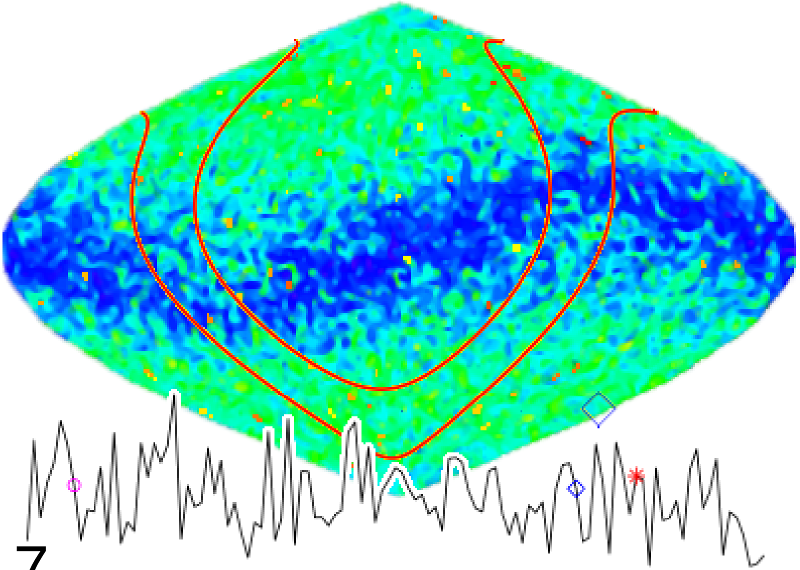
ALBERT EINSTEIN INSTITUTE HANNOVER
Max Planck Institute for Gravitational Physics and
Leibniz Universität Hannover

Leibniz
Universität
Hannover

CONTINUOUS
GRAVITATIONAL
WAVES

Continuous gravitational wave atlas.

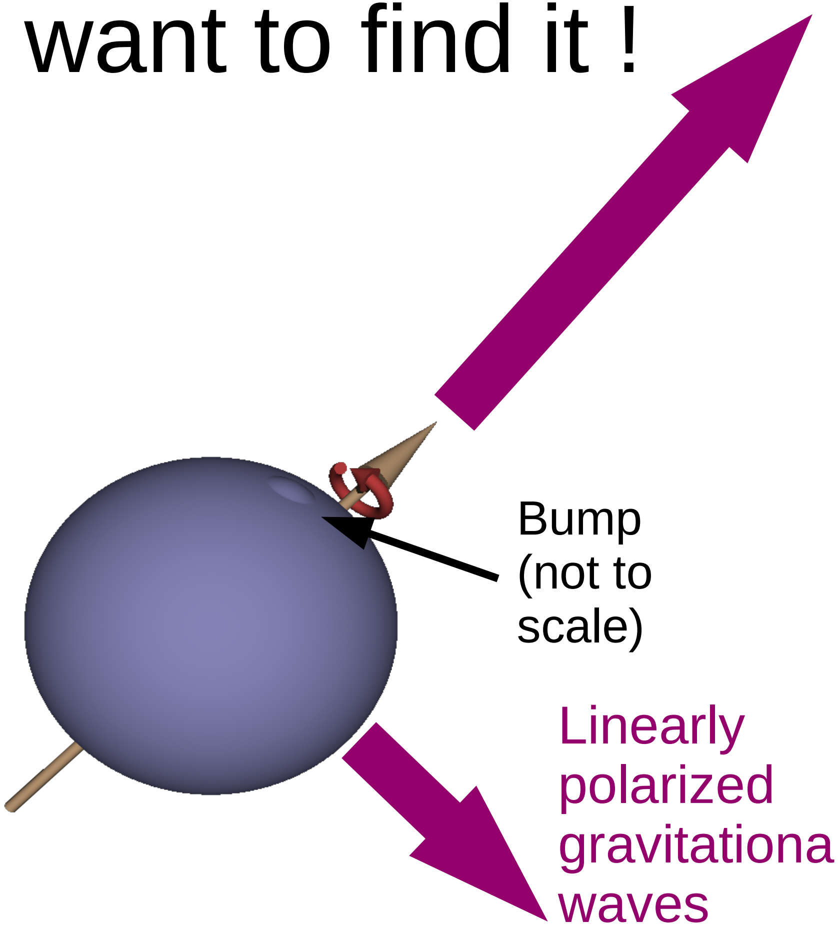
Vladimir Dergachev
Max Planck/AEI Hannover



Hannover, 2024-Jun-17

Somewhere far away there is a neutron star...

We want to find it !



Circularly polarized gravitational waves

Bump (not to scale)

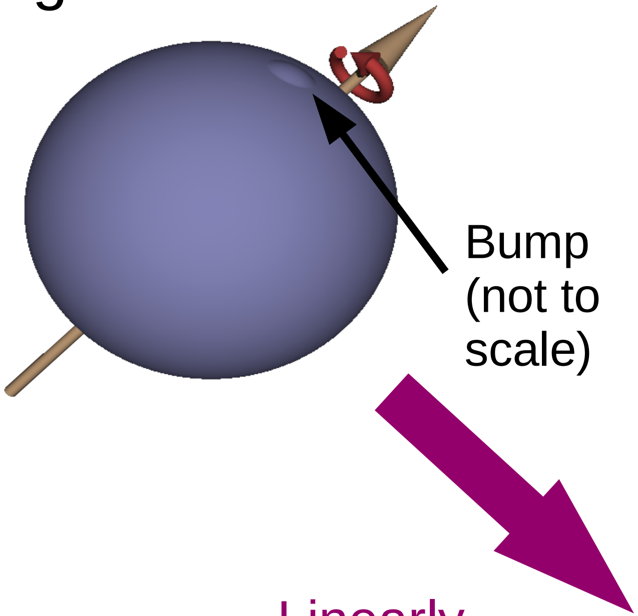
Linearly polarized gravitational waves

- Hard, computationally intensive problem
- Small parameter – equatorial deformation of neutron star ε
- Sensitivity scales as (coherence length)$^{-0.25}$(frequency)^{2} and is proportional to ε
- Computing time scales as (coherence length)^{4}(frequency)^{3} or faster

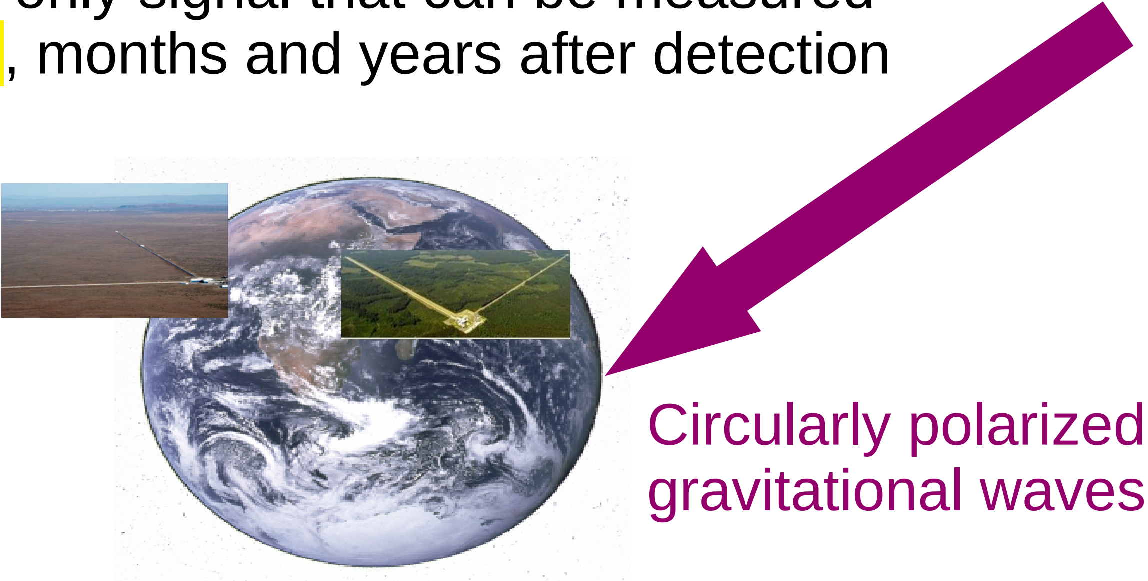
Continuous gravitational waves

- Need a rotating star with non-zero equatorial second moment
- Gravitational radiation is expected to be emitted at twice the rotation frequency
- Continuous wave signals have very narrow bandwidth
- The only signal that can be measured again, months and years after detection

