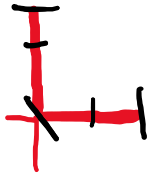
Advances in Gravitational Wave Inference

Greg Ashton – Royal Holloway

ROYAL HOLLOWAY UNIVERSITY OF LONDON

Inference: parameter estimation

The experiment give us data

 $\longrightarrow h(t)$ or $\underline{d}$

The theory $G_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$

gives us: Model M

Parameters θ

Bayes theorem:

"posterior" $p(\theta | \underline{d}, M)$ $\propto$ "likelihood" $\mathcal{L}(\underline{d} ; \theta, M)$ "prior" $\pi(\theta | M)$

Inference: model selection

The normalizing evidence:

$$p(\theta|d,M) = \frac{\mathcal{L}(d;\theta,M)\,\pi(\theta|M)}{Z(d|M)} \implies Z(d|M) = \int d\theta\; \mathcal{L}(d;\theta,M)\,\pi(\theta|M)$$

Can be used to compare models:

$$\underbrace{\frac{P(M_A|d)}{P(M_B|d)}}_{\text{"Odds"}} = \underbrace{\frac{Z(d|M_A)}{Z(d|M_B)}}_{\text{"Bayes Factor"}}\;\underbrace{\frac{\pi(M_A)}{\pi(M_B)}}_{\text{"Prior odds"}}$$

Inference in gravitational wave astronomy

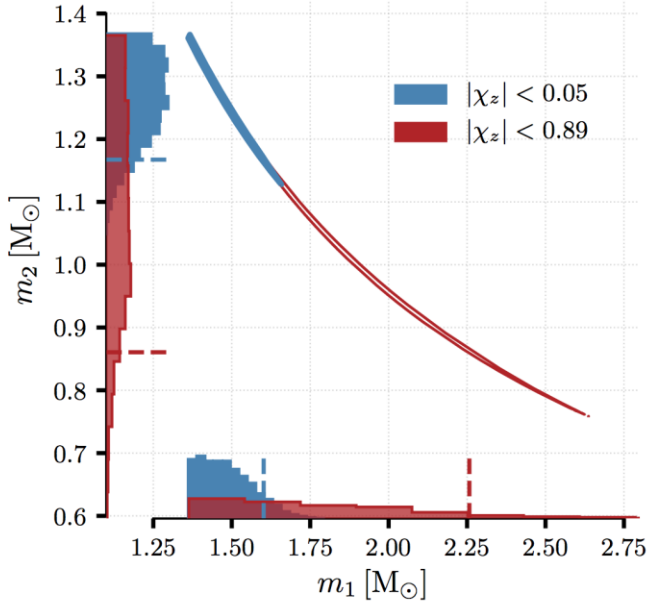
Why do you need Bayesian inference?

1. Non-linear correlations between parameters

2. Informative astrophysical priors

3. Single-event inference fits into hierarchical framework for population modelling

Mass posterior: [GW170817]

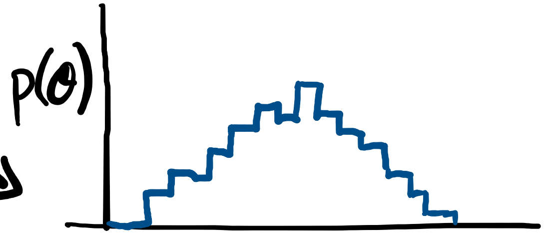


Stochastic sampling

- Approximate the posterior distribution with “samples”

$$p(\theta|d,M) \sim \{\theta_i\}$$

$$\bar{\theta} = \frac{1}{N}\sum \theta_i$$



- Estimate the evidence numerically (e.g. a Riemann sum)

