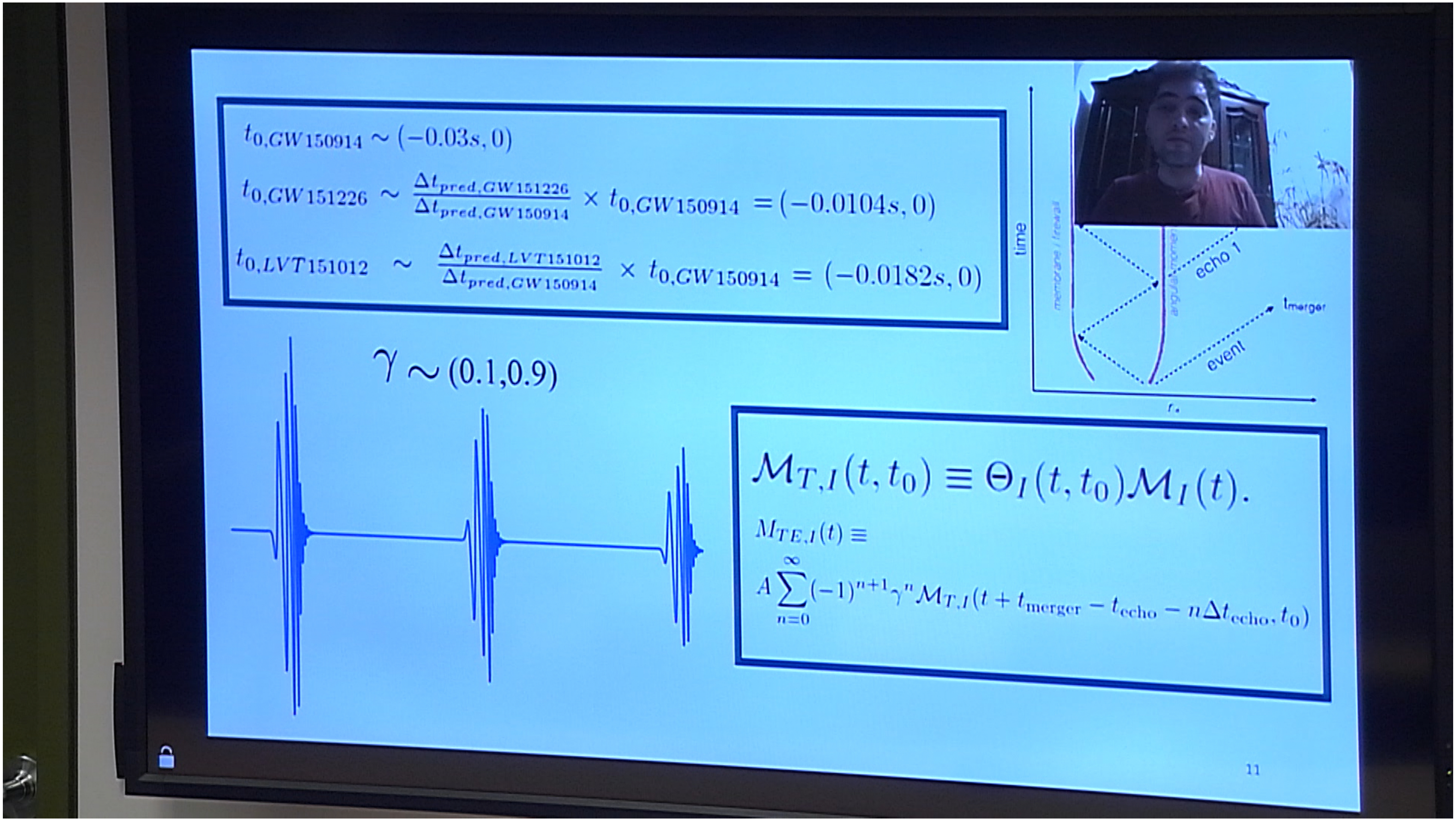
Title: Bayesian Evidence for Echoes

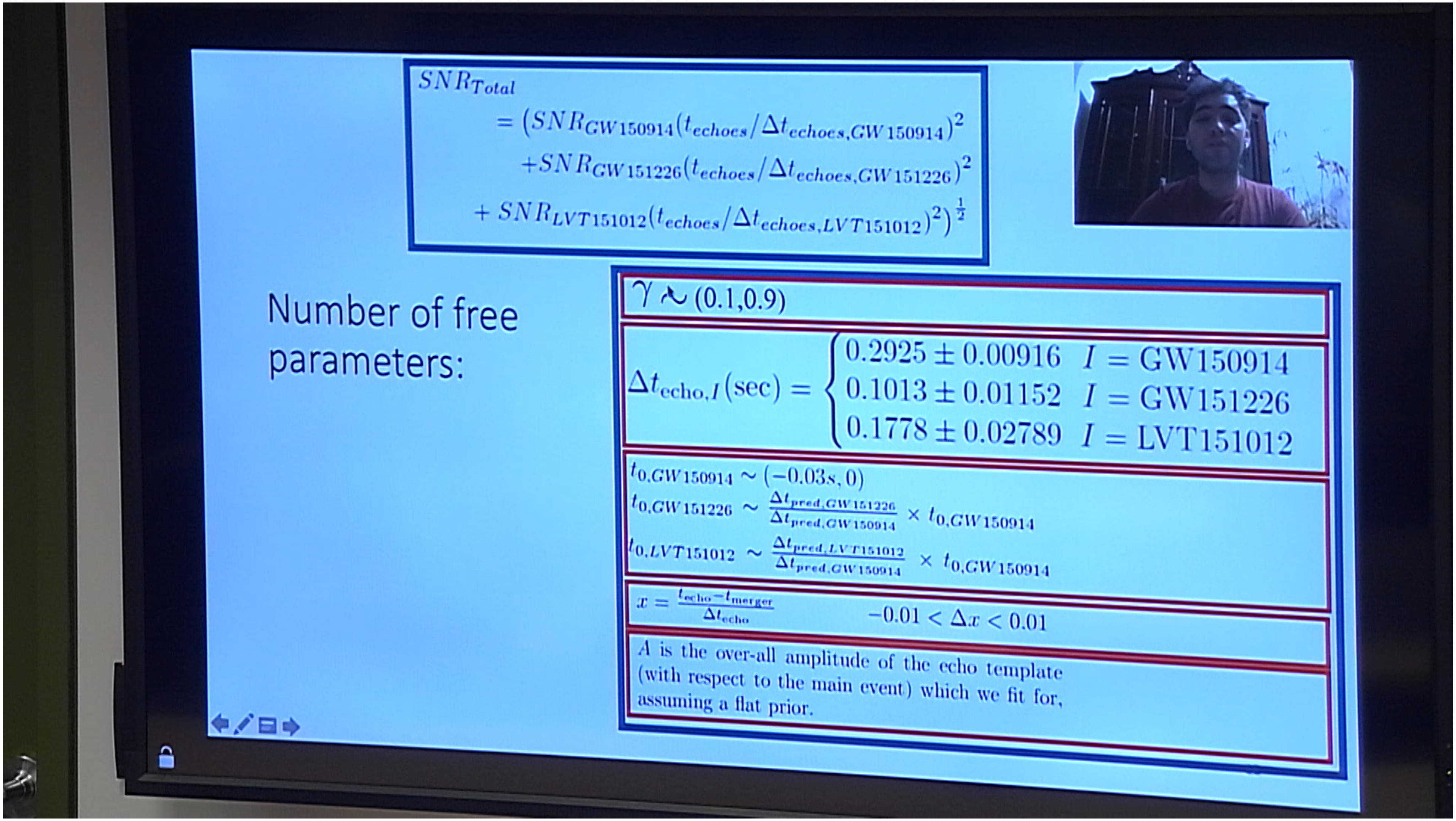
Speakers: Jahed Abedi

Collection: Echoes in Southern Ontario

Date: February 25, 2020 - 9:45 AM

URL: http://pirsa.org/20020097


$t_{0,GW150914} \sim (-0.03s, 0)$
$t_{0,GW151226} \sim \frac{\Delta t_{pred,GW151226}}{\Delta t_{pred,GW150914}} \times t_{0,GW150914} = (-0.0104s, 0)$
$t_{0,LVT151012} \sim \frac{\Delta t_{pred,LVT151012}}{\Delta t_{pred,GW150914}} \times t_{0,GW150914} = (-0.0182s, 0)$
$\gamma \sim (0.1, 0.9)$
time
echo 1
t_{merger}
event
$\mathcal{M}_{T,I}(t,t_0) \equiv \Theta_I(t,t_0)\mathcal{M}_I(t).$
$M_{TE,I}(t) \equiv$
$A\sum_{n=0}^{\infty}(-1)^{n+1}\gamma^n \mathcal{M}_{T,I}(t + t_{\rm merger} - t_{\rm echo} - n\Delta t_{\rm echo}, t_0)$
11


SNR_{Total}
$= (SNR_{GW150914}(t_{echoes}/\Delta t_{echoes,GW150914})^2$
$+SNR_{GW151226}(t_{echoes}/\Delta t_{echoes,GW151226})^2$
$+ SNR_{LVT151012}(t_{echoes}/\Delta t_{echoes,LVT151012})^2)^{\frac{1}{2}}$
Number of free parameters:
$\gamma \sim (0.1, 0.9)$
$\Delta t_{\mathrm{echo},I}(\mathrm{sec}) = \begin{cases} 0.2925 \pm 0.00916 & I = \mathrm{GW150914} \\ 0.1013 \pm 0.01152 & I = \mathrm{GW151226} \\ 0.1778 \pm 0.02789 & I = \mathrm{LVT151012} \end{cases}$
$t_{0,GW150914} \sim (-0.03s, 0)$
