Vetoes Used in the E7 Inspiral Upper Limit Analysis

Part 2

Peter Shawhan, reporting for the Inspiral Upper Limit Working Group

LSC Meeting

August 21, 2002

Vetoes for E7 Inspiral Upper Limit Analysis
Overview

Goal: reduce number of inspiral candidates by vetoing times when a transient is detected in an auxiliary channel

Target sample: events found by inspiral search code with snr>7 & chisq<5
(chisq cut alone eliminates most of the events with snr>7)

Take an empirical approach to deciding which channel(s) and transient detection algorithm(s) to use

Try as many things as possible!

Evaluate trade-off between veto efficiency and deadtime

We studied veto performance (for H2 and L1 only) using the E7 "playground" triple-coincidence data

Details are in document circulated to U.L. groups

Vetoes for E7 Inspiral Upper Limit Analysis

Initial Exploration

We used a variety of existing tools to find transients in auxiliary channels

- DMT monitors: absGlitch, CorrMon, NonMon
- TFCLUSTERS
- Standard inspiral template search code, processing auxiliary channels
- Matlab scripts
- Visual examination with DTT

Many thanks to Burst U.L. group for sharing information!

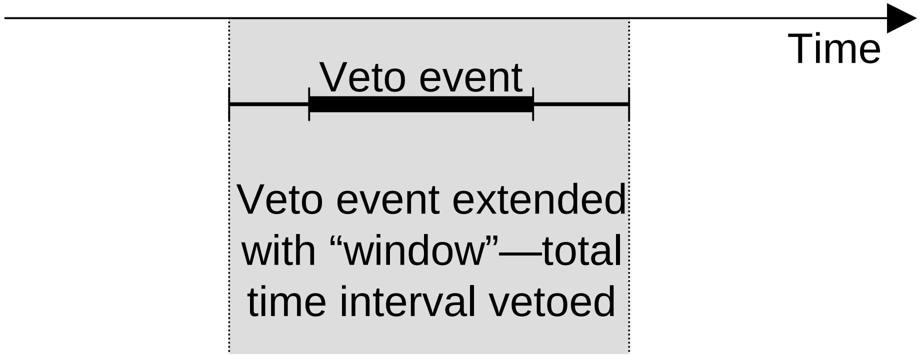
Compared against lists of GW inspiral event candidates

Find that interferometer channels (L1:PSL-ISS_ACTM_F, H2:LSC-MICH_CTRL, etc.) provide better vetoes than PEM channels, for this data

Vetoes for E7 Inspiral Upper Limit Analysis
Quantitative Evaluation

Wrote a C program called “ivana” to read files of inspiral candidates & veto events and compare event times

Can extend intrinsic veto event duration with a user-specified “window”



ivana calculates veto efficiency and deadtime

Figure of merit: residual rate (as defined by Burst Group)

Accounts for deadtime introduced by the veto

Vetoes for E7 Inspiral Upper Limit Analysis
Automated Optimization Study

Wrote a script to loop over many combinations and log the output from ivana in each case

Veto event files evaluated:

- absGlitch applied to CARM_CTRL, MICH_CTRL, REFL_Q, H2:POB_Q, L1:PSL-ISS_ACTM_F, with various thresholds (courtesy of S. Ballmer)
- WaveMon applied to above plus IOO-MC_F (courtesy of S. Klimenko)
- Inspiral template search code applied to MICH_CTRL, REFL_Q, H2:POB_Q, with a few different SNR thresholds

