A novel mathematical method for disclosing oscillations in gene transcription: a comparative study

Athanasios C. Antoulas1,2,3,*, Bokai Zhu3, Qiang Zhang1, Brian York3,4, Bert W. O’Malley3,4, and Clifford C. Dacso1,3,4.

1 Department of Electrical and Computer Engineering, Rice University, Houston, USA
2 Max Planck Institute for the Dynamics of Complex Technical Systems, Magdeburg, Germany
3 Department of Molecular and Cellular Biology, Baylor College of Medicine, Houston, USA
4 Dan L. Duncan Cancer Center, Baylor College of Medicine, Houston, Texas, USA

* corresponding author: aca@rice.edu

Abstract

Circadian rhythmicity, the 24-hour cycle responsive to light and dark, is contributed to by periodic oscillations in gene transcription. This phenomenon has broad ramifications in physiologic function. Recent work has disclosed more cycles in gene transcription and to the uncovering of these we apply a novel signal processing methodology known as the pencil method. Methods: In order to assess periodicity of gene expression over time, we analyzed a database derived from livers of mice entrained to a 12 hour light/12 hour dark cycle. We also analyzed artificially generated signals to identify differences between the pencil decomposition and other similar methods. Results: The pencil decomposition revealed hitherto unsuspected oscillations in gene transcription with 12 periodicity. The pencil method was robust in detecting the 24 hour circadian cycle that was known to exist as well as confirming the existence of shorter period oscillations. A key consequence of this approach is that orthogonality of the different oscillatory components can be demonstrated, This indicates a biological independence of these oscillations, which has been subsequently confirmed empirically by knocking out the gene responsible for the 24 hour clock. Conclusion: system identification techniques can be applied to biological systems and can uncover important characteristics that may elude visual inspection of the data. Significance: The pencil method provides new insights on the essence of gene expression and discloses a wide variety of oscillations in addition to the well-studied circadian pattern. This insight opens the door to the study of novel mechanisms by which oscillatory gene expression signals exert their regulatory effect on cells to influence human diseases.

Keywords: circadian rhythm, system identification, pencil decomposition, gene oscillations, twelve-hour rhythm.
Introduction

Transcription can be considered as a signal transduced from DNA sequences influencing the function of a cell. Here we propose the analysis of transcription using various signal processing approaches. This provides additional insight into cell regulation and expression.

Gene transcription is the process by which the genetic code residing in DNA is transferred to RNA in the nucleus as a predecessor to make proteins. The latter process is called translation and occurs in the cytoplasm as depicted in the figure below. Circadian rhythm, the 24 hour cycle of gene transcription, is a manifestation of an auto transcription-translation loop.

Fig 1. The “Central Dogma” of genetics: DNA makes RNA makes protein [1] Chapter 2.3, [11].

In addition to the circadian oscillation driven by light and dark, other so-called ultradian rhythms have clear biologic import. Blood pressure, some circulating hormones, and physiological functions all appear to have 12 hour periodicity. Accordingly, we sought to uncover novel 12 hour oscillations in gene expression. In many cases, the 12 hour gene oscillation is superimposed on the 24 hour cycle thus hiding it in conventional analysis. Additionally, experiments designed to elucidate the 24 hour circadian often do not have the granularity required to reveal an interval of less than 24 hours.

To reveal periodicities in gene expression shorter than the 24 hour circadian cycle, we applied digital signal processing methodology to this biologic phenomenon. Although this approach is, to our knowledge, less commonly used in the biological field, it is justified because the transcription of DNA to RNA is indeed a signal, packed with information for making the enormous repertoire of proteins.

To extract the fundamental oscillations (amplitude and period) present in the data, we utilized publicly available time-series microarray datasets on circadian gene expression of mice liver (under constant darkness) [5] and analyzed over 18,000 genes spanning a variety of cellular process ranging from core clock control, metabolism, cell cycle to the unfolded protein responses (UPR) – a measure of cell stress. In addition one set of measurements of RER (respiratory exchange ratio) from wild-type mice (generated by us) was also performed. We constructed linear, discrete-time, time-invariant models, of low order, driven by initial conditions, which approximately fit the data and thus reveal the fundamental oscillations present in each data set. In addition to the 24 hour (circadian) cycle known to be present, other fundamental oscillations have been revealed using our approach.

Methods

We searched for 12 hour oscillations in several biological systems. We chose these systems because they represent not only gene transcription but also phenotype, that is the way in which these biological systems are expressed in the whole organism. The reasoning was that if the 12 hour oscillation in transcription was biologically significant, it would be represented in some measurable metabolic function.

Initially, we analyzed a set of transcription data [5] that was collected in mouse liver obtained from animals in constant darkness after being entrained in a 12 hour light/12 hour dark environment. Mice were sacrificed at 1 hour intervals for 48 hours thus providing enough data points to analyze the signal. The dataset thus obtained contains RNA values for all coding genes. The RNA data were generated using the microarray
method prevalent at the time the experiment was performed. In addition RER
(respiratory exchange ratio) measurements in mice were also considered.

The novelty in our analysis consists in using the so-called matrix-pencil method [1].
This is a data-driven system identification method. It constructs dynamical systems
based on time-series data and finds the dominant oscillations present in the ultradian
rhythms. Our purpose here is to compare this method with other established strategies
for spectral estimation, including both parametric spectrum estimation methods like
MUSIC (MUltiple Signal Classification) and ESPRIT (Estimation of Signal Parameters
via Rotational Invariance Techniques) Prony’s (Least squares) as well as classical
nonparametric models like wavelet transforms and statistical methods like RAIN. These
methods are finally compared with each other using both artificial and measured data.

Matrix pencil method

The data. We consider finite records of data resulting as described above. Generically
they are denoted by \( y_i, i = 1, \ldots, N \).