$t_{0,GW151226} \sim \frac{\Delta t_{pred,GW151226}}{\Delta t_{pred,GW150914}} \times t_{0,GW150914}$
$t_{0,LVT151012} \sim \frac{\Delta t_{pred,LVT151012}}{\Delta t_{pred,GW150914}} \times t_{0,GW150914}$
$x = \frac{t_{\mathrm{echo}} - t_{\mathrm{merger}}}{\Delta t_{\mathrm{echo}}}$ $-0.01 < \Delta x < 0.01$
A is the over-all amplitude of the echo template (with respect to the main event) which we fit for, assuming a flat prior.

- Have a good physical model! (or you won’t find them)
- and a simple template (avoid too many arbitrary choices)
- Avoid a posteriori statistics (don’t look at data to make your template)

So far seven independent groups looked echoes in GW data

Positive results

Abedi, et al. 2017-18; Conklin, et al. 2018, Uchikata, et al. 2019

Negative results

Lo, et al. 2019; *Tsnag, et al. 2019; Uchikata, et al. 2019*

Mixed results

Westerweck, et al. 2018; Nielsen, et al. 2019 ; Salemi, et al. 2019

Bayes' theorem:

From the product rule of probability it can be shown that

$$P(A\ and\ B) \equiv P(A,B) = P(A)\,P(B|A) = P(B)\,P(A|B)$$

$P(x|y)$ means "the probability that x is true given y is true"

this gives:

$$P(B|A,I) = \frac{P(A|B,I)P(B|I)}{P(A|I)}$$

This is Bayes' theorem.

Bayes' theorem:

Bayes theorem can be cast in terms of a model, or hypothesis, and some observations, or data.

