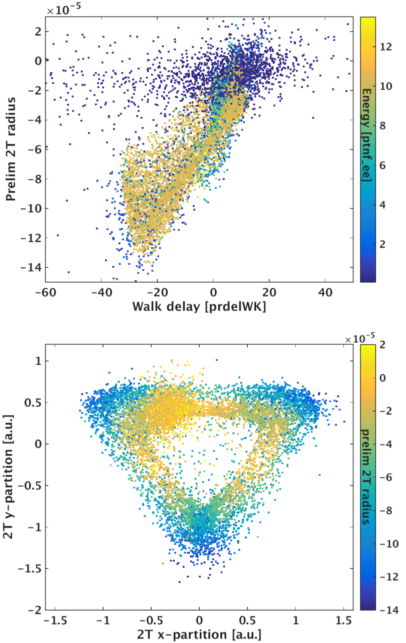
# Multi-template time-shifting filter

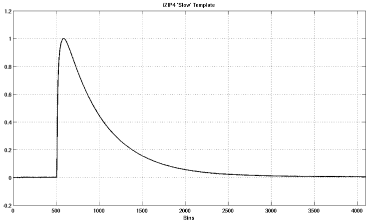
J.S. Wilson

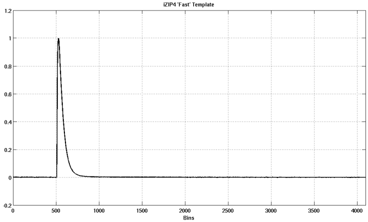
Texas A&M

LT meeting
January 19, 2016

- Two template OF works well
- Especially effective in radial cut for CDMSlite Run 2
- Fast template allows better fit to peaky pulses
- Peakiness carries position information
- So, if 2 templates is better than 1, is 3 better than 2? 4 better than 3?
- How many templates are needed?
- How can we construct the templates?

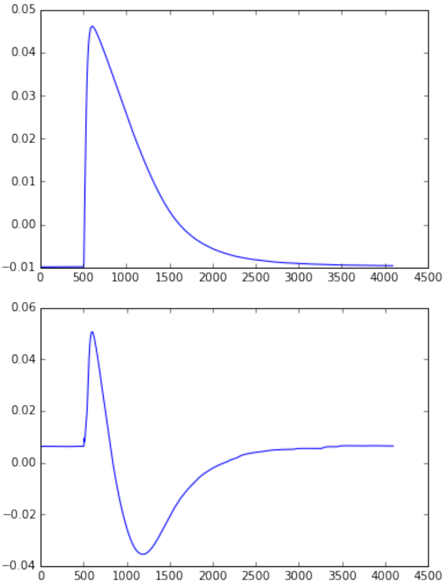


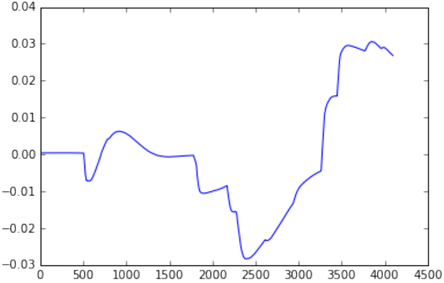



- To build templates, take clean, high energy pulses
- Average them to get first template
- Take the residuals, and sign flip them as needed
- Average the flipped residuals to get second template
- This procedure is reminiscent of Principal Components Analysis (PCA)

- Try PCA on Cf series from R135 (not blinded!)
- Load raw traces
- Minimal cleanup – only remove digitizer overflows
- First template looks like a pulse, good!
- Second template looks similar to the fast template
- Not identical, but this is probably good enough to be getting on with
- Let's look at the third template






- What the heck is this!?
- Doesn't look like any reasonable component of a pulse
- Out-of-time pulses are messing us up
- What can we do about this?

- Time shift is non-linear in both time and frequency domain
- In frequency domain, time shift is multiplication by phase factor
- Phase proportional to frequency (N.B. $\omega_k = k\omega$):

$$r_k \exp[i\phi_k] \rightarrow r_k \exp[i\phi_k] \exp[itk\omega] = r_k \exp[i(\phi_k + t\omega)]$$

