



Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II.

<http://www.jstatsoft.org/>

SILIA: Software Implementation of a Lock-in Amplifier for the Extraction of Periodic Features From Data

Amrut Nadgir

University of California Berkeley
and Lawrence Berkeley National
Laboratory

Richard Thurston

University of California Berkeley
and Lawrence Berkeley National
Laboratory

Kirk A. Larsen

University of California Berkeley
and Lawrence Berkeley National
Laboratory

Niranjan Shivaram

Purdue University

Matthew M. Brister

Lawrence Berkeley
National Laboratory

Daniel S. Slaughter

Lawrence Berkeley
National Laboratory

Abstract

We describe **SILIA**[[amrut_nadgir_richard_thurston_silia_2021](#)], a software implementation of a multi-channel, multi-frequency Lock-in Amplifier to extract modulated signals from noisy data distributed over collections of channels with arbitrary dimensionality and size. This software implementation emulates the functionality of a multi-channel, multi-frequency lock-in amplifier in a post-processing step following data acquisition. Unlike most traditional lock-in amplifiers, **SILIA** can work with any number of input channels and is especially useful to analyze data with a high dimensionality. We demonstrate the versatility and performance for extracting weak signals in spectroscopy and fluorescence microscopy. We also discuss more general applications and exhibit a method to automatically estimate error from a lock-in result.

Keywords: lock-in amplifier, multi-channel, spectroscopy, microscopy, time series.

1. Introduction

Lock-in amplifiers are widely used to suppress noise from a periodic signal generated at a known frequency [1]. This noise suppression is accomplished by using phase-sensitive techniques to remove the signal components that are not modulated at the desired frequency. Early lock-in amplifiers were analog devices, while most modern lock-in amplifiers are fully digital or implemented on a field programmable gate array. Digital lock-in amplifiers tend to be simpler than their analog counterparts, while also achieving wider bandwidth, greater adjustability, higher dynamic reserve and improved accuracy [2–6]. Despite the versatility and broad applications for digital lock-in amplifiers, there are relatively few examples of software implementations of lock-in amplifiers, particularly with highly parallel measurement capabilities for multiple channels and multiple reference frequencies. By implementing a software lock-in amplifier, we adopt the approach where the amplification is a post-processing step after the data acquisition is completed. A software implementation increases the adaptability of the software to the task at hand and simplifies the lock-in process so it can be easily modified to different specifications.

Software Implementation of a Lock-In Amplifier (**SILIA**) can be used to analyze data from an amplitude modulated system, having any number of signal channels and frequency references. As a result, it can serve as a simple yet flexible and cost effective alternative for hardware lock-in amplifiers that are commonly used for multi-channel and multi-frequency applications. Many of the experiments and techniques involving lock-in amplification, such as lock-in imaging [7–10], pump-probe spectroscopy [11–14] and pump-probe microscopy [15, 16], require the use of specialty hardware that can perform the necessary computations during the course of the experiment. **SILIA** is compatible with any instrumentation that can sample data at or above the reference frequency and store the unprocessed data for **SILIA** post-processing. **SILIA** is designed for multi-dimensional data having many channels, provided the data can be stored to local memory.

SILIA offers several benefits not available from traditional lock-in amplification. It can interface with any time series in a specified format. It has the capability to repeatedly retrieve the signal amplitude and relative phase of a weak signal after the initial data acquisition while modifying input parameters and performing post-processing analysis. The software is customizable and readily adapted to a wide variety of spectroscopic or imaging instrumentation. Since **SILIA** is applied as a post-processing step, the speed of the data acquisition is only limited by the digital instrumentation and there are no additional hardware optimizations required for execution. The software can also easily be manipulated to suppress noise from varied forms of measurement, such as frequency sweeps [17]. **SILIA** is also able to estimate the error in the results by partitioning the input signal into overlapping intervals and locking into each of these subintervals.

In addition, **SILIA** is advantageous to using simpler computational error reduction methods such as background subtraction and averaging because it has the ability to extract phase information and can lock into multiple overlaid signals simultaneously.

In this article, we will first discuss the principles behind lock-in amplification and how **SILIA** implements them in software in Section 2. In Section 3, we perform a basic multi-signal functionality test for the software showcasing the capabilities of **SILIA** to perform lock-in on multiple channels and references simultaneously, and produce an accurate estimation of the error for its result. Then, we benchmark the runtime and noise suppression abilities of **SILIA**.

in Section 4. We then apply **SILIA** to spectroscopy and a fluorescence microscopy simulations in Section 5 and present the results. Finally, we discuss the benefits in Section 6, some possible drawbacks of **SILIA**, as well as several possible areas for improvement. Revisions, comprehensive coding examples and tutorials can be found on our Git repository[18].

2. Implementation

A Lock-in amplifier generally performs signal mixing followed by low-pass filtering [1]. The signal mixing step multiplies the input signal by a sinusoidal reference. A sinusoidal reference ensures that the Fourier components of the input signal that oscillate at the reference frequency are shifted to zero frequency in the frequency space representation of the mixed signal. A low-pass filter suppresses the frequency components of the mixed signal that are far from zero frequency. After the lock-in process is completed, the filtered output will be proportional to the amplitude of the frequency components of the input signal within a narrow band centered at the initial reference frequency, thus suppressing noise. The bandwidth is determined by a specified cutoff frequency. The phase of the signal relative to the reference can be extracted by comparing output between the in-phase and quadrature components of the reference [1, 19, 20]. **SILIA** fits a sinusoidal function to the references and locks into each input channel with each reference frequency (Fig. 1). If there are n input channels and k desired frequency references, then the software will output nk lock-in results.

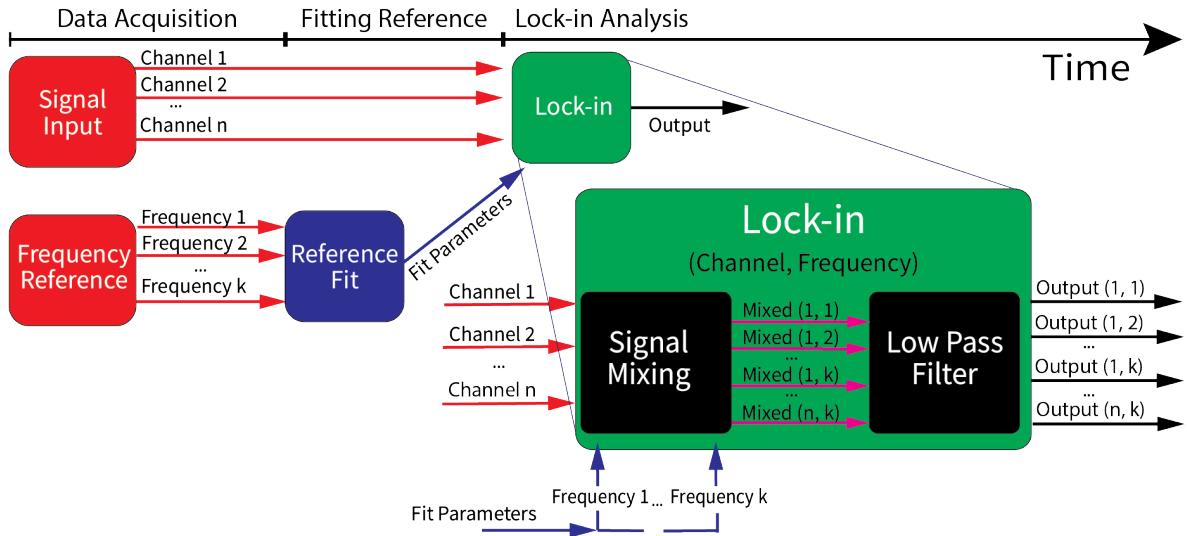


Figure 1: An overview of a single data acquisition cycle and lock-in process, showing the different steps required for the software lock-in analysis in temporal order. The final lock-in amplification step requires the signal data over all n channels and the k fitted frequency references as inputs. The signal mixing and low-pass filter steps are run on the input channels and references where every channel and reference pair can lock in separately. **SILIA** also has the option to skip the reference fit step.

