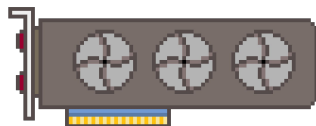


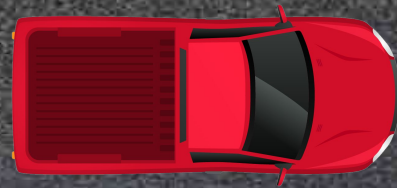
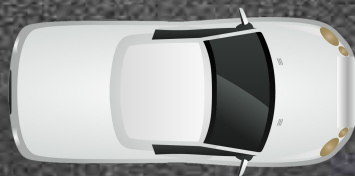
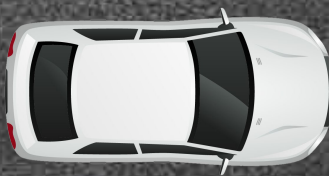
GPU-Based Gravitational Wave Inference

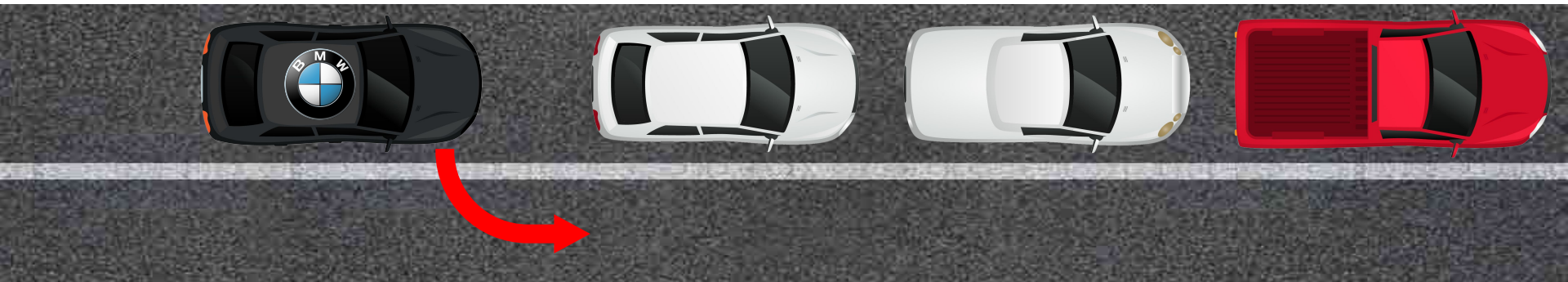


Robin Chan, on behalf of the GW-JAX Team

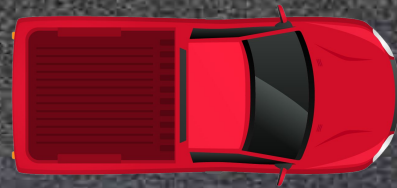
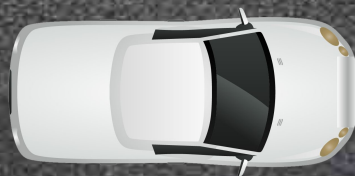
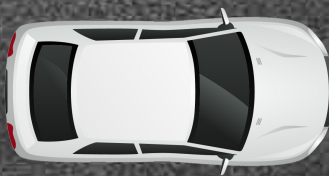


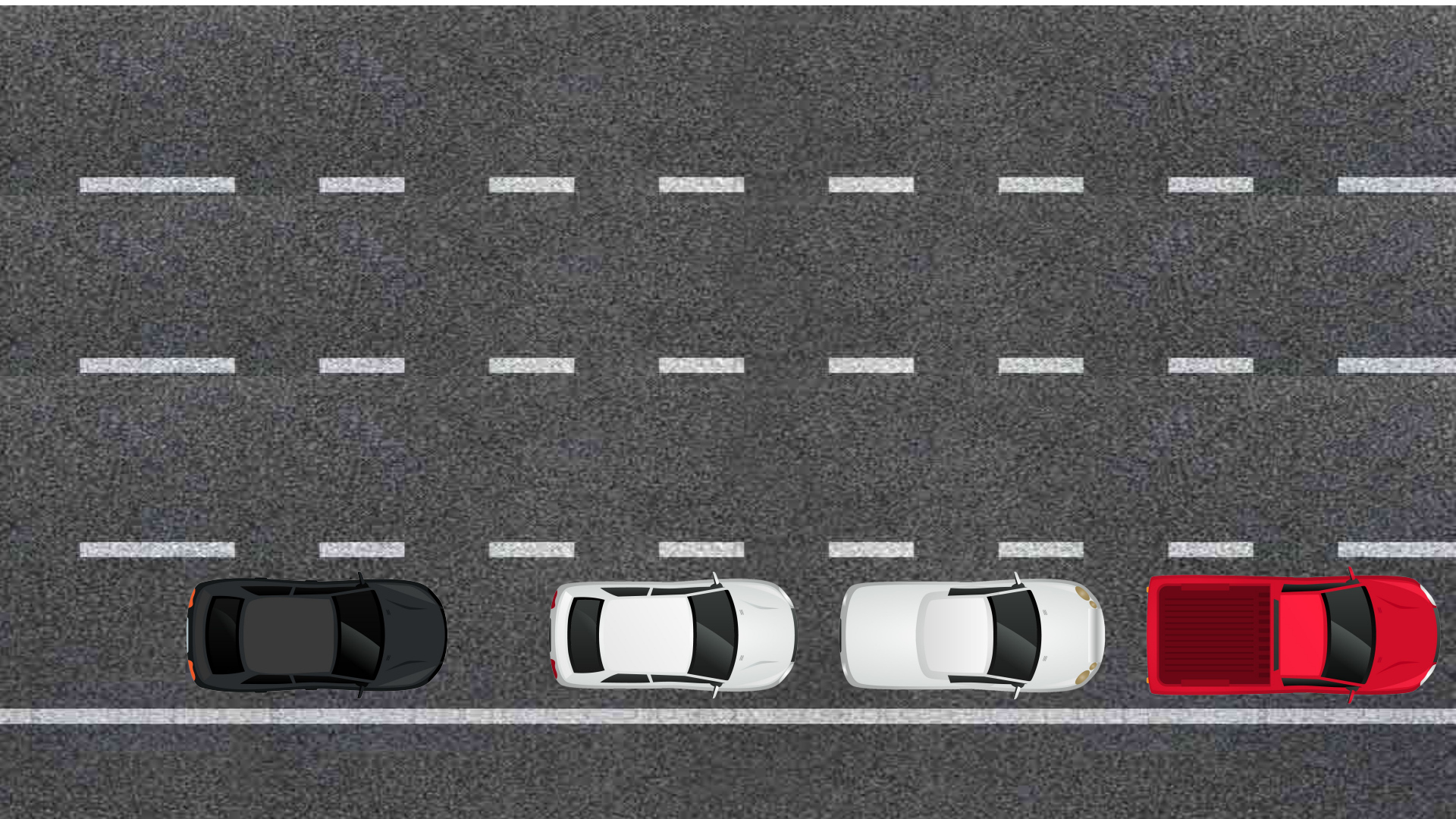
How can we make this faster?

