Tune Computation via Model Fitting to Swept Machine Response Measurement

Michael Abbott

Diamond Light Source

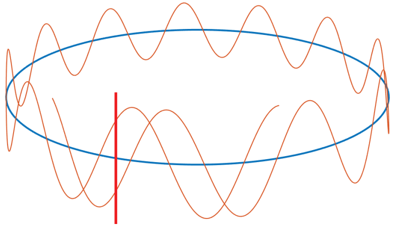
IBIC 2019

Overview of Talk

- Response measurement (*Beam Transfer Function*):
  Using multi-bunch feedback system as vector network analyser.
- Multi-pole resonator model:
  Note that this is *not* a machine physics model.
- Fitting the model, finding the tune:
  Python code for fitter available from author.
- Complications!
  Fitting doesn't always work, some sweeps are hard to interpret.

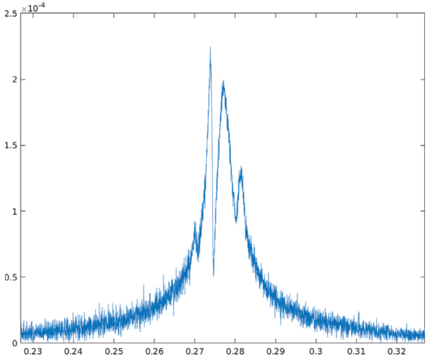
Overview of Tune Measurement Process

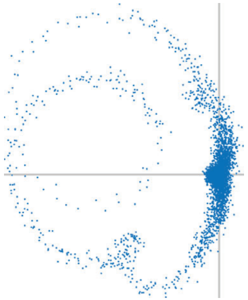
Machine Betatron and Synchrotron Tunes



Tune excitation and measurement at a single location.

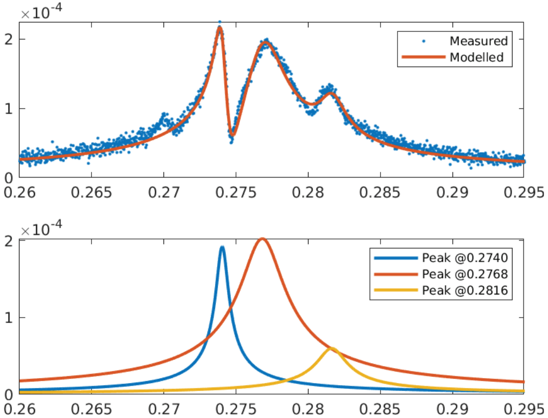
Measured Magnitude





Measured IQ

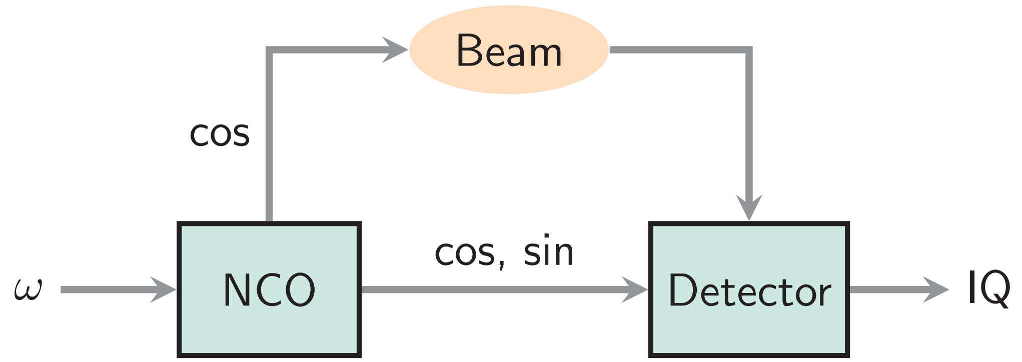
Model fitted to tune sweep



$$\boxed{\begin{array}{c}\text{Tune} = 0.2768 \\ \text{Phase} = 159^\circ\end{array}}$$