SILIA requires the raw input signal for each channel, frequency references and a cutoff frequency for the low-pass filter. The raw input and frequency references are not passed into the software in real time and are processed after the data acquisition period. The data acquisition

4 **SILIA**: Software Implementation of a Multi-Channel, Multi-Frequency Lock-in Amplifier

device is required to store the measured data with timestamps. The data will be provided to **SILIA** in the form of a python dictionary which contains arrays of the signal data, reference data and their respective timestamps with the appropriate labels [18].

For each reference, the in-phase and quadrature mixed components are computed for each signal channel. To mitigate error from nonuniform sampling rates, **SILIA** has the option to perform linear interpolation on the input signal to ensure evenly spaced samples. Regularly spaced samples are necessary because the Fast-Fourier Transform (FFT) algorithm used in subsequent analysis assumes the samples are evenly spaced. The software then applies a low-pass filter to each of the mixed signals. To implement a low-pass filter in software, we applied the real FFT with a Hanning window to the mixed signal and removed all components outside a symmetric interval centered on 0, with the width specified by a frequency cutoff parameter. This acts as a low-pass filter with a response curve that is essentially a step function, thus minimizing any error caused by the filtering step in conventional lock-in amplifiers, but also assumes that samples are evenly spaced (uniform sampling). Depending on the data acquisition system, discrete sampling of the signal can limit the filter to distinguish adjacent frequency components of the signal. Therefore, the cutoff frequency is limited by the sample rate of the measured signal. Assuming nearly uniform sampling, the frequency resolution tends to improve with longer acquisition times. In addition, the maximum frequency that can be locked into is given by the Nyquist frequency $f/2$ where f is the sampling frequency. Lock-in amplitudes and phases are computed from the measured in-phase and quadrature components of the filtered signals [21]. Each data acquisition cycle produces an array of output values indexed by their respective channel and frequency reference (Fig. 1).

There can be predictable scaling factors when the input waveform is non-sinusoidal. The most common example of this is with square wave input signals. The fourier series expansion of an even square wave with a peak to peak amplitude of A and a duty cycle d is

$$\frac{2A}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} \sin(n\pi d) \sin(2n\pi ft)$$

where t is time and f is the frequency of the square wave [22]. As long as the cutoff frequency of the lock-in is less than f , only the $n = 1$ fourier component will remain after the low-pass filter. Therefore, the ideal lock-in output magnitude for a standard square wave signal with a duty cycle of $1/2$ would be $2A/\pi$. Since this type of scaling is predictable, it can also easily be accounted for.

SILIA also has the ability to estimate inner products in frequency space between the reference and the input signal. To compute these projections, **SILIA** has the option to skip the reference fitting step and mix the signal input with a scaled raw reference. The raw reference is scaled by a factor of 2 so the inner product between two purely sinusoidal signals that oscillate at the same frequency is the product of their amplitudes. This feature gives **SILIA** the ability to perform lock-in on non-sinusoidal references, but it cannot extract phase information in this case due to a lack of a well defined orthogonal quadrature reference input for an arbitrary periodic signal.

SILIA includes an error estimation feature which splits the input signal into portions for analysis of each part. After splitting and analyzing the data, **SILIA** evaluates the standard deviation of the results. The size and number of these portions are specified by input parameters. The trade-off is too little overlap and too many windows will result in an overestimation

of the error, while too few windows with very large window sizes and high overlap will underestimate the error.

3. Error Estimation Analysis

3.1. Methodology

In order to quantify the error estimation functionality of **SILIA**, a series of simulated data, consisting of multiple channels with multiple references, were acquired simultaneously. The simulated data were used to compare the output lock-in signal and phase with the input signals and references. The simulation had 100 input channels. Channel groups A, C and E in Fig. 2 contained only Gaussian noise while groups B and D had additional periodic signal modulated at 80 Hz and 120 Hz respectively. The signal was sinusoidal and had a power of $1/2$ while the noise had a variance of 1. We adopt the convention of signal to noise ratio (SNR) defined as the ratio of their powers. Therefore, the SNR in this simulation was $1 : 2$ for the channels that contained the input signals.

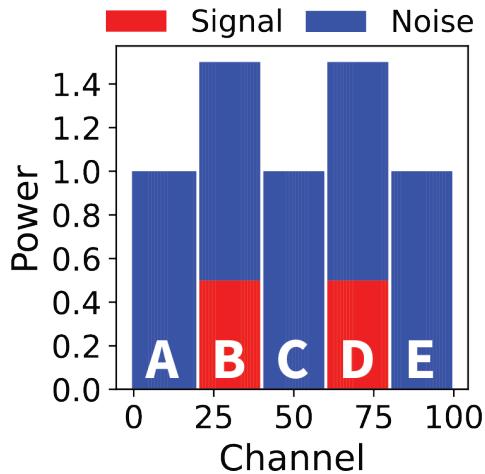


Figure 2: A graphical representation of the SNR for each channel of the input signal. 100 channels are split into groups of 20 channels each. Each group of 20 channels contain Gaussian noise with a variance, or power of one. The second and fourth groups contain additional synthetic signals that are sine waves with a power of $1/2$.

The data acquisition is for a simulated time of 5 seconds at a sampling rate of 5000 Hz and a frequency resolution of 0.2 Hz. The cutoff frequency is also set to 0.2 Hz. The references are 80 and 120 Hz square waves. Output values are generated by using **SILIA** to lock into the 80 Hz and 120 Hz references. Uncertainties are estimated by separately analyzing four evenly spaced time-intervals, each of which contains a third of the input data, and computing the standard deviation of the output lock-in magnitude and phase across all intervals.

3.2. Analysis

The input consisted of sinusoidal signals with an amplitude of 1 oscillating at 80 Hz and 120 Hz, each in one of two groups of channels, and higher-amplitude noise distributed over

6 **SILIA**: Software Implementation of a Multi-Channel, Multi-Frequency Lock-in Amplifier

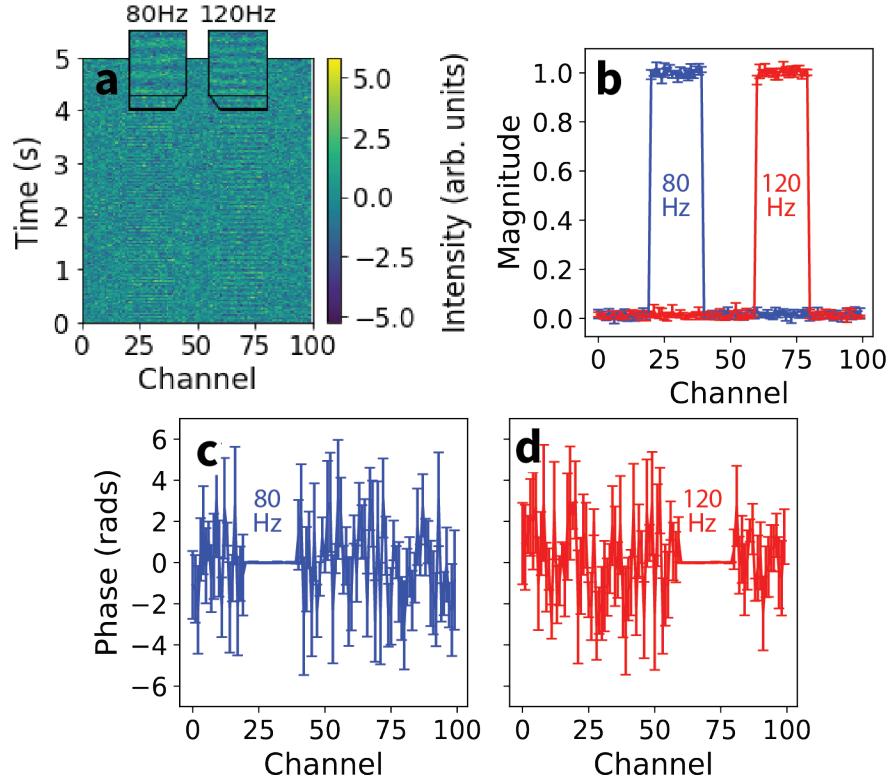


Figure 3: (a) Spectrogram of the simulated signal. The two zoomed insets show the noisy oscillating signals in channels 20-40 and 60-80, both signals oscillate at different frequencies. (b) The multi-frequency **SILIA** outputs. Both curves correctly identify the amplitudes of the 80 and 120 Hz signals at each channel. The output for each channel at the peaks are equal, within the estimated uncertainties, and the uncertainties are greater when there is no signal present. (c, d) Phase output when locking into both frequencies. We see that the phase error is very small in the presence of signal and large for all noisy channels.

