Tracksearch Update

A Time Frequency Method for Gravitational Wave Data Analysis

C. Torres
University of Texas at Brownsville
ASIS Session

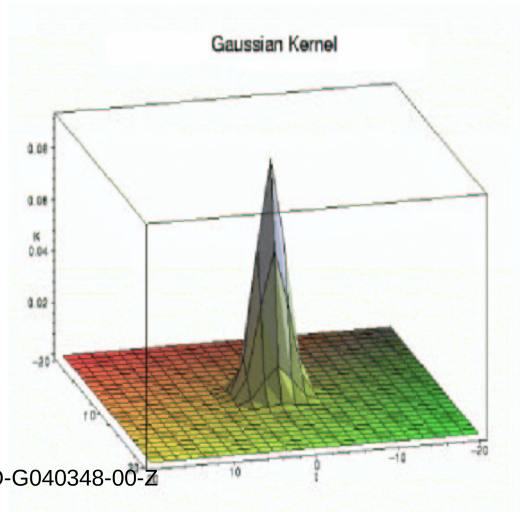
(Warren Anderson)

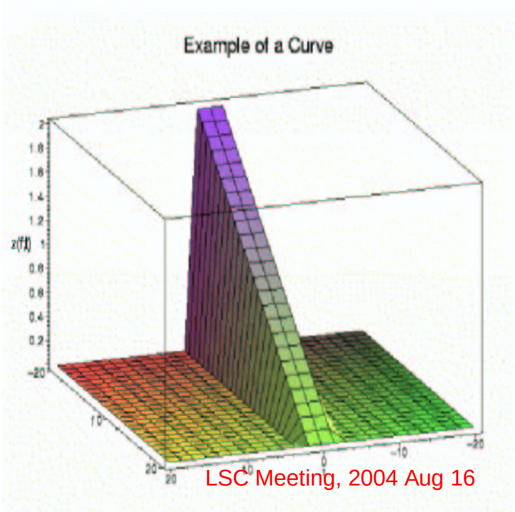
Tracksearch Introduction

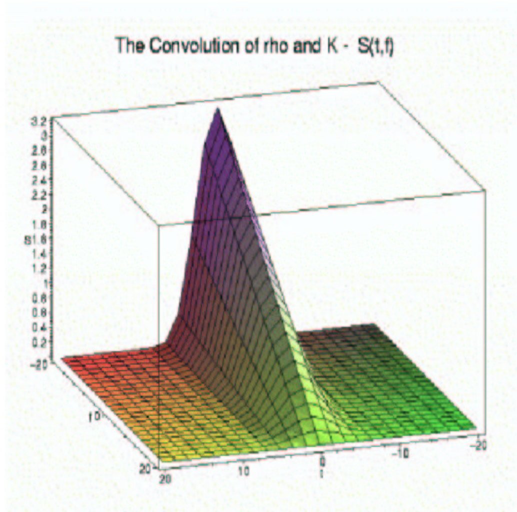
- This algorithm looks for unmodeled sources
- It will be well suited to longer-lived signals (whose durations are the second to minute ranges)
- Longer-lived signals are expected to be the result of perturbation about an equilibrium configuration. The perturbations will have a characteristic frequency which may evolve slowly in time.
- every signal can be expressed as $h(t) = A(t)\sin(2\pi f(t)t)$
- Algorithm search assumptions
  - $f(t)$ is a slowly varying function of time $f(t) \sim f(t+\delta t)$
  - $A(t)$ is slow varying function of time $A(t) \sim A(t+\delta t)$
  - δt is the period of the signal, which is the minimum time interval for which $\phi(t+\delta t) \sim \phi(t)$ where $\phi(t) = \int f(t)dt$ is the signal phase

Introduction: Making TF Representation

- We can consider various Time Frequency representations, such as Spectrogram, Wigner-Ville distribution, ...
- The TF representation is a function of t and f. The topography of white Gaussian noise would appear as hills and valleys in this representation, while the topography of signal would look like ridges.
- Convolving the TF map with a Gaussian kernel will allow us to look for ridge features which we call tracks.
- Second derivative of profile perpendicular to track must exceed upper threshold at some point on track and lower threshold on all of the track







