Package ‘RLumModel’

December 7, 2015

Type Package

Title Modelling Ordinary Differential Equations Leading to Luminescence

Version 0.1.0

Date 2016-XX-XX

Author Johannes Friedrich [aut, trl, cre],
Sebastian Kreutzer [aut, ths],
Christoph Schmidt [aut, ths]

Maintainer Johannes Friedrich <johannes.jriedrich@uni-bayreuth.de>

Description A collection of function to simulate luminescence signals in the mineral quartz based on published models.

Contact Package Developer Team <developer@model.r-luminescence.de>

License GPL-3

Depends R (>= 3.2.2), utils, Luminescence (>= 0.5.0)

Imports deSolve (>= 1.12)

URL http://CRAN.R-project.org/package=RLumModel

Collate RLumModel-package.R RLumModel_CW_OSL.R RLumModel_LM_OSL.R RLumModel_ODE.R RLumModel_ODE_LM_OSL.R RLumModel_RL.R RLumModel_SAR.sequence.R RLumModel_SequenceTranslator.R RLumModel_TL.R RLumModel_calcSignal.R RLumModel_heating.R RLumModel_illumination.R RLumModel_irradiation.R RLumModel_pause.R RLumModel_seq2R.R RLumModel_setPars.R model_LuminescenceSignals.R

RoxygenNote 5.0.1

NeedsCompilation no

R topics documented:

RLumModel-package 2
model_LuminescenceSignals 3

Index **9**

`RLumModel-package` *Modelling Ordinary Differential Equations Leading to Luminescence*

Description

A collection of function to simulate luminescence signals in the mineral quartz based on published models.

Details

Package:	RLumModel
Type:	Package
Version:	0.1.0
Date:	2016-XX-XX
License:	GPL-3

Author(s)

Authors

Johannes Friedrich	University of Bayreuth, Germany
Sebastian Kreutzer	IRAMAT-CRP2A, Universite Bordeaux Montaigne, France,
Christoph Schmidt, University of Bayreuth, Germany	

Supervisor

Christoph Schmidt, Univestiy of Bayreuth, Germany
Sebastian Kreutzer, IRAMAT-CRP2A, Universite Bordeaux Montaigne, France

Support contact

<developers@model.r-luminescence.de>

Project source code repository

https://github.com/R-Lum/RLumModel

Related projects

http://www.r-luminescence.de
http://cran.r-project.org/package=Luminescence
http://shiny.r-luminescence.de
http://cran.r-project.org/package=RLumShiny

Package maintainer

Johannes Friedrich, University of Bayreuth, Germany
<johannes.friedrich@uni-bayreuth.de>

Acknowledgement

The work of Johannes Friedrich is gratefully supported by the DFG in framework of the project 'Modelling quartz luminescence signal dynamics relevant for dating and dosimetry' (SCHM 305114-1)

`model_LuminescenceSignals`

Model Luminescence Signals

Description

This function models luminescence signals for quartz based on published physical models. It is possible to simulate TL, (CW-) OSL, RF measurements in a arbitrary sequence. This sequence is definded as a list of certain abrivations. Furthermore it is possible to load a sequence direct from the Riso Sequence Editor. The output is an RLum.Analysis object and so the plots are done by the plot_RLum.Analysis function. If a SAR sequence is simulated the plot output can be disabled and SAR analyse functions can be used.

Usage

```
model_LuminescenceSignals(model, sequence, lab.doseRate = 1,
  simulate_sample_history = FALSE, plot = TRUE, verbose = TRUE, ...)
```

Arguments

`model`	`character` (**required**): set model to be used
`sequence`	`list` (**required**): set sequence to model as list or as *.seq file from the Riso sequence editor. To simulate SAR measurements there is an extra option to set the sequence list (cf. example 3): (**required**): RegDose: `numeric`, TestDose: `numeric`, PH: `numeric`, CH: `numeric`, OSL_temp: `numeric`. With default are: DoseRate: `numeric`, Irr_temp: `numeric`, optical_power: `numeric`, OSL_duration: `numeric`, PH_duration: `numeric`
`lab.doseRate`	`numeric` (with default): laboratory dose rate in XXX Gy/s for calculating seconds into Gray in the *.seq file.
`simulate_sample_history`	`logical` (with default): FALSE (with default): simulation begins at labour conditions, TRUE: simulations begins at crystallization (all levels 0) process
`plot`	`logical` (with default): Enables or disables plot output
`verbose`	`logical` (with default): Verbose mode on/off
`...`	further arguments and graphical parameters passed to `plot.default`. See details for further information