Frequency (Hz)	Magnitude Error		Phase Error	
	Predicted	Actual	Predicted	Actual
80	0.016	0.008	0.011	0.006
120	0.017	0.012	0.011	0.008

Table 1: A comparison of the errors predicted by the error estimation feature of **SILIA** with the actual standard deviation of **SILIA** output over relevant channels.

all channels (Fig. 3a). **SILIA** suppresses noise and signals not oscillating at the reference frequency. When locking into the 80 or 120 Hz signal, we expect an output magnitude near 1 for channels 20-40 or 60-80 respectively, and near 0 everywhere else. When locking into the 80 and 120 Hz signals, we find mean peak values of 1.000 with standard deviations of 0.008 and 0.012 as well as mean outputs in channels without signal of 0.021 and 0.017 respectively, as expected (Fig. 3b). The error bars at the peak values for the 80 and 120 Hz results are, on average, 0.016 and 0.017 which are somewhat larger than the standard deviation of the output at peak signal (Table 1).

For the 80 and 120 Hz outputs in the channels having signal, we observe a phase of 0.004 and -0.001 respectively with standard deviations of 0.006 and 0.008 and mean predicted errors of 0.011 (Fig. 3c, d, Table 1). The phase of the results closely match an expected phase of 0 in the presence of signal and the error bars provide a reasonable estimate of the standard deviation in phase results. We also observe fluctuations in phase where signal is not present since **SILIA** can only extract noise in those channels (Fig. 3c, d).

4. Benchmarking

4.1. Methodology

For the initial benchmark, the amount of time is measured for the software to run for a variable number of input samples, channels and references by varying each of those three parameters individually while holding the other two constant. We use constant values of 4096 samples, 10 channels and 1 reference. This benchmarking allowed us to confirm the linear runtime of **SILIA** with respect to the data acquisition time, size of each sample and number of signals (Fig. 4).

We quantified the error of the output magnitude with respect to data acquisition time given a signal frequency of 100 Hz and a sampling rate of 2000 Hz. The input signal was a sinusoidal wave with an amplitude of 1 and additional Gaussian white noise with varying standard deviations. The results were averaged over 100 runs and fit to a function of the form, a/x^n where a and n were the fit parameters and x was the total number of samples collected (Fig. 5a, b). The fit was justified through additional simulations, where we verified that error was generally a function of the total number of samples and not of the sampling rate or data acquisition time individually.

When conducting experiments, noise tends to be time-correlated. To showcase how the lock-in method can take advantage of these time correlations for better noise suppression, we simulated a sinusoidal signal with an amplitude of 1 and variable frequency. Gaussian pink noise[23] was added to the signal to simulate time-correlated noise. The results were averaged over 100 runs and fit to a function of the form, a/f^n where a and n were the fit parameters and f is the frequency of the input signal (Fig. 5c, d).

When locking into non-sinusoidal periodic inputs **SILIA** output should match the primary Fourier component of the input signal (Section 2). **SILIA** was challenged to lock into 100 Hz square waves with peak to peak height of 1 and varying duty cycles. After generating the signals and running the lock-in, we proceeded to confirm that our results matched the primary Fourier component of the signal (Fig. 5e).

To determine the response curve of the low-pass filter, a simulated signal frequency of 100 ± 5 Hz was varied however the reference frequency was constant at 100 Hz. The input signal was a sinusoidal wave with an amplitude of 1 and had no additional noise added to it. The lock-in output magnitude was plotted with respect to the input signal frequency with effective cutoffs at 0.06Hz and 1.02Hz (Fig. 5f).

4.2. Analysis

4.2.1. Runtime Benchmarking

The total runtime of the software rises linearly with increasing input channels (Fig. 4a). We see a similar trend with the number of frequency references, where adding an additional reference frequency to the **SILIA** input will increase the runtime of the software proportionally (Fig. 4b). These trends are attributed to the software performing the analysis separately and in sequential order for each channel and frequency reference. We also observe a general linear increase in runtime with respect to the number of input samples but see nonlinear fluctuations which result in increased runtimes when the number of input samples is not close to a power of 2 (Fig. 4c). This is likely due to our use of the radix-2 FFT algorithm, which has a runtime that is optimized for input sample sizes that are a power of 2 [24, 25]. As expected, we see that **SILIA** is slowest when asked to fit the reference signal and interpolate the input, and is significantly faster when the user skips those steps. It is also apparent that using a fitted reference is more costly than applying interpolation to the input signal (Fig. 4). This is due to the additional requirement of computing phase when the reference is fitted.

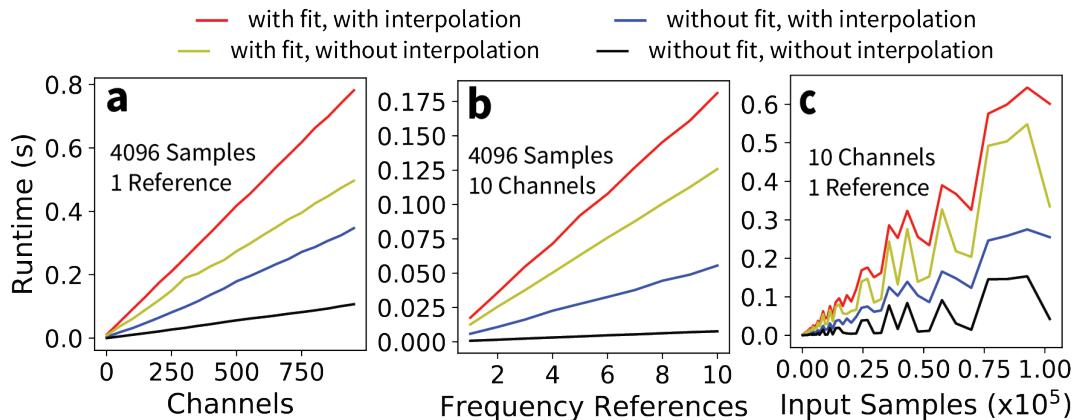


Figure 4: Dependence of the runtime of **SILIA** on the number of channels, input samples, and frequency references. (a, b) There is a linear dependence of runtime on the number of input channels and frequency references. (c) Shows a general linear relationship between the runtime and data acquisition time, but there is nonlinearity introduced by the slower runtime of the Fourier Transform when the number of input samples is not a power of 2.

It is important to note that these benchmark plots do not necessarily reflect the exact runtime of the software in all applications, since those runtimes are dependent on the hardware used to run **SILIA**, future optimizations in the code, and the scale of the user's run. The observed linear trends imply that the post-processing time will generally be proportional to the data acquisition time.

4.2.2. Error Benchmarking

When benchmarking the effects of experimental errors, we focused on three potential sources of uncertainties - Gaussian or oscillatory noise in the input signal, the response of each low-pass filter, and non-sinusoidal signal. A low-pass filter response with slow roll-off can introduce

unwanted artifacts from frequencies outside the cutoff range into the result of the lock-in amplification, and, as mentioned in Section 2, non-sinusoidal input can introduce bias into the result.