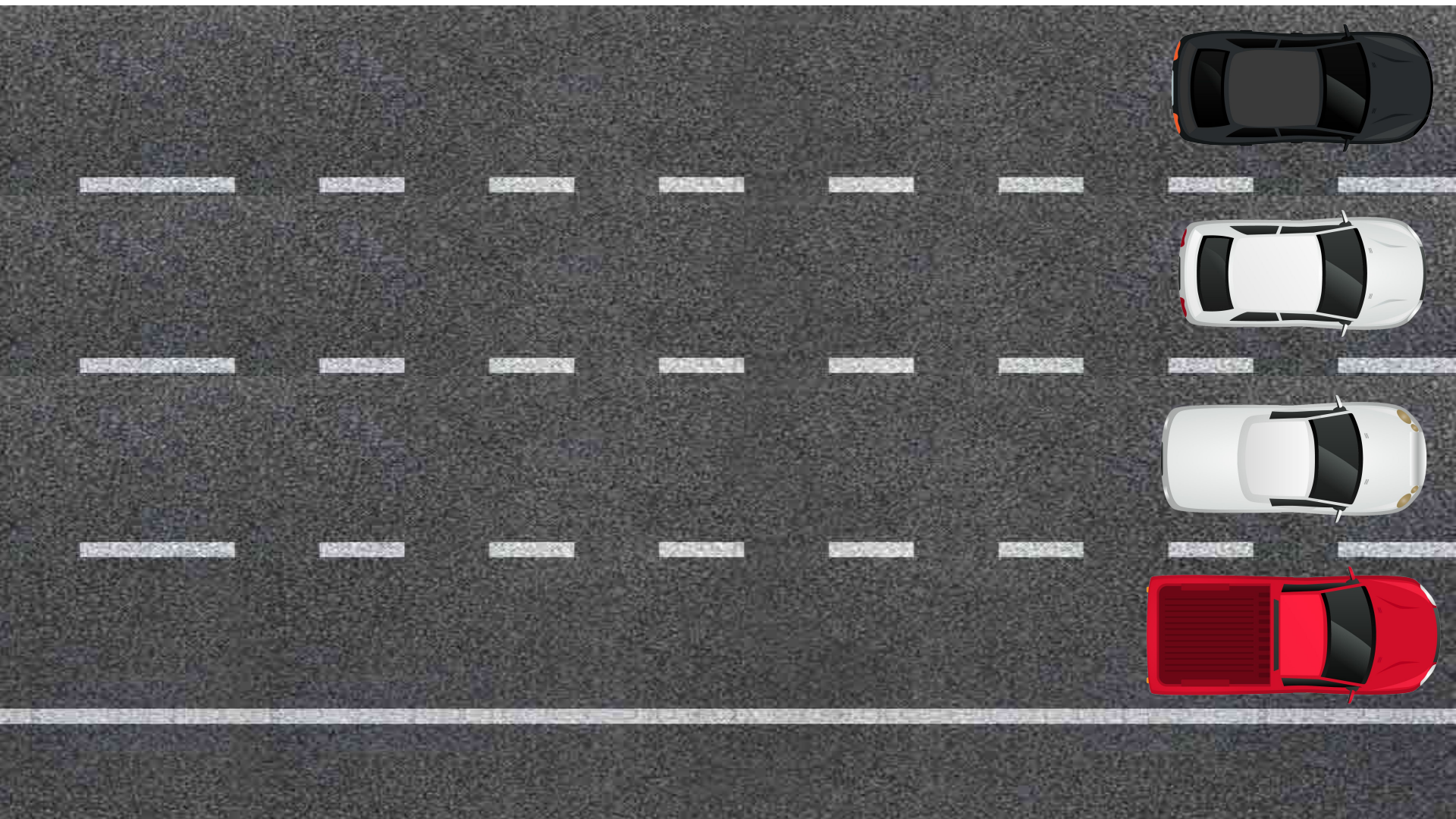




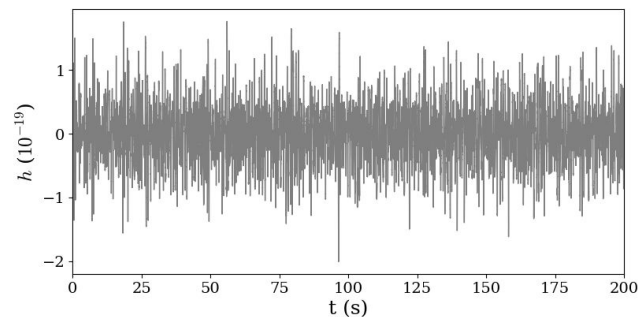
Just add more lanes!