Basic model: sum of exponentials. We seek to approximate the data by means of
linear combinations of exponentials plus noise. Thus we seek \( r \) pairs of complex
numbers \( \alpha_i, \beta_i, i = 1, 2, \ldots, r \), such that

\[
y(t) = y^*(t) + n(t), \quad \text{where} \quad y^*(t) = \sum_{i=1}^{r} \alpha_i e^{\beta_i t},
\]

is the noiseless part of the signal and \( n(t) \) is the noise. The requirement is: \( y(m) \approx y_m, m = 1, 2, \ldots, N \). Existing approaches to address this problem are MUSIC, ESPRIT,
Prony’s (least squares) method, wavelet transform and statistical methods described
later.

Second model: descriptor representation. The equivalent descriptor model uses
an associated internal variable \( x(t) \in \mathbb{R}^k \) of the system. The resulting equations are:

\[
Ex(t + 1) = Ax(t), \quad y(t) = Cx(t) + n(t), \quad x \in \mathbb{R}^k, \quad t = 0, 1, 2, \ldots \tag{2}
\]

with initial condition \( x(0) = x_0 \in \mathbb{R}^k \), where \( E, A \in \mathbb{R}^{k \times k}, C \in \mathbb{R}^{1 \times k} \).

Third model: AR (Auto Regressive) representation. The above model can also
be expressed as an AR model driven by an initial condition. As above we let
\( y(t) = y^*(t) + n(t) \), (where \( y^*(t) \) is the noiseless term and \( n(t) \) the noise). It follows
that (2) can be rewritten as:

\[
y^*(n + k) + \gamma_{k-1}y^*(n + k - 1) + \cdots + \gamma_1y^*(n + 1) + \gamma_0y^*(n) = 0, \tag{3}
\]

with initial conditions \( y^*(\ell), \ell = 0, 1, \ldots, k - 1 \).

Goal. As already stated our goal is, using these models and reduced versions thereof, to
discover the fundamental oscillations inherent in the gene data.

Processing of the data. The data \( y_1, y_2, \ldots, y_N \), is used to form the Hankel
matrix:

\[
\mathcal{H} = \begin{bmatrix}
y_1 & y_2 & y_3 & \cdots & y_{k-1} & y_k & y_{k+1} \\
y_2 & y_3 & y_4 & \cdots & y_k & y_{k+1} & y_{k+2} \\
y_3 & y_4 & y_5 & \cdots & y_{k+1} & y_{k+2} & y_{k+3} \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
y_{k-1} & y_{k-2} & y_{k-3} & \cdots & y_{2k-3} & y_{2k-2} & y_{2k-1} \\
y_k & y_{k+1} & y_{k+2} & \cdots & y_{2k-2} & y_{2k-1} & y_{2k}
\end{bmatrix} \in \mathbb{R}^{k \times (k+1)},
\]
where for simplicity it is assumed that $N = 2k$. Then using MATLAB notation for simplicity, we define the quadruple $(E, A, B, C)$:

$$E = \mathcal{H}(1 : k, 1 : k), \quad A = \mathcal{H}(1 : k, 2 : (k + 1)), \quad B = \mathcal{H}(1 : k, 1), \quad C = \mathcal{H}(1, 1 : k). \quad (4)$$

This quadruple constitutes the raw model of the data. This model is linear, time-invariant and discrete-time with a non-zero initial condition:

$$Ex(n + 1) = Ax(n), \quad y(n) = Cx(n), \quad Ex(0) = B, \quad n = 0, 1, 2, \ldots. \quad (5)$$

Reduced models and fundamental oscillations. The dominant part of the raw model is determined using a model reduction approach [2], [7], [4]. The procedure is as follows. Compute the singular value decompositions (SVDs):

$$[u_1, s_1, v_1] = \text{svd} \left( \begin{bmatrix} E \\ A \end{bmatrix} \right), \quad [u_2, s_2, v_2] = \text{svd} \left( \begin{bmatrix} E \\ A \end{bmatrix} \right).$$

Choose the dimension $r$ of the reduced system (e.g $r = 3, 5, r = 7$ etc.). Then

$$X = u_2(1 : k, 1 : r), \quad Y = v_1(1 : k, 1 : r),$$

are used to project the raw system to the dominant system of order $r$:

$$E_r = X^T E Y \in \mathbb{R}^{r \times r}, A_r = X^T A Y \in \mathbb{R}^{r \times r}, C_r = CY \in \mathbb{R}^{1 \times r}, x_r = X^T x_0 \in \mathbb{R}^{r \times 1}.$$ 

The associated reduced model of size $r$ is:

$$E_r x_r(n + 1) = A_r x_r(n), \quad y_r(n) = C_r x_r(n), \quad E_r x_r(0) = B_r.$$ 

Assuming (as is mostly the case) that $E_r$ is invertible, the approximated data can be expressed as:

$$\hat{y}_n = C_r [E_r^{-1} A_r]^{n-1} [E_r^{-1} B_r].$$

Next, we compute the partial fraction expansion of the associated transfer function:

$$H_r(z) = C_r(zE_r - A_r)^{-1}B_r.$$ 

This involves the eigenvalue decomposition (EVD) of the matrix pencil $(A_r, E_r)$, or equivalently of $E_r^{-1} A_r$; let

$$E_r^{-1} A_r = V_r \Lambda_r V_r^{-1},$$

where the columns of $V_r = [v_1, \ldots, v_r]$ are the eigenvectors, $\Lambda_r = \text{diag}[\lambda_1, \ldots, \lambda_r]$ are the eigenvalues of the reduced system (poles of $H_r(z)$), and $[\hat{v}_1^T, \ldots, \hat{v}_r^T]$ are the rows of $V_r^{-1}$. The approximate data can be expressed as:

$$\hat{y}_n = \sum_{i=1}^{r} [Cv_i] [\hat{v}_i^T B] \lambda_i^{n-1} = \sum_{i=1}^{r} P_i \lambda_i^{n-1} = \alpha_i e^{\sigma_i n} e^{j(\omega_i n + \theta_i)},$$

where $P_i = [Cv_i^T] [\hat{v}_i^T B]$, is the complex amplitude of the $i^{th}$ oscillation; expressing this in polar form $P_i = \alpha_i e^{j\theta_i}$, $\alpha_i$ is the real amplitude and $\theta_i$ the phase. Finally, if we express the eigenvalues as $\lambda_i = e^{\sigma_i + j\omega_i}$, $\sigma_i$ is the decay (growth) rate, and $\omega_i$ the frequency, of the $i^{th}$ oscillation.
MUSIC method

