Multivariate Veto Analysis for Real-time Gravitational Wave Burst Search

Zhen Wang[1], Lindy Blackburn[2],
Junwei Cao[1,2], and Erik Katsavounidis[2]
[1]Tsinghua University [2]MIT LIGO Laboratory

The First Galileo-Xu Guangqi Meeting
Shanghai, China, October 28 2009

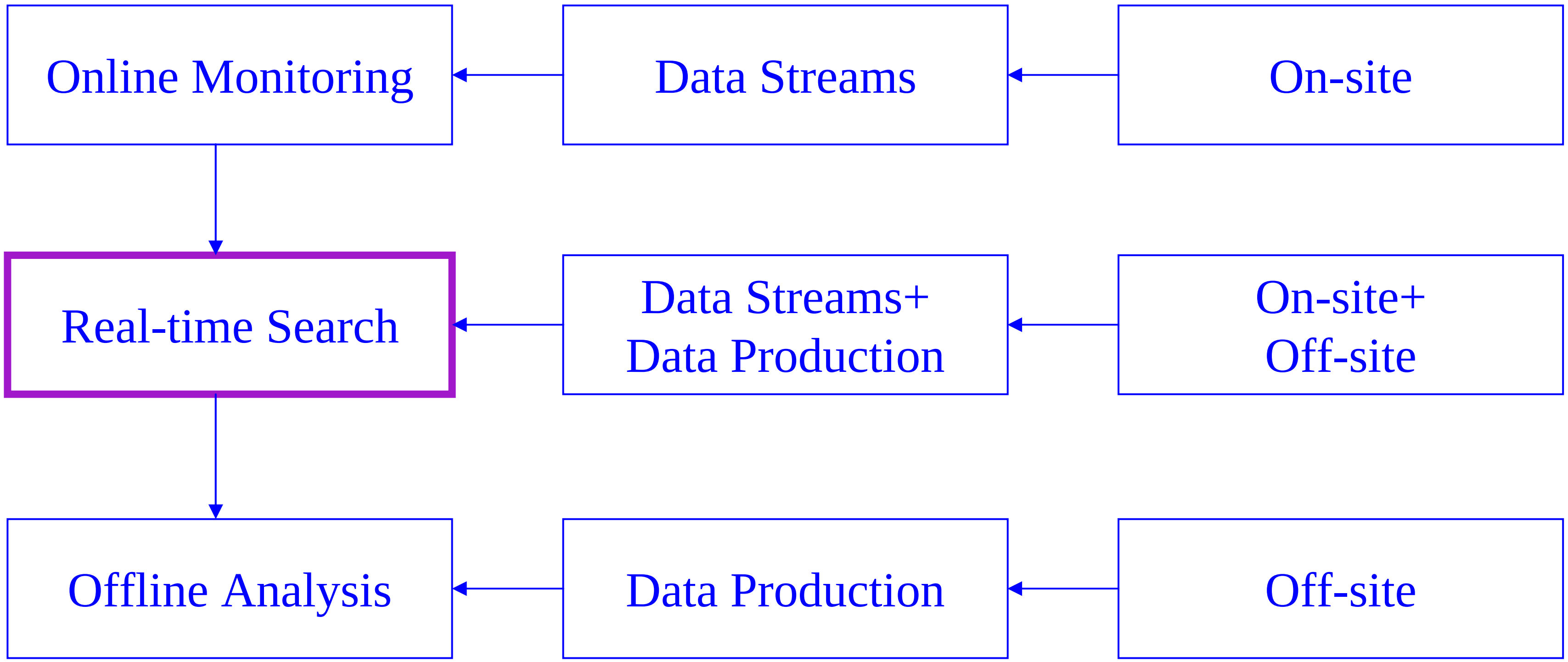
Outline

- Real-time GW burst search
  - » Astrophysical and computing perspectives
  - » Real-time requirements
- Multivariate veto analysis
  - » The current veto method
  - » SVM-based veto approach
- Experimental results
  - » Experimental data
  - » Sample generation
  - » Veto efficiency
  - » Performance comparison
- Conclusions