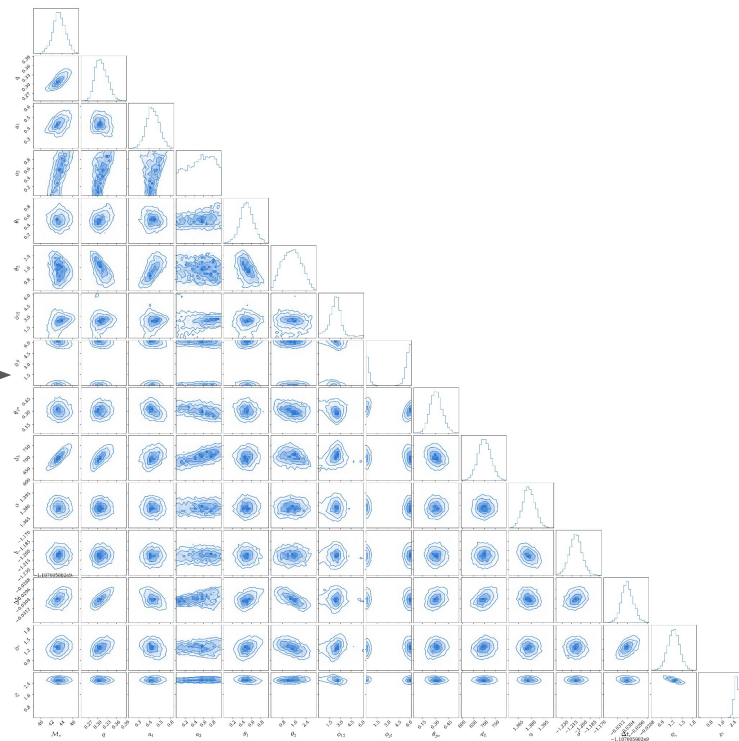




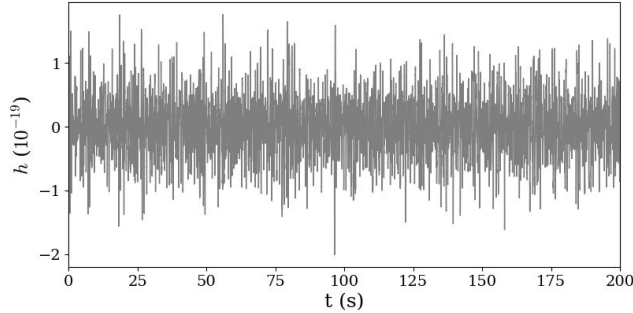
GW Data Analysis



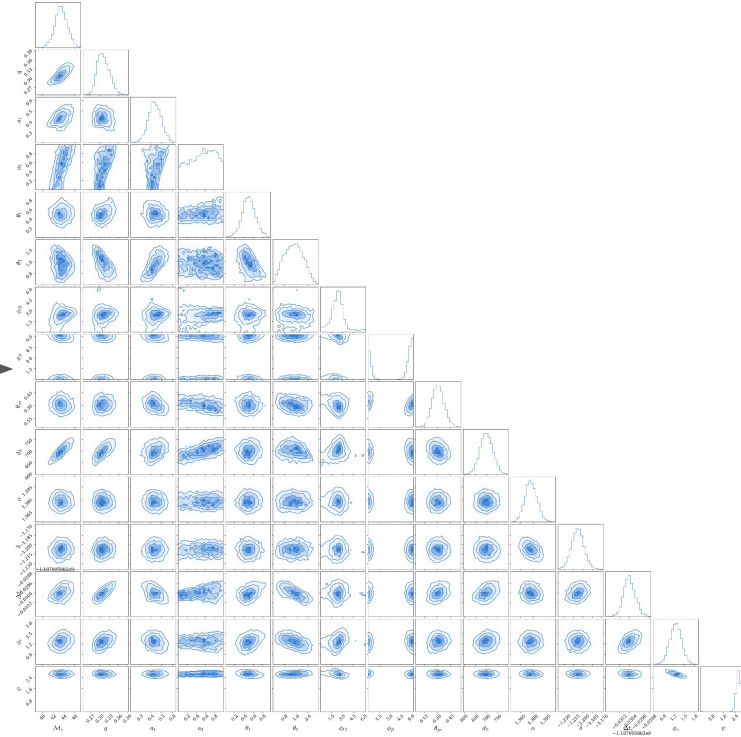
Parameter estimation



GW Data Analysis



Parameter
estimation

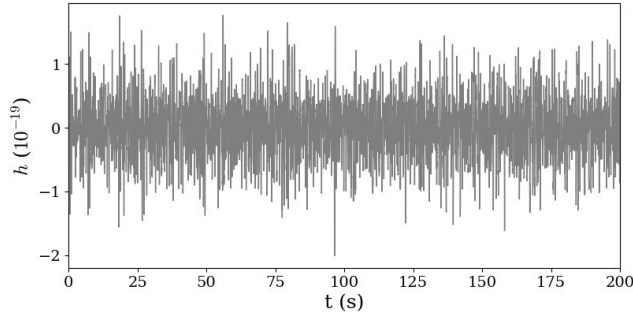


Bayesian inference

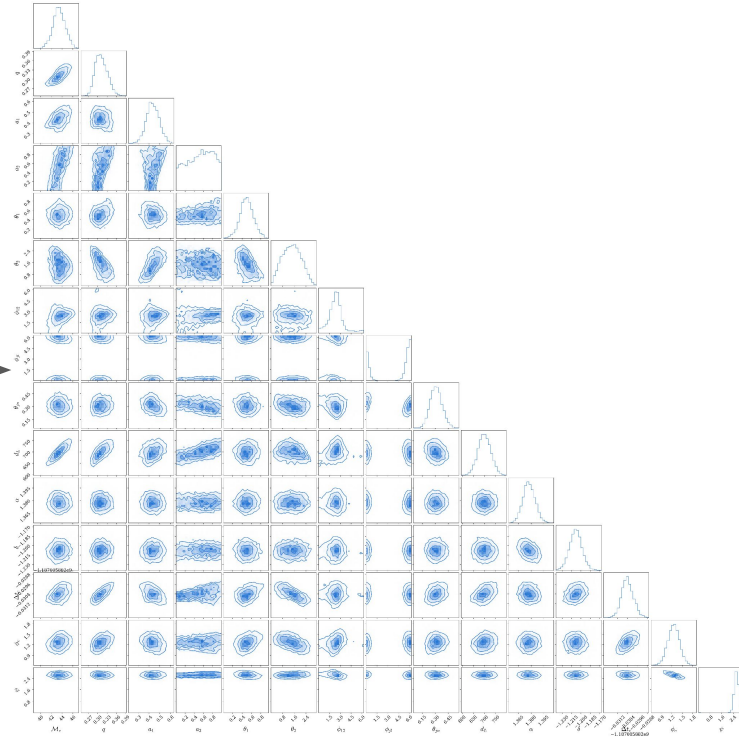
$$P(\theta|\mathcal{H}) = \frac{P(d|\theta)P(\theta)}{P(d|\mathcal{H})}$$



GW Data Analysis



Parameter
estimation

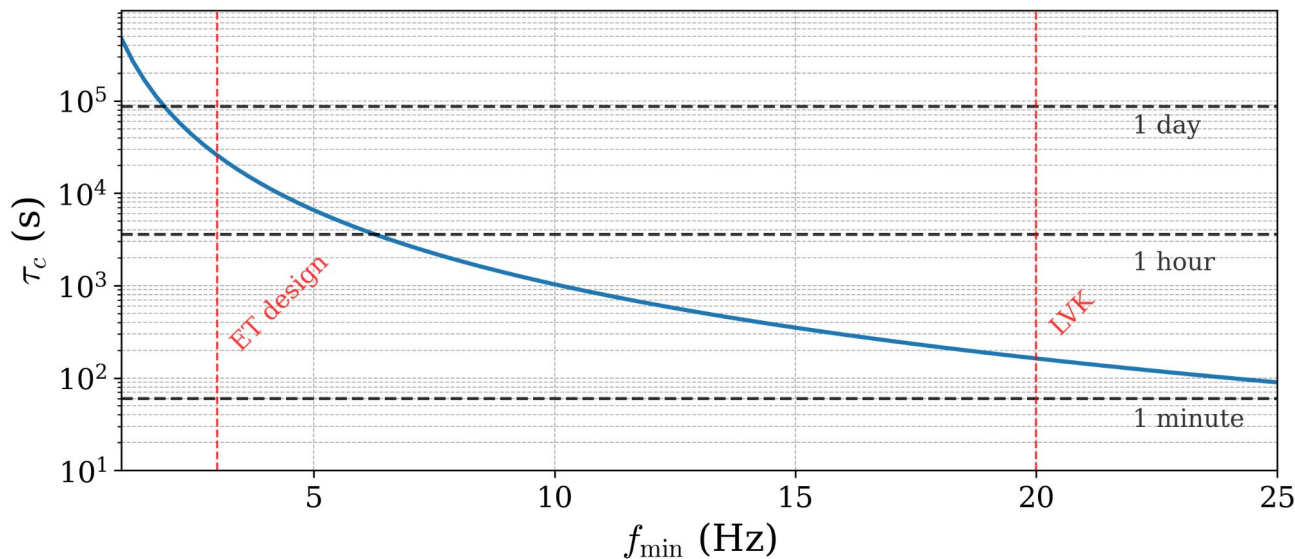


Bayesian inference
$$P(\theta|\mathcal{H}) = \frac{P(d|\theta)P(\theta)}{P(d|\mathcal{H})}$$

→ Requires models, sampling, noise generation, ...

Challenges for GW Data Analysis

Signal duration for GW170817:



GW170817

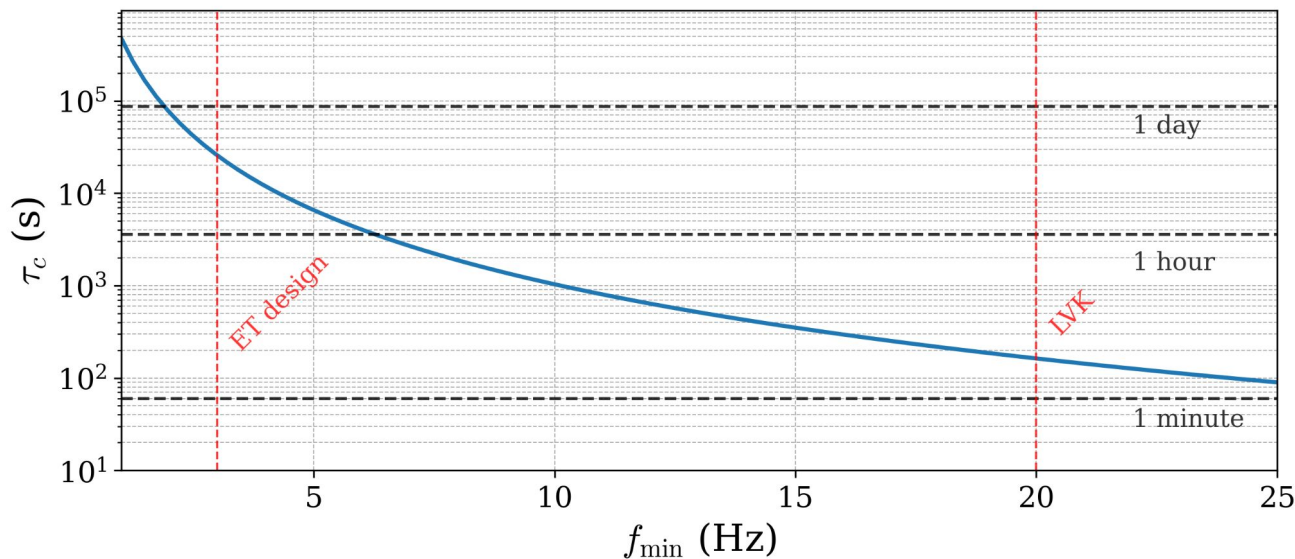
20 Hz

- ~500,000 frequency bins
- ~35 ms to generate waveform
- 1e6 waveforms for PE: almost 10 hrs generating waveforms



Challenges for GW Data Analysis

Signal duration for GW170817:



GW170817

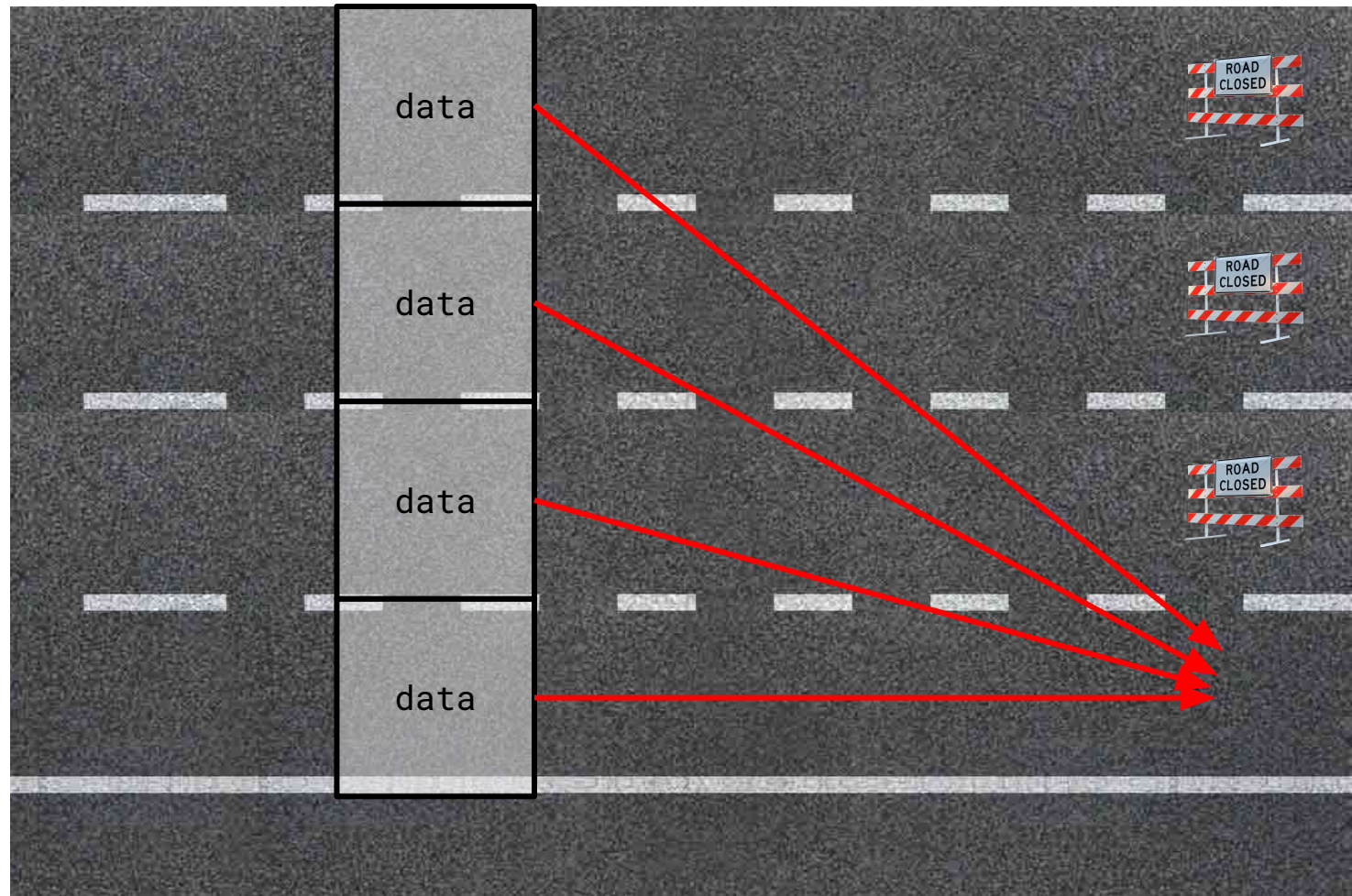
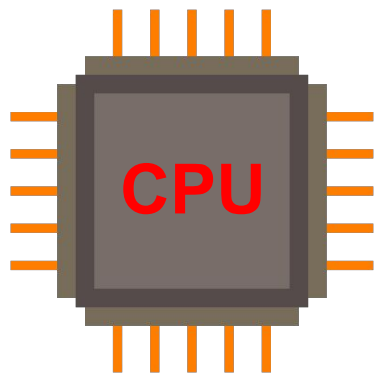
20 Hz

- ~500,000 frequency bins
- ~35 ms to generate waveform
- 1e6 waveforms for PE: almost 10 hrs generating waveforms

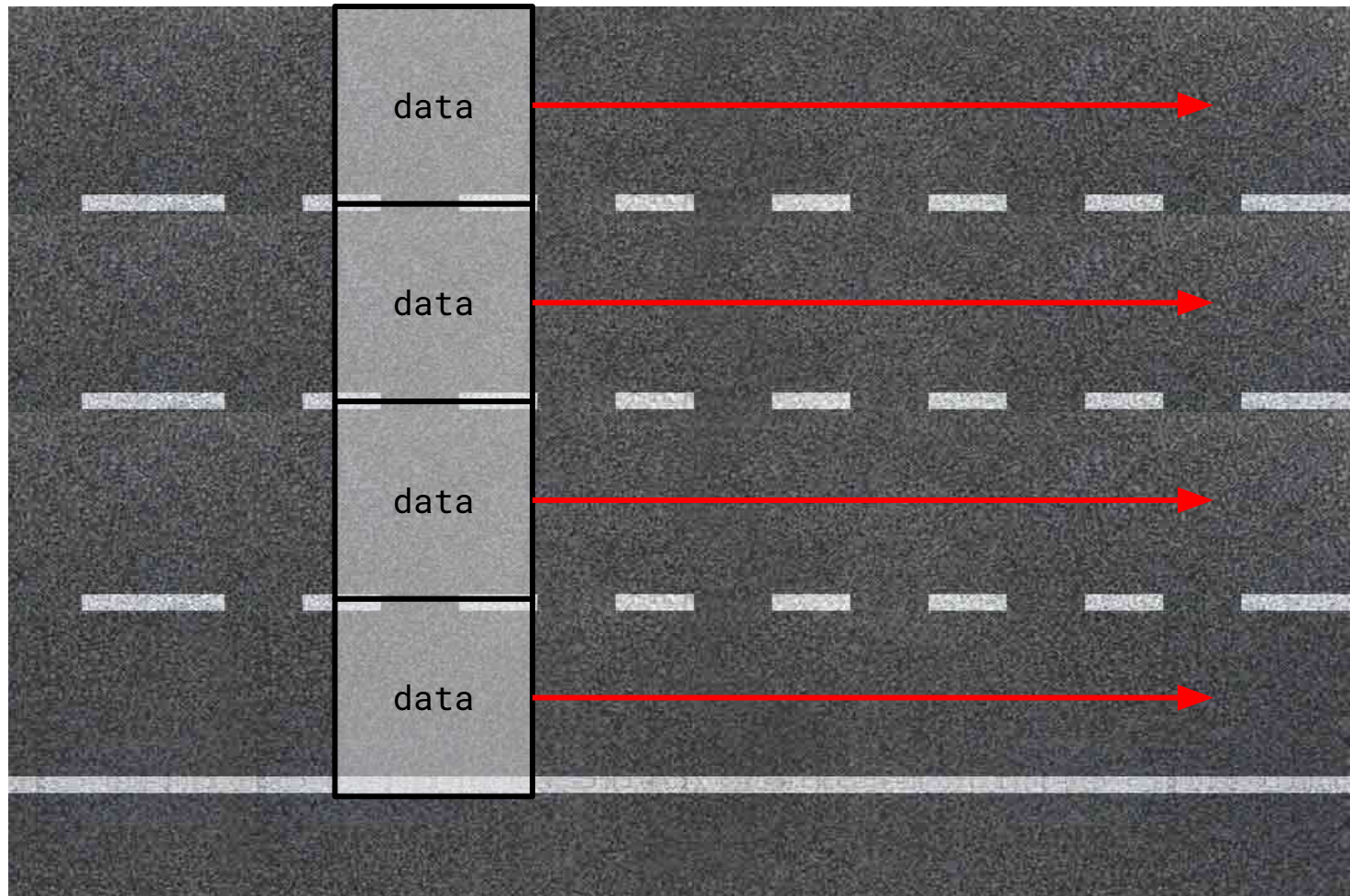
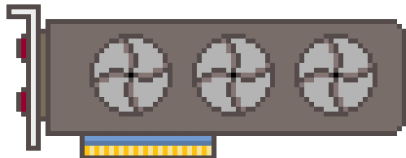
3 Hz

- ~67 million frequency bins
- ~4.4 s to generate waveform
- 1e6 waveforms for PE: almost 2 months generating waveforms





GPU





GPUs for GW Data Analysis

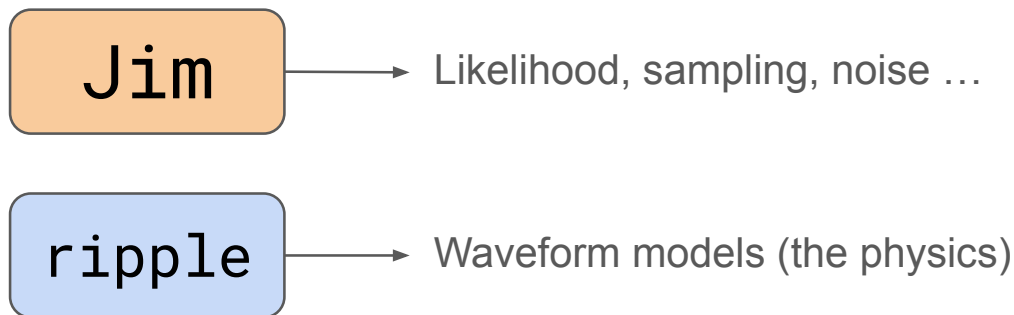
- JAX is a user-friendly Python package that enables GPU-compatibility
 - Just-in-time (JIT) compilation
 - Autodifferentiation
- Full code needs to be written in JAX to exploit these features