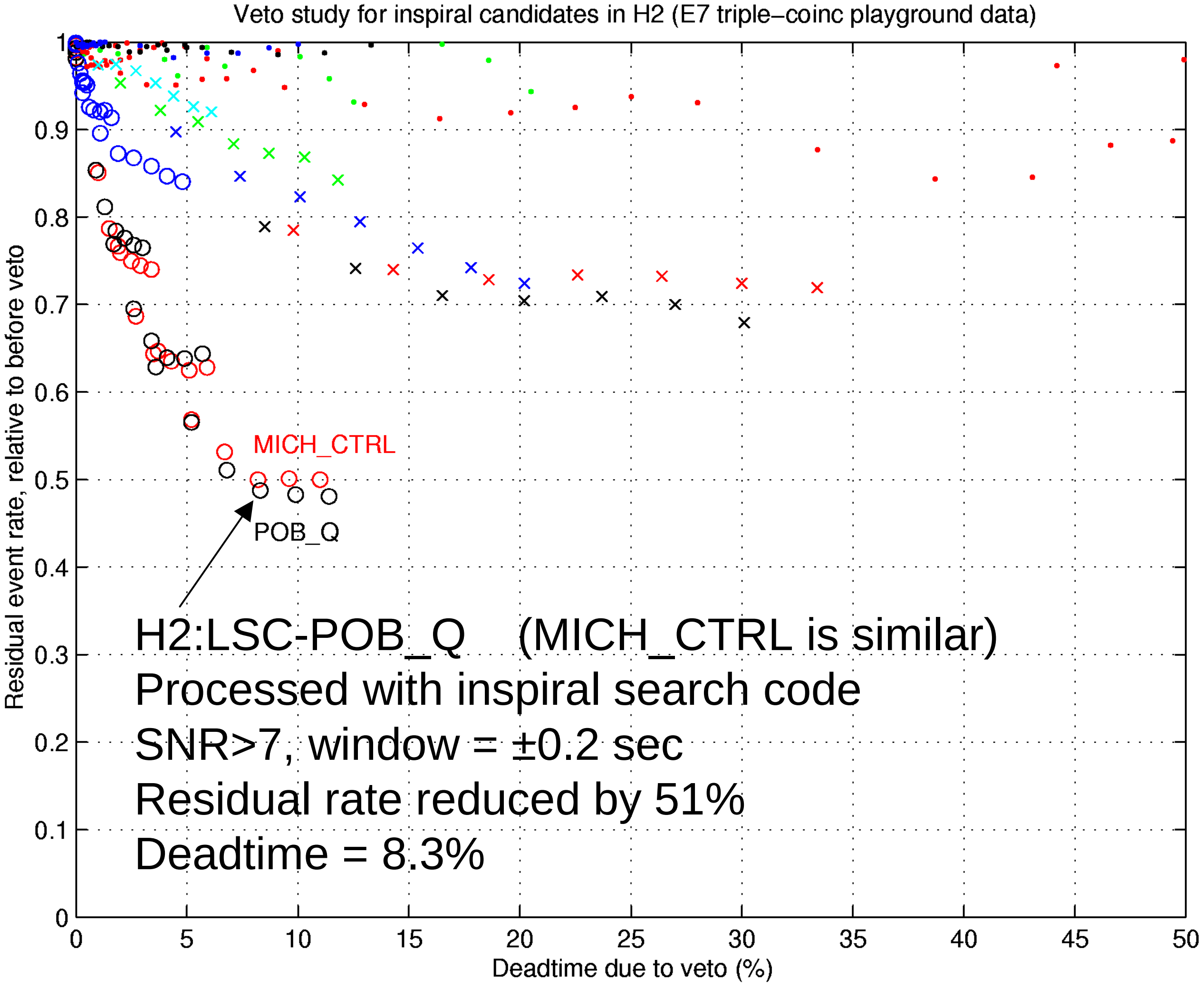
Used seven different windows, from 0 to ±0.3 sec

Total of 1078 combinations evaluated in a few hours

Vetoes for E7 Inspiral Upper Limit Analysis

Results of Optimization Study

H2:

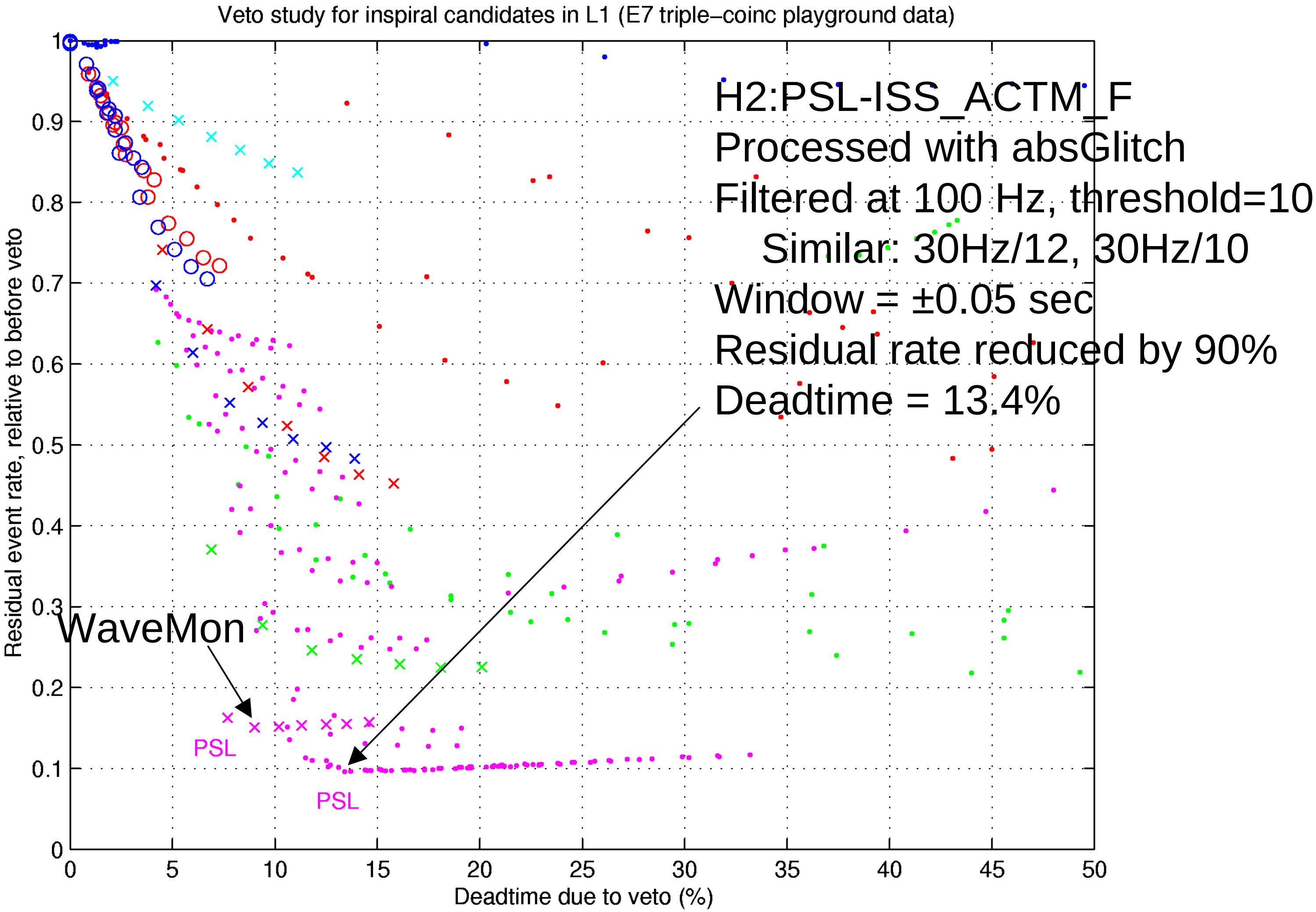


H2:LSC-POB_Q (MICH_CTRL is similar)
Processed with inspiral search code
SNR>7, window = ±0.2 sec
Residual rate reduced by 51%
Deadtime = 8.3%

Vetoes for E7 Inspiral Upper Limit Analysis

Results of Optimization Study

L1:

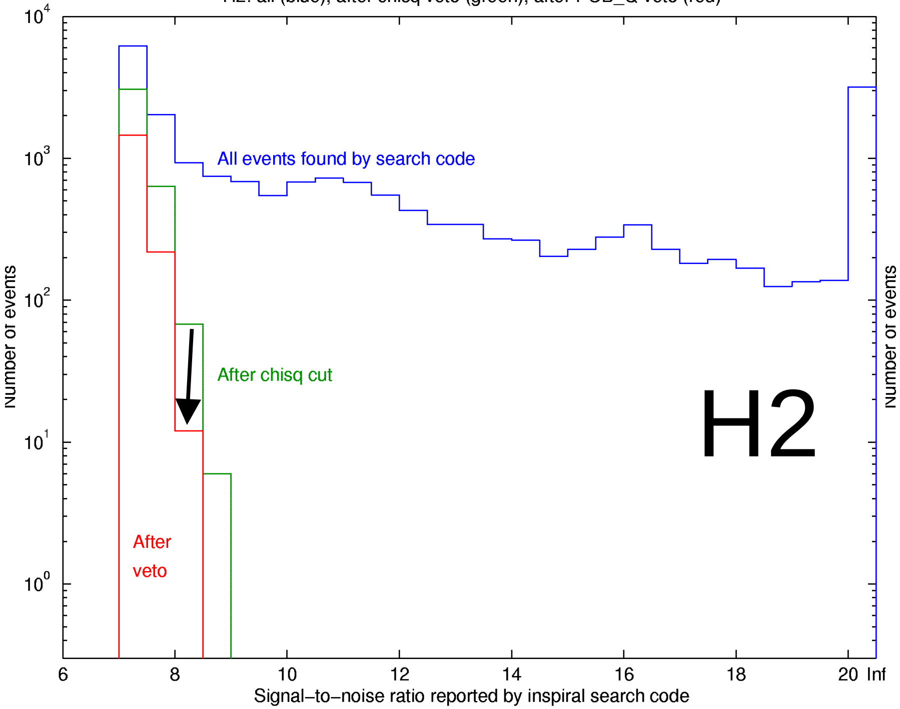


H2:PSL-ISS_ACTM_F
Processed with absGlitch
Filtered at 100 Hz, threshold=10
Similar: 30Hz/12, 30Hz/10
Window = ±0.05 sec
Residual rate reduced by 90%
Deadtime = 13.4%

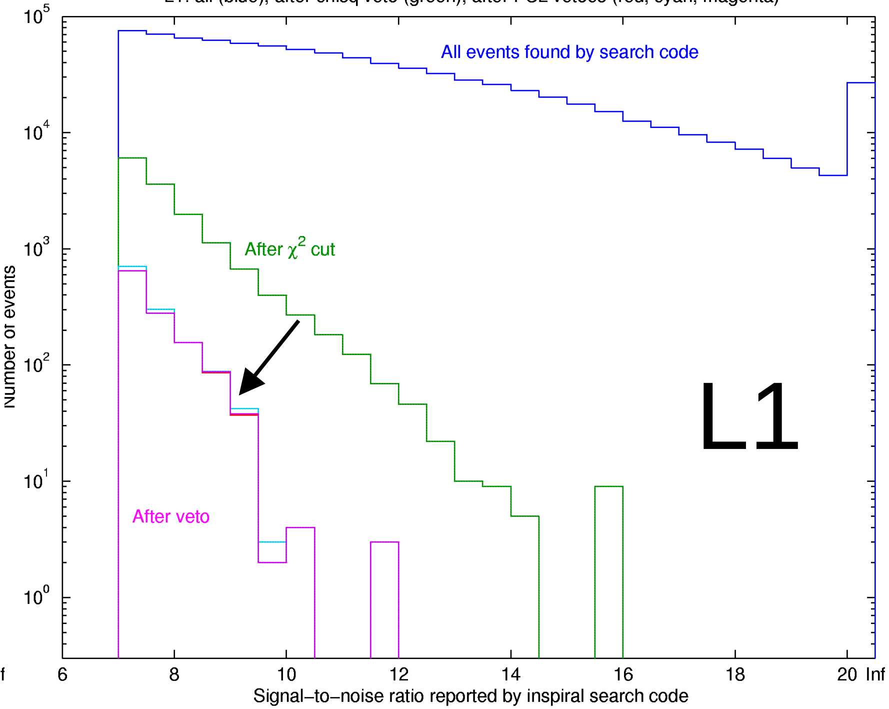
Effect of Vetoes on E7 Inspiral Candidate SNR Distributions

(Fine-tuning of veto windows further reduces deadtime by ~1% without degrading veto efficiency)

H2: all (blue), after chisq veto (green), after POB_Q veto (red)



L1: all (blue), after chisq veto (green), after PSL vetoes (red, cyan, magenta)



chisq cut gets rid of most events, and vetoes help even more

A Sneak Peek at S1

Nelson Christensen and his student, Carl Ebeling, have used ivana to study data taken at LLO on July 11-12

Target sample (for now): transients in AS_Q found by absGlitch

Have looked for correlations with transients found by absGlitch in many PEM and interferometer channels

A few channels may show a little benefit when used as a veto, but nothing like the veto effectiveness achieved for the E7 data