Comparing Veto Algorithms for Gravitational Waves

Ian Wojtowicz, Jan Williams, Jay D. Tasson
Carleton College, Northfield, MN 55057, USA

The detection of gravitational waves, gravitational interactions' analogue of light, rely on highly sensitive measurements with giant instruments. Background noise, such as seismic vibration, is common in gravitational wave data, and thus needs to be removed through the application of noise characterization algorithms. Two common algorithms are Used Percentage Veto (UPV) and Hierarchical Veto (hVeto), but little is known about their respective advantages. To explore this, we ran both algorithms with similar configuration parameters on a week-long data set from the Hanford interferometer site. Preliminary results suggest that hVeto has a higher efficiency in identifying noise, but is much less selective in what it chooses to veto as noise than UPV. Through further interpretation of our data, we hope to make recommendations that will allow UPV and hVeto to capitalize on their strengths and increase their performance in noise identification for gravitational wave analysis.

How gravitational waves are detected

- Gravitational waves are ripples in the fabric of the universe, also known as spactime, and are analogous to ripples in water.
- Laser Interferometer Gravitational wave Observatory (LIGO) measures minuscule stretches and squeezes in spacetime as the waves passes through the Earth.
- The detectors are extremely sensitive: sources of statistical "noise" in the data are also picked up, from passing cars to variations in weather. Many noise-detecting sensors are placed at different locations in the observatories, which serve as "auxiliary channels" for incident noise.
- Computer algorithm programs are used to identify, or "veto" detector noise and determine which noise sources are statistically significant. Two widely used algorithms are Used Percentage Veto (UPV) and hierarchical Veto (hVeto). (For further reading, see Reference 1.) The overall process is shown in Fig. 1.

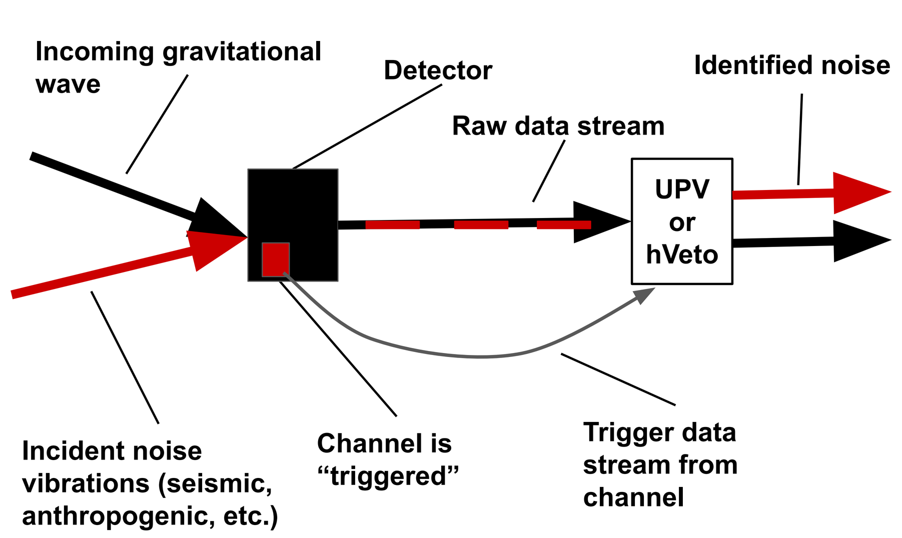


Figure 1. A schematic of the veto algorithm's purpose.

An approach for algorithm comparison

- Goal was to match the input parameters of UPV as closely as possible to hVeto, then running the input over the specified data set. The approach is illustrated in Fig. 2.
- Input parameters include a signal-to-noise ratio threshold, a list of time segments for which the detector was active, and a list of auxiliary channels to run over.