It tells us how to update our degree of belief about our model based on new data.

$$\underbrace{P(model|data)}_{\text{Posterior}} = \frac{\overbrace{P(data|model)}^{\text{Likelihood}}\overbrace{P(model)}^{\text{Prior}}}{P(data)}$$

Prior: what we knew, or our degree of belief, about our model before taking data

Likelihood: the influence of the data in updating our degree of belief

Posterior: our new degree of belief about our model in light of the data

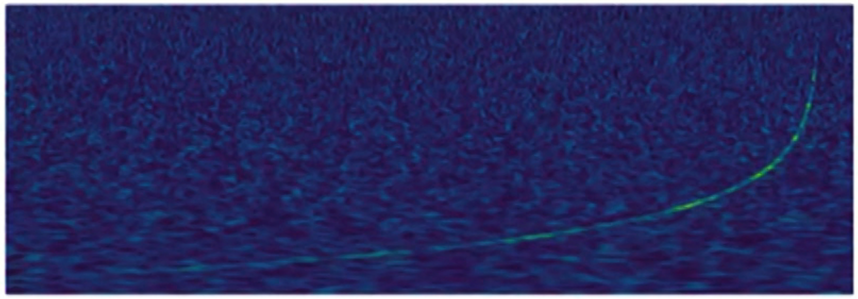
PyCBC

DOCUMENTATION TUTORIALS PUB

PyCBC

Free and open software to study gravitational waves.

https://pycbc.org



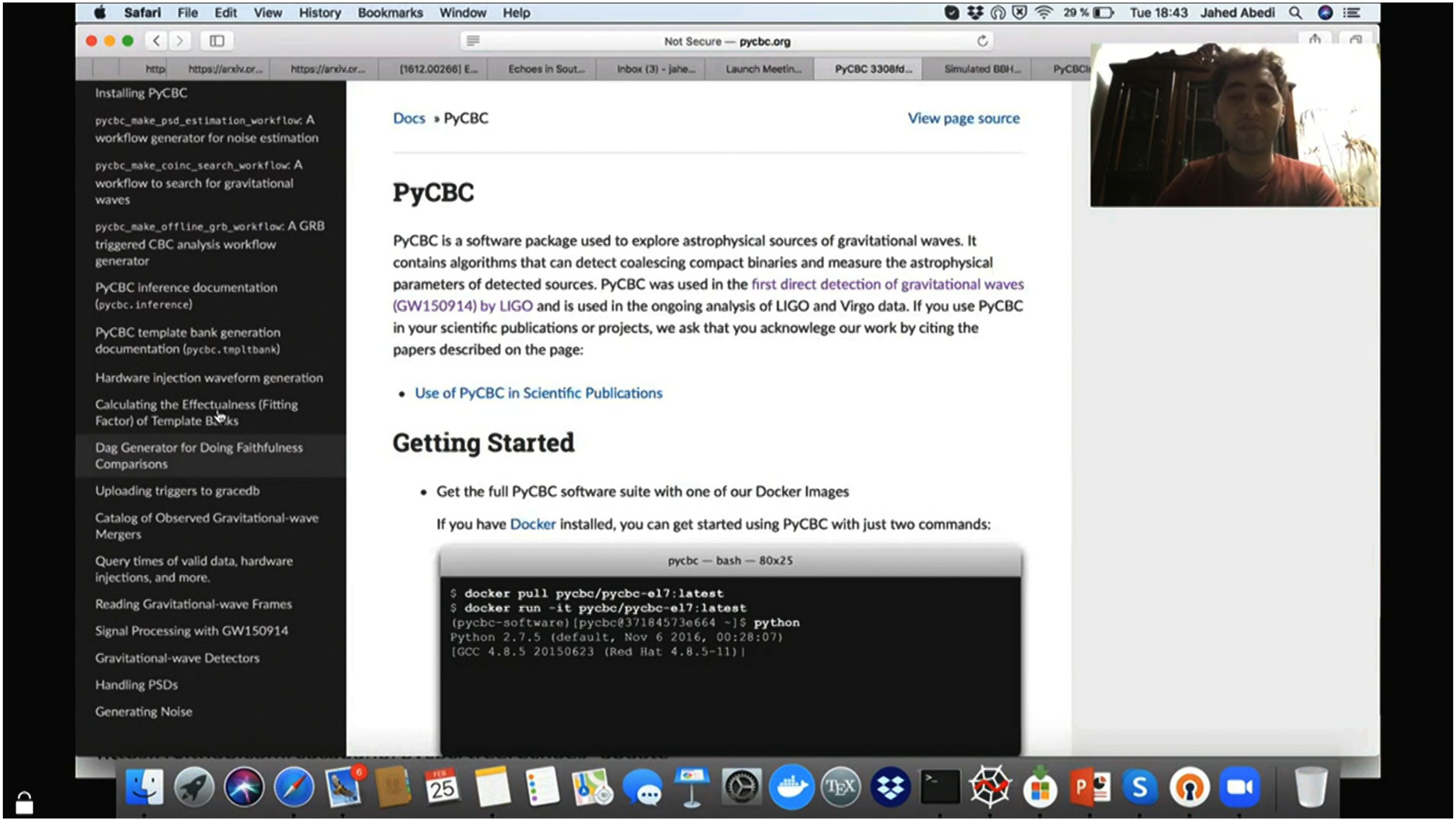
PyCBC is a software package used to explore astrophysical sources of gravitational waves. It contains algorithms that can detect coalescing compact binaries and measure the astrophysical parameters of detected sources. PyCBC was used in the first direct detection of gravitational waves by LIGO and is used in the ongoing analysis of LIGO and Virgo data. PyCBC was featured in Physics World as a good example of a large collaboration publishing its research products, including its software.

Getting Started

You can start using the PyCBC library now in an interactive notebook! See our tutorials