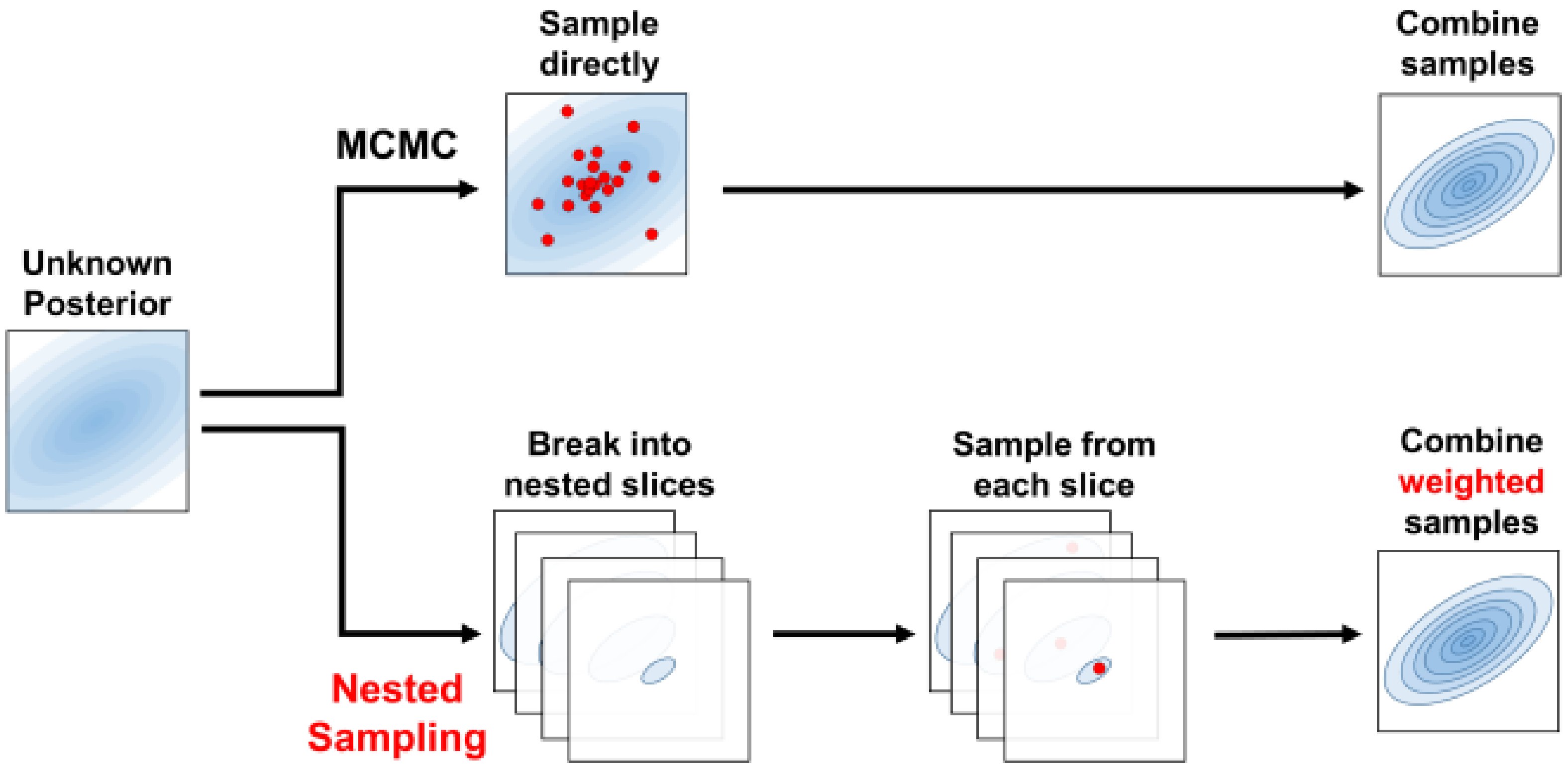
$$Z(d|M) \approx \sum_i \mathcal{L}(d;\theta_i,M)\,\pi(\theta_i|M)\,\Delta\theta$$

A historical overview

- First approaches based on Markov-Chain Monte-Carlo (MCMC) [Christensen & Meyer 1998]
- Then, Nested Sampling was introduced [Veitch & Vecchio 2008]
- A grid and sampling approach has since been applied by RIFT [Pankow et al. 2015]
- LALInference [Veitch et al. 2015]:
  - Used as flagship software O1-O3a
  - Offered both an MCMC and Nested Sampling package
  - Remains in active use
  - Ports to TGR based codes likely to remain in use for some time

Nested Sampling vs MCMC

- Two different approaches to the same problem



[Speagle (2019)](#)

Which is better?

- **Both**: cross-checking results between samplers has been vital
- LALInference MCMC was easier to parallelize:
  - Run tens of independent jobs and combine
  - Fits the high **throughput** computing (HTC) model of the LIGO Data Grid
- NS generally produces a more robust estimate of the evidence
- MCMC generally produces a more robust estimate of the posterior

The new era

LALInference is great, why build anything else?

