

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/51367509>

Spectral Analysis of Calcium Oscillations

Article in Science's STKE · November 2004

DOI: 10.1126/stke.2582004pl115 · Source: PubMed

CITATIONS

72

READS

2,285

1 author:



[Per Uhlén](#)

Karolinska Institutet

176 PUBLICATIONS 5,079 CITATIONS

[SEE PROFILE](#)

Spectral Analysis of Calcium Oscillations

Per Uhlén

(Published 9 November 2004)

INTRODUCTION

MATERIALS

EQUIPMENT

INSTRUCTIONS

TROUBLESHOOTING

RELATED TECHNIQUES

NOTES AND REMARKS

Theory Behind Spectral Analysis Based on Fourier Transform

Explanation of the Code and Algorithms for Sample Analysis with MATLAB

REFERENCES AND NOTES

Departments of Pharmacology, Cell and Molecular Physiology, Yale University School of Medicine, New Haven, CT, USA or Department of Woman and Child Health, Karolinska Institutet, Stockholm, Sweden Department of Pharmacology, Yale University School of Medicine, 333 Cedar Street, New Haven, CT 06520-8066, USA. Telephone, +1-203-737-1158; fax, +1-203-737-2027; e-mail, per.uhlen@yale.edu. Alternatively, Department of Woman and Child Health, Karolinska Institutet, Astrid Lindgren Children's Hospital Q2:09, S-171 76 Stockholm, Sweden. Telephone, +46-8-51777339; fax, +46-8-51777328; e-mail, per.uhlen@kbh.ki.se

Abstract

Calcium (Ca^{2+}) oscillations are universal signals exploited by cells to regulate a vast number of cellular processes. Frequency and amplitude, the key features of oscillating waves, can yield information that can be decoded by intracellular processes. Analysis and quantification of Ca^{2+} oscillations are crucial for understanding the general concept of Ca^{2+} signaling. This protocol presents a method for performing spectral analysis of Ca^{2+} oscillations using MATLAB software.

Introduction

Calcium (Ca^{2+}) is an almost universal intracellular messenger that controls a diverse range of cellular processes, including gene transcription, muscle contraction, and cell proliferation (1). Cells create large Ca^{2+} concentration gradients ($\sim 10^4$ to 1) between the extracellular fluid, cytoplasm, and internal Ca^{2+} stores by means of Ca^{2+} pumps located in the plasma membrane and in the membranes of internal Ca^{2+} stores. These gradients provide ideal conditions for the use of Ca^{2+} as a “cellular currency” that supports the propagation of intracellular Ca^{2+} waves. The concerted actions of Ca^{2+} transporters located in the plasma membrane and in the membranes surrounding internal stores, including the endoplasmic and sarcoplasmic reticulum, the mitochondria, and the nucleus, generate Ca^{2+} oscillations. These oscillations can occur spontaneously, or they may be the result of a specific stimulatory event. Depending on the cell type, the period of Ca^{2+} oscillations ranges from fractions of seconds to tens of minutes.

As with a radio transmitter, information can be encoded in both the frequency and amplitude of the oscillations (2). Within the same biological system, different physiological agents give rise to Ca^{2+} oscillations of different frequencies (3, 4). Moreover, the frequency of the oscillations may also vary in response to the magnitude of stimulation (5, 6). The frequency of Ca^{2+} oscillations regulates such cellular phenomena as gene expression (7, 8), Ca^{2+} -calmodulin protein kinase II activation (9), secretion (10), and neuronal differentiation (11). However, it is not clear how these different responses are specified, nor is it clear exactly how frequency is used to encode information. Analytical methods are thus critical tools needed for further understanding of the specificity of Ca^{2+} oscillations and the signals they encode. The generation of Ca^{2+} oscillations is a consequence of the concerted action of several proteins. Experimental recordings of these signaling events can be difficult to interpret and quantify without sophisticated analysis methods. Nonetheless, important information can be hidden within the data, and analytical methods can extract crucial information. Additionally, such analysis is essential to determine statistical significance when interpreting data and to compare the responses to different treatments. Spectral analysis, in which data collected in the time domain are transformed to information in the frequency domain, is a powerful analytical method with which to detect and quantify Ca^{2+} oscillations (3, 4).

This Protocol describes how to use code written for MATLAB to perform spectral analysis based on Fourier transform on previously acquired Ca^{2+} -signaling data. MATLAB is mathematical analysis software that includes a high-level technical computing language and an environment for algorithm development, data visualization, data analysis, and numerical computation. Because MATLAB can read files of different formats, Ca^{2+} -imaging data obtained through various experimental methods can be analyzed using MATLAB. Variables and predefined MATLAB functions used in this Protocol are described in Tables 1 and 2, respectively. The MATLAB code is usually saved in a text file that can be executed by the program. Changes to the program are made by editing the code in the text file. This Protocol presents essential code for performing spectral analysis on Ca^{2+} oscillations.

Data suitable for this type of analysis can be obtained through several experimental paradigms, such as ion-specific fluorescence microscopy. However, all data showing an oscillatory behavior can be analyzed using this approach. This Protocol deals with Ca^{2+} recordings obtained using fluorescent indicators. There are several Ca^{2+} -sensitive dyes available on the market today, such as Fura-2, Fluo-4, Rhod-2, and Indo-1. Both nonratiometric and ratiometric indicators can be used. Analyzing ratiometric data, however, can reveal additional information about the amplitude and the absolute concentration of Ca^{2+} . Fluorescent probes can be used in both confocal and CCD-equipped microscopes. All acquisition software has options to either save or export recorded data as ASCII text files.

Variables	Description
areaP	Area under the power spectrum
calciumOsc	Single-cell recording of Ca^{2+} oscillation
degree	Polynomial degree
dP	Derivate of the power spectrum
dt	Time step
f	Frequency array
f1, f2	Closest valleys to the peak of interest
fs	Sampling frequency
G	Fourier transform of Ca^{2+} recordings
hanning	Hanning window function
N	Number of bins
P	Power spectrum
peakIndex	Peak indices
peaks	Matrix with information about peaks
sortedValleys	Sorted valleys
t	Time array
tmp	Dummy variable
trend	Polynomial coefficients
valleyIndex	Valley indices

Table 1. Variables and their descriptions.