https://arxiv.org/pdf/1807.10312.pdf

https://github.com/cdcapano/pycbc/tree/echoes_update


Safari File Edit View History Bookmarks Window Help
29 % Tue 18:43 Jahed Abedi
Not Secure — pycbc.org
Installing PyCBC
pycbc_make_psd_estimation_workflow: A workflow generator for noise estimation
pycbc_make_coinc_search_workflow: A workflow to search for gravitational waves
pycbc_make_offline_grb_workflow: A GRB triggered CBC analysis workflow generator
PyCBC inference documentation (pycbc.inference)
PyCBC template bank generation documentation (pycbc.tmpltbank)
Hardware injection waveform generation
Calculating the Effectualness (Fitting Factor) of Template Banks
Dag Generator for Doing Faithfulness Comparisons
Uploading triggers to gracedb
Catalog of Observed Gravitational-wave Mergers
Query times of valid data, hardware injections, and more.
Reading Gravitational-wave Frames
Signal Processing with GW150914
Gravitational-wave Detectors
Handling PSDs
Generating Noise
Docs » PyCBC
View page source
PyCBC
PyCBC is a software package used to explore astrophysical sources of gravitational waves. It contains algorithms that can detect coalescing compact binaries and measure the astrophysical parameters of detected sources. PyCBC was used in the first direct detection of gravitational waves (GW150914) by LIGO and is used in the ongoing analysis of LIGO and Virgo data. If you use PyCBC in your scientific publications or projects, we ask that you acknowlege our work by citing the papers described on the page:
• Use of PyCBC in Scientific Publications
Getting Started
• Get the full PyCBC software suite with one of our Docker Images
If you have Docker installed, you can get started using PyCBC with just two commands:
pycbc — bash — 80x25
$ docker pull pycbc/pycbc-el7:latest
$ docker run -it pycbc/pycbc-el7:latest
(pycbc-software)[pycbc@37184573e664 ~]$ python
Python 2.7.5 (default, Nov 6 2016, 00:28:07)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-11)]

easy to read, we will split the data, sampler, and prior settings into their own configuration files.

Here are the model and prior settings we will use:

```
[model]
name = gaussian_noise
low-frequency-cutoff = 20.0

[variable_params]
; waveform parameters that will vary in MCMC
delta_tc =
mass1 =
mass2 =
spin1_a =
spin1_azimuthal =
spin1_polar =
spin2_a =
spin2_azimuthal =
spin2_polar =
distance =
coa_phase =
inclination =
polarization =
ra =
dec =

[static_params]
; waveform parameters that will not change in MCMC
approximant = IMRPhenomPv2
f_lower = 20
f_ref = 20
; we'll set the tc by using the trigger time in the data
; section of the config file + delta_tc
trigger_time = ${data|trigger-time}

[prior-delta_tc]
; coalescence time prior
name = uniform
min-delta_tc = -0.1
max-delta_tc = 0.1

[waveform_transforms-tc]
; we need to provide tc to the waveform generator
name = custom
```


Hardware injection waveform generation

Calculating the Effectualness (Fitting Factor) of Template Banks

Dag Generator for Doing Faithfulness Comparisons

Uploading triggers to gracedb

Catalog of Observed Gravitational-wave Mergers

Query times of valid data, hardware injections, and more.

Reading Gravitational-wave Frames

Signal Processing with GW150914

Gravitational-wave Detectors

Handling PSDs

Generating Noise

Waveforms

Filtering

Using PyCBC Distributions from PyCBC Inference

Building Bundled Executables

Documenting PyCBC code

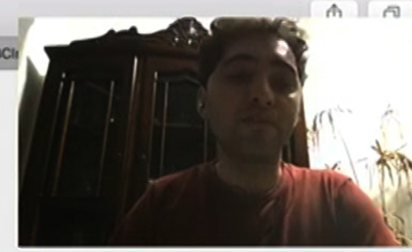
Creating Releases of PyCBC

HDF files within the PyCBC workflow

Workflow: the inspiral analysis workflow generator (pycbc.workflow)

pycbc

```
[model]
name = gaussian_noise
low-frequency-cutoff = 20.0

[variable_params]
; waveform parameters that will vary in MCMC
delta_tc =
mass1 =
mass2 =
spin1_a =
spin1_azimuthal =
spin1_polar =
spin2_a =
spin2_azimuthal =
spin2_polar =
distance =
coa_phase =
inclination =
polarization =
ra =
dec =

[static_params]
; waveform parameters that will not change in MCMC
approximant = IMRPhenomPv2
f_lower = 20
f_ref = 20
; we'll set the tc by using the trigger time in the data
; section of the config file + delta_tc
trigger_time = ${data|trigger-time}

[prior-delta_tc]
; coalescence time prior
name = uniform
min-delta_tc = -0.1
max-delta_tc = 0.1

[waveform_transforms-tc]
; we need to provide tc to the waveform generator
name = custom
inputs = delta_tc
tc = ${data|trigger-time} + delta_tc

[prior-mass1]
name = uniform
```