- C-based code is hard to modify/extend for new users
- Pragmatically: the developers had permanent positions and the developer base was running thin

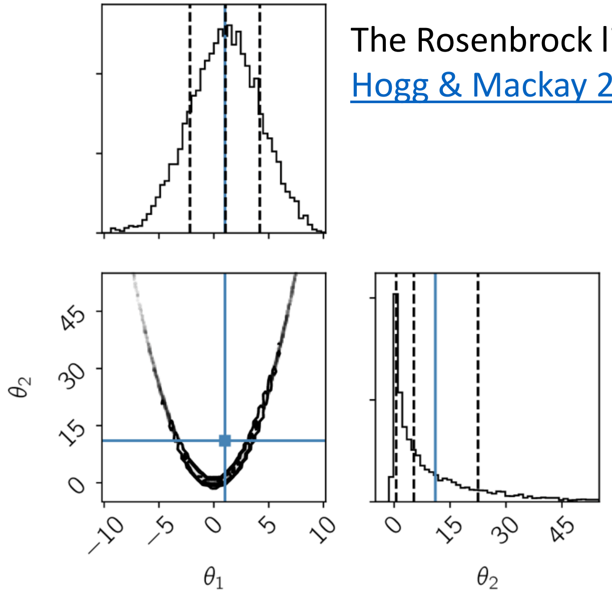
A host of new approaches have since been developed:

- PyCBC inference [Biwer et al. 2018]
- Bilby [Ashton et al. 2018]
- Bajes [Breschi et al. 2021]
- gwmodel [Pagano et al. in prep]
- + more which I have missed (please let me know!)

Bilby

- Package adopted by LIGO-Virgo-KAGRA for use in O3 onwards
- Interface between Bayesian inference concepts and **off-the-shelf** stochastic sampling packages
- Design principles:
  - Modularity, Consistency, Generality, Usability
- These principles have:
  - Lowered the barrier to users
  - Enabled rapid development of new ideas/approaches
- Used to analyse: CBCs, CWs, FRBs, GRBs, X-ray afterglows, …

The devil is in the details



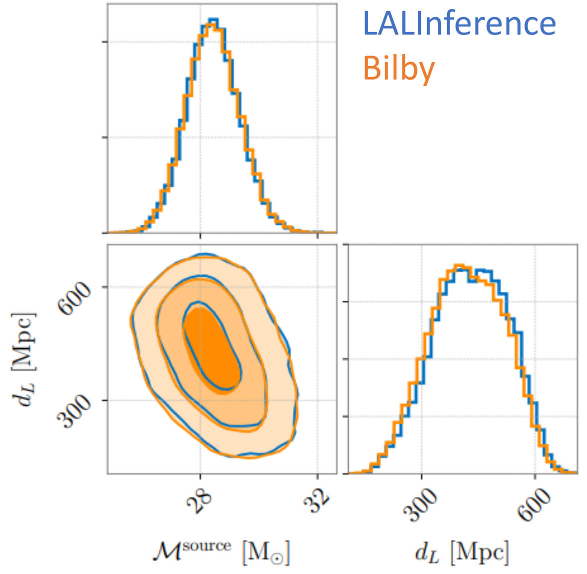
The Rosenbrock likelihood: [Hogg & Mackay 2018](#)

Three types of tests to validate inference software:

1. Standard inference problems
2. CBC inference problems
3. Parameter-Parameter tests

[Romero-Shaw et al. 2021](#) validated Bilby

1. Using the dynesty NS package
2. Demonstrated "statistically identical" to LALInference for CBC inference problems



Why we need to optimize

Typically, we need
$$n_L \sim 10^8 \rightarrow 10^9$$
evaluations of $\mathcal{L}(d|\theta,M)$

Meanwhile:
$$t_L \sim \text{ms} \rightarrow \text{s}$$

$\Rightarrow$ Wall time $= n_L \times t_L \sim$ days $\rightarrow$ years

For CBC systems containing a neutron star, EM observers want the best possible skymap ASAP!

Optimize to reduce the time to produce a skymap

Waveform models with "better physics" tend to be more computationally expensive

Optimize to enable the use of better waveforms

In O4, we will have hundreds of CBCs to analyze. Multiple analyses needed to investigate physics

Optimize to improve the results!