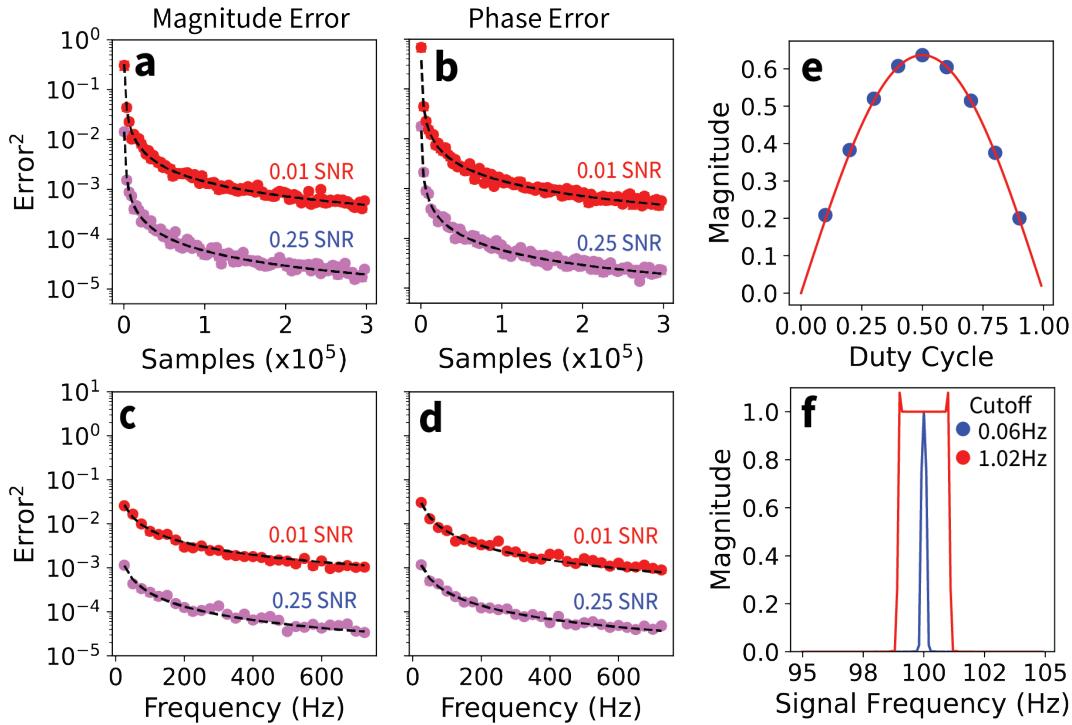


Figure 5: (a, b) Dependence of Mean Square Error of output with respect number of samples of input signal. The black dashed lines are fitted curves. There is a steep decrease in magnitude error (Fig. 5a) as well as phase error (Fig. 5b). (c, d) **SILIA** output magnitude (Fig. 5c) and phase (Fig. 5d) increases in accuracy with respect to the frequency of the true signal when the noise is time-correlated. The dashed lines are fitted curves. (e) Output as a function of the duty cycle of a square wave input signal. **SILIA** outputs are fit to the red curve, which is the first term in the Fourier series expansion of a square wave with varied duty cycles. We calculated the coefficient of determination ($R^2 = 1.0$) using the residuals of a fit to the curve $2\pi \sin(\pi d)$, where d is the duty cycle. (f) Low-pass response curve for a constant reference frequency of 100 Hz.

<u>Fit Parameters</u>	Magnitude Error		Phase Error	
	0.25 SNR	0.01 SNR	0.25 SNR	0.01 SNR
a	5.60	118	5.83	161
n	1.00	0.98	1.00	1.01

Table 2: Fit parameters for the plots in Figure 5a,b. Data was fit to curves of the form a/x^n where a and n are the fit parameters. x is number of samples.

Output error of lock-in amplifiers tends to decrease sharply with respect to data acquisition time and input SNR [2, 21, 26, 27]. We see a similar significant improvement in the **SILIA** output error with respect to the number of samples of the input signal (Fig. 5a, b), and can

Fit Parameters	Magnitude Error		Phase Error	
	<u>0.25 SNR</u>	<u>0.01 SNR</u>	<u>0.25 SNR</u>	<u>0.01 SNR</u>
<i>a</i>	0.025	0.55	0.025	0.61
<i>n</i>	0.96	0.95	0.99	0.98

Table 3: Fit parameters for the plots in Figure 5c,d. Data was fit to curves of the form a/f^n where a and n are the fit parameters. f is the frequency of the true signal.

conclude the necessity of longer data acquisition times and higher sampling rates for more accurate results, especially from data with low SNR. By fitting the error reduction to curves of the form a/x^n where x is variable and equal to the number of samples acquired, we find that the variance in the magnitude and phase of the lock-in result is approximately proportional to $1/x$ with proportionality constants that are a function of signal to noise ratio. (Table 2, Fig. 5a,b).

In addition, in the presence of more realistic time-correlated pink noise, output error decreases with respect to the frequency of the true signal (Fig. 5c,d). This result indicates that increasing the frequency at which an experiment is being done will likely decrease variance in **SILIA** output. Pink noise has a $1/f$ power spectrum, so we expect the variance in output to decrease at the same rate. That is consistent with what we observe in the fitted parameters (Table 3).

Furthermore, the shape of the input signal waveform affects the output in an easily quantifiable manner. We can easily observe this by locking into square waves with different duty cycles using a fitted reference, the output closely matches the primary Fourier component of the input signal, as expected (Fig. 5e). Furthermore, the significant decrease in the output magnitude when the input oscillates outside the cutoff frequency range (Fig. 5f) demonstrates the effectiveness of **SILIA** at filtering out periodic noise.

5. Applications of SILIA

5.1. Spectral Analysis

To test **SILIA** in an experimental setting, we measured light from a 532 nm diode-pumped solid state laser (ThorLabs CPS532) with high ambient background lighting, and measured the fluorescence spectrum emitted from a solution of Rhodamine 6G in ethanol excited at 532 nm.

5.1.1. Methodology

An Ocean Optics OCEAN-FX-XR1-ES spectrometer with a wavelength resolution of 0.42 nm on average and a standard deviation of 0.02 nm (0.42 ± 0.02 nm) was used to measure amplitude-modulated laser light, and the fluorescence spectrum emitted from Rhodamine 6G (95% purity) in ethanol ($\geq 99.5\%$ purity, Sigma-Aldrich) at different concentrations (0.013, 0.13 and 1.3 mM, respectively). The laser pulses were generated from a laser (ThorLabs CPS532) with a continuous-wave power of 4.5 mW. To modulate the amplitude of the signal and measure a frequency reference, a Thorlabs MC2000B chopper that modulated the excitation laser at 50 or 100 Hz and was connected to a National Instruments USB6001 data acquisition device. The experimental setup is depicted in Fig. 6.

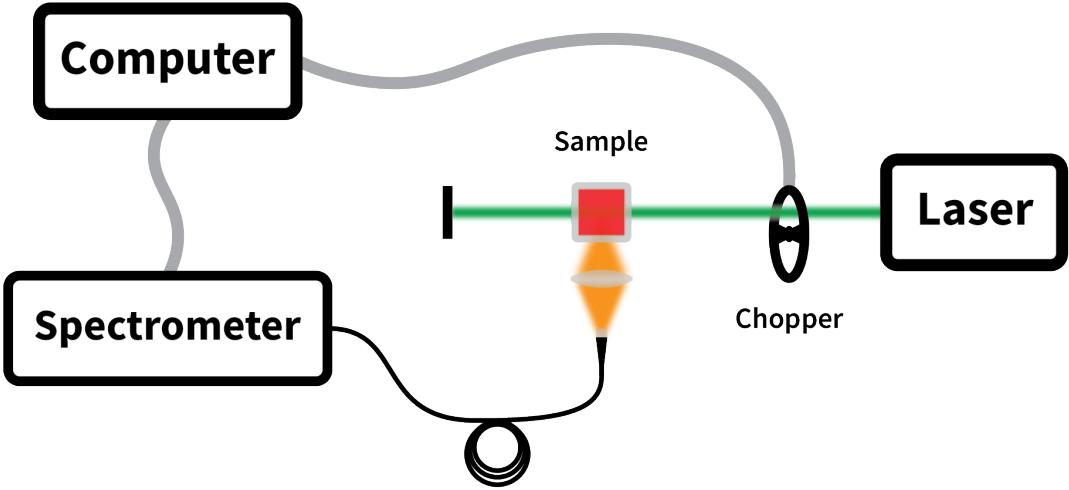


Figure 6: Experimental setup to measure the fluorescence spectrum of a Rhodamine 6G sample. A chopper wheel is used to modulate the laser which excites the Rhodamine while a lens is used to focus the resulting fluorescence signal into the spectrometer. The chopper controller outputs a reference waveform, and both the spectrometer data and reference are acquired (National Instruments USB-6001) and recorded to a personal computer. To measure the spectrum of the excitation laser the Rhodamine sample was replaced with a reflective diffuser.