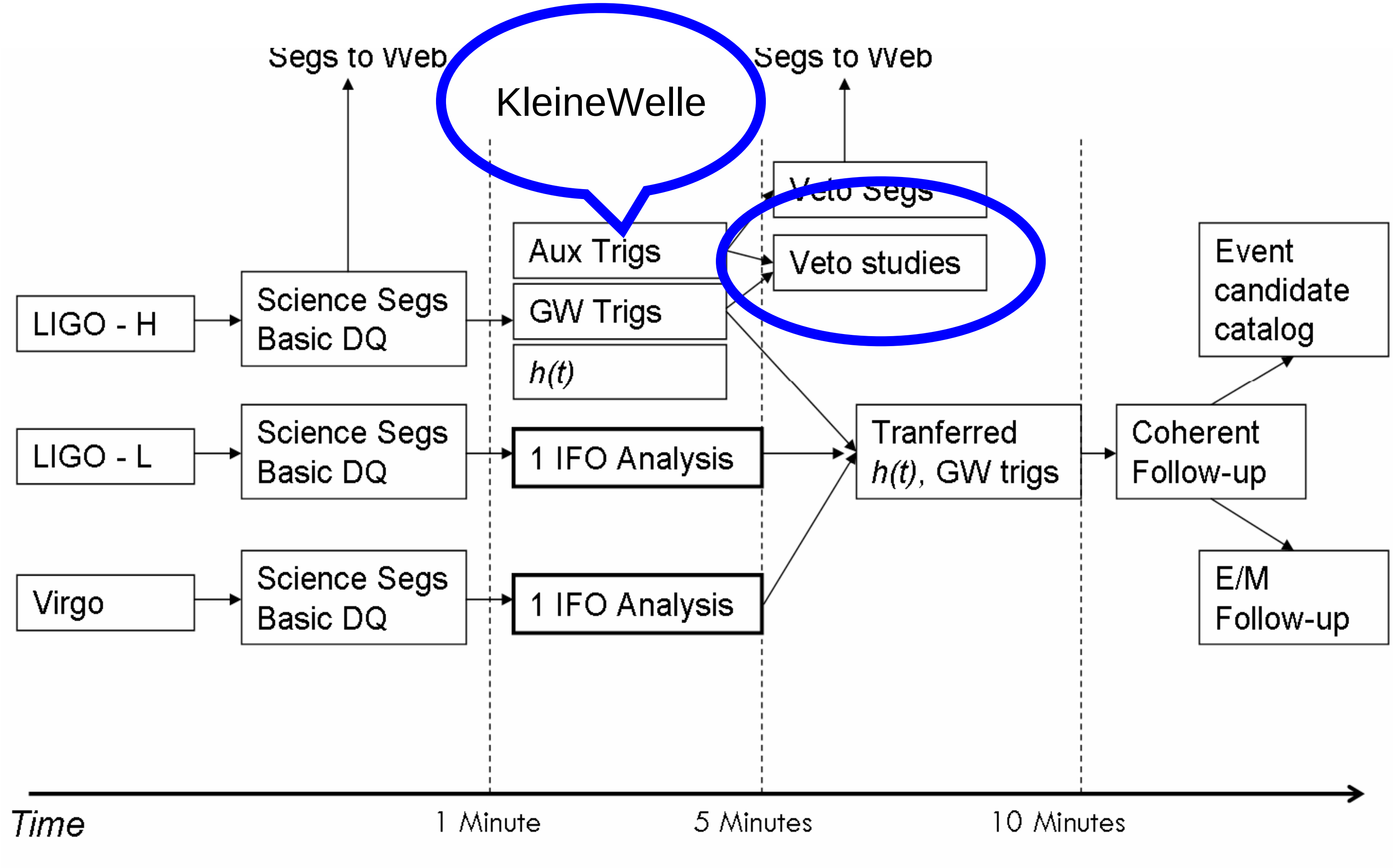
Real-Time Burst Search

- Rapid E/M follow-up of GW candidates
  - » Astrophysical processes which produce GW signals strong enough to be detected must release a lot of energy, so it is very likely that some of that energy is emitted in the form of E/M radiation, which is generally much easier to detect—if one is looking in the right direction.
- Prompt follow-up of “external” E/M and particle triggers
  - » In the case of external triggers and their follow up with GW detectors, online GW burst searches may provide information about an astrophysical event (even if no gravitational wave signal is detected).
- Improve interferometer operation
- Expedite offline searches

