I have reproduced the responses that I put in the discussion here.

Reviewer 1 responses

I thank the reviewer for their careful reading of the manuscript and their insightful comments. I have addressed all concerns, which have helped clarify the implementation and applicability of the Segmented-Overlap Fourier-Filtering and Averaging (SOFFA) method. Below is a point-by-point response to the specific and technical comments.

Comment 1: The examples selected for the samples have relatively broad lines. Would the method work as well for spectra with well-resolved (narrow) hyperfine lines? If so, what data acquisition and signal processing parameters would need to be adjusted?

Response: The SOFFA method performs equally well for well-resolved, narrow hyperfine lines. To properly resolve these features, three primary parameters can be adjusted:

1. *Signal Processing:* The variance of the Gaussian filter (σ^2) applied in Fourier space must be minimized relative to the intrinsic linewidth to prevent artificial spectral broadening or distortion.
2. *Data Acquisition:* The field step size (s) can be adjusted. Decreasing s acts as a "fine-tuning" knob for the number of averages by increasing the number of overlapped segments. However, there are hardware limits; for example, on a Bruker E500, the step size is practically limited to 0.1 G, and conservatively 0.25 G, due to the precision of the field controller and magnet hysteresis. For larger increases in the signal-to-noise ratio, a better approach is to use "repetitions" (repeating the full stack of segments). In practice, fine-tuning the effective averages is achieved via adjusting the step size, while doubling or multiplying the averages is achieved via repetitions.
3. *Rate through the spectrum:* The rate that you go through per segment should not be faster than the rate you go through for the whole spectrum. i.e. if you run 100G over 60s, then you should not run the 20G segment sweep faster than 12s.

I have added a paragraph to the Methods section detailing the adjustments required for narrow hyperfine lines, specifically noting the scaling of σ^2, the limits of the field step size (s) due to hardware constraints, and the distinction between using step size for fine-tuning versus repetitions for larger signal averaging and how to choose the rate.

Comment 2: Line 178 - what tools are used for 'noise-shaping?

Response: The term "noise-shaping" in this context refers to the suppression of high-frequency noise outside the signal's spectral bandwidth. The specific tool utilized to achieve this is the Gaussian filter applied via block-filtering techniques.

I have modified Line 178 to read: "...noise-shaping can be employed to reduce the high frequency content outside the spectral bandwidth by applying finite-impulse response (FIR) block-filtering techniques."

Comment 3: The caption for Fig. 4 should state that the spectral parameters are the same as for Fig. 3.

Response: Agreed. The caption for Figure 4 has been updated to explicitly state: "Simulated spectral parameters are identical to those described in Figure 3."

Comment 4: In Fig. 5 is the lowest-field feature that dominates the signal for the 10 mM sample a background signal? If so, that should be clarified. The caption does not define the significance of the asterisks. Also, the caption states that the 1 mM reference spectrum was acquired by averaging 9 times, although the description in the main text for this spectrum does not mention averaging. This need clarification.

Response: Note: the sample measured was 10μM (not 10 mM), this was probably lost in translation to the website. The asterisked lowest-field feature in the 10μM sample is indeed a background artifact originating from the resonator. Regarding the 1 mM reference spectrum, it was conventionally averaged 9 times to establish a high signal-to-noise ratio "gold standard" for baseline comparison, which was inadvertently omitted from the main text.

The Figure 5 caption has been updated to define the background signal and the asterisks. Additionally, a sentence has been added to the main text confirming that the 1 mM reference spectrum utilized 9 conventional averages to establish the comparative baseline.

Comment 5: The method was implemented on a Bruker spectrometer, which presumably means that sinusoidal modulation scans were used. If so, over what fraction of the sinusoidal scan was assumed to sufficiently linear to be used in reconstructing spectra?

Response: The SOFFA-CW method utilizes discrete, static field stepping rather than a rapid sinusoidal sweep. The field is stepped, and data are collected at each static point. Therefore, the linearity assumptions required for reconstructing rapid sinusoidal scans do not apply here. With SOFFA-NARS, this data used 80% of the sinusoidal modulation in one direction. Interestingly, small edge effects show up as periodic noise outside of the spectral PSD and are averaged away.

I have expanded the description in the Methods section (around Line 62) to clearly state that the ProDEL program relies on discrete, static field stepping, and lock-in detector data acquisitoin thereby avoiding the non-linearities associated with sinusoidal sweeps. I also added a sentence in the SOFFA-NARS to comment about the sinusoidal sweep.