MATLAB functions	Description
. *	Element-by-element multiplication
'	Matrix transpose
&	Logical operator AND
abs	Absolute value and complex magnitude
conj	Complex conjugate
end	Terminate for-statement or indicate last index
fft	One-dimensional FFT
fftshift	Shift zero-frequency component of FFT to center of spectrum
find	Find indices and values of nonzero elements
for	Repeat statements a specific number of times
gradient	Numerical gradient
length	Length of an array
max	Maximum elements of an array
min	Minimum elements of an array
plot	Linear two-dimensional plot
polyfit	Polynomial curve fitting
polyval	Polynomial evaluation
sort	Sort elements in ascending order
sortrows	Sort rows in ascending order
trapz	Trapezoidal numerical integration

Table 2. MATLAB functions and their descriptions.

Materials

Single-cell, time-lapse recordings of Ca^{2+} oscillations saved as a text file.

Note: These can be obtained with both fluorescent microscopy and confocal microscopy by using either ratiometric indicators or non-ratiometric indicators.

Equipment

PC, Macintosh, or Unix/Linux computer with MATLAB Release 12, or later, installed

Note: Current versions of MATLAB will run on multiple operating systems, and the code described here is compatible with them all.

MATLAB Release 12 or later (<http://www.mathworks.com/products/matlab/>)

SpectralAnalysis code (<http://stke.sciencemag.org/cgi/content/full/sigtrans;2004/258/pl15/DC1>)

Instructions

The following instructions show some essential commands needed to perform spectral analysis on recordings of single-cell Ca^{2+} oscillations using MATLAB. Ca^{2+} recordings carried out with a fluorescence microscope typically contain some portion of data that should be excluded from the analysis. Thus, an important step in spectral analysis of Ca^{2+} oscillations is to select the region of interest before analyzing the data. This can be achieved in MATLAB or by preparing the data with another program, for example, Excel. Spectral analysis is based on rather extensive mathematical theories, which are shown in the Notes and Remarks section of this Protocol. The steps are outlined below. For these detailed instructions, a specific example will be used, and the variable names will be based on this example. Variables and MATLAB functions are described in Tables 1 and 2, respectively.

1. Plot and inspect the recorded data (signal intensity versus time) using either the acquisition program or another program, such as Excel. Select single-cell traces of interest for the spectral analysis. Save the data in a tab-delimited text file with the time values in the first column and the single trace Ca^{2+} recordings in the following columns.

Note: If there are apparent discontinuities in the series, such as sudden intensity changes of the signal, separate the data of interest into homogeneous segments.

2. Import the text file into the program using predefined MATLAB functions and assign the data to a matrix, here called “A.”

Note: A subroutine for importing text files into MATLAB is included in the downloadable program. More information about importing files can be found in the MATLAB Help file.

3. Assign the time column to an array “t” and the first calcium recording to analyze to “calciumOsc.”

```
t = A(:,1);
calciumOsc = A(:,2);
```

Note: In MATLAB, matrix manipulations are done using the colon (:). $A(:,j)$ is the j^{th} column of A, and $A(i,:)$ is the i^{th} row of A. The semicolon (;) used after each statement suppresses onscreen printing. Variable names such as “calciumOsc” can be assigned according to user preference.

4. Determine and subtract the trend component of the experimental recording by applying the following commands. These commands also center the signal on the time axes.

```
degree = 2;
trend = polyfit(t,calciumOsc,degree);
calciumOsc = calciumOsc - polyval(trend,t);
```

Note: Use the MATLAB functions “polyfit” and “polyval” to calculate and subtract the trend component. The trend component is a slowly changing component of the signal that can arise from various artifacts. The polynomial degree depends on the shape of the original data and should typically not exceed three (see Notes and Remarks for more information and an explanation of the underlying mathematics). Here, a polynomial function of degree two is used.

5. Plot the function to verify that the polynomial degree was correctly chosen to remove the trend component and that the signal is centered on the time axis.

```
plot(t,calciumOsc)
```

6. A built-in function in MATLAB computes the fast Fourier transform (FFT) of the signal. The frequency resolution can be increased by padding the end of the time series with zeros. This is done by the following commands.

```
N = 2048;
G = fft(calciumOsc,N);
```

Note: For example, a 500-point signal could be padded with zeros to make it 2048 points long. Although N can be any positive integer, a power of two is usually chosen, for example, 128, 256, 512, 1024, 2048, ... (see Notes and Remarks for more information).

7. Obtain and normalize the power spectral density (PSD) by executing the following commands.

```
P = G.*conj(G)/N;
P = P(1:N/2 + 1);
P(2:N/2) = 2*P(2:N/2);
```

8. Now, the PSD data “P” needs to be mapped onto a frequency array that shows the frequencies in the signal. In MATLAB, this is achieved by the following lines.

```
dt = 2;
fs = 1/dt;
f = fs*(0:N/2)/N;
plot(f,P)
```

Note: The time difference between each sample is “dt,” here 2 s.

In Fig. 1, two different sinusoidal functions are plotted and analyzed using the approach described above. Figure 2 shows a spectral analysis of actual Ca^{2+} recordings. Further methods and theories concerning spectral analysis can be found in the Notes and Remarks section of this Protocol. Algorithms sorting the most dominant frequencies from high to low are described. Also, a method that filters the signal to reduce spectral leakage is presented.

