SoftwareX 15 (2021) 100769

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

oreo: An R package for large amplitude oscillatory analysis

Giorgio Luciano [a], Serena Berretta [b], Kristian Hovde Liland [c,*], Gavin J. Donley [d], Simon A. Rogers [d]

[a] CNR - SCITEC, Via De Marini 6, 16149 Genoa, Italy
[b] CNR - Imati, Via De Marini 6, 16149 Genoa, Italy
[c] Norwegian University of Life Sciences, Department of Science and Technology, NO-1432 Ås, Norway
[d] University of Illinois at Urbana-Champaign, Urbana, IL, USA

A R T I C L E I N F O

Article history:
Received 29 March 2021
Received in revised form 13 July 2021
Accepted 14 July 2021

Keywords:
LAOS (large amplitude oscillatory shear)
Rheology
Viscoelastic materials
R

A B S T R A C T

Specialized software designed to analyze nonlinear rheological data from large amplitude oscillatory shear tests is not widespread. Instances that exist that are open source are typically based on proprietary meta-languages. The availability of a package in the R environment can help researchers to rely on a Free Open Source Software reproducible work flow. In this context, we have developed the package *oreo* which is able to analyze data from rheometry measurements using the Sequence of Physical Processes (SPP) framework that allows both the linear and non-linear deformation regimes to be understood withing a single unified framework.

Code metadata

Current code version	v 1.0
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-21-00058
Legal Code License	GPL-2
Software code languages, tools, and services used	R
Compilation requirements, operating environments & dependencies	R no compilation needed; Dependencies: `gridExtra`, `ggplot2`, `openxlsx`, `tools`, `spectral`, `pracma`, `fftwtools`, `scales`
If available Link to developer documentation/manual	https://cran.r-project.org/web/packages/oreo/oreo.pdf
Support email for questions	serena.berretta@ge.imati.cnr.it

1. Motivation and significance

Many industrial, environmental, and biological applications involve subjecting soft materials to intervals of strong flow over short time scales where the stress–strain relationships cannot be described by linear differential equations with constant coefficients [1]. Soft matter scientists in general, and rheologists in particular seek to investigate these nonlinear material behaviors in controlled and reproducible ways. To this end, large-amplitude oscillatory shearing (LAOS) has become an ideal test of nonlinear rheology because time scales can be tuned via the applied angular frequency, and flow strengths can be changed by altering the amplitude of the flow [2,3]. LAOS has a significant role to play in food science, as summarized in a review by Joyner [4]. Cho [5] reports that the LAOS methodology has become popular in various fields of material characterizations from nonlinear architecture of polymer chains or long chain branching to applications to colloidal systems. Despite its popularity, software that is designed to analyze large-amplitude oscillatory shear (LAOS) rheological responses is not widespread and often available only bundled with measurement instruments. Few free and open source instances exist (Reptate [6] RHEOS, [7]) and also open source ones based on proprietary frameworks (MITLAOS [8], SPP software [9]). The availability of a package in the R environment can help researchers to rely on a Free Open Source Software (FOSS) reproducible work flow [10]. In this context the authors developed the package `oreo`.

2. Software framework

Large amplitude oscillatory shear (LAOS) is a well-established technique in the fields of rheology and material science to study the impact of a range of flow strength and time scales and their

* Corresponding author.
E-mail address: kristian.liland@nmbu.no (Kristian Hovde Liland).

https://doi.org/10.1016/j.softx.2021.100769

impact on material responses. In this context, Rogers et al. [9,11,12] presented a robust methodology that views the stress response to large strains inputs as being a sequence of physical processes (SPP). The SPP framework overcame significant problems encountered in Fourier transformation-based analyses, and has since been used to study a wide range of soft materials.

The key idea behind the SPP framework is that the dynamic moduli that define viscoelasticity are not required to remain constant throughout a test, and that their values change transiently over the course of a period of oscillation once deformation exists in the linear regime.

The SPP framework defines two instantaneous moduli by utilizing the mathematics of trajectories from differential geometry. To begin, the measured material response is viewed as a trajectory within a three-dimensional deformation space. See, for instance, fig. 3 of [12].

The framework uses the mathematics of the Frenet–Serret Frame (FSF), or Tangent–Normal–Binormal Frame (TNB Frame) [12,13], to track the differential motion of the trajectory through deformation space. This frame decomposes the motion of the trajectory into three orthonormal vectors: the tangent vector, which indicates the current direction of the trajectory; the normal vector, which indicates how that direction is changing; and the binormal vector, which is the vector cross product of the tangent and normal vectors. Locally speaking, the trajectory can be said to lie within a plane defined by the tangent and normal vectors. This plane is called the osculating plane. The binormal vector is therefore normal to the osculating plane, and defines its orientation, and therefore that of the trajectory. These vectors completely define the Frenet–Serret frame, and are used to determine the various SPP analysis metrics.

