Step 1 - Make a dynamic spectrum to detect the major tone - Carrier line, and compare result with theoretically predicted

Input data file name DataFileName := "Data3.NoSpin.NoPhNoise.byte"

A priori Doppler Polynomial model file name in this case it is 0 DpcFileName0 := "Data3.Det.Pass0.tone0.fitcoeffs.txt"

Pass 1 frequency detection DpcFileName1 := "Data3.Det.Pass1.tone0.fitcoeffs.txt"

Phase model coefficients used to generate the input signal PhModFileName := "Data3.Model.PhaseCoeffs.tone0.txt"

Data set and processing essential parameters

Band Width $BW := 250 \cdot kHz$

Scan length (in samples) $Nt := 8 \cdot 1024 \cdot 1024$

N of points to FFT $Nps := 16 \cdot 1024$ For a second iteration, the FFT can be longer

Overlapping of FFT segments $Ovlp := 2$

Number of spectra to average (not accounting for overlap) $Nav := 8$

Padding coeff and array .Padding 2 means that input array will be padded with the same length of zeroes, so making it two times longer. padding is helpful, although cpu time consuming. Padding 1 means no padding. $Padd := 2$

Derived parameters (some of then are just for illustration)

Sampling rate $Sr := 2 \cdot BW$

And sampling interval $dt := Sr^{-1}$ Dimensionless $dtn := dt \cdot s^{-1}$

Total time span $Tspan := Nt \cdot dt$ $Tspan = 16.777216\ s$ Dimensionless $Tspann := Tspan \cdot s^{-1}$

Input Time grid $jt := 0 .. Nt - 1$ $tt_{jt} := jt \cdot dt$

Binning within FFT segment $jps := 0 .. Nps - 1$

Window function in time domain $Win_{jps} := \cos\left[\frac{\pi}{Nps} \cdot (jps - 0.5 \cdot Nps + 0.5)\right]^2$

Time span of the FFT $tw := dt \cdot Nps$ $tw = 0.032768\ s$

Native frequency resolution $df := tw^{-1}$ $df = 30.517578125\ s^{-1}$

Derived parameters , Continue

Number of spectra to average (accounting for overlap) $Nspav := Nav \cdot Ovlp - (Ovlp - 1)$ $Nspav = 15$

shift (in samples) between overlapping segments $Oshift := \frac{Nps}{Ovlp}$

Number of resulting spectra $Nspec := \frac{Nt}{Nps \cdot Nav}$ $Nspec = 64$ Indexing the spectra $jspec := 0 .. Nspec - 1$

Input data Block length for a single averaged output spectrum $Bav := Nps \cdot Nav$

Time stamps for resulting spectra $tspec_{jspec} := (jspec + 0.5) \cdot Bav \cdot dt$ Output sampling interval $Bav \cdot dt = 0.262144\ s$ $tspec_0 = 0.131072\ s$

Padding array $Npadd := Nps \cdot (Padd - 1)$ $Npadd := if(Padd = 1, 1, Npadd)$ $jpad := 0 .. Npadd - 1$ $dpadd_{jpad} := 0$

Single-sided Output spectrum binning
for 2-sided FFT $Nfp := \frac{Nps \cdot Padd}{2} + 1$ $jfs := 0 .. Nfp - 1$

Output frequency resolution,
accounting for padding $dfs := \frac{1}{tw \cdot Padd}$ $dfs = 15.2587890625\ s^{-1}$

and Output spectrum frequency grid $fs_{jfs} := jfs \cdot dfs$

Read Doppler Frequency polynomial

$Fcd0 := READPRN(DpcFileName0)$

$Fcd1 := READPRN(DpcFileName1)$

Polynomial order $Npf := Fcd0_0$ $Npf = 3$ indexing $jpf := 0..Npf$

Normalization time $Tnorm := Fcd0_1$ $Tnorm = 16.777216$ $Tnorms := Tnorm \cdot s$ $Tspan = 16.777216\ s$

$Tnorms = 16.777216\ s$

Frequency coefficients $Cf0_{jpf} := Fcd0_{jpf+2}$

$Cf1_{jpf} := Fcd1_{jpf+2}$

$Fcd1_1 = 16.777216$

$Cf := Cf0 + Cf1$

Detected frequency polynomial

$$Cf = \begin{pmatrix} 70048.91028919304 \\ 7052.868226490915 \\ -7888.813451185823 \\ 1.104737497866154 \end{pmatrix}$$

Read input model phase coefficients $Cpm := READPRN(PhModFileName)$ $Cpm_4 := 0$

Make Phase polynomial from detected frequency $Cpp_0 := 0$ $Cpp_{jpf+1} := 2\pi \cdot \frac{Cf_{jpf}}{jpf + 1}$

Pass 1 detected phase polynomial

$$Cpp = \begin{pmatrix} 0 \\ 440130.2839129986 \\ 22157.23900708073 \\ -16522.292255857148 \\ 1.735317603720739 \end{pmatrix}$$

Input phase model polynomial

$$Cpm = \begin{pmatrix} 3.083301383650139 \\ 440132.6942198543 \\ 1320.398082667224 \\ -58.6843592296544 \\ 0 \end{pmatrix}$$

Coefficients look rather different..

