SFTs: Scalable data-analysis framework for long-duration GW signals

Rodrigo Tenorio
rodrigo.tenorio@unimib.it

Università degli Studi di Milano-Bicocca

Ten Years to LISA
Jet Propulsion Laboratory
1-3 April 2025

Tenorio & Gerosa
arXiv:2502.11823

github.com/rodrigo-tenorio/sfts

UNIVERSITA' DEGLI STUDI DI MILANO BICOCCA

INFN Istituto Nazionale di Fisica Nucleare

erc

Why do we *need* new tools?

GW data analysis in the advanced detector (2G) era

Typical 2G signal: ~0.1s (BBH), ~1min (BNS).

Detectors can be assumed "static":

- Response functions are constant.
- Noise can be whitened using a PSD.
- There may be glitches, but no gaps.

Simple frequency-domain matched-filtering after "marginalising" extrinsic parameters (sky position, inclination, polarisation angle…):

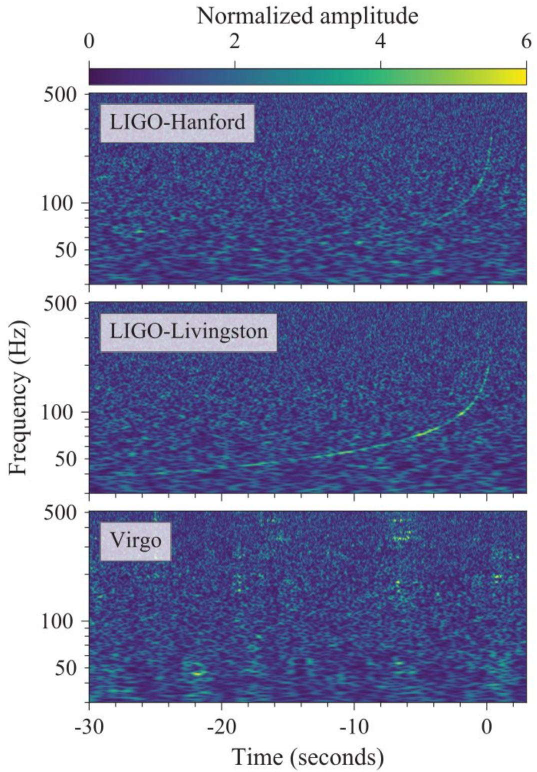
$$\max_{\text{distance},\varphi_0} \mathcal{L} = \frac{\langle x, h_c\rangle^2 + \langle x, h_s\rangle^2}{\langle h_c, h_c\rangle},$$

$$h_c = A(t)\cos\varphi(t)\,,$$

$$h_s = -A(t)\sin\varphi(t)\,.$$

$$\langle x, h\rangle = 4\,\mathrm{Re}\int_0^\infty \mathrm{d}f\,\frac{\tilde{x}^*(f)\tilde{h}(f)}{S_n(f)}$$

Finn PRD 1992
Allen+ PRD 2005



Abbott+ PRL 119 161101 (2017)

The challenge of next-generation detectors

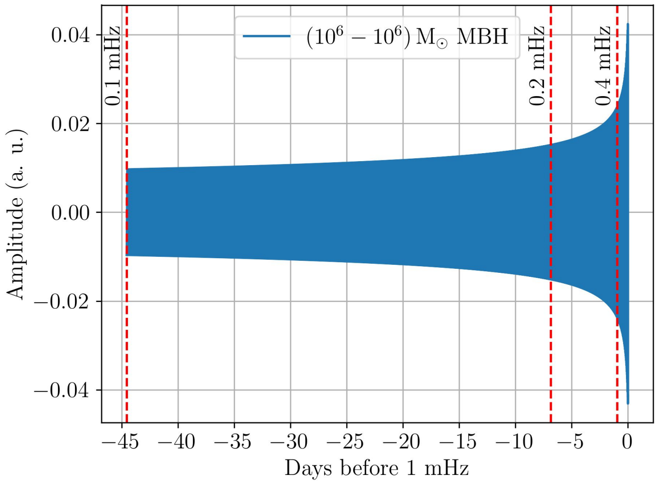
Future detectors will open up lower GW frequencies and have broader sensitivity bands.

Signals will last for longer → Higher computing cost!