The MUSIC algorithm is a parametric spectral estimation method based on eigenvalue analysis of a correlation matrix. It uses the orthogonality of the signal subspace and the noise subspace to estimate the frequency of each oscillation. It assumes that a set of data can be modeled as \( Y = \Gamma a + n \), where \( Y = [y_1 \ y_2 \ \ldots \ y_N]^T \in \mathbb{R}^N \), is a set of gene transcription data, \( \Gamma = [e(\omega_1) \ e(\omega_2) \ \ldots \ e(\omega_K)] \) is the transpose of a Vandermonde matrix, \( K \) is the number of dominant frequencies, and \( e(\omega_i) = [1 \ e^{i\omega_i} \ \ldots \ e^{i(K-1)\omega_i}]^T \), \( a = [a_1 \ a_2 \ \ldots \ a_K]^T \) contains the amplitudes of the dominant \( K \) frequencies, \( n \sim \mathcal{N}(0, \sigma_n^2 I) \), is assumed to be white noise. The autocorrelation matrix is

\[
R_{yy} = \frac{1}{M} \sum_{i=1}^{M} yy^H = \Gamma \Lambda^2 \Gamma^H + \sigma_n^2 I
\]

where \( \Lambda = \text{diag}(\lambda_i) \) and \( M \) is the number of columns in the Hankel matrix. We can see that the rank of the matrix \( \Gamma \Lambda^2 \Gamma^H \) equals \( K \) where the nonzero eigenvalues are \( \{\lambda_m\}_{m=1}^K \). Then the sorted eigenvalues of the autocorrelation matrix \( R_{xx} \) can be expressed as

\[
\lambda_n = \tilde{\lambda}_n + \sigma_n^2, \quad n \leq K, \quad \text{and} \quad \sigma_n^2, \quad K < n \leq N.
\]

It follows that the noise subspace contains the eigenvectors of the autocorrelation matrix \( R_{xx} \) corresponding to the \( N - K \) smallest eigenvalues. Then

\[
R_{xx} G = G \text{diag}[\lambda_{K+1}, \ldots, \lambda_N] = \Gamma \Lambda^2 \Gamma^H G + \sigma_n^2 G
\]

so \( \Gamma^H G = 0 \), and the frequency values \( \{\tilde{\lambda}_k\}_{k=1}^K \) are the only solutions of \( e(\omega)^H GG^H e(\omega) = 0 \). The MUSIC algorithm seeks the peaks of the function

\[
1/ \left[ e(\omega)^H GG^H e(\omega) \right], \quad \omega \in [0, 2\pi].
\]

The Root MUSIC algorithm seeks the roots of \( e(\omega)^H GG^H e(\omega) \) that is the Z-transform of \( e(\omega)^H GG^H e(\omega) \) where \( z = e^{i\omega} \in \mathbb{C} \).

The MUSIC algorithm can only provide the frequency information of the signal. To obtain the amplitude of each oscillation, we need to apply least squares fitting, where the amplitudes of dominant oscillations satisfy \( a = (\Gamma^H \Gamma)^{-1} \Gamma^H x \). It should be mentioned that in contrast with the pencil method, MUSIC cannot provide the decay (growth) rate of the oscillations.

ESPRIT method

The ESPRIT algorithm is another parametric spectral estimation algorithm which analyzes the subspace of the correlation matrix. The algorithm estimates the poles relying on rotational transformations. As in MUSIC: \( \Gamma_{i,j} = z_j^{i-1}, \ j = 1, \ldots, K, \ i = 1, \ldots, N \), where \( z_j \) are the poles. We can construct \( \Gamma_1 = \Gamma(1 : N - 1,:), \) and \( \Gamma_2 = \Gamma(2 : N,:) \). The relationship between these two quantities is \( \Gamma_2 = \Gamma_1 \Phi \), where \( \Phi = \text{diag} \left[ z_1, z_2, \ldots, z_K \right] \), is the phase shift matrix that represents a rotation. Now using the signal matrix \( S \) we construct

\[
S_1 = S(1 : N - 1,:), \quad S_2 = S(2 : N,:).
\]

Note that the relationship between \( S_1 \) and \( S_2 \) is \( S_2 = S_1 \Psi \). Because \( \Gamma \) and \( S \) have the same column space (see the reference), we have that \( \Gamma = ST \), where \( T \) is an invertible subspace rotation matrix. So we have \( \Psi = T^{-1} \Phi T \). Therefore the poles are the eigenvalues of \( \Psi \). Finally use least squares (LS) to obtain \( \Psi = (S_H S_1)^{-1} S_H^T S_2 \). The eigenvalues of \( \Psi \) are the poles \( z_i = e^{j\omega_i+\sigma_i} \). Thus ESPRIT can estimate both the frequency and the decay (growth) rate of the oscillations. However, as with MUSIC, we need to use LS to obtain the amplitude of each oscillation. For details on the MUSIC and ESPRIT methods see [6], [7].
**Wavelet transform**

Wavelet transforms can be divided into two categories, the continuous (CWT) and the discrete (DWT) versions. CWT is more suitable for analyzing biologic rhythms because of the associated two-dimensional heat map.

In CWT a time signal \( x(t) \) is convolved with a wavelet function. This leads to a time-frequency representation which provides spectrum information in a local time window. This transform can be expressed as

\[
W_\psi(t, s) = \int_{-\infty}^{\infty} \frac{1}{s} \psi^* \left( \frac{u-t}{s} \right) x(u) du,
\]

where \( s \) is the frequency scale, \( \psi^* \) is the wavelet function. Since the signal data is obtained by sampling, we can approximately rewrite the equation as

\[
W_\psi(t, s) = \sum_{n=-\infty}^{\infty} \psi^* \left( \frac{n-t}{s} \right) x(n).
\]

Since the limits of integration are \(-\infty\) to \(\infty\), the finite signals \( y \) produce errors, and edge effects become obvious, especially in low-frequencies.