Real-time Search – from a Computing Perspective

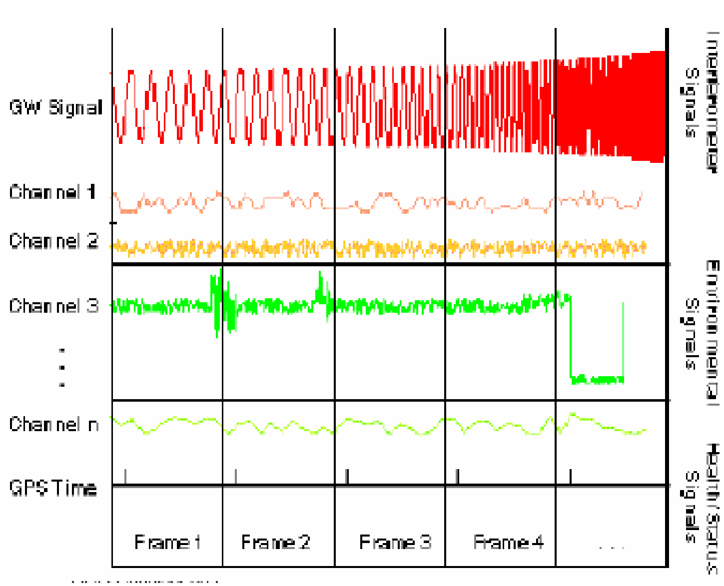


An Example Implementation



Veto Requirements

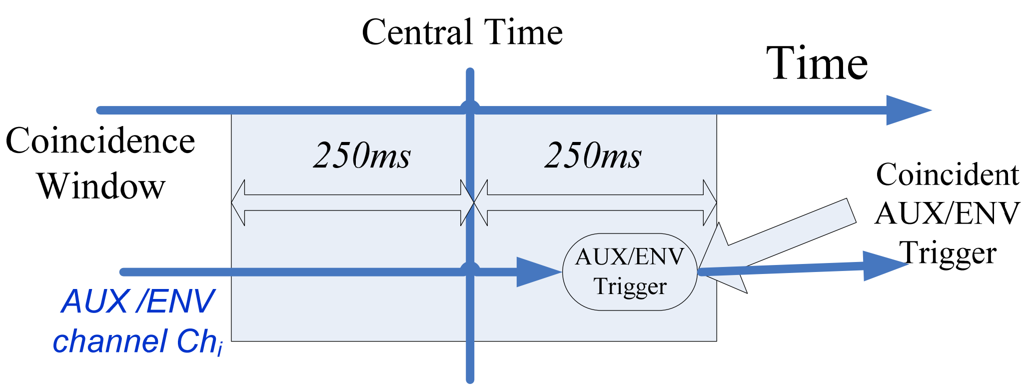
- A GW channel (GC) generates hundreds of triggers everyday, which are resulted from instrument faults (a glitch) or gravitational wave (a signal). Whether or not a GC trigger is a glitch or a signal has to be determined by veto analysis.
  - » Triggers from auxiliary (AUX) channels record instrument status in real time;
  - » Triggers from environmental (ENV) channels record the environmental fluctuate in real time.



The Current Veto Method

- The current event-by-event veto method is to analyze the coincidence between GC triggers and every AUX/ENV triggers independently. If there exits a coincident AUX/ENV trigger similar with a GC trigger, it is considered the GC trigger is associate with the corresponding AUX/ENV channel, and thus vetoed.
  - » Search coincident AUX/ENV triggers.
  - » Compared the coincident AUX/ENV triggers with GC triggers in frequency-amplitude domain.