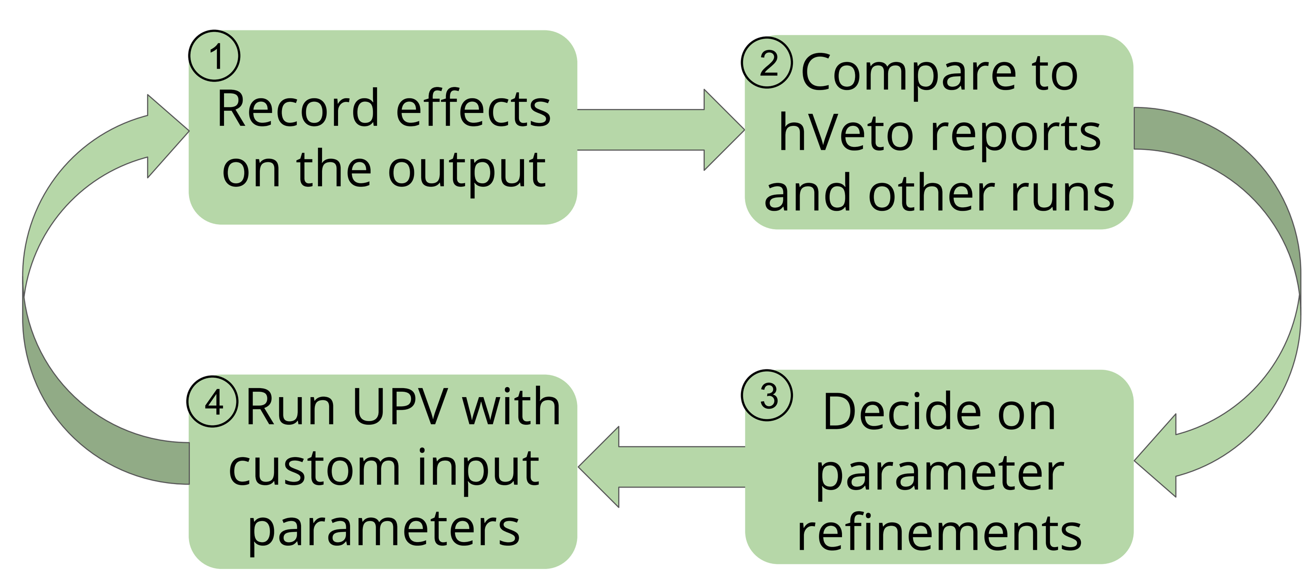
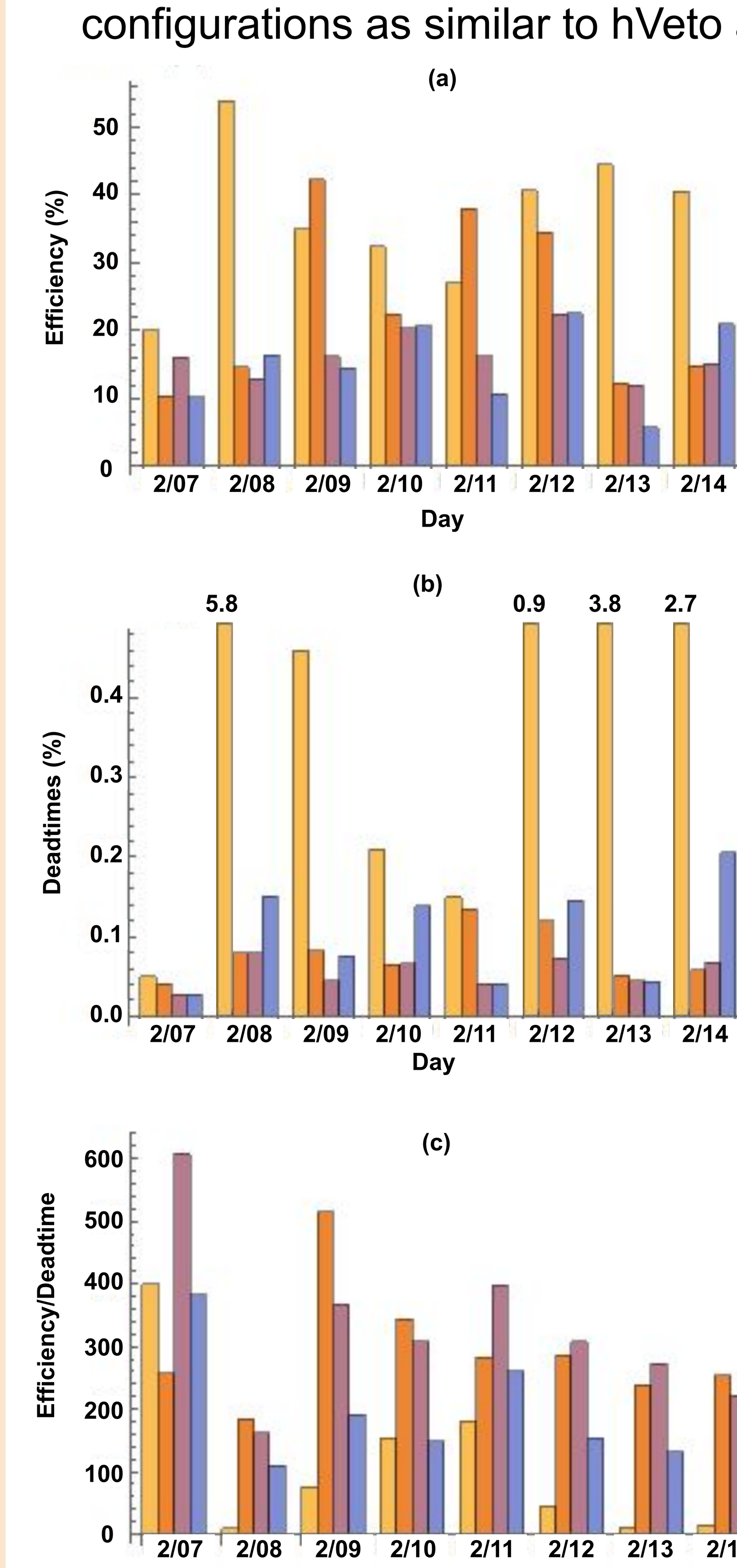