Tune Measurement

Measuring Beam Frequency Response

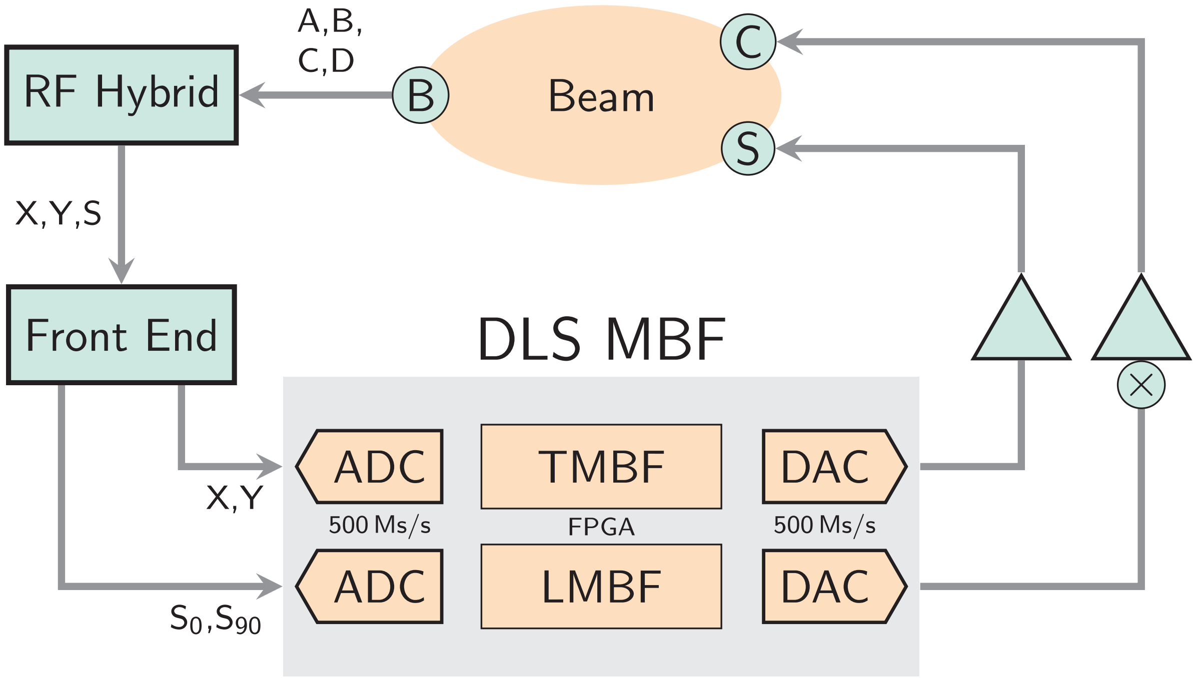


By exciting the beam at a selected frequency ω and measuring the response of the beam at that frequency, we compute the *transfer function* of the machine at the selected frequency:

$$R(\omega) = \sum_{t \in \mathrm{dwell}(\omega)} e^{-i\omega t} x_t$$

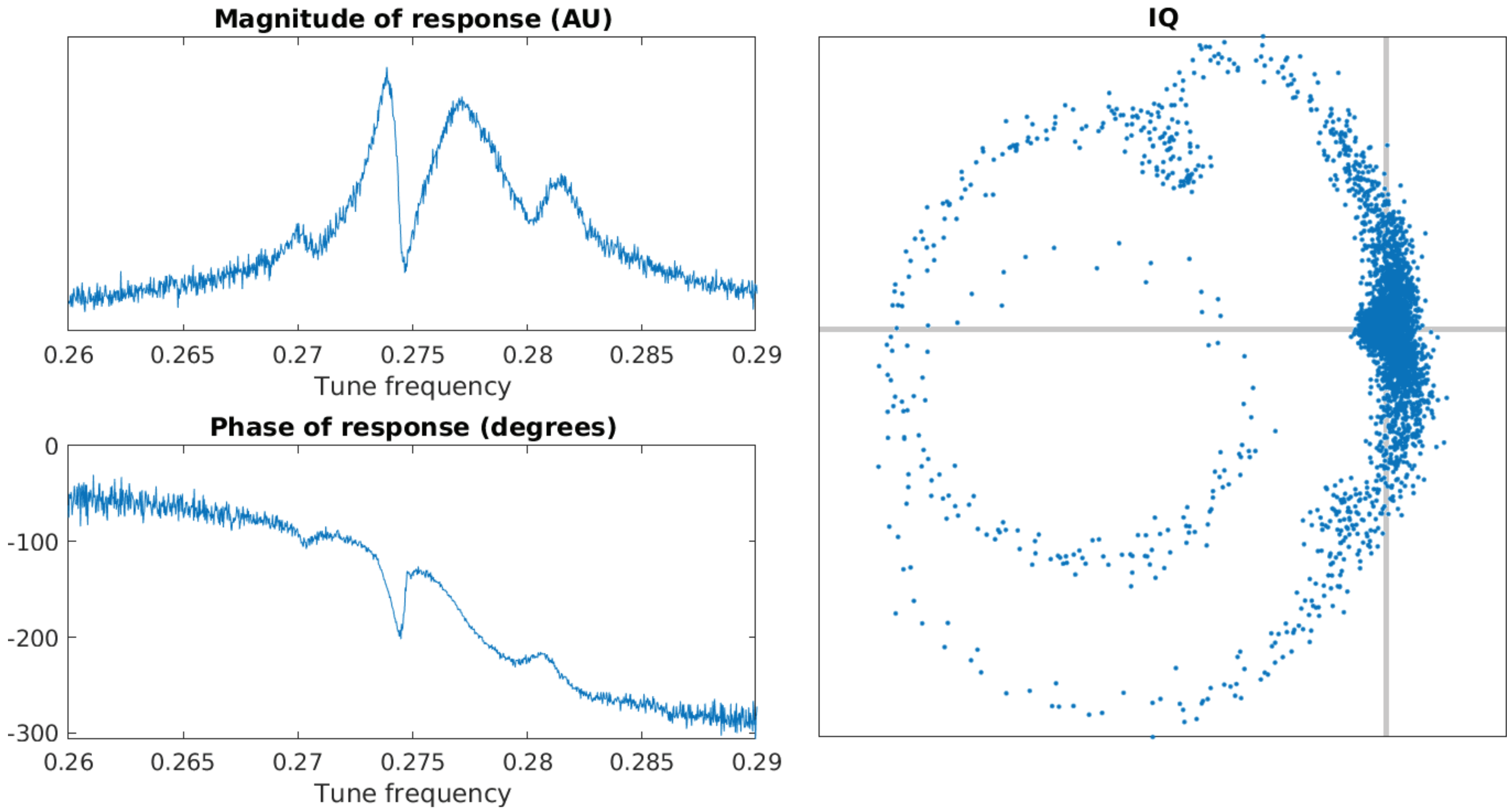
This can be expressed as phase and magnitude, or equivalently as a complex number, or in digital processing terms as a pair (I,Q).