Search coincident AUX/ENV triggers



All-or-Nothing

- Event-by-event vetoes should use information from all channels to decide if a GC trigger should be kept.
  - » If all channels are used for veto, nothing is kept at the end using the existing method;
  - » If only a few channels are used, nothing could be vetoed.
- Current veto methods make little use of information such as coincidence among AUX/ENV channels.
- It takes too long to meet real-time requirements (combination of results from multiple veto methods).
- Difficult to be extended for involvement of a larger number of AUX/ENV channels.

Our Approach

- The veto process can be considered as a classification problem of instrument status:
  - » The input of the classifier is the combination of properties of all coincident AUX/ENV triggers at the time t_i, assuming there is a GC trigger at the time t_i.
  - » The output of the classifier is whether the instrument is fault or not.
    - – If yes, the GC trigger is a glitch;
    - – If not, the GC trigger is a GW signal

This is definitely a GW signal

We want to know if this is a signal or a glitch

No signal and no glitch either! No instrument faults! perfect for training the classifier

GC	1	1	0	0
AUX1	0	1	0	0
AUX*n*	0	0	1	0
ENV1	0	0	1	0
ENV*m*	0	1	0	0

The Feature Vector

- Feature Vector $V(t_i)$:

  The feature vector at time t_i describes the instrument status at the time t_i. Assume there are *k* AUX/ENV channels, the properties of coincident AUX/ENV triggers at t_i are $Ch_1(t_i), Ch_2(t_i) \ldots Ch_k(t_i)$, then the feature vector is $V(t_i)=(Ch_1(t_i), Ch_2(t_i) \ldots Ch_k(t_i))$.
- In our experiments, we use *6* properties to characterize a AUX/ENV trigger (generated by KleineWelle):
  - » Energy, nomal_Energy,npts, frequency, significance and centralTime_Distance

  For example, there are 152 AUX/ENV channels associated with H1 GC channel. In this case, the $V(t_i)$ is a 152*6=912 dimension vector.
- ➔ Multivariate or multidimensional approach is required, e.g.

  S. Mukherjee, Multidimensional classification of kleineWelle triggers from LIGO science run, Class. Quantum Grav. 23 (2006) S661–S672

Glitch and Signal Samples

- Glitch samples:

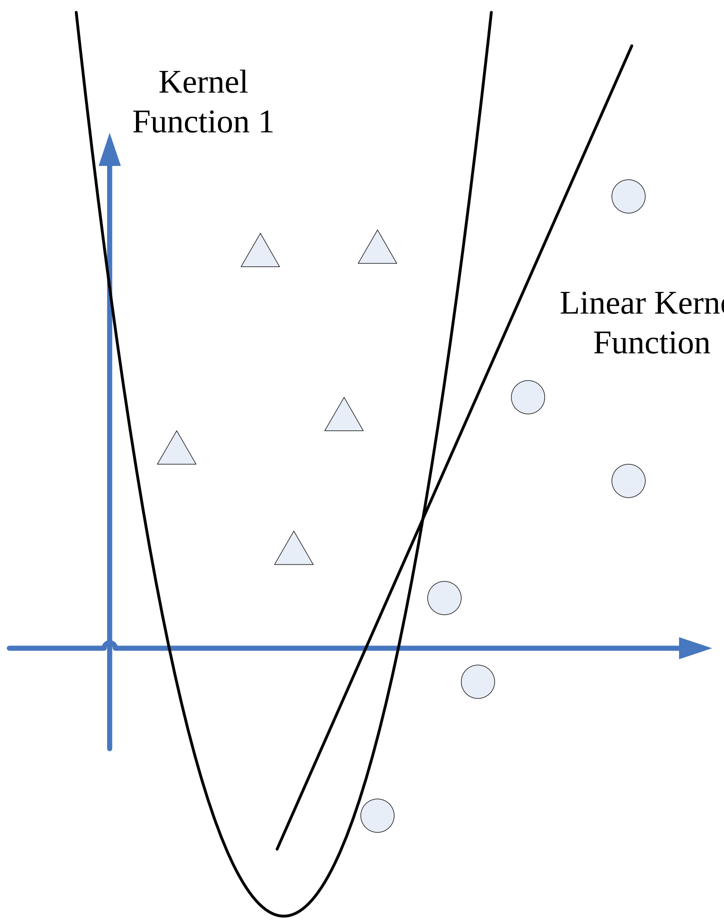
  A feature vector $V(t_i)$ is defined as a glitch sample if a certain instrument fault leads to a GC trigger at time t_i. Generally speaking, most of the GC triggers (>99%) are generated by instrument faults. Assume $TN=(TN_1, TN_2, \ldots TN_n)$ is the central time serial of GC triggers, then $V(TN_i)$, $i=1,2,\ldots.n$ can be regarded as a serial of glitch samples.

