Data Compression
“Standard” Approaches

GWDAW 2000

Benoit MOURS,

Caltech & LAPP-Annecy

Data Compression
Why should you care?

- Interferometers produce several Mbytes/s (~100Tb/y)
  - » Data Handling is complex.
  - » Archiving cost is important.
  - » ***Disk space is a limit in your analysis.***
- Data compression could improve
  - » ***Data access*** from the archive, network, nfs disks...
  - » Data lifetime
  - » Speed of your analysis program
- But data compression may change the data...

Data Compression
What could we do?

- Record the right data
  - » Right sampling frequency?
  - » Right electronic gain? (do not record too much electronic noise)
  - » Right format (integer or float)?

- Compress the data
  - » Without loss of information (like gzip).
  - » With *some* loss of information.
  - » By converting large vector to a few statistical information.
    Produce different kind of data sets (metadata)

Available Frame Compression (Format Spec. & I/O library)

- Only lossless compression method
- Compression done at the vector level
  - » No need to uncompress unused channels.
- Standard gzip
  - » Integer are differentiate to improve compression rate.
- Zero suppress
  - » Differentiate date are store with the minimal number of bit needed.
  - » Available only for integer.

Data Compression Performances

- Numbers from the November E2 Ligo run

	Full Frames	Reduced' Frames
Raw Size	3.2 Mb/s	1.5 MB/s
Size after gzip + Zero supp.	1.7Mb/s	1.0 Mb/s
Fraction of float	31%	64%
Compression ratio for short	2.16	1.98
Compression ratio for float	1.33	1.33
gzip speed (float)	2Mb/s	2Mb/s
Zero supress speed (short)	8Mb/s	8Mb/s

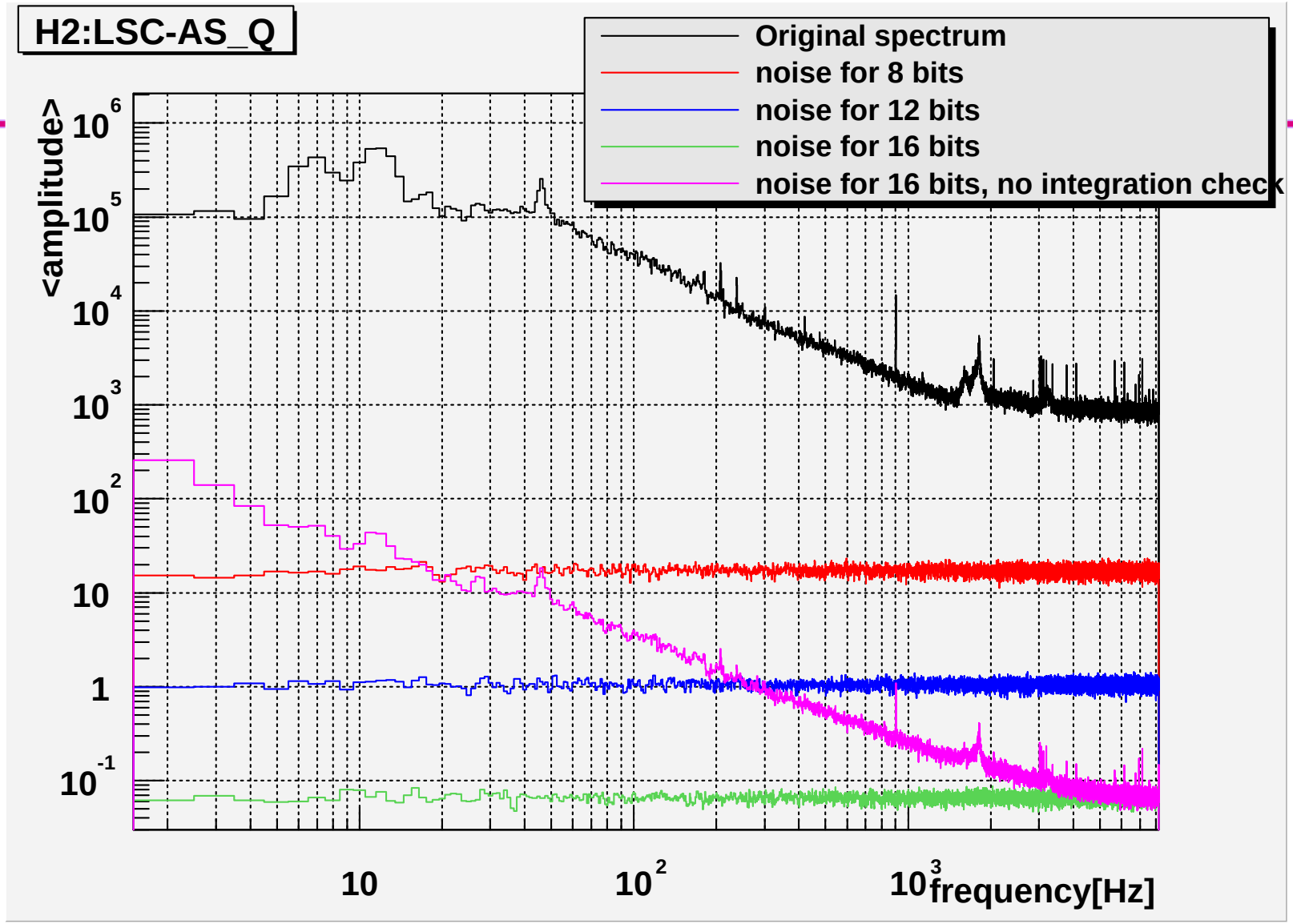
Speed measured on a Sun Ultra 10.

