Package 'RLumCarlo'

November 4, 2019

Type Package

Title Monte-Carlo Methods for Simulating Luminescence Phenomena

Version 0.1.0.9000-122

Date 2019-11-04

Author Johannes Friedrich [aut, trl] (<https://orcid.org/0000-0002-0805-9547>),
Sebastian Kreutzer [aut, trl, cre] (<https://orcid.org/0000-0002-0734-2199>),
Vasilis Pagonis [aut] (<https://orcid.org/0000-0002-4852-9312>),
Christoph Schmidt [aut] (<https://orcid.org/0000-0002-2309-3209>),
Ena Rajovic [ctb],
Alex Roy Duncan [ctb],
Christian Laag [ctb] (<https://orcid.org/0000-0002-6012-1029>)

Maintainer Sebastian Kreutzer <sebastian.kreutzer@u-bordeaux-montaigne.fr>

Description A collection of functions to simulate luminescence production in minerals using Monte-Carlo methods.
Implemented are models for delocalised, localised and tunnelling transitions.
Supported stimulation methods are TL, CW-OSL, LM-OSL, LM-IRSL, and ITL (ISO-TL).

Contact Package Developer Team <sebastian.kreutzer@u-bordeaux-montaigne.fr>

License GPL-3

BugReports https://github.com/R-Lum/RLumCarlo/issues

Depends R (>= 3.3.0),
utils,
magrittr

URL https://CRAN.R-project.org/package=RLumCarlo

LinkingTo Rcpp (>= 1.0.2),
RcppArmadillo (>= 0.9.700.2.0)

Imports abind (>= 1.4-5),
doParallel (>= 1.0.15),
foreach (>= 1.4.7),
khroma (>= 1.2.0),
parallel,
methods,
Rcpp (>= 1.0.2)

Suggests R.rsp (>= 0.43.1),
testthat (>= 2.0.0)

Encoding UTF-8

VignetteBuilder R.rsp

RoxygenNote 6.1.1

R topics documented:

RLumCarlo-package 2
plot_RLumCarlo 3
run_MC_CW_IRSL_LOC 4
run_MC_CW_IRSL_TUN 6
run_MC_CW_OSL_DELOC 8
run_MC_ISO_DELOC 10
run_MC_ISO_LOC 12
run_MC_ISO_TUN 13
run_MC_LM_OSL_DELOC 15
run_MC_LM_OSL_LOC 17
run_MC_LM_OSL_TUN 19
run_MC_TL_DELOC 21
run_MC_TL_LOC 23
run_MC_TL_TUN 25

Index **28**

`RLumCarlo-package` *Monte-Carlo Methods for Simulating Luminescence Phenomena*

Description

A collection of functions to simulate luminescence production in dosimeters using Monte-Carlo methods. Implemented are models for delocalised, localised and tunnelling transitions. Supported stimulation modes are TL, CW-OSL, LM-OSL, LM-IRSL, and ITL (ISO-TL).

Details

Funding

The development of RLumCarlo benefitted from the support by various funding bodies:

- The initial work by Johannes Friedrich, Sebastian Kreutzer and Christoph Schmidt was supported by the Deutsche Forschungsgemmeinschaft (DFG, 2015–2018, SCHM 3051/4-1, "Modelling quartz luminescence signal dynamics relevant for dating and dosimetry", SCHM 3051/4-1).
- Later work (2018-2019) was secured through the project "ULTIMO: Unifying Luminescence Models of quartz and feldspar DAAD: Deutscher Akademischer Austauschdienst (German Academic Exchange Service). Framework: DAAD PPP USA 2018, ID: 57387041.
- The work of Sebastian Kreutzer as maintainer of the package was supported by LabEx LaScArBx (ANR - n. ANR-10-LABX-52) between 2017 and 2019.

Author(s)

Johannes Friedrich (University of Bayreuth, Germany),
Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS-Université Bordeaux Montaigne (France),
Vasilis Pagonis, McDaniel College Westminster (MD, USA),
Christoph Schmidt, University of Bayreuth (Germany),
Ena Rajovic, University of Bayreuth (Germany),
Alex Roy Duncan, University of Bayreuth (Germany),
Christian Laag, University of Bayreuth (Germany)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R., Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects_ A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

`plot_RLumCarlo` *Plot RLumCarlo Monte-Carlo Simulations Results*

Description

Visualise 'RLumCarlo' modelling results without extracting the values manually. Typically visualised values are the averaged signal or the number of remaining electrons with a polygon indicating modelling uncertainties.