In practice, there are many wavelet functions that can be chosen, both real-valued and complex-valued. Real-valued wavelets are useful for treating peaks and discontinuities of signals while complex-valued wavelets yield the information of amplitude and phase simultaneously. For details on this approach, see e.g. [10].

**Statistical methods**

In this section three methods to detect biological rhythms will be briefly discussed. These are ARSER, JTK_CYCLE, RAIN. Those methods focus on the (one) most dominant oscillation in the data, especial JTK_CYCLE and RAIN. Those methods are all statistical tests that calculate the \( p \)-value to determine whether a certain rhythm exists in the data. For details on these methods we refer to the original references [8,9,12].

**ARSER**

ARSER uses the autoregressive (AR) model to obtain the period of oscillation. It then uses linear (harmonic) regression to determine the amplitude and the phase of the oscillation. Finally applying the F-test to pre-processed data concludes whether an oscillation exists.

*Pre-processing the Data.* Because the data may not be stable, ARSER applies linear detrending to the raw data. It then uses linear regression to fit the data. Subsequently it uses a fourth-order Savizky-Golay algorithm to smooth the data (this is a low-pass filter to remove pseudo-peaks in the spectrum).

*Finding the Period.* ARSER uses autoregressive model to get the period of the oscillation. Given a pre-processed data \( \{y_i\}_{i=1}^N \) with period interval \( \Delta \),

\[
y_t = \sum_{i=1}^{n} \alpha_i y_{t-i} + \epsilon_t,
\]

where \( \epsilon_t \) is white noise, \( \alpha_i \) are AR coefficients, \( n \) is the order of model (we choose \( n = \text{length-of-data}/\Delta \)). To calculate the coefficients, ARSER uses the Yule-Walker method, maximum likelihood estimation and the Burg algorithm. Finally the spectrum is:

\[
s(\omega) = \sigma^2 \left| 1 + \sum_{k=1}^{n} \alpha_k \exp^{-i\omega k} \right|^2,
\]

where \( \sigma^2 \) is the variance of white noise.

*Harmonic Regression.* The pre-processed data is:

\[
y_t = \mu + \sum_{i=1}^{m} \{ \beta_{i1} \cos(2\pi t/T_i) + \beta_{i2} \sin(2\pi t/T_i) \} + \epsilon_t,
\]
where \( \beta_{i1} \) and \( \beta_{i2} \) are the amplitudes calculated by linear regression.

**F-test.** This test compares the approximation data \( \{\hat{y}_t\} \) and pre-processed data \( \{y_t\} \). The null and the alternative hypotheses are respectively

\[
H_0: \quad A_1 = A_2 = \cdots = A_r, \quad H_1: \quad A_i \neq 0, \quad \text{for at least one value of } i,
\]

where \( A_i \) are the amplitudes which are calculated using linear regression, \( r \) is the number of coefficients obtained by linear regression. We can calculate the F coefficient by:

\[
F = \frac{\sum_{i=1}^{N}(\hat{y}_i - \bar{y})^2/(r-1)}{\sum_{i=1}^{N}(y_i - \bar{y})^2/(N-r)}.
\]

Then we can calculate the \( p \)-value using the \( F \)-distribution \( p = P(F, r-1, N-r) \), where \( P(\cdot) \) is the probability distribution.

**JTK_CYCLE and RAIN**

JTK_CYCLE and RAIN use statistical method to detect the trends in data. The former can find the increasing or decreasing trends while RAIN combines those together.

A periodic waveform should start from the trough and increase to the peak following a decreasing part to a new trough. Because our data is sampled from the waveform, we can regard every time sampled data point as a variable. Thus we can get \( n \) variables \( \{F_i\}_{i=1}^{n} \) for the waveform that \( T = n\Delta \) (\( T \) is the period of the waveform, \( \Delta \) is the time interval of sampling point). We assume the variances of those variables are the same. And they have the same mean value only when the data only have noise without periodic oscillation. So the null and the alternative hypotheses are

\[
H_0: \quad F_1 = F_2 = \cdots = F_n, \quad H_1: \quad F_1 < F_2 < \cdots < F_n \text{ or } F_1 > F_2 > \cdots > F_n.
\]

The alternative hypotheses for RAIN is

\[
H_1: \quad F_1 < F_2 < \cdots < F_{c} < F_{c+1} > \cdots > F_n > F_1.
\]

**Calculating the statistical coefficient of trend.** Every variable \( F_i \), corresponds to a sampled dataset \( \{X_{ij}\}_{j=1}^{m_i} \), where \( m_i \) is the number of sampled data point of the \( i^{th} \) variable (\( \sum_{i=1}^{n} m_i = N \)). Let

\[
q_{ik,j} = 1, \quad \text{if } X_{ik} \leq X_{j}, \quad \text{and } 0 \text{ otherwise}, \quad \text{and } U_{ij} = \sum_{k=1}^{m_i} \sum_{l=1}^{m_j} q_{ik,j},
\]

which is the Mann–Whitney U-statistic for comparison of two variables. For JTK_CYCLE, the statistical coefficient of trend is

\[
s = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} U_{ij}.
\]

For RAIN, the statistical coefficient of trend is

\[
s = \sum_{i=1}^{e-1} \sum_{j=i+1}^{e} U_{ij} + \sum_{i=e}^{n-1} \sum_{j=i+1}^{n} U_{ij} + \sum_{i=e+1}^{n} U_{ii}
\]

**Calculating the p-value.** This is: \( p(s) = \frac{f(s)}{\sum_{i:s} f(i)} \). In order to calculate the p-value, we need to specify the distribution \( f(i) \) of the statistical coefficient \( s \) when the null
Table 1. Parameters used for the simulation

| i | A  | σ  | θ   | T   |
|---|----|----|-----|-----|
| 1 | 1  | 0.005 | 0   | ∞   |
| 2 | 1  | 0.004 | π/6 | 24.8|
| 3 | 0.3| -0.002 | π/2 | 11.8|
| 4 | 0.1| 0.005 | π/2 | 7.5 |

hypotheses $H_0$ is true. Furthermore the distribution $f(i)$ is computed, using a generating function $G(z) = \sum_{i=0}^{\infty} z^i f(i)$. For JTKCYCLE and RAIN we have respectively:

$$G(z) = \prod_{u=1}^{N} (1 - z^u) = \prod_{u=1}^{N} (1 - z^u_1) \cdot \prod_{u=2}^{N} (1 - z^u_2) \cdot \prod_{u=3}^{N} (1 - z^u_3)$$

Thus $G(z)$ for JTKCYCLE and RAIN are both polynomials. We can get the distribution $f(i)$ by calculating the coefficients of $G(z)$.