Comment 6: How much software overhead is there in the implementation with ProDEL?

Response: The software overhead introduced by the ProDEL script is negligible. The script's primary function is to iterate a static field step, trigger data acquisition, and save the resulting 2D dataset. The time-limiting factor is the hardware settling time of the spectrometer's magnet between steps, not the software execution. This has been tested up to a 8196 point field sweep with 1200 steps (0.1 G over 120 G).

A brief sentence has been added to Section 2 (Methods) noting that computational overhead is negligible and the acquisition speed is primarily constrained by hardware field-settling times.

Comment 7: How sensitive are the S/N improvements to the extent of oversampling that is reduced in the final decimation step?

Response: The oversampling is one of the four pillars of the SOFFA method and this allows for the better separation between high frequency filtering and low-frequency averaging. However, the final decimation step acts essentially as a moving average. It conserves the S/N gains achieved during the Fourier filtering and overlap-averaging stages, simply reducing the array size for conventional formatting. The reduction of points attributes an improvement by the square root of the decimate factor in regards to high frequency noise.

I have expanded Section 3.4 (Decimation) to explicitly state that the decimation process conserves the signal-to-noise improvements generated by the upstream SOFFA processes but does act as a final low-pass filter on the system providing a final square root of the decimation factor.

I would also like to highlight that the SOFFA method circumvents hardware-based memory limitations on older spectrometers. For instance, legacy systems (such as some Bruker E500 consoles) are often hardware-limited to 8192 (8k) data points per continuous sweep. By breaking a broad spectrum into segments, SOFFA decouples the total experimental resolution from this hardware limit. For example, acquiring a 100 G spectrum using 1 G steps where each step utilizes 4096 points results in an effective acquisition of 409,600 data points across the full spectral width—a level of oversampling impossible to achieve in a single conventional sweep on such hardware.

I have added a paragraph highlighting how segmenting allows users to bypass hardware point-limits, using the 409,600 effective point example to demonstrate maximizing the OSR on legacy spectrometers.

Comment 8: The ProDel code is available through github. However the Mathematica code and sample data are only available 'on request'. The paper should only be published if the code and sample data are made publicly available.

Response: I agree that full transparency is necessary for implementation and reproducibility. The Mathematica code is very much in an "engineer" state. However, I have uploaded the most recent version of my python GUI script that I now use on a daily basis to the existing public GitHub repository. The "Code and Data Availability" statement in the manuscript has been updated to reflect this open-source access.

Comment The word ‘data’ is a compound noun which means that the accompanying verbs should be plural.

Response: I appreciate the grammatical correction. A global review of the manuscript has been performed to ensure proper subject-verb agreement (e.g., "Data are collected").

Comment: In the caption for Fig. 1 what is the meaning of the phrase '...$y_{tot}[i]$ are related to the segment sweep width'?

Response: The intent was to state that the discrete indices of the final array map to the physical units of the sweep. I have rewritten the final sentence of the Figure 1 caption to: "The discrete indices of the collected segments $x_k[j]$ and the final concatenated output $y_{tot}[i]$ map directly to the physical sweep axis (e.g., mT for field sweeps or MHz for frequency sweeps)."

Reviewer 2 responses

Thank you again for these fantastic edits. I have significantly rewritten the paper, but I would like to take the time to answer some specific comments.

(1) It is very difficult to understand the method, since its description is long-winded, partially redundant and spread over three sections (Methods; Theory; Appendix C). The notation is confusing. Pseudo-code or actual code for the proposed processing method would be crystal clear, but it is missing (there is no supplementary material). Therefore, it was impossible for me to reproduce and test the claims in this manuscript. The method description should be consolidated, shortened and made more consistent to increase readability. A Python script should be provided that takes acquired segments and required processing parameters as inputs and produces a filtered spectrum as output. This would enhance the impact of this work considerably.

I fully agree. I have included in github the ProDel script and a new version of the reconstruction in python. In rewriting this manuscript I realized that I was explaining the same thing in different ways to ensure there was an understanding. Over the time this manuscript was written and the rewriting my thoughts on this method have become more clear. I hope this is aparent in the newest version.

