

ALPSS: A program for automated analysis of photonic Doppler velocimetry spall signals

(**ALPSS: AnaLysis of Photonic Doppler Velocimetry Spall Signals**)

Jacob M. Diamond, Samuel Salander, K. T. Ramesh

February 6th, 2023

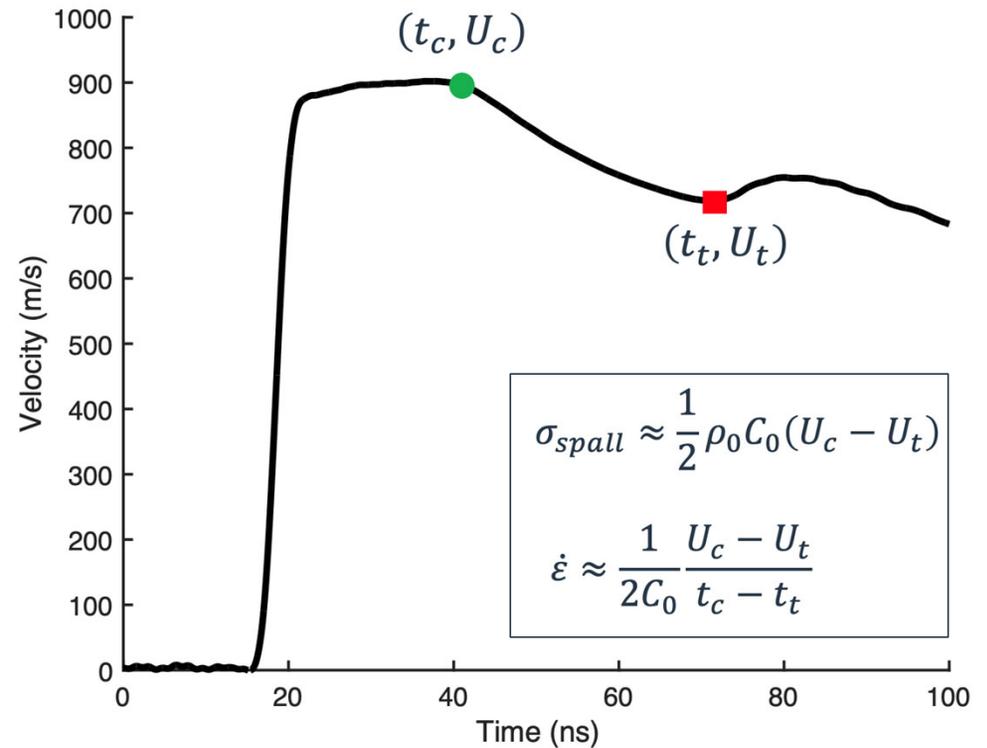
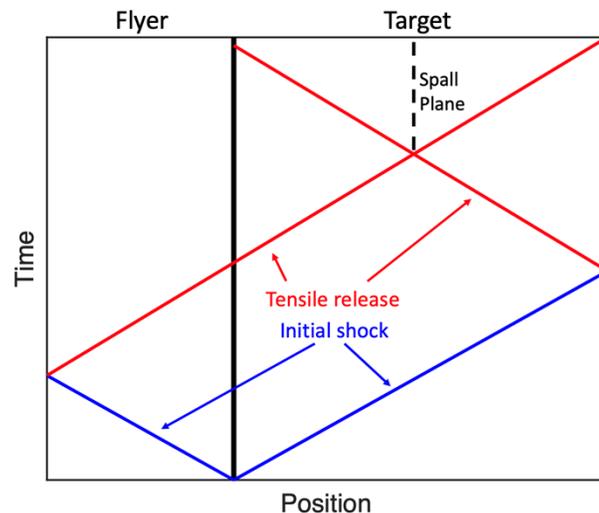
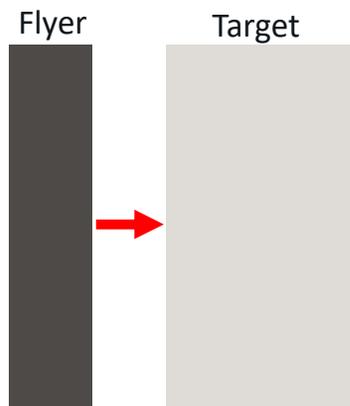


W911NF-22-2-0014



A Primer on Spall Analysis

- Interaction of release waves cause material failure in dynamic tension
- Can estimate the stress at failure using free surface velocity