- Signal samples:

  A feature vector $V(t_i)$ is defined as a signal sample if no instrument fault can lead to GC trigger at time t_i. AUX/ENV coincident triggers at any time besides central time of GC triggers can be regarded as a signal sample. Assume $TP=(TP_1, TP_2, \ldots TP_m)$ is a random time serial, then $V(TP_i)$, $i=1,2,\ldots.m$ can be regarded as a serial of signal samples.

SVM

- Multivariate classification methods are designed to handle this many-dimensional problem, and are well suited to answer the question of whether or not a particular GC trigger is consistent with instrumental origins.
- SVM (Support Vector Machine) is a well established technique to divide a sample into two classes based on a set of parameters. Hyperplane will be chosen to maximize separation between samples in N-dimensional space.
- SVM tries to find a hyperplane (described by the kennel function) to divide two types of samples in two subspaces, maximizing the accuracy of the classification.



Thresholds

- For a GC trigger, a threshold on the significance is used to eliminate the noise. The significance presents the possibility whether or not this trigger is a Gaussian noise. The lower the significance is, the higher possibility the trigger is a Gaussian noise. In our experiments, we use a widely accepted significance threshold 35 for GC triggers.

- For AUX/ENV triggers, the significance threshold is used to eliminate the possible noise, for safety consideration. Some of the AUX/ENV channels are set with particular significance thresholds due to channel requirements. Other AUX/ENV channels usually use 35 as their default significance threshold.

Experimental Data

- S5 H1 Data from 818352000 – 819043200, in which there are 211014 GC triggers. To veto GC triggers, 154 relevant AUX/ENV channels are used in the SVM classifier.
- AUX/ENV channels set with particular significance thresholds:

s5_h1_etmyoplevperror.trg	50
s5_h1_etmyoplevyerror.trg	100
s5_h1_itmypd1ac.trg	100
s5_h1_pobi.trg	50
s5_h1_prcctrl.trg	50
s5_h1_qpdxdc.trg	50
s5_h1_qpdxp.trg	100
s5_h1_wfs2ip.trg	50