Detectors no longer "static":

- Response functions depend on time (and frequency*).
- Noise is non-stationary (forget about *a* PSD).
- There may be glitches and gaps.

2G acceleration techniques (multibanding/heterodyning/rel. bin./ROQs) are not intended to solve these problems, as they are hard to address in the frequency domain [Hu & Veitch (2024), Chen & Johnson-McDaniel (2024), Burke+ (2025)]



The challenge of next-generation detectors

Future detectors will open up lower GW frequencies and have broader sensitivity bands.

Signals will last for longer → Higher computing cost!

Detectors no longer “static”:

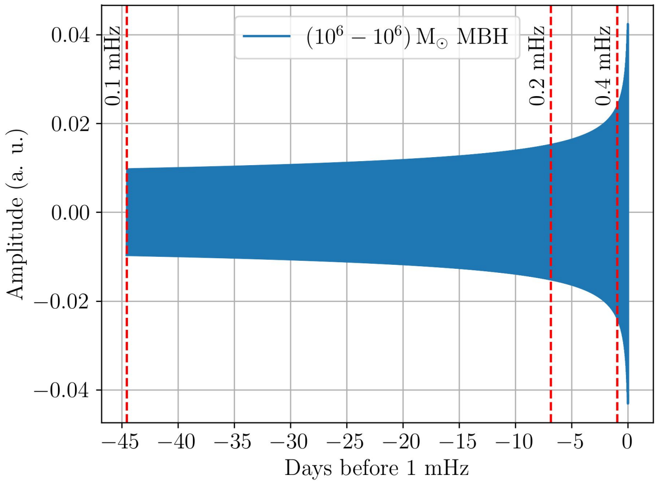
- Response functions depend on time (and frequency*).
- Noise is non-stationary (forget about *a* PSD).
- There may be glitches and gaps.

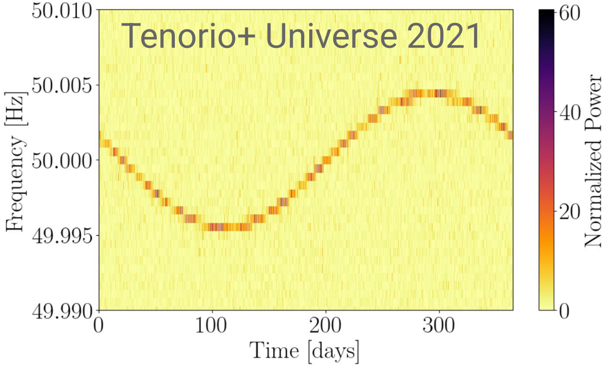
2G acceleration techniques (multibanding/heterodyning/rel. bin./ROQs) are not intended to solve these problems, as they are hard to address in the frequency domain [Hu & Veitch (2024), Chen & Johnson-McDaniel (2024), Burke+ (2025)]

Not a problem at all, however, for continuous gravitational-wave searches.

$$\max_{\text{distance},\varphi_0} \mathcal{L} = \frac{\langle x, h_c\rangle^2 + \langle x, h_s\rangle^2}{\langle h_c, h_c\rangle},$$
$$h_c = A(t)\cos\varphi(t)\,,$$
$$h_s = -A(t)\sin\varphi(t)\,.$$

$$h(t) = \sum_{\nu=0}^{3} c_\nu h_\nu(t)\,,$$
$$h_\nu(t) = A(t)\begin{bmatrix} a(t)\cos\varphi(t) \\ b(t)\cos\varphi(t) \\ a(t)\sin\varphi(t) \\ b(t)\sin\varphi(t) \end{bmatrix}$$





How to use Fourier Transforms

(efficiently)

The broad picture

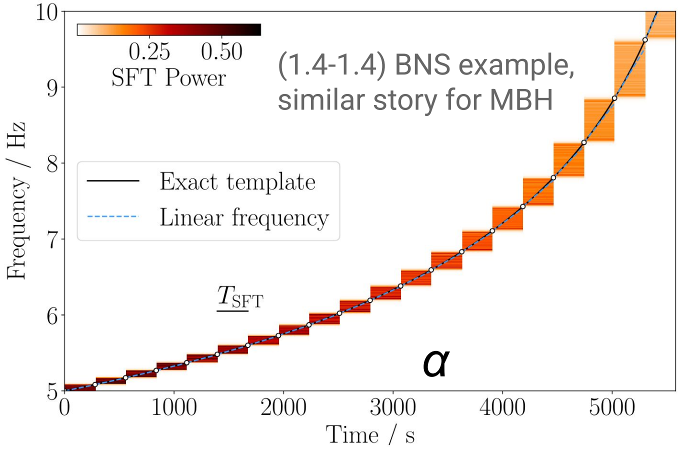
Follow a similar path to the *Demodulated* F-statistic