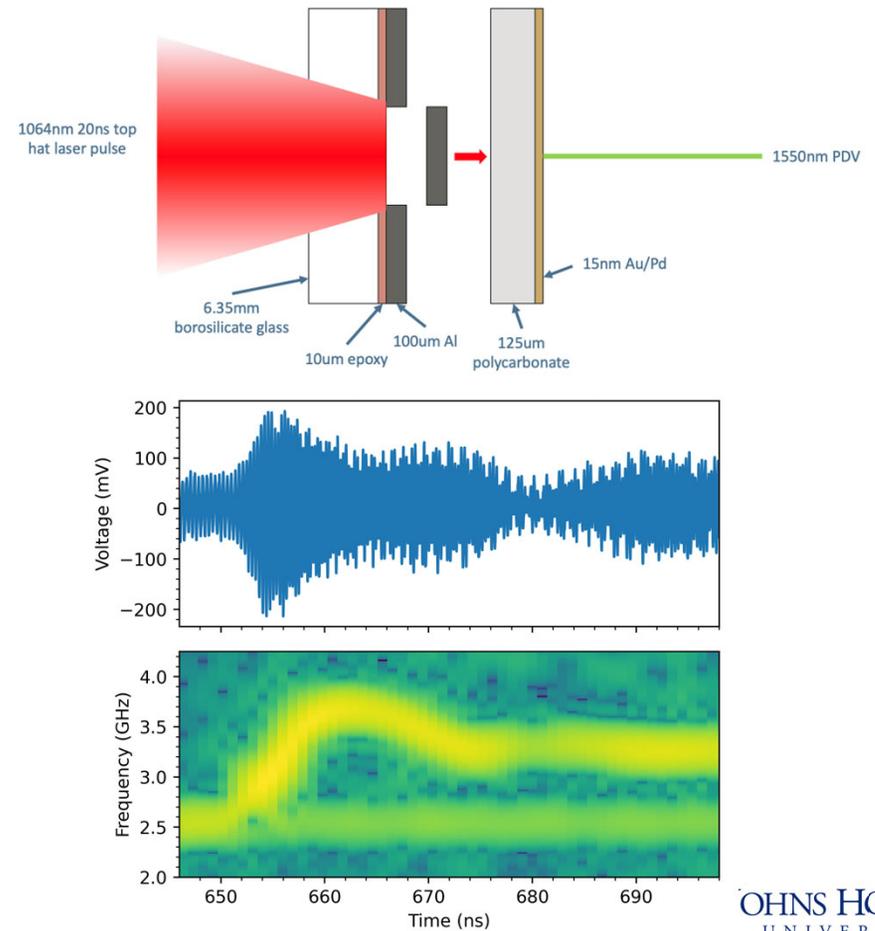




Benchtop Laser-Driven Flyers

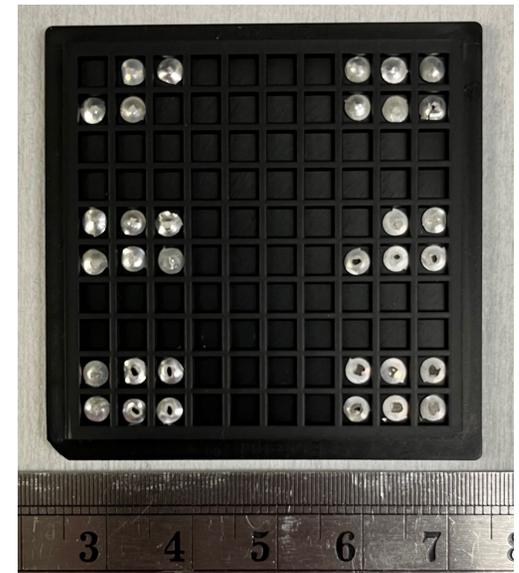
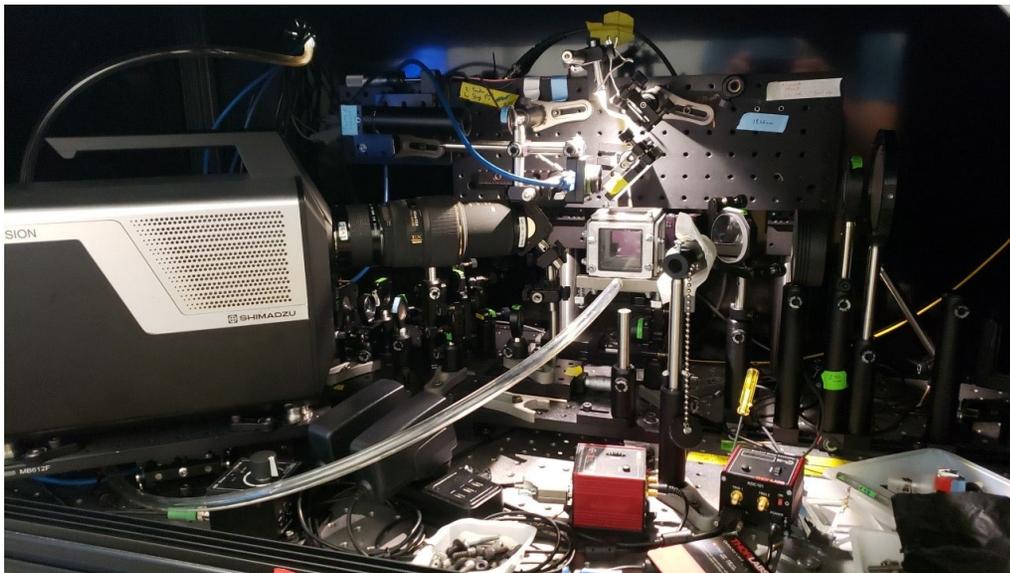
Dr. Debjoy Mallick

- Adapted for spall experiments by Debjoy Mallick (JHU, 2019)
- Can perform hundreds of spall shots on nanosecond timescales
- Already difficult to keep up with the data analysis



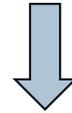
JHU Laser-Driven Flyer Setup

Personal best of 300 shots in 12 hours over two days



Motivation

Autonomous system is being designed for thousands of shots per day with 8-point PDV

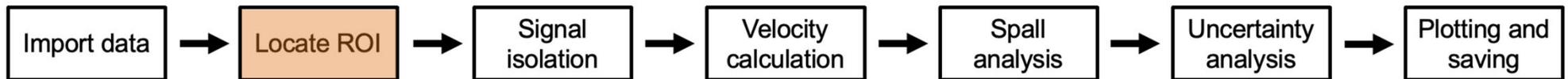


Analysis on this volume of data will be impossible with existing packages

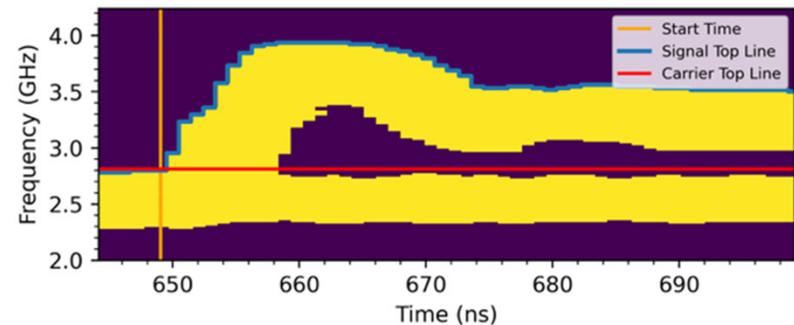
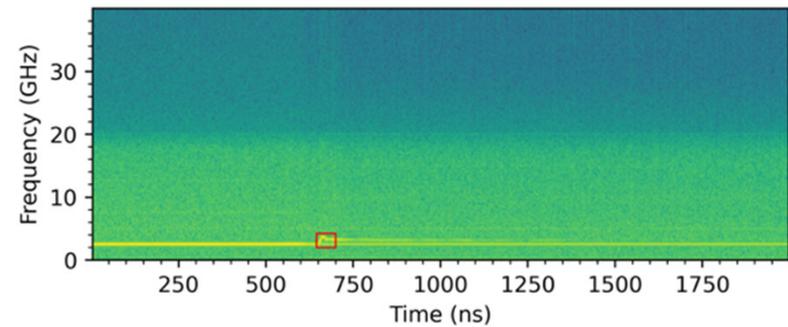