Usage

```
plot_RLumCarlo(object, plot_uncertainty = "range", norm = FALSE,
  add = FALSE, ...)
```

Arguments

`object`	list of class `RLumCarlo_Model_Output` (**required**): input object to be plotted, usually the required input object is generated by one for the functions preceeding with `run_`. Alternatively a list of such objects can be provided.
`plot_uncertainty`	logical (*with default*): sets the nature of the show uncertainty. Allowed values are `range`, `sd` (standard deviation) and `var` (variance). `NULL` disables the uncertainty visualisation.
`norm`	logical (*with default*): normalise curve to the highest intensity value
`add`	logical (*with default*): allows overplotting of results by adding curve to an existing plot. This argument is handled automatically if `object` is of type list
`...`	further arguments that can be passed to control the plot output largely following the argument names in graphics::plot.default. Currently supported are: `xlab`, `xlim`, `ylim`, `main`, `lwd`, `type`, `pch`, `lty`,`col`, `grid`, `legend`. The arguments `lwd`, `type`, `pch`, `lty`, `col` can be provided as a vector if `object` is a list

Details

For colouring the curves, the package khroma::khroma-package is used to provide colours that can be best distinguished, in particular by colour-blind users.

Value

This function returns a graphical output

Function version

0.1.0

How to cite

Friedrich, J., Kreutzer, S., 2019. plot_RLumCarlo(): Plot RLumCarlo Monte-Carlo Simulations Results. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Johannes Friedrich, University of Bayreuth (Germany), Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, Université Bordeaux Montaigne (France)

---

run_MC_CW_IRSL_LOC *Monte-Carlo Simulation for CW-IRSL for Localized Transition*

Description

Runs a Monte-Carlo (MC) simulation of constant-wave infrared stimulated luminesence (CW-IRSL) using the generalized one trap (GOT) model. Localized refers to an excited state that is shared by the electron and the recombination centre, so that the conduction band is not involved in the recombination process.

Usage

```
run_MC_CW_IRSL_LOC(A, times, clusters = 10, n_filled = 100, r,
  method = "par", output = "signal", ...)
```

Arguments

A	numeric (**required**): The optical excitation rate from trap to the excited state (s^-1).
times	numeric (*with default*): The sequence of time steps within the simulation (s).
clusters	numeric (*with default*): The number of MC runs (unitless).
n_filled	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
r	numeric (*with default*): The localized retrapping ratio (unitless).
method	character (*with default*): sequential `'seq'` or parallel `'par'`processing
output	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges/electrons, in the trap)
...	further arguments

Details

$$I_{LOC}(t) = -dn/dt = A * (n^2/(r+n))$$

where in the function:
A := optical excitation rate from trap to the excited state (s^-1)
r := localised retrapping ratio (unitless)
t := time (s)
n := number of filled electron traps

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_CW_IRSL_LOC(): Monte-Carlo Simulation for CW-IRSL for Localized Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading

Chen, R., McKeever, S.W.S., 1997. Theory of Thermoluminescence and Related Phenomena. WORLD SCIENTIFIC. doi: 10.1142/2781

Examples

```
run_MC_CW_IRSL_LOC(
 A = 0.12,
 times = 0:100,
 clusters = 50,
 n_filled = 1,
 r = 1e-7,
 method = "seq",
 output = "signal"
) %>%
plot_RLumCarlo(legend = TRUE)
```

`run_MC_CW_IRSL_TUN` *Run Monte-Carlo Simulation for CW-IRSL for Tunneling Transition*

Description

Runs a Monte-Carlo (MC) simulation of constant wave infrared stimulated luminesence (CW-IRSL) using the model for tunneling transiation. Tunneling refers to the direct movement of electrons from the excited state of a trap directly to the recombination center.

Usage

```
run_MC_CW_IRSL_TUN(A, rho, times, clusters = 10, r_c = 0,
  delta.r = 0.1, N_e = 200, method = "seq", output = "signal", ...)
```

Arguments

`A`	numeric (**required**): The optical excitation rate from ground state of trap to excited state of trap (s^-1).
`rho`	numeric (**required**): The density of recombination centers (defined as rho' in Huntley 2006) (unitless).
`times`	numeric (*with default*): The sequence of time steps within the simulation (s).
`clusters`	numeric (*with default*): The number of MC runs (unitless).
`r_c`	numeric (*with default*): Critical distance (>0) that is to be inserted if the sample has 1 been thermally and/or optically pretreated, so that the electron-hole pairs within r_c have already recombined
`delta.r`	numeric (*with default*): Increments of the unitless distance parameter r
`N_e`	numeric (*width default*): The total number of electron traps available (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$p(t) = A * e^{(-r'/\rho^{(-1/3)})}$$

$$I_{TUN}(t) = 3 * n * p(t) * (r')^2 * e^{(-r'^3)}$$

Where in the function:
p(t) := The experimental stimulation mode
r' := the unitless tunneling radius
ρ' := rho the unitless density of recombination centres
t := time (s)
n := The Instantaneous number of electrons