[Williams & Schutz AIP Conf. Proc. (2000), Jaranowski+ PRD (1998), Cutler & Schutz PRD (2005)]

1. Split the data in time-domain segments.
2. Approximate signal using a “simple” functions in each segment.
3. Compute the matched-filter in terms of segments.

Result:

- Signal-independent downsampling (up to x1000)
- Less mathOps (~ x10 reduction)
- Easier GPU parallelization (~ x100 speed up)



$$h_\alpha(t) = \begin{cases} A_\alpha \Lambda_\alpha e^{i[\varphi_\alpha + \Delta\varphi_\alpha(t)]} & \text{for} \quad t \in \mathcal{T}_\alpha \\ 0 & \text{for} \quad t \notin \mathcal{T}_\alpha \end{cases}$$

$$\varphi(t) \approx \varphi_\alpha + \Delta\varphi_\alpha(t)\,,$$

$$\Delta\varphi_\alpha(t) = 2\pi f_\alpha (t - t_\alpha) + \pi \dot{f}_\alpha (t - t_\alpha)^2\,,$$

$$([d, h]) \doteq \Delta f \sum_{\alpha=0}^{N_{\rm SFT}-1} A_\alpha \Lambda_\alpha e^{i\varphi_\alpha} \mathcal{C}(f_\alpha, \dot{f}_\alpha; \tilde{d}_\alpha)$$

Long signals are basically inspiral

Long-duration GWs are mostly *inspiral*.
Inspiral is "*narrow* in frequency" (see SPA).

Approximate inspiral as a linear chirp in each segment:

$$h(t) \approx \sum_{\alpha=0}^{N_{\mathrm{SFT}}-1} h_\alpha(t) \qquad h_\alpha(t) = \begin{cases} A_\alpha \Lambda_\alpha e^{i[\varphi_\alpha + \Delta\varphi_\alpha(t)]} & \text{for} \quad t \in \mathcal{T}_\alpha \\ 0 & \text{for} \quad t \notin \mathcal{T}_\alpha \end{cases}$$

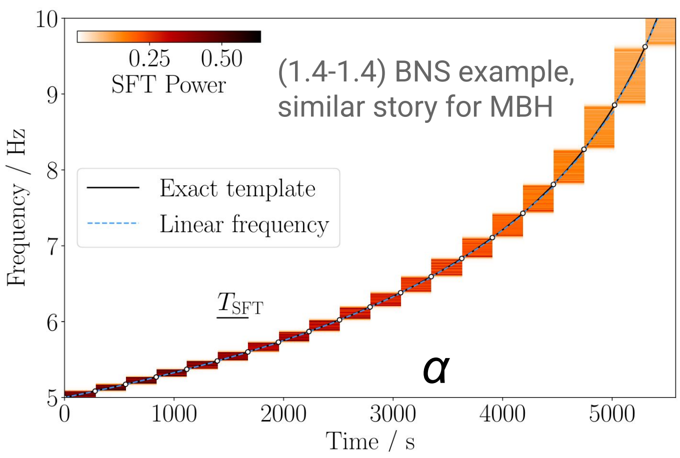
$$\varphi(t) \approx \varphi_\alpha + \Delta\varphi_\alpha(t)\,,$$
$$\Delta\varphi_\alpha(t) = 2\pi f_\alpha (t - t_\alpha) + \pi \dot{f}_\alpha (t - t_\alpha)^2\,,$$

Instead of N samples, each segment requires ~4 numbers to specify a waveform:

$$\{A_\alpha, \varphi_\alpha, f_\alpha, \dot{f}_\alpha\}$$

Detector/extrinsic parameters evaluated in time:

$$\Lambda_t = F_+(t)\mathcal{I}_+ \mathrm{Re} + F_\times(t)\mathcal{I}_\times \mathrm{Im}$$