During the data acquisition, the reference input was measured with a 2000 Hz sampling rate from the chopper, which modulated the laser amplitude at 100 Hz for the laser spectrum measurement. The sampling rate of the spectrometer was 230 ± 49 Hz, with a two second runtime per data acquisition cycle for three cycles.

Fluorescence spectrum measurements were performed for 3 different concentrations (0.013, 0.13 and 1.3 mM) of Rhodamine 6G in ethanol. The fluorescence signal was coupled into the spectrometer by a 250 mm convex lens. The laser was modulated at 50 Hz and the reference input was measured from the chopper with a 2000 Hz sampling rate. For this measurement the sampling rate of the spectrometer was 249 ± 58 Hz, with a three second runtime per data acquisition cycle for three cycles.

5.1.2. Results

Figure 7a depicts the spectrum of the modulated (100 ± 1 Hz) laser light and ambient background measured with the spectrometer. Over the three data acquisition cycles, the 100 ± 1 Hz reference signal was a square wave with a peak to peak height of 1 that was fitted to sinusoidal functions by **SILIA**. The output magnitudes fluctuated with a standard deviation of 5.7% (Fig. 7c) for all 3 cycles, possibly due to irregular sampling by the spectrometer data acquisition (Fig. 7b). Figure 7c shows the retrieved laser spectrum, and the ambient white light background effectively removed from the measured spectrum, by **SILIA**. The retrieved phase difference between the reference and the measured spectrum is small around the 532 nm peak in the spectrum and much higher fluctuations in the other regions of the measured

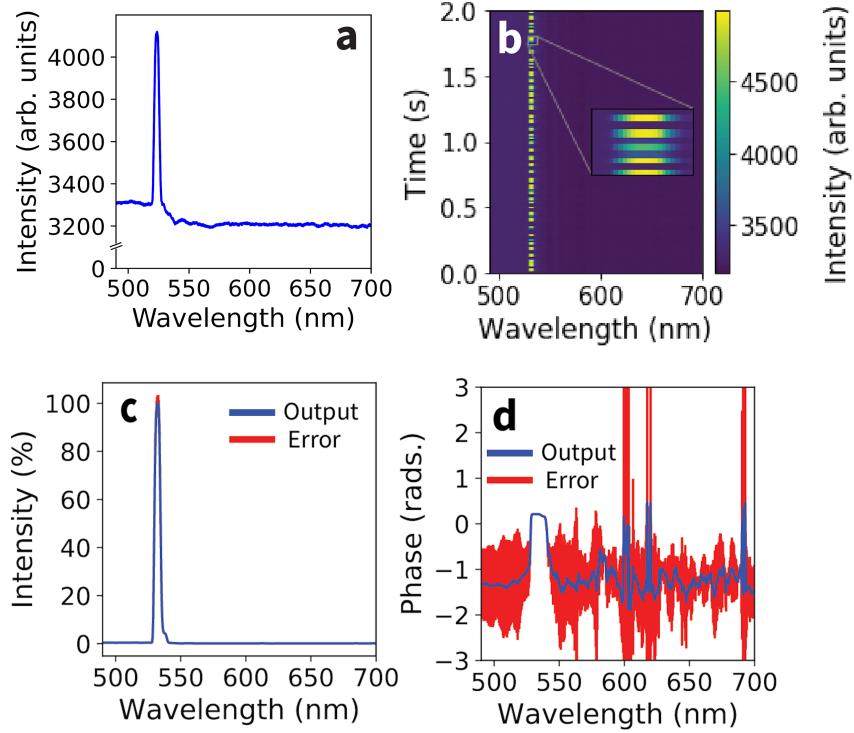


Figure 7: (a) The time-averaged signal of the spectrometer. The peak from the chopped laser signal can be seen at 532 nm. (b) A spectrogram of the chopped laser signal from a single data acquisition cycle of the spectrometer, showing the 100Hz oscillation of the signal input. Irregular sampling of light by the spectrometer leads to intensity variations of $\pm 33\%$ in laser peaks. (c) **SILIA** output magnitude averaged over three data acquisition cycles showcasing a clear laser signal. (d) **SILIA** output phase averaged over three data acquisition cycles demonstrating phase convergence in the presence of signal.

spectrum which are dominated by ambient white light background (Fig. 7d).

Typically, in fluorescence spectroscopy the use of photomultiplier tube is required to obtain sufficient signal from fluorescent samples. Here we apply **SILIA** to measure the fluorescence of solutions of Rhodamine 6G in ethanol for optical excitation at 532 nm without a photomultiplier, but with a much less sensitive spectrometer (Ocean Optics OCEAN-FX-XR1-ES). Since a photomultiplier tube and/or low bandpass filter and/or monochromator was not used, the observed spectrum (Fig. 8a) contains spectral information from scatter photons of the pulsed 532 nm laser light and fluorescent signal of Rhodamine 6G onto the spectrometer. This pollution renders the fluorescent signal indistinguishable from the background signal. Running **SILIA** allows the fluorescent signal to be observed (Fig. 8b). Here we fit the reference signal to sinusoidal functions but did not use interpolation. The output closely matches a Rhodamine 6G fluorescence spectrum from the PhotoChem database [28, 29], shown as the green curve, and we observe that higher 1.3 mM concentration of Rhodamine 6G emits fluorescence at longer wavelengths due to the well-known primary inner filter effect[30].

The difference between the fluorescence spectra for the solutions having lower concentrations (0.013 mM and 0.13 mM) is likely due to the secondary inner filter effect. The phase difference

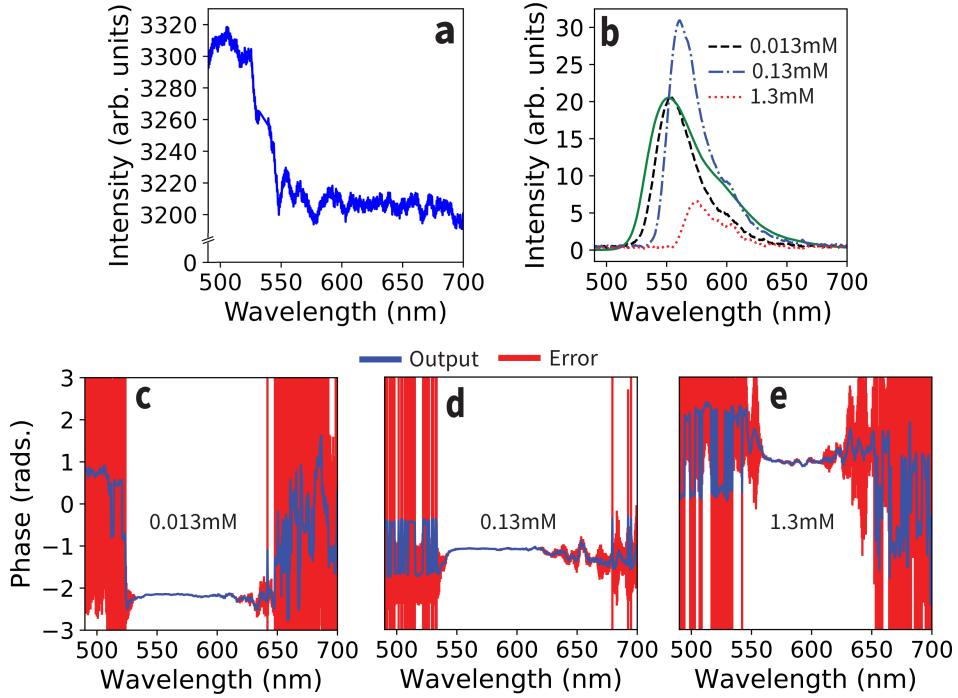


Figure 8: (a) Typical time-averaged signal of the spectrometer. The rhodamine spectrum is invisible in the spectrometer readings. (b) **SILIA** output for different concentrations of Rhodamine averaged over 3-5 data acquisition cycles. The peaks become more distorted with increasing Rhodamine concentration and have varying heights due to the primary and secondary inner-filter effects. The spectra are plotted against the green curve, a rescaled reference rhodamine 6G fluorescence measurement taken using a Spex FluoroMax with an excitation wavelength of 480 nm and spectral bandwidth of 4.25 nm [28, 29]. Our 0.013 mM results closely match the reference despite being taken with inferior equipment. Output peaks shift to higher wavelengths when rhodamine concentration is increased. (c,d,e) **SILIA** phase output for the various concentrations of rhodamine. We observe a consistent phase convergence across all three concentrations in the presence of signal.

between the signal and reference retrieved by **SILIA** is almost constant over the wavelength channels displaying significant fluorescence signal, due to the laser modulation by the chopper at the reference frequency, and very noisy for the channels having no fluorescence. As expected, we also see a consistent phase convergence in the presence of signal and note that the convergence is stronger at lower Rhodamine 6G concentrations (Fig. 8c,d,e). The stronger phase convergence at lower concentrations of Rhodamine 6G is consistent with our observation those spectra are in better agreement with the reference spectrum.