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.2.0

How to cite

Friedrich, J., Kreutzer, S., 2019. run_MC_CW_IRSL_TUN(): Run Monte-Carlo Simulation for CW-IRSL for Tunneling Transition. Function version 0.2.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Johannes Friedrich, University of Bayreuth (Germany), Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, Université Bordeaux Montaigne (France)

References

Huntley, D.J., 2006. An explanation of the power-law decay of luminescence. Journal of Physics: Condensed Matter, 18(4), 1359.doi: 10.1088/09538984/18/4/020

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading

Aitken, M.J., 1985. Thermoluminescence dating. Academic Press.

Chen, R., McKeever, S.W.S., 1997. Theory of Thermoluminescence and Related Phenomena. WORLD SCIENTIFIC. doi: 10.1142/2781

Examples

```
run_MC_CW_IRSL_TUN(
 A = 0.8,
 rho = 1e-4,
 times = 0:10,
 r_c = 0.05,
 delta.r = 1e-2,
 method = "seq",
 clusters = 2,
  output = "signal") %>%
 plot_RLumCarlo(norm = TRUE, legend = TRUE)
```

`run_MC_CW_OSL_DELOC` *Run Monte-Carlo Simulation for CW-OSL for Delocalized Transition*

Description

Runs a Monte-Carlo (MC) simulation of constant wave optically stimulated luminesence (CW-OSL) using the one trap one recombination center (OTOR) model. The term delocalized here refers to the involvement of the conduction band.

Usage

```
run_MC_CW_OSL_DELOC(A, times, clusters = 10, N_e = 200,
  n_filled = N_e, R, method = "par", output = "signal", ...)
```

Arguments

`A`	numeric (**required**): The optical excitation rate from trap to conduction band (s^-1).
`times`	numeric (*with default*): The sequence of temperature steps within the simulation (s)
`clusters`	numeric (*with default*): The number of MC runs (unitless).
`N_e`	integer (*with default*): The total number of electron traps available (unitless).
`n_filled`	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
`R`	numeric (*with default*): The delocalized retrapping ratio (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$I_{DELOC}(t) = -dn/dt = p(t) * (n^2/(N * R + n(1 - R)))$$

Where in the function:
t := Time
p(t) := The experimental stimulation mode
n := The instantaneous number of electrons
N := `N_e` the available number of electron traps available
R := Delocalised retrapping ratio

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_CW_OSL_DELOC(): Run Monte-Carlo Simulation for CW-OSL for Delocalized Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading

Chen, R., McKeever, S.W.S., 1997. Theory of Thermoluminescence and Related Phenomena. WORLD SCIENTIFIC. doi: 10.1142/2781

Examples

```
run_MC_CW_OSL_DELOC(
 A = 0.12,
 R = 0.1,
 times = 0:10,
 clusters = 10,
 method = "seq") %>%
plot_RLumCarlo(legend = TRUE)

##================##
## A long example
##================##
## Not run:

A <- c(0.1,0.3,0.5,1)
times <- seq(0, 60, 1)
s <- 1e12
E <- 1
R <- c(1e-7, 1e-6, 0.01, 0.1) # sequence of different R values
clusters <- 1000 # number of Monte Carlo simulations
N_e <- c(200, 500, 700, 400) # number of free electrons
n_filled <- c(200, 500, 100, 70) # number of filled traps
method <-"par"
output <- "signal"
col <- c(1,2,3,4) # ifferent colours for the individual curves
plot_uncertainty <- c(TRUE,FALSE,TRUE,FALSE)  # do you want to see the uncertainty?
add_TF <- c(FALSE,rep(TRUE, (length(R)-1)))

#loop to plot different curves into one plotfor
for (u in 1:length(R)){
 results <- run_MC_CW_OSL_DELOC(
  A = A[u],
  times,
  clusters = clusters,
```

```
  N_e = N_e[u],
  n_filled = n_filled[u],
  R = R[u],
  method = method,
  output = output)

plot_RLumCarlo(
results,
 add = add_TF[u],
 legend = FALSE,
 col = col[u],
 main = "Delocalised Transition")
}
# add your legend with your parameters
legend("topright",
  ncol = 4,
  cex = 0.55,
  title = "parameters",
  legend=c(
   paste0("A = ", A),
   paste0("n_filled = ", n_filled),
   paste0("N_e = ", N_e),
   paste0("R = ", R)),
   bty = "n",
   text.col = col)

## End(Not run)
```

`run_MC_ISO_DELOC` *Run Monte-Carlo Simulation for ISO-TL for Delocalized transition*

Description

Runs a Monte-Carlo (MC) simulation of isothermally stimulated luminesence (ISO-TL or ITL) using the one trap one recombination center (OTOR) model. Delocalized refers to involvement of the conduction band.

