Scalable Bayesian Inference with Hardware Accelerators and Normalizing Flows

Thibeau Wouters

Utrecht University

Nikhef

Contents

Parameter estimation

Estimate parameters θ of a model for data d with Bayesian inference:

$$p(\theta|d) \propto p(d|\theta)p(\theta)$$

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

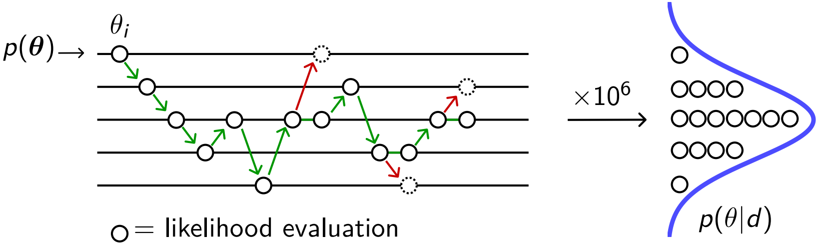
Parameter estimation

Estimate parameters θ of a model for data d with Bayesian inference:

$$p(\theta|d) \propto p(d|\theta)p(\theta)$$

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

- Sample the posterior: MCMC or nested sampling
- Propose samples, accept/reject based on likelihood
- $\mathcal{O}(10^6)$ likelihood evaluations: computational bottleneck



Future gravitational wave detectors

Future GW detectors: $10\times$ more sensitive

- $\mathcal{O}(10^5)$ events/year (now: $\mathcal{O}(10^2)$ events/decade)
- Signals are longer, louder, and overlap

Future gravitational wave detectors

Future GW detectors: $10\times$ more sensitive

- $\mathcal{O}(10^5)$ events/year (now: $\mathcal{O}(10^2)$ events/decade)
- Signals are longer, louder, and overlap

Premise: Current software does not meet these demands [1]

Future gravitational wave detectors

Future GW detectors: $10\times$ more sensitive

- $\mathcal{O}(10^5)$ events/year (now: $\mathcal{O}(10^2)$ events/decade)
- Signals are longer, louder, and overlap

Premise: Current software does not meet these demands [1]

How to make parameter estimation scalable?

- Reduce cost of likelihood evaluations
- Improve MCMC proposals

Future gravitational wave detectors

Future GW detectors: $10\times$ more sensitive

- $\mathcal{O}(10^5)$ events/year (now: $\mathcal{O}(10^2)$ events/decade)
- Signals are longer, louder, and overlap

Premise: Current software does not meet these demands [1]

How to make parameter estimation scalable?

- Reduce cost of likelihood evaluations
- Improve MCMC proposals

Goal: Fast sampling with minimal pretraining: flexible alternative to simulation-based inference [2–6]

Contents

JAX & FLOWMC (Kaze Wong)

Accelerate Python with JAX:

- GPUs
- Automatic differentiation:
    - Gradient-based samplers
    - Optimization

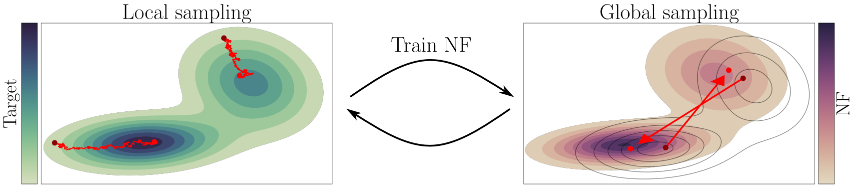
JAX & FLOWMC (Kaze Wong)

Accelerate Python with JAX:

- GPUs
- Automatic differentiation:
  - Gradient-based samplers
  - Optimization

FLOWMC [7, 8]:

- MCMC + normalizing flow proposals in JAX
- Training data: MCMC chains → no pre-training



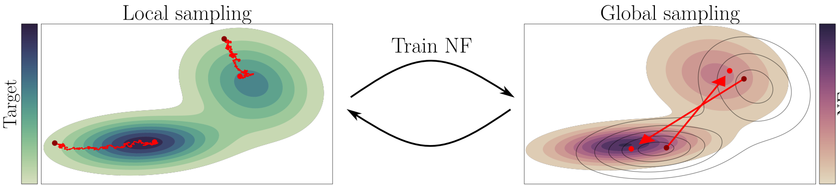
JAX & FLOWMC (Kaze Wong)

Accelerate Python with JAX:

- GPUs
- Automatic differentiation:
  - Gradient-based samplers
  - Optimization

FLOWMC [7, 8]:

- MCMC + normalizing flow proposals in JAX
- Training data: MCMC chains → no pre-training
- Also see NESSAI [9, 10], POCOMC [11]

