

# Package ‘oreo’

May 9, 2026

**Type** Package

**Title** Large Amplitude Oscillatory Shear (LAOS)

**Version** 1.0

**Date** 2021-03-01

**Maintainer** Serena Berretta <serena.berretta@ge.imati.cnr.it>

**Description** The Sequence of Physical Processes (SPP) framework is a way of interpreting the transient data derived from oscillatory rheological tests. It is designed to allow both the linear and non-linear deformation regimes to be understood within a single unified framework. This code provides a convenient way to determine the SPP framework metrics for a given sample of oscillatory data. It will produce a text file containing the SPP metrics, which the user can then plot using their software of choice. It can also produce a second text file with additional derived data (components of tangent, normal, and binormal vectors), as well as pre-plotted figures if so desired. It is the R version of the Package SPP by Simon Rogers Group for Soft Matter (Simon A. Rogers, Brian M. Erwin, Dimitris Vlassopoulos, Michel Cloitre (2011) <doi:10.1122/1.3544591>).

**Imports** gridExtra,ggplot2,openxlsx,tools,spectral,pracma,fftwtools,scales

**Depends** R (>= 3.5.0)

**License** GPL-2

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Serena Berretta [aut, cre],  
Giorgio Luciano [aut],  
Kristian Hovde Liland [ctb],  
Simon Rogers [ctb]

**Repository** CRAN

**Date/Publication** 2021-03-10 19:20:07 UTC

## Contents

mydata . . . . .	2
------------------	---

plotColeCole . . . . .	3
plotDeltaStrain . . . . .	4
plotDisp . . . . .	5
plotFft . . . . .	6
plotGpdot . . . . .	7
plotPAV . . . . .	8
plotSpeedGp . . . . .	9
plotSpeedGpp . . . . .	10
plotStrain . . . . .	11
plotStressRate . . . . .	12
plotStressStrain . . . . .	14
plotStressTime . . . . .	15
plotTimeRate . . . . .	16
plotTimeStrain . . . . .	17
plotTimeStress . . . . .	18
plotVGP . . . . .	19
rpp_fft . . . . .	20
Rpp_num . . . . .	21
rpp_out_csv . . . . .	22
rpp_out_excel . . . . .	23
rpp_read . . . . .	24
rpp_read2 . . . . .	24

<b>Index</b>	<b>26</b>
--------------	-----------

---

mydata	<i>Data from the Giesikus model</i>
--------	-------------------------------------

---

## Description

The data is arranged into four columns: Time (s), Strain (-), Rate (1/s) and Stress (Pa). reflecting the applied strain- control

## Usage

```
data(mydata)
```

## Format

A data frame with 1024 rows and 4 columns

**V1** Time

**V2** Strain

**V3** Rate

**V4** Stress

## References

PPP

---

plotColeCole	<i>Cole-Cole plot</i>
--------------	-----------------------

---

**Description**

create Cole-Cole plot

create Cole-Cole plot

**Usage**

```
plotColeCole(Gp_t, Gpp_t, ...)
```

```
plotColeCole(Gp_t, Gpp_t, ...)
```

**Arguments**

Gp\_t                    from the output matrix from fft analysis or numerical differentiation analysis

Gpp\_t                   from the output matrix from fft analysis or numerical differentiation analysis

...                     parameters of plot()

**Value**

No return value

No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**References**

Simon A. Rogers, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogers, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

## Examples

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
Gp_t= out$spp_data_out$Gp_t
Gpp_t= out$spp_data_out$Gpp_t
plotColeCole(Gp_t,Gpp_t)
```

---

plotDeltaStrain      *Strain Delta Plot*

---

## Description

create Strain Delta Plot  
create Strain Delta Plot

## Usage

```
plotDeltaStrain(strain, delta_t, ...)  
plotDeltaStrain(strain, delta_t, ...)
```

## Arguments

strain	from the output matrix from fft analysis or numerical differentiation analysis
delta_t	from the output matrix from fft analysis or numerical differentiation analysis
...	parameters of plot()

## Value

No return value  
No return value

## Author(s)

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers  
Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers  
Group for Soft Matter

## References

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

## Examples

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
strain= out$spp_data_out$strain
delta_t= out$spp_data_out$delta_t
plotDeltaStrain(strain,delta_t)
```