(2) The work does not clearly explain the origin of the apparent SNR improvements. Therefore, it does not convince me that the claim of SNR improvement is actually valid. In general, any improvement in SNR during processing comes at the cost of a reduction in bandwidth. One can always increase SNR by low-pass filtering a signal, making sure the filter stopband and passband droop do not encroach on the signal band. Also, it appears that the SOFFA method acquires more points in the same amount of time (see Fig. 7 for instance; "oversampling" is mentioned a lot) - for a bandwidth-limited signal this would provide more data, and an improved SNR would be the trivial consequence. What is the true origin of the (apparent) SNR improvement?

I agree that the manuscript does not have a SNR gain equation, but this work shows the implementation. I want to stress that the gain comes from the combination of all of the parts, which makes this difficult to pin down. The use of oversampling (more points for the same sweep; say 1024 to 8192) is always available to the spectroscopist and you will gain by the ability to filter high frequency noise. However, SOFFA uses segments at 4096 or 8192 points which creates an oversampling that is typically limited by the hardware. Additionally, since a segment can contain a portion of the spectrum the bandwidth required to filter is significantly reduced. This is then coupled with averaging and then decimating down to a standard number of points (e.g. 1024). The exact gain equation is reserved for future work, but it will be a combination of oversampling, filtering improvements, averaging, and decimation. Where filtering and decimation are not multiplicative because they filter the same kind of data in different ways. This becomes even more interesting (and out of scope of this paper) when each segment is filtered by a match filter (Weiner) and the power spectral density is matched and the filter is variable based on the spectral information in each segment. Traditional CW has no corresponding filter ability.

I hope the rewrite of this manuscript describes this in sufficient detail and that the experimental examples are sufficient to warrant publication.

- It is unclear what t represents. In Figs.1 and B1, t appears to be a time or time index, defining the spectral axis. In Eq.(2), it is the total spectral width (which is indicated as g in Eq.(1)). What is the difference between t and g?

- It is unclear which of the parameters t, s, g, N, L, m are the independent ones and which are the dependent ones.

I have generated a table with all parameters discussed and removed figure 1.

- Eq.(1): Why is this formulated as an estimate - there must be a precise relationship between integer n and the right-hand side quantities.

This is because one may need to round up to the nearest integer value since this is discrete timepoints.

- Eq.(2) appears not general enough. Assume the sweep width is 10 G, the step size is 1 G, and m is set to 5, then there are only 6 segments, but this equation gives K = 10. It appears the equation is only correct if m=1.

- Line 81: D is not defined. Is it equal to N-L, as inferred from Fig.1? This should be stated explicitly.

These have been replaced in the discussion and placed in the Table.

- Eq.(3) mentions frequency omega, whereas up to this point only field sweeps and their FT are discussed. Unclear how the frequency relates to the field.

- gamma in Eq.(3) is defined as the "central point in the Fourier transformed data". What does this mean? The expectation value of the frequency over the FT of the field sweep? Add mathematical definition.

Omega is the discretized frequency of the

- Line 87: Provide more details about the concatenation. Are all segments just appended? If m=5 and there are K=6 segments, this leads to a total length m*K = 30 segments, which exceeds the length of the original field sweep. This probably boils down to what a "segment" is. Fig.1 and Fib.B1 indicate that it includes the overlap regions - segments are 3*n long in both figures. Fig.1 caption says "segments x_k[j]".

Concatenation with overlap is averaging. This has been clarified in the new version.

- Line 93: How is the "mean value of the peak-to-peak signal" defined? Is the peak-to-peak height averaged over several points, or several repeat measurements?

This was used for the simulation data definition. I have removed the simulated data section and removed the term "mean." SNR is picking the peak-to-peak as the amplitude and comparing it to the standard deviation of the noise at some off resonance region.

- Line 98: Mathematica doesn't appear to have a RandReal function, but a RandomReal function.

Changed.

- Line 100: The use of a uniform distribution for (white) noise is entirely non-physical. For instance, it has a non-zero mean, whereas physical noise (white or pink) has zero mean. A normal distribution should be used instead.

Agreed, however, the uniform distribution employed in the simulation was symmetric and zero-mean, U(−a,a)U(-a,a) U(−a,a), satisfying the zero-mean condition. More fundamentally, white noise is defined by its flat power spectral density rather than its amplitude distribution, and the SNR improvement factors reported depend only on the variance reduction under averaging. THis is a property shared by all zero-mean, finite-variance distributions. However, I have removed the simulated spectra in favor of just showing collected data.

- Line 103: What is the reason that white and pink noise are combined? The noise model used should match the actual noise as it occurs in experimental spectra.

white noise creates a noise floor for pink noise. There is always an "elbow" of when white noise dominates over pink noise. However, I have removed the simulated spectra in favor of just showing collected data.