Compute the phase, ignoring first 2 terms, which are Constant Initial phase and Constant Initial Angular Frequency

Do it for Pass 1 detected phase

$$Phdopp_{jt} := Tnorm \cdot \sum_{jjp=2}^{Npf} \left[Cpp_{jjp} \cdot \left(\frac{tt_{jt}}{Tnorms} \right)^{jjp} \right]$$

And for input model coefficients

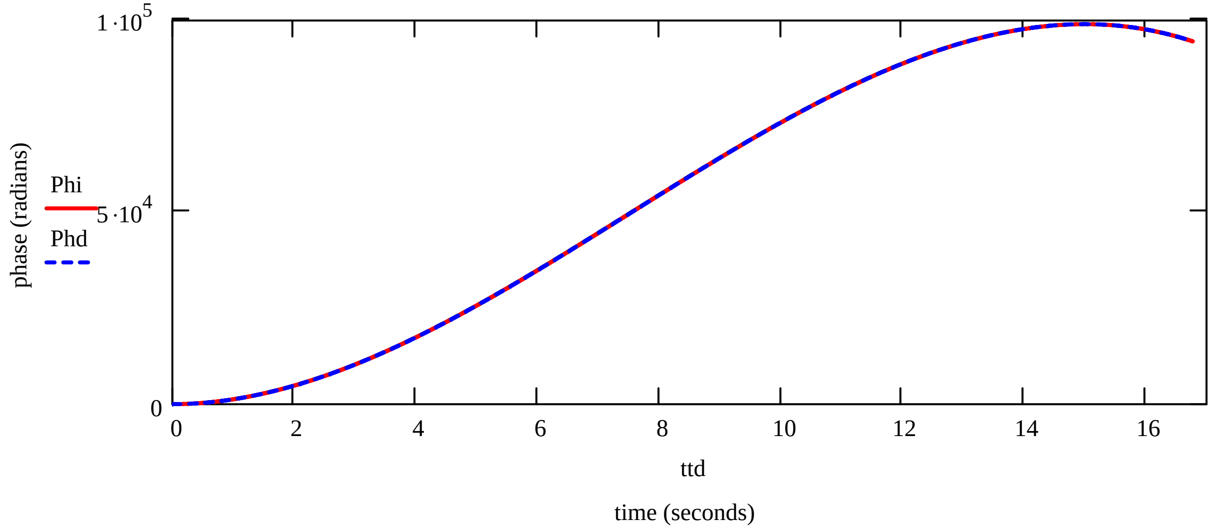
$$Phinput_{jt} := \sum_{jjp=2}^{Npf} \left[Cpm_{jjp} \cdot \left(tt_{jt} \cdot s^{-1} \right)^{jjp} \right]$$

Compare the detected and theoretical phases over scan time range

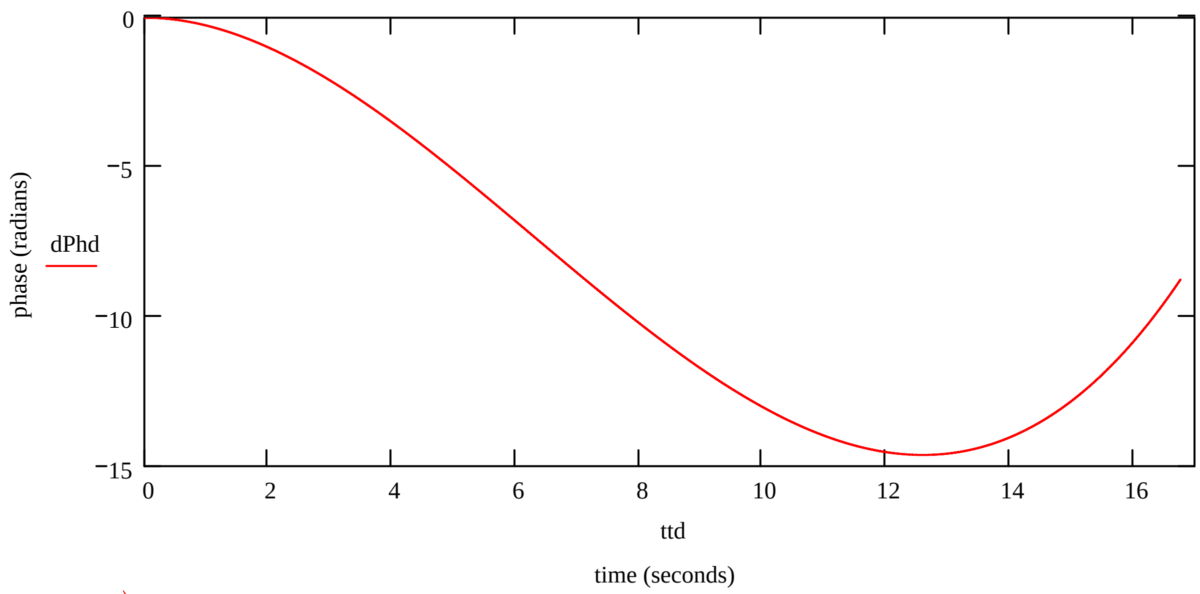
$\mathrm{Ndec} := 1000$ $\quad \mathrm{jtd} := 0 .. \frac{\mathrm{Nt}}{\mathrm{Ndec}} - 1$ $\quad \mathrm{ttd}_{\mathrm{jtd}} := \mathrm{tt}_{\mathrm{jtd}\cdot\mathrm{Ndec}}$ $\quad \mathrm{jtd1} := 0 .. \frac{\mathrm{Nt}}{\mathrm{Ndec}} - 2$