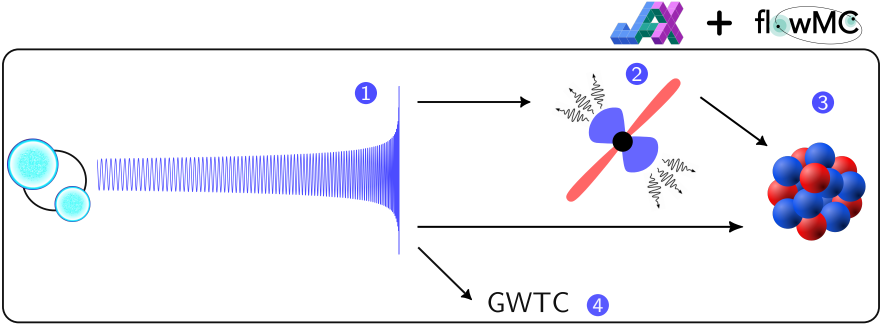


Contents

Overview

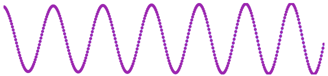
Analyzing a multi-messenger **binary neutron star** signal:

1. **Gravitational waves**
2. Electromagnetic counterparts
3. Nuclear equation of state
4. Gravitational wave transient catalogue



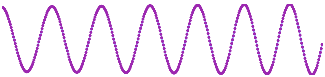
Gravitational waves

- Waveforms on GPU: $\mathcal{O}(10^3)$ faster
- From LALSUITE to JAX: RIPPLE [12]
  - Also see SFTS [13]



Gravitational waves

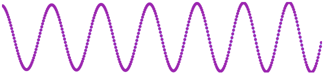
- Waveforms on GPU: $\mathcal{O}(10^3)$ faster
- From LALSUITE to JAX: RIPPLE [12]
  - Also see SFTS [13]



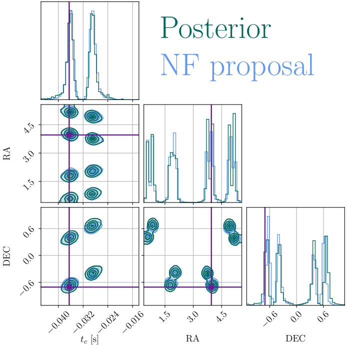
- Parameter estimation: JIM [14, 15]
- ✓ Current detectors
  - Hours $\rightarrow$ minutes

Gravitational waves

- Waveforms on GPU: $\mathcal{O}(10^3)$ faster
- From LALSUITE to JAX: RIPPLE [12]
  - Also see SFTS [13]



- Parameter estimation: JIM [14, 15]
- ✓ Current detectors
  - Hours → minutes
- Ongoing work for future detectors:
  - Binary neutron star: 13D
  - Einstein Telescope
  - 30 mins on H100 GPU

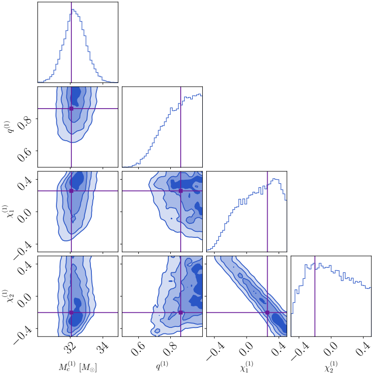


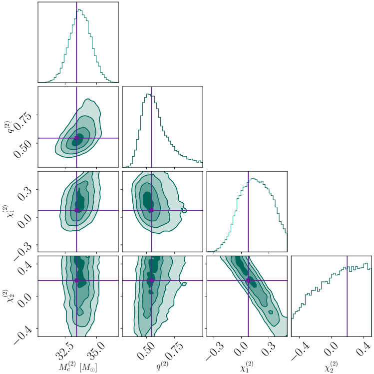
Overlapping signals (Luca Negri, Justin Janquart, James Alvey, Uddipta Bhardwaj)

- Assess scaling of JIM: BBH+BBH with LIGO-Virgo
  - 2 binary black hole mergers: 22 parameters
  - $M_c^{(1)} = 32M_\odot$, $M_c^{(2)} = 33M_\odot$, $\Delta t = 70$ ms
  - $\mathrm{SNR}^{(1)} = 25.76$, $\mathrm{SNR}^{(2)} = 25.24$

Overlapping signals (Luca Negri, Justin Janquart, James Alvey, Uddipta Bhardwaj)

- Assess scaling of JIM: BBH+BBH with LIGO-Virgo
  - 2 binary black hole mergers: 22 parameters
  - $M_c^{(1)} = 32M_\odot$, $M_c^{(2)} = 33M_\odot$, $\Delta t = 70$ ms
  - $\mathrm{SNR}^{(1)} = 25.76$, $\mathrm{SNR}^{(2)} = 25.24$
  - 1h28m on H100 (vs 23 days on 16 CPUs [16])



