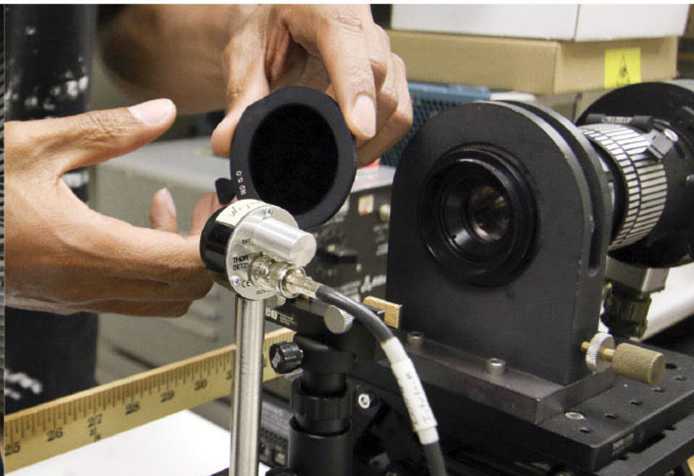
NEVADA NATIONAL NNSS SECURITY SITE

Exhaustively searching PDV waveforms using fast circular-convolution/cross-correlation to perform improved extraction of dynamic surface velocity



Template matching dynamic frequency waveforms to observed PDV data

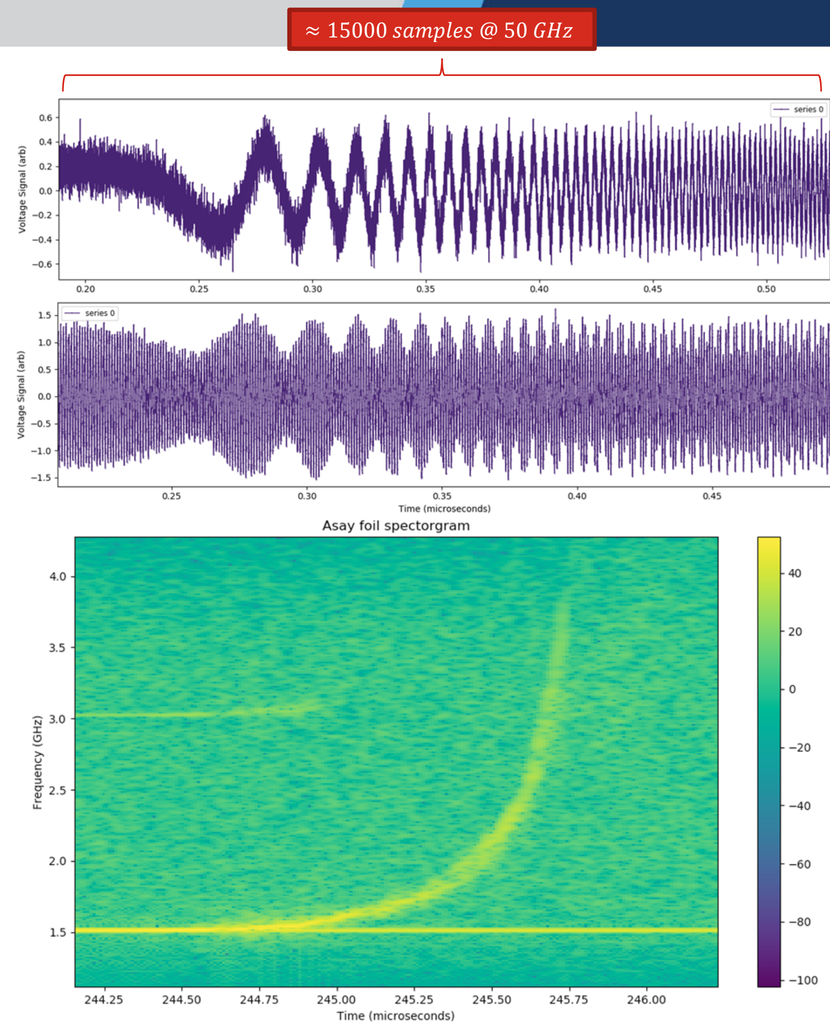
Daniel Champion

The Nevada National Security Site is managed and operated by MSTS under contract number DE-NA0003624.

This work was done by Mission Support and Test Services, LLC, under Contract No. DE-NA0003624 with the U.S. Department of Energy. DOE/NV/03624--1575

Dynamic/accelerating surface signals

- A traditional frequency domain analysis method for PDV signals transforms the waveform data into frequency components using the discrete time Fourier transform (DFT):
  - Let $\{\boldsymbol{x_n}\} = x_0, x_1, \ldots, x_{N-1}$ be our PDV waveform data (voltages), then $\mathcal{F}(\boldsymbol{x}) = \boldsymbol{X}$ with
  $$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi kn/N}$$
  - The inverse transform, $\mathcal{F}^{-1}(\boldsymbol{X}) = \boldsymbol{x}$, is given by:
  $$x_n = \mathcal{F}^{-1}(\boldsymbol{X})_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k \cdot e^{i2\pi kn/N}, \qquad n \in \mathbb{Z}$$
  - The Fourier coefficients X_k have many interpretations, but a relevant one for this talk is:
    - *X_k is the cross correlation (dot product) of the input sequence $\{\boldsymbol{x_n}\}$ and a complex sinusoid at frequency $\frac{k}{N}$*
  - That is, when we are interpreting frequency domain transforms of PDV data, we are viewing the data as it appears under a decomposition/comparison with respect to a discrete collection of constant frequency sinusoids.
- Accelerating surfaces are common in some PDV applications (Asay foils/windows, ringing surfaces, HE gas accelerations,…) and manifest as dynamically changing frequency PDV waveforms.
  - **Are there advantages in considering non-constant frequency components as an alternative to the DFT? (accuracy, low-velocity extraction, DFT artifacts)**



Waveform model for constant velocity surfaces

- A well used and studied (e.g. Dolan 2010) simple model for a homodyne PDV waveform of a constant velocity surface is given by:

$$s_k = \cos(2\pi f T k + \delta) + \sigma R_k$$

  - f=frequency, δ=phase offset, T =sample time, σ=noise scale parameter
  - R_k could be drawn from a standard normal distribution

- The velocity of the surface is given by the familiar expression:

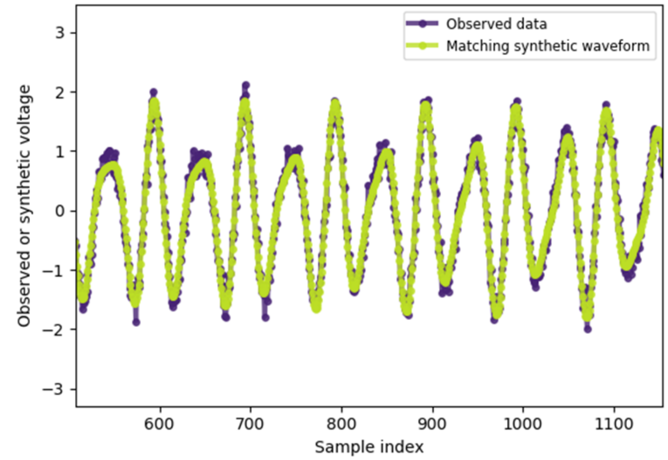
$$v = \frac{f \cdot \lambda}{2}$$

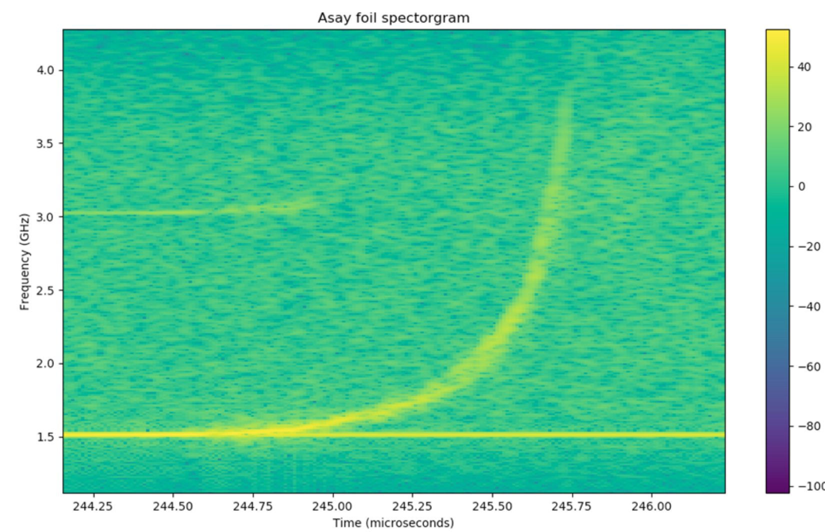
- For a heterodyne PDV waveform, we extend the above by adding terms to include the baseline frequency f_0 of the system as follows:

$$s_k = A_{sig} \cdot \cos(2\pi(f + f_0)Tk + \delta) + A_{base} \cdot \cos(2\pi f_0 Tk) + \sigma R_k$$

  - A_{sig}=surface signal amplitude, A_{base}=baseline/reference signal amplitude, σ=combined noise scale parameter

- The velocity is given by the same expression as above after f is recovered (and f_0 removed) using a method such as the DFT.





Waveform model for *non-constant* velocity surfaces

- Consider the ***instantaneous phase*** of a general (non-constant velocity) PDV waveform: $\boldsymbol{\varphi(t)}$. The instantaneous phase is related to ***instantaneous frequency*** according to the expression:

$$\boldsymbol{\varphi(t) = \varphi(0) + \int_0^t \omega(\tau)d\tau = \varphi(0) + \int_0^t 2\pi \cdot f(\tau)d\tau}$$

- Thus, for a homodyne PDV system, a non-constant waveform could manifest according to the following:

$$\boldsymbol{s(t) = cos\left(\varphi(0) + \int_0^t 2\pi \cdot f(\tau)d\tau\right) + \sigma R_k}$$

Heterodyne has additional terms, but this is the main idea

- The DFT leverages the local approximation of the above to a constant velocity/constant frequency waveform (previous slide) to identify the dominant frequency and thus velocity of the surface.

- **Why don't we brute force extend the interpretation of the Fourier coefficients from the prior slide:**
  - *"X_k is the cross correlation (dot product) of the input sequence $\{x_n\}$ and a complex sinusoid at frequency $\frac{k}{N}$"*

To something like this:

- *For a suitable collection of instantaneous frequency functions $f(t)$, generate synthetic waveforms related to the base form:*

$$\boldsymbol{s(t) = A \cdot cos\left(\varphi(0) + \int_0^t 2\pi \cdot f(\tau)d\tau\right)}$$

  and cross-correlate with our PDV waveform.
- *Identify the **best matching synthetic waveform** and read off the instantaneous frequency at the best-match location.*

Waveform model for *non-constant* velocity surfaces

For a suitable collection of instantaneous frequency functions $f(t)$, generate synthetic waveforms of the form $s(t) = cos\left(\varphi(0) + \int_0^t 2\pi \cdot f(\tau)d\tau\right)$ and cross correlate with our observed PDV waveform. Identify the best matching synthetic waveform and read off the instantaneous frequency at the best-match location.

- ***Some obvious questions:***
  1. *What collection of instantaneous frequency functions?*
  2. *Heterodyne would add a lot of complexity, can we handle that?*
  3. *This sounds like argmax extraction, isn't that worse than Gaussian?*
  4. *Yeah, but this can't be done efficiently, too much searching/compute!*
     - ***Actually, we can compute cross-correlations super fast, there is a math cheat code for computing them…***
       - *Not that fast! We don't have a supercomputer to do PDV work!!*
         - ***Hold my beer…***

Linear, quadratic, …

Adds amplitude, phase, and baseline frequency terms. Amplitude and ~~phase~~ will need to be part of the collection.

Yes, this is like "higher order" argmax extraction. The higher order mitigates the omission of a regression step

We can compute a single cross-correlation test in $N \cdot log(N)$ time where N is the length of the reference/synthetic signal we are testing.

- *Math "cheat code" for cross correlations:*
  - https://en.wikipedia.org/wiki/Discrete_Fourier_transform#Circular_convolution_theorem_and_cross-correlation_theorem

$$x * y_N = \mathrm{DTFT}^{-1}\left[\mathrm{DTFT}\{x\} \cdot \mathrm{DTFT}\{y_N\}\right] = \mathrm{DFT}^{-1}\left[\mathrm{DFT}\{x_N\} \cdot \mathrm{DFT}\{y_N\}\right],$$

$$\mathcal{F}^{-1}\{X \cdot Y\}_n = \sum_{\ell=0}^{N-1} x_\ell \cdot y_{(n-\ell) \bmod N}$$

With appropriate zero-padding of two waveforms: $\{x_n\}$, $\{y_n\}$ we can obtain cross-correlation results without circular-wrapping artifacts:

$$\{x'_n\} = [\{x_n\} \quad [0]_{|\{y_n\}|}]$$

$$\{y'_n\} = [[0]_{|\{x_n\}|} \quad \{y_n\}]$$

$$corr = ifft\left(conj\left(fft(\{x'_n\})\right) \cdot \{y'_n\}\right)$$

- python™

```
corr = scipy.signal.correlate(sig1,
       sig2, method = 'fft')
```

- MATLAB

```
corr = xcorr(sig1, sig2);
```

Instantaneous function collection (search space)

Recommendations (..of where to start, limited by time and creativity):

- *Relative heterodyne baseline –vs- signal amplitude:* $\boldsymbol{A}$
  - $A \in \{1.25^j \; for \; -7 \le j \le 7\}$
- *Instantaneous frequency rate of change:* $\boldsymbol{f_1}$
  - $f_1 \in \left\{\frac{1}{\lambda} \cdot 50 \cdot 1.3^j * 1e6 \;\; for \; 0 \le j \le 19\right\}$
- *Phase offset between baseline and surface signals:* $\boldsymbol{\varphi_0}$
  - φ_0 = ***evenly spaced from 0 to*** 2π ***in steps of*** ≈ 0.2