5.2. Imaging Simulation

5.2.1. Methodology

SILIA is designed to work with signal inputs with arbitrary dimensionality. To demonstrate its ability to analyze microscopy data, we used public domain images produced by fluorescence

microscopy of bovine pulmonary artery endothelial (BPAE) cells (Fig. 9a) and human lymph node cells stained with DAPI (Fig. 9b) to simulate a series of identical images with fluctuating noise in lieu of a fluorescence microscopy measurement. Here we demonstrate that lock-in amplification by **SILIA** can be used on microscopy data to retrieve simultaneous fluorescence signals from noise [10], assuming the excitation source is modulated.

To simulate a simultaneous fluorescence signal, we multiplied the images by square waves with a peak to peak height of 1 so they would periodically flash on and off. The BPAE cells were modulated with a frequency of 100 Hz while the human lymph node cells were modulated at 75 Hz. We then overlayed both signals and added a significant amount of Gaussian noise with a standard deviation of 75 (RGB value) to each pixel. We took 500 total samples of this noisy, combined signal at 10 samples per cycle. The images were in a RGB format and had a size of 512x512 pixels which resulted in an input signal of 786432 channels. Each RGB value in the image can range from 0 to 255, where the values of (0, 0, 0) represents a black color and (255, 255, 255) represents a white color.

5.2.2. Results

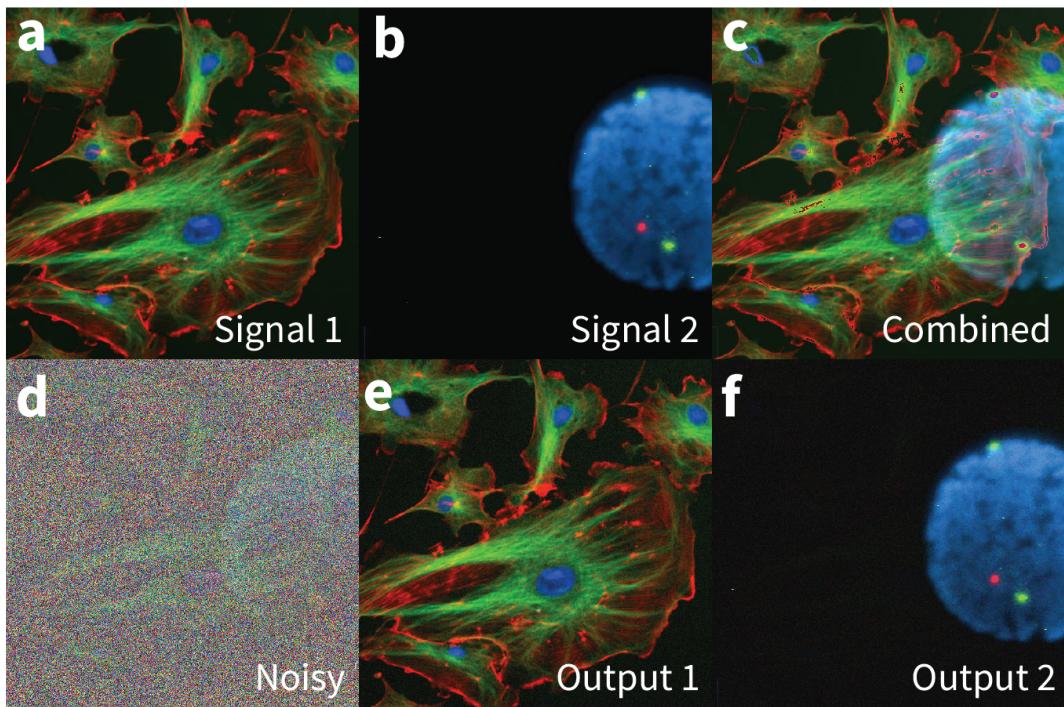


Figure 9: Simulated clean fluorescence signal of (a) a public domain image of BPAE cells with stained nuclei and marked microtubules, (b) a public domain image of a human lymphnode nucleus stained with DAPI, and (c) when both images are fluorescing. (d) Combined fluorescence signal with noise. (e,f) **SILIA** recovered the original fluorescence images despite significant noise. The result is scaled by $\pi/2$ to compensate for diminished output when locking into a square wave signal with a fitted, sinusoidal reference.

We seek to recover the original microscopy images (Fig. 9a,b) from the combined and noisy signal (Fig. 9c,d) by applying **SILIA**. Since the input signals are square waves with a 50% duty

cycle, the raw **SILIA** output was approximately $2/\pi$ times the original RGB values and was then scaled by $\pi/2$ to recover the original image (Fig. 9c). Since RGB values range from 0 to 255, the scaled output was capped at 255. The difference between the original and amplified images and found to be small, with an average value of 8.56 ± 6.00 (RGB value) for signal 1 (Fig. 9a) and 7.36 ± 5.85 (RGB Value) for signal 2 (Fig. 9b). **SILIA** took about 10 minutes to compute this output on a laptop computer with a 2.7 GHz Intel i7 CPU and 16.0 GB RAM.

6. Discussion

SILIA is an inexpensive and general alternative to multi-channel and multi-frequency lock-in amplifiers. Some advantages in the present implementation include convenient capabilities to estimate error in lock-in results, simplicity compared to other digital lock-in amplifiers, and the flexibility to extract periodic signals from data sources of arbitrary dimensionality. For example, the code is readily adaptable to a frequency sweep measurement [17] by using a frequency modulated reference signal. **SILIA** is also advantageous to using simpler computational error reduction methods such as background subtraction and averaging because it has the ability to extract phase information and can lock into multiple overlayed signals (Fig. 9).

In addition, since the the lock-in procedure is implemented as a post-processing step, measured data can be preprocessed prior to performing the lock-in operation, and **SILIA** can be run multiple times, e.g. with adjusted input parameters, on the same data. For example, by artificially scaling the time axis of the reference prior to lock-in, harmonics of the reference frequency can also be applied to recreate a Fourier series approximation to the exact signal waveform in each channel. **SILIA** could also be employed to extract periodic features from video data without a separate reference input by using a reference pixel [31].

Some limitations of using this implementation include the potentially significant runtime of the software when analyzing large datasets. The software currently has a linear runtime with respect to the number of channels and frequency references (Fig. 4a, b), which can be reduced through parallelization of **SILIA** with respect to each channel and frequency reference [32–34]. In addition, the Python package **Numba** could be used to speed up numerical calculations in the future. There are also superior techniques to the optional interpolation procedures for the reference input, which can be implemented to mitigate error from the FFT due to inconsistent sampling rates [34–37]. **SILIA** is also limited in the size of the dataset it can process since it stores all the input data in RAM. However, this can be circumvented in future implementations by including methods which take advantage of big data tools.

7. Conclusions

We have developed and demonstrated the software package **SILIA** that effectively suppresses background noise from periodic features in datasets with an arbitrary number of channels and frequency references. By adopting a paradigm where lock-in amplification is implemented as a post-processing step, we find a higher degree of flexibility than available in other methods while providing accurate results for modulated multi-channel measurements collected with common data acquisition approaches. We anticipate broad potential applications in spectroscopy, microscopy and imaging technologies. Editing **SILIA** is straightforward, and we encourage users to apply, improve and customize the code to suit their needs.

8. Acknowledgements

This material is based upon work performed at the Lawrence Berkeley National Laboratory and was supported by the U.S. Department of Energy, Office of Science, Division of Chemical Sciences of the Office of Basic Energy Sciences under Contract DE-AC02-05CH11231.