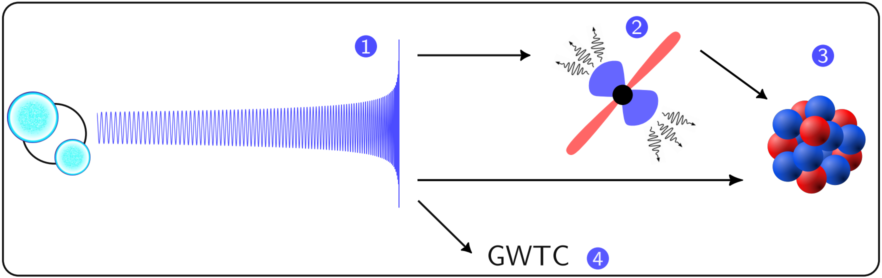


Overview

Analyzing a multi-messenger **binary neutron star** signal:

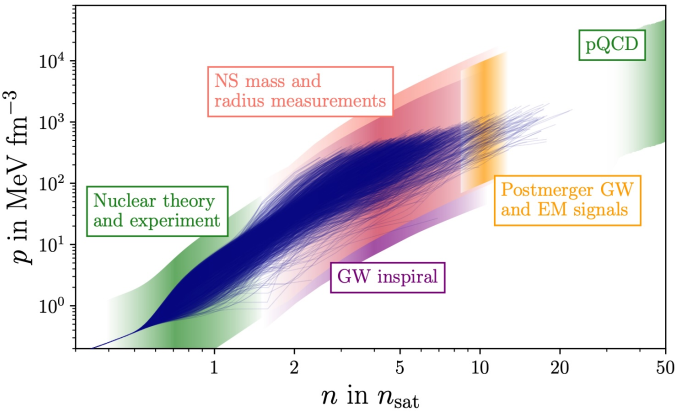
1. Gravitational waves
2. Electromagnetic counterparts
3. **Nuclear equation of state**
4. Gravitational wave transient catalogue

+



The nuclear equation of state

- The equation of state of dense nuclear matter is uncertain [17]
- Neutron stars probe its high density regime
- Solve inverse problem with Bayesian inference



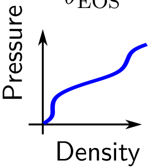
Equation of state

- Parametrization θ_{EOS}: constrain with Bayesian inference

Equation of state

- Parametrization θ_{EOS}: constrain with Bayesian inference
- To predict neutron star properties, we solve the TOV equations: ordinary differential equations (ODEs)

θ_{EOS}



$\longrightarrow$

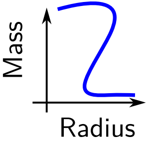
Solve TOV equations

$$\frac{\mathrm{d}p}{\mathrm{d}r} = -\frac{\varepsilon(r)m(r)}{r^2}\frac{\left[1+\frac{p(r)}{\varepsilon(r)}\right]\left[1+\frac{4\pi r^3 p(r)}{m(r)}\right]}{\left[1-\frac{2m(r)}{r}\right]}$$

$$\frac{\mathrm{d}m}{\mathrm{d}r} = 4\pi r^2 \varepsilon(r)\,,$$

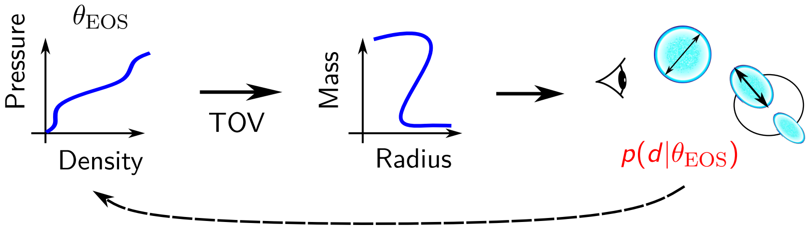
$\longrightarrow$

Neutron stars



Equation of state

- Parametrization θ_{EOS}: constrain with Bayesian inference
- To predict neutron star properties, we solve the TOV equations: ordinary differential equations (ODEs)
- Done for each sample θ_{EOS}: costly likelihood

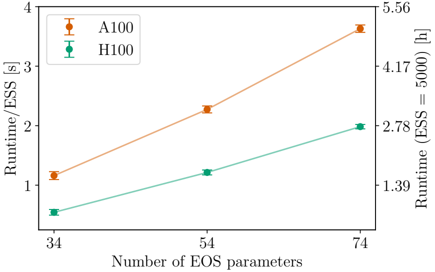


JESTER (Peter T.H. Pang)

- Solving TOV equations (EOS $\rightarrow$ NS) is slow

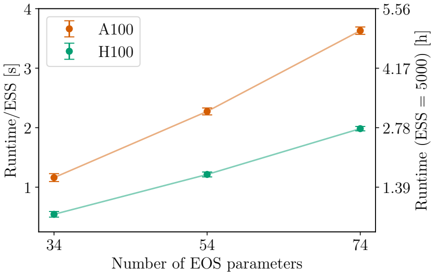
Jester (Peter T.H. Pang)