2.1. Theory

The fundamental concepts that underlay the Sequence of Physical Processes framework will now be presented. A full derivation and discussion of the metrics discussed here can be found in Rogers [9]. Within the fully quantitative SPP approach, a material's response to oscillatory shearing produces a trajectory in deformation space given by [11,12]:

$$A = [A_\gamma \quad A_{\dot\gamma/\omega} \quad A_\sigma] = [\gamma_0 \sin(\omega t) \quad \gamma_0 \cos(\omega t) \quad \sigma(t)] \\ = [\gamma(t) \quad \dot\gamma(t)/\omega \quad \sigma(t)]. \tag{1}$$

The trajectory can be described by a set of three orthonormal vectors called the tangent ($\boldsymbol{T}$), principal normal ($\boldsymbol{N}$), and binormal ($\boldsymbol{B}$), vectors. The tangent vector points in the direction of instantaneous motion, and the principal normal points in the direction of the derivative of the tangent:

$$\mathbf{T} = \frac{\dot{\mathbf{A}}}{|\dot{\mathbf{A}}|} \tag{2}$$

$$\mathbf{N} = \frac{\dot{\mathbf{T}}}{|\dot{\mathbf{T}}|} \tag{3}$$

Here, we use the dot notation to represent differentiation with respect to time. The tangent and principal normal vectors span the osculating plane, which can be thought of as the plane in which the trajectory sits on a local scale. The binormal vector is given by the vector cross product of the tangent and principal normal vectors and therefore defines the orientation of the osculating plane:

$$\mathbf{B} = \mathbf{T} \times \mathbf{N} \tag{4}$$

The SPP framework defines two transient moduli (G'_t and G''_t) which are differential parameters that represent the orientation of the trajectory in deformation space. While the transient moduli define the orientation of the osculating plane, a complete description of any trajectory requires information regarding the position of the osculating plane as well as the plane's orientation, as discussed in Rogers [9]. The SPP is therefore unique in understanding LAOS, as it defines a third instantaneous parameter, called the displacement stress (σ^d), that accounts for the position of the plane:

$$\sigma = G'_t \gamma + G''_t \dot\gamma/\omega + \sigma^d. \tag{5}$$

To determine the required form of the displacement stress, the point-normal form of the equation of a plane becomes useful, as the binormal vector is normal to the osculating plane by definition:

$$B_x(x - A_x) + B_y(y - A_y) + B_z(z - A_z) = 0 \tag{6}$$

The SPP analysis determines the position of the osculating plane along the stress axis, and so it is customary to set the x and y-components of the plane to zero and solve:

$$\sigma^d = \frac{B_\gamma}{B_\sigma}\gamma + \frac{B_{\dot\gamma/\omega}}{B_\sigma}\dot\gamma/\omega + \sigma. \tag{7}$$

Substituting this form of the displacement stress back into Eq. (5) leads to a description of the trajectory that may be rearranged as follows:

$$\left(G'_t + \frac{B_\gamma}{B_\sigma}\right)\gamma + \left(G''_t + \frac{B_{\dot\gamma/\omega}}{B_\sigma}\right)\dot\gamma/\omega = 0. \tag{8}$$

On the basis of Eq. (8), the SPP framework therefore defines the transient moduli as:

$$G'_t(t) = -\frac{B_\gamma(t)}{B_\sigma(t)} \tag{9}$$

$$G''_t(t) = -\frac{B_{\dot\gamma/\omega}(t)}{B_\sigma(t)} \tag{10}$$

In addition to defining transient moduli, the SPP framework also provides explicit definitions of their derivatives, which can be used to inform researchers not only whether the response is softening, stiffening, thickening, or thinning, but also when and by how much. The derivatives of the transient moduli have slightly more complex forms than the transient moduli themselves, and require the principal normal vector, the binormal vector, and the torsion ($\tau = -|\dot{\boldsymbol{A}}|\boldsymbol{NB}$) which geometrically tells us how fast the osculating plane rotates around the axis given by the tangent vector:

$$\dot G'_t = \tau|\dot{\boldsymbol{A}}|\left(\frac{N_\gamma}{B_\sigma} - \frac{B_\gamma N_\sigma}{B_\sigma^2}\right) \tag{11}$$

$$\dot G''_t = \tau|\dot{\boldsymbol{A}}|\left(\frac{N_{\dot\gamma}/\omega}{B_\sigma} - \frac{B_{\dot\gamma}/\omega N_\sigma}{B_\sigma^2}\right). \tag{12}$$