Experimental Data

- AUX channels for H1 applied are :

```
s5_h1_asac.trg
s5_h1_bsoplevperror.trg
s5_h1_bsoplevyerror.trg
s5_h1_bsp.trg
s5_h1_bsy.trg
s5_h1_etmxexcdaq.trg
s5_h1_etmxoplevperror.trg
s5_h1_etmxoplevyerror.trg
s5_h1_etmxp.trg
s5_h1_etmxy.trg
s5_h1_etmyoplevperror.trg
s5_h1_etmyoplevyerror.trg
s5_h1_etmyp.trg
s5_h1_etmyy.trg
s5_h1_itmxoplevperror.trg
s5_h1_itmxoplevyerror.trg
s5_h1_itmxpd1ac.trg
s5_h1_itmxpd2ac.trg
s5_h1_itmxp.trg
s5_h1_itmxy.trg
s5_h1_itmyoplevperror.trg
s5_h1_itmyoplevyerror.trg
s5_h1_itmypd1ac.trg
s5_h1_itmypd2ac.trg
s5_h1_itmyp.trg
s5_h1_itmyy.trg
s5_h1_mcf.trg
s5_h1_mcl.trg
s5_h1_michctrl.trg
s5_h1_mmt3oplevperror.trg
s5_h1_mmt3oplevyerror.trg
s5_h1_pobi.trg
s5_h1_pobq.trg
s5_h1_pobsdc.trg
s5_h1_prcctrl.trg
s5_h1_qpdxdc.trg
s5_h1_qpdxp.trg
s5_h1_qpdxy.trg
s5_h1_qpdydc.trg
s5_h1_qpdyp.trg
s5_h1_qpdyy.trg
s5_h1_refldc.trg
s5_h1_refli.trg
s5_h1_reflq.trg
s5_h1_rmoplevperror.trg
s5_h1_rmoplevyerror.trg
s5_h1_rmp.trg
s5_h1_rmy.trg
s5_h1_wfs1qp.trg
s5_h1_wfs1qy.trg
s5_h1_wfs2ip.trg
s5_h1_wfs2iy.trg
s5_h1_wfs2qp.trg
s5_h1_wfs2qy.trg
s5_h1_wfs3ip.trg
s5_h1_wfs3iy.trg
s5_h1_wfs4ip.trg
s5_h1_wfs4iy.trg
```

Experimental Data

- ENV channels for H1 applied are:

s5_h0_bsc10acc1y.trg	s5_h0_bsc6magx.trg	s5_h0_eyseisy.trg	s5_h0_isct4accy.trg	s5_h0_mxv1.trg
s5_h0_bsc10magx.trg	s5_h0_bsc6magy.trg	s5_h0_eyseisz.trg	s5_h0_isct4accz.trg	s5_h0_mxv2.trg
s5_h0_bsc10magy.trg	s5_h0_bsc6magz.trg	s5_h0_eyv1.trg	s5_h0_isct4mic.trg	s5_h0_myseisx.trg
s5_h0_bsc10magz.trg	s5_h0_bsc6mic.trg	s5_h0_eyv2.trg	s5_h0_isct7accx.trg	s5_h0_myseisy.trg
s5_h0_bsc10mic.trg	s5_h0_bsc7accx.trg	s5_h0_ham1accx.trg	s5_h0_isct7accy.trg	s5_h0_myseisz.trg
s5_h0_bsc1accy.trg	s5_h0_bsc7mic.trg	s5_h0_ham1accz.trg	s5_h0_isct7accz.trg	s5_h0_myv1.trg
s5_h0_bsc1mag1x.trg	s5_h0_bsc8accy.trg	s5_h0_ham3accx.trg	s5_h0_isct7mic.trg	s5_h0_myv2.trg
s5_h0_bsc1mag1y.trg	s5_h0_bsc8mic.trg	s5_h0_ham7accx.trg	s5_h0_lvea2v1.trg	s5_h0_psl1accx.trg
s5_h0_bsc1mag1z.trg	s5_h0_bsc9acc1x.trg	s5_h0_ham7accz.trg	s5_h0_lvea2v2.trg	s5_h0_psl1accz.trg
s5_h0_bsc2accx.trg	s5_h0_bsc9magx.trg	s5_h0_ham9accx.trg	s5_h0_lvea2v3.trg	s5_h0_psl1mic.trg
s5_h0_bsc2accy.trg	s5_h0_bsc9magy.trg	s5_h0_iot1mic.trg	s5_h0_lveamagx.trg	s5_h0_psl2accx.trg
s5_h0_bsc3accx.trg	s5_h0_bsc9magz.trg	s5_h0_isct10accx.trg	s5_h0_lveamagy.trg	s5_h0_psl2accz.trg
s5_h0_bsc4accx.trg	s5_h0_coilmagx.trg	s5_h0_isct10accy.trg	s5_h0_lveamagz.trg	s5_h0_psl2mic.trg
s5_h0_bsc4accy.trg	s5_h0_coilmagz.trg	s5_h0_isct10accz.trg	s5_h0_lveamic.trg	s5_h0_radiocs1.trg
s5_h0_bsc5accx.trg	s5_h0_exseisx.trg	s5_h0_isct10mic.trg	s5_h0_lveaseisx.trg	s5_h0_radiocs2.trg
s5_h0_bsc5magx.trg	s5_h0_exseisy.trg	s5_h0_isct1accx.trg	s5_h0_lveaseisy.trg	s5_h0_radiolvea.trg
s5_h0_bsc5magy.trg	s5_h0_exseisz.trg	s5_h0_isct1accy.trg	s5_h0_lveaseisz.trg	
s5_h0_bsc5magz.trg	s5_h0_exv1.trg	s5_h0_isct1accz.trg	s5_h0_mxseisx.trg	
s5_h0_bsc5mic.trg	s5_h0_exv2.trg	s5_h0_isct1mic.trg	s5_h0_mxseisy.trg	
s5_h0_bsc6accy.trg	s5_h0_eyseisx.trg	s5_h0_isct4accx.trg	s5_h0_mxseisz.trg	