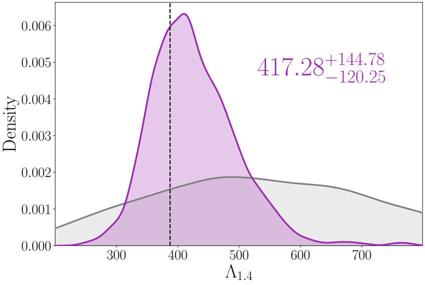
- Solving TOV equations (EOS $\rightarrow$ NS) is slow
- Jester [18]: Jax-based TOV solver
  - Full inference in $\sim$hours
  - No need for ML emulators



JESTER (Peter T.H. Pang)

- Solving TOV equations (EOS $\rightarrow$ NS) is slow
- JESTER [18]: JAX-based TOV solver
  - Full inference in $\sim$hours
  - No need for ML emulators
- End-to-end analysis: from gravitational waves of neutron star mergers to the equation of state
  - Example: 20 BNS in O5





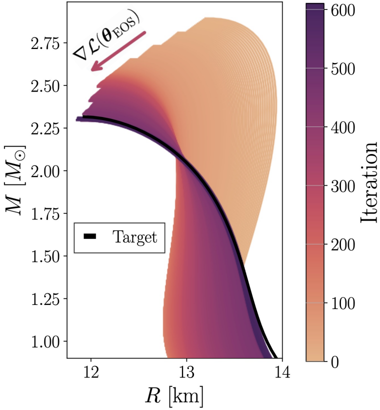
Auto-differentiable ODE solvers

- ODE solvers in JAX are auto-differentiable (DIFFRAX)
- Frame inference as optimization problem:
  - Gradient descent on loss function $\mathcal{L}(\theta_{\rm EOS})$

Auto-differentiable ODE solvers

- ODE solvers in JAX are auto-differentiable (DIFFRAX)
- Frame inference as optimization problem:
  - Gradient descent on loss function $\mathcal{L}(\theta_{\mathrm{EOS}})$

$$\mathcal{L}(\boldsymbol{\theta}_{\mathrm{EOS}}) = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{R_i(\boldsymbol{\theta}_{\mathrm{EOS}}) - \hat{R}_i}{\hat{R}_i}\right|$$



Contents

Conclusion

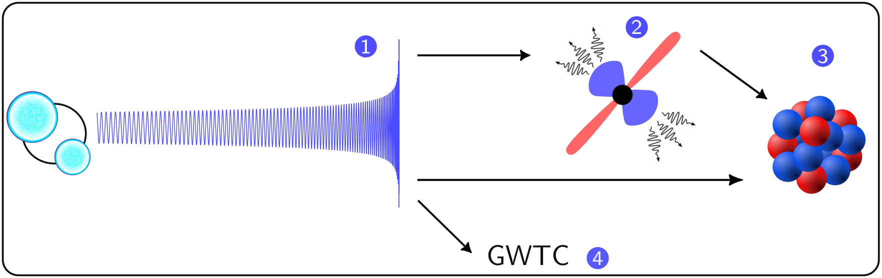
- Progress on scalable Bayesian inference, with minimal pre-training
- Hybrid acceleration: GPUs + normalizing flow proposals
  - JAX/GPU: faster likelihoods
  - flowMC: sampling converges faster
- Simulators in JAX can remove the need for emulators (GW, TOV)
- Auto-differentiable ODE solvers: inference as optimization problem

Let's talk!

Thank you for your attention!

Software written in JAX:

- FLOWMC [7, 8]
- JIM [14, 15] 1 2 4
- FIESTA 2
- JESTER [18] (built with DIFFRAX) 3
- HARMONIC [19–21]



References I

[1] Qian Hu and John Veitch. “Costs of Bayesian Parameter Estimation in Third-Generation Gravitational Wave Detectors: a Review of Acceleration Methods”. In: (Dec. 2024). arXiv: 2412.02651 [gr-qc].

[2] Jurriaan Langendorff et al. “Normalizing Flows as an Avenue to Studying Overlapping Gravitational Wave Signals”. In: *Phys. Rev. Lett.* 130.17 (2023), p. 171402. DOI: 10.1103/PhysRevLett.130.171402. arXiv: 2211.15097 [gr-qc].

[3] Uddipta Bhardwaj et al. “Sequential simulation-based inference for gravitational wave signals”. In: *Phys. Rev. D* 108.4 (2023), p. 042004. DOI: 10.1103/PhysRevD.108.042004. arXiv: 2304.02035 [gr-qc].

[4] Maximilian Dax et al. “Real-time inference for binary neutron star mergers using machine learning”. In: *Nature* 639.8053 (2025), pp. 49–53. DOI: 10.1038/s41586-025-08593-z. arXiv: 2407.09602 [gr-qc].

[5] Qian Hu et al. “Decoding Long-duration Gravitational Waves from Binary Neutron Stars with Machine Learning: Parameter Estimation and Equations of State”. In: (Dec. 2024). arXiv: 2412.03454 [gr-qc].

[6] Filippo Santoliquido et al. “Fast and accurate parameter estimation of high-redshift sources with the Einstein Telescope”. In: (Apr. 2025). arXiv: 2504.21087 [astro-ph.HE].