- *Synthetic signal (for a choice of* A, f_1, φ_0*):*
  - $T = [\Delta t \cdot i]_{i=0,\ldots,n-1}$
  - $F = f_0 - 100e6 + f_1 \cdot T$
  - $\Phi = \varphi_0 + \text{cumsum}(\Delta t \cdot 2\pi \cdot F)$
  - $V = F \cdot \frac{\lambda}{2}$
  - $S = A \cdot \cos(\Phi) + \cos(2\pi \cdot f_0 \cdot T)$

```
In [51]: 1.25**np.arange(-7,8,1)
Out[51]:
array([0.2097152 , 0.262144  , 0.32768   , 0.4096    , 0.512     ,
       0.64      , 0.8       , 1.        , 1.25      , 1.5625    ,
       1.953125  , 2.44140625, 3.05175781, 3.81469727, 4.76837158])
```

```
In [54]: (1./wavelength) * 50.*1.3**np.arange(20) / 1e-6
Out[54]:
array([3.22580645e+13, 4.19354839e+13, 5.45161290e+13, 7.08709677e+13,
       9.21322581e+13, 1.19771935e+14, 1.55703516e+14, 2.02414571e+14,
       2.63138942e+14, 3.42080625e+14, 4.44704812e+14, 5.78116256e+14,
       7.51551133e+14, 9.77016473e+14, 1.27012141e+15, 1.65115784e+15,
       2.14650519e+15, 2.79045675e+15, 3.62759377e+15, 4.71587190e+15])
```

```
In [55]: np.arange(0.0, 2.0 * np.pi, 0.2)
Out[55]:
array([0. , 0.2, 0.4, 0.6, 0.8, 1. , 1.2, 1.4, 1.6, 1.8, 2. , 2.2, 2.4,
       2.6, 2.8, 3. , 3.2, 3.4, 3.6, 3.8, 4. , 4.2, 4.4, 4.6, 4.8, 5. ,
       5.2, 5.4, 5.6, 5.8, 6. , 6.2])
```

- Produce constant acceleration synthetic surface waveforms with no added noise.
- Clip these to a reasonable velocity range (using V), and concatenate if you so desire
- The negative 100 MHz used in the F computation is so that we can recover **unbiased** velocities. You could omit this to force positivity, ***but you shouldn't do that.***

One final surprisingly useful trick

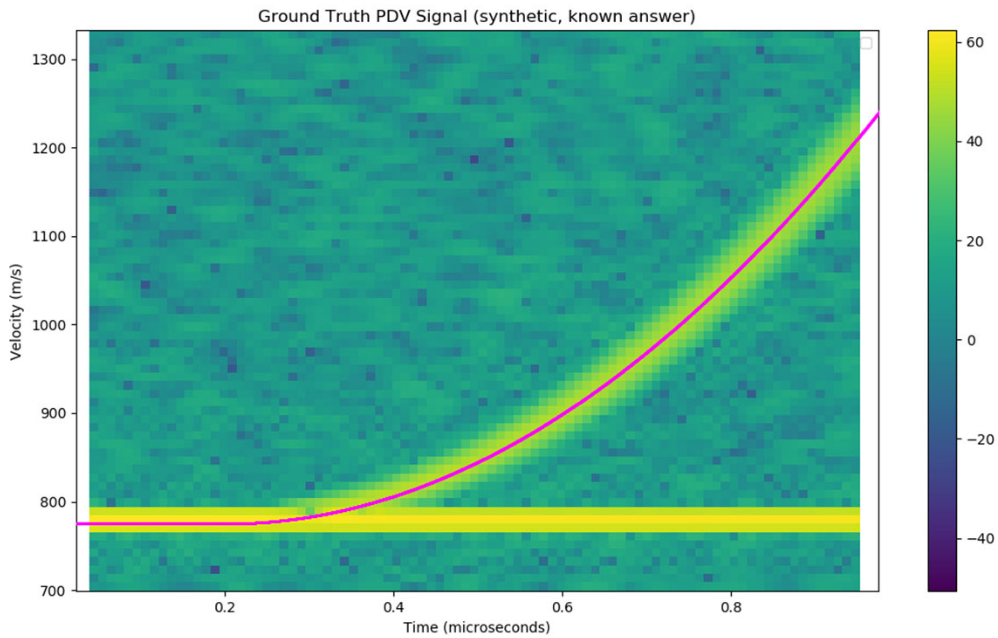
- The synthetic waveforms were **detrended** (subtract mean) and **“binarized”** using the sign function prior to cross correlation:

$$sign(x) = \begin{cases} 1 & x > 0 \\ -1 & x < 0 \\ 0 & x = 0 \end{cases}$$

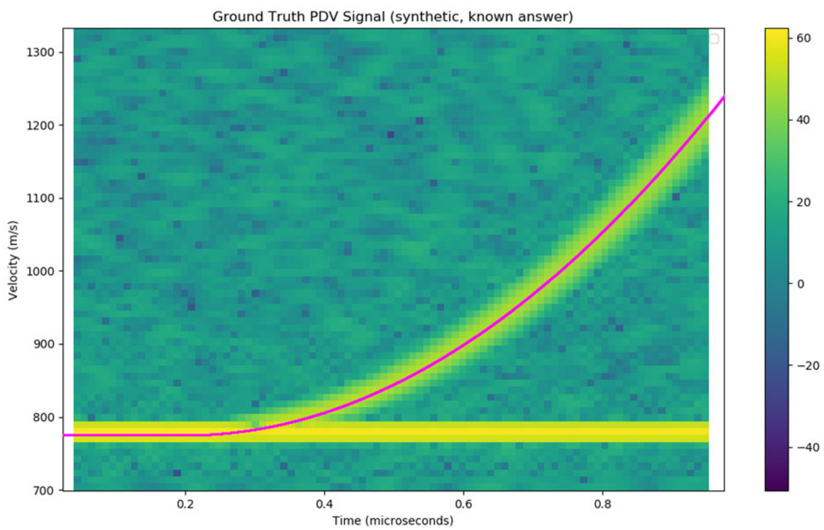
- This resulted in improved and more robust cross-correlation statistics.
- Attempts without this transformation resulted in many outliers in the recoveries because the cross-correlation scores can be impacted/inflated by fluctuations in waveform amplitude.
- There is probably a better normalization based approach than this? Thoughts?