- Remarks:
  - » Poor speed and compression ratio for float
  - » People want floating points
  - » More floating points than originally foreseen

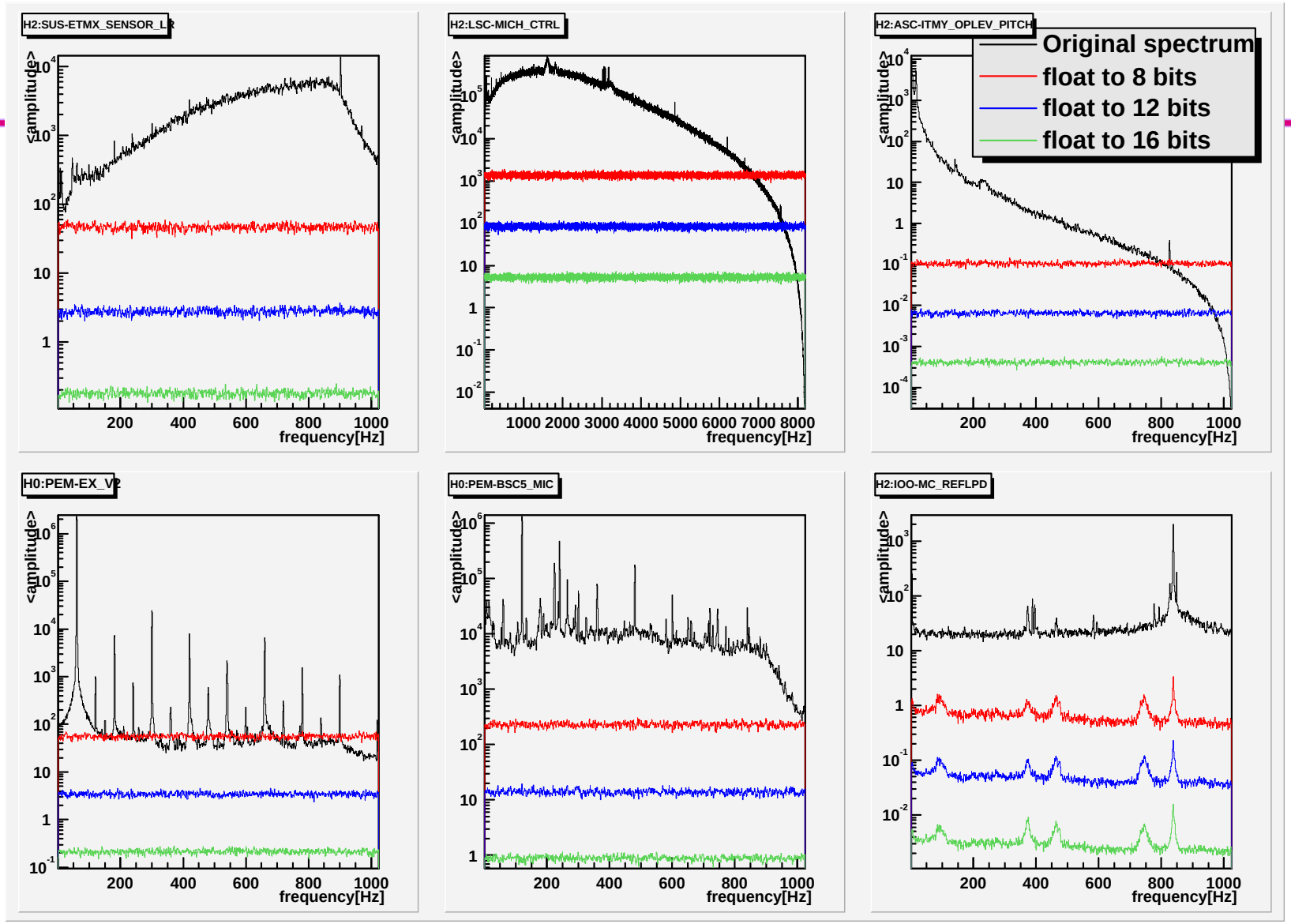
New Compression for float?