Usage

```
run_MC_ISO_DELOC(s, E, T = 20, times, clusters = 10, N_e = 200,
  n_filled = N_e, R, method = "par", output = "signal", ...)
```

Arguments

`s`	numeric (**required**): The frequency factor of the trap (s^-1).
`E`	numeric (**required**): Thermal activation energy of the trap (eV).
`T`	numeric (*with default*): Constant stimulation temperature (degrees C).
`times`	numeric (*with default*): the sequence of temperature steps within the simulation (s).
`clusters`	numeric (*with default*): The number of MC runs (unitless).
`N_e`	integer (*with default*): The total number of electron traps available (unitless).

`n_filled`	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
`R`	numeric (*with default*): The delocalized retrapping ratio (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$I_{DELOC}(t) = -dn/dt = (s * e^{-}E/k_b * T_{ISO}) * (n^2/(N * R + n(1 - R)))$$

Where in the function:
`t` := Time
`e`:= Exponentional function
k_B := Boltzmann constant
T_{ISO} = Temperature
n := `n_filled` the number of filled electron traps at the beginning of the simulation
N := `N_e` is the total number of electron traps available (unitless)
R := R the delocalised retrapping ratio (unitless)

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_ISO_DELOC(): Run Monte-Carlo Simulation for ISO-TL for Delocalized transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading

Chen, R., McKeever, S.W.S., 1997. Theory of Thermoluminescence and Related Phenomena. WORLD SCIENTIFIC. doi: 10.1142/2781

Examples

```
run_MC_ISO_DELOC(
 s = 3.5e12,
 E = 1.45,
 T = 200,
 R = 1,
 method = 'seq',
 times = 0:100) %>%
plot_RLumCarlo(legend = TRUE)
```

`run_MC_ISO_LOC` *Run Monte-Carlo simulation for ISO-TL for Localized Transition*

Description

Runs a Monte-Carlo (MC) simulation of isothermally stimulated luminesence (ISO-TL or ITL) using the genralized one trap (GOT) model. Localized refers to excitation of an electron before it recombines, but without the involvement of the conduction band.

Usage

```
run_MC_ISO_LOC(s, E, T = 20, times, clusters = 10, n_filled = 100, r,
  method = "par", output = "signal", ...)
```

Arguments

`s`	numeric (**required**): The frequency factor of the trap (s^-1).
`E`	numeric (**required**): Thermal activation energy of the trap (eV).
`T`	numeric (*with default*): Constant stimulation temperature (degrees C).
`times`	numeric (*with default*): The sequence of temperature steps within the simulation (s).
`clusters`	numeric (*with default*): The number of MC runs (unitless).
`n_filled`	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
`r`	numeric (*with default*): the localized retrapping ratio (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$I_{LOC}(t) = -dn/dt = (s * e^{-}E/k_b * T_ISO) * (n^2/(r+n)))$$

Where in the function:
`t` := Time (s)
k_B := `Boltzmann constant`

T_{ISO} = Isothermal temperature
n := n_filled
r := the localized retrapping ratio (unitless)

Value

This function returns an object of class RLumCarlo_Model_Output which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_ISO_LOC(): Run Monte-Carlo simulation for ISO-TL for Localized Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Examples

```
run_MC_ISO_LOC(
 E = 1.45,
 s = 3.5e12,
 T = 200,
 times = 0:100,
 method = 'seq',
 r = 1) %>%
plot_RLumCarlo(legend = TRUE)
```

---

run_MC_ISO_TUN *Monte-Carlo Simulation for Isothermal-TL for Tunneling Transition*

Description

Runs a Monte-Carlo (MC) simulation of isothermally stimulated luminesence (ISO-TL or ITL) using the tunneling (TUN) model. Tunneling refers to the direct transition of electrons from an excited state directly into the recombination center without involving the conduction band.