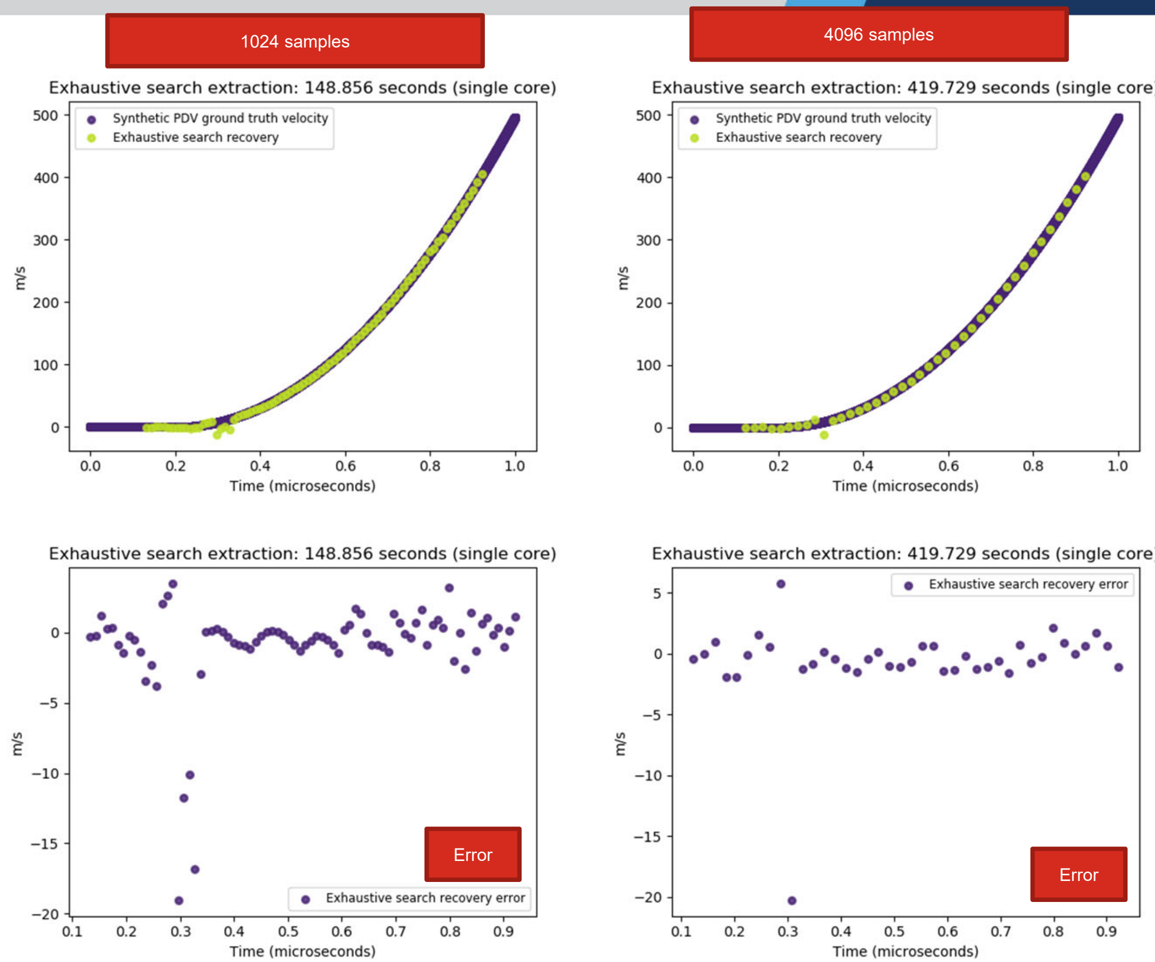
Tests on synthetic data (known answer)

- Constant jerk (linearly accelerating) surface.
  - Use a quadratic term in the computation of F on the previous slide.
  - Tested noise-free, 0.1, 0.5 proportional noise levels (compared to the heterodyne baseline signal amplitude)
  - Tested a relative signal amplitude of 0.1, 0.333 (10% and 33.3% that of the heterodyne baseline signal)
  - Extended as well as shortened search spaces for the instantaneous frequency function collection.
  - Between 2-minute and 10-minute computation time (single core)
  - GPU accel is possible: potential for 100x speed-up
  - 1024 and 4096 waveforms samples per recovery were investigated
  - Can make use of velocity bounds (ROI) to decrease run time and outliers:
    - Spurious pre-motion velocities
    - Harmonics