System Setup for Bunch-by-Bunch Control

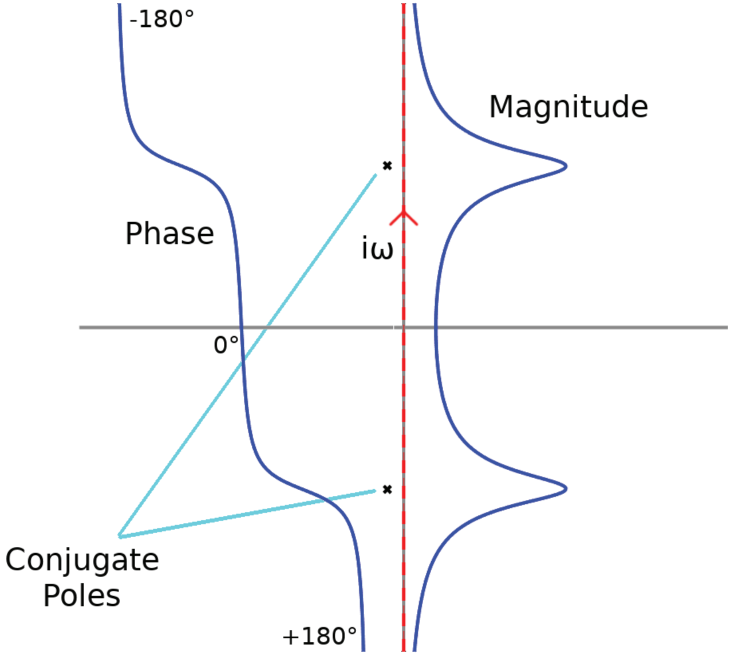


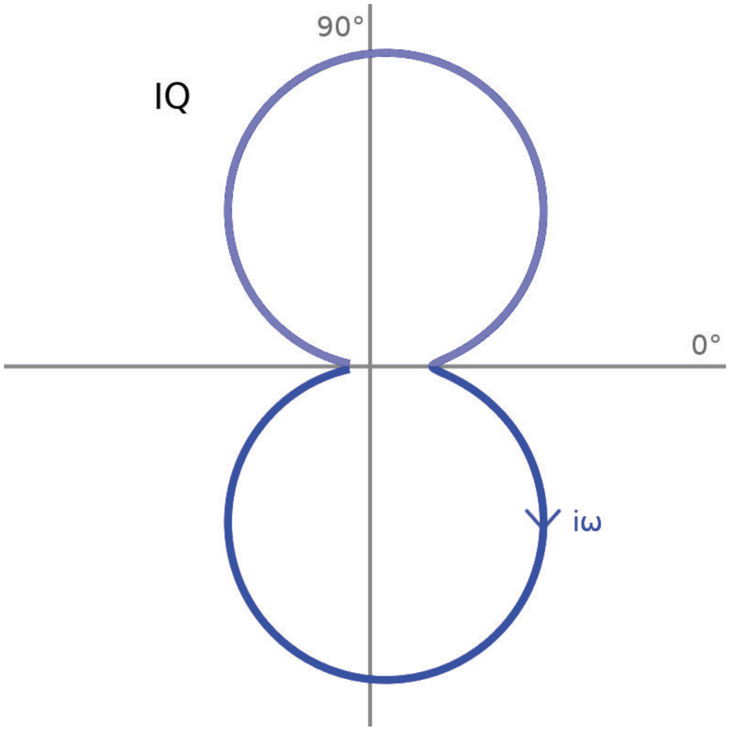
(B) EBPM pickup; (C) Longitudinal cavity; (S) Transverse striplines.