$\mathrm{Phd}_{\mathrm{jtd}} := \mathrm{Phdopp}_{\mathrm{jtd}\cdot\mathrm{Ndec}}$ $\quad \mathrm{Phi}_{\mathrm{jtd}} := \mathrm{Phinput}_{\mathrm{jtd}\cdot\mathrm{Ndec}}$ $\quad \mathrm{dPhd} := \mathrm{Phi} - \mathrm{Phd}$

Detected phase - red, theoretical phase - blue



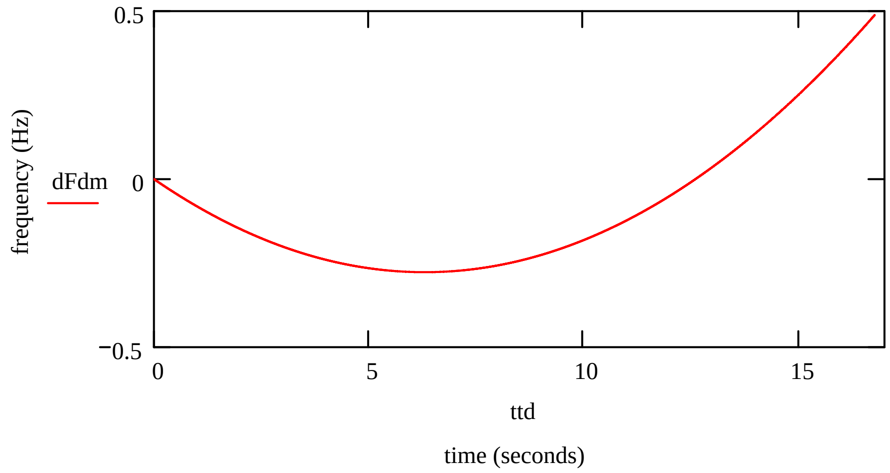
Difference between the two



$\mathrm{WxRITEPRN}(\text{"Data3.Det.diffPhase.txt"}) := \mathrm{augment}\left(\mathrm{ttd}\cdot s^{-1}, \mathrm{dPhd}\right)$

Also compute and plot the differential frequency between two phase models

$$dFdm_{jtd1+1} := \frac{dPhd_{jtd1+1} - dPhd_{jtd1}}{2\pi \cdot dt \cdot Ndec}$$



Major function to compute a dynamic spectrum
Input : Jspec - averaged spectrum number to compute,
Phcorr - model of the Doppler phase correction.
Parameters defined in the text above:
Nfp - number of output spectral bins
Nps - number of input samples to FFT (not accounting for padding)
Nspav - number of spectra to average to produce a single output spectrum
Bav - input data block length on which the output spectrum is computed
Ovlp - Overlapping parameter
dtn and Tspann - sampling interval and scan length
Padd and dpadd - padding coeff and padding array

```
MakeSpec(Jspec, Phcorr) := | for jj ∈ 0 .. Nfp − 1                                   Initialize an array for spectrum accumulation
                           |   spa_jj ← 0
                           | for jspav ∈ 0 .. Nspav − 1
                           |   | skip ← Jspec · Bav + jspav · Nps/Ovlp               Compute a skip pointer to get a proper data segment
                           |   | din ← READBIN(DataFileName, "byte", 0, 1, skip, Nps) Read data segment to be FFTed
                           |   | din ← din − 127
                           |   | phc ← submatrix(Phcorr, skip, skip + Nps − 1, 0, 0)  Read the current correction phase for data segment
                           |   | ephc ← vectorize(exp(i · phc))                      Compute a complex exponent of phase
                           |   | din ← vectorize(din · ephc)                         Multiply data segment by complex exponent and window
                           |   | din ← vectorize(din · Win)                          Note the signal becomes complex after phase correction
                           |   | dinp ← if (Padd > 1, stack(din, dpadd), din)        Padd with zero array for super-resolution, if asked for
                           |   | sp ← cfft(dinp)                                     Here the two-sided complex-to-complex FFT is used because
                           |   | for jj ∈ 0 .. Nfp − 1                               after Doppler correction the input signal becomes complex
                           |   |   spa_jj ← spa_jj + Re(sp_jj)^2 + Im(sp_jj)^2       Accumulate the power spectrum, only positive frequencies of FFT
                           | return spa                                              Return the power spectrum for a given averaging interval
```