Rotating neutron star



Linearly polarized gravitational waves

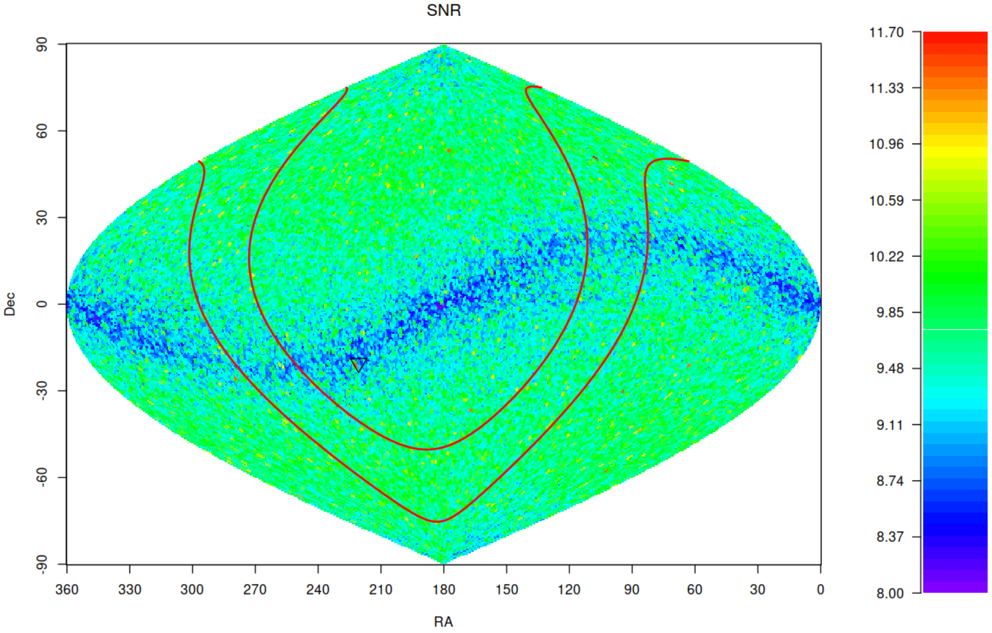


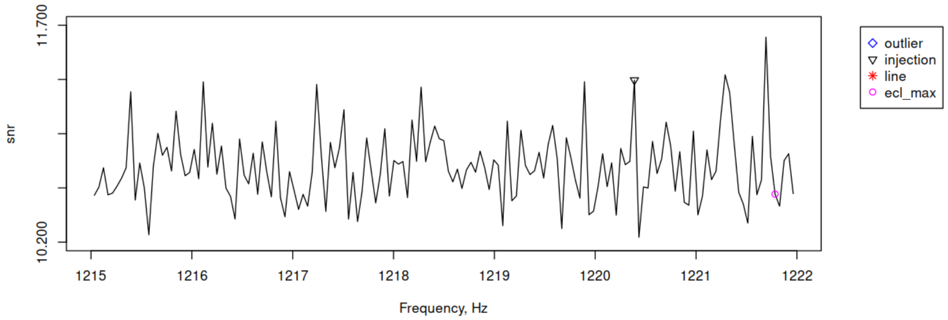
Circularly polarized gravitational waves

Continuous gravitational wave atlas

- Instant search on your notebook !
- Robust results for the entire sky and all covered frequencies, no exclusions.
- First atlas released 2022-02-22, covering 500-1000 Hz
- Latest atlas released 2023-11-16, covering 20-1500 Hz
- There is also Gaia DR3 data in the same format as the atlas for easy analysis

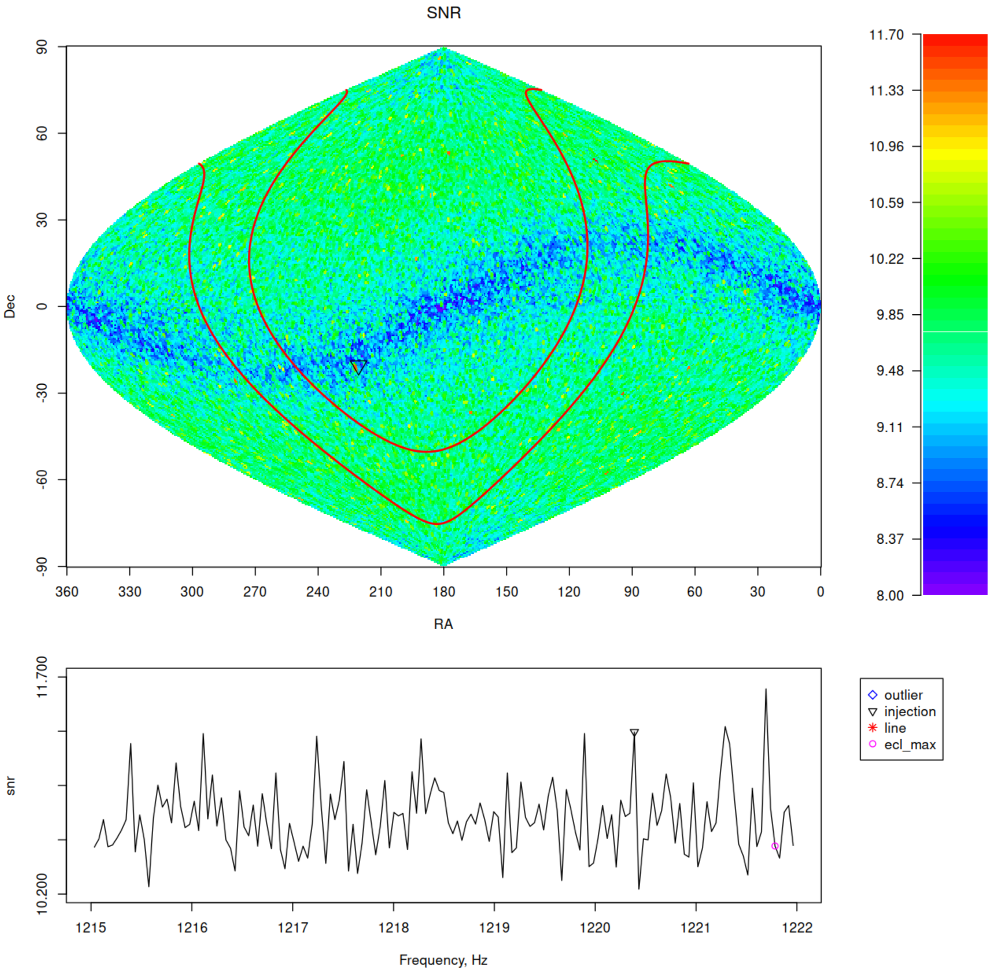
Phys. Rev. X 13, 021020 (2023)