In addition to defining the time-dependent moduli and their derivatives, the SPP framework is unique among oscillatory shear analysis techniques in that it accounts for unrecoverable strain via the inclusion of a moving strain equilibrium position, and a yield stress that is not represented by the moduli. While the orientation of the osculating plane gives information regarding the local moduli, it is its displacement that contains information about the strain equilibrium position and the yield stress. The displacement stress, defined by Eq. (7), is physically interpreted as being equal to

$$\sigma^d(t) = \sigma_y(t) - G'_t(t)\gamma_{eq}(t) = \sigma(t) - G'_t(t)\gamma(t) + G''_t(t)\dot\gamma(t)/(\omega) \tag{13}$$

When the response is predominantly elastic, $G'_t(t) \gg G''_t(t)$, the equality expressed in Eq. (13) can be simplified to

$$\gamma(t) - \gamma_{eq}(t) = \frac{\sigma(t)}{G'_t(t)} \tag{14}$$

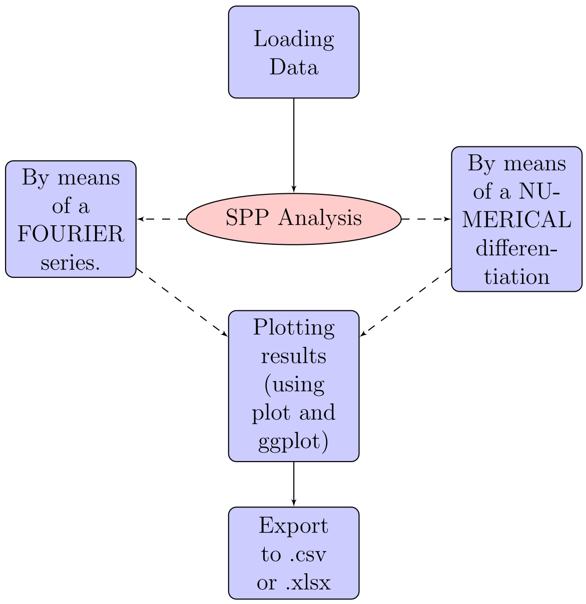

Fig. 1. Float chart of data handling and analysis using the oreo package.

allowing for a straightforward determination of the equilibrium position, and therefore also the recoverable and unrecoverable components of the strain.

2.2. Input data requirements

For the oreo code to function properly, the input should be provided as a *comma-separated variables* file (.csv). All data should be laid out in column orientation.

For strain-controlled tests, the file must contain the following data in order: Time, Strain, Strain Rate, and Shear Stress to be input to the RPPread function. A full discussion of dimensions and data for stress-controlled experiments can be found in Appendix 1 of [9].

The preferred units for each of the columns are as follows: `Time [s]`, `Strain [-] (strain units)`, `Strain Rate [1/s]`, and `Shear Stress [Pa]`. If the units in the data differ from these units, the code can perform the appropriate unit conversion(s).

For the input data, an integer number of periods must be covered with an even number of data points per period.

Within the `oreo` package, three sample files .csv containing properly formatted input data (`0010.csv`, `0100.csv`, and `1000.csv`) are provided for reference. These files contain data from responses of the Giesekus model to different applied strain amplitudes at an angular frequency of $\omega = 3.16\ rad/s$. The model parameters for all files are $\lambda_1 = 1\ s, \eta_s = 0.01\ Pa \cdot s, \eta_p = 0.01\ Pa \cdot s, \alpha = 0.3$. The data are arranged into four columns: Time (s), Strain (-), Rate (1/s) and Shear Stress (Pa), as required for a strain-controlled test. The files `0010.csv`, `0100.csv`, and `1000.csv` have strain amplitudes of $\gamma_0 = 10$, 100, and 1000 strain units, respectively.

Its latest version (1.0), is downloadable from the Comprehensive R Archive Network (CRAN)
http://CRAN.R-project.org/package=oreo

The package is installed into R software writing the following commands in an R terminal:

```
install.packages('oreo')
library(oreo)
```

we can then read a csv file with the command

```
filename="/Folder/SubFolder/0010.csv"
dat <- read.csv(filename, na.strings = "NA",
             dec = ".", header = TRUE)
```

or directly use a saved dataframe in `.RData` format. The function `rpp_read` will take care of checking the presence of the columns needed and will re-arrange the file

```
rpp_read <- function(filename,
         selected = c(2, 3, 4, 0, 0, 1, 0, 0))
```

where the `selected = c(2, 3, 4, 0, 0, 1, 0, 0)` option provides the numbers of the columns that refer in the original data to strain, strain-rate, stress, elast stress, visco stress, raw time, raw stress, raw strain. In the example given we have that the strain in the original data is the second column, the strain-rate the third, the stress the fourth, we do not have elast stress neither visco stress, the raw time in the original data is the first column and finally we do not have any column that represents raw stress neither raw strain. The function will in fact give as output

```
[1] "Elastic-Stress is missing"
[1] "Visco-Stress is missing"
[1] "Raw Stress is missing"
[1] "Raw Strain is missing"
```