Optimization: the likelihood

Waveform approximant	per-likelihood evaluation [ms]	per-waveform evaluation [ms]
IMRPhenomPv2_NRTidal	93 ± 5	53 ± 4
IMRPhenomPv2	87 ± 6	47 ± 4
TaylorF2	60 ± 8	13.3 ± 0.7

Timings for a fiducial BNS signal [You, Ashton, et al. 2021]

The waveform tends to dominate

- For BBH typically more than 90%
- For BNS can be more like 50%
- More physics in the waveform increases t_ℓ
- Some waveforms take several seconds

Two approaches available to reduce t_ℓ:

- Reduced Order Quadrature [E.g., Smith et al. 2016, Morisaki & Raymond 2020]
- Hetrodyned likelihood [Cornish 2010] / Relative binning [Zackay et al. 2018]
- Both offer reductions in t_ℓ over over 1000
- No silver bullet solution to any waveform yet..

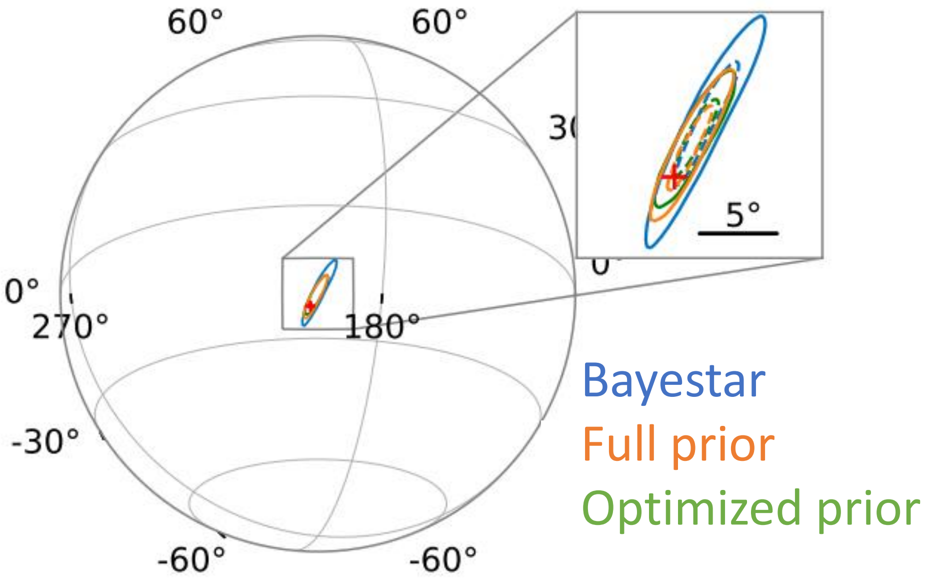
Optimization: priors

For astrophysical applications, priors should represent our prior belief

But, if we need a skymap as soon as possible:

- Optimize the prior:
  - For BNS/NSBH systems non-spinning and equal-mass can reduce wall-times by up to 50% (on top of other optimizations)

  - For BBH systems such optimization is unwise

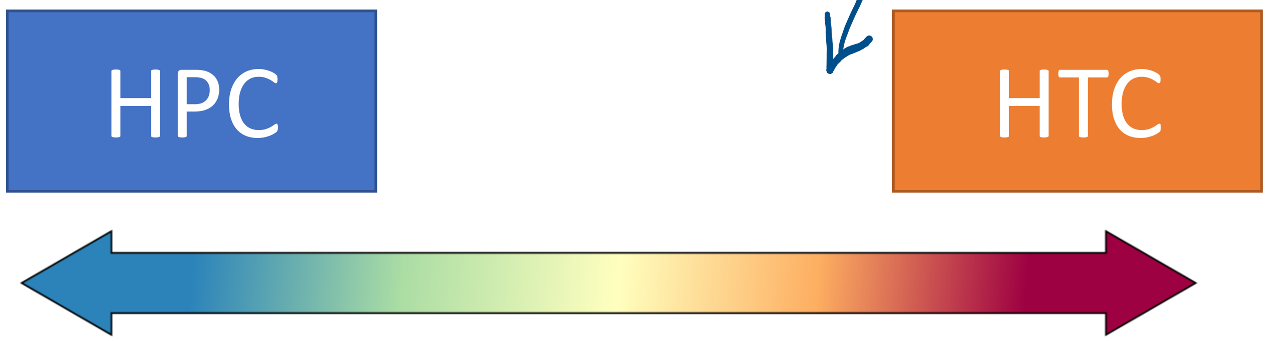


GW170817 skymap [You, Ashton, et al. 2021]