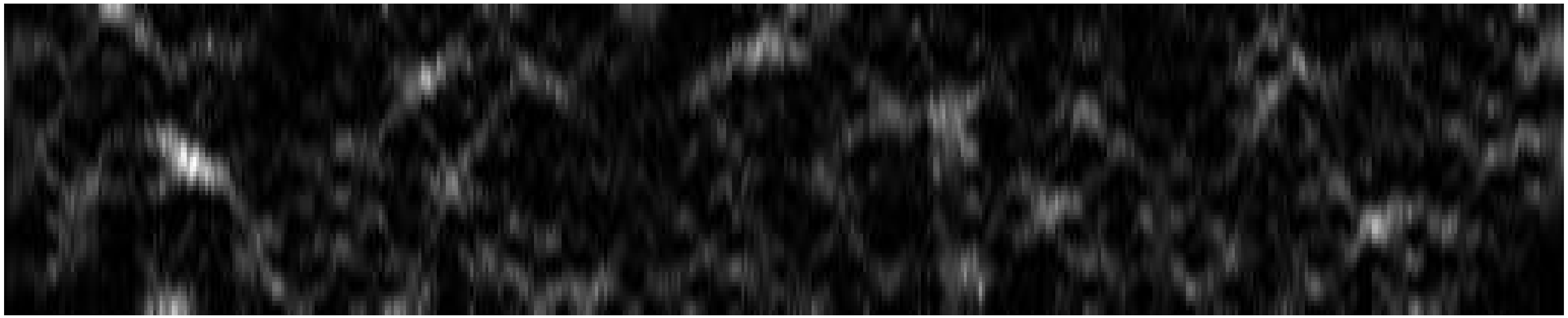
Setting Tracksearch Parameters

- Map Duration in Points
- Number of Frequency Bins in TF Representation
- Window Size
- Window Type

- Curve Start Threshold λ_h
- Curve Member Threshold λ_l
- Gaussian kernel width σ
- Threshold Statistics
  - Integrated Power Threshold
  - Curve Length(Points) Threshold