Both numeric and fast Fourier transform (FFT) decompositions used to determine the SPP framework can be invoked given the required parameters using the following commands:

```
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain, df$strain_rate, df$stress)

# Set parameters for the analysis
L = dim(df)[1]
omega = 3.16
M = 15
k = 8
num_mode = 1
p = 1

# Analyse the data using the fft approach
out <- rpp_fft(time_wave,resp_wave,L,omega,M,p)
```

where the arguments are defined as: `time_wave` vector of time at each measurement point, `resp_wave` matrix of the strain, rate and stress data with each row representing a measuring point, `L` number of measurements points in the extracted data, `omega` frequency of oscillation expressed in (rad/s), `M` is the number of harmonics used in the reconstruction of the stress and the SPP parameters (as laid out in [12,14]), and `p` number of cycles of the signal in exam.

The numeric differentiation analysis differs from the fft input for the following additional parameters: `k`: the step size for the numerical differentiation (positive integer) and `num_mode` that indicates the procedure used for the numerical differentiation using a single number. The first one, "standard differentiation" (called if `num_mode = 1`), makes no assumptions about the form of the data. It utilizes a forward difference to calculate the derivative for the first `2*k` points of the data, a backward difference for the final `2*k` points, and a centered difference elsewhere. The second, "looped differentiation" (called if `num_mode = 2`), assumes that the input data is taken under oscillatory conditions at steady alternance, and represents an integer number of periods. These assumptions allow a centered difference to be calculated everywhere by looping over the ends of the data.

Finally the package provides functions for creating summaries in csv format and Microsoft Excel (xlsx) format. They can be invoked as:

```
# Save the output in csv format
rpp_out_csv(out, "test2.csv")
# Save the file in xlsx format
rpp_out_excel(out, "test2.xlsx")
```

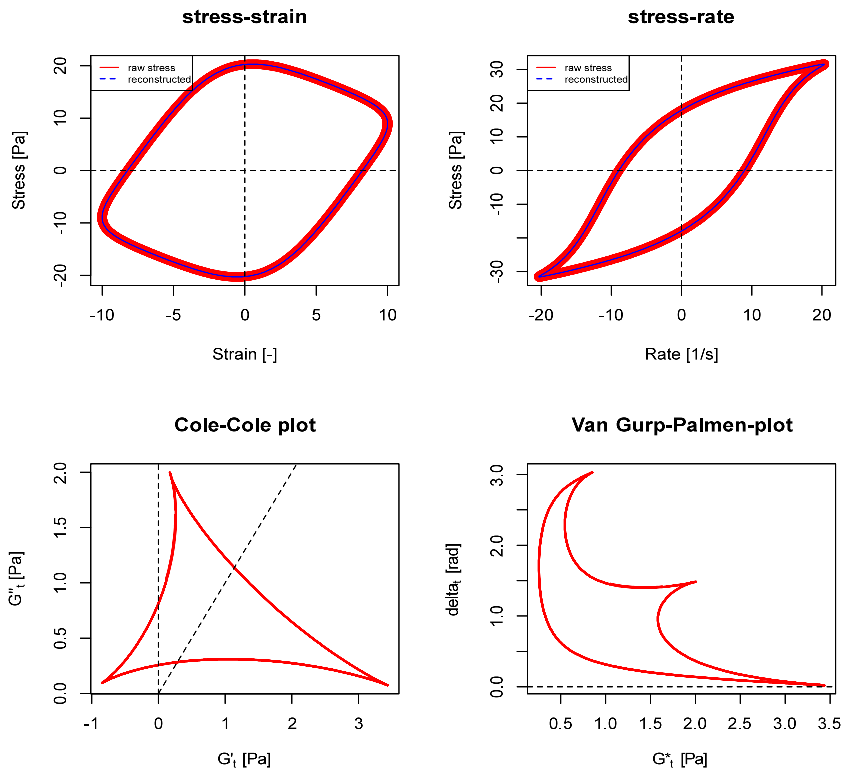

Fig. 2. The SPP metrics plotted from the Fourier analysis for a strain-controlled experiment the blue lines in the stress–strain plot and stress-rate plot represent the reconstructed signal after the analysis. The dashed line represents axes and bisection. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