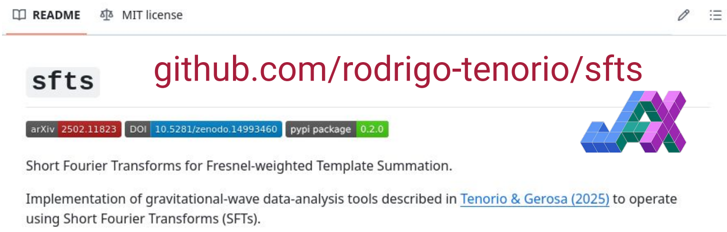
What about beyond inspiral?

↓

Short, should not have these problems

Frequency and derivatives can be readily computed for PhenomT
[Estellés+ PRD (2021, 2022, 2022)]

github.com/rodrigo-tenorio/sfts



Segment-wise matched-filter ~ Tracks on a complex spectrogram

$$\langle x, h \rangle = 4\,\mathrm{Re} \int_0^\infty \mathrm{d}f\, \frac{\tilde{x}^*(f)\tilde{h}(f)}{S_\mathrm{n}(f)}$$

Segment-wise matched-filter ~ Tracks on a complex spectrogram

$$\cancel{\langle x, h \rangle = 4\,\mathrm{Re} \int_0^{\infty} \mathrm{d}f \frac{\tilde{x}^*(f)\tilde{h}(f)}{S_{\mathrm{n}}(f)}}$$

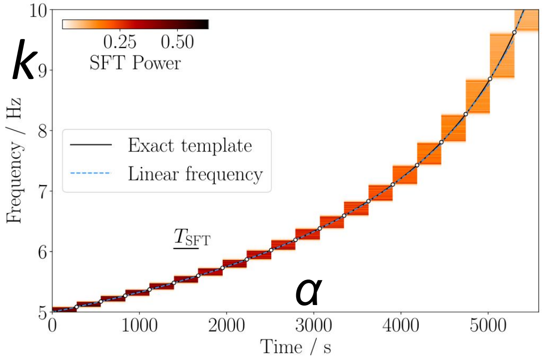
$$\boxed{([d, h]) \doteq \Delta f \sum_{\alpha=0}^{N_{\mathrm{SFT}}-1} A_\alpha \Lambda_\alpha e^{i\varphi_\alpha} \mathcal{C}(f_\alpha, \dot{f}_\alpha; \tilde{d}_\alpha)}$$

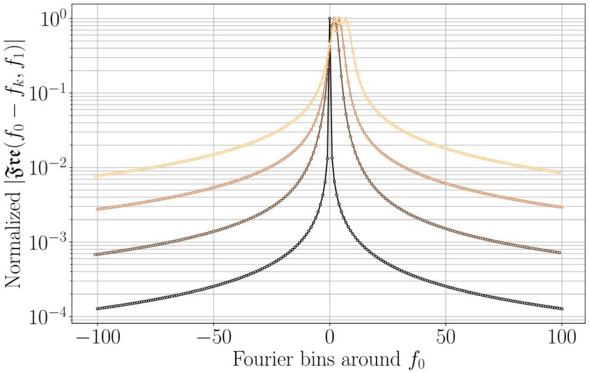
$$\mathcal{C}(f_\alpha, \dot{f}_\alpha; \tilde{d}_\alpha) \doteq \left\langle d_\alpha, e^{i\Delta\varphi_\alpha} \right\rangle$$
$$= \Delta f \sum_{k=k_{\min}}^{k_{\max}} \tilde{d}_\alpha^*[k]\, \mathfrak{Fre}(f_\alpha - f_k, \dot{f}_\alpha)$$

SFTs!

$$\tilde{d}_\alpha[k] = \Delta t \sum_{j=0}^{n_{\mathrm{SFT}}-1} d_\alpha[j] e^{-i 2\pi \tau_j f_k}$$

$$\mathfrak{Fre}(f_0, f_1) = \Delta t \sum_{j=0}^{n_{\mathrm{SFT}}-1} e^{i(2\pi f_0 \tau_j + \pi f_1 \tau_j^2)}$$





Advantages:

- Waveform & detector quantities are evaluated in “time domain”.
- Per-segment analysis: Transparent to glitches, easy to whiten.
- Filters can be truncated following the kernel → Compute less!