Sample Generation

- Three steps to generate glitch samples:
  - » Step 1: Select the GC triggers whose significance is higher than the threshold value 35.
  - » Step 2: Obtain the central time of the selected GC trigger and search coincident AUX/ENV triggers. (If there are more than one coincident AUX/ENV triggers inner the window, we select the nearest one.)
  - » Step 3: If the significance of the selected AUX/ENV trigger is smaller than its threshold, set this AUX/ENV trigger zero and fill it into the feature vector. Otherwise, directly fill the properties of this trigger into the feature vector.
- Three steps to generate signal samples:
  - » Step 1: generate a serial of random time TP_i.
  - » Step 2: Use TP_i to search coincident AUX/ENV triggers (If there are more than one coincident AUX/ENV triggers inner the window, we select the nearest one.).
  - » Step 3: If the significance of the selected AUX/ENV trigger is smaller than its threshold, set this AUX/ENV trigger zero and fill it into the feature vector. Otherwise, directly fill the properties of this trigger into the feature vector.
- 300 glitch samples and 300 signal samples are generated to train the SVM classifiers (pre-processing is necessary for normalization).

Veto Efficiency

Number of GC triggers with non-zero feature vectors: 1586	SVM veto efficiency= 1003/1586=63%
	Current veto Efficiency= 750/1586=48%
Number of GC triggers with zero feature vectors: 1447	SVM veto efficiency=0
	Current veto Efficiency = 63/1447=4%
Total number of GC triggers with a GC threshold 35: 3033	SVM Efficiency= 1003/3033=33%
	Current veto Efficiency= 813/3033=27%

Performance Comparison

	Number of signals using the current method	Number of glitches using the current method	Total
Number of signals using SVM	511,32%	72,5%	583,37%
Number of glitches using SVM	325,20%	678,43%	SVM veto efficiency= 1003,63%
Total	836,52%	Current veto Efficiency= 750,48%	1586,100%

Conclusions

- The SVM is a statistical approach. No instrument knowledge is involved. How do we know the veto results make sense? Huge overlap (75%) between results of our approach and the current method.
- Our approach provides significant improvement on veto efficiency for GC triggers with non-zero feature vectors.
- In the current veto method, most of zero-feature GC triggers are also regarded as signals, which means our assumption that zero-feature GC triggers (can't be classified by SVM) are all signals makes sense.
- Our approach can be easily extended with more channels involved. Since more channels may result in more GC triggers with non-zero feature vectors and reduce the number of GC triggers with zero feature vectors (directly considered as signals), veto efficiency could be improved.
- SVM veto can be processed very fast. In our experiment, 3033 GC triggers are processed within 1 minute using a general laptop. This is very important to meet real-time search requirements.
- Our approach is not safe. Signals could also be vetoed and classified as glitches, though with a very low possibility.

Thanks

- More information on the LIGO Scientific Collaboration Research Group at Tsinghua University, Beijing, China:
  - » http://ligo.org.cn
  - » jcao@tsinghua.edu.cn