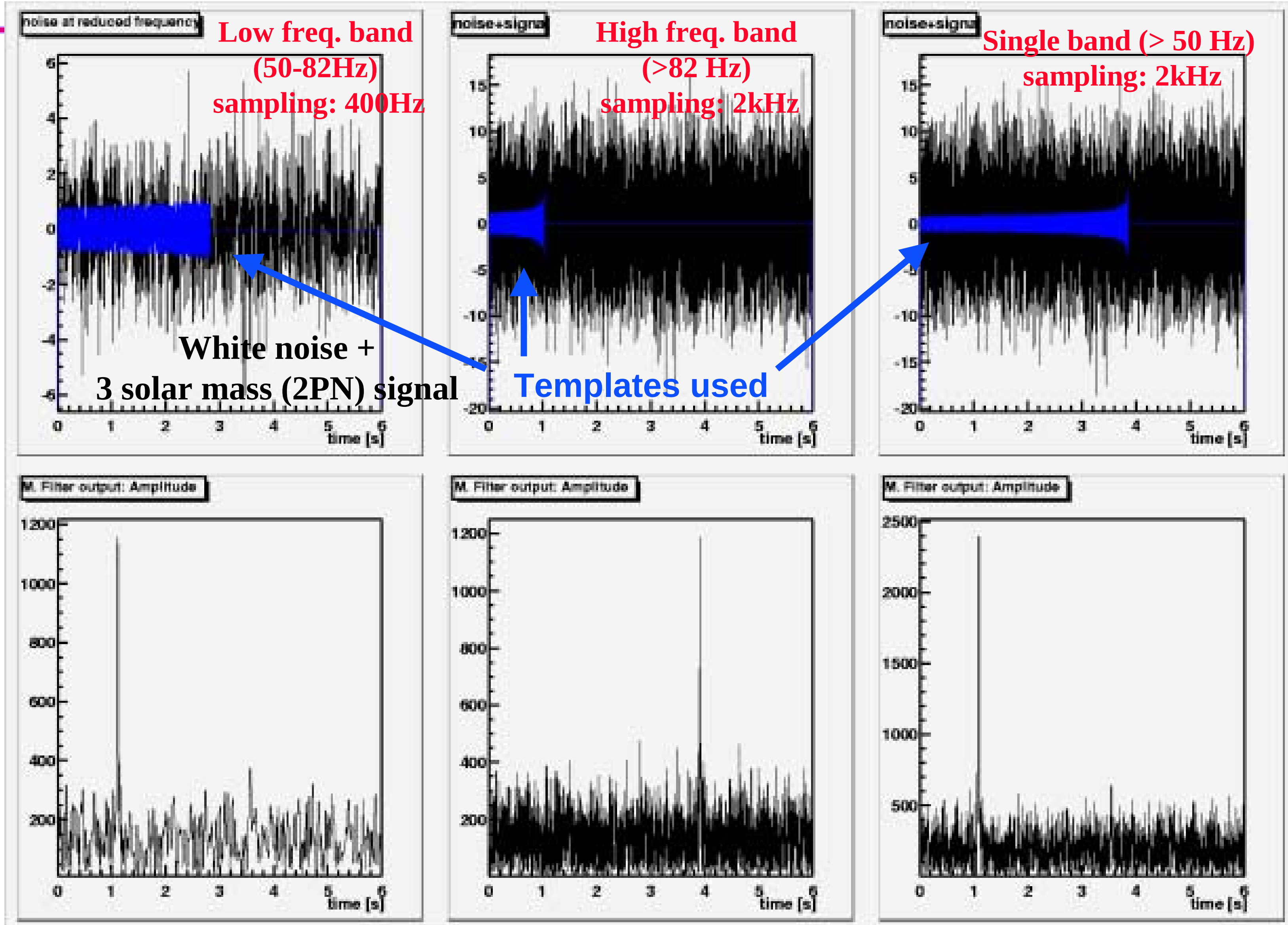
- If f_{min} = 40 Hz, f_{max} = 2kHz, M_{min} = 0.5M