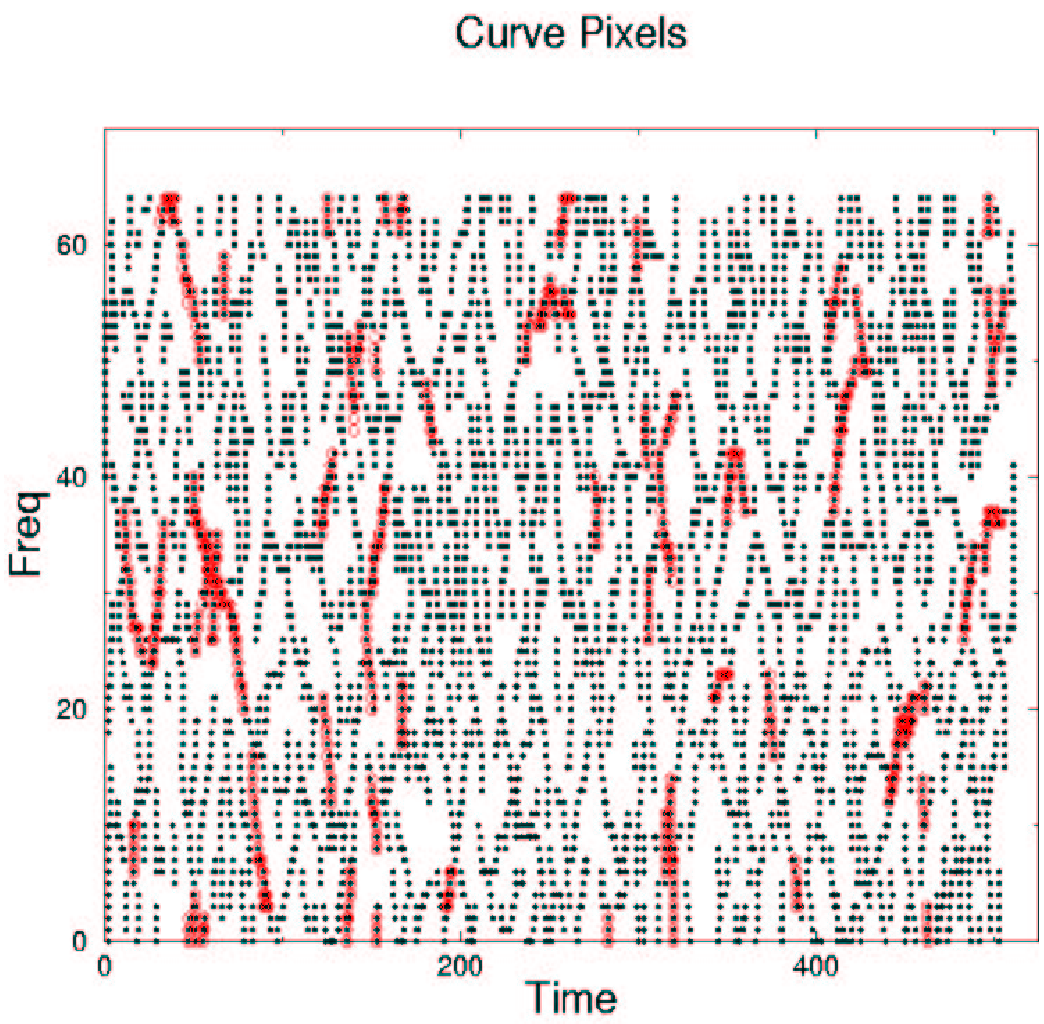
Setting up TF Representations

- Number of Frequency Bins: 64
- Number of Map/Segment/Data Set Length: 512
- TF Representation: Spectrogram
- Window Size: 17
- Window Type: Rectangular
- FFT Length: 128
- Overlap: 127
- σ:1



Tracksearch Setting Parameters

- Used unit variance Gaussian white noise
- Only a small subset of λ values explored
- The four λ_h values chosen corresponded to Tracksearch connecting 1% 5% 10% 15% of the total possible map pixels are connected in arbitrary curves.
- For each λ_h we take fractional values of $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and $\frac{1}{16}$ to determine λ_l.