$$\text{MakeSpec}(\text{Jspec},\text{Phcorr}) := \left|\begin{array}{l} \text{for } jj \in 0..\text{Nfp}-1 \\ \quad \text{spa}_{jj} \leftarrow 0 \\ \text{for } \text{jspav} \in 0..\text{Nspav}-1 \\ \quad \left|\begin{array}{l} \text{skip} \leftarrow \text{Jspec}\cdot\text{Bav} + \text{jspav}\cdot\dfrac{\text{Nps}}{\text{Ovlp}} \\ \text{din} \leftarrow \text{READBIN}(\text{DataFileName},\text{"byte"},0,1,\text{skip},\text{Nps}) \\ \text{din} \leftarrow \text{din} - 127 \\ \text{phc} \leftarrow \text{submatrix}(\text{Phcorr},\text{skip},\text{skip}+\text{Nps}-1,0,0) \\ \text{ephc} \leftarrow \overrightarrow{\exp(i\cdot\text{phc})} \\ \text{din} \leftarrow \overrightarrow{(\text{din}\cdot\text{ephc})} \\ \text{din} \leftarrow \overrightarrow{(\text{din}\cdot\text{Win})} \\ \text{dinp} \leftarrow \text{if}(\text{Padd}>1,\text{stack}(\text{din},\text{dpadd}),\text{din}) \\ \text{sp} \leftarrow \text{cfft}(\text{dinp}) \\ \text{for } jj \in 0..\text{Nfp}-1 \\ \quad \text{spa}_{jj} \leftarrow \text{spa}_{jj} + \text{Re}(\text{sp}_{jj})^2 + \text{Im}(\text{sp}_{jj})^2 \end{array}\right. \\ \text{return spa} \end{array}\right.$$

Initialize an array for spectrum accumulation

Compute a skip pointer to get a proper data segment

Read data segment to be FFTed

Read the current correction phase for data segment

Compute a complex exponent of phase

Multiply data segment by complex exponent and window

Note the signal becomes complex after phase correction

Padd with zero array for super-resolution, if asked for

Here the two-sided complex-to-complex FFT is used because
after Doppler correction the input signal becomes complex

Accumulate the power spectrum, only positive frequencies of FFT

Return the power spectrum for a given averaging interval

Make a dynamic spectrum $Spd^{\langle jspec \rangle} := \text{MakeSpec}(jspec, Phdopp)$ With detected phase correction

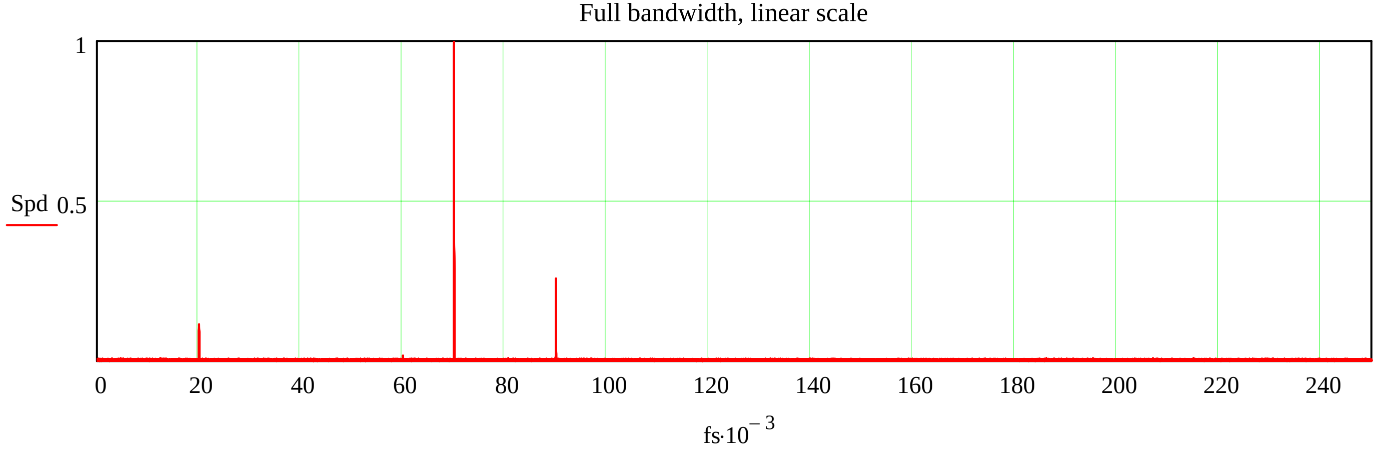
Make a dynamic spectrum $Spm^{\langle jspec \rangle} := \text{MakeSpec}(jspec, Phinput)$ With input phase correction

