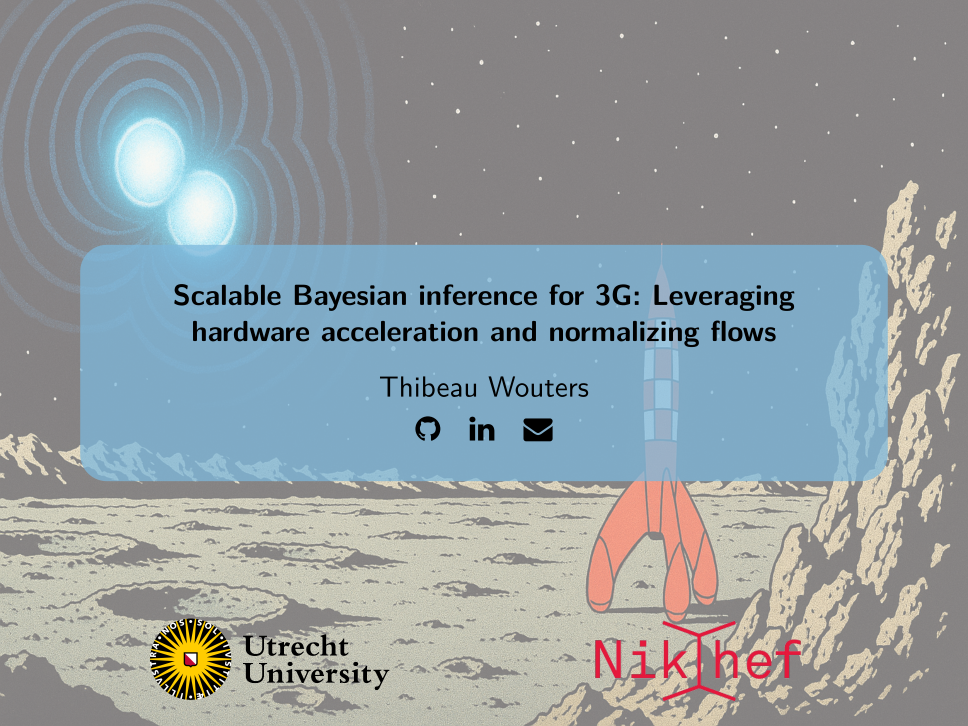


Scalable Bayesian inference for 3G: Leveraging hardware acceleration and normalizing flows

Thibeau Wouters

Utrecht University

Nikhef

Contents

Parameter estimation in 3G

Parameter estimation is done with Bayesian inference:

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

- Sample the posterior: MCMC or nested sampling
- $\mathcal{O}(10^6) - \mathcal{O}(10^8)$ likelihood evaluations per inference

Parameter estimation in 3G

Parameter estimation is done with Bayesian inference:

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

- Sample the posterior: MCMC or nested sampling
- $\mathcal{O}(10^6) - \mathcal{O}(10^8)$ likelihood evaluations per inference

What about 3G detectors?

- ET will observe $\mathcal{O}(10^5)$ events per year
- Signals will be longer ($f_{\min} = 5$ Hz) and have higher SNRs

Parameter estimation in 3G

Parameter estimation is done with Bayesian inference:

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

- Sample the posterior: MCMC or nested sampling
- $\mathcal{O}(10^6) - \mathcal{O}(10^8)$ likelihood evaluations per inference

What about 3G detectors?

- ET will observe $\mathcal{O}(10^5)$ events per year
- Signals will be longer ($f_{\text{min}} = 5$ Hz) and have higher SNRs

Premise: Current software will not scale to 3G [1]

Towards scalable inference

Ingredients to make inference scalable:

- Hardware acceleration: GPUs for parallel computations
- Normalizing flows (NFs): neural density estimators

Towards scalable inference

Ingredients to make inference scalable:

- Hardware acceleration: GPUs for parallel computations
- Normalizing flows (NFs): neural density estimators

Both are used in two ways:

- Simulation-based inference [2–6]:
  - \+ Fast at inference ($\sim$seconds)
  - – Needs pre-trained model
  - See Lucia Papalini, Filippo Santoliquido,...