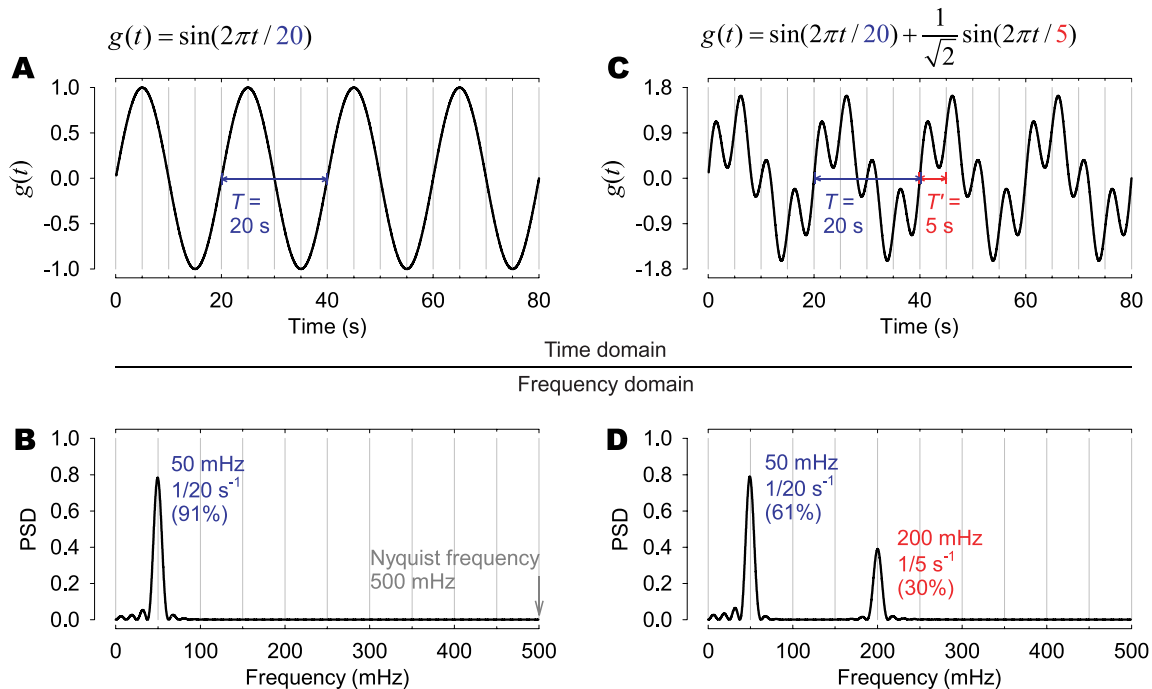


Fig. 1. (A) A sinusoidal function $g(t)$ plotted in the time domain and (B) its corresponding PSD in the frequency domain. The frequency of $g(t)$ is 50 mHz, which is equal to a periodicity (T) of 20 s. One strong peak at 50 mHz is observed in the power spectrum. The relative power (in parentheses) of this peak is 91%. A function $g(t)$ composed of two sinusoidal terms with different periodicity and amplitude is plotted (C) in the time domain and (D) in the frequency domain. The frequencies of $g(t)$ are 50 mHz and 200 mHz, which is equal to the periodicities 20 s and 5 s. In the power spectrum, two strong peaks at 50 mHz and 200 mHz are observed. The frequency with higher amplitude, which is more dominant, will be observed with a higher PSD peak. Both the relative powers and the height of the peaks are proportional to the squared amplitudes of the original sine waves.

Troubleshooting

The algorithm presented in this Protocol can be tested using sine waves of known frequencies, by applying $g(t) = A_1 \sin(2\pi t/T) + A_2 \sin(2\pi t/T') + \dots$, where $f = 1/T$, $f' = 1/T'$. Each different frequency should be observed as a separate peak in the power spectrum (Fig. 1). Execution problems in MATLAB usually occur when sizes and dimensions of arrays and matrices do not match. Arrays and matrices of different sizes or dimensions cannot be processed with each other. Therefore, check sizes of arrays and matrixes if the program does not execute. MATLAB is equipped with several functions to manipulate arrays and matrixes.

Related Techniques

There are several mathematical approaches for performing spectral analysis on time series (12–16). Most commonly, Fourier transform is used to change the unit time to frequency. The frequency axis can be plotted as a function of different magnitudes, such as energy, power, or power spectral density. Each of these presentations will provide different views of the data and should be selected to suit the specific study. Commercial programs that perform spectral analysis are available. However, most of these programs require some programming skills to acquire adequate spectral information.