Details

Defining a **sequence**

Abrivation	Description	Arguments
TL	thermally stimulated luminescence	'temp begin', 'temp end', 'heating rate'
OSL	optically stimulated luminescence	'temp', 'duration','optical power'
LM_OSL	linear modulated OSL	'temp', 'duration'

RL/RF	radioluminescence	'temp','dose', 'doserate'
IRR	irradiation	'temp','dose', 'doserate'
CH	cutheat	'temp', 'duration'
PH	preheat	'temp', 'duration'
PAUSE	pause	'temp', 'duration'

Value

This function returns an RLum.Analysis object with all TL, (LM-) OSL and RF/RL steps in the sequence. Every entry is a RLum.Data.Curve obejct and can be plotted, analysed etc. with further RLum functions.

Function version

0.1.0

Note

This function can do just nothing at the moment.

Author(s)

Johannes Friedrich, University of Bayreuth (Germany), Sebastian Kreutzer, IRAMAT-CRP2A, Universite Bordeaux Montaigne (France)

References

Bailey, R.M., 2001. Towards a general kinetic model for optically and thermally stimulated luminescence of quartz. Radiation Measurements 33, 17-45.

Bailey, R.M., 2002. Simulations of variability in the luminescence characteristics of natural quartz and its implications for estimates of absorbed dose. Radiation Protection Dosimetry 100, 33-38.

Bailey, R.M., 2004. Paper I-simulation of dose absorption in quartz over geological timescales and it simplications for the precision and accuracy of optical dating. Radiation Measurements 38, 299-310.

Pagonis, V., Chen, R., Wintle, A.G., 2007: Modelling thermal transfer in optically stimulated luminescence of quartz. Journal of Physics D: Applied Physics 40, 998-1006.

Pagonis, V., Wintle, A.G., Chen, R., Wang, X.L., 2008. A theoretical model for a new dating protocol for quartz based on thermally transferred OSL (TT-OSL). Radiation Measurements 43, 704-708.