Typical Machine Response Measurement

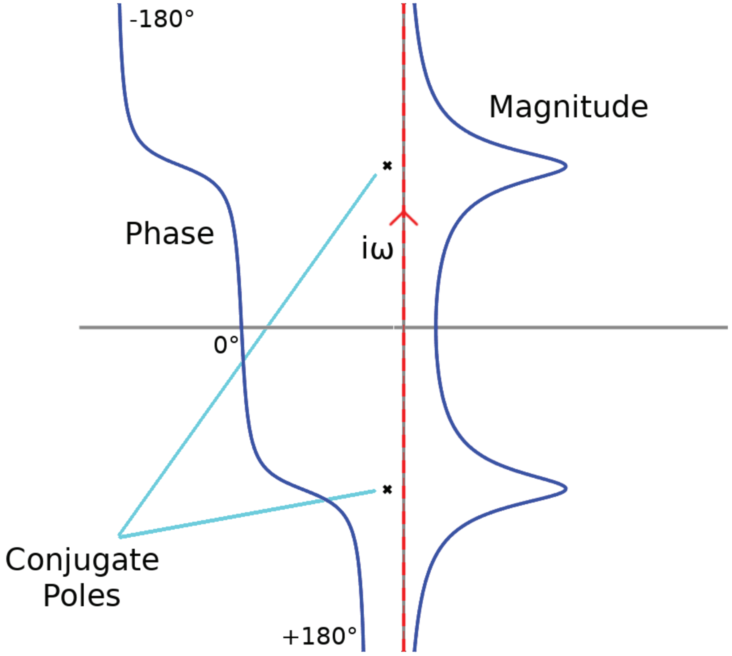


Damped Harmonic Oscillator





Damped Harmonic Oscillator



Damped harmonic oscillator:

$$\ddot{x} + 2\nu\dot{x} + \omega_0^2 x = y$$

Laplace transform:

$$s^2 X + 2\nu s X + \omega_0^2 X = Y$$

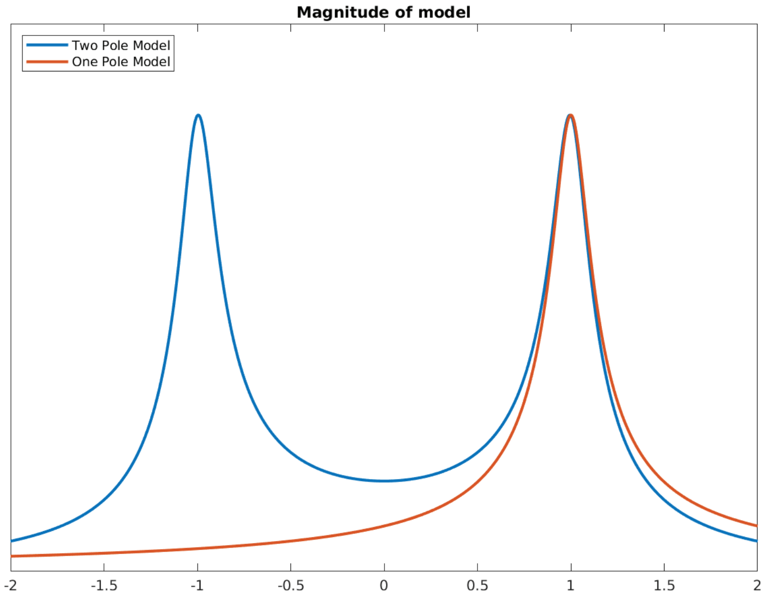
Response:

$$\frac{X}{Y} = \frac{1}{(s-b)(s-b^*)} = \frac{a}{s-b} - \frac{a}{s-b^*}$$

where

$$a = \frac{1}{2i\omega_c}\ , \quad b = \nu + i\omega_c\ , \quad \omega_c^2 = \omega_0^2 - \nu^2\ .$$

Single Pole Resonator Model



Our measurements are very narrow band, and we only sweep a narrow range, so we can ignore one pole:

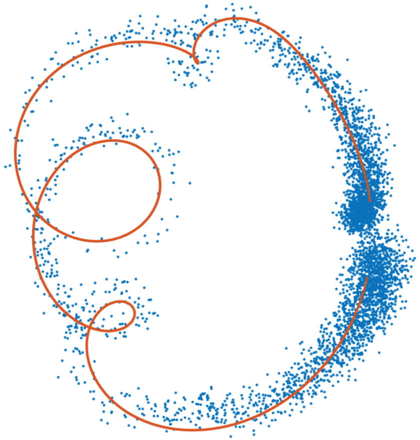
$$M(\omega) = \frac{X}{Y}(i\omega) \approx \frac{a}{i\omega - b} = \frac{a'}{\omega - b'}$$

Result is a “Single Pole Resonator” model.

Example Q is 10, typical tune Q is 100s to 1000s.

Multi-pole Resonator Model

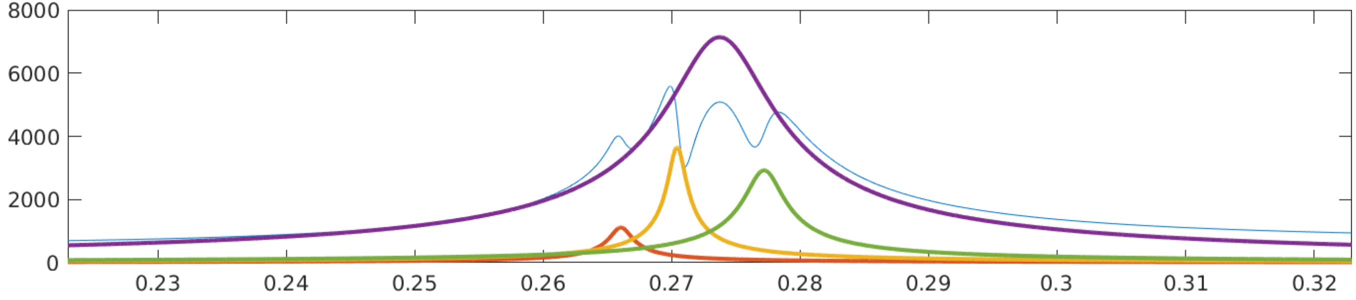
Modelling the raw data as a sum of one pole resonators produces a good fit to experimental data:



$$M(\omega) = \sum_{n=1}^{N} \frac{a_n}{\omega - b_n} + c = \frac{P(\omega)}{Q(\omega)}$$

This is mathematically sound, produces a convincing fit when successful, but is surprisingly tricky to fit numerically.

Extracting Data From Model



	Equation	Peak -2	Peak -1	Tune	Peak +1	
Tune	$\mathrm{re}(b)$	0.2661	0.2704	0.2737	0.2772	
Width	$\mathrm{im}(b)$	0.9	0.7	3.9	1.5	$(\times 10^{-3})$
"Power"	$\int_{\mathbb{R}} \lvert M(\omega)\rvert^2 \, d\omega = \frac{\lvert a\rvert^2}{\mathrm{im}(b)}$	0.04	0.25	1	0.5	[1]
Phase	$\angle(i \cdot a)$	170°	-110°	180°	110°	

[1] Relative to tune power

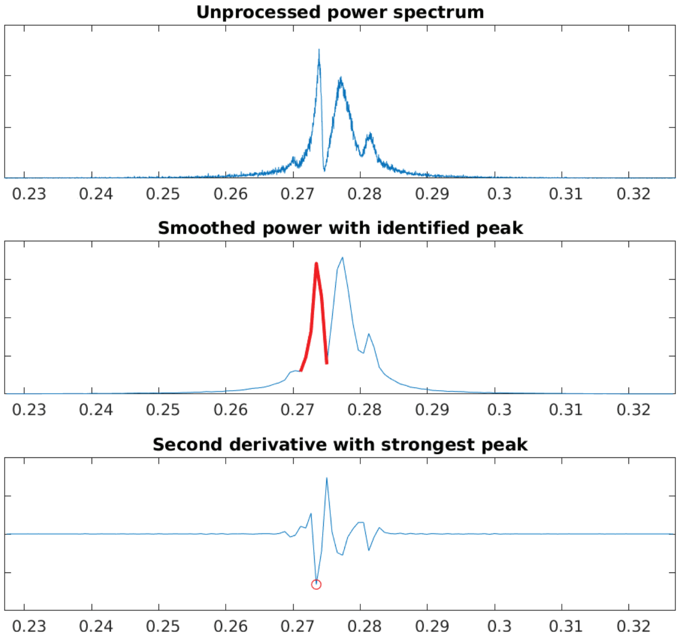
Algorithm

We fit as many peaks as we can up to a configured number of peaks.

- Fit one peak at a time until done
- Find largest peak in residual response power
- Simple linear fit to discovered peak
- Refine fit with non-linear optimisation
- Assess quality of resulting model, discard and stop if poor

When fitting is done take peak with largest “power” as the tune.

Peak Discovery



- Compute residue by subtracting model so far from data to fit:

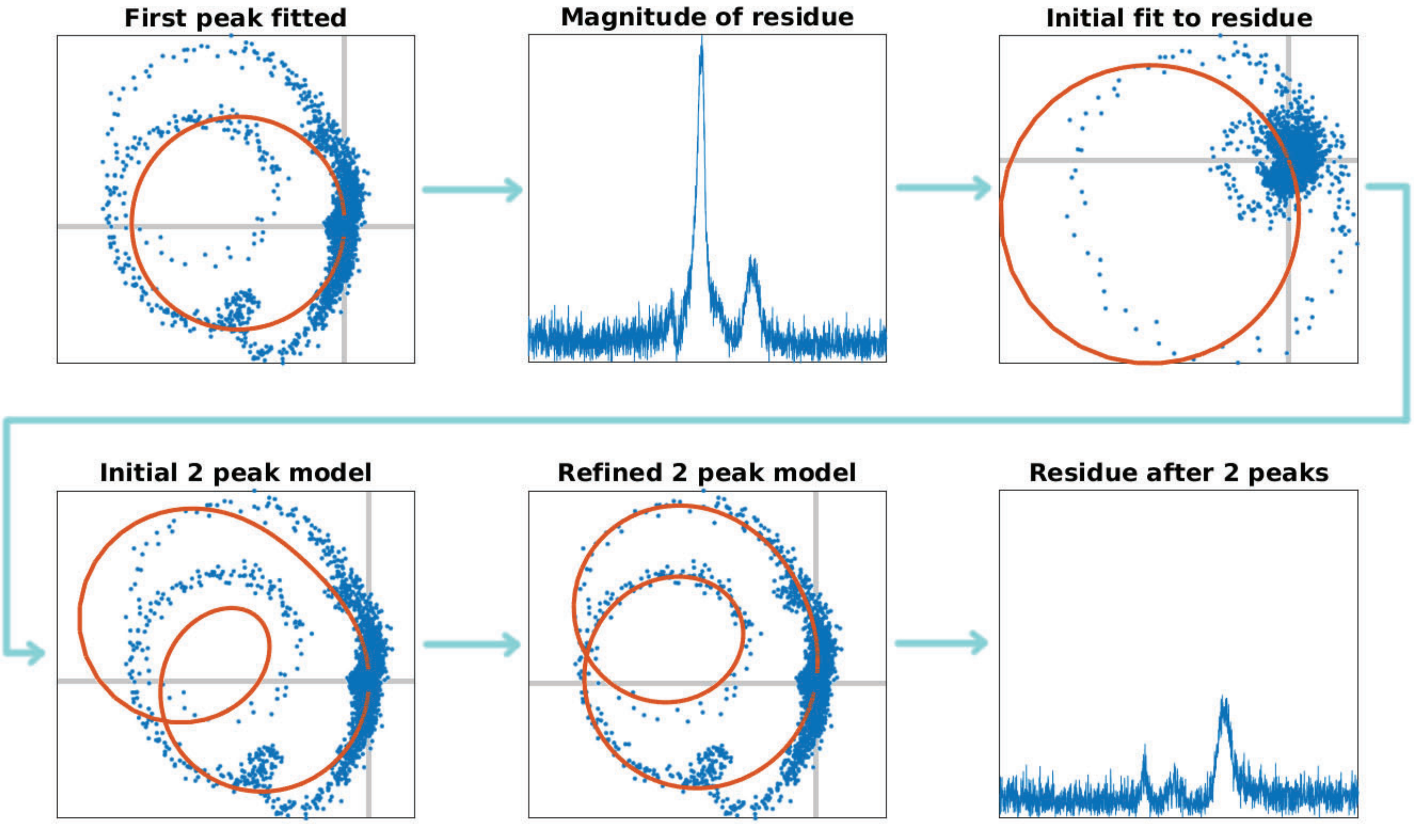
$$r(\omega) = R(\omega) - M(\omega)$$

- Smooth and decimate power $|r|^2$, take second derivative $d^2\mathcal{S}(|r|^2)/d\omega^2$.

  Select point with largest curvature as peak: inspired by early computer vision methods!
- Follow curve to define interval for initial fit.

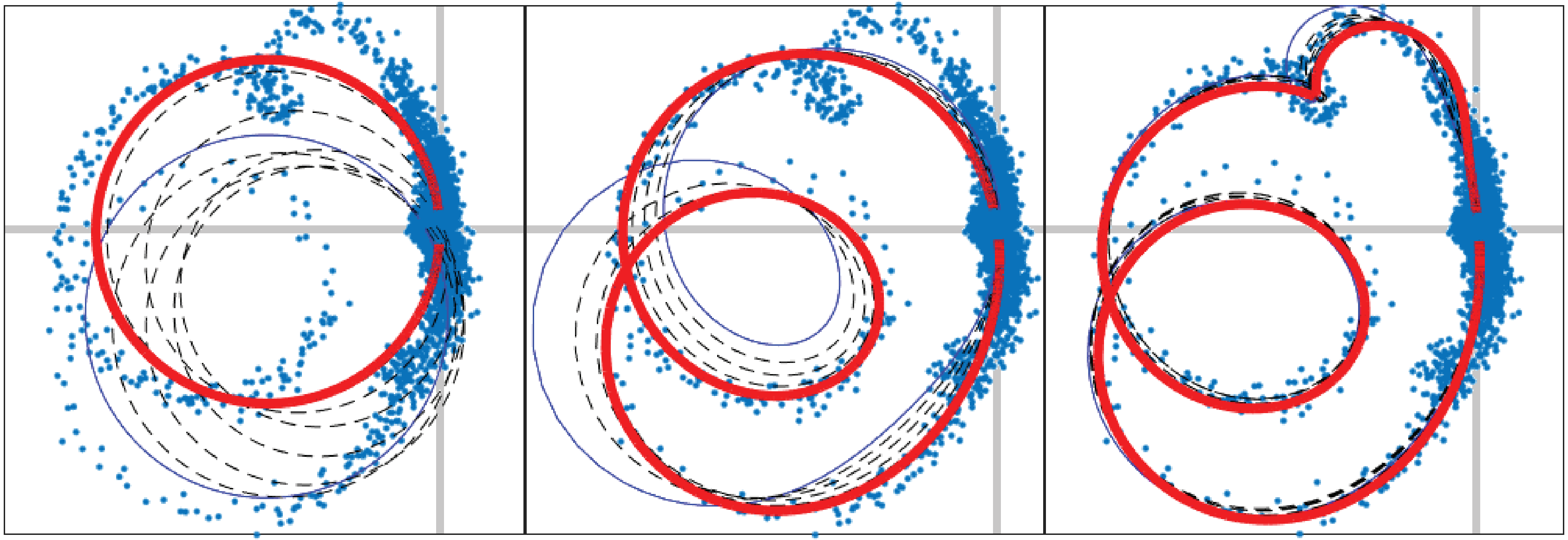
This method tends to become brittle as more peaks are fitted.