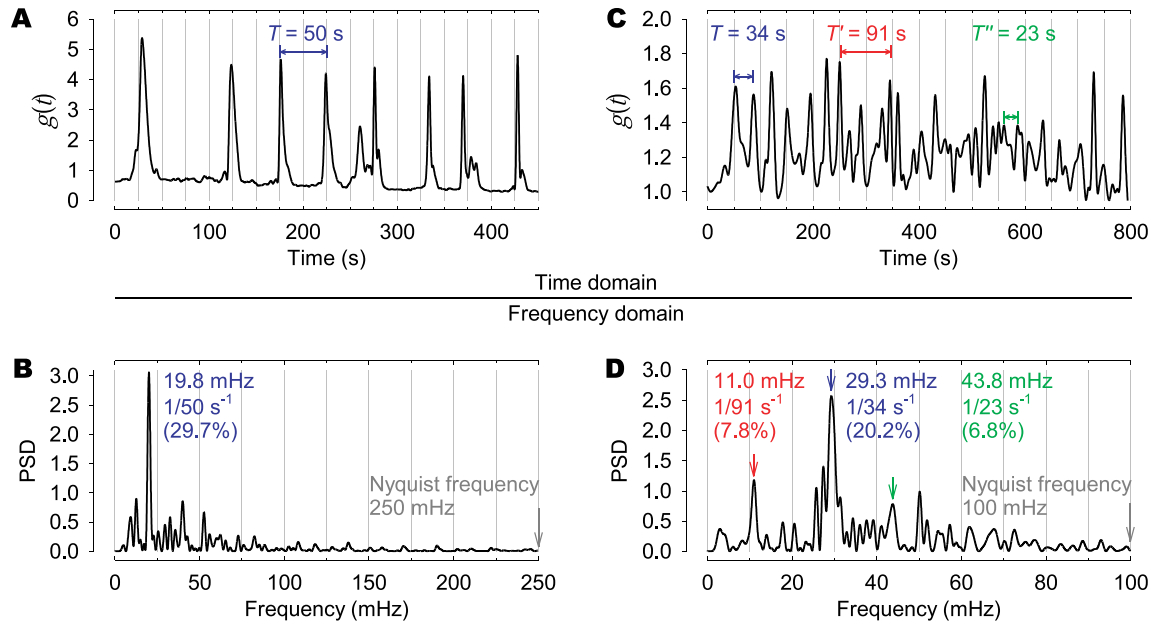


Fig. 2. (A) A Ca^{2+} recording showing one oscillating cell with highly regular periodicity in the time domain, and **(B)** its corresponding PSD in the frequency domain (18). The periodicity (T) of the Ca^{2+} oscillation is 50 s, which is equal to a frequency of 20 mHz. This value can be observed as a prominent peak at 19.8 mHz in the frequency domain. A second single-cell recording with a more complex Ca^{2+} periodicity pattern is plotted **(C)** in the time domain and **(D)** in the frequency domain (18). Numerous periodicities comprise the signal: $T = 34$ s, $T' = 91$ s, $T'' = 23$ s, and so on. Arrows indicate the three peaks with highest relative power (in parentheses). Several peaks with approximately 7% in relative power are observed in the spectrum. The most dominant peak in the spectrum at 29.3 mHz has a relative power of 20.2%.

Notes and Remarks

The MATLAB code described in this Protocol presents the basic idea for spectral analysis and can be further optimized. Time series are assumed evenly distributed in this Protocol. However, theories describing spectral analysis on unevenly distributed time series have also been described (17). Evenly distributed time series are measured at uniform time steps, whereas uneven time series have nonuniform time steps. In this survey of spectral analysis, the correlation between the amplitude of the original data and the power magnitude are not discussed because Ca^{2+} recordings are commonly uncalibrated with values corresponding to measurements of the fluorophore intensity and not the absolute Ca^{2+} concentration of the signal. Calibrated data from fluorophores, such as Fura-2, may be used to derive additional information about the amplitude and the corresponding Ca^{2+} concentration of the oscillatory signal. However, when estimating the amplitude of Ca^{2+} oscillations from the PSD, one must assume that the shape of the Ca^{2+} response is sinusoidal, which is usually not true for biological systems.

Theory Behind Spectral Analysis Based on Fourier Transform

Spectral analysis is generally based on the Fourier transform, which was developed by the French mathematician and physicist Jean Baptiste Joseph Fourier in the beginning of the 19th century. With this mathematical approach, periodic signals (that is to say, signals that occur with a particular frequency) can be converted from the time domain to the frequency domain. In other words, the Fourier transform swaps the dimension of time with the dimension of frequency. Thus, information acquired with time-lapse recordings can be depicted from a different point of view that yields more information about the regulatory behavior of the system. A power spectrum shows peaks at different frequencies that correspond to measured values that recur repeatedly within the time-lapse recording.

In the next several paragraphs, the mathematical theory underlying spectral analysis is briefly described, followed by a scheme for applying this theory to data obtained from experimental recordings of changes in the fluorescence of intracellular Ca^{2+} indicators. Symbols used in this section can be found in Table 3. A more extensive description of the theory can be found in several textbooks addressing this topic (12–16).

Symbols	Description
Δf	Frequency difference between points in the frequency domain
Δt	Time difference between each recording
f_k	Discrete frequency values
f_{max}	Nyquist frequency—highest frequency to be examined
f_{min}	Lowest frequency to be examined
f_s	Sampling frequency
$G(f_k)$	Fourier transform of $g(t_n)$
$g(t_n)$	Actual Ca^{2+} response
k	Frequency domain index
$m(t_n)$	Trend component
N	Number of Ca^{2+} recordings
n	Time domain index
$P(f_k)$	PSD
$P_{rel}(f)$	Relative power
$r(t_n)$	Recorded samples
t_n	Discrete time values
$w_{Hanning}(t_n)$	Hanning window function

Table 3. Mathematical symbols and abbreviations.

First, assume that a time-lapse recording is made of some number (N) of single-cell Ca^{2+} measurements that are evenly spaced with the time difference (Δt) between each measurement. Then, consider the Ca^{2+} recording as a representation of the following process:

$$r(t_n) = m(t_n) + g(t_n) \quad n = 1, 2, \dots, N \quad t = (n-1)\Delta t \quad (\text{Eq. 1})$$

where $m(t_n)$ is a slowly changing function known as a “trend component,” $g(t_n)$ is the actual Ca^{2+} response being calculated, and $r(t_n)$ is the data recorded by the acquisition system. The slow trend component can arise from several artifactual processes that influence the recording, such as bleaching or leakage of the Ca^{2+} fluorophore, and must be eliminated from each single-cell time-lapse recording, to optimize the outcome of the spectral analysis. One approach to mathematically estimating the trend is to apply a Gauss’ least-squares approximation to approximate the single-cell Ca^{2+} recording to a polynomial function. The polynomial degree depends on the shape of the original data and should typically not exceed three. Otherwise, the polynomial function by itself shows an oscillatory behavior and could hence perturb the spectral analysis if subtracted from the original signal. In this example, we fit a polynomial function of degree two,

$$m(t_n) = a_0 + a_1 t_n + a_2 t_n^2 \quad (\text{Eq. 2})$$

to the data by choosing the parameters a_0 , a_1 , and a_2 to minimize $\sum_n [r(t) - m(t)]^2$. When $m(t_n)$ is derived, the trend function can be subtracted from the recorded data $r(t_n)$ to get $g(t_n)$. This subtraction also centers the data on the time axis; that is, the time axis goes through the recorded signal, with one part on the negative “y axis” and the other part on the positive “y axis.” Centering the data are essential for the analysis.