---

plotDisp

*Strain Displacement Stress*

---

## Description

Strain Displacement Stress

Strain Displacement Stress

## Usage

```
plotDisp(strain, disp_stress, ...)
```

```
plotDisp(strain, disp_stress, ...)
```

## Arguments

strain	from the output matrix from fft analysis or numerical differentiation analysis
disp_stress	from the output matrix from fft analysis or numerical differentiation analysis
...	parameters of plot()

## Value

No return value

No return value

## References

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

## Examples

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
strain= out$spp_data_out$strain
disp_stress= out$spp_data_out$disp_stress
plotDisp(strain,disp_stress)
```

---

plotFft

*Fourier Harmonic Magnitudes plot*

---

## Description

create Fourier Harmonic Magnitudes plot

create Fourier Harmonic Magnitudes plot

## Usage

```
plotFft(ft_amp, fft_resp, spp_params, ...)
```

```
plotFft(ft_amp, fft_resp, spp_params, ...)
```

## Arguments

ft_amp	from the output matrix from fft analysis or numerical differentiation analysis
fft_resp	from the output matrix from fft analysis or numerical differentiation analysis
spp_params	input parameters used for the fft analysis or numerical differentiation analysis
...	parameters of plot()

## Value

No return value

No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- rpp_fft(time_wave,resp_wave,L=1024,omega=3.16 , M=15,p=1)
ft_amp= out$ft_out$ft_amp
fft_resp= out$ft_out$fft_resp
spp_params= out$spp_params
plotFft(ft_amp,fft_resp,spp_params)
```

---

plotGpdot

*Gp\_t\_dot vs Gpp\_t\_dot*


---

**Description**

create Gp\_t\_dot vs Gpp\_t\_dot

create Gp\_t\_dot vs Gpp\_t\_dot

**Usage**

```
plotGpdot(Gp_t_dot, Gpp_t_dot, ...)
```

```
plotGpdot(Gp_t_dot, Gpp_t_dot, ...)
```

**Arguments**

Gp\_t\_dot from the output matrix from fft analysis or numerical differentiation analysis

Gpp\_t\_dot from the output matrix from fft analysis or numerical differentiation analysis

... parameters of plot()

**Value**

No return value

No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
Gp_t_dot= out$spp_data_out$Gp_t_dot
Gpp_t_dot= out$spp_data_out$Gpp_t_dot
plotGpdot(Gp_t_dot,Gpp_t_dot)
```

---

plotPAV

*Strain Delta Plot*

---

**Description**

create Strain Delta Plot

create Strain Delta Plot

**Usage**

```
plotPAV(strain, delta_t_dot, ...)
```

```
plotPAV(strain, delta_t_dot, ...)
```

**Arguments**

strain from the output matrix from fft analysis or numerical differentiation analysis

delta\_t\_dot from the output matrix from fft analysis or numerical differentiation analysis

... parameters of plot()

**Value**

No return value

No return value

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
strain= out$spp_data_out$strain
delta_t_dot= out$spp_data_out$delta_t_dot
plotPAV(strain,delta_t_dot)
```

---

plotSpeedGp

*Speed-G' \_t plot*

---

**Description**

create Speed-G' \_t plot

create Speed-G' \_t plot

**Usage**

```
plotSpeedGp(Gp_t, G_speed, ...)
```

```
plotSpeedGp(Gp_t, G_speed, ...)
```

**Arguments**

Gp\_t from the output matrix from fft analysis or numerical differentiation analysis

G\_speed from the output matrix from fft analysis or numerical differentiation analysis

... parameters of plot()