Prepare for plotting $xSpd := \max(Spd)$ $Spd := Spd \cdot xSpd^{-1}$

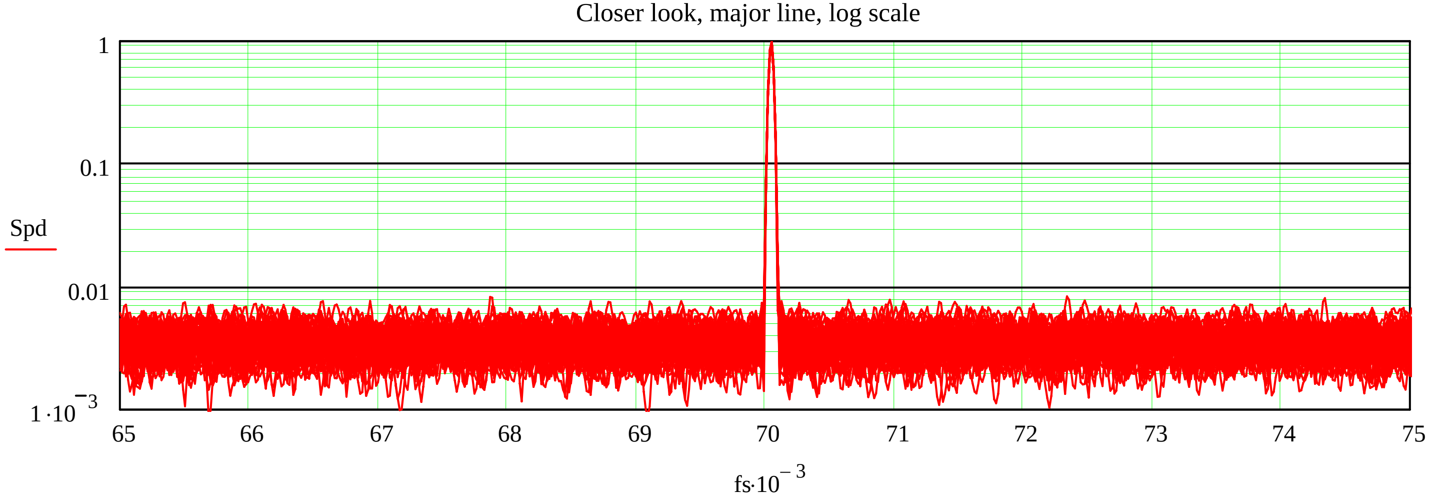
$xSpm := \max(Spm)$ $Spm := Spm \cdot xSpm^{-1}$

Show the spectra (frequency scale in kHz)

Dynamic spectrum after correction with the detected phase model



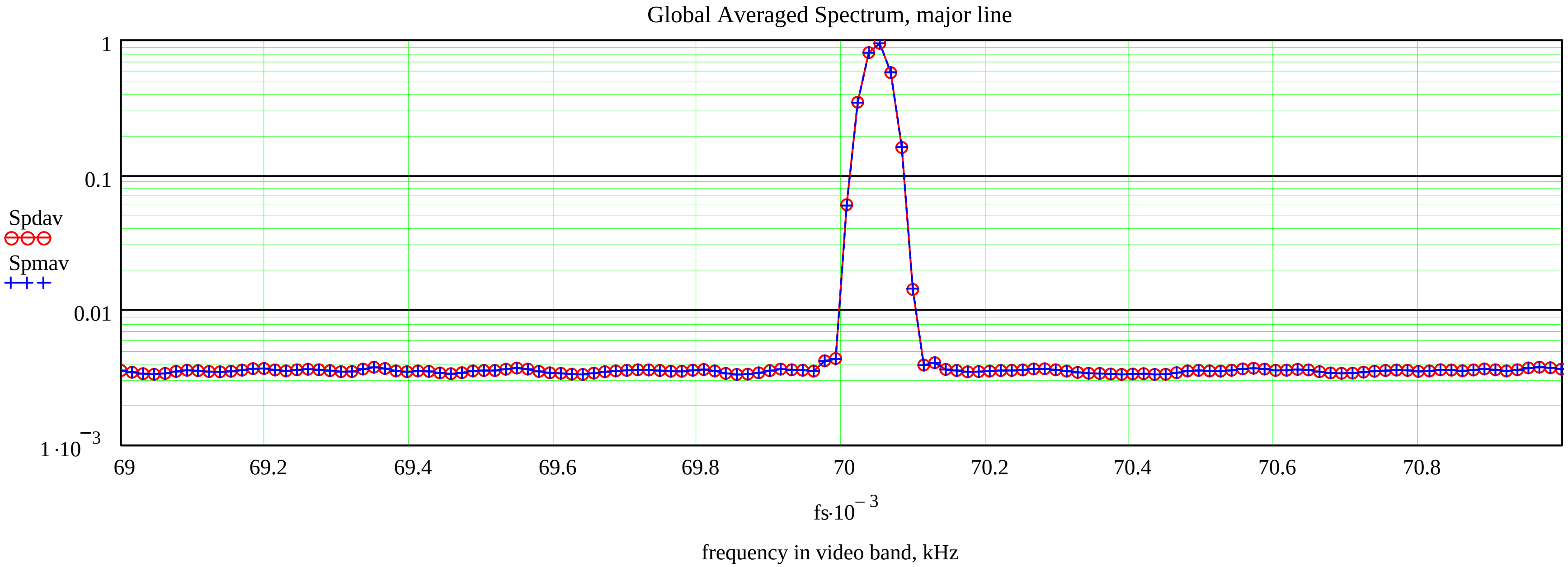
All the $Nspec = 64$ spectra are presented in overlay