GPUs for GW Data Analysis

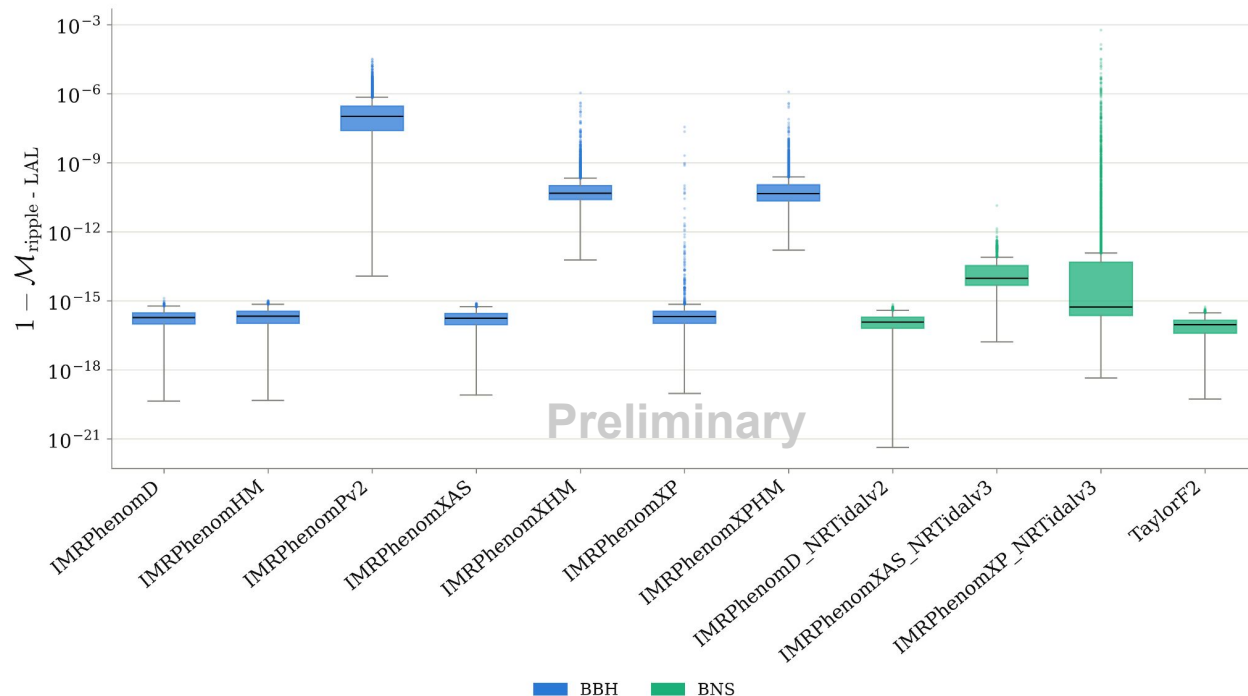
- JAX is a user-friendly Python package that enables GPU-compatibility
 - Just-in-time (JIT) compilation
 - Autodifferentiation
- Full code needs to be written in JAX to exploit these features



ripple

Fast & Accurate Waveforms

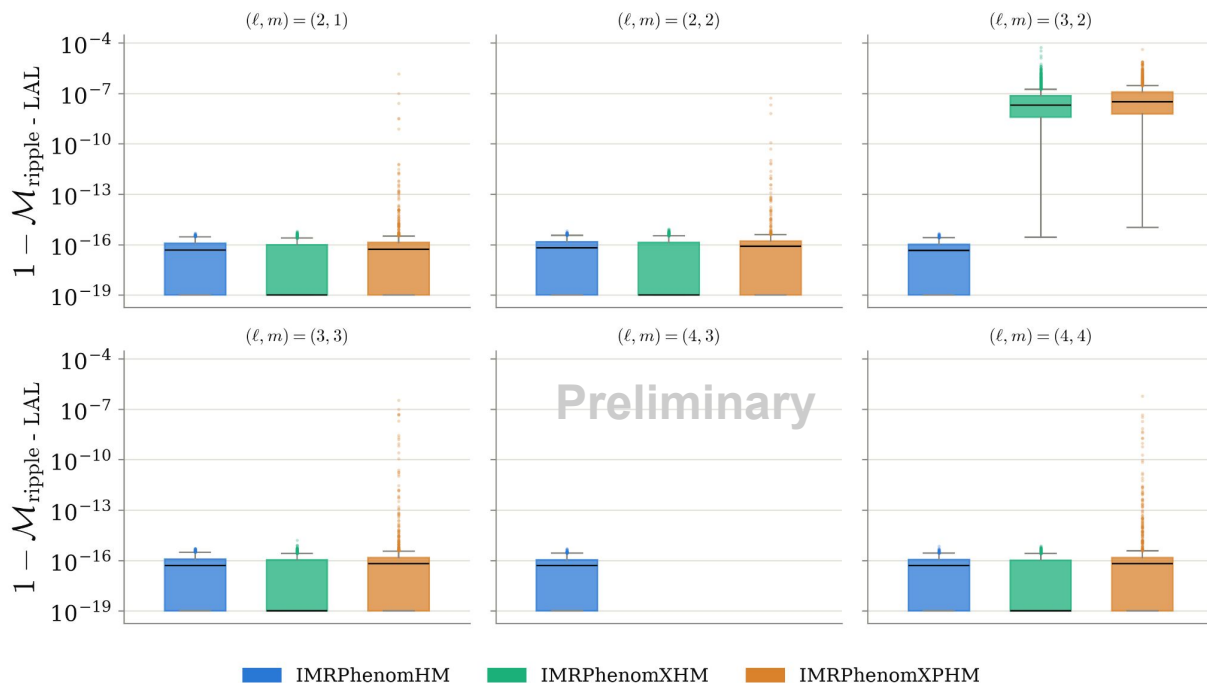
Accurate waveforms: validation against LAL



New hardware-accelerated and accurate gravitational waveform approximants for fast parameter estimation. Chan et al. (in prep)



Accurate waveforms: validation against LAL



New hardware-accelerated and accurate gravitational waveform approximants for fast parameter estimation. Chan et al. (in prep)



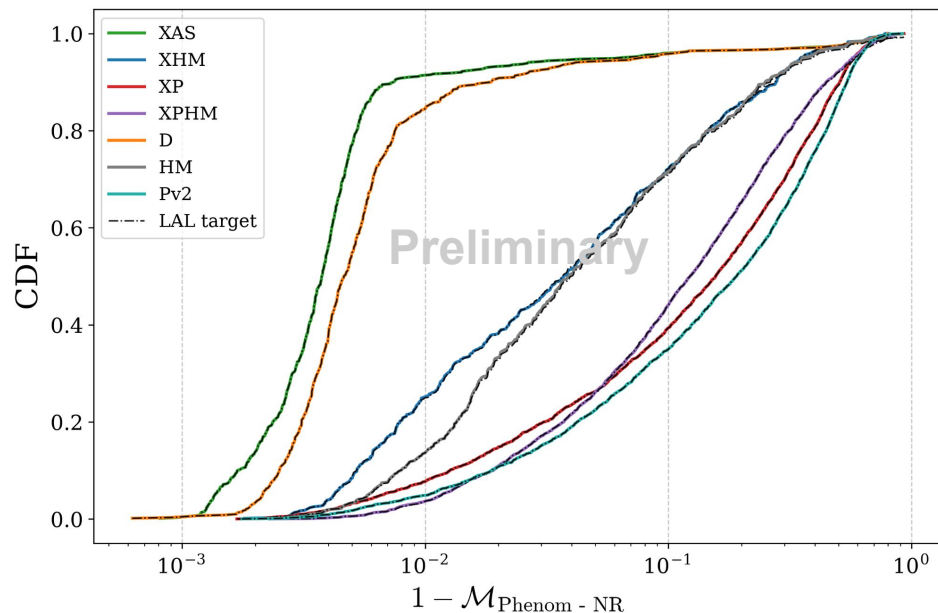
Accurate waveforms: consistency with NR

These phenomenological waveforms are calibrated to numerical relativity (NR) simulations

→ we should be consistent with LAL's mismatches against these

Validation with SXS BBH catalogue:

- Isolate relevant modes
- Compare relevant simulations:
 - Split into precessing and aligned-spin
- No hybrids
 - Higher MM than in literature



New hardware-accelerated and accurate gravitational waveform approximants for fast parameter estimation. Chan et al. (in prep)



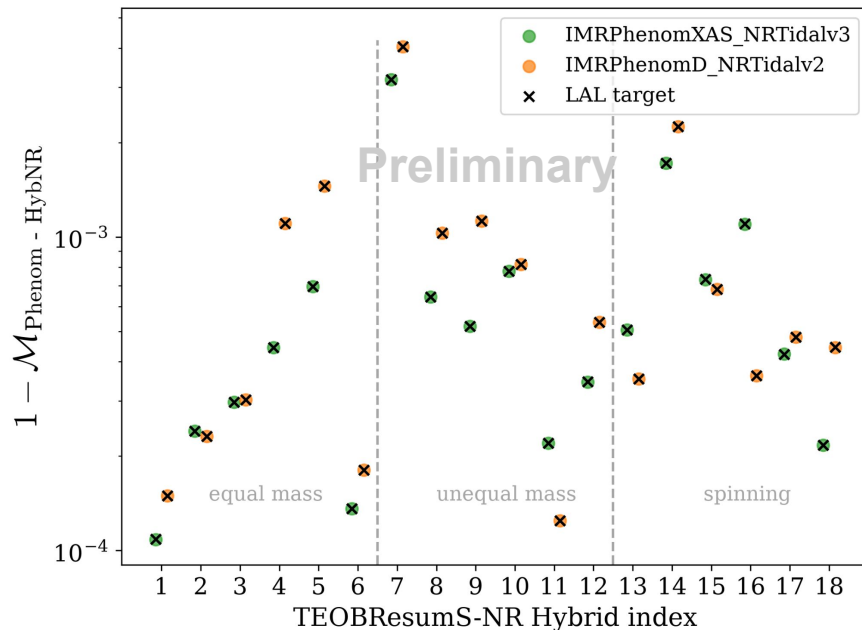
Accurate waveforms: consistency with NR

These phenomenological waveforms are calibrated to numerical relativity (NR) simulations

→ we should be consistent with LAL's mismatches against these

Validation with
TEOBResumS-NR BNS
hybrids:

- Include inspiral
- Consistent with literature

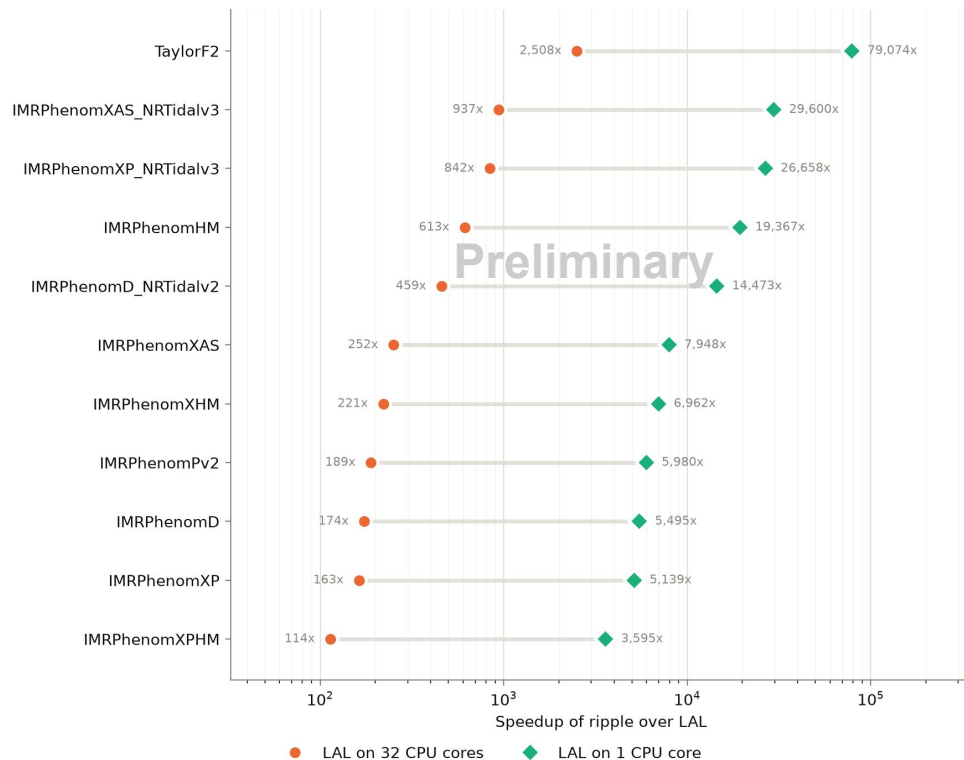


New hardware-accelerated and accurate gravitational waveform
approximants for fast parameter estimation. Chan et al. (in prep)



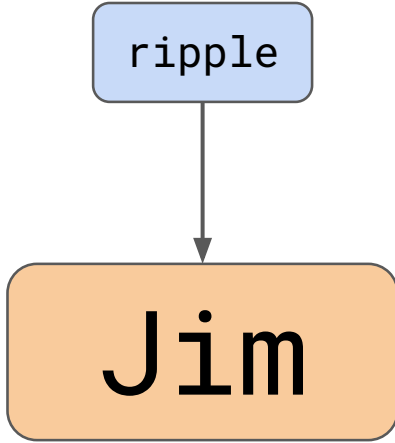
Fast waveforms

- Time to generate $1e6$ 4s waveforms
- NVIDIA A100 GPU



New hardware-accelerated and accurate gravitational waveform approximants for fast parameter estimation. Chan et al. (in prep)





Fast Parameter Estimation Without Compromises

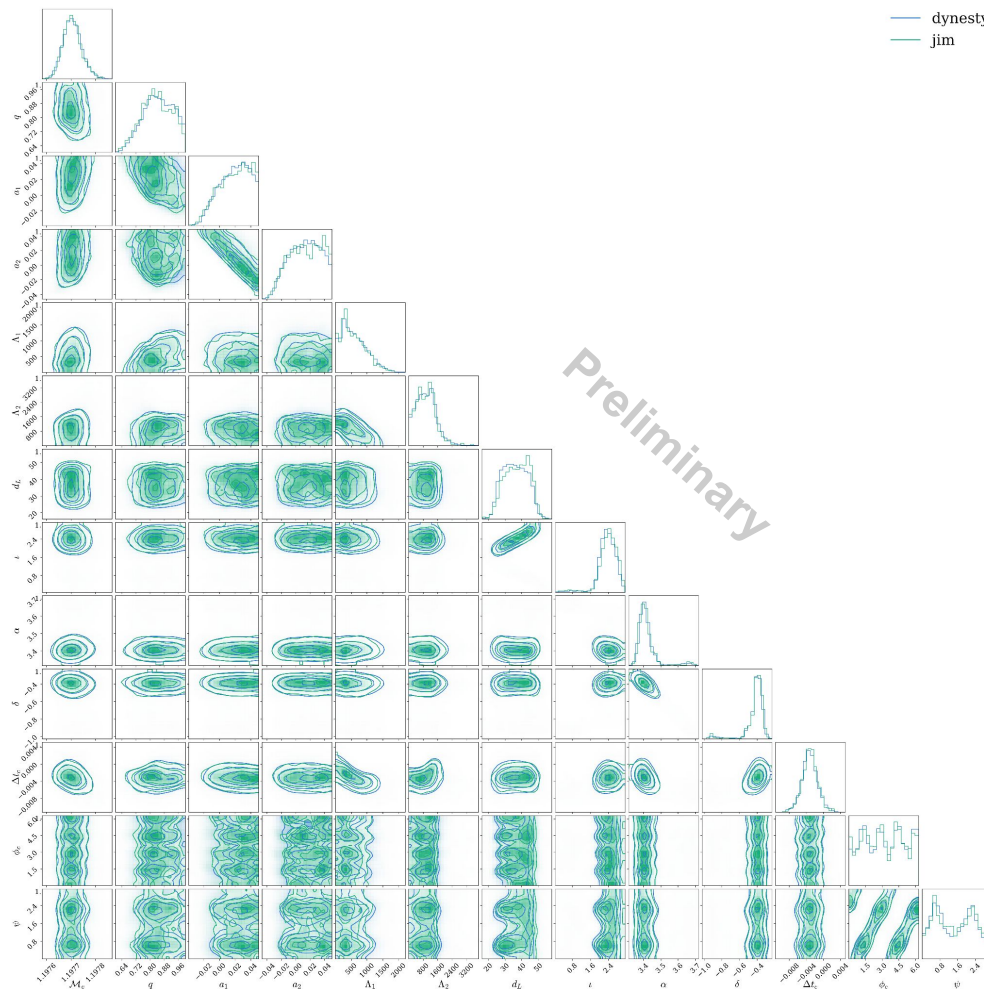
Fast Parameter Estimation

- Performs Bayesian inference
- Several samplers available
 - FlowMC
 - Blackjax samplers
 - NS AW
 - SMC
 - NSS
 - SwiG (see also [talk by Thibeau](#))
- Data compression
 - Multibanding
 - Relative binning