- Transform frequency-space amplitudes by replacing phase with phase *difference*:

$$r_k \exp[i\phi_k] \rightarrow r_k \exp[i(\phi_k - \phi_{k+1})]$$

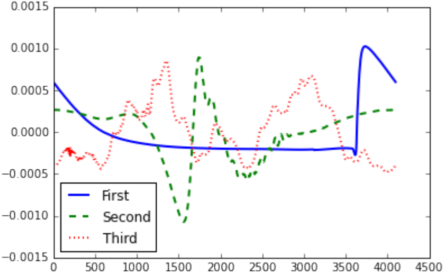
- Then, time shift is linearized:

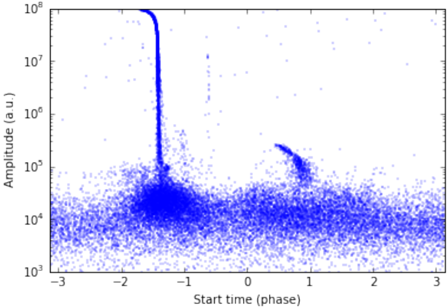
$$\begin{aligned} r_k \exp[i(\phi_k - \phi_{k+1})] &\rightarrow r_k \exp[i(\phi_k + tk\omega - \phi_{k+1} - t(k+1)\omega)] \\ &= r_k \exp[i(\phi_k - \phi_{k+1})] \exp[it\omega] \end{aligned}$$

- Time shift now just multiplication by complex number, independent of $k$ – linearity!

- This transformation is reversible, just take cumulative sum of phase differences to get back original phase
- Boundary effects: We have to lose either the DC component or the highest-frequency component. I chose to lose the DC component.
- Cannot have negative amplitude pulses: negating the amplitude is just a phase shift of $\pi$
- Time shift is circular
- Both the transformation and its inverse are not linear – some things don't behave like you expect them to
- PCA has overall phase ambiguity, so individual templates have a random "start time".

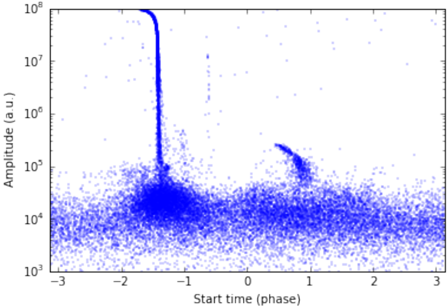
- Apply PCA to pulses in transformed frequency domain
- Get templates also in transformed frequency domain
- Reverse transform to see time-domain templates
- Templates all look plausible
- Nothing has obviously gone wrong
- Not immediately obvious that these templates correspond to fast/slow phonons
- Keep non-linearity in mind





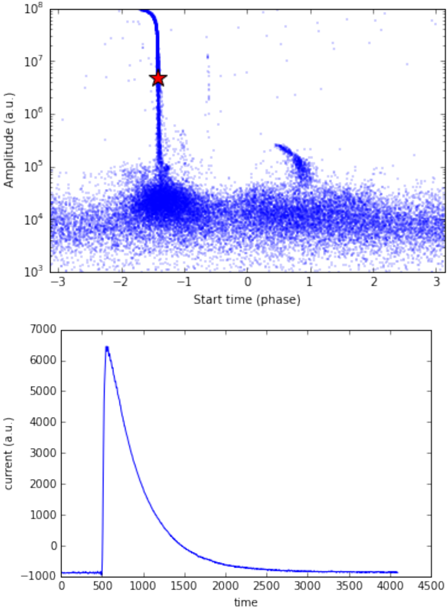

First component, phase vs magnitude

- Since PCA templates are orthogonal, just have to multiply by traces and sum (in transformed frequency space)
- Result is complex scalar for each component
- Magnitude is amplitude of that component
- Phase is time shift of that component
- Several distinct populations of events
- We'll examine each in turn

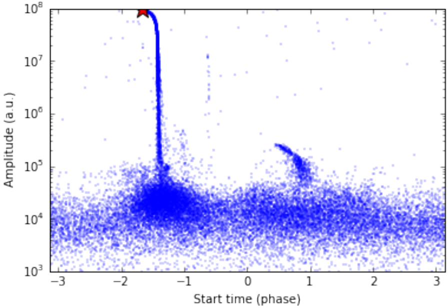


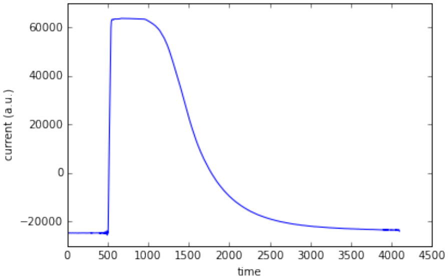

First component, phase vs magnitude

- Band across lower part of plot is obviously noise events – no pulse
- Narrow vertical line is good, in-time pulses
- Good pulses line bends left at the top, due to saturation events
- Sharp feature on right is upside-down, squareish pulses
- Sparse narrow vertical line to right of good pulses is early pulses, before readout window opens
- Small spot (hard to see) at base of good pulses is glitches
- Scattered high-amplitude points are out-of-time pulses