Global averaged spectrum

$$Spdav_{jfs} := \frac{1}{Nspec} \cdot \left(\sum_{jspec} Spd_{jfs,\, jspec} \right) \qquad Spmav_{jfs} := \frac{1}{Nspec} \cdot \left(\sum_{jspec} Spm_{jfs,\, jspec} \right)$$

Both spectra shown, corrected with detection - red, circles, corrected with the input model - blue, crosses



Width is about 50 Hz, with about 30 Hz resolution

Analyze the results

Input : spectrum, frequency scale,
search window defined by max and min frequency.

Output : integer bin number of the max value,
parabolic estimate of max position,
max value itself

$$\mathrm{FindMax}(Spec, Fscale, Fmin, Fmax) := \left|\begin{array}{l} np \leftarrow \mathrm{length}(Spec) \\ mx \leftarrow 0 \\ \text{for } jj \in 0 .. np-1 \\ \quad mx \leftarrow \mathrm{if}\left(Spec_{jj} > mx \wedge Fmin < Fscale_{jj} < Fmax, Spec_{jj}, mx\right) \\ jmax \leftarrow 0 \\ \text{for } jj \in 0 .. np-1 \\ \quad jmax \leftarrow \mathrm{if}\left(Spec_{jj} = mx \wedge Fmin < Fscale_{jj} < Fmax, jj, jmax\right) \\ jmax \leftarrow \mathrm{if}(jmax = 0, 1, jmax) \\ jmax \leftarrow \mathrm{if}(jmax = np-1, np-2, jmax) \\ a2 \leftarrow 0.5 \cdot \left(Spec_{jmax-1} + Spec_{jmax+1} - 2 \cdot Spec_{jmax}\right) \\ a1 \leftarrow 0.5 \cdot \left(Spec_{jmax+1} - Spec_{jmax-1}\right) \\ djx \leftarrow \dfrac{-a1}{2 \cdot a2} \\ xmax \leftarrow jmax + djx \\ \text{return } \mathrm{stack}(jmax, xmax, mx) \end{array}\right.$$

Input : spectrum, frequency scale, line frequency,
estimation window defined by half width with respect
to the line position and and half width of line avoidance.

Output : mean value in the window and rms in the window

$$\mathrm{GetRMS}(Spec, Fscale, Fline, Fspan, Fvoid) := \left|\begin{array}{l} np \leftarrow \mathrm{length}(Spec) \\ mw \leftarrow 0 \\ ww \leftarrow 0 \\ \text{for } jj \in 0 .. np-1 \\ \quad \left|\begin{array}{l} ww \leftarrow ww + \mathrm{if}\left(\left|Fscale_{jj} - Fline\right| < Fspan \wedge \left|Fscale_{jj} - Fline\right| > Fvoid, 1, 0\right) \\ mw \leftarrow mw + \mathrm{if}\left(\left|Fscale_{jj} - Fline\right| < Fspan \wedge \left|Fscale_{jj} - Fline\right| > Fvoid, Spec_{jj}, 0\right) \end{array}\right. \\ mm \leftarrow \dfrac{mw}{ww} \\ dw \leftarrow 0 \\ \text{for } jj \in 0 .. np-1 \\ \quad dw \leftarrow dw + \mathrm{if}\left[\left|Fscale_{jj} - Fline\right| < Fspan \wedge \left|Fscale_{jj} - Fline\right| > Fvoid, \left(Spec_{jj} - mm\right)^2, 0\right] \\ rm \leftarrow \sqrt{\dfrac{dw}{ww}} \\ \text{return } \mathrm{stack}(mm, rm) \end{array}\right.$$