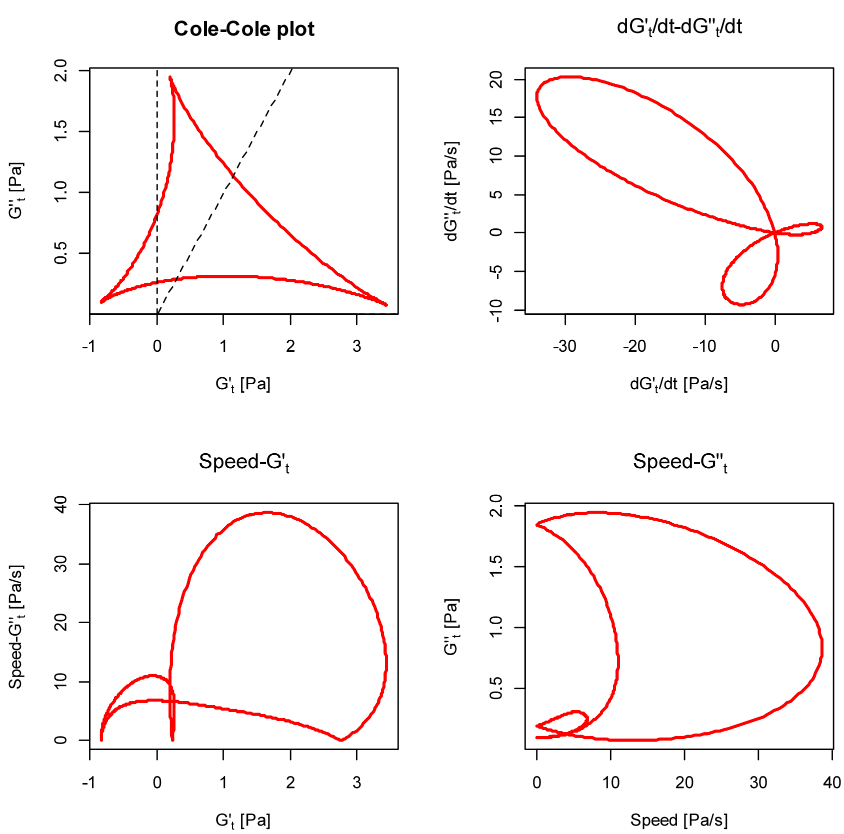

Fig. 3. The derivatives of the SPP metrics plot from the Fourier analysis.

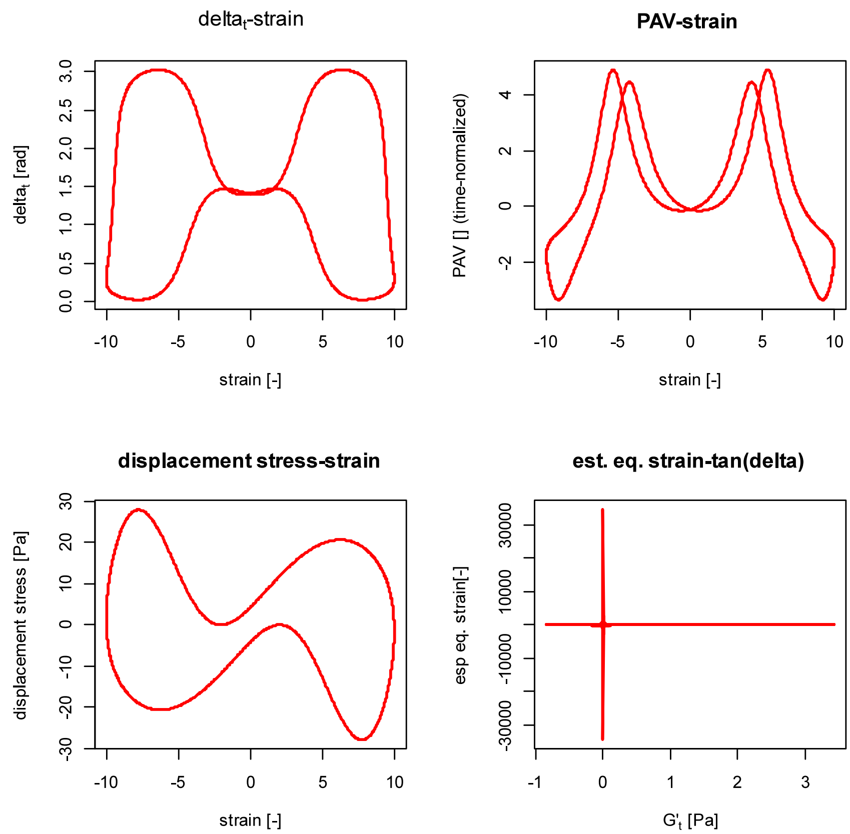

Fig. 4. The additional SPP metrics plot from the Fourier analysis.