Need a new program that can keep up with high-throughput experiments

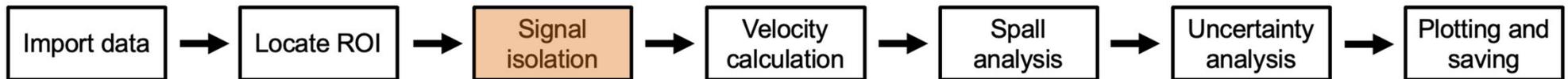
Program Workflow



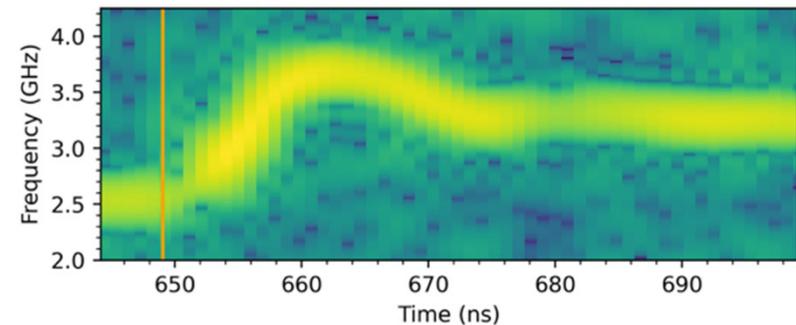
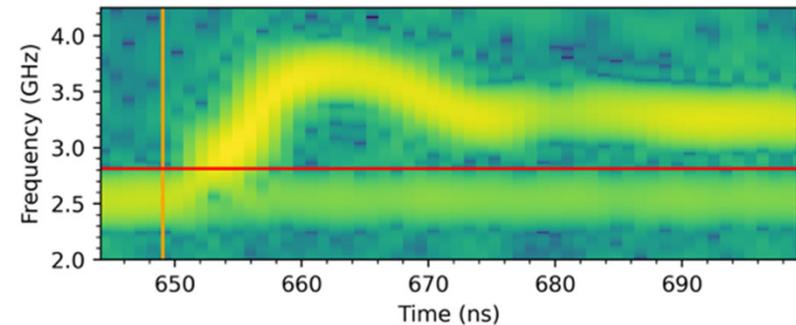
- Calculate spectrogram with short time Fourier transform (STFT)
- Generate binary mask
- Find intersection of the signal with the carrier band



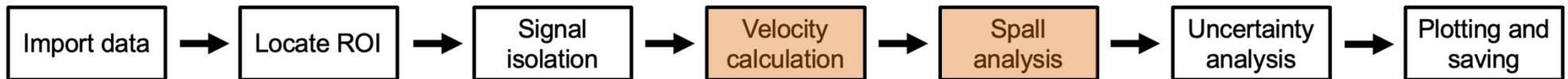
Program Workflow



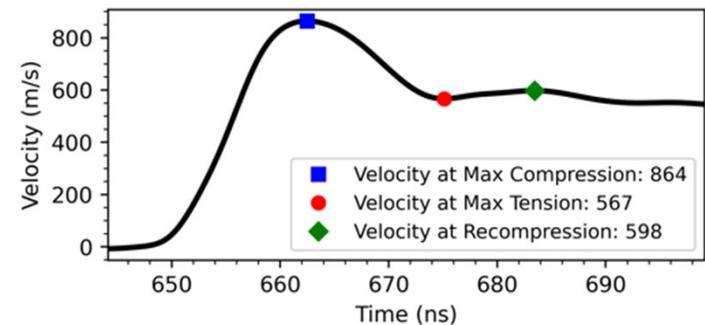
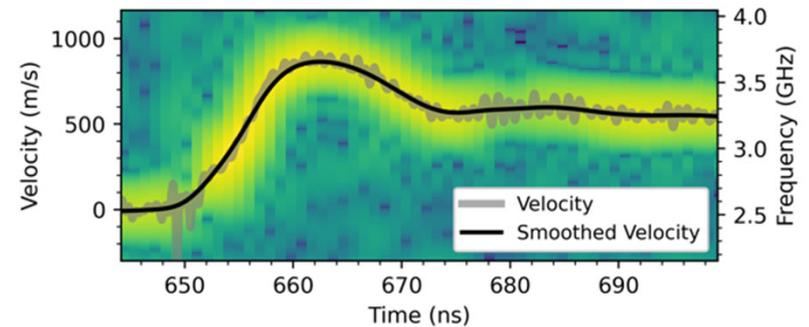
- Apply band-pass filter to isolate frequency range of interest
- Apply Gaussian notch filter to eliminate carrier band
- All filtering is done on the original voltage signal



Program Workflow



- Calculate velocity from the phase angle derivative of the voltage
- Smooth with a Gaussian weighted moving average
- Extract spall points with peak finding