References II

[7] Marylou Gabrié, Grant M. Rotskoff, and Eric Vanden-Eijnden. "Adaptive Monte Carlo augmented with normalizing flows". In: *Proc. Nat. Acad. Sci.* 119.10 (2022), e2109420119. DOI: 10.1073/pnas.2109420119. arXiv: 2105.12603 [physics.data-an].

[8] Kaze W. k. Wong, Marylou Gabrié, and Daniel Foreman-Mackey. "flowMC: Normalizing flow enhanced sampling package for probabilistic inference in JAX". In: *J. Open Source Softw.* 8.83 (2023), p. 5021. DOI: 10.21105/joss.05021. arXiv: 2211.06397 [astro-ph.IM].

[9] Michael J. Williams, John Veitch, and Chris Messenger. "Nested sampling with normalizing flows for gravitational-wave inference". In: *Phys. Rev. D* 103.10 (2021), p. 103006. DOI: 10.1103/PhysRevD.103.103006. arXiv: 2102.11056 [gr-qc].

[10] Michael J. Williams, John Veitch, and Chris Messenger. "Importance nested sampling with normalising flows". In: *Mach. Learn. Sci. Tech.* 4.3 (2023), p. 035011. DOI: 10.1088/2632-2153/acd5aa. arXiv: 2302.08526 [astro-ph.IM].

[11] Minas Karamanis et al. "pocoMC: A Python package for accelerated Bayesian inference in astronomy and cosmology". In: *J. Open Source Softw.* 7.79 (2022), p. 4634. DOI: 10.21105/joss.04634. arXiv: 2207.05660 [astro-ph.IM].

[12] Thomas D. P. Edwards et al. "Differentiable and hardware-accelerated waveforms for gravitational wave data analysis". In: *Phys. Rev. D* 110.6 (2024), p. 064028. DOI: 10.1103/PhysRevD.110.064028. arXiv: 2302.05329 [astro-ph.IM].

References III

[13] Rodrigo Tenorio and Davide Gerosa. "Scalable data-analysis framework for long-duration gravitational waves from compact binaries using short Fourier transforms". In: *Phys. Rev. D* 111.10 (2025), p. 104044. DOI: 10.1103/PhysRevD.111.104044. arXiv: 2502.11823 [gr-qc].

[14] Kaze W. K. Wong, Maximiliano Isi, and Thomas D. P. Edwards. "Fast Gravitational-wave Parameter Estimation without Compromises". In: *Astrophys. J.* 958.2 (2023), p. 129. DOI: 10.3847/1538-4357/acf5cd. arXiv: 2302.05333 [astro-ph.IM].

[15] Thibeau Wouters et al. "Robust parameter estimation within minutes on gravitational wave signals from binary neutron star inspirals". In: *Phys. Rev. D* 110.8 (2024), p. 083033. DOI: 10.1103/PhysRevD.110.083033. arXiv: 2404.11397 [astro-ph.IM].

[16] Justin Janquart et al. "Analyses of overlapping gravitational wave signals using hierarchical subtraction and joint parameter estimation". In: *Mon. Not. Roy. Astron. Soc.* 523.2 (2023), pp. 1699–1710. DOI: 10.1093/mnras/stad1542. arXiv: 2211.01304 [gr-qc].

[17] Hauke Koehn et al. "From existing and new nuclear and astrophysical constraints to stringent limits on the equation of state of neutron-rich dense matter". In: (Feb. 2024). arXiv: 2402.04172 [astro-ph.HE].

[18] Thibeau Wouters et al. "Leveraging differentiable programming in the inverse problem of neutron stars". In: (Apr. 2025). arXiv: 2504.15893 [astro-ph.HE].

References IV

[19] Jason D. McEwen et al. *Machine learning assisted Bayesian model comparison: learnt harmonic mean estimator*. 2023. arXiv: 2111.12720 [stat.ME]. URL: https://arxiv.org/abs/2111.12720.

[20] Alicja Polanska et al. *Learned harmonic mean estimation of the marginal likelihood with normalizing flows*. 2024. arXiv: 2307.00048 [stat.ME]. URL: https://arxiv.org/abs/2307.00048.

[21] Alicja Polanska et al. "Accelerated Bayesian parameter estimation and model selection for gravitational waves with normalizing flows". In: *38th conference on Neural Information Processing Systems*. Oct. 2024. arXiv: 2410.21076 [astro-ph.IM].

[22] Kurzgesagt. *Figures taken from "Neutron Stars - The Most Extreme Things that are not Black Holes"*. Accessed on May 14, 2025. 2019. URL: https://www.youtube.com/watch?v=udFxKZRyQt4.