where `out` represents the result of the `fft` or `numeric analysis`. The package also includes functions for creating diagnostics plots and more as follows

- SPP metric plots for the Fourier analysis (or differential numeric).
- Derivative of the SPP metric plots for the Fourier analysis (or differential numeric).
- Additional SPP metrics plots from the Fourier analysis (or differential numeric).
- Reconstructed waveform comparison plot from the Fourier analysis
- Plot of harmonics for the stress response from the Fourier analysis

3. Illustrative example

A float chart of typical usage of the `oreo` package is seen in Fig. 1. As a demonstration we will report a generic script used for creating part of the data sets and the analysis reported. The code can be found in Appendix A. Results can easily be plotted with standard R commands (Fig. 2) using the command in Appendix B. The resulting graphics are found in Figs. 2, 3, 4, 5, 6.

4. Impact

The package *oreo* will let researchers analyze their oscillatory rheological data without the need for licensing fees associated with proprietary meta-languages. It will speed up the workflow, letting users take advantage of a scripting language. Once implemented, scripts can be used in a laboratory and routinely applied to collected data, shifting the focus to the analysis of the results obtained instead on their calculation. The package is routinely used (and updated) in the laboratory of SCITEC "Giulio Natta" to analyze LAOS behavior in food and polymer samples allowing, via custom scripts, to merge data from different instruments (vendors and models). Two papers, that included LAOS analysis performed with the package are in the drafting stage. In order to spread its use, presentations at national and international rheology congress are planned. Download statistics from the official CRAN mirror reports 1359 downloads of the package at the time of writing (13/07/2021).

5. Conclusions

In this paper, we have presented *oreo*, an R-based, flexible, and powerful collection of functions for analyzing rheology data through the SPP framework. The package *oreo* is an ongoing project. We intend to improve its performance using specific architectures (Compute Unified Device Architecture) and to extend its features and functionality by implementing other ways of calculating Tangent–Normal–Binormal Frame (TNB Frame).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1847389 and the Laboratory Directed Research and Development program at Sandia National Laboratories. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the US Department of

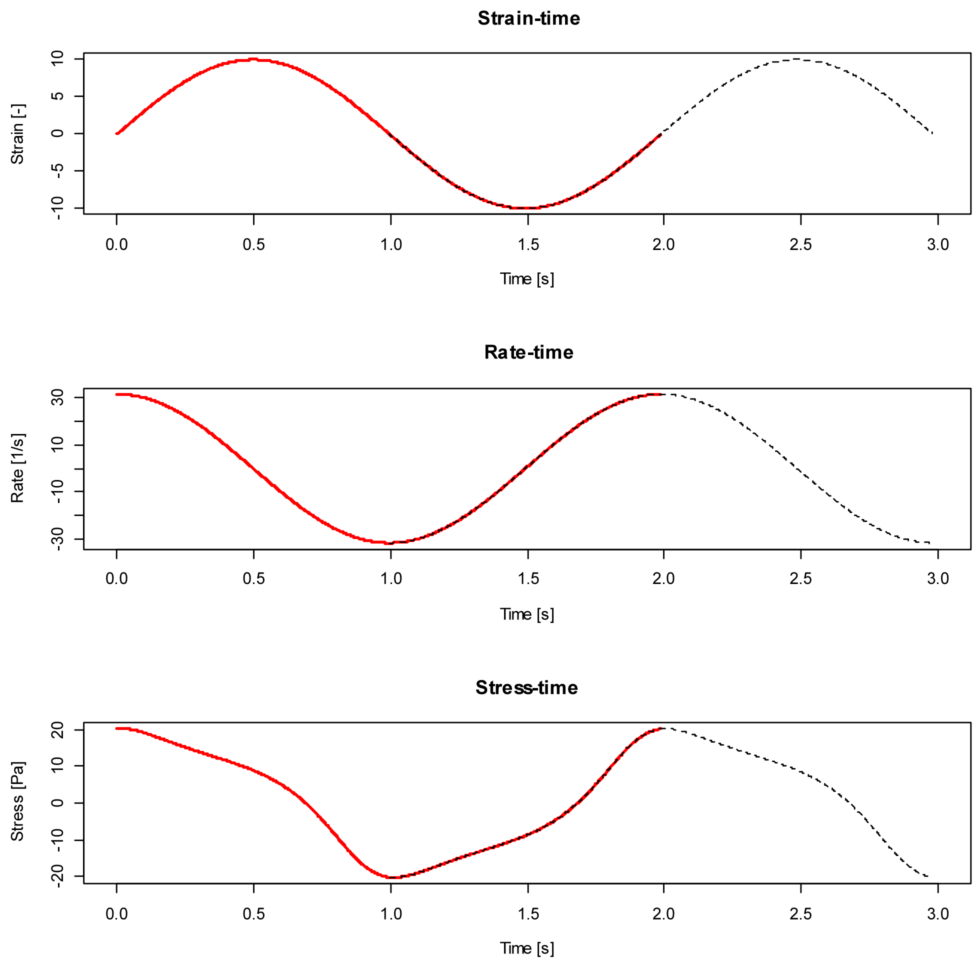

Fig. 5. The reconstructed waveform (red lines) comparison plot from the Fourier analysis. The dashed lines represent the input values. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Energy's National Nuclear Security Administration Contract No. DE-NA0003525.