As mentioned above, spectral analysis is commonly based on the Fourier transform. The Fourier transform represents the contribution of every oscillation signal within the data in a holistic way. The discrete time version, which operates on time domain signals that are sampled at specific time-intervals, is shown as

$$G(f_k) = \sum_{n=1}^N g(t_n) e^{-i2\pi kn/N} \quad k = 1 - \frac{N}{2}, \dots, \frac{N}{2} \quad (\text{Eq. 3})$$

where $g(t_n)$ represents the original signal in the time domain, $G(f_k)$ represents the Fourier transform in the frequency domain, and $e^{-i2\pi kn/N}$ is a function containing both a real and imaginary part. In the time domain, $g(t_n)$ consists of N points running from 0 to $N-1$. In the frequency domain, the discrete Fourier transform produces two signals, one real part and one imaginary part; in other words, Equation 3 has one real and one imaginary solution. Each of these frequency domain signals are N points long, and run from 0 to N

$$f_k = \frac{k}{N\Delta t} = k\Delta f \quad k = 1 - \frac{N}{2}, \dots, \frac{N}{2} \quad (\text{Eq. 4})$$

where Δf is the frequency difference between each point in the frequency domain. The time difference (Δt) between every sample also gives us the sampling frequency (f_s), which, together with the number of samples (N), determines the amount of information that can be derived from the experimental recording. The following equations explain their relationships.

$$f_s = \frac{1}{\Delta t} \quad f_{\max} \leq \frac{f_s}{2} = \frac{1}{2\Delta t} \quad \text{m} \quad f_{\min} \geq \frac{1}{(N-1)\Delta t} \quad f_{\min} \leq f \leq f_{\max} \quad (\text{Eq. 5})$$

Thus, the sampling frequency f_s must be greater than or equal to twice the highest frequency (f_{\max}) encountered. This limitation is referred to as the Nyquist sampling theorem and it states that the minimum sampling interval of a sine wave is two samples per period. If a time series is collected with a sampling interval of 1 s, the minimum resolvable period is 2 s, which corresponds to the frequency 0.5 Hz. Any attempt to sample at a lower rate results in a phenomenon known as aliasing (14, 15). This is usually not a problem when acquiring Ca^{2+} data, because of today's fast recording systems. The lowest frequency (f_{\min}) examined is the inverse of the entire time span of the series. These two statements give the bounding limits for the frequency analysis. Higher frequencies are determined by increasing the time resolution; that is, faster sampling rates and lower frequencies are determined by longer time series.

To identify and quantify oscillatory patterns within the Ca^{2+} recordings, a power spectrum should be defined. There are several ways to calculate power spectra, depending on the particular analysis of interest. The power of a signal, however, is a measure of the squared amplitude of the signal, and the PSD [$P(f_k)$] is defined as

$$P(f_k) = \frac{1}{N} |G(f_k)|^2 \quad (\text{Eq. 6})$$

and describes how much signal is present per unit of bandwidth. A peak with power in excess of the background distribution in the power spectrum indicates the existence of a periodic component in the Ca^{2+} recording. When estimating the amplitude of a periodic signal from the PSD spectrum, one must assume that the shape of a signal pulse is sinusoidal, which is usually not true, because of variability due to real physical and biological processes. The actual relationship between the PSD and the variability due to real physical and biological processes occurring in the time domain is given by Parseval's theorem,

$$\frac{1}{N} \sum_{k=-N/2}^{N/2-1} |G(f_k)|^2 = \sum_{n=1}^N |g(t_n)|^2 \quad (\text{Eq. 7})$$

Parseval's theorem states that the integral of the PSD over the whole frequency range is equal to the power of the variability component of the same process in the time domain. In fact, that is why the quantity in Equation 6 is called PSD. Plotting the Fourier power density allows us to investigate the different frequencies and their corresponding amplitudes, which arise due to biological and physical processes in the cell, and their relationships within the recorded signal. The relative power can be used to quantitatively determine the power of one frequency relative to that of other frequencies. This quantity is calculated by dividing the power from one frequency by the power of all frequencies, in other words, the ratio of the area under one peak with the area under the entire curve. The calculation follows as

$$P_{rel}(f) = \frac{\int_{f_1}^{f_2} P(f_k) df_k}{\int_{f_0}^{f_N} P(f_k) df_k} \quad (\text{Eq. 8})$$

where f_1 and f_2 are the boundaries of the peak of interest, and f_0 and f_N maps the entire frequency domain so that $f_0 = 0$ and $f_N = f_{\max}$. Relative power can be used to determine the most dominant frequencies within a time series. In Figure 1, two different sinusoidal functions are plotted and analyzed using this approach. Figures 1A and 1B show a single sinusoidal function and the corresponding power spectrum. In the frequency domain, one dominant peak that reflects the period of the function is observed. However, some spectral leakage into neighboring frequencies can be seen. The standard signal processing technique used to reduce this leakage is to apply filtering and windowing to the original data before performing the Fourier transform (see below) (12-16). Figures 1C and 1D show the same function as in Fig. 1A, but with an additional sinusoidal wave with different period and amplitude superimposed. These two frequencies can be difficult to resolve solely from the data by visual inspection (Fig. 1C). However, after transformation to the frequency domain, two peaks reflecting the different periods of the two sinusoidal functions can be observed clearly (Fig. 1D).

Explanation of the Code and Algorithms for Sample Analysis with MATLAB

Ca^{2+} measurements are usually acquired using Ca^{2+} -sensitive fluorophores, such as Fluo-4. All fluorescent molecules undergo some level of bleaching and leaking during an experiment. This fact will cause a decreasing trend in the fluorescent light intensity over time. For longer time-lapse recordings, these effects can have a major influence on the recording. Other factors, such as focus shift, may also cause a trend in the recording. Moreover, biological mechanisms unrelated to the particular study can give a trend increase or decrease. In this section, the mathematics behind the code is described for a concrete example. Variables and MATLAB functions are described in Tables 1 and 2, respectively.