See Also

plot

Examples

```
## Not run:
##============================================================##
## Example 1: Simulate sample history of Bailey2001
## (cf. Bailey, 2001, Fig. 1)
##===========================================================##
```

```
##set sequence with the following steps
## (1) Irraditation at 20 deg. C with a dose of 1000 Gy and a dose rate of 1 Gy/s
## (2) Preheat to 350 deg. C and hold for 10 s
## (3) Illumination at 200 deg. C. for 100 s with 100 % optical power
## (4) Irradiation at 220 deg. C with a dose of 20 Gy and a dose rate of 0.01 Gy/s
## (5) Irradiation at 20 deg. C with a dose of 10 Gy and a dose rate of 1 Gy/s
## (6) TL from 20-400 deg. C with a rate of 5 K/s
sequence <-
  list(
   IRR = c(20, 1000, 1),
    PH = c(350, 10),
    ILL = c(200, 100, 100),
    IRR = c(220, 20, 0.01),
    IRR = c(20, 10, 1),
    TL = c(20, 400, 5)
  )

##model sequence
model.output <- model_LuminescenceSignals(
  sequence = sequence,
  model = "Bailey2001",
  simulate_sample_history = TRUE
)


##============================================================================##
## Example 2: Simulate sequence at labour without sample history
##============================================================================##

##set sequence with the following steps
## (1) Irraditation at 30 deg. C with a dose of 100 Gy and a dose rate of 1 Gy/s
## (2) Preheat to 2000 deg. C and hold for 10 s
## (3) LM-OSL at 125 deg. C. for 1000 s
## (4) OSL at 20 deg. C for 100 s with 90 % optical power
## (5) Cutheat at 220 deg. C
## (6) Irradiation at 20 deg. C with a dose of 10 Gy and a dose rate of 1 Gy/s
## (7) Pause at 200 deg. C for 100 s
## (8) TL from 20-400 deg. C with a heat rate of 5 K/s
## (9) Radiolumiescence at 20 deg. C with a dose of 20 Gy and a dose rate of 1 Gy/s

sequence <-
 list(
   IRR = c(20, 100, 1),
   PH = c(200, 10),
   LM_OSL = c(125, 100),
   CH = c(200),
   IRR = c(20, 10, 1),
   PAUSE = c(200, 100),
   OSL = c(125, 100, 90),
   PAUSE = c(200, 100),
   TL = c(20, 400, 5),
   RF = c(20, 200, 0.01)
)

# call function "model_LuminescenceSignals", set sequence = sequence, model = "Pagonis2008" (palaeodose = 200
# and simulate_sample_history = FALSE (default), because the sample history is not part of the sequence
```

```
model.output <- model_LuminescenceSignals(
   sequence = sequence,
   model = "Pagonis2008"
   )



##============================================================================##
## Example 3: Simulate SAR sequence
##============================================================================##

##set SAR sequence with the following steps
## (1) RegDose: set regenerative dose [Gy] as vector
## (2) TestDose: set test dose [Gy]
## (3) PH: set preheat temperature in deg. C
## (4) CH: Set cutheat temperature in deg. C
## (5) OSL_temp: set OSL reading temperature in deg. C
## (6) OSL_duration: set OSL reading duration in s

sequence <- list(
 RegDose = c(0,10,20,50,90,0,10),
 TestDose = 5,
 PH = 240,
 CH = 200,
 OSL_temp = 125,
 OSL_duration = 70)

# call function "model_LuminescenceSignals", set sequence = sequence, model = "Pagonis2007" (palaeodose = 20 (
# and simulate_sample_history = FALSE (default), because the sample history is not part of the sequence

 model.output <- model_LuminescenceSignals(

 sequence = sequence,
 model = "Pagonis2007",
 plot = FALSE
 )

# in environment is a new object "model.output" with the results of every step of the given sequence.
# Plots are done at OSL and TL steps and the growth curve

# call "analyse_SAR.CWOSL" from RLum package
 results <- analyse_SAR.CWOSL(model.output,
                            signal.integral.min = 1,
                            signal.integral.max = 15,
                            background.integral.min = 601,
                            background.integral.max = 701,
                            fit.method = "EXP",
                            dose.points = c(0,10,20,50,90,0,10))


##============================================================================##
## Example 4: generate sequence from *.seq file and run SAR simulation
##============================================================================##

## call function "model_LuminescenceSignals", load *.seq file for sequence, set model = "Bailey2002" (palaeod
## and simulate_sample_history = FALSE (default), because the sample history is not part of the sequence
```

```
model.output <- model_LuminescenceSignals(
  sequence = "inst/extdata/sample_SAR_cycle.SEQ",
  model = "Bailey2002",
  plot = FALSE
)


## call RLum package function "analyse_SAR.CWOSL" to analyse the simulated SAR cycle

results <- analyse_SAR.CWOSL(model.output,
                             signal.integral.min = 1,
                             signal.integral.max = 10,
                             background.integral.min = 601,
                             background.integral.max = 701,
                             dose.points = c(0,5,10,20,50,5,0),
                             fit.method = "EXP")

print(get_RLum(results))


##============================================================================##
## Example 5: compare different optical powers of stimulation light
##============================================================================##

## call function "model_LuminescenceSignals", model = "Bailey2004"
## and simulate_sample_history = FALSE (default), because the sample history is not part of the sequence
## the optical_power of the LED is varied and then compared.

optical_power <- seq(from = 0,to = 100,by = 20)

model.output <- lapply(1:length(optical_power), function(x){

sequence <- list(IRR = c(20, 50, 1),
                 PH = c(220, 10, 5),
                 OSL = c(125, 50, optical_power[x])
                 )

return(model_LuminescenceSignals(
       sequence = sequence,
       model = "Bailey2004",
       plot = FALSE
       ))
})

##combine output curves
model.output.merged <- merge_RLum(model.output)

##plot
plot_RLum(
 object = model.output.merged,
 xlab = "Illumination time [s]",
 ylab = "OSL signal [a.u.]",
 main = "OSL signal dependency on optical power of stimulation light",
 legend.text = paste("Optical power density", 20*optical_power/100, "mW/cm^2"),
 combine = TRUE)
```

```
## End(Not run)
```

Index

∗Topic **package**
RLumModel-package, 2

character, *3*

list, *3*
logical, *3*

model_LuminescenceSignals, 3

numeric, *3*

plot, *4*
plot.default, *3*

RLumModel (RLumModel-package), 2
RLumModel-package, 2