Continuous gravitational wave atlas

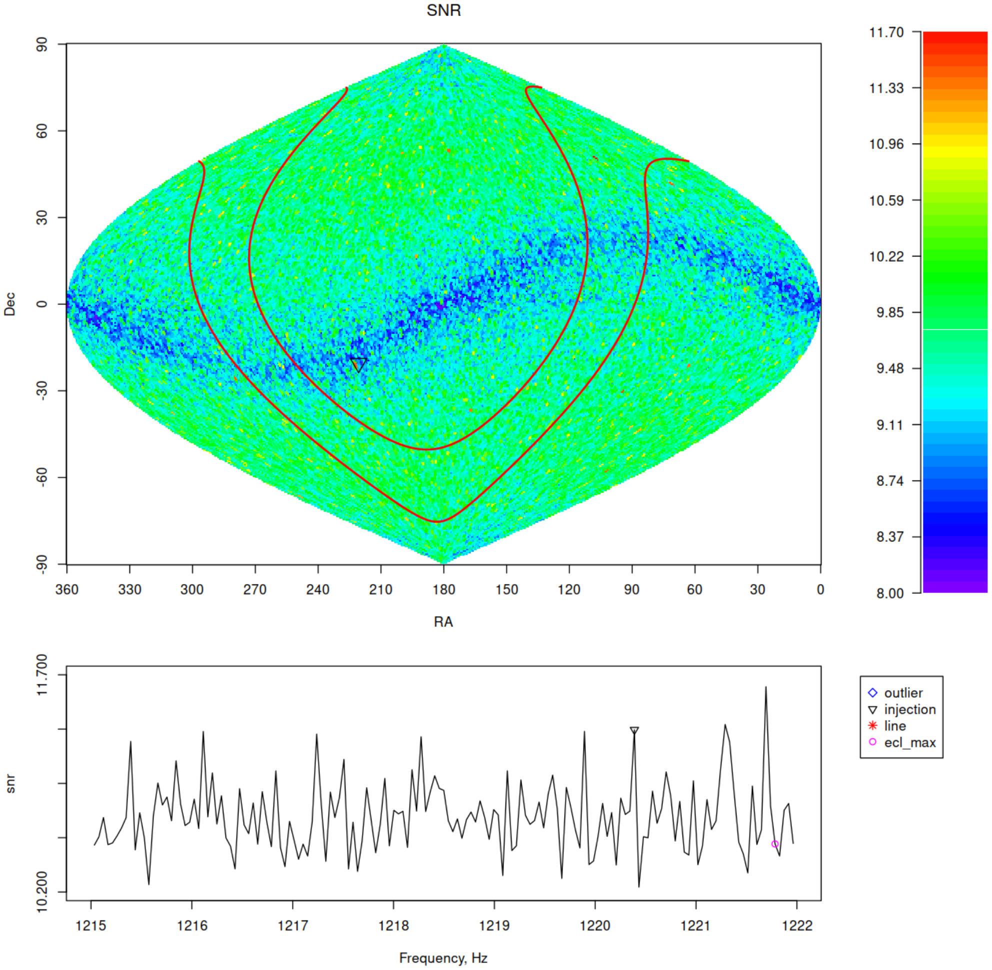
- An atlas is a collection of skymaps, such as shown on the right
- Separate skymaps for each 45mHz frequency
- Each pixel has data for many different metrics



Continuous gravitational wave atlas

- **New early release (2023 Nov) while we are still analyzing outliers**
- 20-1500 Hz
- |fdot| < 5e-10 Hz/s
- Data from two stages, 12 and 24 hour coherence length

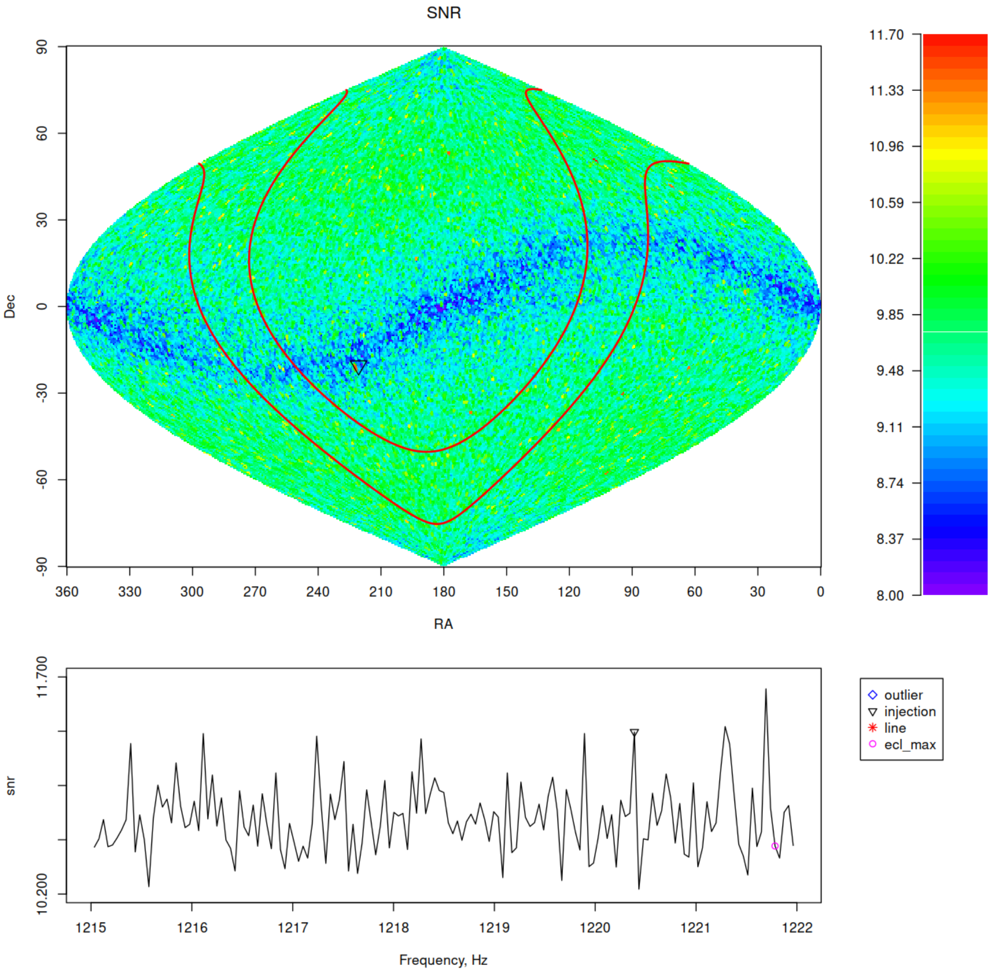
Phys. Rev. D 109, 022007 (2024)



Metrics

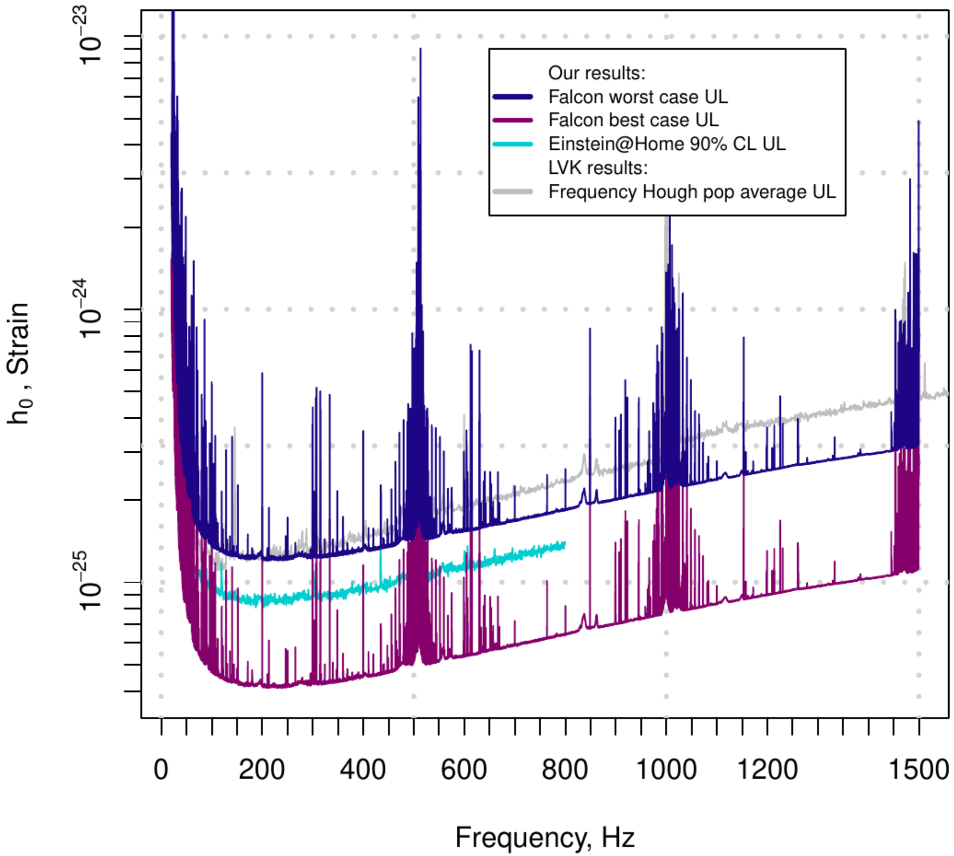
Each sky location and frequency band has the following metrics:

- maximum SNR
- frequency and polarization where SNR was achieved
- upper limit on arbitrary polarized signals (“worst case”)
- upper limits for circularly polarized signals
- data to compute polarization specific upper limits



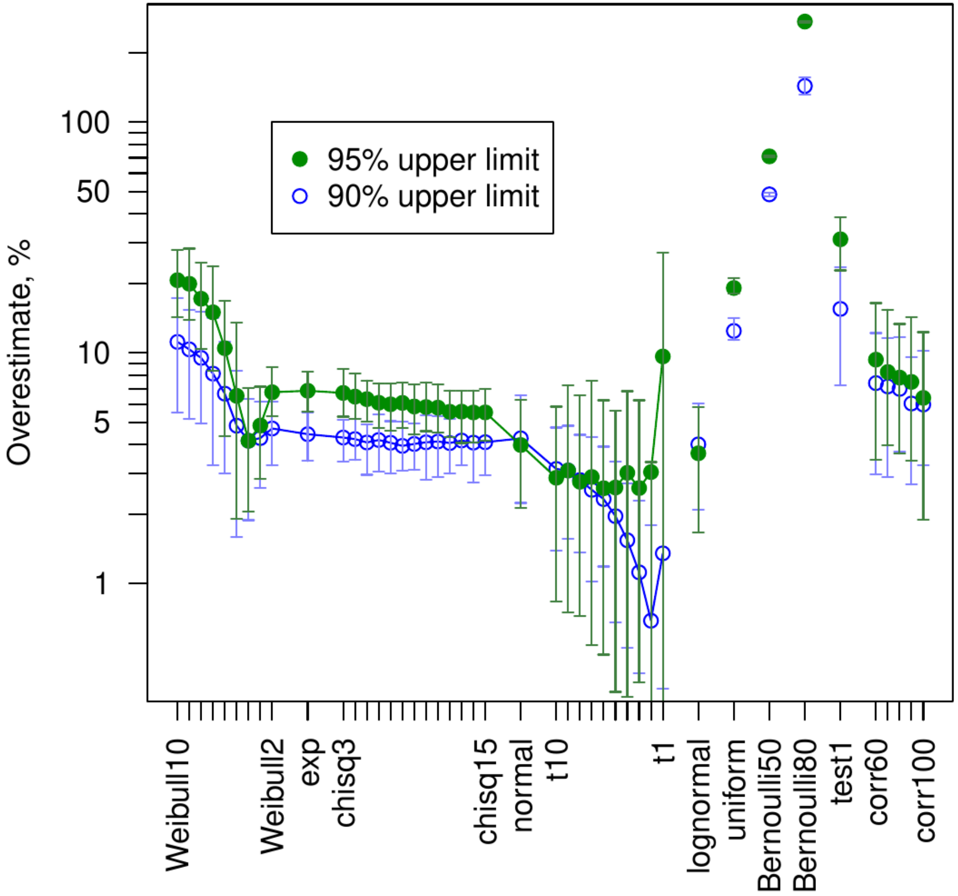
Atlas is constructed by analyzing signal power

- The h(t) data is demodulated, passed through narrow band filter, squared and summed.
- This makes it easy to relate the output statistics with the strength of astrophysical signals
- Plot on the right shows upper limits, derived using power measurements.
- No necessity for software injections
- No problem producing separate upper limit for each sky position and each frequency band



The power is analyzed using universal statistics

- Upper limits are produced using universal statistic method
- Universal means that it does not assume any particular distribution of noise in input data.
- The penalty is a slight overestimate of upper limits, compared to what could be achieved if distribution was known (plot on the right)
- Atlas reports 95% confidence level upper limits. They are valid in all frequency bands, over all the sky – no exclusions



Polarization specific upper limits

- Atlas includes *functional* upper limits, with a separate 95% confidence level value for each i and ψ
- The coefficients c_1-c_{14} are chosen large enough that upper limits are always valid
- It is possible to choose them so that the overestimate is small (~5%) for noise dominated data

$$\begin{aligned}\widehat{\mathrm{UL}}^2 = \big(& c_1 + f_{pp}c_2 + f_{pc}c_3 + f_{cc}c_4 + f_{impc}c_5 + f_{pp}^2 c_6 + \\ & + f_{cc}^2 c_7 + f_{pc}^2 c_8 + f_{impc}f_{pp}c_9 + f_{impc}f_{pc}c_{10} + \\ & + f_{impc}f_{cc}c_{11} + f_{pp}f_{pc}c_{12} + f_{cc}f_{pc}c_{13} + \\ & + f_{pp}f_{cc}c_{14}\big) \,/\, \left(f_{pp} + f_{cc}\right)\end{aligned}$$

$$\begin{aligned} a_+ &= \frac{\left(1+\cos^2\iota\right)^2}{4} \\ a_\times &= \cos^2\iota \\ f_{pp} &= 2|\tilde{w}_1|^2 = \tfrac{1}{4}\left(a_+ + a_\times + (a_+ - a_\times)\cos 4\psi\right) \\ f_{pc} &= 4\mathrm{Re}\,\tilde{w}_1\tilde{w}_2^* = \tfrac{1}{2}\left((a_+ - a_\times)\sin 4\psi\right) \\ f_{cc} &= 2|\tilde{w}_2|^2 = \tfrac{1}{4}\left(a_+ + a_\times - (a_+ - a_\times)\cos 4\psi\right) \\ f_{impc} &= 2\mathrm{Im}\,\tilde{w}_1\tilde{w}_2^* = \tfrac{1}{4}\left(1+\cos^2\iota\right)\cos\iota \end{aligned}$$