Detect the frequency of the major tone (in a proper frequency window)

$\mathrm{FsearchMin} := 60 \cdot \mathrm{kHz}$

$\mathrm{FsearchMax} := 80 \cdot \mathrm{kHz}$

Using dynamic spectra after detected correction $\quad \mathrm{xfd}^{\langle \mathrm{jspec} \rangle} := \mathrm{FindMax}\left(\mathrm{Spd}^{\langle \mathrm{jspec} \rangle}, \mathrm{fs}, \mathrm{FsearchMin}, \mathrm{FsearchMax}\right) \qquad \mathrm{Fdet}_{\mathrm{jspec}} := \mathrm{dfs} \cdot \mathrm{xfd}_{1,\,\mathrm{jspec}}$

Using dynamic spectra after model correction $\quad \mathrm{xfm}^{\langle \mathrm{jspec} \rangle} := \mathrm{FindMax}\left(\mathrm{Spm}^{\langle \mathrm{jspec} \rangle}, \mathrm{fs}, \mathrm{FsearchMin}, \mathrm{FsearchMax}\right) \qquad \mathrm{Fmod}_{\mathrm{jspec}} := \mathrm{dfs} \cdot \mathrm{xfm}_{1,\,\mathrm{jspec}}$

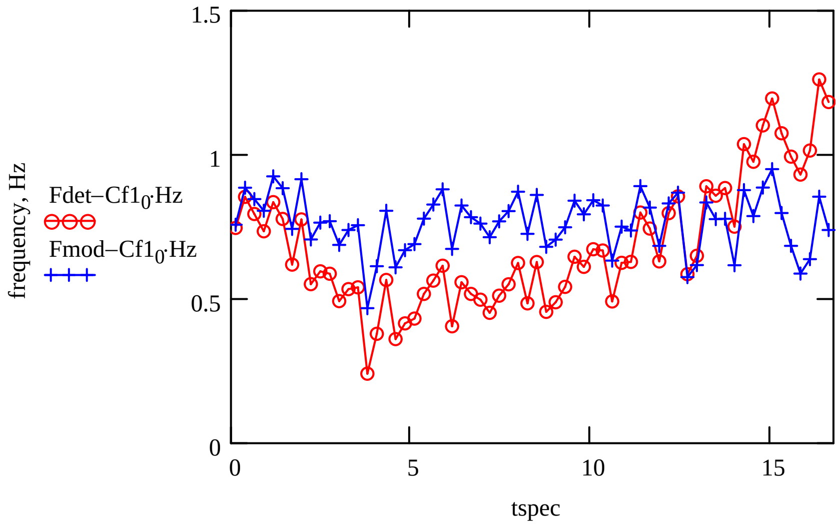
Get SNR

$\mathrm{HalfWindow} := 8 \cdot \mathrm{kHz} \qquad \mathrm{LineAvoidance} := 0.5 \cdot \mathrm{kHz}$

$\mathrm{rmsd}^{\langle \mathrm{jspec} \rangle} := \mathrm{GetRMS}\left(\mathrm{Spd}^{\langle \mathrm{jspec} \rangle}, \mathrm{fs}, \mathrm{Fdet}_{\mathrm{jspec}}, \mathrm{HalfWindow}, \mathrm{LineAvoidance}\right) \qquad \mathrm{Snrd}_{\mathrm{jspec}} := \dfrac{\mathrm{xfd}_{2,\,\mathrm{jspec}} - \mathrm{rmsd}_{0,\,\mathrm{jspec}}}{\mathrm{rmsd}_{1,\,\mathrm{jspec}}} \qquad \mathrm{mean}(\mathrm{Snrd}) = 1012.2748988299695$