[Allen+ PRD (2002)]

Note we didn’t *choose* SFTs, they simply *arose* from the signal model [vid. Bretthorst 1988].

The cost of using SFTs ([,])

$$\varphi(t) \approx \varphi_\alpha + \Delta\varphi_\alpha(t)\,,$$
$$\Delta\varphi_\alpha(t) = 2\pi f_\alpha (t - t_\alpha) + \pi \dot{f}_\alpha (t - t_\alpha)^2\,,$$

Frequency error must be lower than a fraction δ of a frequency bin

[Jaranowski+ PRD (1998), Krishnan+ PRD (2004)]

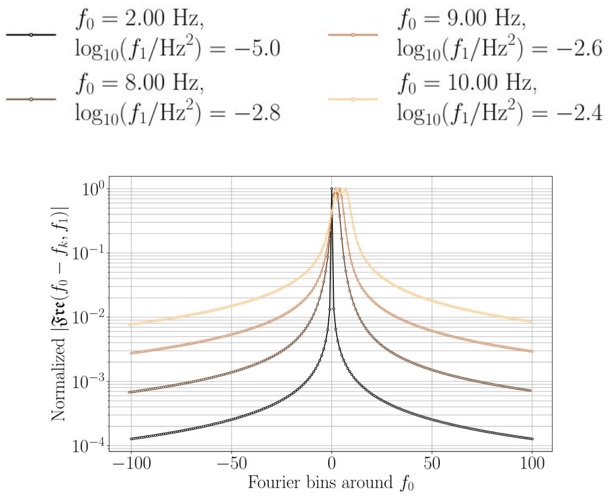
$$T_{\mathrm{SFT}}(\delta) = \left(\frac{2\delta}{\max_\alpha \ddot{f}_\alpha} \right)^{1/3}$$

We quantify relative error (units of *mismatch*):
Abot (1-5)% should be acceptable.

$$\boxed{r = 1 - \sqrt{\frac{([d,h])}{\langle d, h\rangle}}}$$

$$\mathcal{C}(f_\alpha, \dot{f}_\alpha; \tilde{d}_\alpha) \doteq \langle d_\alpha, e^{i\Delta\varphi_\alpha} \rangle$$
$$= \Delta f \sum_{k=k_{\min}}^{k_{\max}} \tilde{d}^*_\alpha[k]\, \mathfrak{Fre}(f_\alpha - f_k, \dot{f}_\alpha)$$

Take [kmin, kmax] so that Fre is low enough.
Depends on signal population.



Example: Early detection of MBH in LISA

$(10^6-10^6)\ M_\odot$ MBH from 0.1 mHz to 1mHz (~MECO).
Signal duration: ~1 month (10^6 samples @ 1Hz).

Long wavelength approximation with yearly modulation.

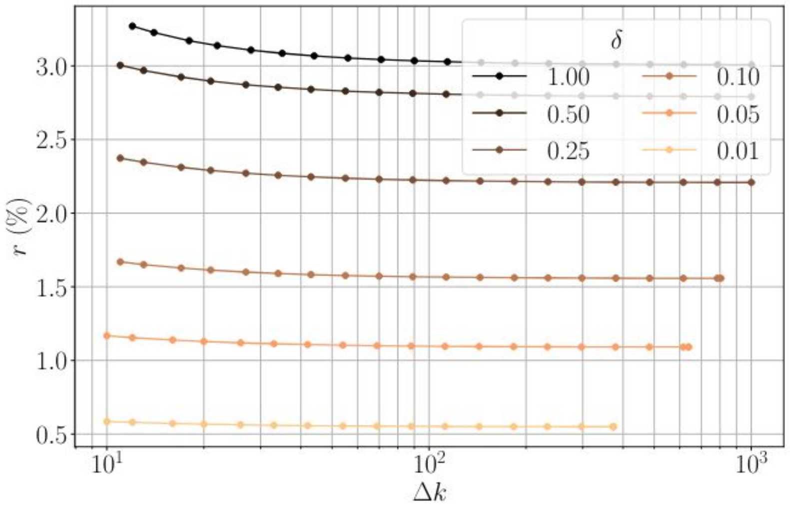
Accept ~1% mismatch:

- Tsft ~ 10 minutes → ~6 x 10^3 SFTs @ 0.001Hz
- kmax - kmin ~ 10 → 6 x 10^4 time-frequency bins.