arXiv:2311.09911

Example: hardware injections

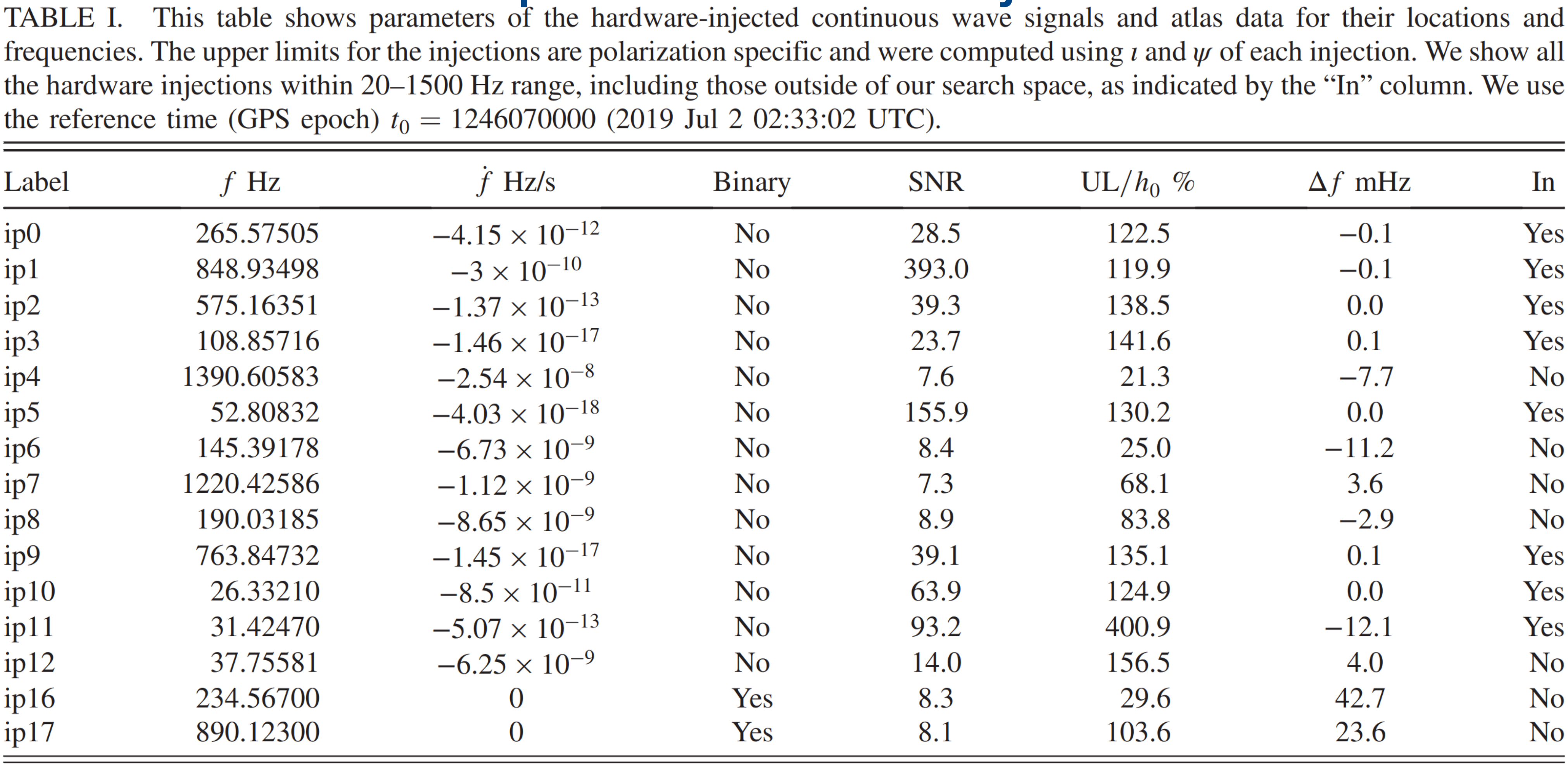
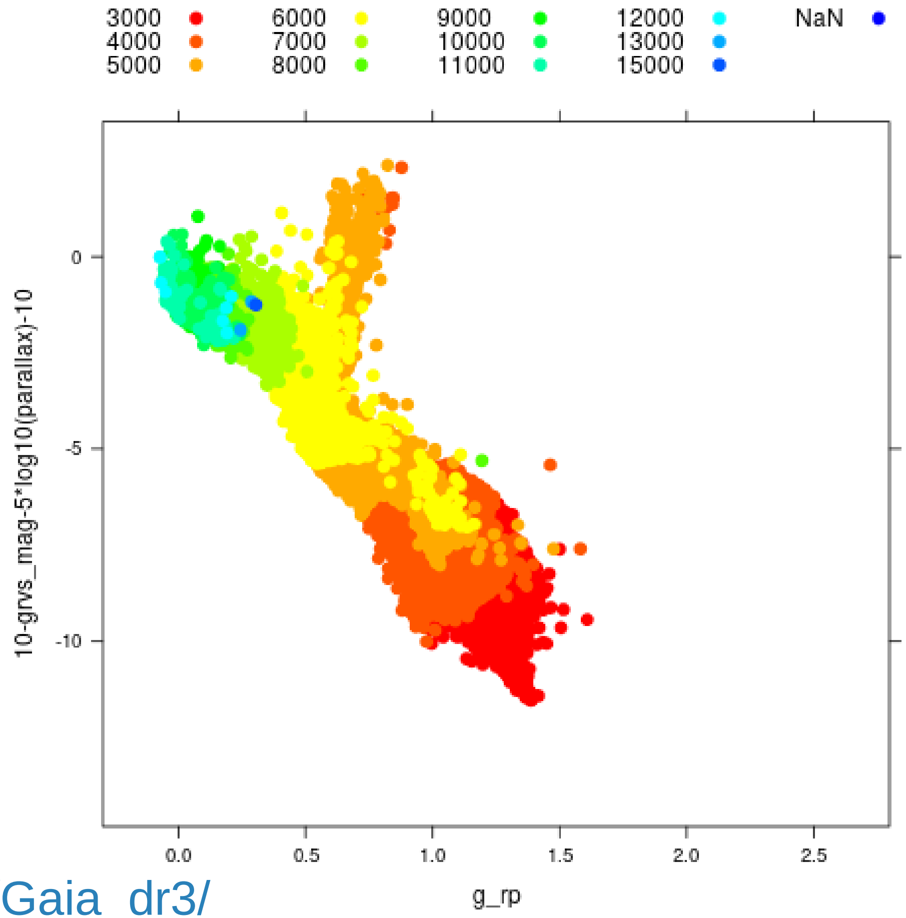


TABLE I. This table shows parameters of the hardware-injected continuous wave signals and atlas data for their locations and frequencies. The upper limits for the injections are polarization specific and were computed using ι and ψ of each injection. We show all the hardware injections within 20–1500 Hz range, including those outside of our search space, as indicated by the "In" column. We use the reference time (GPS epoch) $t_0 = 1246070000$ (2019 Jul 2 02:33:02 UTC).

Label	f Hz	$\dot{f}$ Hz/s	Binary	SNR	UL$/h_0$ %	Δf mHz	In
ip0	265.57505	-4.15×10^{-12}	No	28.5	122.5	−0.1	Yes
ip1	848.93498	-3×10^{-10}	No	393.0	119.9	−0.1	Yes
ip2	575.16351	-1.37×10^{-13}	No	39.3	138.5	0.0	Yes
ip3	108.85716	-1.46×10^{-17}	No	23.7	141.6	0.1	Yes
ip4	1390.60583	-2.54×10^{-8}	No	7.6	21.3	−7.7	No
ip5	52.80832	-4.03×10^{-18}	No	155.9	130.2	0.0	Yes
ip6	145.39178	-6.73×10^{-9}	No	8.4	25.0	−11.2	No
ip7	1220.42586	-1.12×10^{-9}	No	7.3	68.1	3.6	No
ip8	190.03185	-8.65×10^{-9}	No	8.9	83.8	−2.9	No
ip9	763.84732	-1.45×10^{-17}	No	39.1	135.1	0.1	Yes
ip10	26.33210	-8.5×10^{-11}	No	63.9	124.9	0.0	Yes
ip11	31.42470	-5.07×10^{-13}	No	93.2	400.9	−12.1	Yes
ip12	37.75581	-6.25×10^{-9}	No	14.0	156.5	4.0	No
ip16	234.56700	0	Yes	8.3	29.6	42.7	No
ip17	890.12300	0	Yes	8.1	103.6	23.6	No

Atlas data uses MVL file format

- Designed for efficient access by memory mapping
- Useful for interactive and scripted analysis of large data
- In addition to Falcon atlas, there is also a Gaia DR3 dataset in MVL format
- Each data set has examples of common searches