Experimental Results: artificial data

In this section we test the performance of different methods using artificially generated signals. For the continuous wavelet transform, we chose the complex morlet wavelet because it allows changes to the resolution in frequency and time domain. For simulation data, we assume the data has the form

$$y(n) = \sum_{i=1}^{n} f_i(n) + n(n)$$

where $n$ is white noise with zero mean and variance $\sigma^2$ and $f_i$ is the $i^{th}$ oscillation, where:

$$f_i(n) = A_i e^{-\sigma_i n} \cos \left( \frac{2\pi}{T_i} n + \theta_i \right)$$

where $A_i$ is the amplitude, $\sigma_i$ is the decay (growth) rate, $\theta_i$ is the phase and $T_i$ is the period. At first we assume that the samples are collected in unit time intervals. The parameters are defined in Table 1; the first oscillation is almost constant with small decay; the other three oscillations have a period of approximately 24-12- and 8-hours.

The experiment has the following parts. First, the sensitivity to noise is investigated. Here, the variance of noise is changed and the performance of different methods is examined. Second, the impact of the length of the data is investigated. Finally, the frequency of data collection (can be referred to as sampling rate) is examined. Recall that the Nyquist sampling theorem provides the lower bound for the sampling frequency in order to prevent aliasing. This can be used to determine appropriate sampling frequencies for continuous-time signals.

Sensitivity to noise

To test the sensitivity of the various methods to noise, we set the standard deviation of $n$ to $\sigma = [0, 0.03, 0.1, 0.3]$. 
Fig 2. This figure shows fit curves of different methods and simulation data (length 50) with $\sigma$ as stated. The red points are simulation data, blue, green and magenta are the curves of the pencil, ESPRIT and MUSIC methods respectively. This figure shows that the pencil and ESPRIT methods yield a perfect fit in all situations. The MUSIC algorithm gives a good fit only for small noise. Next we also display the poles obtained by using each method.

| $\sigma = 0.01$ | $\sigma = 0.1$ | $\sigma = 0.3$ |
|-----------------|-----------------|-----------------|
| orig. poles | Pencil | ESPRIT | MUSIC | orig. poles | Pencil | ESPRIT | MUSIC | orig. poles | Pencil | ESPRIT | MUSIC |
| 0.990 | 0.990 | 0.990 | 1.000 | 0.990 | 0.989 | 0.989 | 1.000 | 0.990 | 0.990 | 0.990 | 1.000 |
| 0.958 ± 0.248i | 0.958 ± 0.248i | 0.958 ± 0.248i | 0.970 ± 0.239i | 0.958 ± 0.248i | 0.960 ± 0.248i | 0.960 ± 0.249i | 0.974 ± 0.225i | 0.958 ± 0.248i | 0.960 ± 0.248i | 0.960 ± 0.249i | 0.974 ± 0.225i |
| 0.870 ± 0.502i | 0.870 ± 0.512i | 0.870 ± 0.512i | 0.867 ± 0.497i | 0.870 ± 0.502i | 0.867 ± 0.511i | 0.867 ± 0.512i | 0.834 ± 0.551i | 0.870 ± 0.502i | 0.867 ± 0.511i | 0.867 ± 0.512i | 0.834 ± 0.551i |
| 0.662 ± 0.735i | 0.662 ± 0.735i | 0.662 ± 0.735i | 0.693 ± 0.721i | 0.662 ± 0.735i | 0.669 − 0.772i | 0.662 ± 0.751i | −0.974 ± 0.2235i | 0.662 ± 0.735i | 0.669 − 0.772i | 0.662 ± 0.751i | −0.974 ± 0.2235i |

Fig 3

In Fig 3 the heat map of the wavelet transform is shown. Thus the yellow region is such that we cannot distinguish two oscillations with close periods. We can recognize 12h and 8h oscillations when the noise is weak. However when the noise is strong ($\sigma = 0.3$), only the strongest oscillation can be determined. The edge effect is obvious and there are ghost lines e.g. around 15h, that may lead to false estimation.

From these considerations, we conclude that the pencil and ESPRIT methods are robust to noise. This is not the case for MUSIC and CWT.

Impact of data length

Fig 4. The plot shows fit curves of different methods and simulation data (noise standard deviation 0.05) with duration $L = [30, 50, 100, 200]$. The time interval for data collection is 1. Red points indicate simulation data, blue, green and magenta are the fit curves of pencil, ESPRIT and MUSIC algorithms, respectively.

Fig 5. The plot shows poles of oscillations estimated with different methods (noise standard deviation 0.05) with duration $L = [30, 50, 100, 200]$. The time interval for data collection is 1. Black * indicates the original poles of the simulation data, blue, green and magenta are the estimated poles using the pencil, ESPRIT and MUSIC algorithm, respectively. For more accuracy, the poles are also listed in the Table 3.