λ_h values: 15%,1%

λ_l values: 1/2,1/2

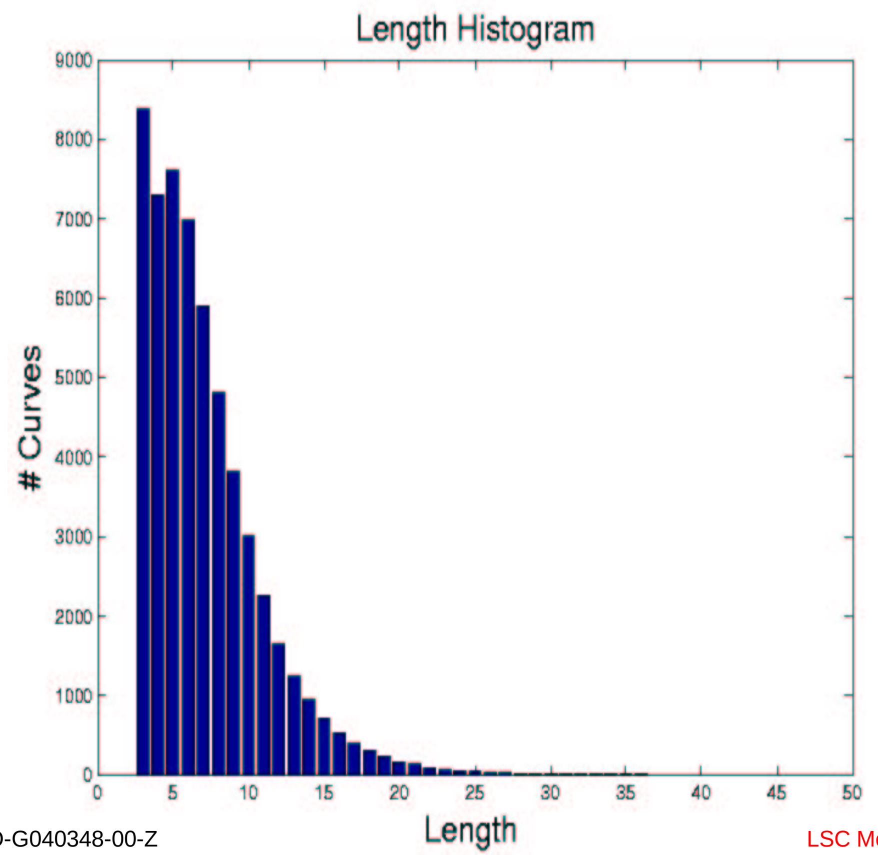
False Alarm Thresholds

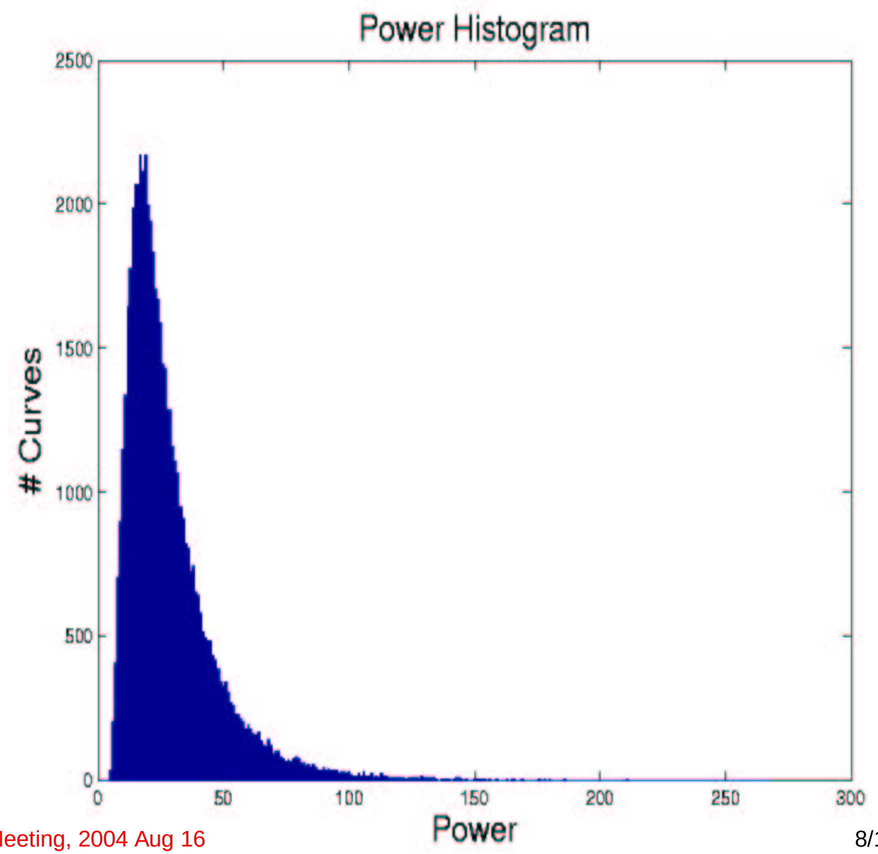
- Explore power and length as detection statistics
- Set false alarm rate to be 1%
- Tracksearch false alarm rate is defined as one or more curves that exceed specified thresholds are present in the current map

λ_h	λ_l	Power	Length
20%	1/16	221	55
20%	1/8	221	55
20%	1/4	221	55
20%	1/2	221	55
10%	1/16	221	54
10%	1/8	221	54
10%	1/4	221	53
10%	1/2	221	47
5%	1/16	228	52
5%	1/8	228	52
5%	1/4	228	52
5%	1/2	222	45
2%	1/16	213	49
2%	1/8	213	49
2%	1/4	213	44
2%	1/2	212	37

Setting False Alarm Thresholds

- False alarms set by Monte Carloing 1000 TF maps of white Gaussian noise with each λ_h and λ_l pairing
- Construct histograms of curve length and integrated power separately
- Sum over histogram bins such that we ignore only 1% percent of the *maps*





Detection Rates

- Estimate detection rate by Monte Carloing maps with injected signals in noise
- Choose SNR for signals by requiring that we had an approximate 85% detection rate for λ_h of 2% and λ_l of 1/2
- For our trials we picked a calibrating SNR for the power and length threshold separately
- These value for the SNR are 15 for power threshold and 23 for length threshold
- Then using these SNR and threshold we perform 1000 trials across all the possible λ_h and λ_l pairs to test detection efficiency