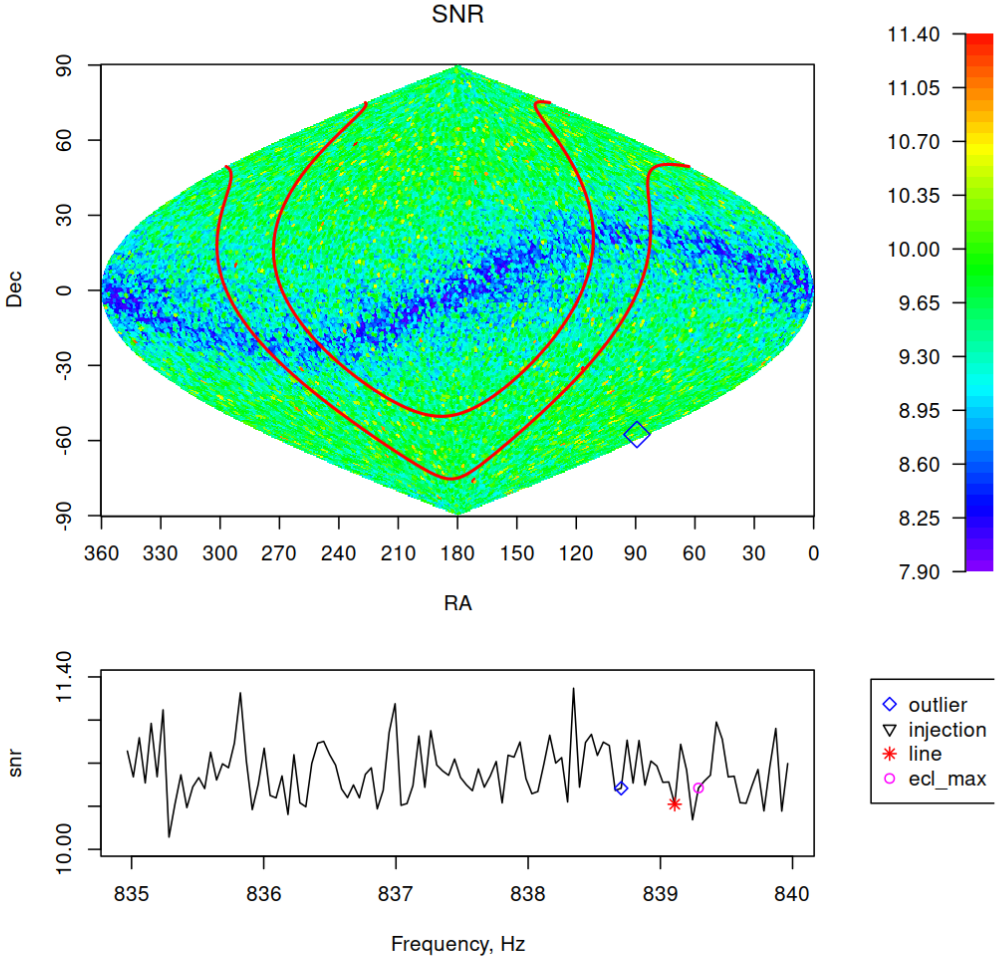
https://www.atlas.aei.uni-hannover.de/work/volodya/Gaia_dr3/

MySQL/MariaDB/sqlite vs MVL files

Mysql/MariaDB/sqlite/Postgresql	MVL
Collection of tables, each table consists of fixed length rows	Can store tables, but also lists, trees and other complex data structures
Lookup based indices, usually log(N) scaling	Hash based indices – O(1) scaling with length
	Spatial indices - find objects near query
Needs setup, dedicated server	Just files – use as is
Server needs to be large enough to support cluster usage	Files are memory mapped and just need a fast enough file system.
	Loaded data is shared between processes
Supports bulk data storage as well as frequently changed data, such as created by transactions	Focused on large data storage, optimized for solid state drives

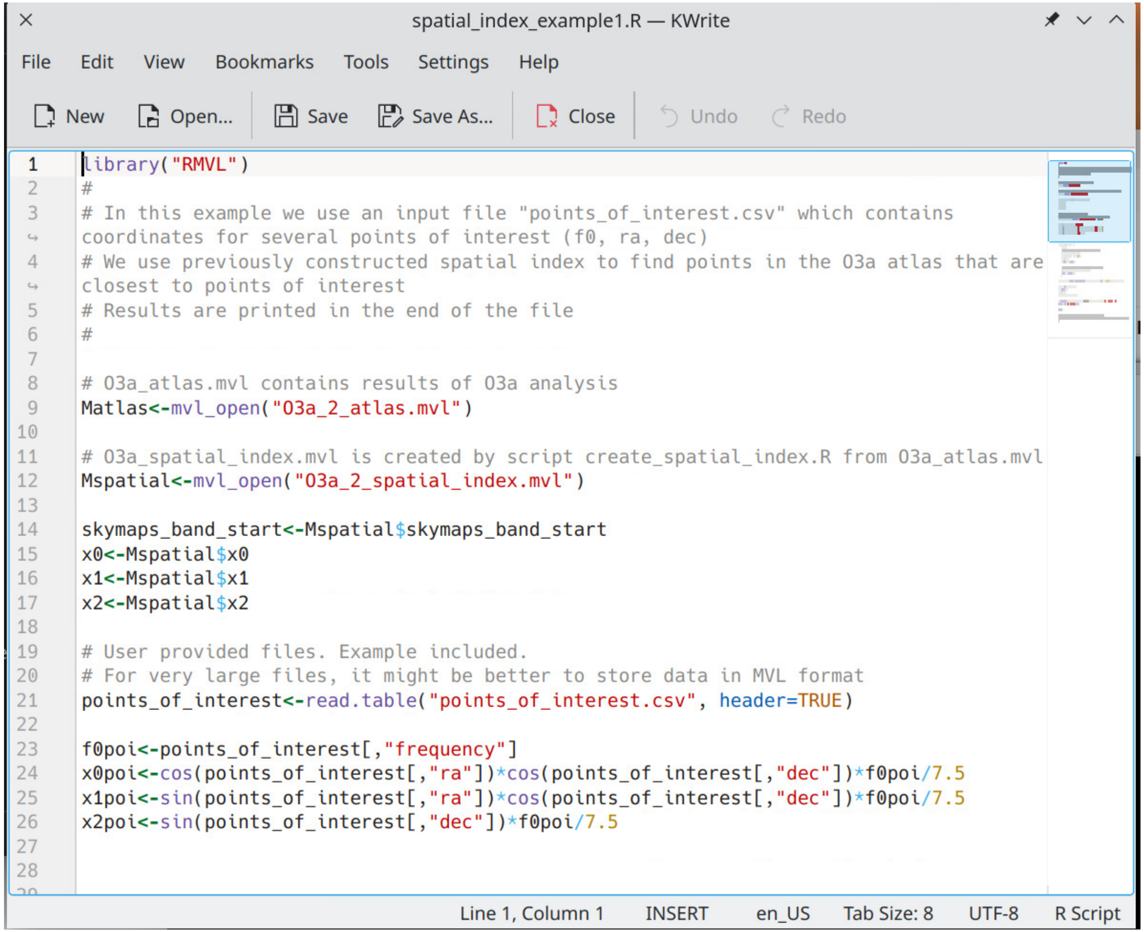
Usage examples: interactive browser

- R example `view_summary.R`
- Interactively displays maximums of SNR or upper limits across a frequency band and over sky
- Plot on the right was made with `plot_gw(835, 840, “snr”)`
- Also see `view_help()`