Appendix A. Script

```
# Analyse the data using the fft approach
out <- rpp_fft(time_wave, resp_wave, L, omega, M, p)

# Calculations using numerical decomposition
# out <- rpp_num(time_wave, resp_wave, L, omega, M, p)

# Save the output in csv format
rpp_out_csv(out, "test2.csv")
# Save the file in xlsx format
rpp_out_excel(out, "test2.xlsx")
```

Appendix B. Plotting

```
# Create plot using base plot functions
# Prepare x and y for each plot

spp_data_in <- out$spp_data_in
spp_params <- out$spp_params
spp_data_out <- out$spp_data_out
fsf_data_out <- out$fsf_data_out
ft_out <- out$ft_out

time_in <- spp_data_in$time_wave
strain_in <- spp_data_in$strain
strain_rate_in <- spp_data_in$strain_rate
stress_in <- spp_data_in$stress

out_time <- spp_data_out$time_wave
strain <- spp_data_out$strain
rate <- spp_data_out$rate
stress <- spp_data_out$stress

Gp_t <- spp_data_out$Gp_t
Gpp_t <- spp_data_out$Gpp_t
G_star_t <- spp_data_out$G_star_t

tan_delta_t <- spp_data_out$tan_delta_t
delta_t <- spp_data_out$delta_t
disp_stress <- spp_data_out$disp_stress
eq_strain_est <- spp_data_out$eq_strain_est

Gp_t_dot <- spp_data_out$Gp_t_dot
Gpp_t_dot <- spp_data_out$Gpp_t_dot
G_speed <- spp_data_out$G_speed

delta_t_dot <- spp_data_out$delta_t_dot

ft_amp <- ft_out$ft_amp
fft_resp <- ft_out$fft_resp

# First panel
# Base SPP metrics figure
windows()
par(mfrow = c(2, 2))
# Stress-Strain plot
plotStressStrain(stress, strain)
# Stress-Rate plot
plotStressRate(stress, rate)
```