	1 Band	2 Bands	3 Bands
CPU(Gflops)	30	1.3	0.6
Storage (Gbytes)	300	5	2.4
T. size (Mbytes)	2	0.13	0.04

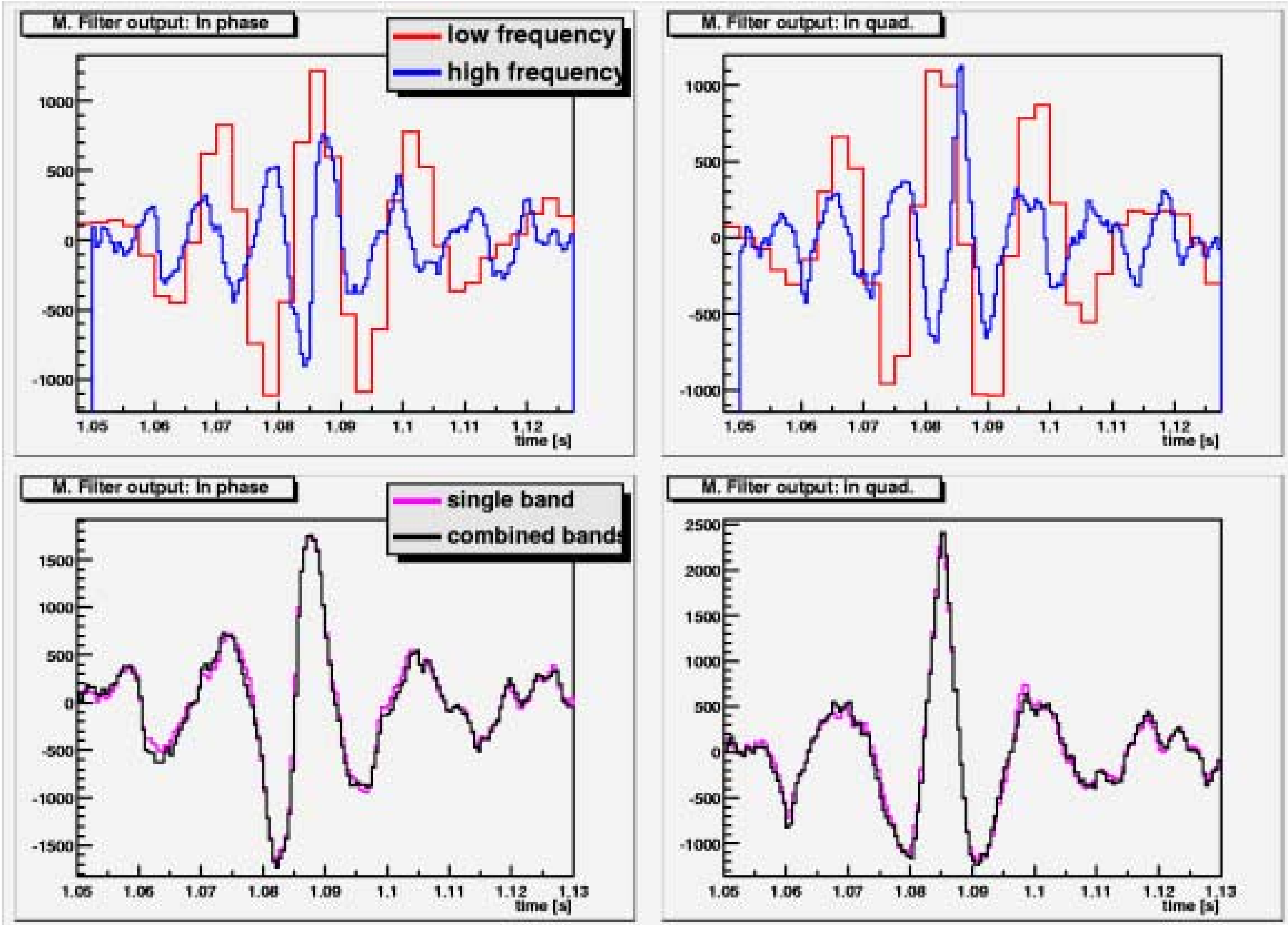
- If f_{min} = 20 Hz, f_{max} = 2kHz, M_{min} = 0.5M