Usage examples: investigating single templates

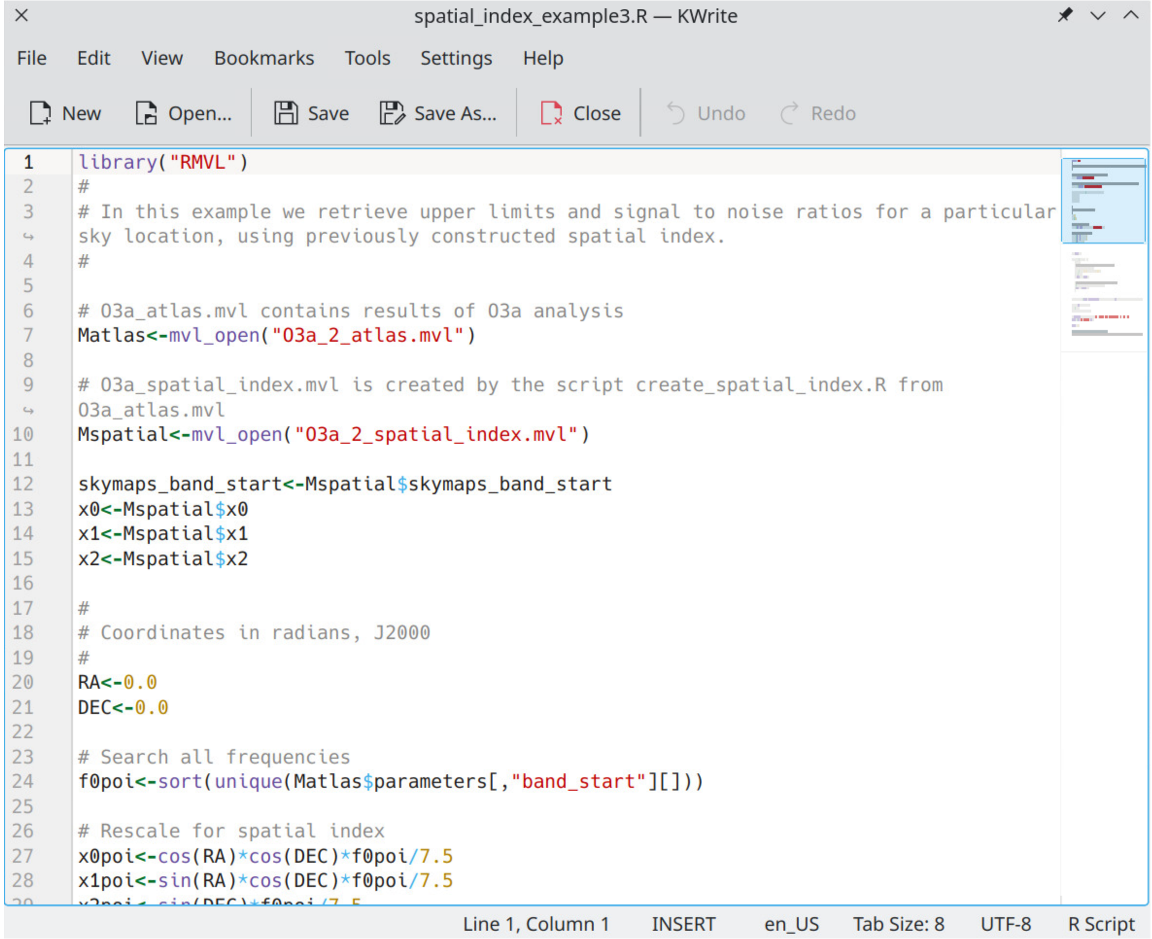
- R example `spatial_index_example1.R`
- Uses spatial index to quickly find data for specific sky location and frequency



```r
library("RMVL")
#
# In this example we use an input file "points_of_interest.csv" which contains coordinates for several points of interest (f0, ra, dec)
# We use previously constructed spatial index to find points in the O3a atlas that are closest to points of interest
# Results are printed in the end of the file
#

# O3a_atlas.mvl contains results of O3a analysis
Matlas<-mvl_open("O3a_2_atlas.mvl")

# O3a_spatial_index.mvl is created by script create_spatial_index.R from O3a_atlas.mvl
Mspatial<-mvl_open("O3a_2_spatial_index.mvl")

skymaps_band_start<-Mspatial$skymaps_band_start
x0<-Mspatial$x0
x1<-Mspatial$x1
x2<-Mspatial$x2

# User provided files. Example included.
# For very large files, it might be better to store data in MVL format
points_of_interest<-read.table("points_of_interest.csv", header=TRUE)

f0poi<-points_of_interest[,"frequency"]
x0poi<-cos(points_of_interest[,"ra"])*cos(points_of_interest[,"dec"])*f0poi/7.5
x1poi<-sin(points_of_interest[,"ra"])*cos(points_of_interest[,"dec"])*f0poi/7.5
x2poi<-sin(points_of_interest[,"dec"])*f0poi/7.5
```

Usage examples: investigating specific sky location

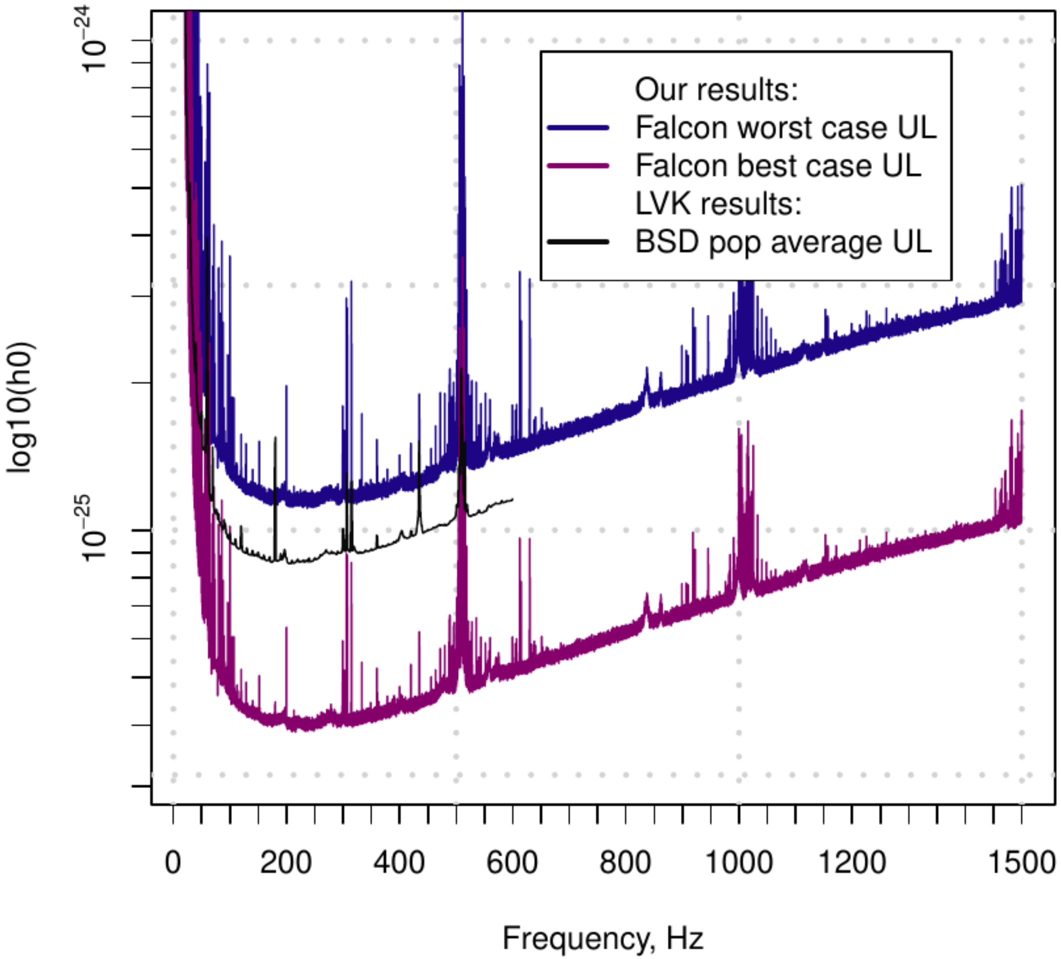
- R example `spatial_index_example3.R`
- Uses spatial index to quickly find all data for a specific sky location



```r
library("RMVL")
#
# In this example we retrieve upper limits and signal to noise ratios for a particular sky location, using previously constructed spatial index.
#

# O3a_atlas.mvl contains results of O3a analysis
Matlas<-mvl_open("O3a_2_atlas.mvl")

# O3a_spatial_index.mvl is created by the script create_spatial_index.R from O3a_atlas.mvl
Mspatial<-mvl_open("O3a_2_spatial_index.mvl")

skymaps_band_start<-Mspatial$skymaps_band_start
x0<-Mspatial$x0
x1<-Mspatial$x1
x2<-Mspatial$x2

#
# Coordinates in radians, J2000
#
RA<-0.0
DEC<-0.0

# Search all frequencies
f0poi<-sort(unique(Matlas$parameters[,"band_start"][]))

# Rescale for spatial index
x0poi<-cos(RA)*cos(DEC)*f0poi/7.5
x1poi<-sin(RA)*cos(DEC)*f0poi/7.5
```

Example: directed searches

- G189.1+3.0 – data on the right is an extract from Falcon atlas
- Latest LVK results shown for comparison



Summary

- Atlas provides all-sky, spectrally resolved data for continuous gravitational wave sources – a starting point for new searches
- New MVL file format for large scale data analysis
- Ready to use examples of searches using Falcon atlas and Gaia DR3 data

- Get the data !