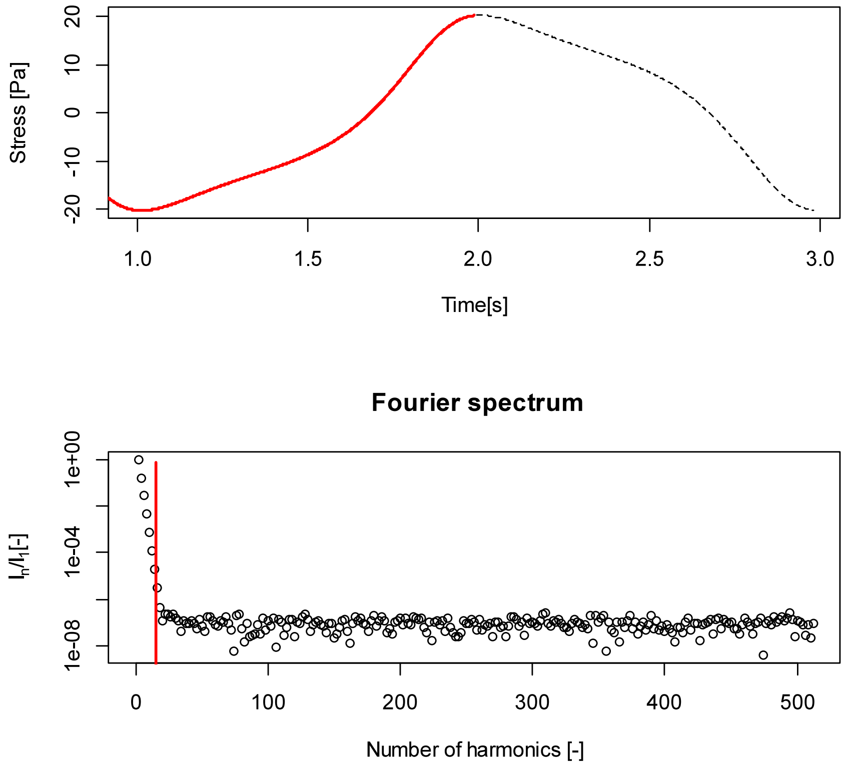

Fig. 6. The plot of harmonics for the stress response from the Fourier Analysis. The dashed line represents the input value while the red one shows the signal after analysis. In the 'number of harmonics' plot the red vertical bar highlights the number of harmonics chosen in the data analysis. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

```
# Cole-Cole plot
plotColeCole(Gp_t, Gpp_t)
# VGP Van Gurp-Palmen-plot plot
plotVGP(G_star_t, delta_t)

# Second Panel
# SPP metrics speed figure
windows()
par(mfrow = c(2, 2))
# Cole-Cole plot
plotColeCole(Gp_t, Gpp_t)
# dG"_{t}/dt-dG'_{t}/dt plot
plotGpdot(Gp_t_dot, Gpp_t_dot)
# Speed-G'_{t} plot
plotSpeedGp(Gp_t_dot, Gpp_t_dot)
# Speed-G"_{t} plot
plotSpeedGpp(Gp_t_dot, Gpp_t_dot)

# Third Panel
# SPP metrics additional plots
windows()
par(mfrow = c(2, 2))
plotDeltaStrain(Gp_t_dot, Gpp_t_dot)
# PAV v strain plot
plotPAV(Gp_t_dot, Gpp_t_dot)
# Displacement stress plot
plotDisp(strain, disp_stress)
# Estimated eq strain plot
plotStrain(strain, disp_stress)

# Fourth Panel
# Waveform Comparison
windows()
par(mfrow = c(3, 1))
plotTimeStrain(time_wave, strain)
# Rate-Time plot
plotTimeRate(time_wave, rate)
# Stress-Time plot
plotTimeStress(time_wave, stress)

# FT-spectra for stress
windows()
par(mfrow = c(2, 1))
plotStressTime(time_in, stress_in)
# Fourier Harmonic Magnitudes plot
plotFft(ft_amp, fft_resp)
```

References

[1] Larson RG. The structure and rheology of complex fluids, Vol. 150. Oxford university press New York; 1999.

[2] Rogers S. Large amplitude oscillatory shear: simple to describe, hard to interpret. Phys Today 2018;71(7):34–40.

[3] Saengow C, Giacomin AJ. Review of nonlinear oscillatory shear flow notations and presentations: polymeric liquids. Curr Opin Colloid Interface Sci 2019;43:26–38, Cited By :14.

[4] Joyner HS. Nonlinear (large-amplitude oscillatory shear) rheological properties and their impact on food processing and quality. Annu Rev Food Sci Technol 2021;12(1):591–609. http://dx.doi.org/10.1146/annurev-food-061220-100714, PMID: 33770471.

[5] Cho KS. Large amplitude oscillatory shear. In: Viscoelasticity of polymers: Theory and numerical algorithms. Dordrecht: Springer Netherlands; 2016, p. 545–99. http://dx.doi.org/10.1007/978-94-017-7564-9_11.

[6] Boudara VAH, Read DJ, Ramírez J. Reptate rheology software: Toolkit for the analysis of theories and experiments. J Rheol 2020;64:709–22. http://dx.doi.org/10.1122/8.0000002.

[7] Kaplan J, Bonfanti A, Kabla A. Rheos.jl – a julia package for rheology data analysis. J Open Source Softw 2019;4(41):1700. http://dx.doi.org/10.21105/joss.01700.

[8] Ewoldt R, Winter P, McKinley G. MITlaos version 2.1 beta for MATLAB. Cambridge, MA: Self-Published: MATLAB-Based Data Analysis Software for Characterizing Nonlinear Viscoelastic Responses To Oscillatory Shear Strain; 2007.

[9] Rogers SA. In search of physical meaning: defining transient parameters for nonlinear viscoelasticity. Rheol Acta 2017. http://dx.doi.org/10.1007/s00397-017-1008-1.
[10] Stodden V, Leisch F, Peng RD. Implementing Reproducible Research. Boca Raton: Chapman and Hall/CRC; 2014, http://dx.doi.org/10.1201/b16868.
[11] Rogers SA, Erwin BM, Vlassopoulos D, Cloitre M. A sequence of physical processes determined and quantified in LAOS: Application to a yield stress fluid. J Rheol 2011. http://dx.doi.org/10.1122/1.3544591.
[12] Rogers SA, Lettinga MP. A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models. J Rheol 2012. http://dx.doi.org/10.1122/1.3662962.
[13] Rogers SA. In search of physical meaning: Defining transient parameters for nonlinear viscoelasticity. Rheol Acta 2017;56(5):501–25.
[14] Choi J, Nettesheim F, Rogers SA. The unification of disparate rheological measures in oscillatory shearing. Phys Fluids 2019;31(7):073107.