Optimization: parallelization I

- Two types of computing available
- High Performance Computing (HPC)
- High Throughput Computing (HTC)

LIGO Data Grid
Open Science Grid

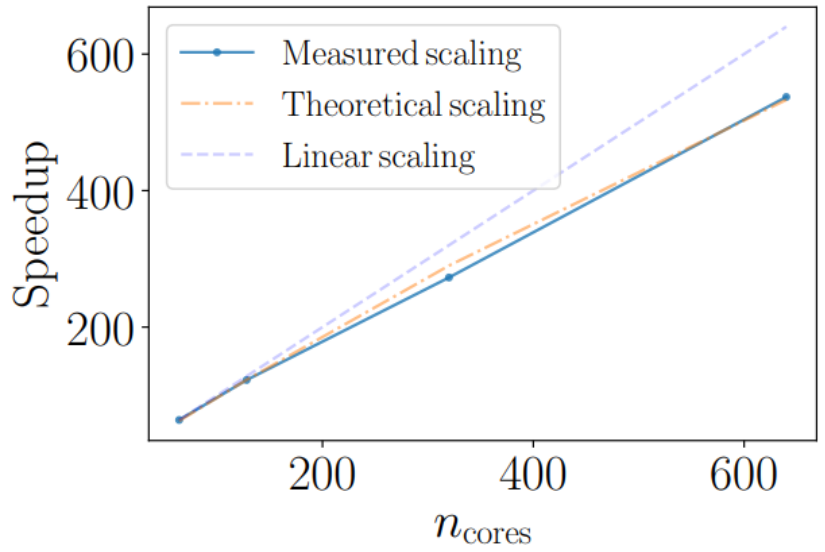


Fine-grained Applications
- Many-node
- Few concurrent runs
- High interconnect use

Course-grained Applications
- Single-node
- Many concurrent runs
- No interconnect use

Optimization: parallelization I

- An obvious route to speed things up is computational parallelization
- The nested sampling algorithm can be parallelized
- [Smith, Ashton et al. 2020] introduced parallel-Bilby
  - Uses MPI to leverage HPC environments
  - Achieved near-linear scaling
- For GW190412:
  - Using 16 cores (e.g. a standard analysis): ~3 years
  - Using 640 cores: 12 days
  - Leveraged an old "student" HPC environment



Optimization: parallelization II

- The LIGO Data Grid & Open Science Grid are predominantly HTC
- Moreover, it is difficult to organize taking over an entire cluster
- How did we do things before we had parallel-Bilby?

MCMC is trivially parallelizable

- Run N independent runs
- Each chain produces n samples
- Combine them together
- Used with great effect by LALInference MCMC
- Can be abused: there is a limit!

Nested is trivially parallelizable, but you can't stop it early

- Stopping criteria is "remaining evidence"
- Need to run each analyses to completion
- Unable to reduce wall time

Optimization: faster convergence

Can we reduce this?

Yes: improve the convergence

1. Use analytic marginalization of parts of the likelihood [See Thrane & Talbot 2019 for a review]

2. Improve the sampling efficiency:
   - Improve the proposal density (see, e.g. Williams et al. 2021 for ML approach)
   - CBC-specific jump proposals important for the LALInference MCMC sampler

Optimization: take-aways

- In O4, we could achieve posteriors for BNS/NSBH in $<$ 1hr
  - Main speed up comes from likelihood optimization fROQ/heterodyne
  - Further optimizations from waveform model / prior

- Modern inference packages lack a LALInference-like solution
  - Utilization of HTC parallelisation
  - Faster convergence from GW-specific jump proposals

Replacing LALInference MCMC?