Detection Rates Continued

Power False Dismissal

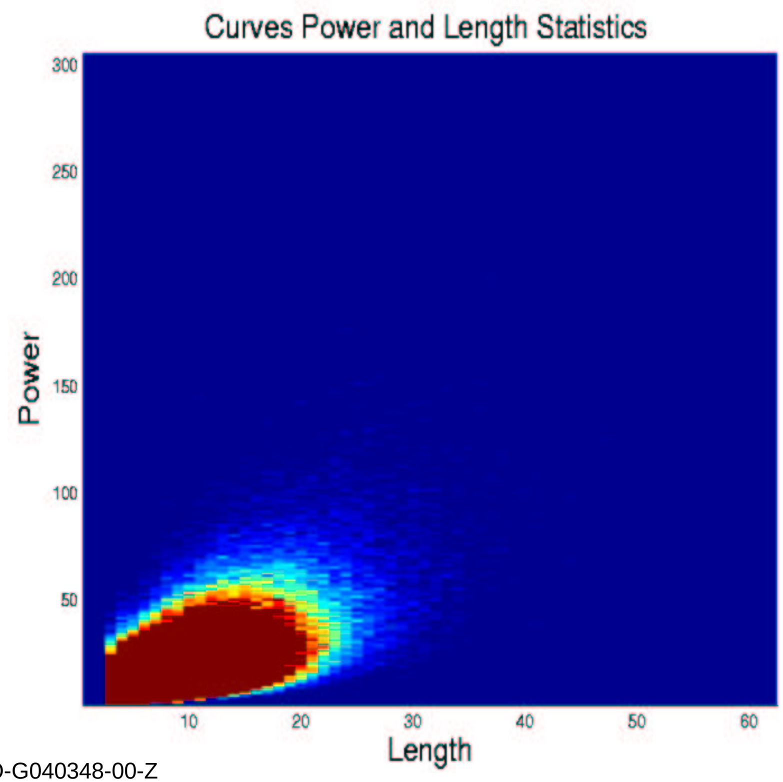
Detection Rate	λ_h	λ_l
70.8	20%	1/16
70.8	20%	1/8
70.8	20%	1/4
70.8	20%	1/2
71.1	10%	1/16
71.1	10%	1/8
71.1	10%	1/4
71.0	10%	1/2
67.3	5%	1/16
67.3	5%	1/8
67.3	5%	1/4
69.3	5%	1/2
75.0	2%	1/16
75.0	2%	1/8
75.0	2%	1/4
75.5	2%	1/2

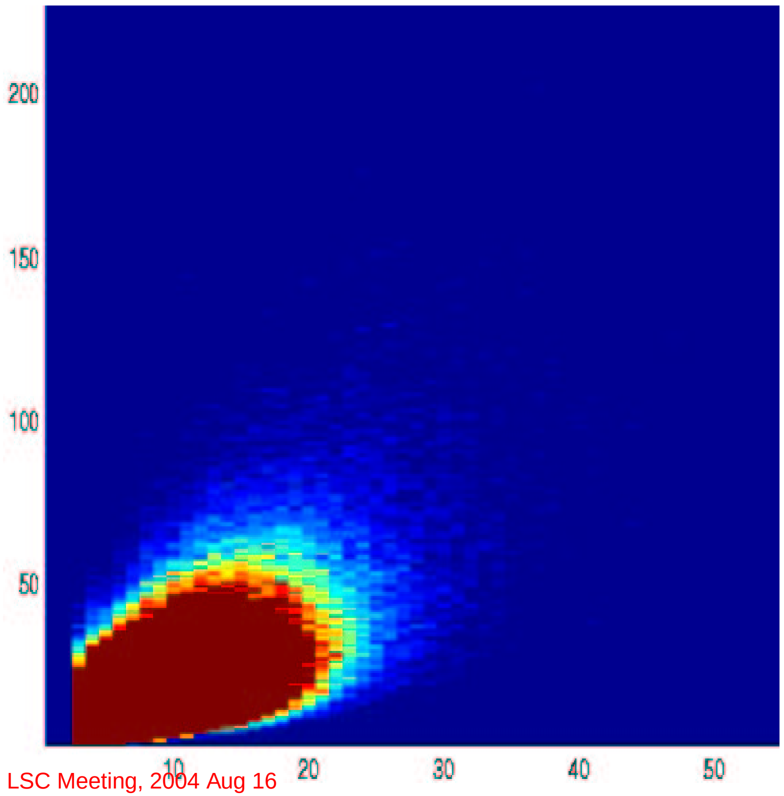
Length False Dismissal

Detection Rate	λ_h	λ_l
44.9	20%	1/16
44.8	20%	1/8
44.7	20%	1/4
44.5	20%	1/2
46.5	10%	1/16
46.4	10%	1/8
48.0	10%	1/4
63.6	10%	1/2
51.2	5%	1/16
51.2	5%	1/8
51.1	5%	1/4
68.7	5%	1/2
58.9	2%	1/16
58.9	2%	1/8
71.4	2%	1/4
87.3	2%	1/2