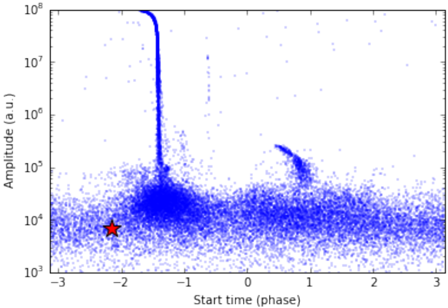


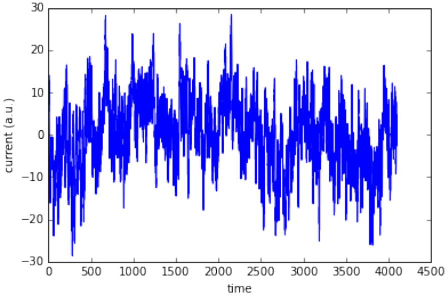

Here is a good pulse. Location of this event marked with red star in scatter plot.



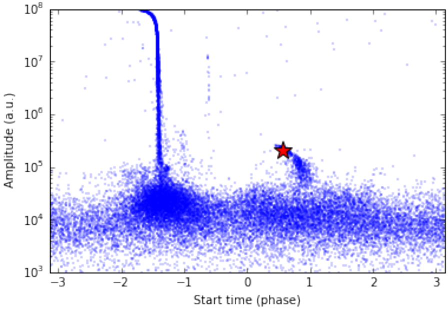



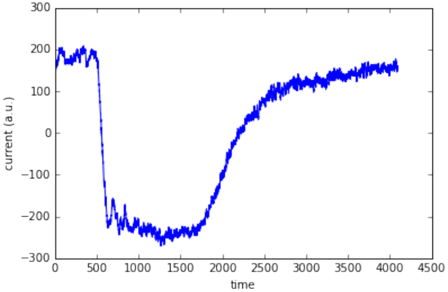

Here is a good pulse, but with enough energy to saturate the TES



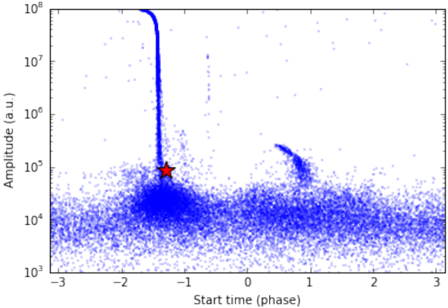



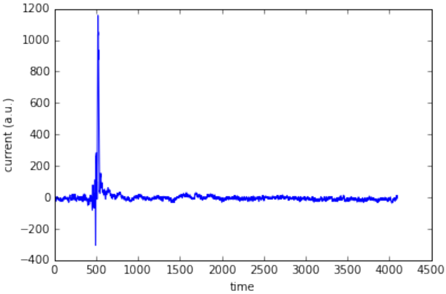

Here is a trace with no pulse

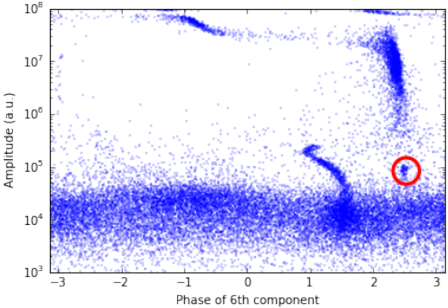


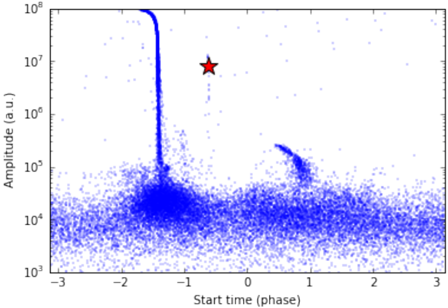

These events are upside down and squared off. What are these? There are a lot of them.