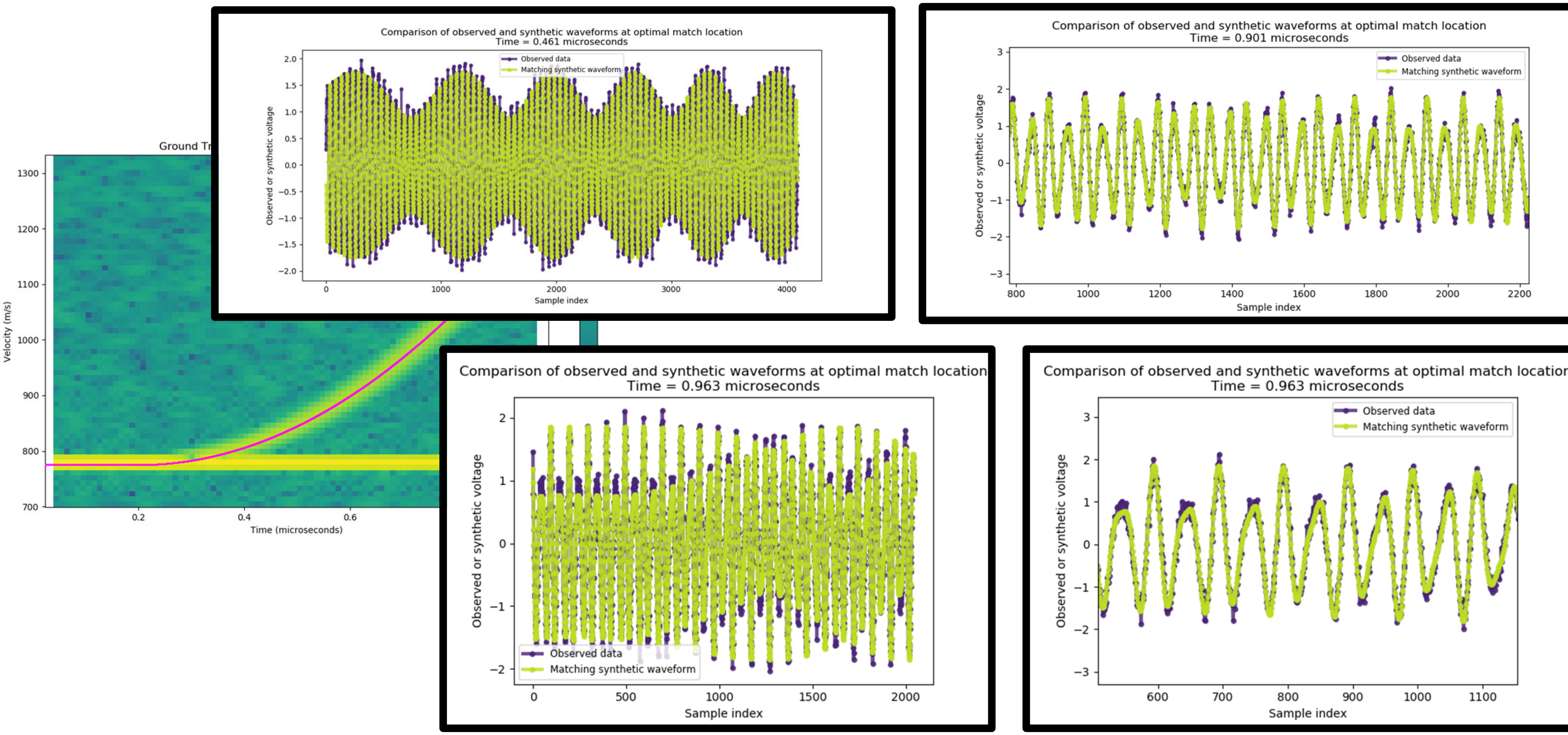


Tests on synthetic data (known answer): extractions



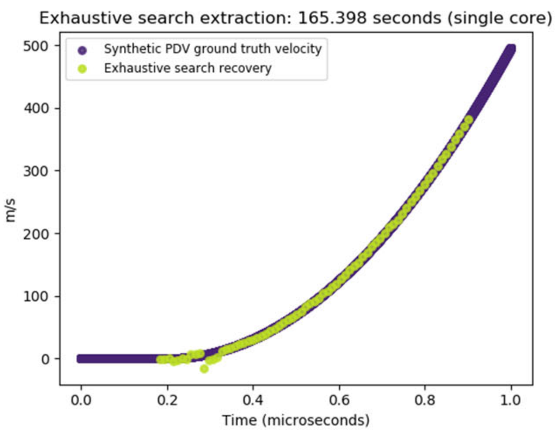


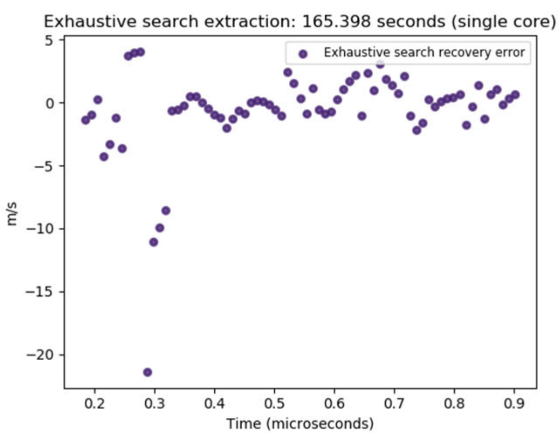
Tests on synthetic data (known answer): waveform comparisons



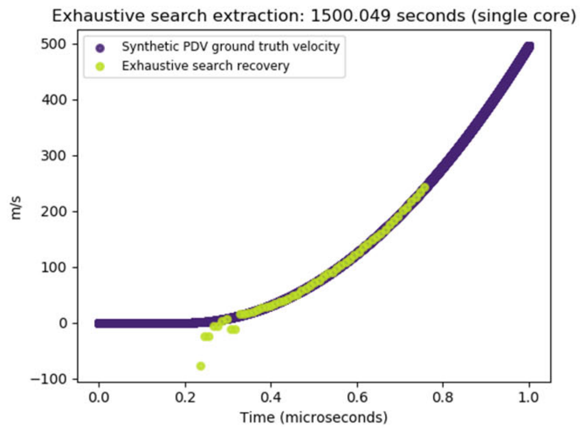
More Noise (5x), quick vs extensive search