References

- [1] K. Kishore and S. A. Akbar. “Evolution of Lock-In Amplifier as Portable Sensor Interface Platform: A Review”. In: *IEEE Sensors Journal* 20.18 (Sept. 2020). Conference Name: IEEE Sensors Journal, pp. 10345–10354. ISSN: 1558-1748. DOI: [10.1109/JSEN.2020.2993309](https://doi.org/10.1109/JSEN.2020.2993309).
- [2] Andreas Mandelis. “Signal-to-noise ratio in lock-in amplifier synchronous detection: A generalized communications systems approach with applications to frequency, time, and hybrid (rate window) photothermal measurements”. In: *Review of Scientific Instruments* 65.11 (Nov. 1994). Publisher: American Institute of Physics, pp. 3309–3323. ISSN: 0034-6748. DOI: [10.1063/1.1144568](https://doi.org/10.1063/1.1144568). URL: <https://aip.scitation.org/doi/10.1063/1.1144568> (visited on 07/27/2020).
- [3] Xiaoyi Wang. “Sensitive digital lock-in amplifier using a personal computer”. In: *Review of Scientific Instruments* 61.7 (July 1990). Publisher: American Institute of Physics, pp. 1999–2001. ISSN: 0034-6748. DOI: [10.1063/1.1141413](https://doi.org/10.1063/1.1141413). URL: <https://aip.scitation.org/doi/10.1063/1.1141413> (visited on 07/28/2020).
- [4] G. A. Stimpson et al. “An open-source high-frequency lock-in amplifier”. In: *Review of Scientific Instruments* 90.9 (Sept. 2019). Publisher: American Institute of Physics, p. 094701. ISSN: 0034-6748. DOI: [10.1063/1.5083797](https://doi.org/10.1063/1.5083797). URL: <https://aip.scitation.org/doi/10.1063/1.5083797> (visited on 07/28/2020).
- [5] Giuseppe Costantino Giaconia et al. “Exploring FPGA-Based Lock-In Techniques for Brain Monitoring Applications”. en. In: *Electronics* 6.1 (Mar. 2017). Number: 1 Publisher: Multidisciplinary Digital Publishing Institute, p. 18. DOI: [10.3390/electronics6010018](https://doi.org/10.3390/electronics6010018). URL: <https://www.mdpi.com/2079-9292/6/1/18> (visited on 07/27/2020).
- [6] M. Carminati et al. “Note: Differential configurations for the mitigation of slow fluctuations limiting the resolution of digital lock-in amplifiers”. In: *Review of Scientific Instruments* 87.2 (Feb. 2016). Publisher: American Institute of Physics, p. 026102. ISSN: 0034-6748. DOI: [10.1063/1.4941721](https://doi.org/10.1063/1.4941721). URL: <https://aip.scitation.org/doi/10.1063/1.4941721> (visited on 07/27/2020).
- [7] Yan Liu et al. “Lock-in camera based heterodyne holography for ultrasound-modulated optical tomography inside dynamic scattering media”. In: *Applied Physics Letters* 108.23 (June 2016). Publisher: American Institute of Physics, p. 231106. ISSN: 0003-6951. DOI: [10.1063/1.4953630](https://doi.org/10.1063/1.4953630). URL: <https://aip.scitation.org/doi/abs/10.1063/1.4953630> (visited on 07/27/2020).

- [8] Pierre-Alain Probst and Alain Jaquier. "Multiple-channel digital lock-in amplifier with PPM resolution". In: *Review of Scientific Instruments* 65.3 (Mar. 1994). Publisher: American Institute of Physics, pp. 747–750. ISSN: 0034-6748. DOI: [10.1063/1.1145096](https://doi.org/10.1063/1.1145096). URL: <https://aip.scitation.org/doi/10.1063/1.1145096> (visited on 07/28/2020).
- [9] Sergi Foix, Guillem Alenyà, and Carme Torras. "Lock-in Time-of-Flight (ToF) Cameras: A Survey". In: *IEEE Sensors Journal* 11.9 (Sept. 2011). Conference Name: IEEE Sensors Journal, pp. 1917–1926. ISSN: 1558-1748. DOI: [10.1109/JSEN.2010.2101060](https://doi.org/10.1109/JSEN.2010.2101060).
- [10] G. Marriott et al. "Optical lock-in detection imaging microscopy for contrast-enhanced imaging in living cells". en. In: *Proceedings of the National Academy of Sciences* 105.46 (Nov. 2008), pp. 17789–17794. ISSN: 0027-8424, 1091-6490. DOI: [10.1073/pnas.0808882105](https://doi.org/10.1073/pnas.0808882105). URL: <http://www.pnas.org/cgi/doi/10.1073/pnas.0808882105> (visited on 08/30/2020).
- [11] S. Bourquin et al. "High-speed femtosecond pump probe spectroscopy with a smart pixel detector array". en. In: *Optics Letters* 28.17 (Sept. 2003), p. 1588. ISSN: 0146-9592, 1539-4794. DOI: [10.1364/OL.28.001588](https://doi.org/10.1364/OL.28.001588). URL: <https://www.osapublishing.org/abstract.cfm?URI=ol-28-17-1588> (visited on 07/27/2020).
- [12] Mirco Kolarczik et al. "Sideband pump-probe technique resolves nonlinear modulation response of PbS/CdS quantum dots on a silicon nitride waveguide". In: *APL Photonics* 3.1 (Dec. 2017). Publisher: American Institute of Physics, p. 016101. DOI: [10.1063/1.5005490](https://doi.org/10.1063/1.5005490). URL: <https://aip.scitation.org/doi/full/10.1063/1.5005490> (visited on 07/27/2020).
- [13] Leonid Gilburd et al. "Near-Field Infrared Pump–Probe Imaging of Surface Phonon Coupling in Boron Nitride Nanotubes". en. In: *The Journal of Physical Chemistry Letters* 7.2 (Jan. 2016), pp. 289–294. ISSN: 1948-7185. DOI: [10.1021/acs.jpclett.5b02438](https://doi.org/10.1021/acs.jpclett.5b02438). URL: <https://pubs.acs.org/doi/10.1021/acs.jpclett.5b02438> (visited on 07/27/2020).
- [14] Mizuho Fushitani. "Applications of pump-probe spectroscopy". en. In: *Annual Reports Section 'C' (Physical Chemistry)* 104.0 (June 2008). Publisher: The Royal Society of Chemistry, pp. 272–297. ISSN: 1460-4787. DOI: [10.1039/B703983M](https://doi.org/10.1039/B703983M). URL: <https://pubs.rsc.org/en/content/articlelanding/2008/pc/b703983m> (visited on 07/27/2020).
- [15] Martin C. Fischer et al. "Invited Review Article: Pump-probe microscopy". In: *Review of Scientific Instruments* 87.3 (Mar. 2016). Publisher: American Institute of Physics, p. 031101. ISSN: 0034-6748. DOI: [10.1063/1.4943211](https://doi.org/10.1063/1.4943211). URL: <https://aip.scitation.org/doi/10.1063/1.4943211> (visited on 07/27/2020).
- [16] P.-T Dong and J.-X Cheng. "Pump-probe microscopy: Theory, instrumentation, and applications". In: *Spectroscopy (Santa Monica)* 32 (Apr. 2017), pp. 24–36.
- [17] Maximiliano Osvaldo Sonnaión and Fabian Jose Bonetto. "Lock-in amplifier error prediction and correction in frequency sweep measurements". In: *Review of Scientific Instruments* 78.1 (Jan. 2007). Publisher: American Institute of Physics, p. 014701. ISSN: 0034-6748. DOI: [10.1063/1.2428269](https://doi.org/10.1063/1.2428269). URL: <https://aip.scitation.org/doi/10.1063/1.2428269> (visited on 07/28/2020).
- [18] Amrut Nadgir, Richard Thurston. *SILIA*. original-date: 2020-06-12T17:51:43Z. Feb. 2021. URL: <https://github.com/AMOS-experiment/SILIA> (visited on 02/11/2021).