**Value**

No return value

No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**References**

Simon A. Rogers, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogers, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
Gp_t= out$spp_data_out$Gp_t
G_speed= out$spp_data_out$G_speed
plotSpeedGp(Gp_t,G_speed)
```

---

plotSpeedGpp

*Speed-G''\_t plot*

---

**Description**

create Speed-G''\_t plot

create Speed-G''\_t plot

**Usage**

```
plotSpeedGpp(G_speed, Gpp_t, ...)
```

```
plotSpeedGpp(G_speed, Gpp_t, ...)
```

**Arguments**

G\_speed from the output matrix from fft analysis or numerical differentiation analysis  
 Gpp\_t from the output matrix from fft analysis or numerical differentiation analysis  
 ... parameters of plot()

**Value**

No return value  
 No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
G_speed= out$spp_data_out$G_speed
Gpp_t= out$spp_data_out$Gpp_t
plotSpeedGpp(G_speed,Gpp_t)
```

---

 plotStrain

*Strain Gp\_t,eq\_strain\_est*


---

**Description**

Strain Gp\_t,eq\_strain\_est  
 Strain Gp\_t,eq\_strain\_est

**Usage**

```
plotStrain(Gp_t, eq_strain_est, ...)
```

```
plotStrain(Gp_t, eq_strain_est, ...)
```

**Arguments**

Gp\_t                from the output matrix from fft analysis or numerical differentiation analysis  
 eq\_strain\_est    from the output matrix from fft analysis or numerical differentiation analysis  
 ...                parameters of plot()

**Value**

No return value

No return value

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
Gp_t= out$spp_data_out$Gp_t
eq_strain_est= out$spp_data_out$eq_strain_est
plotStrain(Gp_t,eq_strain_est)
```

---

plotStressRate

*Stress-Rate plot*

---

**Description**

create Stress Rate Plot

create Stress Rate Plot

**Usage**

```
plotStressRate(stress, rate, ...)
```

```
plotStressRate(stress, rate, ...)
```

**Arguments**

stress	data the output matrix from fft analysis or numerical differentiation analysis
rate	data the output matrix from fft analysis or numerical differentiation analysis
...	parameters of plot()

**Value**

No return value

No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
rate= out$spp_data_out$rate
stress= out$spp_data_out$stress
plotStressRate(stress, rate)
```

---

plotStressStrain      *Stress-Strain plot*

---

**Description**

create Stress Strain Plot  
create Stress Strain Plot

**Usage**

```
plotStressStrain(stress, strain, strain_in, stress_in, ...)  
plotStressStrain(stress, strain, strain_in, stress_in, ...)
```

**Arguments**

stress	data the output matrix from fft analysis or numerical differentiation analysis
strain	data the output matrix from fft analysis or numerical differentiation analysis
strain_in	data the input matrix from fft analysis or numerical differentiation analysis
stress_in	data the input matrix from fft analysis or numerical differentiation analysis
...	parameters of plot()

**Value**

No return value  
No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**References**

Simon A. Rogers, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogers, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```

data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
strain= out$spp_data_out$strain
stress= out$spp_data_out$stress
strain_in= out$spp_data_in$strain
stress_in= out$spp_data_in$stress
plotStressStrain(stress, strain,strain_in,stress_in)

```

---

plotStressTime	<i>Stress-Time plot</i>
----------------	-------------------------

---

**Description**

create Stress-Time plot  
create Stress-Time plot

**Usage**

```

plotStressTime(time_wave_in, stress_in, time_wave, stress)

plotStressTime(time_wave_in, stress_in, time_wave, stress)

```

**Arguments**

time_wave_in	raw time from input data
stress_in	stress from input data
time_wave	from the output matrix from fft analysis or numerical differentiation analysis
stress	from the output matrix from fft analysis or numerical differentiation analysis

**Value**

No return value  
No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**Examples**

```

data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
time_wave_in= out$spp_data_in$time_wave
stress_in= out$spp_data_in$stress
time_wave= out$spp_data_out$time_wave
stress= out$spp_data_out$stress
plotStressTime(time_wave_in, stress_in, time_wave, stress)

```

---

plotTimeRate	<i>Rate, time_wave plot</i>
--------------	-----------------------------

---

**Description**

create Rate, time\_wave plot  
create Rate, time\_wave plot

**Usage**