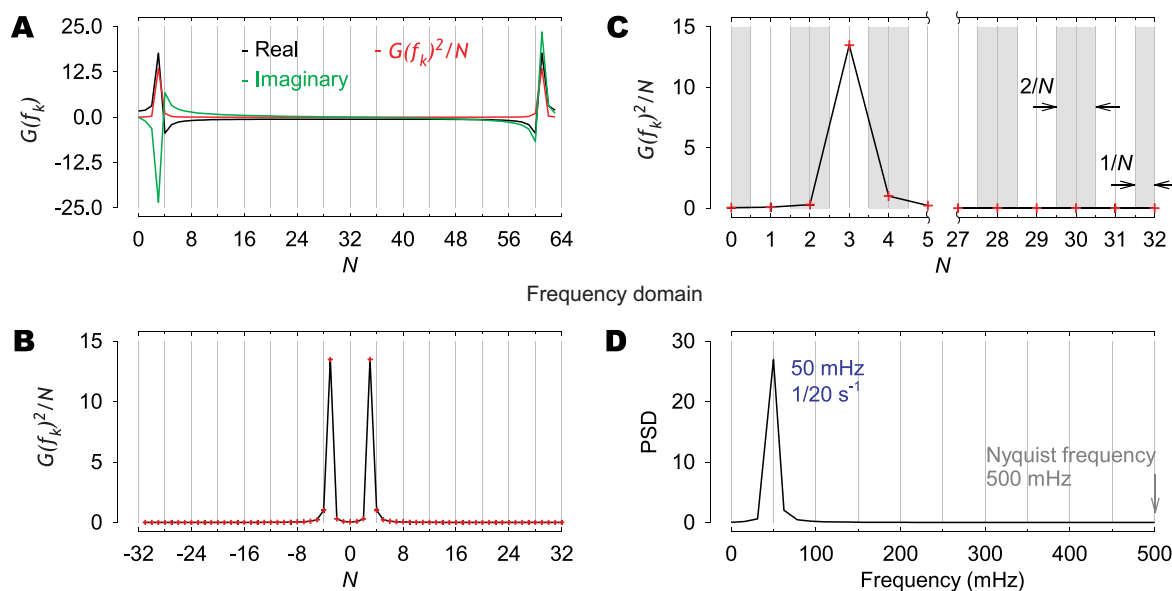


Fig. 3. (A) The real (black trace) and imaginary (green trace) part of the Fourier transform $G(f_k)$ derived from the sinusoidal function $g(t)$ in Fig. 1A. Notice that the Fourier transform produces both positive and negative values. The function $g(t)$ has a periodicity of 20 s and is sampled every second for 60 s; that is, $N = 60$. To get an N that is in power of two, the discrete Fourier transform is padded with four zeros, which gives $N = 64$. The power of the signal is the square of the Fourier transform (red trace). This result is normalized by dividing with N . (B) Visualizing the Fourier transform in Fig. 1A (red trace) with the zero-frequency component in the middle of the spectrum. This is achieved using the build-in MATLAB function “fftshift.” Notice that the spectrum has a mirror image on the negative side, separated by the red line. (C) In the frequency domain, each sample has a bandwidth of width $2/N$, expressed as a fraction of the total bandwidth. The first and last samples, however, have a bandwidth of only one-half this wide, $1/N$. (D) The final PSD plotted in the frequency domain. One strong peak at 50 mHz, which is equal to a periodicity (T) of 20 s, is observed in the power spectrum. The Nyquist frequency of a sampling frequency of 1 Hz is 0.5 Hz = 500 mHz.

Once the experimental time series that is to be analyzed is selected and assigned to an array, “calciumOsc,” then a trend correction and centering is performed. The trend component can be estimated using a Gauss’ least squares approximation:

```
degree = 2;
trend = polyfit(t,calciumOsc,degree);
calciumOsc = calciumOsc - polyval(trend,t);
```

Here, “t” is an array with the corresponding time values to “calciumOsc,” and “trend” is an array with coefficients for a polynomial function, described in Equation 2. The variable “degree” determines the polynomial degree. By subtracting the trend component from the original recording, the data series of interest to analyze is derived.

After correcting the trend and centering the signal, we can transfer the data from the time domain to the frequency domain by applying the discrete Fourier transform, which is computed with a FFT algorithm. From an N point time domain signal, the FFT calculates an N point frequency signal that spans between zero and the Nyquist frequency, given by the statements shown in Equation 5. Thus, the length of the time signal limits the frequency resolution. The frequency resolution can be increased by extending the length of the time series with zeros. This process, referred to as zero padding, adds more points to the frequency signal, which produces a smoother curve. A 256-point signal could be padded with zeros to make it 2048 points long. Taking a 2048-point discrete Fourier transform produces a frequency spectrum with 1025 samples ($N/2 + 1$). From the mathematics, the discrete Fourier transform gives two identical frequency spectra. The added zeros do not change the shape of the spectrum; they only provide more samples in the frequency domain. It is possible to use a discrete Fourier transform on data sets as large as the computer processor can handle. However, although N can be any positive integer, a power of two is usually chosen, for example, 64, 128, 256, 512, 1024, 2048, ... There are two reasons for this choice. First, digital data storage uses binary addressing, making powers of two a natural signal length. Second, the most efficient algorithm for computing the FFT usually operates with N that is a power of two. This is easily achieved in MATLAB by setting the following.

```
N = 64;
G = fft(calciumOsc,N);
```

Here, a 64-point discrete Fourier transform is calculated from the original time series. In Fig. 3A, both the real and imaginary part derived by the Fourier transform of a sinusoidal function is shown. Normally, N is chosen higher than the number of points in the original time series, in order to accommodate the extra zeros. If N is lower than the number of points in the original time series, the series will be truncated. The value of N should be at least as long as the start value of N . To obtain the power spectrum, the Fourier transform is squared and normalized by dividing by N , according to Equation 6.

```
P = G.*conj(G)/N;
```

This performance derives a spectrum of N points, where only the first $N/2$ values are valid. The second half is a mirror image of the first half and therefore redundant. This phenomenon is easier to observe by converting the zero frequency to the middle of the axes, as illustrated in Fig. 3B. The redundant part, however, holds the identical power information, and must be accounted for. Therefore, the power spectrum is multiplied by a factor of two to compensate for this fact. An exception to this is the first and last sample.

```
P = P(1:N/2 + 1);
P(2:N/2) = 2*P(2:N/2);
```

The difference between the first and last points and all of the other points occurs because the frequency domain is defined as a spectral density. Spectral density describes how much signal (amplitude) is present per unit of bandwidth. The bandwidth of the power spectrum can be defined as shown in Fig. 3C. Fig. 3B also reveals that there are only one zero value and one $N/2 = 32$ value, whereas there are two of all the other values, one positive and one negative value.