Rate of data collection (sampling frequency)

To investigate the impact of sampling of the underlying continuous-time signal, we generate artificial data with $L = 50$. Then we apply all methods to the original dataset, the half-data set (time collection interval $I = 2$) and third-data set (that is 1, 4, 7, 10 · · · with time interval $I = 3$).
Table 3

| $L = 30$ | $L = 100$ |
|----------|----------|
| orig. poles | Pencil | ESPRIT | MUSIC | orig. poles | Pencil | ESPRIT | MUSIC |
| 0.995 | 0.896 | $-1.043$ | 1.000 | 0.995 | 0.994 | 0.994 | 1.000 |
| 0.964 ± 0.249i | 0.778 ± 0.661i | 0.305 ± 0.000i | 0.977 ± 0.213i | 0.964 ± 0.249i | 0.964 ± 0.249i | 0.964 ± 0.249i | 0.969 ± 0.246i |
| 0.863 ± 0.505i | 0.447 ± 0.000i | 0.772 ± 0.653i | 0.806 ± 0.591i | 0.863 ± 0.505i | 0.863 ± 0.508i | 0.863 ± 0.508i | 0.857 ± 0.514i |
| 0.665 ± 0.739i | 1.093 ± 0.329i | 1.085 ± 0.324i | 0.456 ± 0.889i | 0.665 ± 0.739i | 0.661 ± 0.734i | 0.659 ± 0.733i | 0.648 ± 0.761i |

$L = 50$ $L = 200$

| orig. poles | Pencil | ESPRIT | MUSIC | orig. poles | Pencil | ESPRIT | MUSIC |
| 0.995 | 0.995 | 0.995 | 1.000 | 0.995 | 0.995 | 0.995 | 1.000 |
| 0.964 ± 0.249i | 0.964 ± 0.250i | 0.964 ± 0.250i | 0.970 ± 0.239i | 0.964 ± 0.249i | 0.964 ± 0.249i | 0.964 ± 0.249i | 0.972 ± 0.234i |
| 0.863 ± 0.505i | 0.863 ± 0.511i | 0.863 ± 0.510i | 0.824 ± 0.566i | 0.863 ± 0.505i | 0.863 ± 0.508i | 0.863 ± 0.508i | 0.857 ± 0.514i |
| 0.665 ± 0.739i | 0.655 ± 0.727i | 0.652 ± 0.731i | $-0.336 ± 0.941i$ | 0.665 ± 0.739i | 0.663 ± 0.737i | 0.663 ± 0.737i | $-0.336 ± 0.941i$ |

Fig 6. The plot shows heat maps (Y-axis is frequency domain, X-axis is time domain) of simulation data (noise standard deviation 0.05) with duration $L = [30, 50, 100, 200]$.

Fig 7. The plot shows data fit for the various methods.

**Conclusion.** From the above considerations it follows that decreasing the sampling frequency does not affect the estimation significantly. This means that the data rate collection (sampling frequency) is not an important factor. In contrast, the data length is a crucial factor for all methods.

**Experimental Results: the pencil method applied to gene data**

In this section we analyze a small part of the measured data in order to validate some of the aspects of the pencil method compared to the other methods.

Fig 8. Batch consisting of 171 measurements every 40min

Fig 9. Batch consisting of RER for restrictively fed mice (218 meas. every 40min)

We analyze the relationship among the decomposed oscillations, by calculating the angle among these oscillations for 10 different genes. We set $r = 9$, i.e. the gene signals contain four oscillations $f_i$, $i = 1, \ldots, 4$. The approximant is thus $\hat{y} = f_0 + f_1 + f_2 + f_3 + f_4$.

From Table 6 and 7, we can see that the angle between oscillations is around $90^\circ$ in most situations. So oscillations are nearly orthogonal with each other ($f_i \perp f_j$, $i \neq j$), in other words they are mathematically independent of each other.

Fig 10. Batch consisting of various measurements using mice — 38 min intervals

**Variation of data collection rate**

We compare the oscillations using all data (AD), the first half of data (FHD), the second half of data (SHD), odd-position data (OD), even-position data (ED). This is done for a particular set measurements, but the results are indicative of what happens in general.
Table 4. Relative approximation error

| Gene | 3-fit | 5-fit | 7-fit | 9-fit |
|------|-------|-------|-------|-------|
| Gene 1 | 0.1973 | 0.1276 | 0.1122 | 0.1299 |
| Gene 2 | 0.2217 | 0.2028 | 0.1669 | 0.1375 |
| Gene 3 | 0.2801 | 0.3940 | 0.2038 | 0.2112 |
| Gene 4 | 0.2654 | 0.2525 | -- | 0.2026 |
| Gene 5 | 0.4296 | 0.3780 | 0.1970 | -- |
| Gene 6 | 0.2493 | 0.2563 | 0.1918 | 0.1929 |
| Gene 7 | 0.1971 | 0.1525 | 0.1475 | 0.1547 |
| Gene 8 | 0.1914 | 0.1681 | 0.1402 | 0.1619 |
| Gene 9 | 0.1832 | 0.1913 | 0.1403 | 0.1357 |
| Gene 10 | 0.2016 | 0.2013 | 0.1874 | 0.2089 |
| Gene 11 | 0.2637 | 0.2623 | -- | 0.2083 |
| Gene 12 | 0.2174 | 0.1681 | 0.2116 | 0.1484 |
| Gene 13 | 0.3420 | 0.2154 | -- | 0.2270 |
| Gene 14 | 0.3140 | 0.2671 | 0.2452 | 0.2034 |
| Gene 15 | 0.4058 | 0.3374 | 0.3052 | 0.2281 |

Table 5. Angle between approximant and error

| Gene | 3-fit | 5-fit | 7-fit | 9-fit |
|------|-------|-------|-------|-------|
| Gene 1 | 88.72 | 88.65 | 88.66 | 90.46 |
| Gene 2 | 88.00 | 89.84 | 87.27 | 86.17 |
| Gene 3 | 91.92 | -- | 92.25 | 91.54 |
| Gene 4 | 89.82 | 94.18 | -- | 92.30 |
| Gene 5 | 84.35 | 86.36 | 89.74 | -- |
| Gene 6 | 86.94 | 91.78 | 88.39 | 88.78 |
| Gene 7 | 89.71 | 88.23 | 88.33 | 90.17 |
| Gene 8 | 87.45 | 88.19 | 87.02 | 89.11 |
| Gene 9 | 86.36 | 92.63 | 86.64 | 86.68 |
| Gene 10 | 86.78 | 87.81 | 86.42 | 89.90 |
| Gene 11 | 92.80 | 91.36 | 180 | 90.92 |
| Gene 12 | 91.20 | 90.18 | 94.12 | 90.59 |
| Gene 13 | 87.25 | 88.50 | 180 | 91.57 |
| Gene 14 | 90.36 | 94.35 | 93.30 | 91.35 |
| Gene 15 | 88.15 | 84.41 | 91.66 | 90.31 |