Safari File Edit View History Bookmarks Window Help

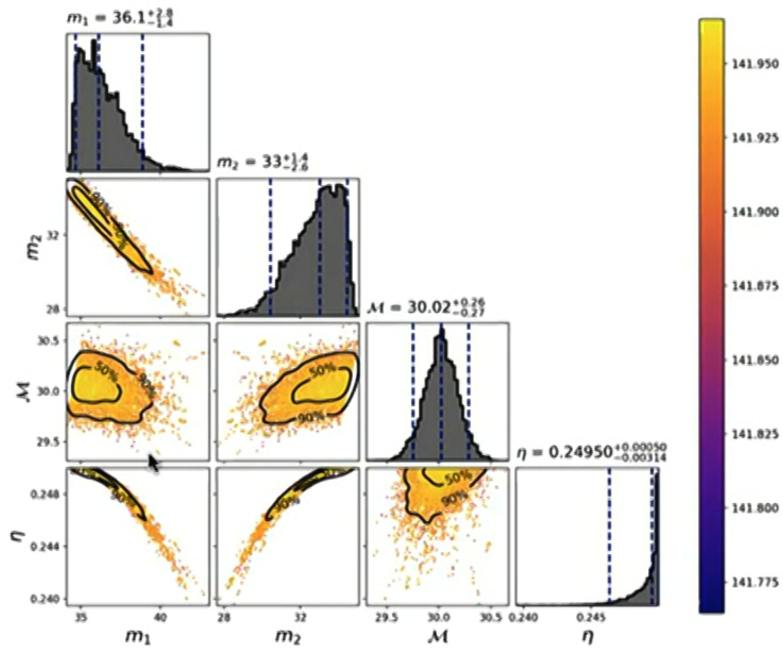
Not Secure — pycbc.org

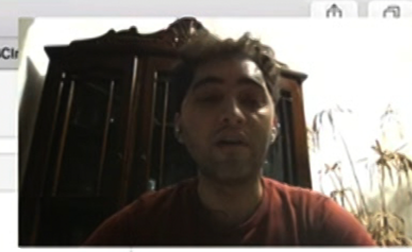

```
[prior-delta_tc]
; coalescence time prior
name = uniform
min-delta_tc = -0.1
max-delta_tc = 0.1

[waveform_transforms-tc]
; we need to provide tc to the waveform generator
name = custom
inputs = delta_tc
tc = ${data|trigger-time} + delta_tc

[prior-mass1]
name = uniform
min-mass1 = 10.
max-mass1 = 80.

[prior-mass2]
name = uniform
min-mass2 = 10.
max-mass2 = 80.

[prior-spin1_a]
name = uniform
min-spin1_a = 0.0
max-spin1_a = 0.99

[prior-spin1_polar+spin1_azimuthal]
name = uniform_solidangle
polar-angle = spin1_polar
azimuthal-angle = spin1_azimuthal

[prior-spin2_a]
name = uniform
min-spin2_a = 0.0
max-spin2_a = 0.99

[prior-spin2_polar+spin2_azimuthal]
name = uniform_solidangle
polar-angle = spin2_polar
azimuthal-angle = spin2_azimuthal

[prior-distance]
; following gives a uniform volume prior
name = uniform_radius
min-distance = 10
```

```
--plot-contours \
--plot-marginal \
--z-arg snr
```

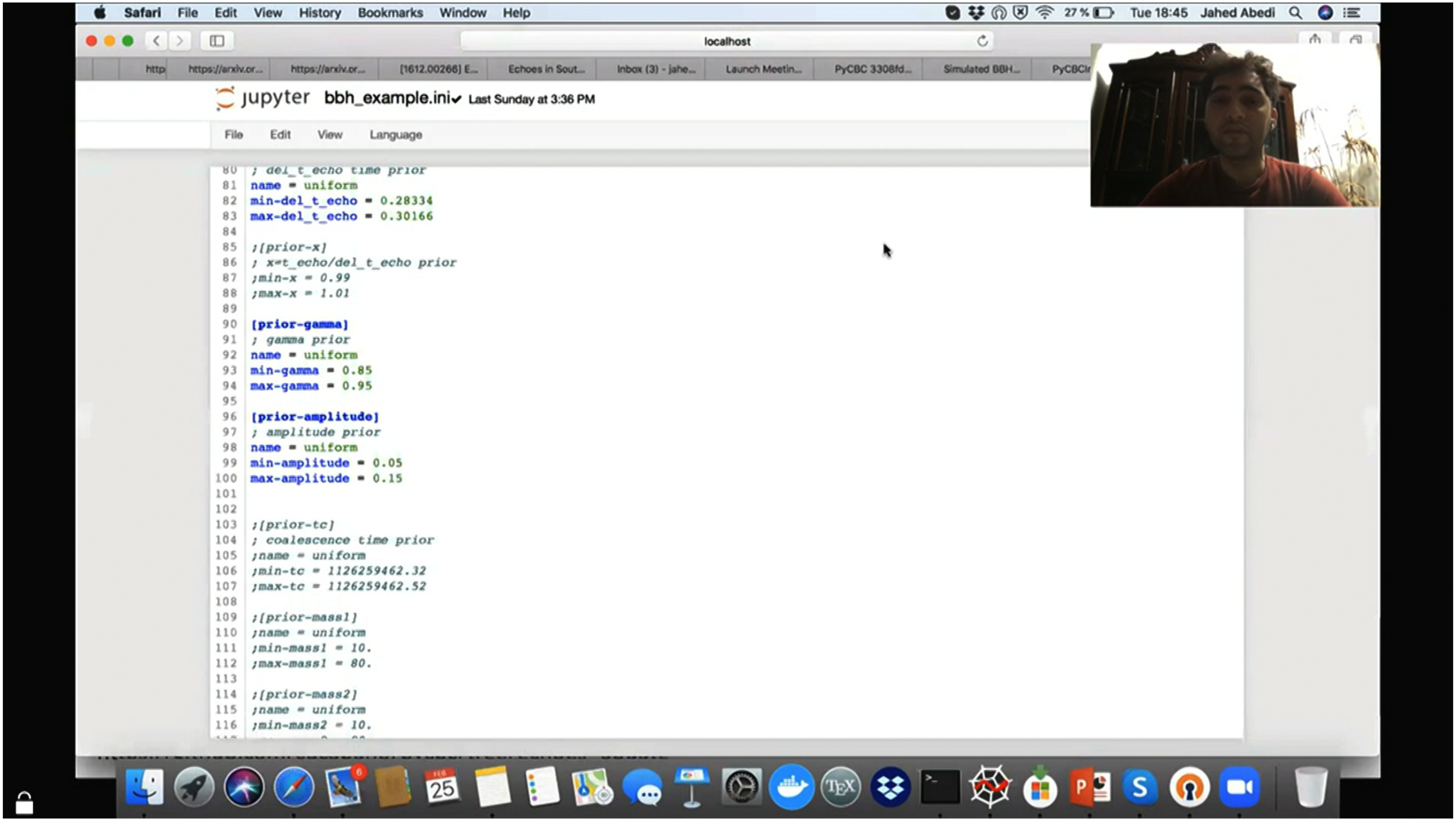
```
In [44]: Image(filename='posterior_mass1_mass2.png', width=640, height=480)
```

Out[44]:





Challenge:


Safari File Edit View History Bookmarks Window Help
27 % Tue 18:45 Jahed Abedi
localhost
jupyter bbh_example.ini Last Sunday at 3:36 PM
File Edit View Language
name = uniform
min-del_t_echo = 0.28334
max-del_t_echo = 0.30166