Tracksearch Relationships

- The thresholds of power and length with pure whiten noise background seem to be directly correlated
- The apparent correlation for white noise would most likely be altered when working in a colored noise background
- The power threshold appears to be less sensitive to the changes in the λ values than the length threshold
- There may be come combination of power and length that can improve detection rates



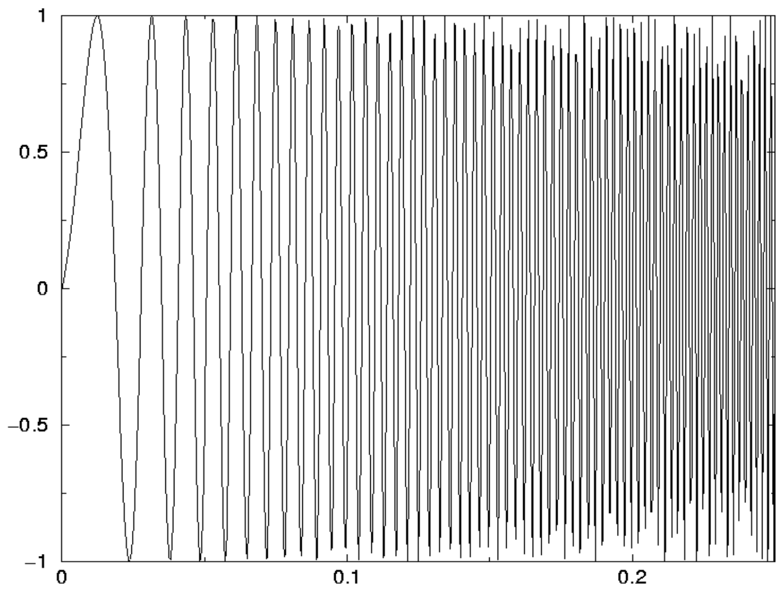


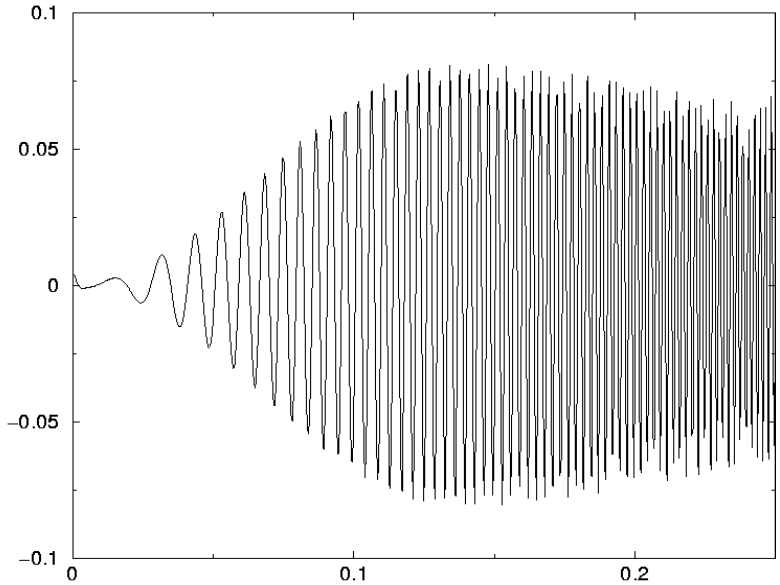
Improvements

- Initial exploration is small, need more thorough exploration of parameter space, we may find a *sweet spot* in our parameter space
- Disconnected curves could possibly belong to the same track. Joining neighboring curves which are within the Gaussian kernel width σ parameter could be connected to construct meaningful tracks.
- Every data segment could be run though multiple TF representations to try to find various signals which don't appear in a normal spectrogram
- Everything will need to be done on real IFO noise
- Tracksearch will need to be run in coincidence to help rule out noise tracks that should not be coincident in the interferometer data streams

Colored Noise Alternative

We start with a defined Noise Power Spectrum $N(f)$. We take our know signal S(t) and transform into the Fourier domain to get $S(f)$. Then we can define our colored signal in the Fourier domain as $\frac{S(f)}{N(f)}$. This quantity if inverse FFTed to yield our time domain representation of the signal which has been *colored* by the LIGO Noise PSD. It is this quantity that we inject into white noise to effectively simulated whitening a colored data stream plus signal. We chose this alternative because it was easy to implement. In the future tracksearch will whiten incoming data and this testing method will be abandoned.





Sample TF Maps