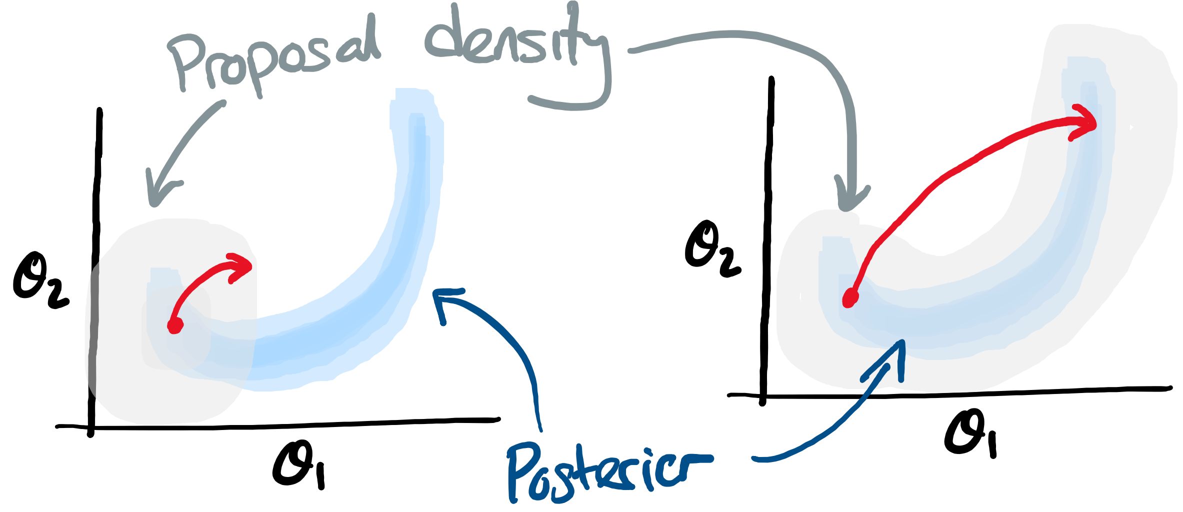
- Bilby enables access to several off-the-shelf MCMC packages
- None of them have been validated in Bilby

Sampler	Ensemble?	Parallel Tempering?	GW-tuned proposals?	Cross-check validated?
LALInfernce-MCMC	No	Yes	Yes	Yes
emcee	Yes	No	No	No
ptemcee	Yes	Yes	No	No*
Bilby-MCMC	**Yes****	**Yes**	**Yes**	**Yes**

- Bilby-MCMC [Ashton & Talbot 2021] was conceived to implement an MCMC sampler in Bilby with GW-tuned proposal

*ptemcee has been used with success by PyCBC-inference; **The ensemble approach was found to be ineffective

Bilby-MCMC



Improved proposals increase efficiency by 10-1000

- GW-tuned proposals
- Machine-Learning proposals inspired by [Williams et al. 2021]
  - Use past distribution to learn efficient proposal density
  - Normalizing Flows, Kernel Density Estimates, and Gaussian Mixture Models

Validated against dynesty and standard problems

Bilby-MCMC will enable the use of HTC parallelization

Sampling: take aways

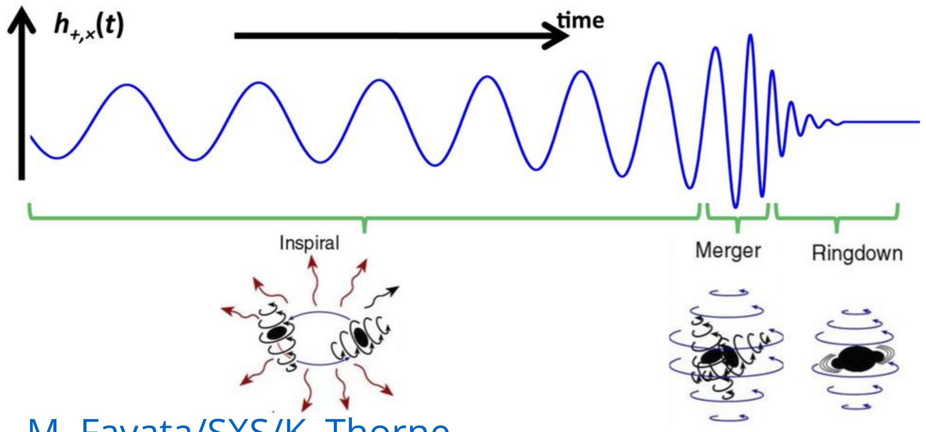
- Bilby now equipped with a NS and MCMC approach

- The high level flexible nature of Bilby offers the opportunity to prototype new ideas

- **In the rest of this talk: utilizing Bilby-MCMC to develop new approaches to modelling uncertainty**