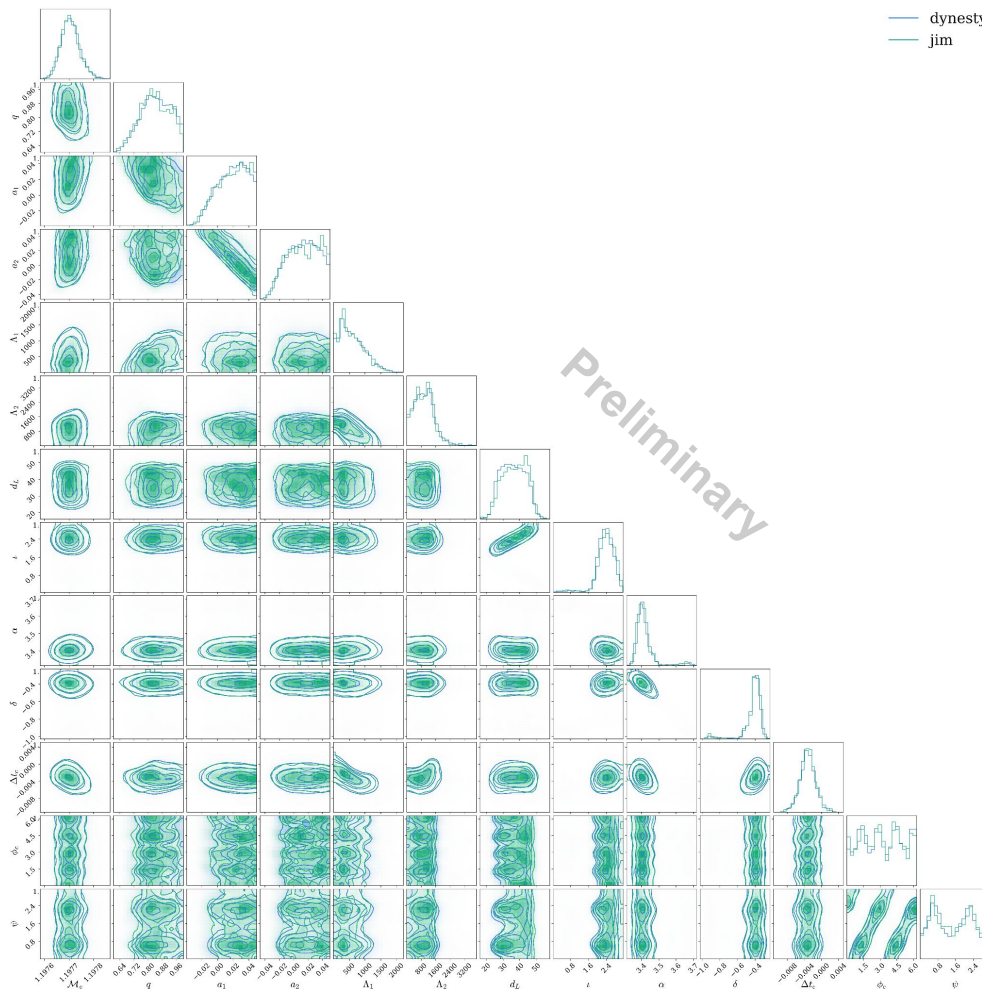


GW170817

- Full Bayesian PE
 - IMRPhenomXAS_NRTv3
 - Relative binning
- Bilby
 - 32 CPUs
 - dynesty
 - 2000 live points
 - ~15.5 hrs

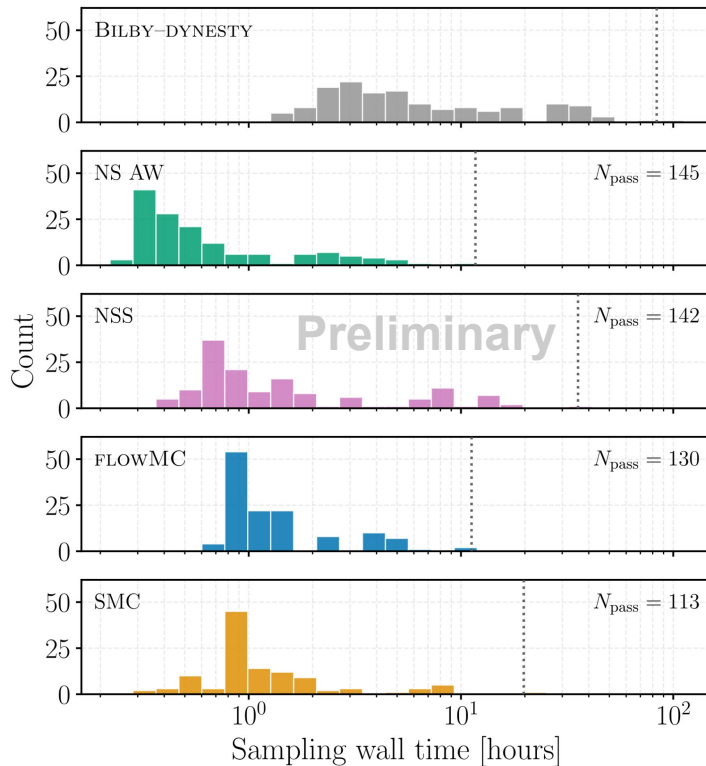


- Full Bayesian PE
 - IMRPhenomXAS_NRTv3
 - Relative binning
- Bilby
 - 32 CPUs
 - dynesty
 - 2000 live points
 - ~15.5 hrs
- Jim
 - A100 GPU
 - NS AW
 - 5000 live points
 - ~20 minutes



GWTC-4.0 BBH Catalogue

- Benchmark different samplers in Jim
- IMRPhenomXPHM
- 150 events



Hardware:

Bilby

- 24 CPUs

Jim

- H100 GPU



Conclusions & Outlook

- GPU-compatible framework available for GW analysis
 - Up to date with most recent waveform models
- Waveform generation→[ripple](#)
 - Consistent with
 - LAL, to machine precision
 - NR
 - Now with precessing, HM waveforms
- Parameter estimation→[Jim](#)
 - Consistent with bilby
 - Many new samplers



Conclusions & Outlook

- GPU-compatible framework available for GW analysis
 - Up to date with most recent waveform models
- Waveform generation→[ripple](#)
 - Consistent with
 - LAL, to machine precision
 - NR
 - Now with precessing, HM waveforms
- Parameter estimation→[Jim](#)
 - Consistent with bilby
 - Many new samplers

Does this mean PE is solved?



Still work
left!



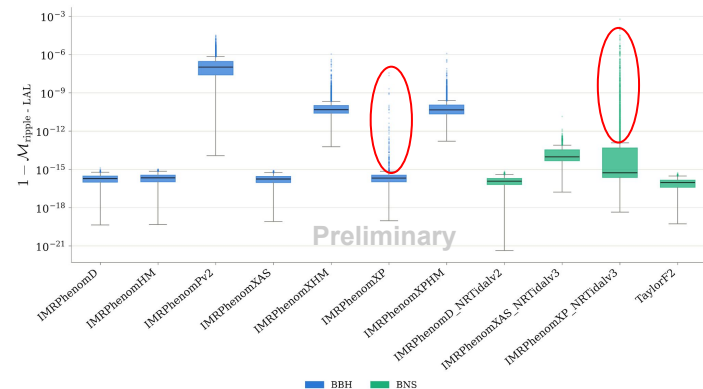
The annoying part
(porting code) is
done.

Now we need
clever algorithms
to use these new
lanes efficiently!