Overview of One Round of Fitting



Fit Refinement

Fit refinement is done using the Levenberg-Marquardt algorithm to optimise the fit. The figure below shows the impact of this step.

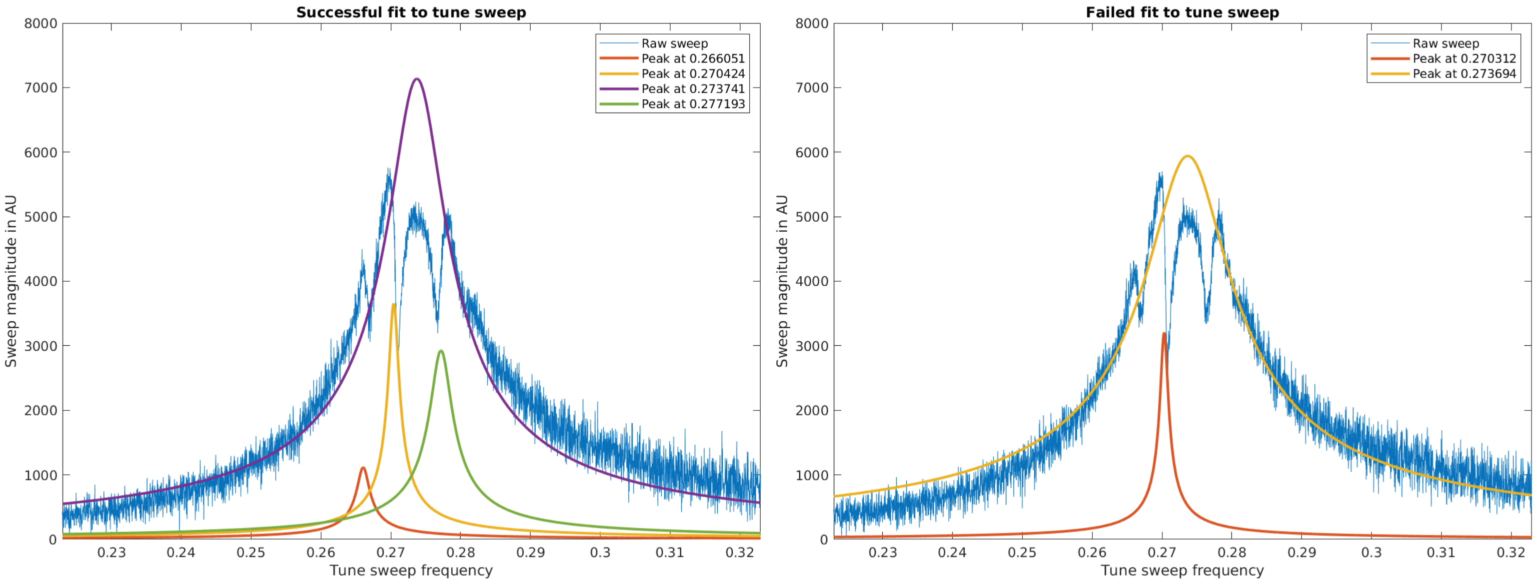


Dots: data to fit; thin blue line: initial model; thick red line: refined model.