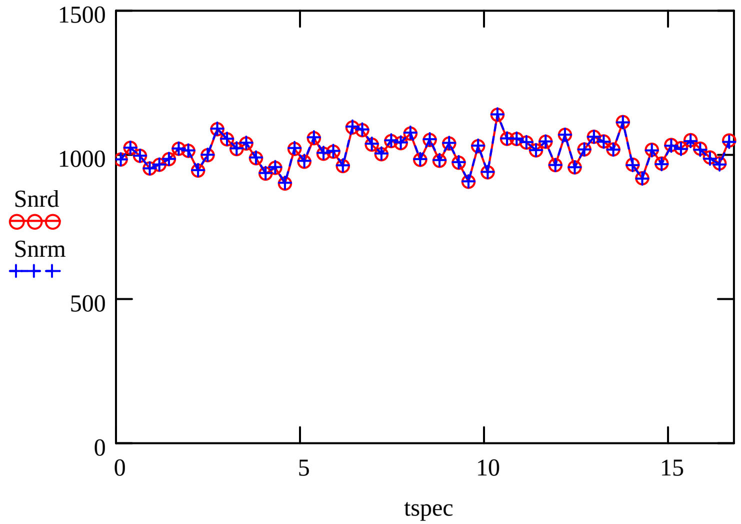
$\mathrm{rmsm}^{\langle \mathrm{jspec} \rangle} := \mathrm{GetRMS}\left(\mathrm{Spm}^{\langle \mathrm{jspec} \rangle}, \mathrm{fs}, \mathrm{Fdet}_{\mathrm{jspec}}, \mathrm{HalfWindow}, \mathrm{LineAvoidance}\right) \qquad \mathrm{Snrm}_{\mathrm{jspec}} := \dfrac{\mathrm{xfm}_{2,\,\mathrm{jspec}} - \mathrm{rmsm}_{0,\,\mathrm{jspec}}}{\mathrm{rmsm}_{1,\,\mathrm{jspec}}} \qquad \mathrm{mean}(\mathrm{Snrm}) = 1013.2471047638651$

Plot the detected residual frequency for two dynamic spectra



time in the scan, seconds

SNR for 2 dynamic spectra

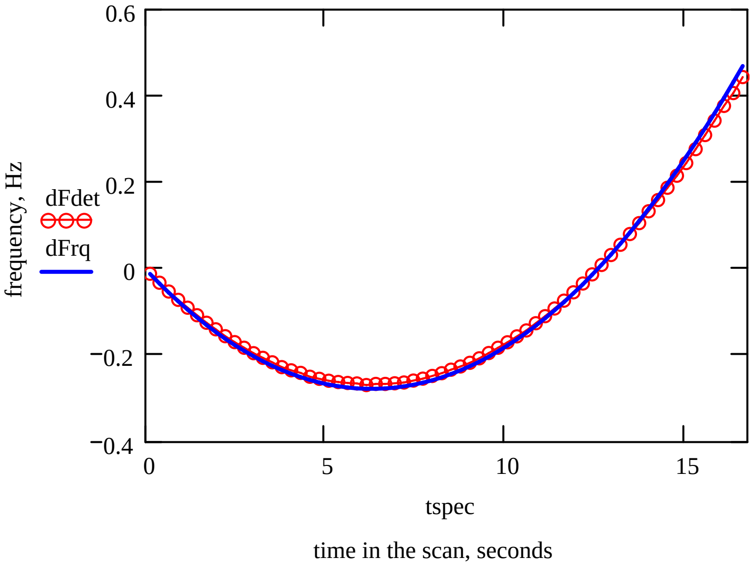


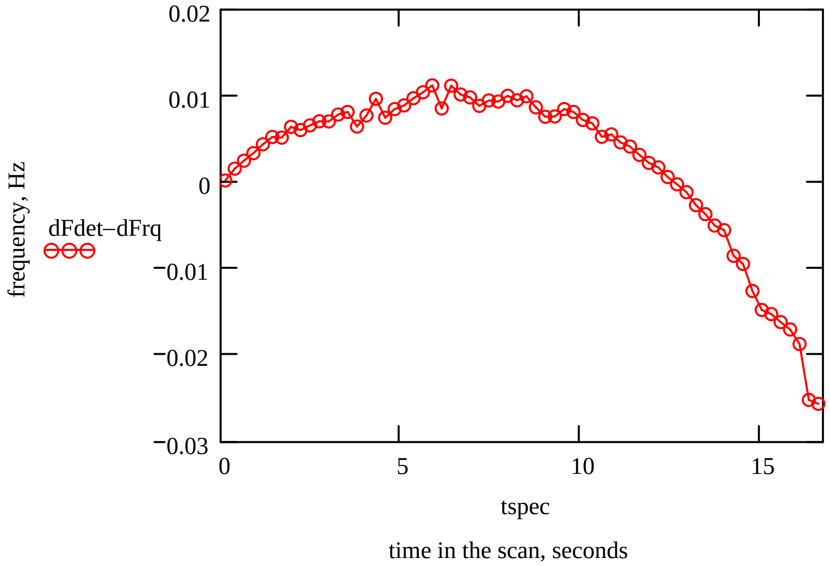
Residual frequency after input model correction is much flatter. Another surpizing thing is that noise on both curves is rather the same!

$dFdet := Fdet - Fmod \qquad dFrq := Spline(ttd, dFdm, tspec)$

Difference between two detected residual frequency curves is plotted as red circles,
while predicted difference, based on polynomials only (as computed above) - blue line

$stdev(dFdet - dFrq) = 0.009139250928488\ s^{-1}$





RMS frequency difference is ~9 mHz, although frequency difference is not random.