Figure 2. The workflow of how we approached our study.

- The data chosen was a set of 7 days (2/7/20 - 2/14/20) from the LIGO site in Hanford, Washington. The data was chosen because of its continuity, and was relatively free of unusual glitches that could introduce anomalies.
- The aspects of the output data studied were deadtime, efficiency, and efficiency divided by deadtime.
- Deadtime is the percentage of time in the channel data that is vetoed out of the overall analysis time that is removed from the data set.
- The efficiency is a percentage of the gravitational wave channels that are vetoed out of the total number of channels analyzed.
- The ratio of efficiency over deadtime is a crude estimate of how good a certain veto is. A high efficiency and low deadtime indicates a better veto.

The results

- In Fig. 3, the yellow and the orange bars respectively represent data from the hVeto and UPV runs from a LIGO data summary webpage. These are the controls.
- The purple bar is a UPV run using a sort of "middle ground" in configuration parameters of UPV and hVeto. The blue bar represents UPV runs taken with configurations as similar to hVeto as possible.



- Both runs of UPV with altered inputs (purple and blue) had consistently similar or lower efficiencies than either hVeto or UPV control data.
- Cutoff threshold used for the signal-to-noise ratio only slightly increased deadtimes. This suggests some deeper differences in underlying behavior between UPV and hVeto.
- The intermediate configurations (purple) sometimes had the best efficiency to deadtime ratio.

Figure 3. (a) Efficiency of each run. (b) Deadtimes of each run. Some of the values were very large, so they are labeled above their bars. (c) Efficiency divided by deadtime for each run.

Conclusions and future work

- As the run parameters of UPV were made more similar to hVeto, all of the results became more similar to hVeto's control.
- Preliminary results suggest hVeto provides more efficient but less selective vetoing channel triggers than UPV.
- The future approach is to start looking at how individual glitches in data noise are handled by each algorithm.

Acknowledgements

This project was funded by the Towsley Endowment at Carleton College. We acknowledge the use of LIGO scientific collaboration's computing resources. We also thank Nelson Christensen, Florent Robinet, and Jack Heinzel for their contributions to this project.

References

1. Tomoki Isogai and the Ligo Scientific Collaboration and the Virgo Collaboration *2010 J. Phys.: Conf. Ser.* **243** 012005
2. Associated Symbols | Creative Strategy | Carleton College, https://apps.carleton.edu/creative-strategy/identity/symbols/
3. *LIGO- H1 Network Summary,* https://ldas-jobs.ligo-wa.caltech.edu/~detchar/summary/.
4. LSC - LIGO Scientific Collaboration, https://www.ligo.org/.