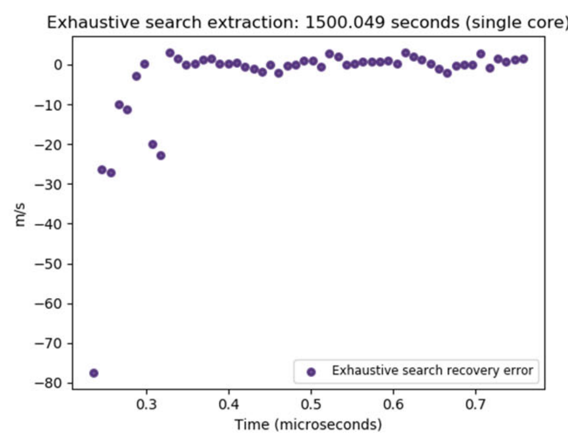
Smaller search space, 4096



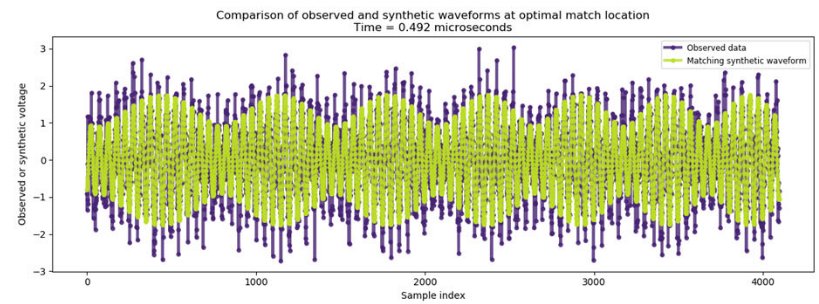


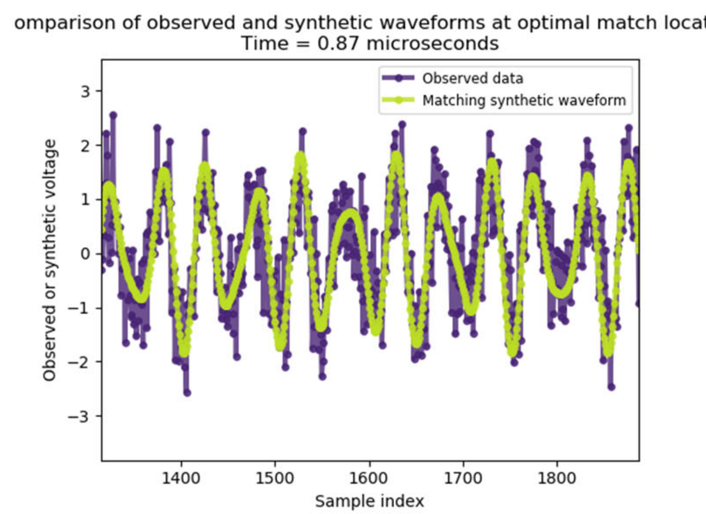
Larger search space, 4096





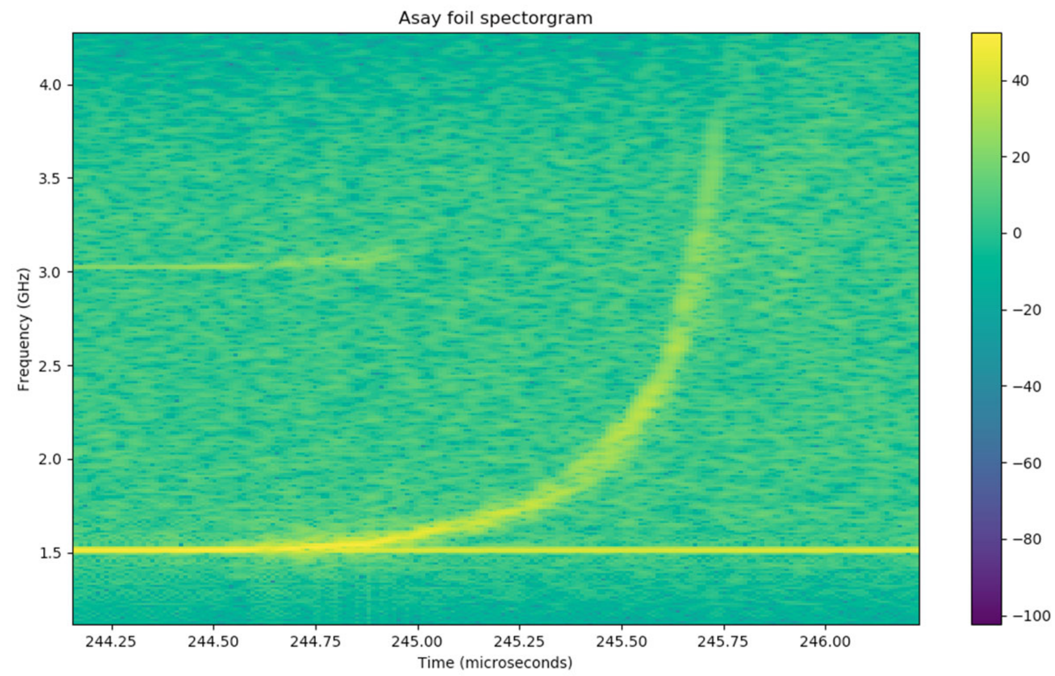
Waveform comparisons





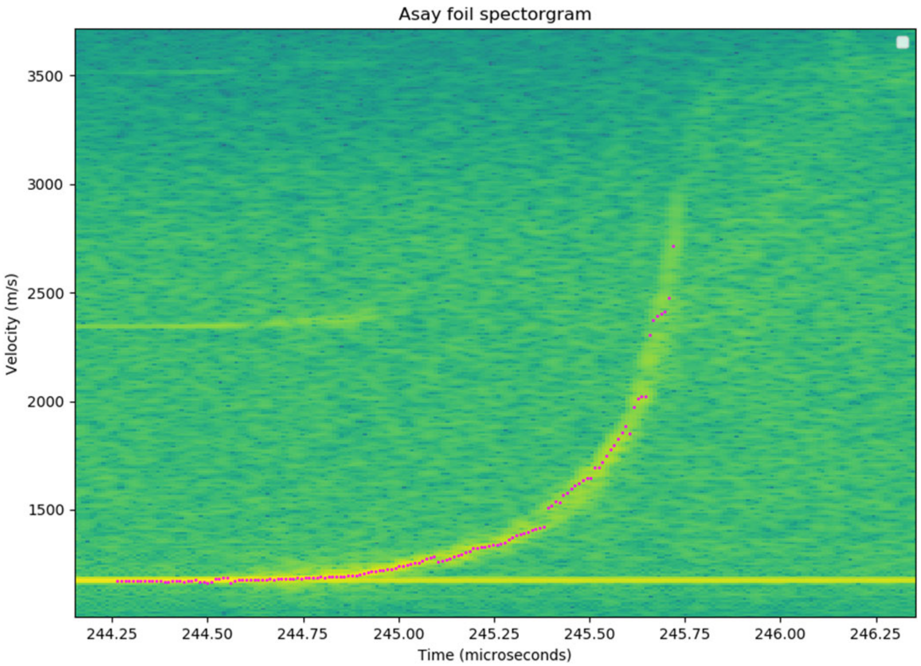
Tests on real Asay foil data

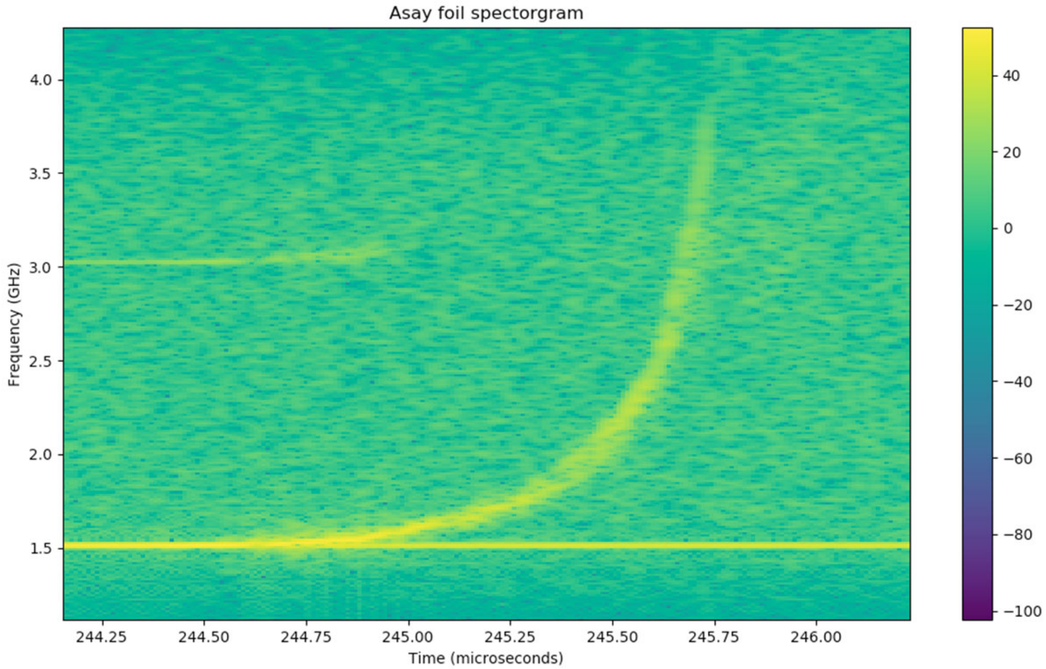
- Constant jerk (linearly accelerating) surface.
  - STL Foil fielded on coupon experiment
  - Used smaller search space for the instantaneous frequency function collection.
  - ~2-minute computation time (single core)
  - 4096 waveform samples per recovery
  - Made use of max/min velocity bounds (ROI) to decrease run time and outliers:
    - **Spurious pre-motion velocities**
    - **Harmonics**
  - *Bandpass filtered the input waveform to remove high and low frequency components (harmonic of the heterodyne baseline appeared above the foil)*