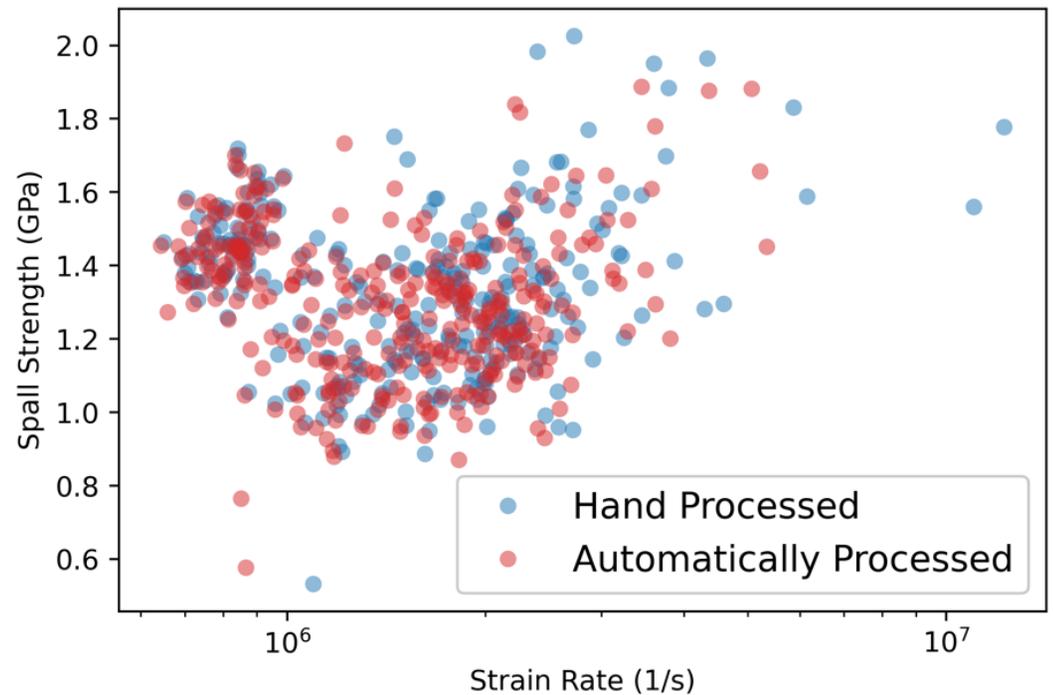




Results

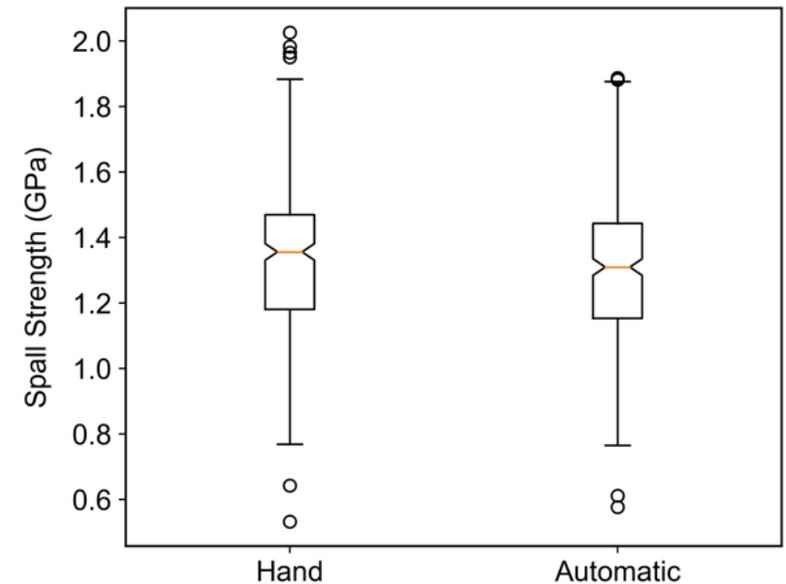
Dr. Chris DiMarco

- Automated results closely match hand processed results
- Slight difference mainly due to different smoothing methods



Results

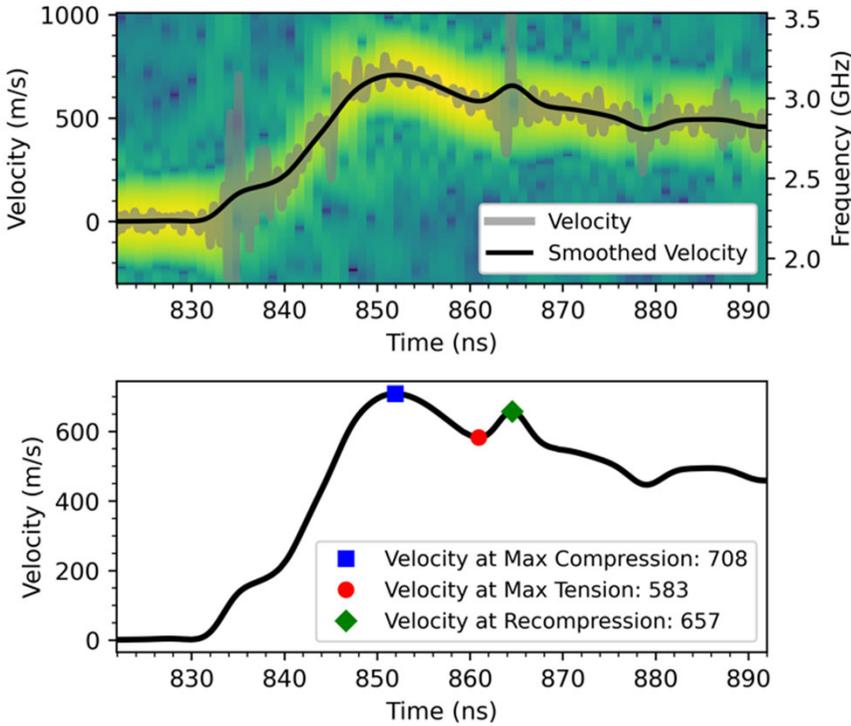
- Final distribution average is 2.5% different from hand analysis
- Each signal takes only 5 – 8s to analyze (Apple M1 Pro, 16GB RAM)