- [19] John H. Scofield. “Frequency-domain description of a lock-in amplifier”. In: *American Journal of Physics* 62.2 (Feb. 1994). Publisher: American Association of Physics Teachers, pp. 129–133. ISSN: 0002-9505. DOI: [10.1119/1.17629](https://doi.org/10.1119/1.17629). URL: <https://aapt.scitation.org/doi/10.1119/1.17629> (visited on 07/28/2020).
- [20] Paul A. Temple. “An introduction to phase-sensitive amplifiers: An inexpensive student instrument”. In: *American Journal of Physics* 43.9 (Sept. 1975). Publisher: American Association of Physics Teachers, pp. 801–807. ISSN: 0002-9505. DOI: [10.1119/1.9690](https://doi.org/10.1119/1.9690). URL: <https://aapt.scitation.org/doi/10.1119/1.9690> (visited on 07/28/2020).
- [21] Sabyasachi Bhattacharyya et al. “Implementation of Digital Lock-in Amplifier”. en. In: *Journal of Physics: Conference Series* 759 (Oct. 2016). Publisher: IOP Publishing, p. 012096. ISSN: 1742-6596. DOI: [10.1088/1742-6596/759/1/012096](https://doi.org/10.1088/1742-6596/759/1/012096). URL: <https://doi.org/10.1088%2F1742-6596%2F759%2F1%2F012096> (visited on 07/27/2020).
- [22] Nasser M. Abbasi. *Fourier Series Coefficients of a Rectangular Pulse Signal*. Wolfram Demonstrations Project. Mar. 2011. URL: <http://demonstrations.wolfram.com/FourierSeriesCoefficientsOfARectangularPulseSignal/> (visited on 02/11/2021).
- [23] J. Timmer and M. Koenig. “On generating power law noise.” In: *Astronomy and Astrophysics* 300 (Aug. 1995), p. 707. ISSN: 0004-6361. URL: <http://adsabs.harvard.edu/abs/1995A&A...300..707T> (visited on 03/05/2021).
- [24] Mohammad Rafiq Muqri, Eric John Wilson, and Javad Shakib. “A Taste of Python – Discrete and Fast Fourier Transforms”. In: ISSN: 2153-5965. June 2015, pp. 26.123.1–26.123.11. ISBN: 978-0-692-50180-1. URL: <https://peer.asee.org/a-taste-of-python-discrete-and-fast-fourier-transforms> (visited on 07/27/2020).
- [25] Rassoul Amirfattahi. “Calculation of computational complexity for radix-2 p fast fourier transform algorithms for medical signals”. en. In: *Journal of Medical Signals & Sensors* 3.4 (Oct. 2013). Publisher: Medknow Publications, p. 217. ISSN: 2228-7477. DOI: [10.4103/2228-7477.128310](https://www.jmssjournal.net/article.asp?issn=2228-7477;year=2013;volume=3;issue=4;spage=217;epage=224;aulast=Amirfattahi;type=0). URL: <http://www.jmssjournal.net/article.asp?issn=2228-7477;year=2013;volume=3;issue=4;spage=217;epage=224;aulast=Amirfattahi;type=0> (visited on 07/27/2020).
- [26] D. A. Van Baak and George Herold. “Response of a lock-in amplifier to noise”. In: *American Journal of Physics* 82.8 (July 2014). Publisher: American Association of Physics Teachers, pp. 785–797. ISSN: 0002-9505. DOI: [10.1119/1.4873915](https://doi.org/10.1119/1.4873915). URL: <https://aapt.scitation.org/doi/10.1119/1.4873915> (visited on 07/28/2020).
- [27] K. Neelakantan, S. Dattagupta, and K. P. Rajappan. “Signal to noise enhancement of lock-in amplifiers”. en. In: *Review of Scientific Instruments* 51.2 (Feb. 1980), pp. 251–252. ISSN: 0034-6748, 1089-7623. DOI: [10.1063/1.1136167](https://doi.org/10.1063/1.1136167). URL: <http://aip.scitation.org/doi/10.1063/1.1136167> (visited on 07/28/2020).
- [28] Masahiko Taniguchi and Jonathan S. Lindsey. “Database of Absorption and Fluorescence Spectra of >300 Common Compounds for use in PhotochemCAD”. en. In: *Photochemistry and Photobiology* 94.2 (2018). _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/php.12860>, pp. 290–327. ISSN: 1751-1097. DOI: [10.1111/php.12860](https://doi.org/10.1111/php.12860). URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/php.12860> (visited on 09/13/2020).

- [29] Masahiko Taniguchi, Hai Du, and Jonathan S. Lindsey. “PhotochemCAD 3: Diverse Modules for Photophysical Calculations with Multiple Spectral Databases”. en. In: *Photochemistry and Photobiology* 94.2 (2018). _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/php.12862>, pp. 277–289. ISSN: 1751-1097. DOI: [10.1111/php.12862](https://doi.org/10.1111/php.12862). URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/php.12862> (visited on 09/13/2020).
- [30] Suraj Kumar Panigrahi and Ashok Kumar Mishra. “Inner filter effect in fluorescence spectroscopy: As a problem and as a solution”. en. In: *Journal of Photochemistry and Photobiology C: Photochemistry Reviews* 41 (Dec. 2019), p. 100318. ISSN: 1389-5567. DOI: [10.1016/j.jphotochemrev.2019.100318](https://doi.org/10.1016/j.jphotochemrev.2019.100318). URL: <https://www.sciencedirect.com/science/article/pii/S1389556719300875> (visited on 02/11/2021).
- [31] Davide Alinovi. “Video Processing for Remote Respiration Monitoring”. In: (2018). DOI: [10.5565/rev/elcvia.1124](https://doi.org/10.5565/rev/elcvia.1124).
- [32] Manreet Sohal and Rasmeet Kaur. “Automatic Parallelization: A Review”. en. In: 2016, p. 5.
- [33] Sudhakar Sah and Vinay Vaidya. “A Review of Parallelization Tools and Introduction to Easypar”. In: *International Journal of Computer Applications* 56 (Oct. 2012), pp. 30–34. DOI: [10.5120/8944-3108](https://doi.org/10.5120/8944-3108).
- [34] Alexander H. Barnett, Jeremy Magland, and Ludvig af Klinteberg. “A Parallel Nonuniform Fast Fourier Transform Library Based on an “Exponential of Semicircle” Kernel”. en. In: *SIAM Journal on Scientific Computing* 41.5 (Jan. 2019), pp. C479–C504. ISSN: 1064-8275, 1095-7197. DOI: [10.1137/18M120885X](https://doi.org/10.1137/18M120885X). URL: <https://pubs.siam.org/doi/10.1137/18M120885X> (visited on 08/06/2020).
- [35] Leslie Greengard and June-Yub Lee. “Accelerating the Nonuniform Fast Fourier Transform”. en. In: *SIAM Review* 46.3 (Jan. 2004), pp. 443–454. ISSN: 0036-1445, 1095-7200. DOI: [10.1137/S003614450343200X](https://doi.org/10.1137/S003614450343200X). URL: [http://pubs.siam.org/doi/10.1137/S003614450343200X](https://pubs.siam.org/doi/10.1137/S003614450343200X) (visited on 08/03/2020).
- [36] Lexing Ying. “Sparse Fourier Transform via Butterfly Algorithm”. In: *SIAM Journal on Scientific Computing* 31.3 (Feb. 2009), pp. 1678–1694. ISSN: 1064-8275. DOI: [10.1137/08071291X](https://doi.org/10.1137/08071291X). URL: <https://doi.org/10.1137/08071291X> (visited on 08/03/2020).
- [37] Emmanuel Candès, Laurent Demanet, and Lexing Ying. “A Fast Butterfly Algorithm for the Computation of Fourier Integral Operators”. In: *Multiscale Modeling and Simulation* 7.4 (June 2009). Number: 4 Publisher: Society for Industrial and Applied Mathematics, pp. 1727–1750. ISSN: 1540-3459. URL: <https://resolver.caltech.edu/CaltechAUTHORS:20091026-135743447> (visited on 08/03/2020).

Affiliation:

Daniel S. Slaughter
Chemical Sciences Division
Lawrence Berkeley National Laboratory
1 Cyclotron Rd, Berkeley CA 94720
E-mail: DSSlaughter@lbl.gov