- Line 121: "either random (white) or frequency-dependent (pink) noise". This is inaccurate language. All noise, including pink, is random. White noise has a frequency-independent power spectrum.

agreed. This has been modified in the text.

- Paragraph starting Line 140: It is not clear how the discussion of parallel processing and real-time processing is relevant here. This should be clarified.

The intent was to talk about data processing overhead. It has been removed because this is primarily a post-processing method.

- Line 158 says that there are k total segments, whereas Eq.(2) says it is K. Make notation consistent.

Thank you k is the index, K is the total.

- Line 160 mentions a "zero-fill parameter". How is it defined?

To line up all segments in the appropriate field bin position a sparce matrix is created with a zero filling at the beginning and end of the data.

- "Oversampling" is mentioned many times throughout the manuscript. This term should be defined. Does it relate to the Nyquist sampling criterion? The FT of a Gaussian line is a Gaussian, so theoretically the signal is not band limited, and there is no oversampling.

Oversampling is the traditional Nyquist sense, but us as spectroscopists don't normally think about it when collecting CW data. For many spectra, 1024 points is more than needed to reproduce our spectrum in the field axis. In SOFFA, oversampling occurs because each segment of sweep width is collected with as many points as would conventionally represent the entire spectrum of width, meaning the sampling density exceeds the Nyquist requirement for that segment by a large factor. The consequence is that noise power is spread across a bandwidth far wider than the signal occupies, allowing a narrow filter to remove the excess noise and improve SNR. For me, it has helped to think about the power spectral density of the spectral information. Appendix A.

- Line 214: A moving-average decimation filter is possibly suboptimal for this step, since it has the worst sidelobe behavior. Are there better alternatives? What about polyphase filtering, see SciPy.signal.resample_poly.

Absolutely! in the python file I have added the ability to use savitzky golay filter. But nothing

- Line 258: Correct subscripts for BGO chemical formula.

Fixed

- Section 4.2 and Figure 5: The FT of the CW and SOFF-CW spectra should be shown, so that the frequency composition of the noise remaining after filtering/averaging can be assessed.

In general, I have used the lack of line broadening as my indicator of the aggressiveness of the filter. I have uploaded a figure showing for the 100uM data. It is very interesting, but I do not think I want to include it into the manuscript. It is clear that the SOFFA data is not broadening the spectrum shown by the residuals. And that there is the clear low frequency spectral data that is filled shown by the raw cw (zoomed below 0.05 c/s). The falloff of the noise is the same, suggesting the filtering is the same but there is no variation in the noise power. What I find really interesting is the differences in the spectral features (arguably sub 0.01 c/s). This exemplifies the separation of high frequency filtering from low frequency averaging.

- Figure 5B is not referenced anywhere in the text.

Fixed. Thank you.

- Paragraph starting in line 288: issues with citations.

- Figure 7 caption: typo "On 13C"

- Figure 7 caption: "3mT sweep ..., with 100 steps of 0.1 mT each" - unclear how this applies to the total sweep width of 10 mT. Is the 0.1 mT the step size s from

Fig.1? That would give 100 subdivisions of the 10 mT range. Does a 3 mT sweep then mean that m=3/0.1=30? Clarify.

- Figure 7 should give the SNRs for both spectra and both noise models.

Great addition!

- The paragraph starting line 302 appears to describe a method to generate 1/f noise. Is this consistent with the use of AudioGenerator["pink"] earlier on? Also, is there are reference for this 1/f generation algorithm?

This was built into Mathematica.

- Line 316: muM -> µM

fixed

- Line 318: This is the first time phase noise is mentioned. If this is relevant, it should be introduced earlier. Its effect on the measured amplitude noise as well its behavior under the SOFFA method should be discussed. Experimentally, what is the source of this phase noise?

Phase noise is equivalent to 1/f noise.

- Line 332: "be adjusted to optimize the oversampled segments". What aspect of the segments is optimized? Maximize the SNR? Minimize the acquisition time? Details would be helpful here.

- Line 335: "higher frequency-content" -> "higher-freuqency content"

- Line 355: Remove first names from citation

- Line 378: ProDEL code is mentioned, but not included.

- Line 381: A Mathematica script is mentioned, but not included. I suggest providing a Phython script instead, since access to Mathematica is rare, and Python is free.

- Line 398: Brackets around citations

Worked diligently on this.