After this calculation, the PSD can be plotted. First, the PSD data needs to be mapped onto a frequency array to be displayed in a graph. The time difference “dt” between each sample gives the sampling frequency “fs,” which determines the frequency array “f.”

```
fs = 1/dt;
f = fs*(0:N/2)/N;
plot(f,P)
```

This power spectrum shows all frequencies within the original time series. If the signal contains a periodicity, the most dominant frequencies in the Ca^{2+} oscillation will be visible. In Fig. 3D, the final power spectrum of a well-defined sine wave is plotted using this algorithm. The periodicity of the sine wave is easily observed. In reality, recorded physical and biological data does not consist of sine waves, and their power spectra can show several peaks. These peaks can be sorted in relation to power to determine the most dominant frequencies within the time series. If the frequency resolution is low, we only need to sort the power values from high to low. For higher frequency resolutions, we can take the derivative of the power spectrum and find the extreme points. In MATLAB, this approach can be implemented as,

```
dP = gradient(P);
peakIndex = find(dP(1:end-1)>0 & dP(2:end)<0);
peaks = [f(peakIndex) P(peakIndex)];
peaks = sortrows(peaks,2);
```

Here, “dP” is an array with slopes calculated for each point in the power spectrum. This algorithm creates a matrix “peaks” with the power of the peaks ordered from low to high. From this matrix, information can be selected for further statistical analysis. The relative power can also be used to determine the most dominant frequencies. This quantity tells us how much power one peak holds in relation to the total power of the spectrum.

```
valleyIndex = [1; find(dP(1:end-1)<0 & dP(2:end)>0)];
areaP = trapz(f,P);
for x = 1:length(peaks)
    [tmp,sortedValleys] = sort(abs(f(valleyIndex)-peaks(x,1)));
    f1 = min(valleyIndex(sortedValleys(1:2)));
    f2 = max(valleyIndex(sortedValleys(1:2)));
    peaks(x,3) = 100*trapz(f(f1:f2),P(f1:f2))/areaP;
end
peaks = sortrows(peaks,3);
```

Here, the boundaries for each peak are assigned to the array “valleyIndex.” The matrix “peaks” is now extended with one more column describing the relative power for each peak in the power spectrum. Depending on the specific study, the matrix can be sorted with regard to power magnitude or relative power.

This algorithm has been used to quantitatively determine the discrepancy of Ca^{2+} oscillations induced by specific agonists (3, 4). Regulatory patterns were verified from large numbers of single-cell Ca^{2+} recordings by applying spectral analysis. Fig. 2 shows actual Ca^{2+} recordings with their corresponding power spectrums. A rather easily analyzed Ca^{2+} recording is plotted in Fig. 2A. From this single-cell Ca^{2+} trace, the oscillatory frequency can be determined using visual inspection. This interpretation assumption is confirmed by the prominent peak in the frequency domain (Fig. 2B). Figure 2C shows a more complex Ca^{2+} recording where visual inspection is complicated and several periodicities appear to be present. Moreover, it is difficult to determine the frequencies with the greatest impact on the Ca^{2+} signal. In the frequency domain, several peaks are clear (Fig. 2D). Calculating the relative power of these peaks reveals that the most dominant frequency is 29.3 mHz, with a relative power of slightly more than 20%. The ten most dominant frequencies and their corresponding relative powers are shown in Table 4.

Frequency (mHz)	Relative Power (%)
29.3	20.2
10.9	7.8
43.6	6.8
27.3	5.4
50.2	4.7
25.7	3.9
31.4	3.6
62.2	3.1
54.4	2.8
37.5	2.4

Table 4. The ten most dominant frequencies and their relative powers.

Spectral leakage is a phenomenon that occurs when calculating the power spectrum from sampled data over a finite interval using discrete Fourier transform. In Figs. 1B, 1D, 2B, and 2D, several peaks surrounding the prominent peaks are observed. This smearing of the power spectrum arises mainly because the transform is computed during a finite-width window of time, N . The discrete Fourier transform assumes that $f(t + N\Delta t) = f(t)$; that is, the start value and the end value are identical. Typically, recorded signals execute many oscillations during the sampled interval, but not an integral number of periods. The effective discontinuity introduced by the finite width of the sampling window introduces spurious high-frequency components into the power spectrum. This effect can be significantly attenuated by multiplying the recorded data with a window function that ramps smoothly up from zero at the beginning of the sampling interval, and smoothly back down to zero by the end of the interval (12–16). Several different windows are available, most of them named after their original developers. One of the most common used is the Hanning window function, which is defined as

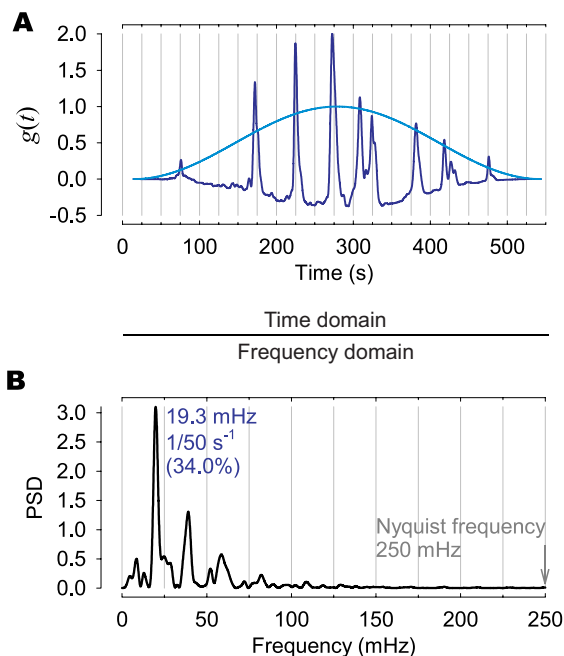
$$w_{\text{Hanning}}(t_n) = 0.5 - 0.5 \cos \frac{2\pi t_n}{t_N} \quad 0 \leq t_n \leq t_N \quad (\text{Eq. 9})$$