;[prior-x]
; x=t_echo/del_t_echo prior
;min-x = 0.99
;max-x = 1.01

[prior-gamma]
; gamma prior
name = uniform
min-gamma = 0.85
max-gamma = 0.95

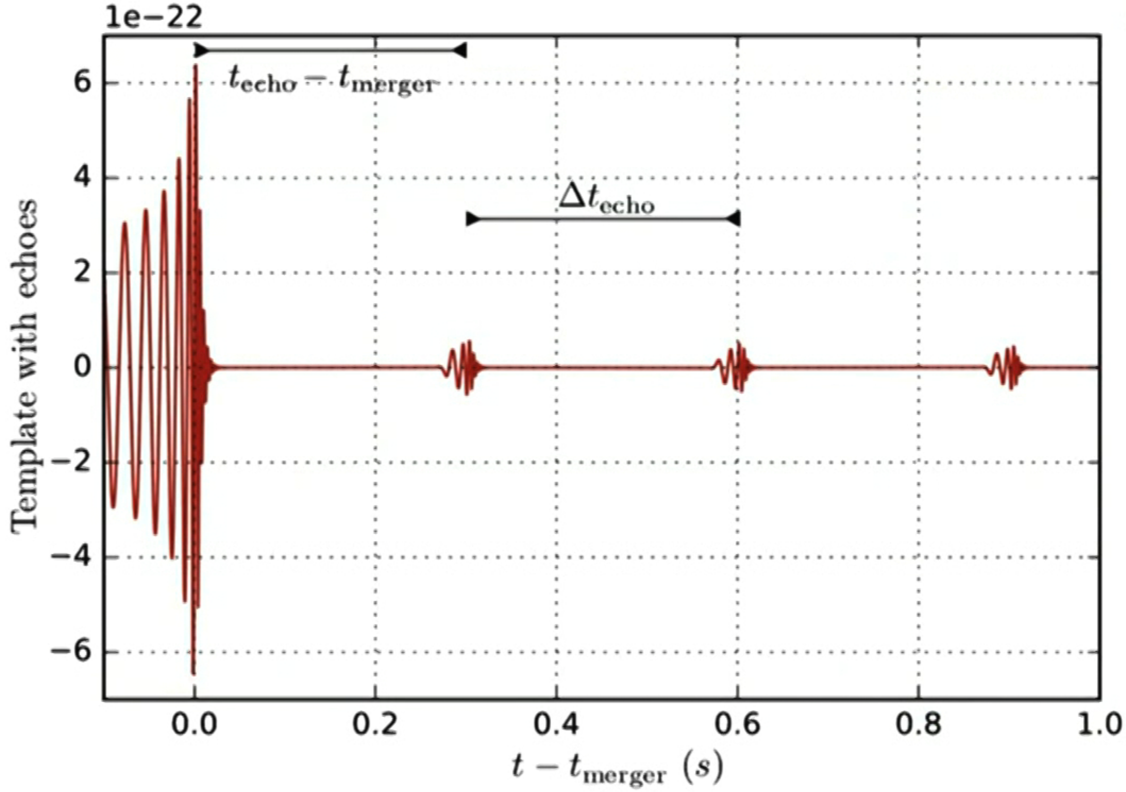
[prior-amplitude]
; amplitude prior
name = uniform
min-amplitude = 0.05
max-amplitude = 0.15

;[prior-tc]
; coalescence time prior
;name = uniform
;min-tc = 1126259462.32
;max-tc = 1126259462.52

;[prior-mass1]
;name = uniform
;min-mass1 = 10.
;max-mass1 = 80.

;[prior-mass2]
;name = uniform
;min-mass2 = 10.

Parameter estimation of ech



Parameter estimation and statistical significance of echoes fol hole signals in the first Advanced LIGO observing

Alex B. Nielsen,[1,2,*] Collin D. Capano,[1,2,†] Ofek Birnholtz,[3,‡] and Julian W

[1]*Max-Planck-Institut für Gravitationsphysik, D-30167 Hannover, Germany*
[2]*Leibniz Universität Hannover, D-30167 Hannover, Germany*
[3]*Center for Computational Relativity and Gravitation, Rochester Institute of Technology, 170 Lomb Memorial Drive, Rochester, New York 14623, USA*

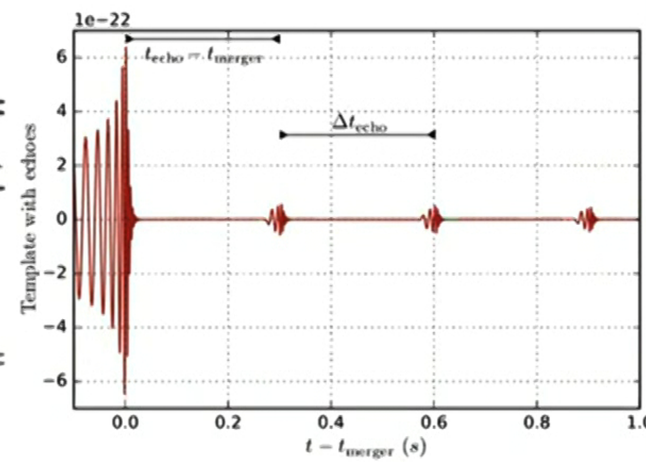
(Received 19 February 2019; published 7 May 2019)

Searching for black hole echo signals with gravitational waves provides a means of probing the near-horizon regime of these objects. We demonstrate a pipeline to efficiently search for these signals in gravitational wave data and calculate model selection probabilities between signal and no-signal hypotheses. As an example of its use we calculate Bayes factors for the Abedi-Dykaar-Afshordi (ADA) model on events in LIGO's first observing run and compare to existing results in the literature. We discuss the benefits of using a full likelihood exploration over existing search methods that used template banks and calculated p-values. We use the waveforms of ADA, although the method is easily extendable to other waveforms. With these waveforms we are able to demonstrate a range of echo amplitudes that is already is ruled out by the data.