Challenges

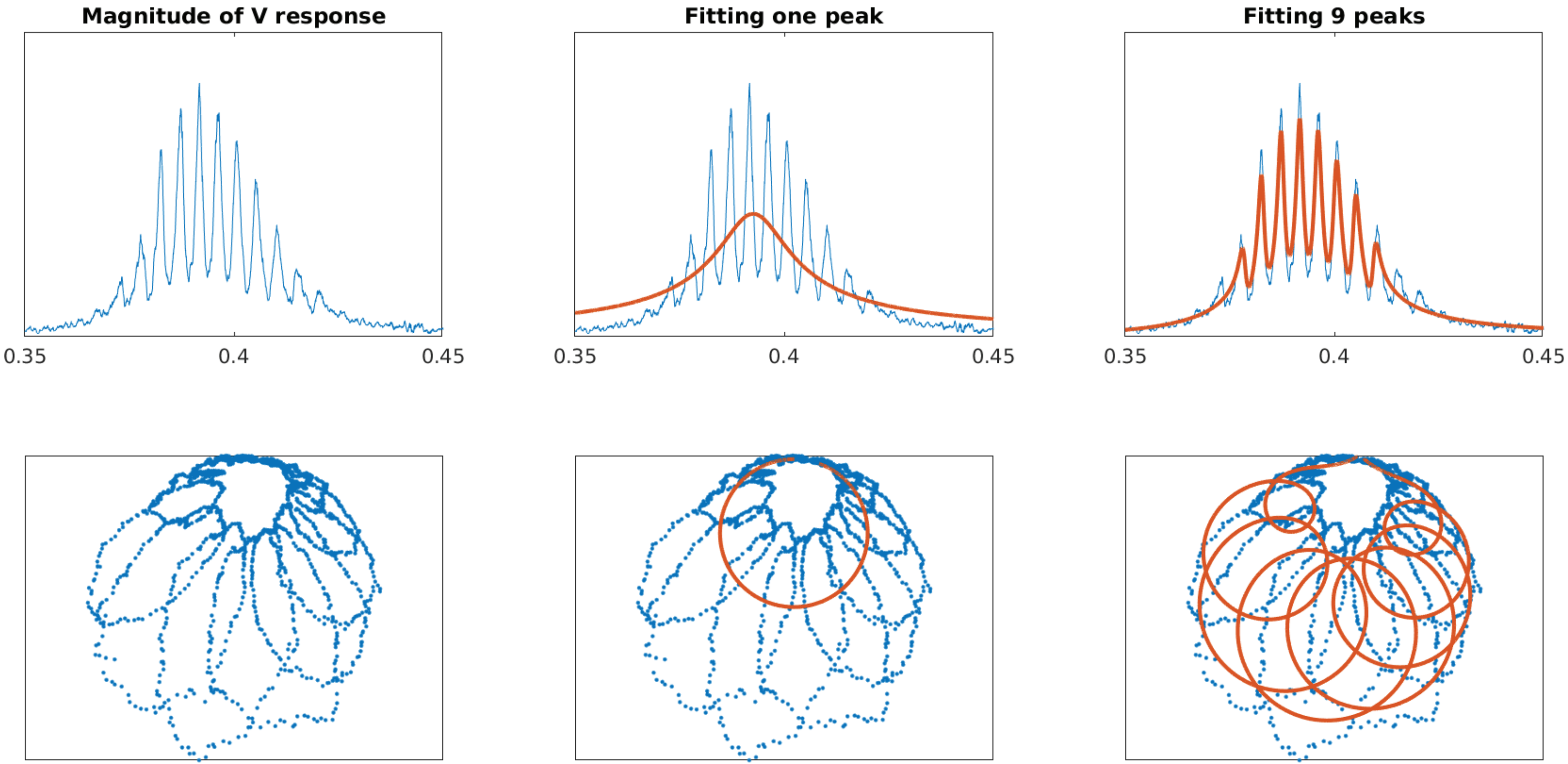
I will end with a few challenging fits.

Sometimes it fits

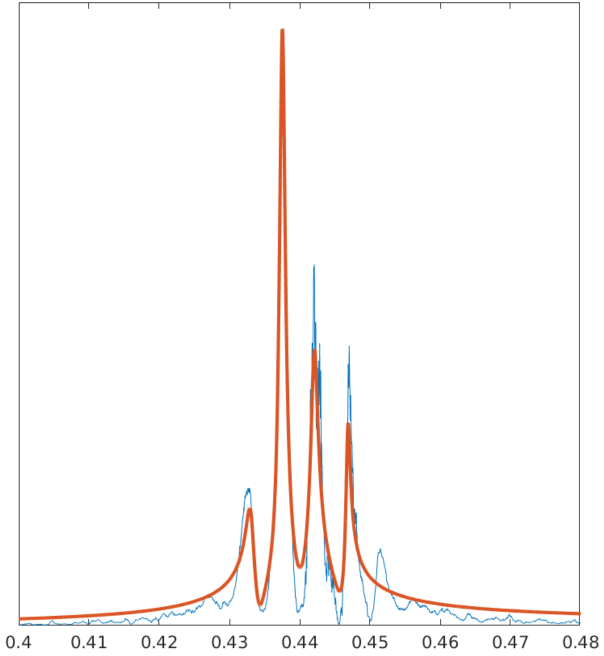


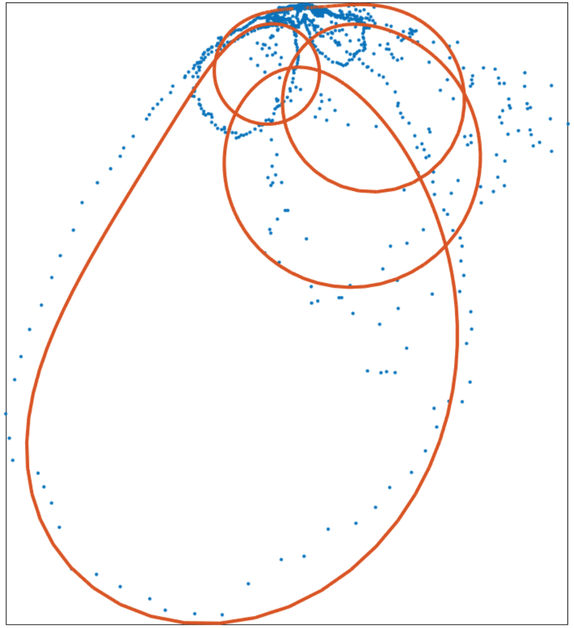
Here we see two almost identical sweeps. Perfect fit on the left, only two peaks fitted on the right. In this case we tried and failed to fit the smallest peak next!

ESRF: Vertical Tune



ESRF: Horizontal Tune





END