This function describes an array with the same number of elements (N) as the time-lapse recording in Equation 1. The signal is filtered by multiplying the recorded calcium signal “calciumOsc” with the Hanning window (“hanning”) before performing the Fourier transform.

```
hanning = 0.5 - 0.5*cos(2*pi*t/t(end));
calciumOsc = calciumOsc.*hanning;
N = 2048;
G = fft(calciumOsc,N);
```

In Fig. 4A, the Hanning filter together with the filtered calcium trace are plotted. This mathematical approach eliminates the discontinuity in the signal when the start point does not coincide with the end point. The power spectrum of the Hanning filtered calcium signal is seen in Fig. 4B. This power spectrum can be compared with the power spectrum of the unfiltered calcium signal, seen in Fig. 2B. The spectrum derived from the filtered signal has fewer peaks and thus less spectral leakage. Also, the relative power of the most dominant peak at ~20 mHz has increased from 29.7% to 34.0% after using the Hanning filter.

Fig. 4. (A) A Ca^{2+} recording showing one oscillating cell (dark blue), multiplied with a Hanning filter (light blue), and **(B)** its corresponding PSD in the frequency domain. The same transform without the filtering can be seen in Fig. 2B. Notice that the spectrum is cleaner with fewer peaks after using the filter; that is, the spectral leakage is reduced. The relative power of the most dominant peak has increased from 29.7% to 34.0%.



References and Notes

1. M. J. Berridge, M. D. Bootman, H. L. Roderick, Calcium signalling: Dynamics, homeostasis and remodelling. *Nat. Rev. Mol. Cell Biol.* **4**, 517–529 (2003).
2. M. J. Berridge, The AM and FM of calcium signalling. *Nature* **386**, 759–760 (1997).
3. P. Uhlen, A. Laestadius, T. Jahnukainen, T. Soderblom, F. Backhed, G. Celsi, H. Brismar, S. Normark, A. Aperia, A. Richter-Dahlfors, Alpha-haemolysin of uropathogenic *E. coli* induces Ca^{2+} oscillations in renal epithelial cells. *Nature* **405**, 694–697 (2000).
4. O. Aizman, P. Uhlen, M. Lal, H. Brismar, A. Aperia, Ouabain, a steroid hormone that signals with slow calcium oscillations. *Proc. Natl. Acad. Sci. U.S.A.* **98**, 13420–13424 (2001).
5. C. C. Petersen, O. H. Petersen, M. J. Berridge, The role of endoplasmic reticulum calcium pumps during cytosolic calcium spiking in pancreatic acinar cells. *J. Biol. Chem.* **268**, 22262–22264 (1993).
6. C. J. Nieman, D. A. Eisner, Effects of caffeine, tetracaine, and ryanodine on calcium-dependent oscillations in sheep cardiac Purkinje fibers. *J. Gen. Physiol.* **86**, 877–889 (1985).
7. R. E. Dolmetsch, K. Xu, R. S. Lewis, Calcium oscillations increase the efficiency and specificity of gene expression. *Nature* **392**, 933–936 (1998).
8. W. Li, J. Llopis, M. Whitney, G. Zlokarnik, R. Y. Tsien, Cell-permeant caged InsP3 ester shows that Ca^{2+} spike frequency can optimize gene expression. *Nature* **392**, 936–941 (1998).
9. P. De Koninck, H. Schulman, Sensitivity of CaM kinase II to the frequency of Ca^{2+} oscillations. *Science* **279**, 227–230 (1998).
10. R. W. Holl, M. O. Thorner, G. L. Mandell, J. A. Sullivan, Y. N. Sinha, D. A. Leong, Spontaneous oscillations of intracellular calcium and growth hormone secretion. *J. Biol. Chem.* **263**, 9682–9685 (1988).
11. X. Gu, N. C. Spitzer, Distinct aspects of neuronal differentiation encoded by frequency of spontaneous Ca^{2+} transients. *Nature* **375**, 784–787 (1995).
12. G. M. Jenkins, D. G. Watts, *Spectral Analysis and its Applications* (Holden-Day series in time series analysis, Holden-Day, San Francisco, 1968).
13. P. J. Brockwell, R. A. Davis, *Time Series: Theory and Methods* (Springer series in statistics, Springer-Verlag, New York; Berlin, ed. 2, 1991).
14. P. Stoica, R. L. Moses, *Introduction to Spectral Analysis* (Prentice Hall, Upper Saddle River, NJ, 1997).
15. D. B. Percival, A. T. Walden, *Spectral Analysis for Physical Applications: Multitaper and Conventional Univariate Techniques* (Cambridge Univ. Press, Cambridge, 1993).
16. S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing* (California Technical Pub, San Diego, CA, ed. 1, 1997).
17. W. H. Press, *Numerical Recipes in C: The Art of Scientific Computing* (Cambridge University Press, Cambridge, ed. 2, 1992).
18. P. Uhlen, unpublished data.
19. I am grateful to B. E. Ehrlich and M. Lal for critically reading the manuscript and to W. Grenawitzke and S. Malmersjö for helpful comments. This work was supported by a grant from the Swedish Research Council and a grant from the NIH (GM63496) to BEE.

Citation: P. Uhlen, Spectral analysis of calcium oscillations. *Sci. STKE* **2004**, pl15 (2004).