DOI: 10.1103/PhysRevD.99.104012

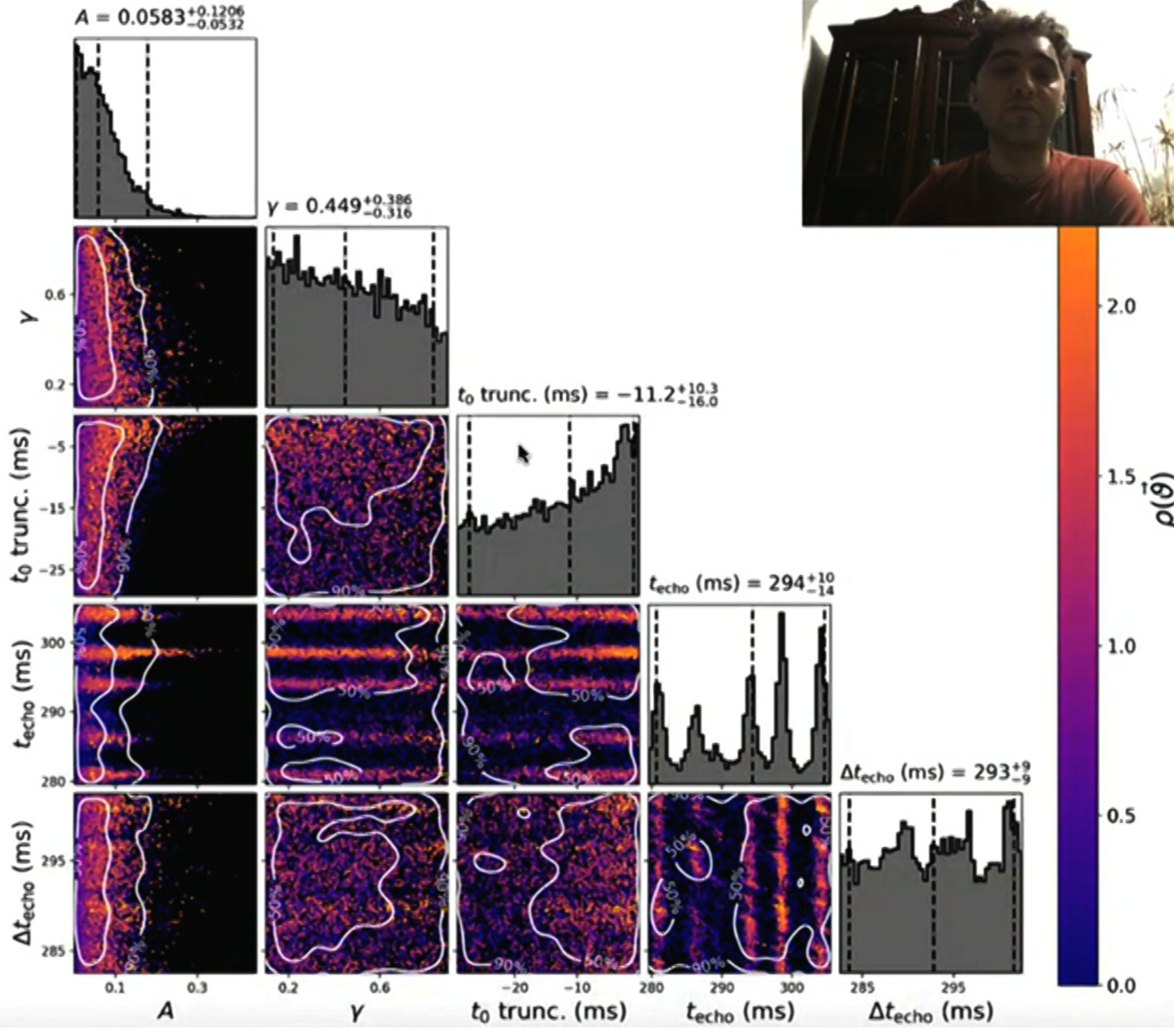
Echo param.	Prior range	GW150914 ra
Δt_{echo}	Inferred	0.2825 to 0.3025 s
t_{echo}	$\Delta t_{echo} \pm 1\%$	0.2795 to 0.3055 s
t_0 trunc.	$(-0.1 \text{ to } 0)\Delta t_{echo}$	−0.02925 to 0 s
γ	0.1 to 0.9	0.1 to 0.9
A	Unconstrained	0.00001 to 0.9

Event	Log Bayes factor	Max SNR
GW150914	−1.8056	2.86
GW151012	1.2499	5.57
GW151226	0.4186	4.07



Results of Nielsen et al. Phys. Rev. D 99, 104012 (2019) for GW150914 are showing preference for noise

Log Bayes factor ≃ -1.8



Comment on Nielsen et al. Phys. Rev. D 99, 10401

Comparing two papers of the same group Westerweck et al. (Phys. Rev. D 97, 124037 (2018)) and Nielsen et al. (Phys. Rev. D 99, 104012 (2019)) it is puzzling that why max SNR that is reported in Phys. Rev. D 97, 124037 (2018) for GW151012 and GW151225 is roughly consistent with what was found by Abedi, Dykaar & Afshordi (Phys. Rev. D 96, 082004 (2017)). However, the max SNR found for GW150914 is, much smaller than what we found (4.13), or they found in (Phys. Rev2.86. D 97, 124037 (2018)).

Event	Log Bayes factor	Max SNR
GW150914	−1.8056	2.86
GW151012	1.2499	5.57
GW151226	0.4186	4.07