Limitations

ALPSS is unsuccessful in 2 main cases

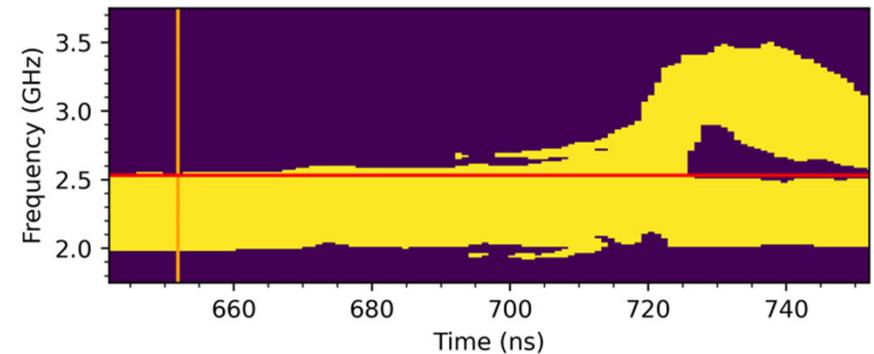
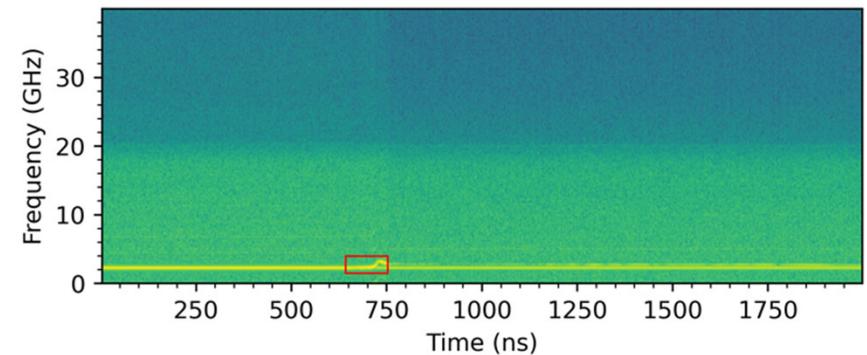
1) Signal drop-out



Limitations

ALPSS is unsuccessful in 2 main cases

- 1) Signal drop-out
- 2) Noise around carrier band



Conclusions and Future Work

- 1) High-throughput spall experiments are outpacing data analysis. ALPSS will allow us to keep up
- 2) ALPSS provides a fast and consistent framework to analyze spall signals
- 3) Automatically analyzed signals closely match hand processed results
- 4) Need improvements on “success” rate and automatic rejection of bad results

Acknowledgements and Code Access



Dr. Chris DiMarco



Dr. Velat Kilic



Dr. Debjoy Mallick



Dr. Maggie Eminizer



Dr. David Elbert



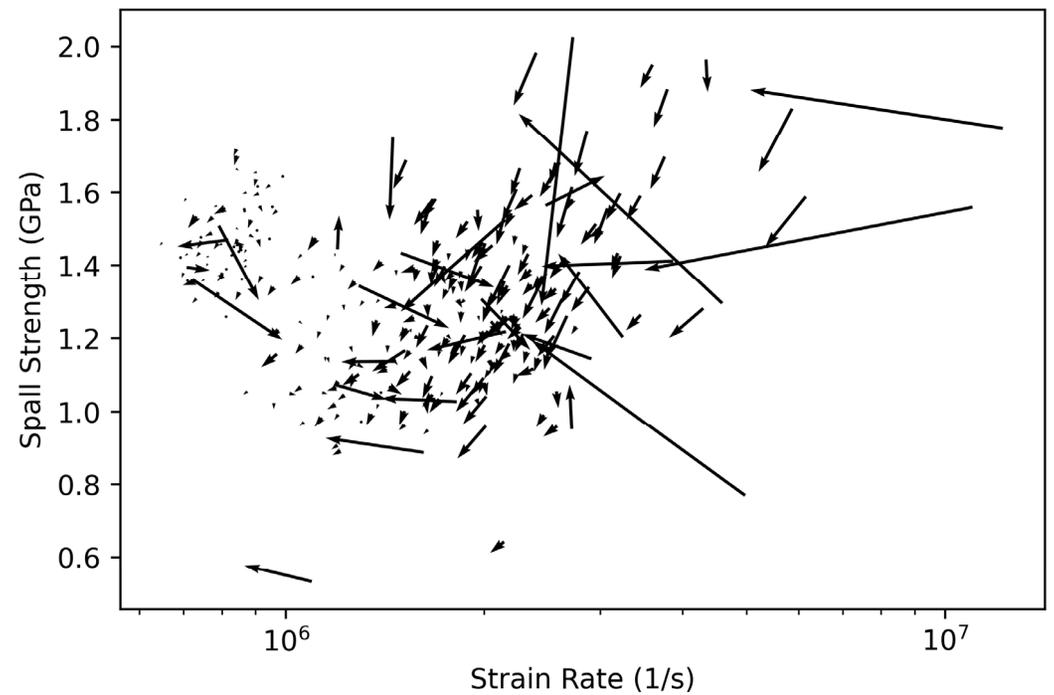
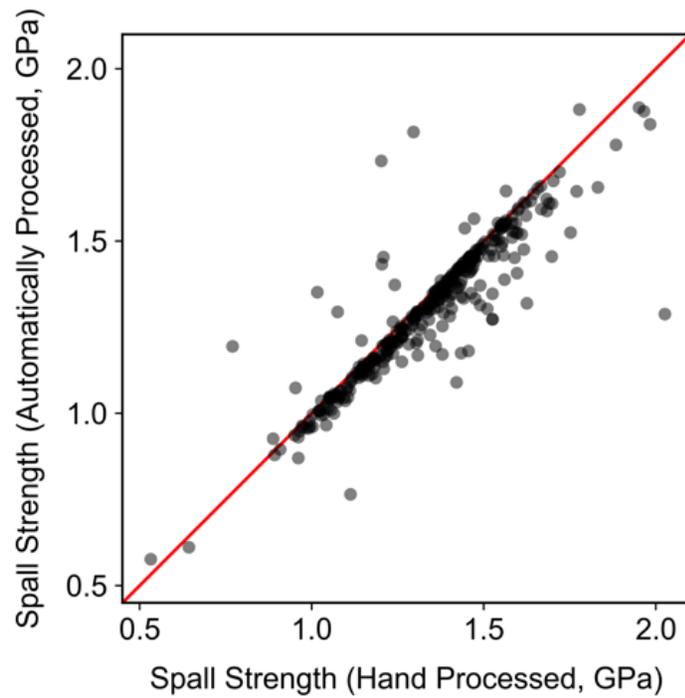
GitHub