Modelling uncertainty

- Numerical Relativity is our only means to model CBC mergers in GR
- Computationally infeasible to model the full signal
- Waveform models combine the inspiral, merger, and ringdown using different approximations
- This results in **systematic waveform uncertainty**
- Different waveforms make different predictions



M. Favata/SXS/K. Thorne

Modelling uncertainty: current approach

As of the GWTC3 catalogue, the current approach is:

1. Analyse each event with two waveform models
   - In practise, two different stochastic samplers are also applied

2. Compare between the waveform models to look for cases where they disagree

3. Combine equal numbers of samples from each waveform to produce a posterior which captures the uncertainty

Modelling uncertainty: improved approach

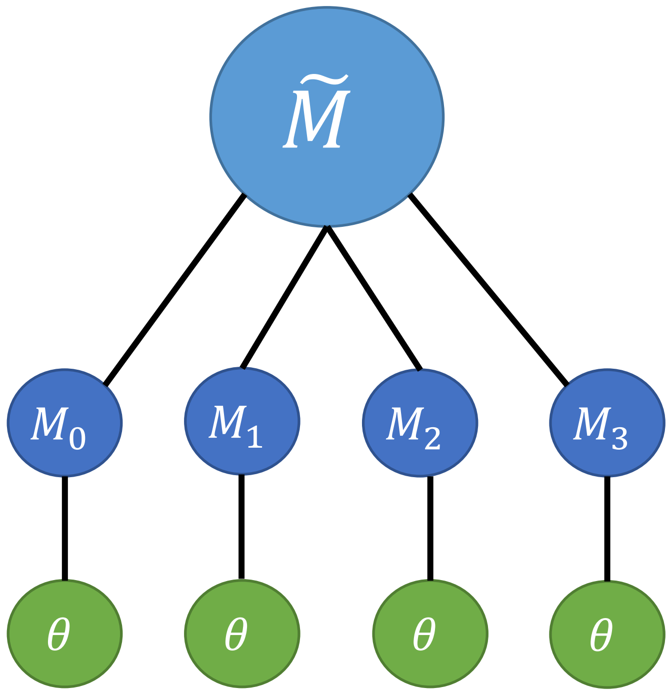
- The current approach neglects the evidence
- The evidence tells us how well each model explains the data
- Ashton & Khan 2020 demonstrate how to use the evidence to produce weighted posteriors

$$\mathcal{E}_i = \frac{\mathcal{Z}(d|M_i)}{\sum \mathcal{Z}_i}$$

- Pragmatic difficulties persist:
  - Relies on robust evidence estimates
  - Have both your analyses used identical data, likelihood, priors, models?

Modelling uncertainty: hypermodel approach

- Alternative to evidence-based approaches
- Define a hierarchical **hypermodel** $\widetilde{M} = \{M_0, M_1, \ldots, M_N\}$
- Calculate the posterior of $p(\theta |\ \text{data}, \widetilde{M})$
  - During sampling, first propose jumps between models
  - Then analyze the likelihood under the given waveform
  - Uses a special Reversible-Jump MCMC implemented in Bilby-MCMC

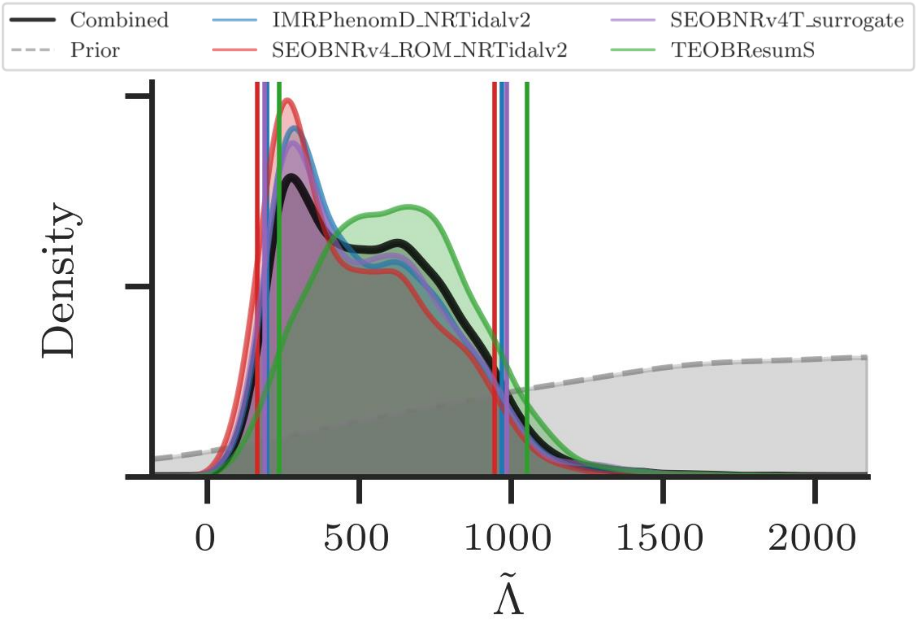


Benefits:

- "All in one" analysis approach
- Produces a posterior marginalized over $\widetilde{M}$
- Posteriors for individual waveforms can be "pulled out"
- Odds between models can be calculated

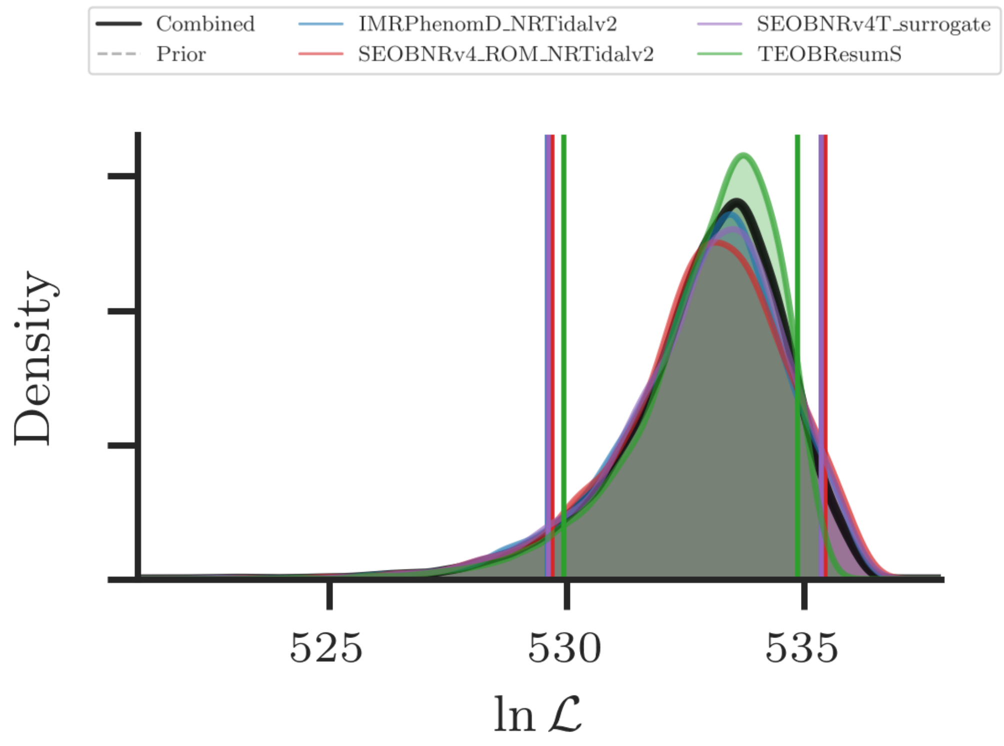
Applying hypermodels to observed BNS

- Ashton & Dietrich 2021: GW170817, GW190525,and GW200311_103121
- Four cutting-edge aligned-spin BNS models
- **Consistent preference TEOBResumS**
  - Bayesian odds range from 1.7 – 2.3
  - Not conclusive, but tantalizing
- Evidence suggests it is the tidal sector
  - Implications for BNS physics
  - Predictions of larger tidal deformability
- Combining the events in O4 will cement this result



Applying hypermodels to observed BNS

- TEOBResumS does not have the largest likelihood!
- It is the distribution which matters
- Important demonstration of why Bayesian approaches matter



Outlook

- Gravitational-wave inference is still a fast-evolving field
- Bilby provides an interface to develop new ideas
- In O4, we should reduce wall times by improved optimization
- Stochastic sampling remains the gold standard for analyses
  - Lots of work in Machine-Learning based approaches
  - These need to demonstrated to work out the box
  - See Chris's talk later today!

- My biggest concerns:
  - Waveform systematics studies need to be robust
  - Non-Gaussian noise (not discussed here)