	1 Band	2 Bands	3 Bands
CPU(Gflops)	200	4.3	1.3
Storage (Gbytes)	10000	100	43
T. size (Mbytes)	11	0.6	0.2

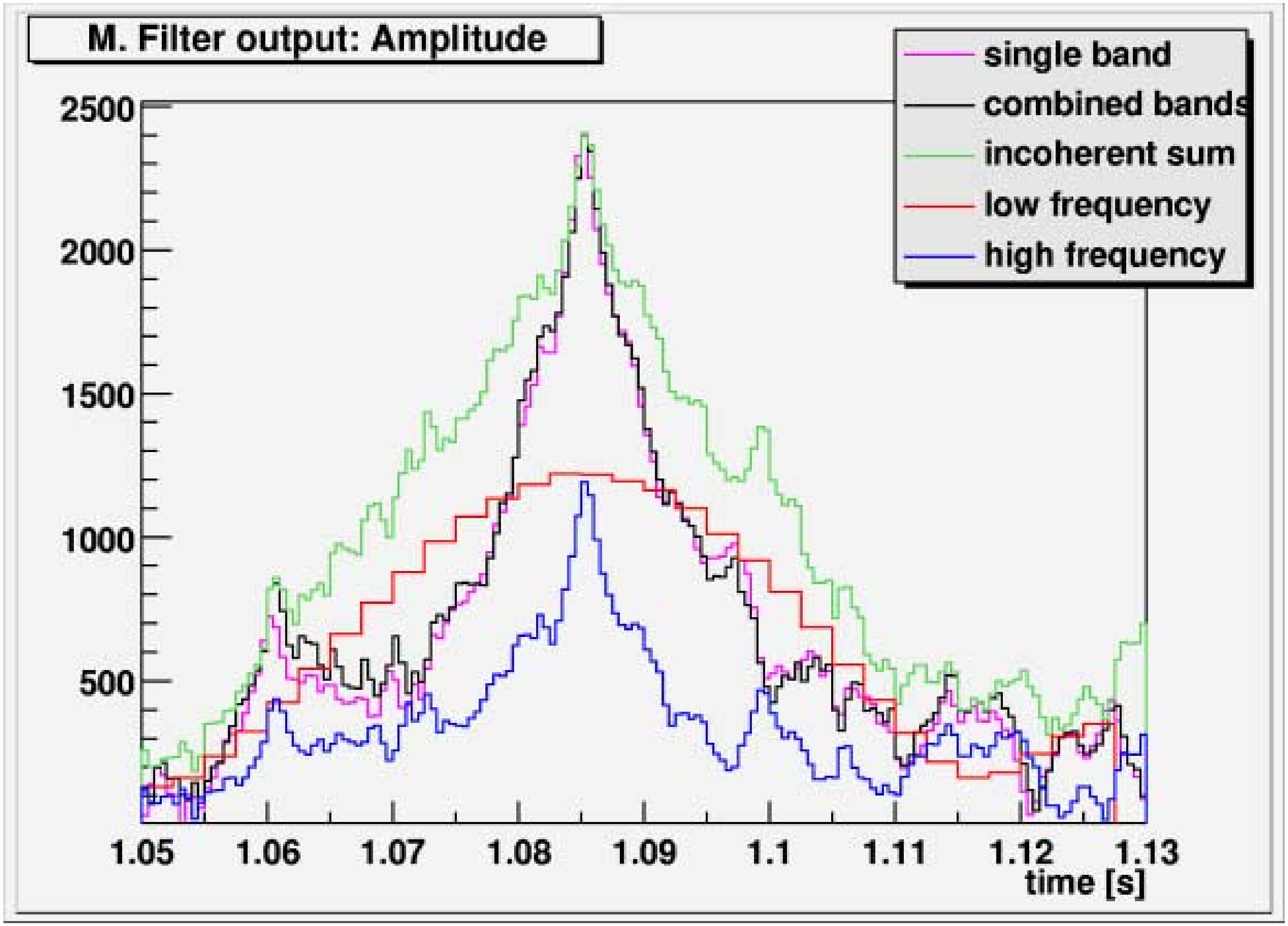
Does it works? Test with 2 bands



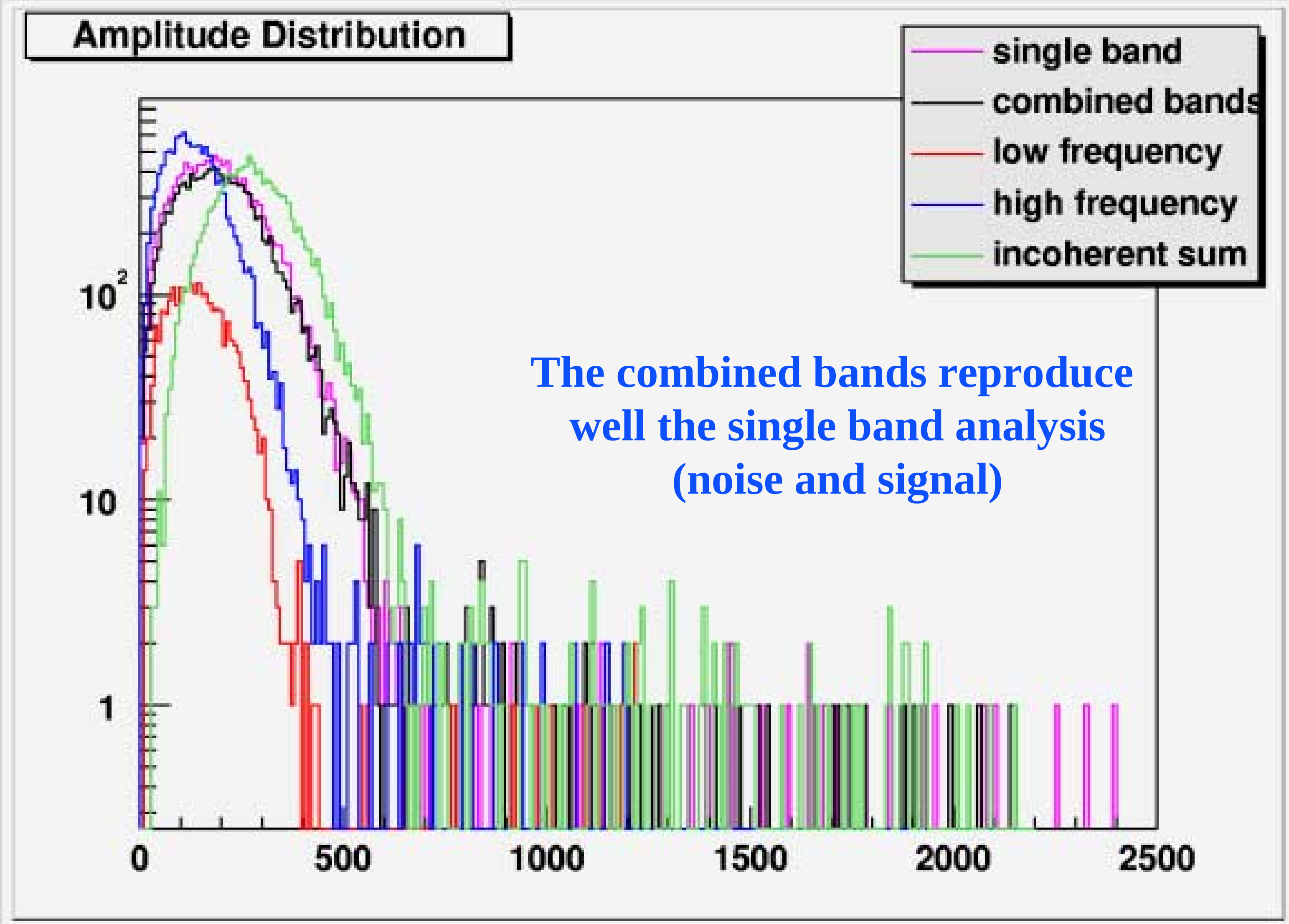
Zoom on each templates



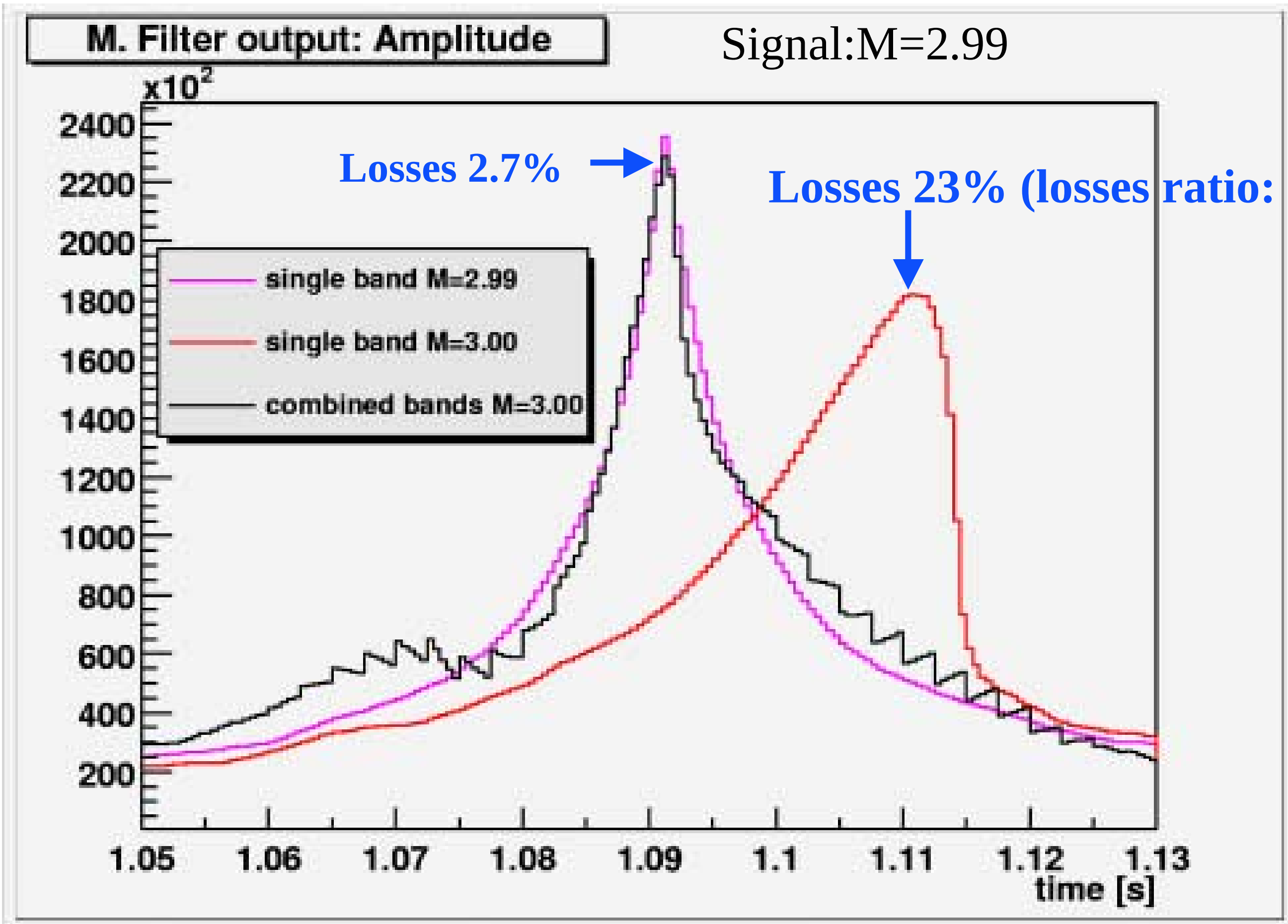
Comparison of the outputs



Comparison of the noises



If signal and template are mismatched



The parameter space is 8 time larger,

The templates are 5 times smaller

⇒reduction of ~10 for CPU, 40 for template storage

Summary

- Network: It is more efficient to do separate analysis and combine results
  - » It is possible to combine coherently the analysis results.
- The Multi Band Analysis has many advantages
  - » No SNR change
  - » Reduce the computing requirements
  - » Work on small FFT (fit in the CPU cache, use single precision)
  - » Build-in hierarchical approach without compromise on SNR
  - » Build-in consistency tests
- More study in progress
  - » Implementation problems? Is the gain as good as expected?
    $\Rightarrow$ Building a prototype code