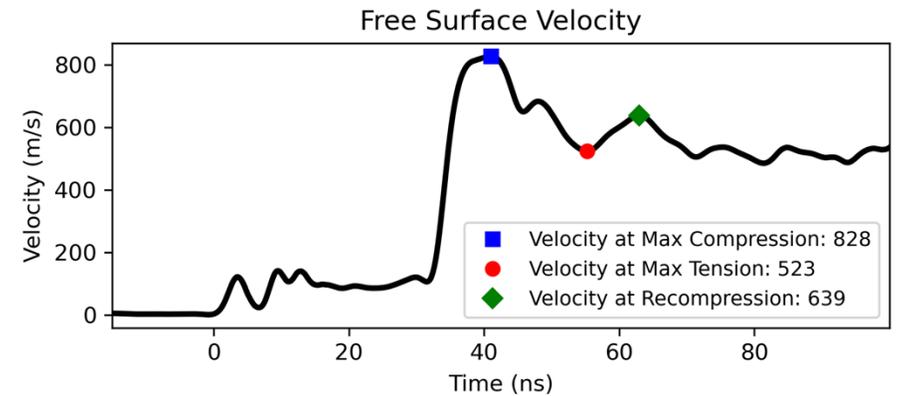
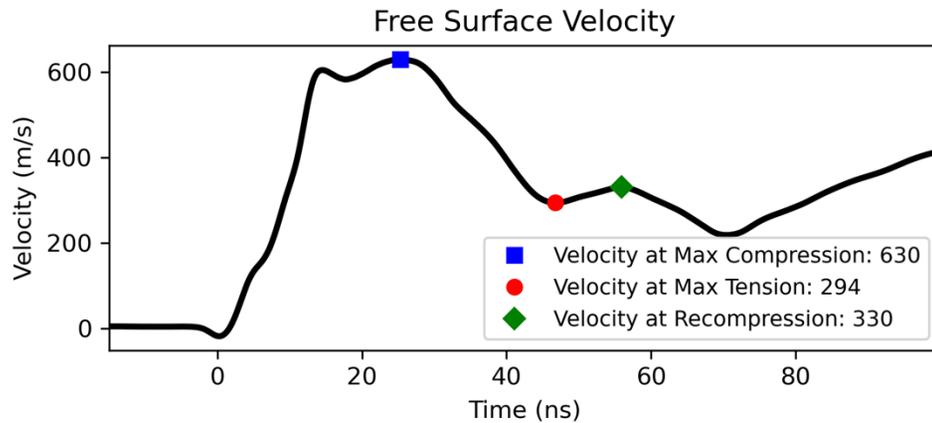
<https://github.com/Jake-Diamond-9/ALPSS>

jdiamo15@jhu.edu

Supplementary Results

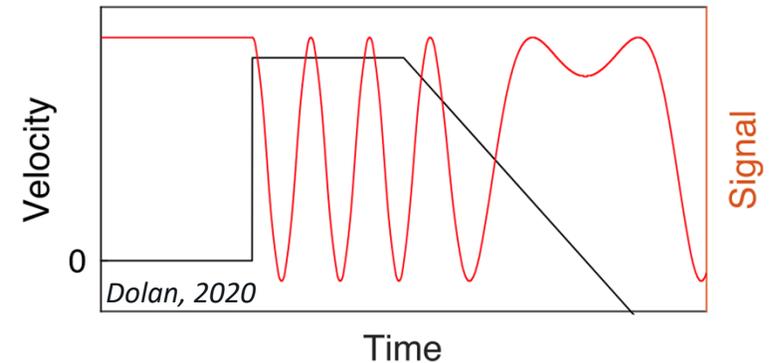


Picking Different Data Points



Phase Angle Derivative

- Finer time resolution is helpful for nanosecond scale experiments
- Have individual control over both the numerical derivative and smoothing procedure