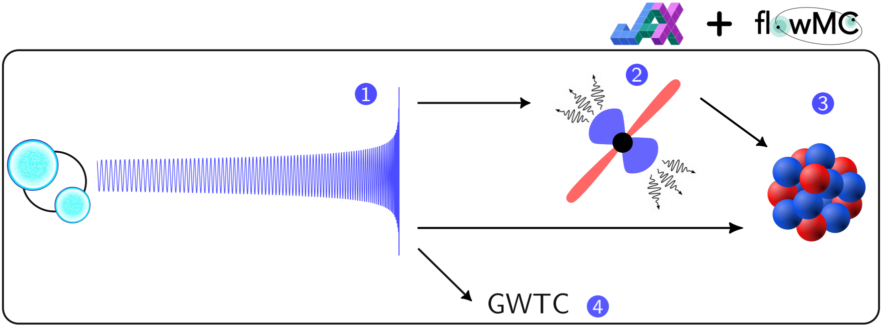
[23] Hergé. *Cover figure created with ChatGPT using this input figure from the comic Destination Moon*. Accessed on May 14, 2025. 2019. URL: https://www.youtube.com/watch?v=udFxKZRyQt4.

[24] Geoffrey Ryan et al. "Gamma-Ray Burst Afterglows in the Multimessenger Era: Numerical Models and Closure Relations". In: *Astrophys. J.* 896.2 (2020), p. 166. DOI: 10.3847/1538-4357/ab93cf. arXiv: 1909.11691 [astro-ph.HE].

Overview

Analyzing a multi-messenger **binary neutron star** signal:

1. Gravitational waves
2. **Electromagnetic counterparts**
3. Nuclear equation of state
4. Gravitational wave transient catalogue

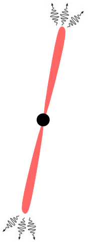


Electromagnetic counterparts (Hauke Koehn, Tim Dietrich)

- BNS mergers lead to kilonovae, gamma-ray bursts (afterglows)
- Numerical models are expensive (e.g. AFTERGLOWPY [24])

Electromagnetic counterparts (Hauke Koehn, Tim Dietrich)

- BNS mergers lead to kilonovae, gamma-ray bursts (afterglows)
- Numerical models are expensive (e.g. AFTERGLOWPY [24])
- Neural network emulators for inference: FIESTA

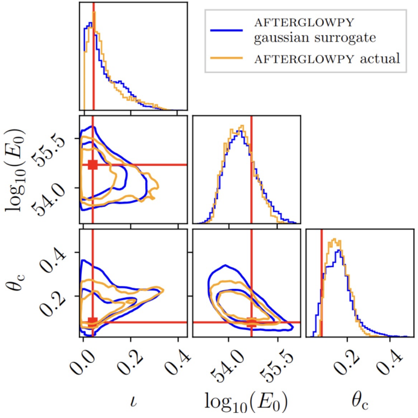


FIESTA

- 1m36s
- 1 H100 GPU

AFTERGLOWPY

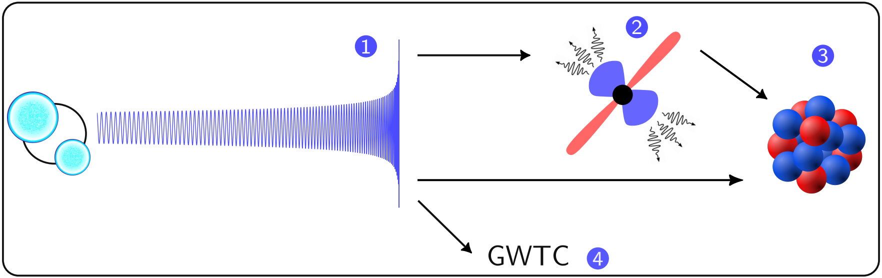
- 4 hours
- 30 CPUs



Overview

Analyzing a multi-messenger **binary neutron star** signal:

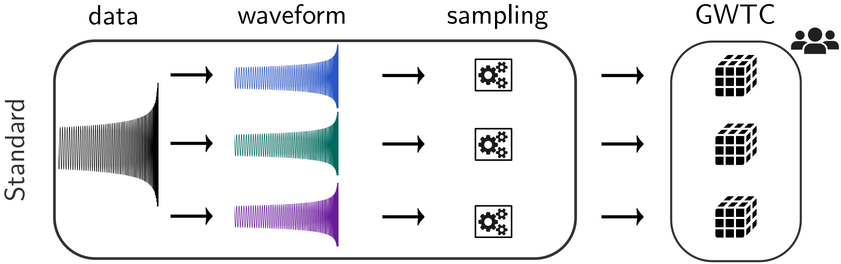
1. Gravitational waves
2. Electromagnetic counterparts
3. Nuclear equation of state
4. **Gravitational wave transient catalogue**



Constructing GWTCs (Thomas Ng, Kaze Wong)

GWTCs do not scale well in memory:

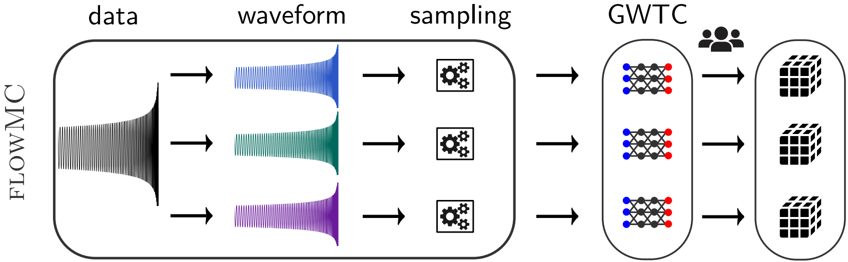
- GWTC stores several samples (different waveforms)
- Standard: fixed sample size, $\sim$ 100 MB



Constructing GWTCs (Thomas Ng, Kaze Wong)

GWTCs do not scale well in memory:

- GWTC stores several samples (different waveforms)
- Standard: fixed sample size, $\sim$ 100 MB
- FLOWMC: generate samples from normalizing flows, $\sim$ 10 MB
  - Also see Michael Williams' talk/poster



Evidence calculation: HARMONIC I

Evidence Z can be computed from posterior samples with HARMONIC [19] with the harmonic mean estimator

$$\rho \equiv \mathbb{E}_{P(\theta|d)}\left[\frac{1}{L(\theta)}\right] = \int \mathrm{d}\theta \frac{1}{\mathcal{L}(\theta)} P(\theta|d) = \int \mathrm{d}\theta \frac{1}{\mathcal{L}(\theta)} \frac{\mathcal{L}(\theta)\pi(\theta)}{Z} = \frac{1}{Z}$$

Therefore, estimate ρ with posterior samples:

$$\hat{\rho} = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{\mathcal{L}(\theta)}, \quad \theta_i \sim P(\theta|d)$$

Evidence calculation: HARMONIC II

Can be interpreted as importance sampling

$$\rho = \int \mathrm{d}\theta \frac{1}{Z} \frac{\pi(\theta)}{P(\theta|d)} P(\theta|d) \,,$$

but with target = prior and sampling density = posterior. Therefore, importance sampling is inefficient – how to solve?
New proposal:

$$\begin{aligned}\rho &= \mathbb{E}_{P(\theta|d)} \left[\frac{\varphi(\theta)}{\mathcal{L}(\theta)\pi(\theta)} \right] \\ &= \int \mathrm{d}\theta \, \frac{\varphi(\theta)}{\mathcal{L}(\theta)\pi(\theta)} \, P(\theta|d) \\ &= \int \mathrm{d}\theta \, \frac{\varphi(\theta)}{\mathcal{L}(\theta)\pi(\theta)} \, \frac{\mathcal{L}(\theta)\pi(\theta)}{Z} = \frac{1}{Z}\end{aligned}$$

Evidence calculation: HARMONIC III

Use the following estimator:

$$\hat{\rho} = \frac{1}{N} \sum_{i=1}^{N} \frac{\varphi(\theta_i)}{\mathcal{L}(\theta_i)\pi(\theta_i)}, \quad \theta_i \sim P(\theta|d)$$

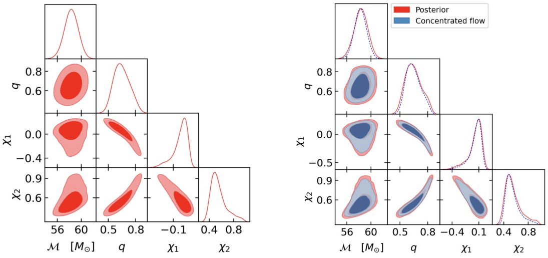
Replace the target distribution π with φ: only requirement is that it is normalized

In practice, this can be achieved with a normalizing flow [20].

This has been verified to give accurate evidences (similar values as nested sampling) when GW posteriors are used [21].

Table 1: Total wall times to compute the evidence estimates for the examples discussed in the main text. We run BILBY on 16 CPU cores and JIM + `harmonic` on 1 GPU.

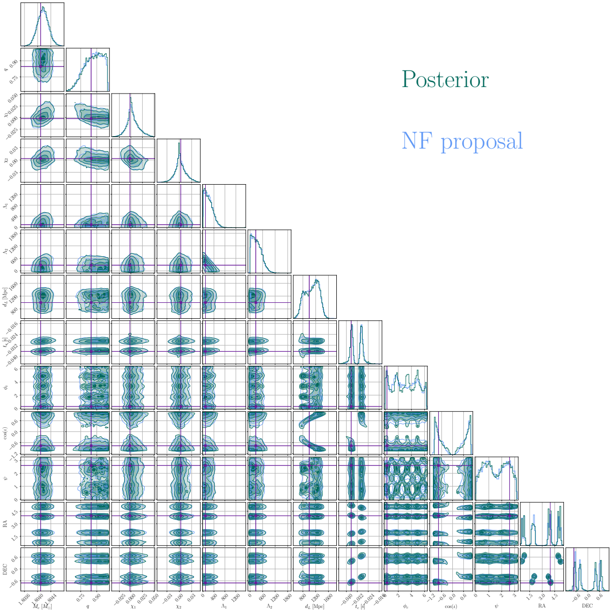
Example	Method	$\log(z)$	Sampling time	Evidence estimation time
4D	BILBY	390.33 ± 0.11	31.3 min	–
	JIM + `harmonic`	$390.360^{+0.006}_{-0.006}$	3.4 min	1.9 min
11D	BILBY	378.29 ± 0.15	3.5 h	–
	JIM + `harmonic`	$378.420^{+0.09}_{-0.08}$	11.8 min	2.4 min