Discussion and comments

1. Orthogonality. Let the original vector of measurements for one data set be denoted by $y \in \mathbb{R}^N$; let also $f_i, i = 0, 1, 2, 3, 4$, denote the vectors of the DC-component and of the first four fundamental oscillations obtained by means of the pencil reduction method described above. Then the corresponding approximant is

\[
\hat{y} = f_0 + f_1 + f_2 + f_3 + f_4.
\]

It follows that:
Table 6. Angle between Error vector and approximants.

| Gene     | $r = 3$    | $r = 5$  | $r = 7$  | $r = 9$ |
|----------|------------|----------|----------|---------|
| Bmal     | 89.4040    | 89.0189  | 88.7227  | 89.4645 |
| Clock    | 97.5846    | 95.6007  | –        | 154.5354|
| per1     | 87.3120    | 87.0905  | –        | 122.6093|
| per2     | 84.0943    | 84.3410  | 84.2252  | 97.1281 |
| cry1     | 83.6787    | 85.7345  | 83.9466  | –       |
| cry2     | 88.0607    | 85.8548  | 85.7156  | 87.9577 |
| rorc     | 88.2740    | 87.0592  | 90.5345  | –       |
| rora     | 92.5359    | –        | 90.2449  | 90.3424 |
| rev-erba | 93.4881    | 92.5612  | 91.1162  | 91.4786 |
| reb-rebb | 89.2219    | 89.2972  | 89.0471  | 90.6819 |

Table 7. Angle between Oscillations.

| Gene    | $f_1$ vs $f_2$ | $f_1$ vs $f_3$ | $f_1$ vs $f_4$ | $f_2$ vs $f_3$ | $f_2$ vs $f_4$ | $f_3$ vs $f_4$ |
|---------|----------------|----------------|----------------|----------------|----------------|----------------|
| Bmal    | 90.9499        | 91.8664        | 87.7962        | 85.2451        | 91.2452        | 91.7038        |
| Clock   | 89.4592        | 87.9364        | –              | 106.0165       | –              | –              |
| per1    | 85.4061        | 93.9105        | 87.4712        | 74.9960        | 90.2287        | 101.0929       |
| per2    | 91.6425        | 94.1211        | 89.7681        | 88.9246        | 90.6757        | 90.4533        |
| cry1    | 83.3704        | 87.0513        | –              | 89.2173        | –              | –              |
| cry2    | 84.0615        | 91.3131        | 90.0791        | 90.9828        | 86.2981        | 88.1623        |
| rorc    | 88.6977        | 94.5379        | 87.0044        | 99.9135        | 85.2751        | 93.1401        |
| rora    | 91.3788        | 89.7184        | 89.8657        | 92.8563        | 88.6223        | 90.5763        |
| rev-erba | 94.9717      | 83.6197        | 88.9055        | 98.3908        | 90.8681        | 91.7753        |
| reb-rebb | 88.4669      | 89.5753        | 90.7263        | 90.9262        | 88.9671        | 92.8038        |

Table 8. Estimated periods using different part of the data. It follows that the estimation of periods of oscillations are consistent using AD, FHD, SHD.

|        | AD/h | FHD/h | SHD/h | OD/h | ED/h |
|--------|------|-------|-------|------|------|
| 1      | 24.37| 23.01 | 24.36 | 24.37| 24.37|
| 2      | 12.34| 12.41 | 12.46 | 11.90| 12.58|
| 3      | 8.12 | 8.42  | 7.45  | 8.25 | 8.13 |

▲ The fundamental oscillations are orthogonal among themselves: $f_i \perp f_j$, $i \neq j$.

▲ The approximant (composed of these oscillations) is orthogonal to the error (noise): $\hat{y} \perp \epsilon = y - \hat{y}$.

2. Interpretation of orthogonality. Orthogonality means that once an oscillation (e.g. the circadian or the 12h rhythm) has been determined, further computations will
The least squares (Prony's) method (red dots) gives oscillatory poles, this is by far not the case with the LS (prony's) method (green dots).

6. Comparison of different Methods.
Table 10

| Method          | Parameter Estimation | Estimation Performance | Detection of orthogonality |
|-----------------|----------------------|------------------------|----------------------------|
|                 | Period               | Decay Rate             | Amplitude                  | Phase | Accuracy | Robustness |                     |
| DFT             | Yes                  | No                     | Yes                        | Yes   | Low      | Yes        | No               |
| Wavelet         | Yes                  | Yes                    | Yes                        | No    | Low      | No         | No               |
| MUSIC           | Yes                  | No                     | No                         | No    | High     | No         | No               |
| ESPRIT          | Yes                  | Yes                    | No                         | No    | High     | Yes        | No               |
| Prony (LS)      | Yes                  | Yes                    | No                         | No    | No       | No         | No               |
| Pencil          | Yes                  | Yes                    | Yes                        | Yes   | High     | Yes        | Yes              |

Final result

The dataset consists of 18484 genes; transcription is analyzed using the pencil method [4], the ESPRIT method, Prony’s method and the three statistical methods. The distribution of the poles follow; recall that the poles of ideal oscillations have magnitude equal to 1.

Fig 12. Pencil 12h: 2354 genes

Fig 13. ESPRIT 12h: 2345 genes

Fig 14. Prony (LS) 12h: 265 genes

Using the three statistical methods we get (see Table 11): Furthermore the DFT and Wavelet methods are also not competitive.