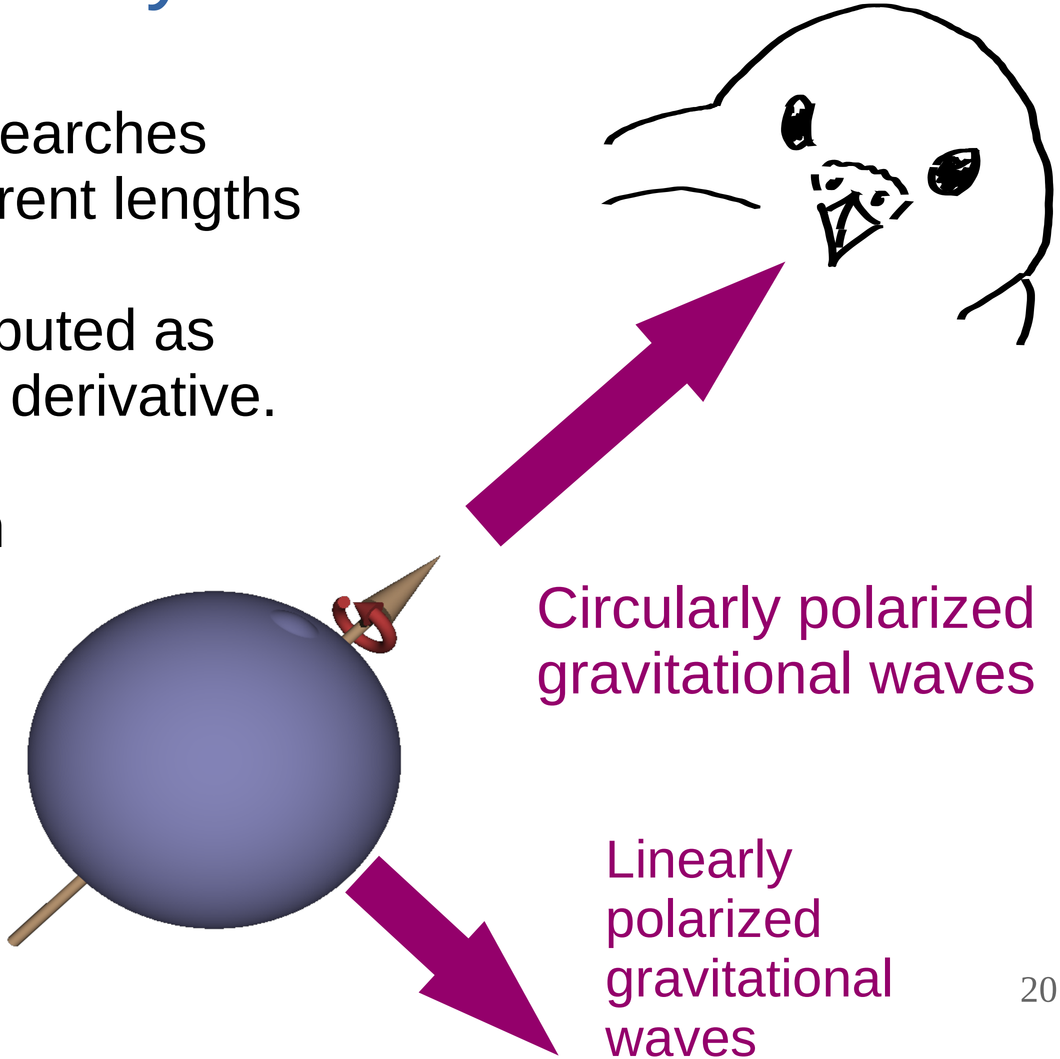
https://www.atlas.aei.uni-hannover.de/work/volodya/O3a_2_atlas/

END OF TALK

Falcon – Fast Loosely Coherent Search

- Designed for wide band all-sky searches
- Optimized for analysis with coherent lengths from few hours to several days.
- Worst case upper limits are computed as maximum over sky and frequency derivative. They are valid for any subset
- Detection pipeline produces high quality outliers

Phys. Rev. Lett. 123, 101101 (2019)
Phys. Rev. D 101, 022001 (2020)
Phys. Rev. Lett. 125, 171101 (2020)
Phys. Rev. D 103, 063019 (2021)
Phys. Rev. X 13, 021020 (2023)



Circularly polarized gravitational waves

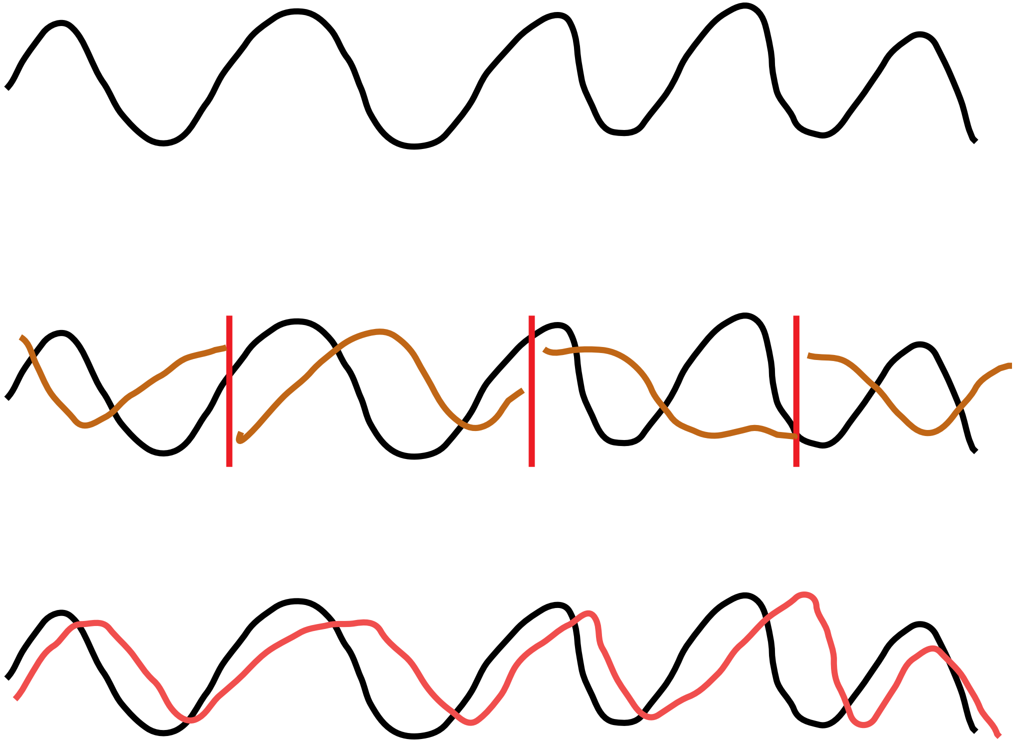
Linearly polarized gravitational waves

What is a loosely coherent search ?

Conventional matched filter looks for one waveform at a time. Sensitive, but very large parameter space

Semi-coherent searches partition data and integrate results of analysis in each chunk. Sensitivity lost due to unphysical waveforms.

Loosely coherent search analyses sets of trajectories at a time. The set of allowed waveforms is controlled for best sensitivity and computational efficiency



Low-ellipticity pulsars

- It is known that neutron star crust can support ellipticities of ≈10^{-6}
- But we do not know what physical process will produce them naturally
- No detections in previous searches
  - This might be due to lack of sensitivity, with signals just below noise floor
  - Or because natural sources do not perfectly follow assumed model

There are generic arguments that many known pulsars have ellipticities of 10^{-8} and that there is a minimum ellipticity of 10^{-9}
ApJ 863 2 G. Woan, M. D. Pitkin, B. Haskell, D. I. Jones, P. D. Lasky

Low-ellipticity pulsars

- Plot on the right shows distance to pulsars with ellipticity of 10^{-8}
- We are sensitive to sources up to 150 pc away
- Frequency derivatives up to $\pm 5 \cdot 10^{-11}$
- +50% sensitivity compared to O2

Phys. Rev. X 13, 021020 (2023)