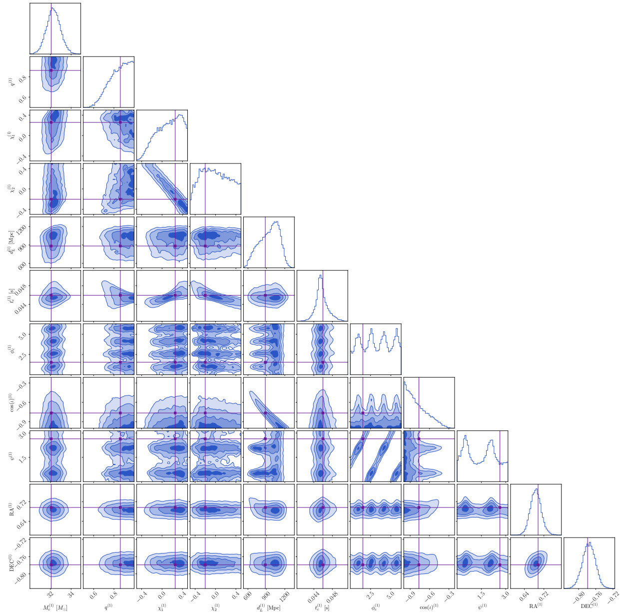
(a) BILBY (b) JIM + `harmonic`

Figure 1: Corner plots for the 4-dimensional posterior samples from (a) BILBY and (b) JIM used for inference (solid red) alongside the concentrated flow at $T = 0.8$ used in the learned harmonic mean (dashed blue).

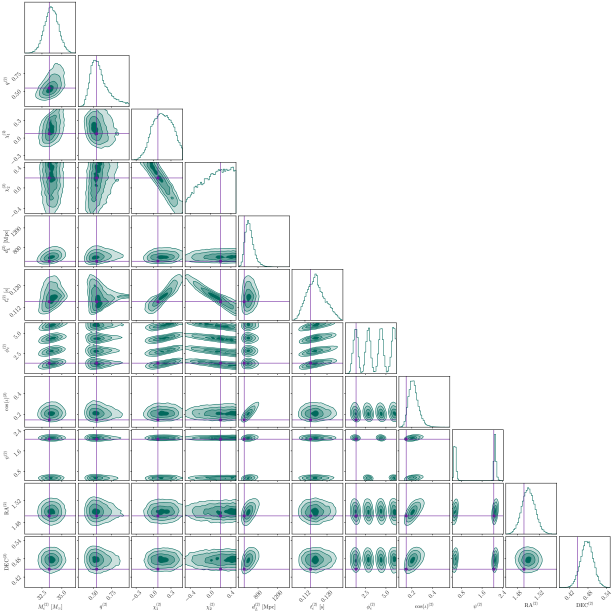
BNS in ET-Δ example: all parameters



Overlapping signals: all parameters signal A

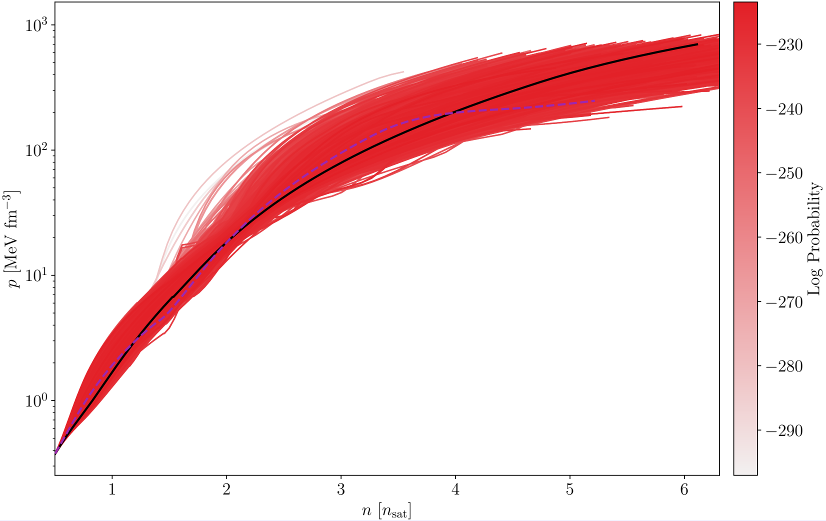


Overlapping signals: all parameters signal B



Equation of state O5 projection with 20 BNS: EOS

- **Purple**: target
- **Red**: posterior EOS samples (**black**: maximum log posterior)



Equation of state O5 projection with 20 BNS: NS