Result:

- Smaller dataset (x150).
- Less mathOps (x20).
- “Immune” to gaps, detector response, non-stationary PSD.

The closest we’ll ever get to free lunch :)



δ	1.0	0.5	0.25	0.1	0.05	0.01
$T_{\mathrm{SFT}}/\mathrm{s}$	1735	1377	1092	805	639	373
N_{SFT}	2222	2799	3530	4789	6033	10336

And now, faster

Vectorised matched filtering

Matched-filtering cost: ~10ms for 1 waveform, 1 CPU
(Waveform generation) + (Data Gathering) + (mathOps)

Using SFTs, inspiral takes significantly less memory.
Can GPU batch-evaluate 1000 PhenomT waveforms.
Cost is negligible: ~ 0.001ms / waveform.

PhenomT is "closed-form", other waveforms may benefit from SFTs in other ways.

Dominant cost: (Data gathering) + (mathOps)
~0.01ms/waveform

LVK likelihood: ~10ms/waveform (?)

github.com/rodrigo-tenorio/sfts

README MIT license

sfts

Powered by

arXiv 2502.11823 DOI 10.5281/zenodo.14993460 pypi package 0.2.0

Short Fourier Transforms for Fresnel-weighted Template Summation.

Implementation of gravitational-wave data-analysis tools described in Tenorio & Gerosa (2025) to operate using Short Fourier Transforms (SFTs).

For the massive BBH case ($\delta = 0.05$, $\Delta k = 10$), $([d, h])$ for a single waveform on a CPU takes 0.01 s. The batch size on a CPU in this case can be increased up to 1000 waveforms, yielding an average cost of 3 ms per waveform. On a GPU with a batch size of 1000 waveforms the average computing cost drops to 0.01 ms per waveform.

Vectorised matched filtering

Matched-filtering cost: ~10ms for 1 waveform, 1 CPU
(Waveform generation) + (Data Gathering) + (mathOps)

Using SFTs, inspiral takes significantly less memory.
Can GPU batch-evaluate 1000 PhenomT waveforms.
Cost is negligible: ~ 0.001ms / waveform.

PhenomT is "closed-form", other waveforms may benefit from SFTs in other ways.

Dominant cost: (Data gathering) + (mathOps)
~0.01ms/waveform

LVK likelihood: ~10ms/waveform (?)

README MIT license

github.com/rodrigo-tenorio/sfts

sfts

Powered by

arXiv 2502.11823 DOI 10.5281/zenodo.14993460 pypi package 0.2.0

Short Fourier Transforms for Fresnel-weighted Template Summation.

Implementation of gravitational-wave data-analysis tools described in Tenorio & Gerosa (2025) to operate using Short Fourier Transforms (SFTs).

For the massive BBH case ($\delta = 0.05$, $\Delta k = 10$), $([d, h])$ for a single waveform on a CPU takes 0.01 s. The batch size on a CPU in this case can be increased up to 1000 waveforms, yielding an average cost of 3 ms per waveform. On a GPU with a batch size of 1000 waveforms the average computing cost drops to 0.01 ms per waveform.

tl;dr
SFTs are orders of magnitude faster than current LVK likelihoods,
and simpler to deal with for long-duration signals.

Conclusion

Conclusion

github.com/rodrigo-tenorio/sfts

rodrigo.tenorio@unimib.it – SFTs: Scalable analysis of long-duration GWs – Tenorio & Gerosa 2502.11823

Next-generation GW data-analysis will face a whole new suite of problems (non-stationary noise flors, time-dependent responses, gaps).
"2G methods but with longer data streams"-approach won't cut it.

We presented SFTs, a highly-efficient framework for long-duration GWs which solves all long-duration data problems in a simple manner.

Pilot application in early-warning for LISA (& 3G ground-based) demonstrate ~thousand-fold acceleration at negligible sensitivity impact.

WIP on real-world applications.

UNIVERSITA' DEGLI STUDI DI MILANO BICOCCA

INFN Istituto Nazionale di Fisica Nucleare

erc