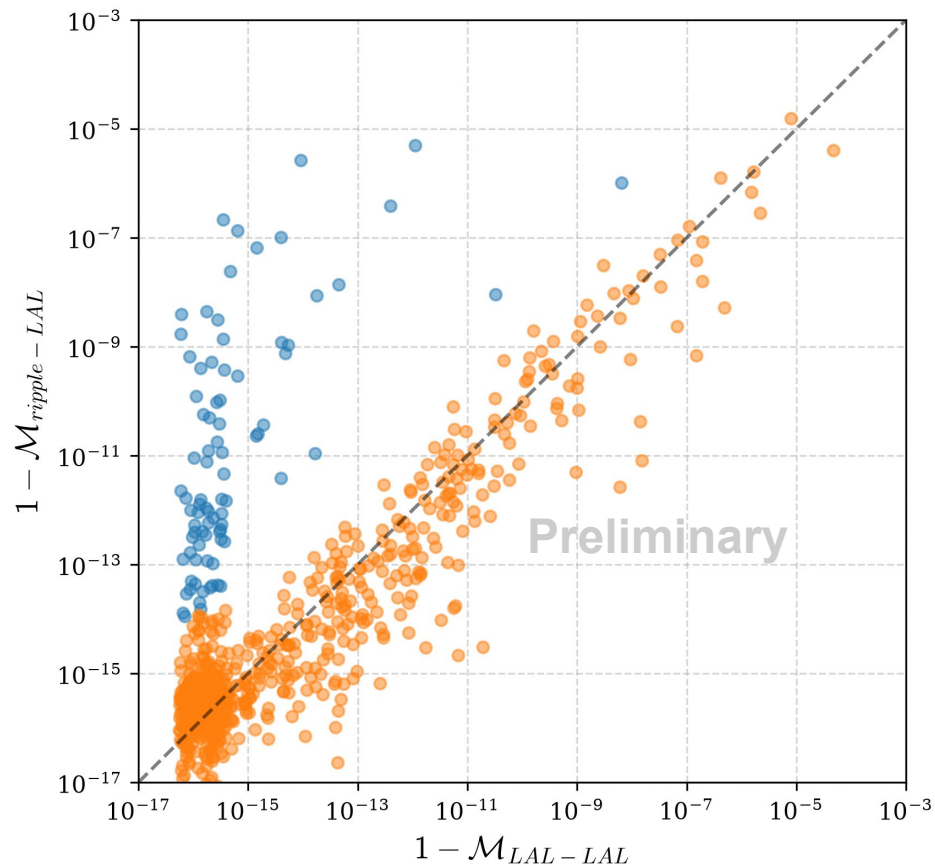
Backup Slides

Precession tail

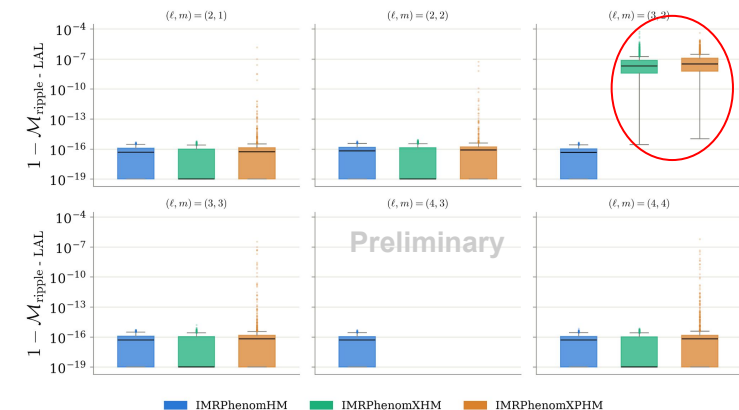
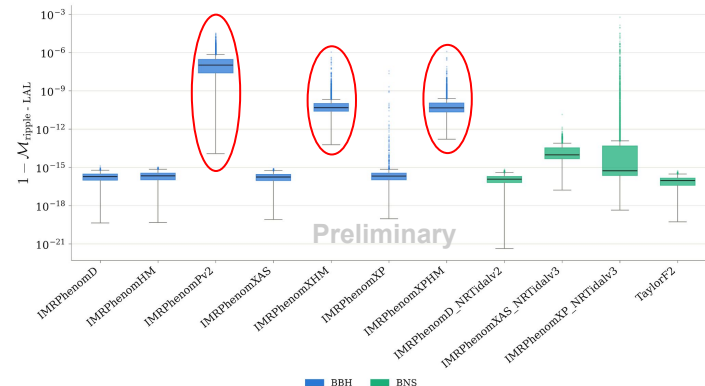
MSA instability
at small
precession cone
opening angles



Perturbing LAL inputs by
 $1e-16$ in IMRPhenomXP
tracks ripple-LAL mismatch



AD vs finite difference



Pv2:

- Derivative in time of coalescence

X(P)HM: [2001.10914](https://arxiv.org/abs/2001.10914)

- (3,2)-mode affected by mode mixing
- Transformation from spheroidal harmonics -> spherical harmonics basis includes derivatives

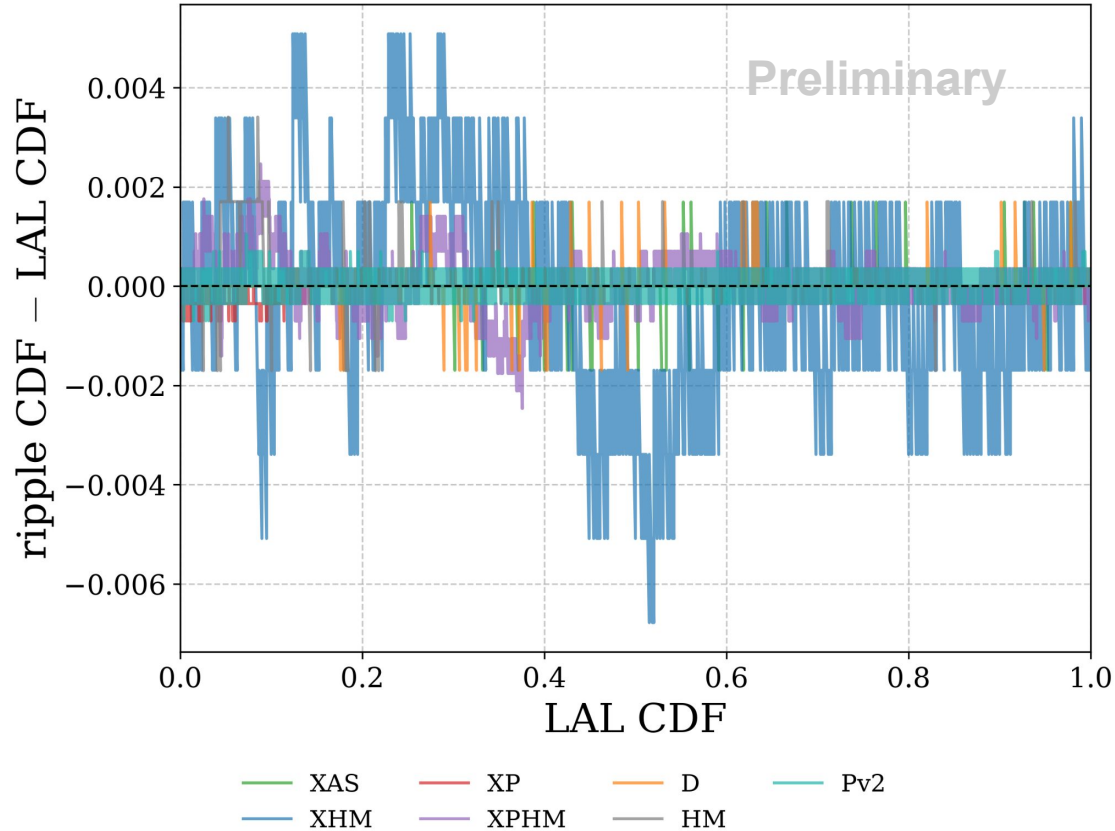
$$\phi_{32,S} = -\frac{\alpha_2^{32}}{f} - \frac{\alpha_4^{32}}{3f^3} + \alpha_\lambda^{32} \tan^{-1}\left(\frac{f - f_{ring}^{32}}{f_{damp}^{32}}\right) + d\phi_{RD}^{32}f + \phi_{RD}^{32}, \quad (6.7)$$

Actually a double derivative!

$$\Delta T_{32,S} = \phi'_{32,S}(f_0) - \phi'_{22}(f_0) \quad (6.9)$$

$$\Delta \phi_{32,S} = \phi_{32,S}(f_1) - \phi_{22}(f_1), \quad (6.10)$$

A messy pp-plot for NR comparisons



JSD divergences in catalogue runs