```

plotTimeRate(time_wave, rate, time_wave_in, strain_rate, ...)

plotTimeRate(time_wave, rate, time_wave_in, strain_rate, ...)

```

**Arguments**

time_wave	from the output matrix from fft analysis or numerical differentiation analysis
rate	from the output matrix from fft analysis or numerical differentiation analysis
time_wave_in	raw time from input data
strain_rate	strain rate from input data
...	parameters of plot()

**Value**

No return value  
No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

## References

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

## Examples

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
time_wave= out$spp_data_out$time_wave
rate= out$spp_data_out$rate
time_wave_in= out$spp_data_in$time_wave
strain_rate= out$spp_data_in$strain_rate
plotTimeRate(time_wave,rate,time_wave_in,strain_rate)
```

---

plotTimeStrain	<i>Strain time_wave, strain</i>
----------------	---------------------------------

---

## Description

Strain time\_wave, strain

Strain time\_wave, strain

## Usage

```
plotTimeStrain(time_wave, strain, time_wave_in, strain_in, ...)
```

```
plotTimeStrain(time_wave, strain, time_wave_in, strain_in, ...)
```

## Arguments

time_wave	time from output data
strain	from the output matrix from fft analysis or numerical differentiation analysis
time_wave_in	time from input data
strain_in	from the input matrix from fft analysis or numerical differentiation analysis
...	parameters of plot()

**Value**

No return value

No return value

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
time_wave= out$spp_data_out$time_wave
strain= out$spp_data_out$strain
time_wave_in= out$spp_data_in$time_wave
strain_in= out$spp_data_in$strain
plotTimeStrain(time_wave,strain,time_wave_in,strain_in)
```

---

plotTimeStress	<i>Stress-Time plot</i>
----------------	-------------------------

---

**Description**

create Stress-Time plot

create Stress-Time plot

**Usage**

```
plotTimeStress(time_wave, stress, time_wave_in, strain_rate, ...)
```

```
plotTimeStress(time_wave, stress, time_wave_in, strain_rate, ...)
```

**Arguments**

time_wave	from the output matrix from fft analysis or numerical differentiation analysis
stress	from the output matrix from fft analysis or numerical differentiation analysis
time_wave_in	raw time from input data
strain_rate	strain rate from input data
...	parameters of plot()

**Value**

No return value

No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
time_wave= out$spp_data_out$time_wave
stress= out$spp_data_out$stress
time_wave_in= out$spp_data_in$time_wave
strain_rate= out$spp_data_in$strain_rate
plotTimeStress(time_wave, stress, time_wave_in, strain_rate)
```

---

plotVGP

*VGP plot*

---

**Description**

create VGP plot

create VGP plot

**Usage**

```
plotVGP(G_star_t, delta_t, ...)
```

```
plotVGP(G_star_t, delta_t, ...)
```

**Arguments**

G\_star\_t from the output matrix from fft analysis or numerical differentiation analysis

delta\_t from the output matrix from fft analysis or numerical differentiation analysis

... parameters of plot()

**Value**

No return value

No return value

**Author(s)**

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

Giorgio Luciano and Serena Beretta, based on the Plotting functions created by Simon Rogers Group for Soft Matter

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models *Journal of Rheology* 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
G_star_t= out$spp_data_out$G_star_t
delta_t= out$spp_data_out$delta_t
plotVGP(G_star_t,delta_t)
```

---

rpp\_fft

*SPP Analysis via fourier*

---

**Description**

applies the SPP Analysis by means of a fourier series.

**Usage**