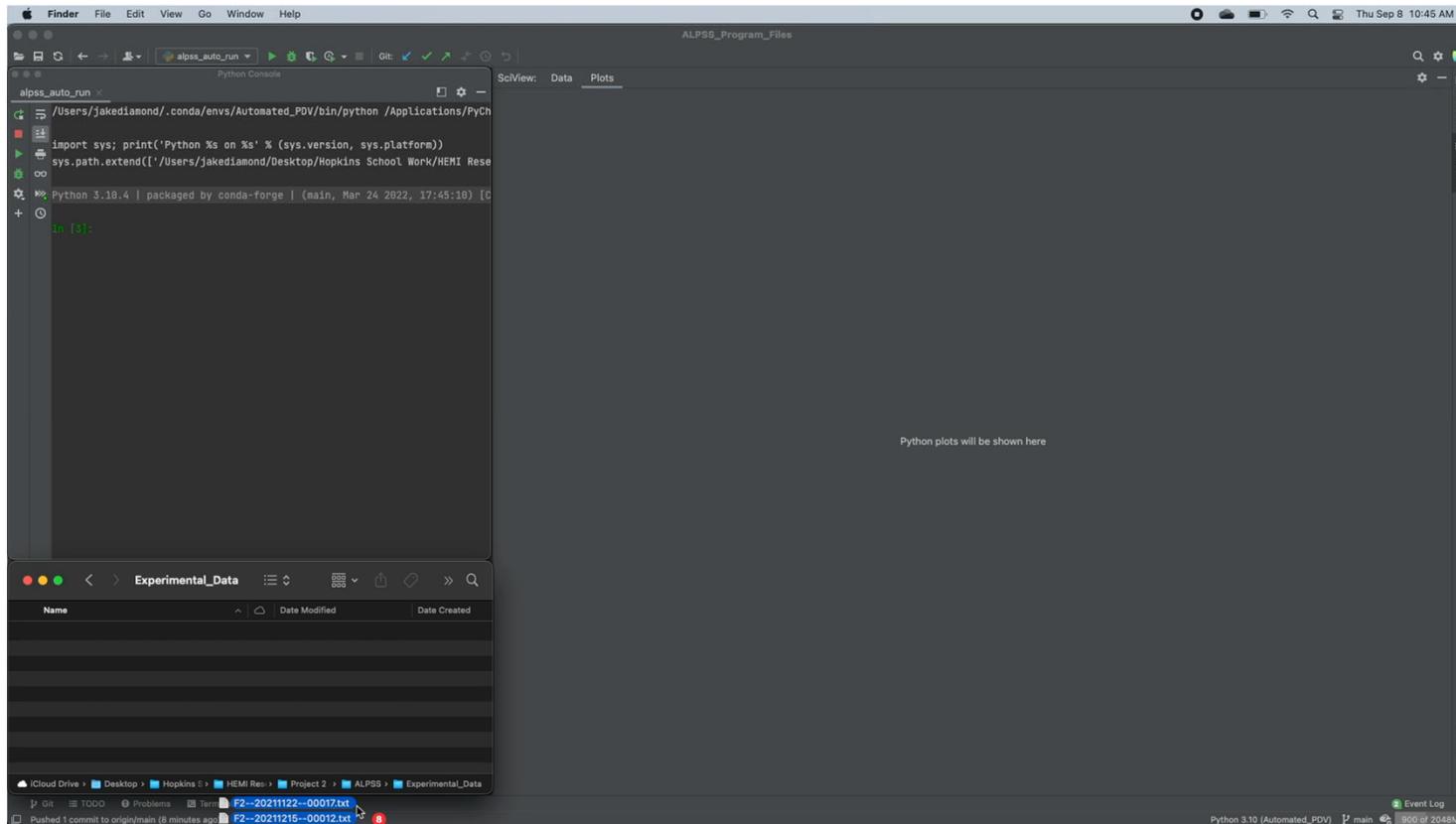


$$V(t) \propto \sin(2\pi\omega(t))$$

$$u(t) = \frac{\lambda_o}{2} \omega(t)$$

$$v(t) = \frac{\lambda_o}{2} \frac{d\omega(t)}{dt}$$

Watchdog Demonstration



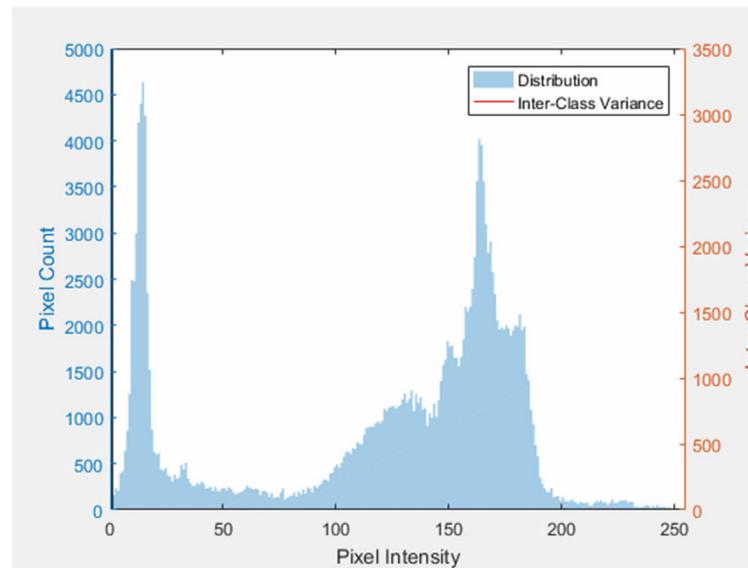
The screenshot shows a JupyterLab environment. The top bar indicates the current directory is `ALPSS_Program_Files`. The left sidebar shows a file browser for the `Experimental_Data` directory, which is currently empty. The main area is a Python console with the following code and output:

```
Python Console
/Users/jakediamond/.conda/envs/Automated_PDV/bin/python /Applications/PyCh...
import sys; print('Python %s on %s' % (sys.version, sys.platform))
sys.path.extend(['/Users/jakediamond/Desktop/Hopkins School Work/HEMI Rese...
Python 3.10.4 | packaged by conda-forge | (main, Mar 24 2022, 17:45:18) [C...
In [1]:
```

The right sidebar shows a plot area with the text "Python plots will be shown here". The bottom status bar shows the current environment is `Python 3.10 (Automated_PDV)` and the user is on the `main` branch of a Git repository.

Otsu's Binarization

- Method for automated thresholding commonly used in image processing
- Works by maximizing inter-class variance (2-point k-means clustering)



https://en.wikipedia.org/wiki/Otsu%27s_method

Uncertainty Analysis

- Calculates the first-order second moment (first standard deviation) for spall strength and strain rate
- Need underlying values of uncertainty for material parameters, PDV laser stability, etc...

$$\delta \dot{\epsilon} = \left\{ \sum_i \left[\frac{\partial \dot{\epsilon}(X_i)}{\partial X_i} \cdot \delta X_i \right]^2 \right\}^{1/2}$$

$$\delta \dot{\epsilon} = \left\{ \left[-\frac{\lambda_D}{4C_0^2 \cos \theta} \frac{f(t_C) - f(t_D)}{|t_C - t_D|} \cdot \delta C_0 \right]^2 + \left[\frac{1}{4C_0 \cos \theta} \frac{f(t_C) - f(t_D)}{|t_C - t_D|} \cdot \delta \lambda_D \right]^2 + \left[\frac{\lambda_D \tan \theta \sec \theta}{4C_0} \frac{f(t_C) - f(t_D)}{|t_C - t_D|} \cdot \delta \theta \right]^2 + \left[\frac{\lambda_D}{4C_0 \cos \theta} \frac{1}{|t_C - t_D|} \cdot \delta f(t_C) \right]^2 + \left[-\frac{\lambda_D}{4C_0 \cos \theta} \frac{1}{|t_C - t_D|} \cdot \delta f(t_D) \right]^2 + \left[-\frac{\lambda_D}{4C_0 \cos \theta} \frac{f(t_C) - f(t_D)}{|t_C - t_D|^2} \cdot \delta t_C \right]^2 + \left[\frac{\lambda_D}{4C_0 \cos \theta} \frac{f(t_C) - f(t_D)}{|t_C - t_D|^2} \cdot \delta t_D \right]^2 \right\}^{1/2}$$