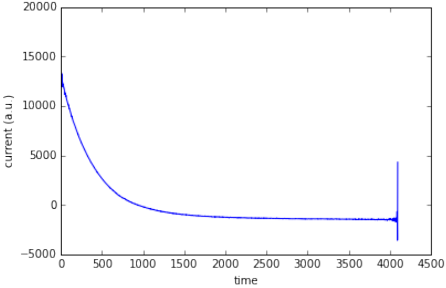


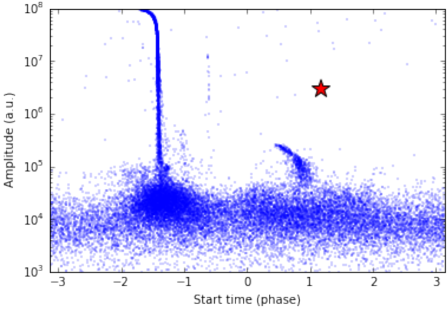



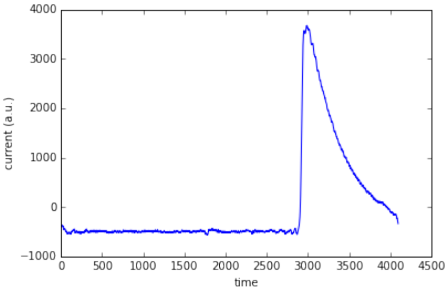

Here is a glitch. The lower plot shows the magnitude of the first template against the phase of the 6th template. In this plot, glitches much better separated from good pulses (around 2.5, 1e5), but I didn't mark this event with a star because it covers up all the glitches.



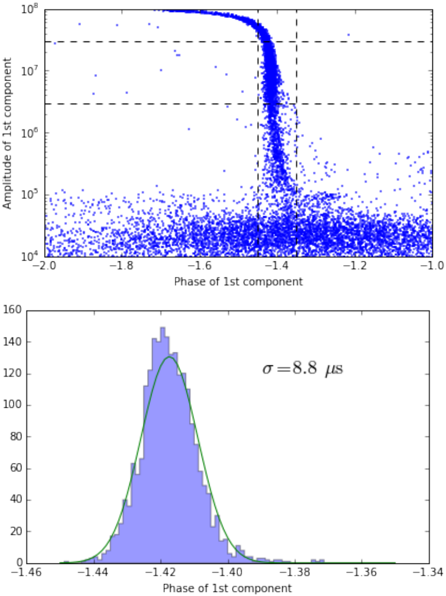

This event contains a pulse that started before the readout window, so we only see the falling part. The "ringing" at the tail of the pulse is an artifact of reconstructing the pulse from the PCA templates.

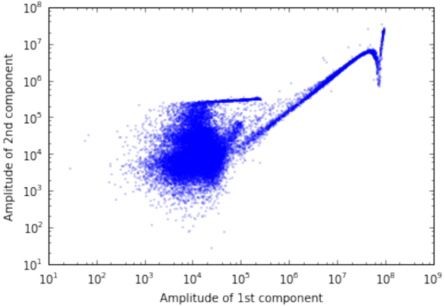


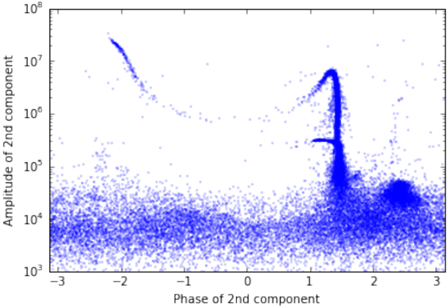



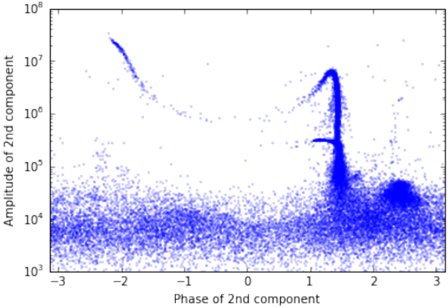

A good pulse, just a bit late

- For high amplitude pulses, timing resolution is pretty decent
- About 8.8 μs
- Gets worse, of course, as amplitude decreases






- Many other components to exploit
- Second component shown here
- Structures continue to be apparent
- Lots of shape information – but how to use it?

- Now we have a way to construct as many templates as we like from minimally-cleaned data
- Filter finds both amplitude and start time
- Exposes interesting subpopulations of events easily
- Timing resolution is not bad
- Lots more information to exploit
- What I haven't shown yet:
  - Most components – just too many plots, but plenty of interesting information
  - Results when running on individual channels separately or all channels together
  - Use of individual channel results to extract position information – looks very promising, but still too preliminary