Towards scalable inference

Ingredients to make inference scalable:

- Hardware acceleration: GPUs for parallel computations
- Normalizing flows (NFs): neural density estimators

Both are used in two ways:

- Simulation-based inference [2–6]:
  - + Fast at inference ($\sim$seconds)
  - − Needs pre-trained model
  - See Lucia Papalini, Filippo Santoliquido,...
- Hybrid approach: faster likelihoods + NF proposals
  - + No pre-training: more flexibility
  - − Re-implement software
  - Also see Luca Negri's poster: neural likelihood estimators

Towards scalable inference

Ingredients to make inference scalable:

- Hardware acceleration: GPUs for parallel computations
- Normalizing flows (NFs): neural density estimators

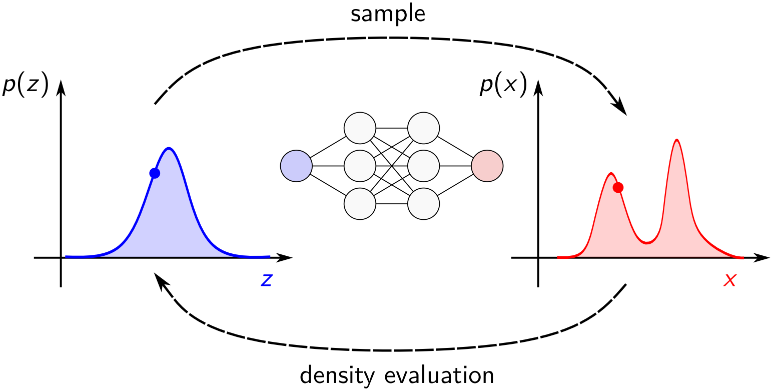
Both are used in two ways:

- Simulation-based inference [2–6]:
  - \+ Fast at inference (∼seconds)
  - – Needs pre-trained model
  - See Lucia Papalini, Filippo Santoliquido,...
- **Hybrid approach: faster likelihoods + NF proposals → this talk**
  - \+ No pre-training: more flexibility
  - – Re-implement software
  - Also see Luca Negri's poster: neural likelihood estimators

Contents

Normalizing flows (NFs)

- Trainable bijection between latent and data spaces
- Sample and evaluate complicated densities
- Used as proposal distribution, trained from MCMC chains



Acceleration with JAX features:

- Use GPU accelerators
- Automatic differentiation $\rightarrow$ gradient-based samplers
- Just-in-time (JIT) compilation

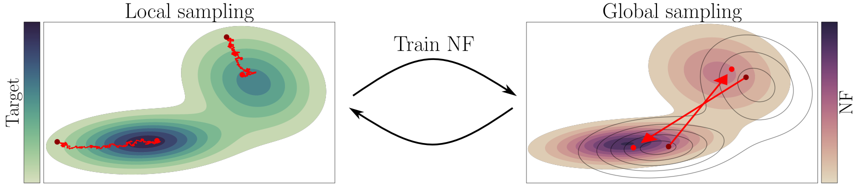
JAX & FLOWMC

Acceleration with JAX features:

- Use GPU accelerators
- Automatic differentiation → gradient-based samplers
- Just-in-time (JIT) compilation

FLOWMC [7, 8]: MCMC + normalizing flows + JAX

- MCMC chains as training data: no pre-training
- Also see NESSAI [9, 10]