Tests on real Asay foil data

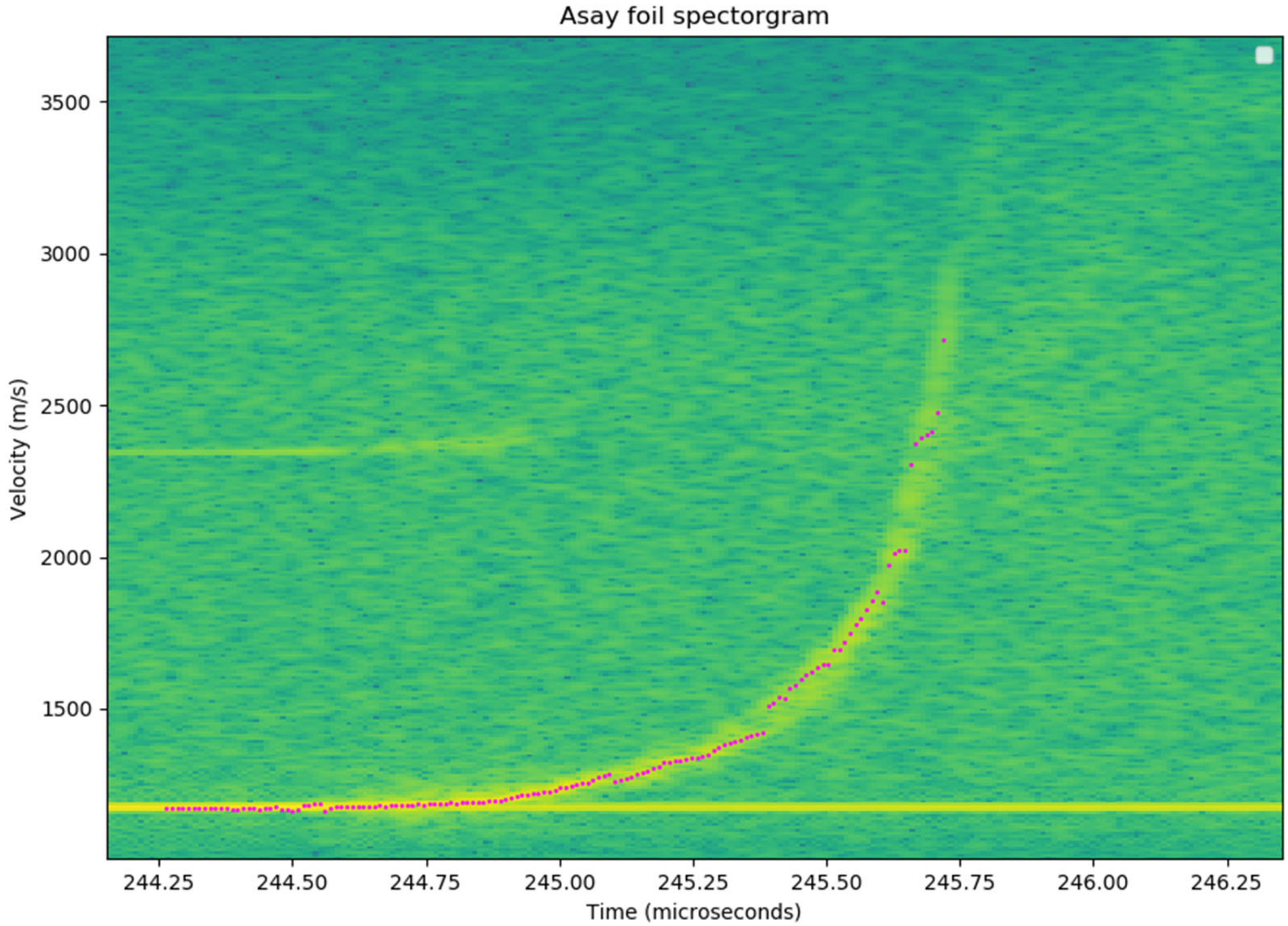
Velocity recovery, smaller search space, 4096





Tests on real Asay foil data

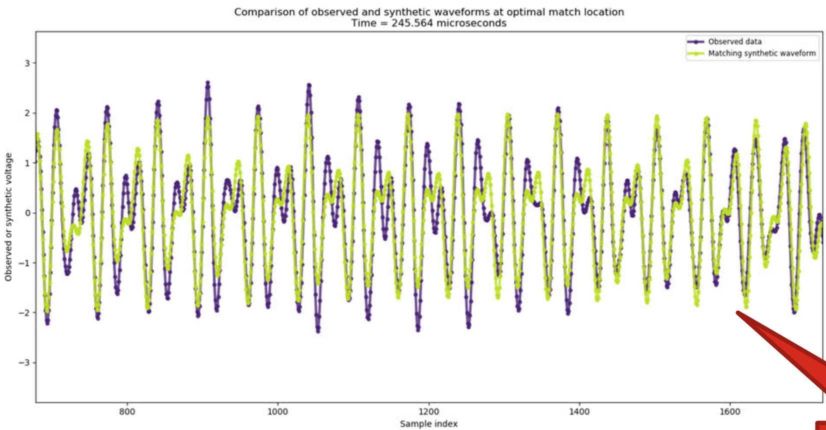
Velocity recovery, smaller search space, 4096