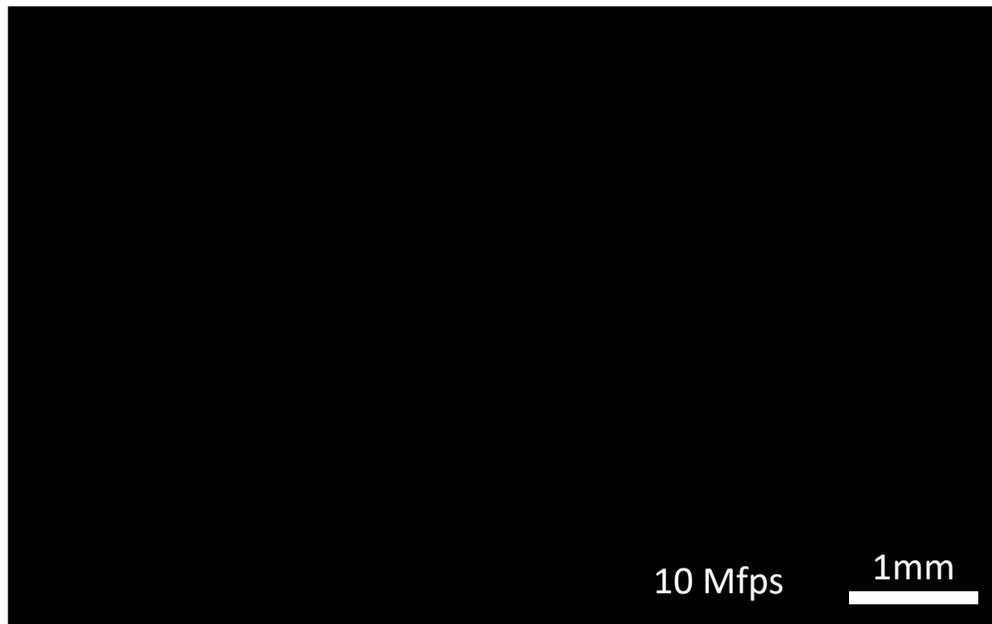
Mallick, D.D., Zhao, M., Parker, J. *et al.* Laser-Driven Flyers and Nanosecond-Resolved Velocimetry for Spall Studies in Thin Metal Foils. *Exp Mech* **59**, 611–628 (2019).
<https://doi.org/10.1007/s11340-019-00519-x>

$$\delta \sigma_F = \left\{ \sum_i \left[\frac{\partial \sigma_F(X_i)}{\partial X_i} \cdot \delta X_i \right]^2 \right\}^{1/2}$$

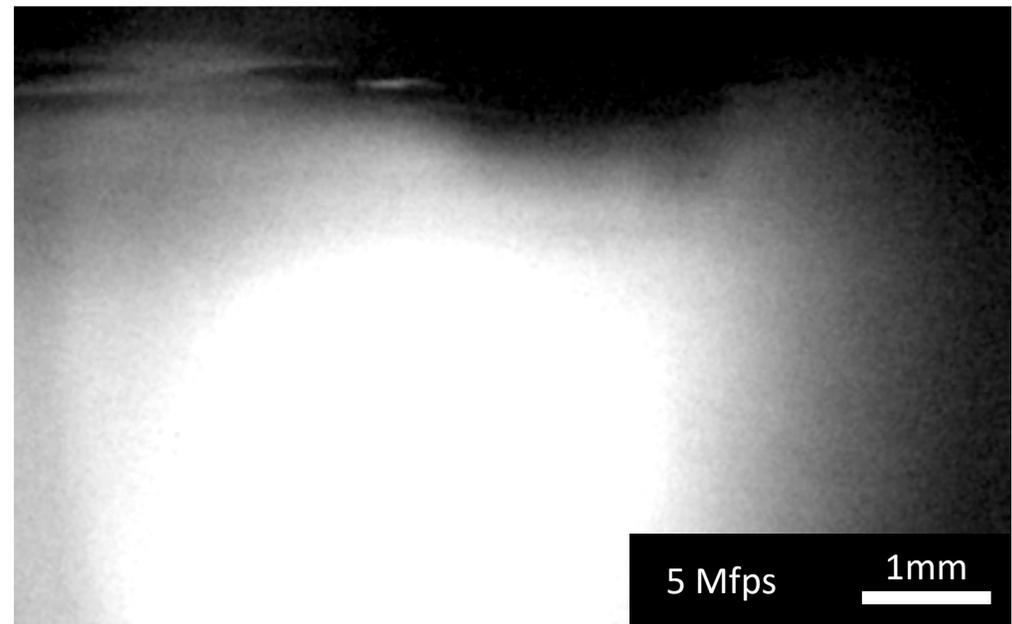
$$\delta \sigma_F = \left\{ \left[-\frac{1}{2} \rho C_0 \frac{\lambda_D}{2} \tan \theta \sec \theta (f(t_B) - f(t_D)) \cdot \delta \theta \right]^2 + \left[\frac{1}{2} \rho C_0 \frac{\lambda_D}{2 \cos \theta} \cdot \delta f(t_B) \right]^2 + \left[-\frac{1}{2} \rho C_0 \frac{\lambda_D}{2 \cos \theta} \cdot \delta f(t_D) \right]^2 + [0 \cdot \delta f_D]^2 + [0 \cdot \delta f_r]^2 + \left[\frac{1}{2} \rho C_0 \frac{1}{2 \cos \theta} (f(t_B) - f(t_D)) \cdot \delta \lambda_D \right]^2 + \left[\frac{1}{2} \rho \frac{\lambda_D}{2 \cos \theta} (f(t_B) - f(t_D)) \cdot \delta C_0 \right]^2 + \left[\frac{1}{2} C_0 \frac{\lambda_D}{2 \cos \theta} (f(t_B) - f(t_D)) \cdot \delta \rho \right]^2 \right\}^{1/2}, \quad (9)$$

Flyer Launch and Impact

Aluminum flyer in free flight



Impact on polycarbonate at ~ 700 m/s



AIMD