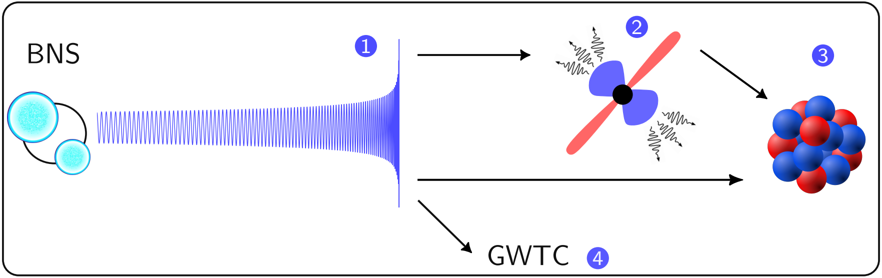


Contents

Overview

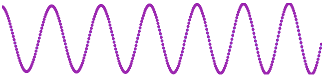
Analyzing a multi-messenger **binary neutron star** (BNS) signal:

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



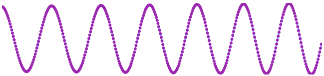
Gravitational waves

- Waveforms on GPU are $\mathcal{O}(10^3)$ faster
- From LALSUITE to JAX: RIPPLE [11]
  - Also see SFTS



Gravitational waves

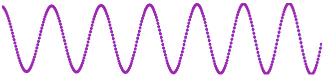
- Waveforms on GPU are $\mathcal{O}(10^3)$ faster
- From LALSUITE to JAX: RIPPLE [11]
  - Also see SFTS



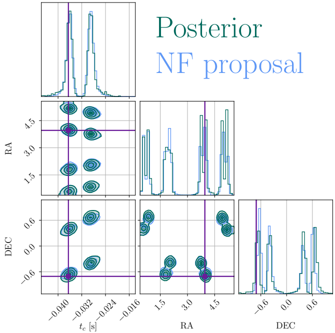
- Parameter estimation: JIM [12, 13]
- BNS in LVK analyzed in $\sim$ 15 min

Gravitational waves

- Waveforms on GPU are $\mathcal{O}(10^3)$ faster
- From LALSUITE to JAX: RIPPLE [11]
  - Also see SFTS



- Parameter estimation: JIM [12, 13]
- BNS in LVK analyzed in ~ 15 min
- Ongoing work for ET – example:
  - BNS, $f_{\min} = 20$ Hz, SNR $= 21$
  - ET-Δ, `IMRPhenomD_NRTidalv2`
  - 30 mins on H100 GPU
- Evidence: HARMONIC [14]



Overlapping signals

- In 3G, we expect to see overlapping signals
- Assess scaling: BBH+BBH in LVK

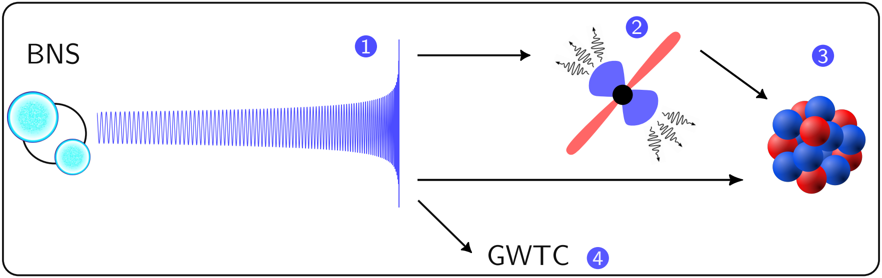
Either 61 or 102 or one of the later ones, need to check in detail

Overview

Analyzing a multi-messenger **binary neutron star** (BNS) signal:

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

+ flowMC



Electromagnetic counterparts (Hauke Koehn, Tim Dietrich)

- BNS mergers lead to kilonovae, gamma-ray bursts
- Numerical models are too expensive (e.g. AFTERGLOWPY [15])

Electromagnetic counterparts (Hauke Koehn, Tim Dietrich)

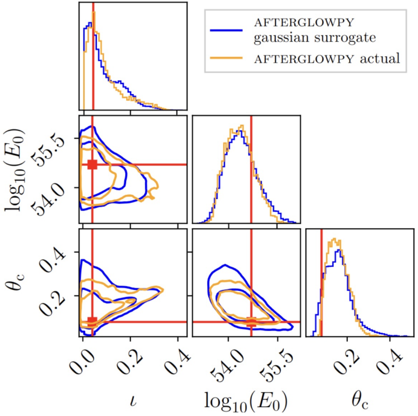
- BNS mergers lead to kilonovae, gamma-ray bursts
- Numerical models are too expensive (e.g. AFTERGLOWPY [15])
- Neural network surrogates for inference: FIESTA