The above distributions show that the Pencil method has uncovered real oscillations, since the mean of the magnitude of all poles is 1.0058 and the standard deviation is 0.0010. The ESPRIT method follows in terms of discovering oscillations, while the Prony or LS (Least Squares) method and the three statistical methods give weak results. As explained above the main drawback of the ESPRIT method concerns the fact that it has nothing to say about the orthogonality of the oscillations, which proves to be a key outcome of the pencil method.

Concluding remarks and outlook

Summarizing: the matrix pencil method allows the consistent determination of the dominant reduced-order models, which reveals the fundamental oscillations present in the data. The essence of the matrix pencil method is that it provides a
continuous-time tool for treating a discrete-time (sampled-data) problem. Instead, the DFT for instance, is only a discrete-time tool for treating a discrete-time problem, hence its failure.

A key consequence of this approach is the orthogonality of the different oscillatory components, in particular the 24h and the 12h ones. This points to an independence of these oscillations. This fact has been subsequently confirmed in the laboratory experiments reported in [13].

References

1. Alberts B, Bray D, Hopkin K, Johnson A, Lewis J, Raff M, et al. Essential cell biology. Garland Science; 2013.
2. Antoulas AC. Approximation of large-scale dynamical systems (Advances in Design and Control). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics; 2005.
3. Antoulas A, Lefteriu S, Ionita A. A tutorial introduction to the Loewner framework for model reduction. Model Reduction and Approximation for Complex Systems, Birkhäuser, ISNM Series. 2015.
4. Ionita AC, Antoulas AC. Matrix pencils in time and frequency domain system identification. Control, Robotics and Sensors Institution of Engineering and Technology. 2012; p. 79–88.
5. Zhang R, Lahens NF, Ballance HI, Hughes ME, Hogenesch JB. A circadian gene expression atlas in mammals: Implications for biology and medicine. Proceedings of the National Academy of Sciences. 2014;111(45):16219–16224.
6. Kay SM. Modern spectral estimation: Theory and application. Englewood Cliffs, NJ: Prentice Hall; 1988.
7. Stoica P, Moses RL. Introduction to spectral analysis. Upper Saddle River, NJ: Prentice Hall; 1997.
8. Yang R, Su Z. Analyzing circadian expression data by harmonic regression based on autoregressive spectral estimation. Bioinformatics. 2010;26(12):i168.
9. Hughes ME, Hogenesch JB, Kornacker K. JTK_CYCLE: An efficient nonparametric algorithm for detecting rhythmic components in genome-scale data sets. Journal of Biological Rhythms. 2010;25(5):372–380.
10. Leise TL. Wavelet analysis of circadian and ultradian behavioral rhythms. Journal of Circadian Rhythms. 2013;11:Art. 5.
11. Takahashi JS. Transcriptional architecture of the mammalian circadian clock. Nature Reviews Genetics. 2017;18:164–179.
12. Thaben PF, Westermark PO. Detecting rhythms in time series with RAIN. Journal of Biological Rhythms. 2014;29(6):391–400.
13. Zhu B, Zhang Q, Pan Y, Mace EM, York B, Antoulas AC, et al. A cell-autonomous mammalian 12-hr clock coordinates metabolic and stress rhythms. Cell Metabolism. 2017;25(6):1305 – 1319.e9.
Data averaged over all mice

| A    | P    | T     |
|------|------|-------|
| 0.1594 | 0.9022 | ——   |
| 0.0010 | 1.0050 | 1.4483 |
| 0.0017 | 0.9985 | 1.8434 |
| 0.0034 | 0.9956 | 9.8050 |
| 0.0164 | 1.0013 | 23.9361 |
| 0.9239 | 0.9986 | dc   |

Approximation by 1, 2 and 3 oscillations

First 5 oscillations

Plots for averaged data
Approximation by 1, 2 and 3 oscillations

| Mouse #1 | A  | P  | T       |
|---------|----|----|---------|
| 0.0037  | 1.0005 | 4.8275 |
| 0.0116  | 0.9961 | 7.4236 |
| 0.0256  | 0.9993 | 7.9961 |
| 0.0010  | 1.0043 | 20.2774 |
| 0.0817  | 1.0001 | 23.9264 |
| 0.8843  | 1.0001 | dc    |

First 4 oscillations
| Food intake | Ambulatory activity | Total activity | ZTOT | Heat |
|-------------|---------------------|----------------|------|------|
| A           | P                   | T              | A    | P    | T    | A    | P    | T    | A    | P    | T    |
| 0.0049      | 1.0014              | 1.4798         | 34.3158 | 1.0029 | 2.1857 | 46.2589 | 0.9996 | 2.1752 | 39.9181 | 1.0055 | 6.0855 | 0.0076 | 1.0013 | —      |
| 0.0143      | 0.9946              | 1.5812         | 87.9712 | 0.9997 | 8.0524 | 139.9357 | 1.0002 | 8.0445 | 86.2169 | 1.0052 | 8.1064 | 0.0225 | 0.9936 | 8.1278 |
| 0.0106      | 1.0002              | 8.5909         | 111.7862 | 1.0004 | 12.1124 | 183.2241 | 1.0009 | 12.1327 | 138.1809 | 1.0052 | 12.1725 | 0.0095 | 1.0019 | 12.3403 |
| 0.0302      | 0.9977              | 23.9810        | 185.3298 | 1.0016 | 24.4907 | 317.1999 | 1.0021 | 24.4595 | 195.7413 | 1.0071 | 24.3164 | 0.0281 | 1.0027 | 24.3605 |
| 0.1189      | 0.9992              | dc             | 504.7523 | 1.0003 | dc     | 1045.0577 | 1.0005 | dc     | 338.0709 | 1.0062 | dc     | 0.5181 | 0.9999 | dc     |

**Approximation by 1, 2 and 3 oscillations**

**First 4 oscillations**
Eigenvalues of the 19th order pencil (red triangles) and LS (green circles) models
Pencil (num = 2354, mean = 1.0056, var = 0.0010677, skew = 0.069728)
ESPRIT (num = 2345, mean=0.98779, var=0.0012169, skew=-0.23674)
Prony (num = 265, mean=0.93397, var=0.00089266, skew=1.4874)