```
rpp_fft(time_wave, resp_wave, L, omega, M, p)
```

**Arguments**

time_wave	Lx1 vector of time at each measurement point
resp_wave	Lx3 matrix of the strain, rate and stress data, with each row representing a measuring point
L	number of measurement points in the extracted data
omega	frequency of oscillation (rad/s)
M	number of harmonics for stress
p	number of cycles

**Value**

a list with the following data frame spp\_data\_in= the data frame with the data spp\_params=spp\_params, spp\_data\_out= Length,frequency,harmonics,cycles,max\_harmonics,step\_size fsf\_data\_out= Tx,Ty,Tz,Nx,Ny,Nz,Bx,By,Bz coordinates of the trajectory (T=tangent,N=principal Normal,B=Binormal Vectors) ft\_out=data frame with that includes time\_wave,strain,rate,stress,Gp\_t,Gpp\_t,G\_star\_t,tan\_delta\_t,delta\_t,disp\_stress,eq\_strain\_est,Gp\_t\_dot,C

**Author(s)**

Simon Rogers Group for Soft Matter (matlab version), Giorgio Luciano and Serena Berretta (R version)

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models Journal of Rheology 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- rpp_fft(time_wave,resp_wave,L=1024,omega=3.16 , M=15,p=1)
```

---

Rpp\_num

*SPP Analysis via numerical differentiation*


---

**Description**

applies the SPP Analysis by means of a numerical differentiation.

**Usage**

```
Rpp_num(time_wave, resp_wave, L, k, num_mode)
```

**Arguments**

time_wave	Lx1 vector of time at each measurement point
resp_wave	Lx3 matrix of the strain, rate and stress data, with each row representing a measuring point
L	number of measurement points in the extracted data
k	step size for numerical differentiation
num_mode	numerical method

**Value**

a list with the following data frame spp\_data\_in= the data frame with the data spp\_params=spp\_params, spp\_data\_out= Length,frequency,harmonics,cycles,max\_harmonics,step\_size fsf\_data\_out= Tx,Ty,Tz,Nx,Ny,Nz,Bx,By,Bz coordinates of the trajectory (T=tangent,N=principal Normal,B=Binormal Vectors) ft\_out=data frame with that includes time\_wave,strain,rate,stress,Gp\_t,Gpp\_t,G\_star\_t,tan\_delta\_t,delta\_t,disp\_stress,eq\_strain\_est,Gp\_t\_dot,C

**Author(s)**

Simon Rogers Group for Soft Matter (matlab version), Giorgio Luciano and Serena Berretta (R version)

**References**

Simon A. Rogersa, M. Paul Letting, A sequence of physical processes determined and quantified in large-amplitude oscillatory shear (LAOS): Application to theoretical nonlinear models Journal of Rheology 56:1, 1-25

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
time_wave <- df$raw_time
resp_wave <- data.frame(df$strain,df$strain_rate,df$stress)
out <- Rpp_num(time_wave, resp_wave , L=1024, k=8, num_mode=1)
```

---

rpp\_out\_csv

*Export results of the performed SPP analysis in csv format*


---

**Description**

```
# This function export the output the SPP analysis (performed via FFT or Numeric Analysis) and export it to csv files
```

**Usage**

```
rpp_out_csv(out, myfilename = "my_models.xlsx")
```

**Arguments**

out                    output of the SPP analysis (performed via FFT or Numeric Analysis)  
myfilename            name of the file where to save results (csv)

**Value**

No return value

**Author(s)**

Simon Rogers Group for Soft Matter (matlab version), Giorgio Luciano and Serena Berretta (R version)

---

rpp\_out\_excel                    *Export results of the performed SPP analysis in xls format*

---

**Description**

# This function export the output the SPP analysis (performed via FFT or Numeric Analysis) and export it to xls files

**Usage**

```
rpp_out_excel(out, myfilename = "my_models.xlsx")
```

**Arguments**

out                    output of the SPP analysis (performed via FFT or Numeric Analysis)  
myfilename            name of the file where to save results in xls format

**Value**

No return value

**Author(s)**

Simon Rogers Group for Soft Matter (matlab version), Giorgio Luciano and Serena Berretta (R version)

---

 rpp\_read

*Read function*


---

**Description**

This function reads data from the selected file, and assign it to a dataframe

**Usage**