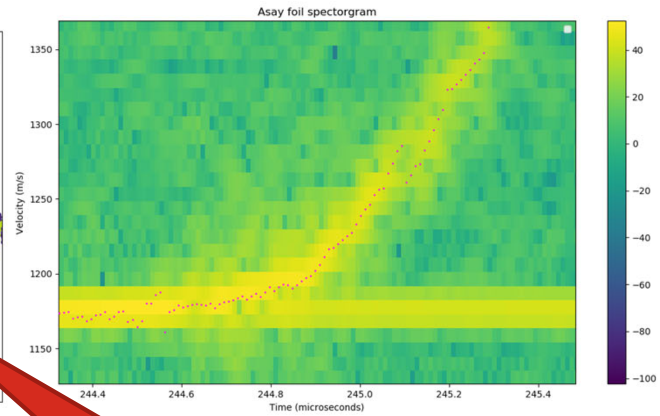


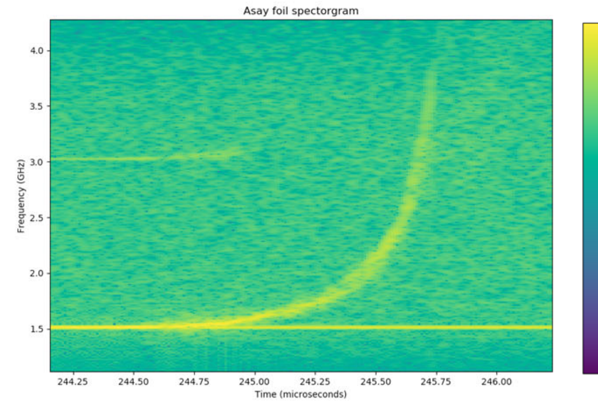
Tests on real Asay foil data

Waveform comparisons

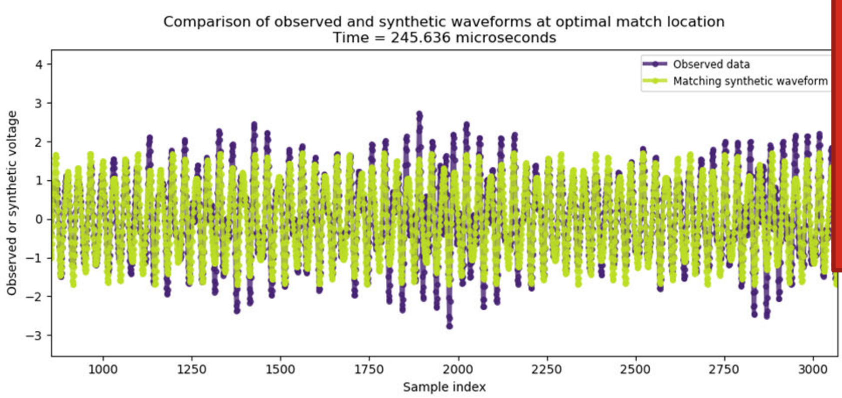
Low-velocity recovery and detial

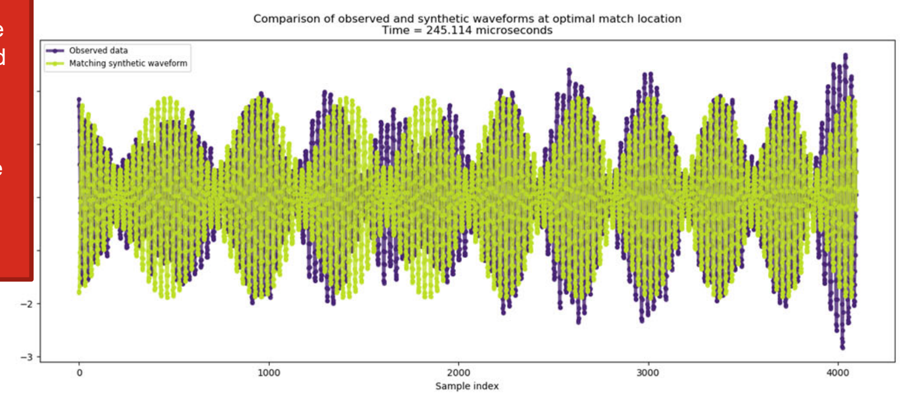






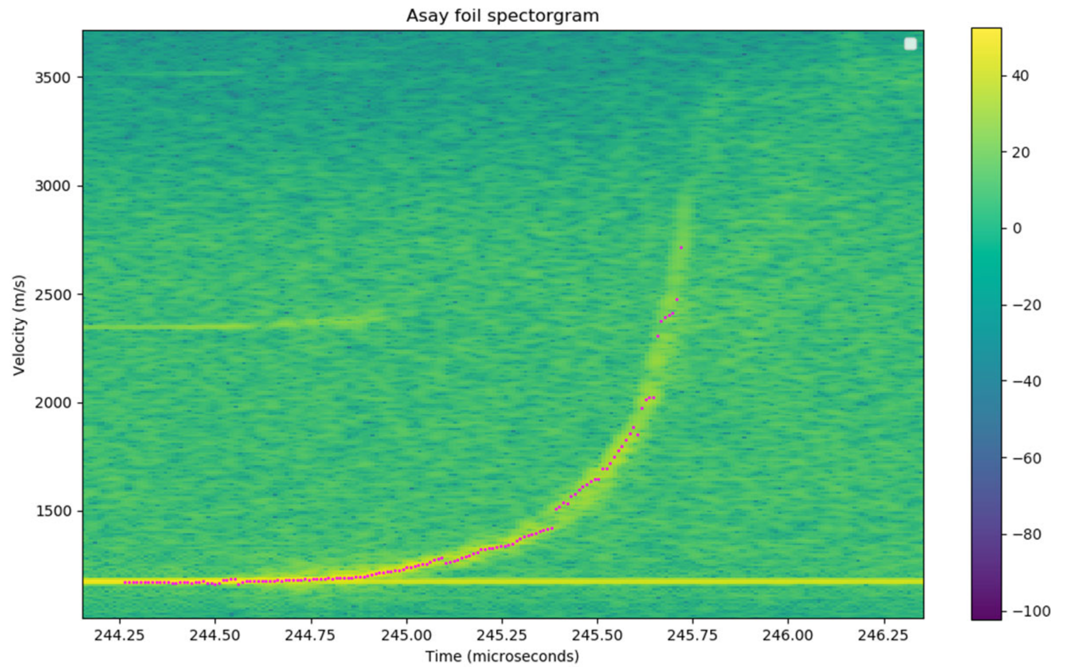
We could extract the recovered signal and obtain a residual waveform!!! Could this tell use more information (multiple surfaces, fractures, multiple velocities)





Conclusions and next steps

- The Exhaustive cross-correlation methodology is robust to the tested noise levels and performs on real foil data (med-good SNR)
  - Need more testing on low-SNR and atypical surface features
- In rapid accelerating surfaces does this method out-perform traditional approaches?
  - Ringing, some foils/windows, HE drives
  - Can prevent too-long FFT windows from negatively impacting the analysis results
- Include a battery of constant velocity synthetic waveforms in the instantaneous frequency collection.
- Deceleration needs to be added for ringing (doubles the search space).
- Hypothesis: traditional FFT-based extraction will returned slightly biased results in accelerating surfaces, this method ***could*** reduce that bias.



References

References

- PDV uncertainty and parametric waveform:

  - Dolan, D. H. "Accuracy and precision in photonic Doppler velocimetry." Review of Scientific Instruments 81.5 (2010): 053905.

- FFT's, circular convolutions:

  - Oppenheim, Alan V.; Schafer, Ronald W.; Buck, John R. (1999). Discrete-time signal processing (2nd ed.). Upper Saddle River, N.J.: Prentice Hall. p. 571. ISBN 0-13-754920-2.
  - McGillem, Clare D.; Cooper, George R. (1984). *Continuous and Discrete Signal and System Analysis* (2 ed.). Holt, Rinehart and Winston. pp. 171–172. ISBN 0-03-061703-0

- Asay foils and equations of motion (acceleration mechanics):

  - Tregillis, Ian Lee, and Alan Kent Harrison. *Notes on Asay Foil Analysis*. No. LA-UR-20-29695. Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2020.
  - Tregillis, Ian Lee, Aaron Koskelo, and Alan K. Harrison. "Analytic solutions for Asay foil trajectories with implications for ejecta source models and mass measurements." *Journal of Applied Physics* 130.12 (2021): 124504.
  - Asay, J. R., L. P. Mix, and F. C. Perry. "Ejection of material from shocked surfaces." *Applied Physics Letters* 29.5 (1976): 284-287.

Acknowledgements:

- The authors thank Jerry Stevens and Brandon LaLone and the boom-box team at STL for their ongoing assistance with experimental validation and theoretical physics discussions on Asay foils, Asay windows and low velocity extractions. Furthermore, the author thanks this team for the Asay foil data used in this presentation.
- The author also thanks Andrew Robinson for discussions and encouraging progress on this work.

NEVADA NATIONAL NNSS SECURITY SITE

Questions?

Thank you for your consideration!

Contact:
Daniel Champion:
champidj@nv.doe.gov

NEVADA NATIONAL NNSS SECURITY SITE