FIESTA surrogates

- 1m36s
- 1 H100 GPU

AFTERGLOWPY

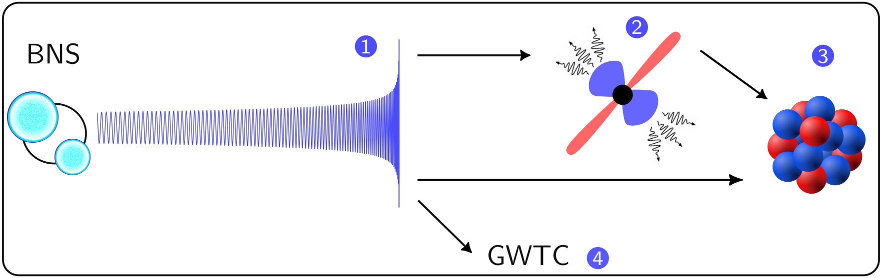
- 4 hours
- 30 CPUs



Overview

Analyzing a multi-messenger **binary neutron star** (BNS) signal:

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

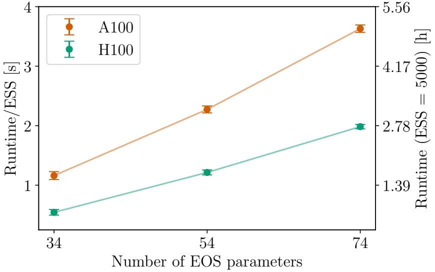


Equation of state inference

- Goal: infer nuclear equation of state (EOS) of neutron stars [16]
- Computational bottleneck: solve TOV equations

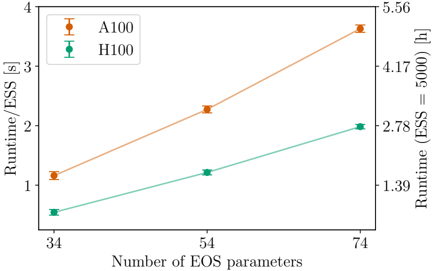
Equation of state inference

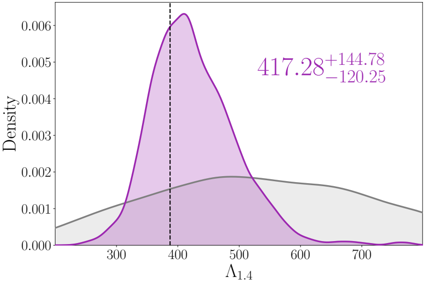
- Goal: infer nuclear equation of state (EOS) of neutron stars [16]
- Computational bottleneck: solve TOV equations
- JESTER [17]: JAX-based TOV solver
  - Full inference in ∼hours
  - No need for machine learning surrogates



Equation of state inference

- Goal: infer nuclear equation of state (EOS) of neutron stars [16]
- Computational bottleneck: solve TOV equations
- JESTER [17]: JAX-based TOV solver
  - Full inference in ∼hours
  - No need for machine learning surrogates
- End-to-end analysis: constrain EOS from 20 BNS in O5





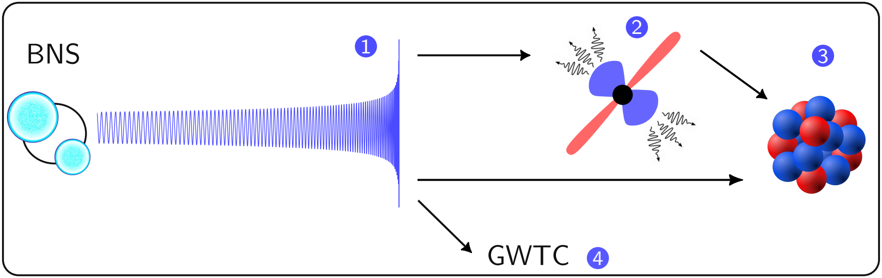
Contents

Overview

Analyzing a multi-messenger **binary neutron star** (BNS) signal:

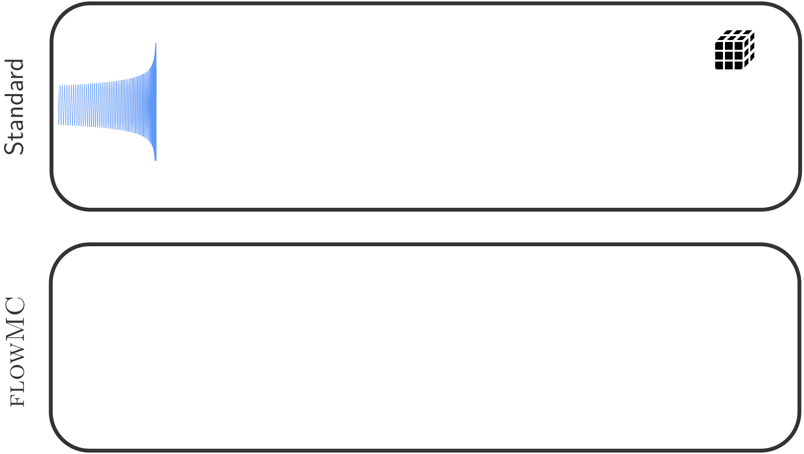
1. Gravitational waves
2. Electromagnetic counterparts
3. Nuclear equation of state
4. **Gravitational wave transient catalogue**

+ flowMC



Gravitational wave transient catalogue

How can we use the trained normalizing flow proposal from FlowMC?



Conclusion

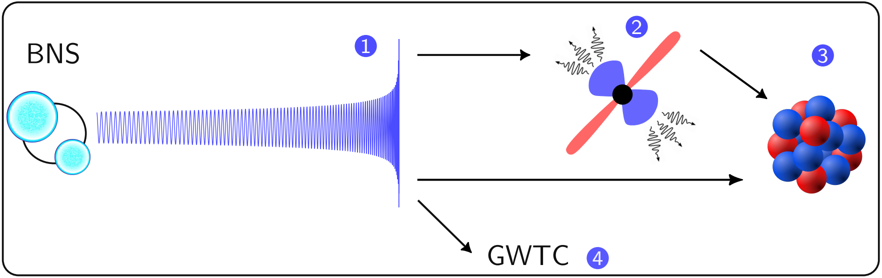
- Progress on scalable Bayesian inference software for 3G, with minimal amount of pre-training
- Hybrid acceleration: GPU + normalizing flow
  - JAX/GPU: likelihoods faster
  - FLOWMC: sampling converges faster
- Goal: joint multimessenger analyses in ~hours (NMMA [18] in JAX)
- To do
  - GW injection studies for ET
  - More waveform models in JAX
  - Equation of state study with ET data

Let's talk!

Thank you for your attention!

Software:

- FLOWMC [7, 8]
- JIM [12, 13] 1 2 4
- FIESTA 2
- JESTER [17] 3
- HARMONIC [14, 19, 20]



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] 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].

[12] 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].

References III

[13] 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].

[14] 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].

[15] 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].

[16] Adrian Abac et al. "The Science of the Einstein Telescope". In: (Mar. 2025). arXiv: 2503.12263 [gr-qc].

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

[18] Peter T. H. Pang et al. "An updated nuclear-physics and multi-messenger astrophysics framework for binary neutron star mergers". In: *Nature Commun.* 14.1 (2023). Available at https://github.com/nuclear-multimessenger-astronomy/nmma, p. 8352. DOI: 10.1038/s41467-023-43932-6. arXiv: 2205.08513 [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] 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.

[22] 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.

Evidence calculation: HARMONIC I

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

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

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 [14].

BNS in ET-Δ example: all parameters