- Principle: convert float to integer
- Method:
  - » Differentiate the data and digitize the differences: $(s_{i+1} - s_i)/k$
  - » Round off is done by checking that the rebuild data do not diverge from the original date.
- Data saved
  - » First value, the differences convert to an integer, a scaling factor.
- One parameter: number of bits to store the integer
- Compression rate: 32/number of bits
  - » Example: If result stored on 16 bits: compression of 2
- Speed: Fast: 20 Mbytes/s (if result stored on 16 bits)

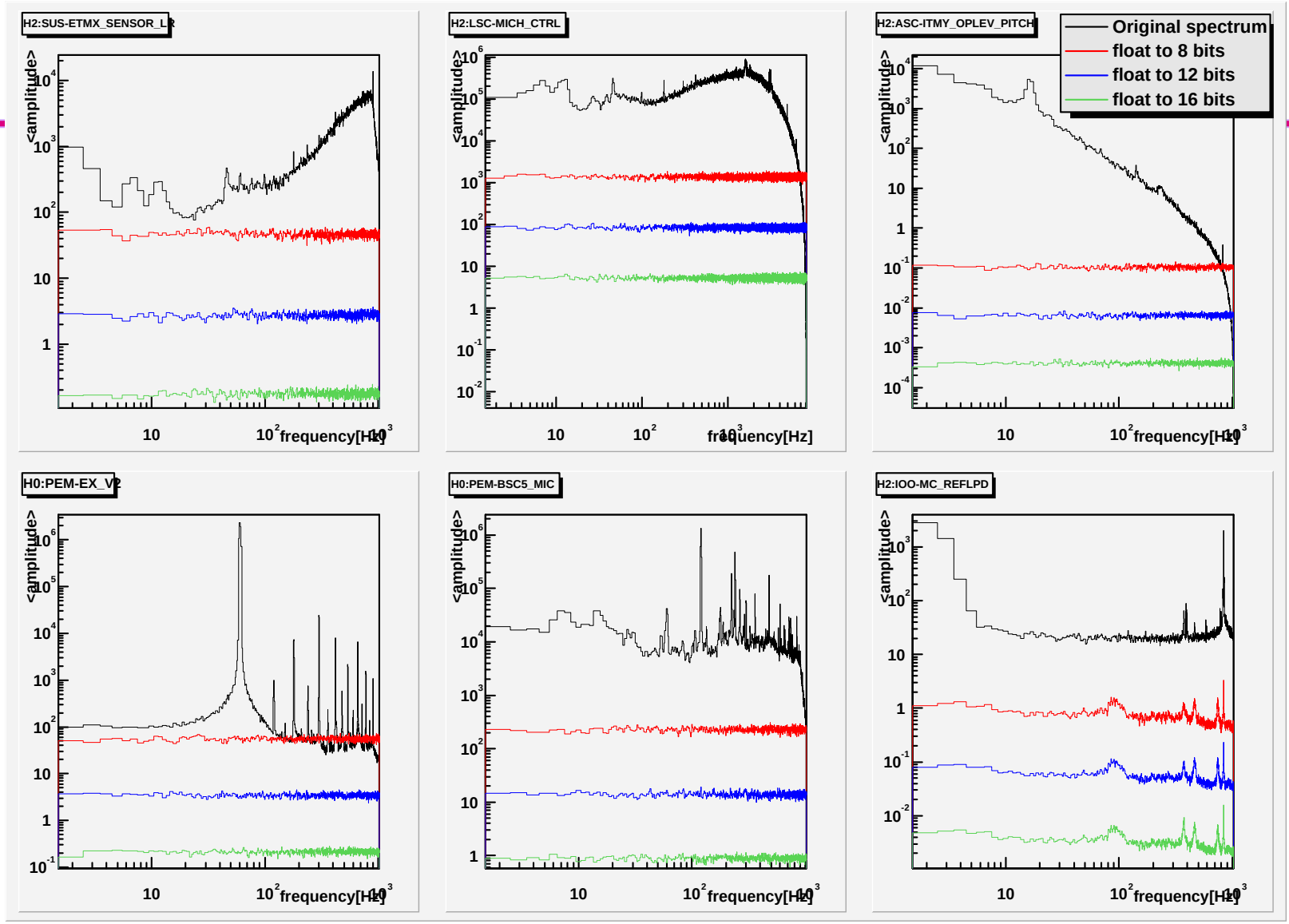
Compression for float: Noise



Noise for other channels



Noise for other channels



Summary

- The best ‘compression’ is to record only what you need:
  - » right channel, right frequency, right type (integer are better than float)
- Existing tools
  - » Work OK short integer.
  - » Poor on float.
- Simple lossy compression for float seems possible
  - » Small white noise introduced.
  - » Fast.
  - » Data could be stored in 8 to 16 bits.
  - » But not as good as if the data were stored as integer (8 bits enough).
- More aggressive compression? see S. Klimenko talk