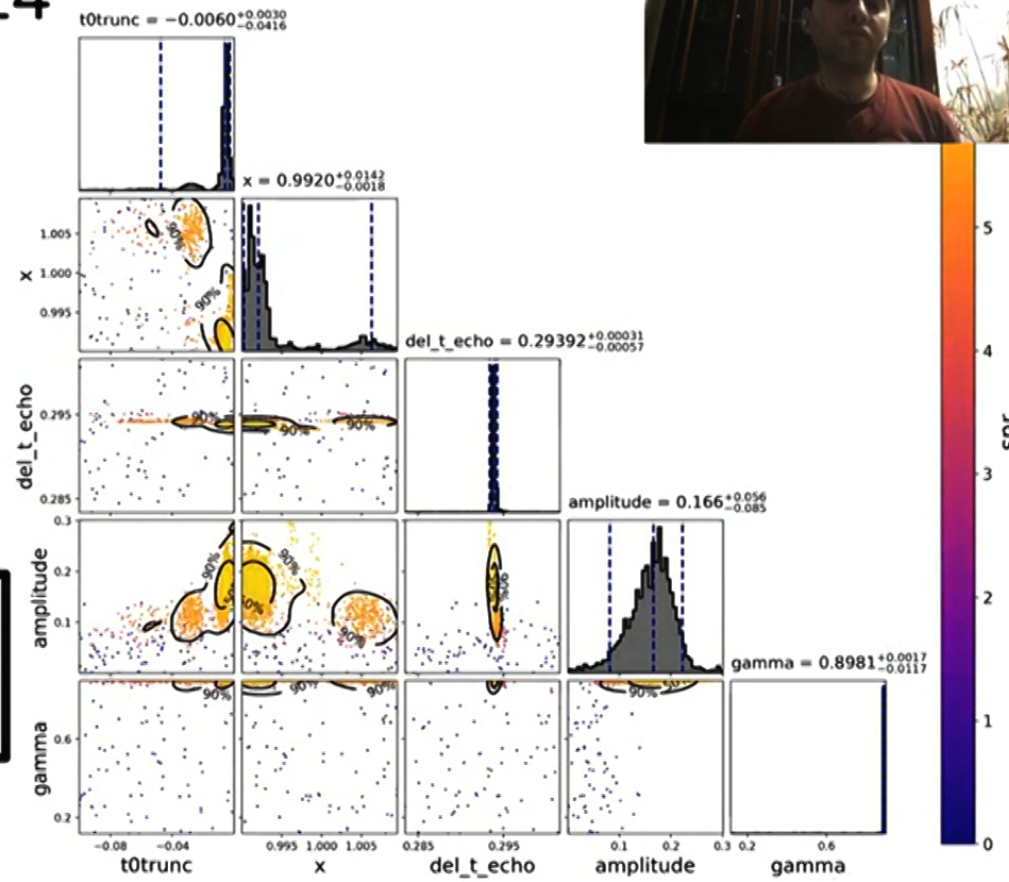
Our Results for GW150914

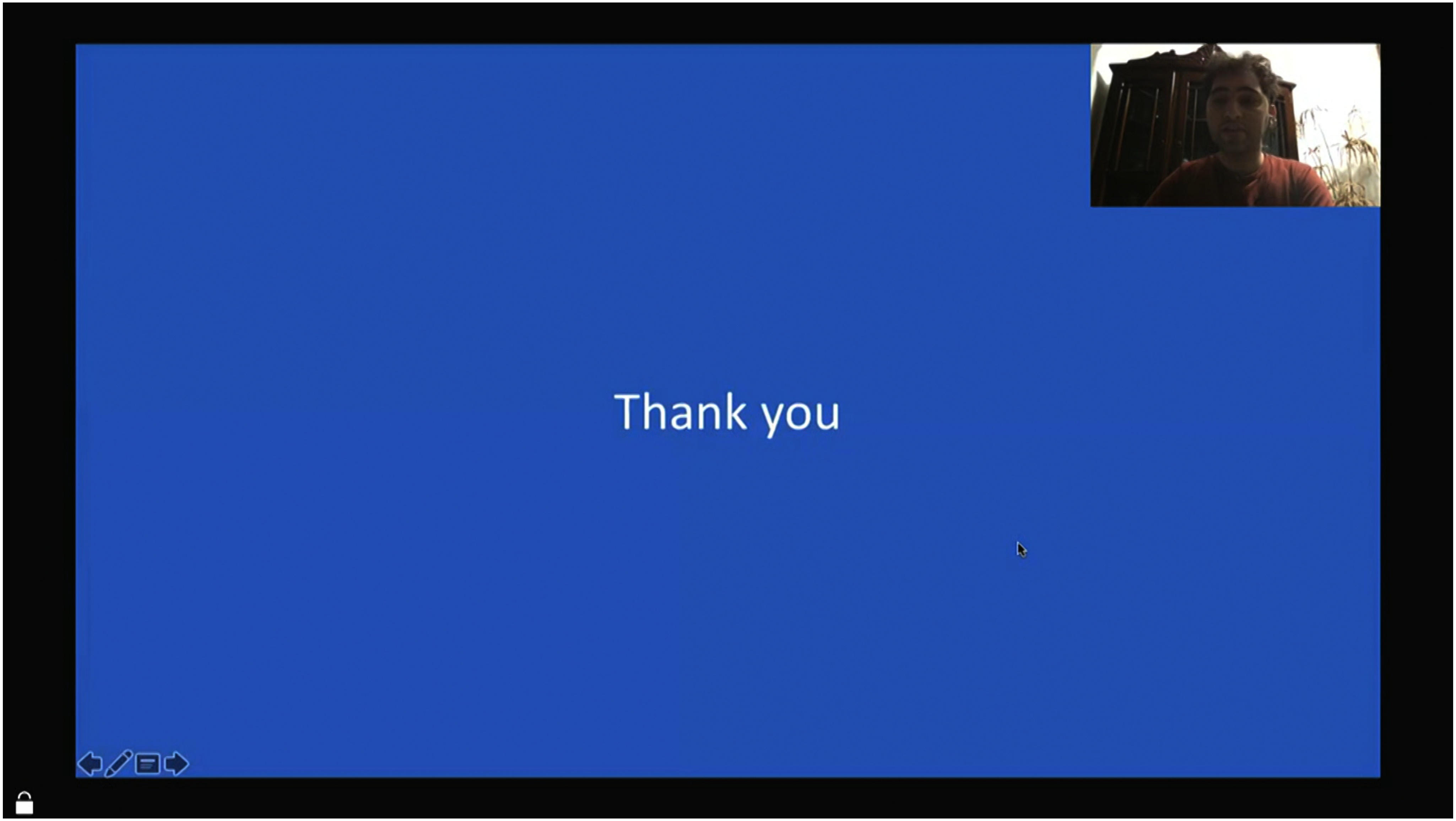
Log Bayes factor ≃ 3

We are puzzled by why results of Nielsen et al. are different, given that we are actually using their code!!!

Already, their max SNR in their own two papers are not consistent!!

Preference for signal!!!
Strong evidence




Thank you