```
rpp_read(filename, header = TRUE, selected = c(2, 3, 4, 0, 0, 1, 0, 0), ...)
```

**Arguments**

filename	the name of the file to read
header	TRUE if colnames are present FALSE if colnames are not present
selected	the user should input the number of the columns that represent strain-smoothed (gamma), strain rate-smoothed (gamma dot), stress smoothed (tau recon), Elast-Stress (FTtau_e), Visco-Stress (FTtau_v), raw time (time), raw stress (tau), raw strain (gamma) i.e. selected=c(2, 3, 4, 0, 0, 1, 0, 0) means that the second column of your data is the strain rate smoothed, the third column is the stress smoothed, the stress smoothed is the fourth column in the original data, and finally that we do not have data for the raw stress and raw strain
...	parameters of read.csv

**Value**

a dataframe with all the columns assigned

**Author(s)**

Giorgio Luciano and Serena Berretta, Simon Rogers Group for Soft Matter (matlab version)

---

 rpp\_read2

*Read function*


---

**Description**

This function reads data from a dataframe

**Usage**

```
rpp_read2(dat, selected = c(2, 3, 4, 0, 0, 1, 0, 0), ...)
```

**Arguments**

<code>dat</code>	dataframe of input
<code>selected</code>	the user should input the number of the columns that represent strain-smoothed ( <code>gamma</code> ), strain rate-smoothed ( <code>gamma dot</code> ), stress smoothed ( <code>tau recon</code> ), Elast-Stress ( <code>FTtau_e</code> ), Visco-Stress ( <code>FTtau_v</code> ), raw time ( <code>time</code> ), raw stress ( <code>tau</code> ), raw strain ( <code>gamma</code> ) i.e. <code>selected=c(2, 3, 4, 0, 0, 1, 0, 0)</code> means that the second column of your data is the strain rate smoothed, the third column is the stress smoothed, the stress smoothed is the fourth column in the original data, and finally that we do not have data for the raw stress and raw strain
<code>...</code>	parameters of <code>read.csv</code>

**Value**

a dataframe with all the columns assigned

**Author(s)**

Giorgio Luciano and Serena Berretta, Simon Rogers Group for Soft Matter (matlab version)

**Examples**

```
data(mydata)
df <- rpp_read2(mydata , selected=c(2, 3, 4, 0, 0, 1, 0, 0))
```

# Index

- \* **Cole-Cole**
    - plotColeCole, 3
  - \* **SPP**
    - plotColeCole, 3
    - plotStressRate, 12
    - plotStressStrain, 14
    - rpp\_fft, 20
    - Rpp\_num, 21
  - \* **Stress-Strain**
    - plotStressRate, 12
    - plotStressStrain, 14
  - \* **csv**
    - rpp\_out\_csv, 22
  - \* **datasets**
    - mydata, 2
  - \* **differentiation**
    - Rpp\_num, 21
  - \* **export**
    - rpp\_out\_csv, 22
    - rpp\_out\_excel, 23
  - \* **fourier**
    - rpp\_fft, 20
  - \* **numerical**
    - Rpp\_num, 21
  - \* **plot**
    - plotColeCole, 3
    - plotStressRate, 12
    - plotStressStrain, 14
  - \* **results**
    - rpp\_out\_csv, 22
    - rpp\_out\_excel, 23
  - \* **xls**
    - rpp\_out\_excel, 23
- mydata, 2
- plotColeCole, 3
- plotDeltaStrain, 4
- plotDisp, 5
- plotFft, 6
- plotGpdot, 7
- plotPAV, 8
- plotSpeedGp, 9
- plotSpeedGpp, 10
- plotStrain, 11
- plotStressRate, 12
- plotStressStrain, 14
- plotStressTime, 15
- plotTimeRate, 16
- plotTimeStrain, 17
- plotTimeStress, 18
- plotVGP, 19
- rpp\_fft, 20
- Rpp\_num, 21
- rpp\_out\_csv, 22
- rpp\_out\_excel, 23
- rpp\_read, 24
- rpp\_read2, 24