Usage

```
run_MC_ISO_TUN(E, s, T = 200, rho, times, clusters = 10, r_c = 0,
  delta.r = 0.1, N_e = 200L, method = "par", output = "signal",
  ...)
```

Arguments

E	numeric (**required**): Thermal activation energy of the trap (eV).
s	numeric (**required**): Frequency factor of the trap (s^-1).
T	numeric (*with default*): Constant stimulation temperature (degrees C).
rho	numeric (**required**): The density of recombination centres (defined as rho' in Huntley 2006) (unitless).
times	numeric (*with default*): The sequence of temperature steps within the simulation (s).
clusters	numeric (*with default*): The number of MC runs (unitless).
r_c	numeric (*with default*): The radius of tunneling (dimensionless)
delta.r	numeric (*with default*): Fractional change of the dimensionless distance of nearest recombination centres (r', which is preset at 2)
N_e	numeric (*width default*): The total number of electron traps available (unitless).
method	character (*with default*): sequential `'seq'` or parallel processing `'par'`
output	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
...	further arguments

Details

** Model description **

$$p(t) = s * e^{(} - E/k_B * T) * e^{(-r'/\rho'^{1/3})}$$

$$I_{TUN}(t) = 3 * n * p(t) * (r')^{2*e^{(-r'^3)}}$$

Where in the function:
`p(t)` := The experimental stimulation mode
k_B := Boltzmann constant
`r` := `r`
ρ := `rho`
`t` := `Time`
`n` := The Instantaneous number of electrons
`n` := `n_filled`
`t`:= `times`

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Friedrich, J., Kreutzer, S., 2019. run_MC_ISO_TUN(): Monte-Carlo Simulation for Isothermal-TL for Tunneling Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Johannes Friedrich, University of Bayreuth (Germany), Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V. and Kulp, C., 2017. Monte Carlo simulations of tunneling phenomena and nearest neighbor hopping mechanism in feldspars. Journal of Luminescence 181, 114–120. doi: 10.1016/j.jlumin.2016.09.014

Further reading Aitken, M.J., 1985. Thermoluminescence dating. Academic Press.

Huntley, D.J., 2006. An explanation of the power-law decay of luminescence. Journal of Physics: Condensed Matter, 18(4), 1359.doi: 10.1088/09538984/18/4/020

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Examples

```
run_MC_ISO_TUN(
 E = .8,
 s = 1e16,
 T = 50,
 rho = 1e-4,
 times = 0:100,
 clusters = 10,
 N_e = 2,
 r_c = 1e-4,
 delta.r = 0.5,
 method = "seq") %>%
 plot_RLumCarlo(legend = TRUE)
```

---

run_MC_LM_OSL_DELOC *Run Monte-Carlo Simulation for LM-OSL for Delocalized Transition*

Description

Runs a Monte-Carlo (MC) simulation of linearly modulated optically stimulated luminesence (LM-OSL) using the one trap one recombination center (OTOR) model. Delocalized refers to involvement of the conduction band.

Usage

```
run_MC_LM_OSL_DELOC(A, times, clusters = 10, N_e = 200,
  n_filled = N_e, R, method = "par", output = "signal", ...)
```

Arguments

A	numeric (**required**): The optical excitation rate from trap to conduction band (s^-1).
times	numeric (*with default*): The sequence of temperature steps within the simulation (s).
clusters	numeric (*with default*): The number of MC runs (unitless).
N_e	integer (*with default*): The total number of electron traps available (unitless).
n_filled	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
R	numeric (*with default*): The delocalized retrapping ratio (unitless).
method	character (*with default*): sequential `'seq'` or parallel processing `'par'`
output	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
...	further arguments

Details

$$I_{DELOC}(t) = -dn/dt = p(t) * (n^2/(N * R + n(1 - R)))$$

Where in the function:
t := Time (s)
p(t) := The experimental stimulation mode
n := The Instantaneous number of electrons
R := delocalized retrapping ratio (unitless)
N := `N_e` total number of electron traps available (unitless)

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_LM_OSL_DELOC(): Run Monte-Carlo Simulation for LM-OSL for Delocalized Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading

Chen, R., McKeever, S.W.S., 1997. Theory of Thermoluminescence and Related Phenomena. WORLD SCIENTIFIC. doi: 10.1142/2781

Examples

```
run_MC_LM_OSL_DELOC(
 A = 0.12,
 R = 0.1,
 times = 0:50,
 method = "seq",
 clusters = 10) %>%
plot_RLumCarlo(legend = TRUE)
```

`run_MC_LM_OSL_LOC` *Run Monte-Carlo Simulation for LM-OSL for Localized Transition*

Description

Runs a Monte-Carlo (MC) simulation of linearly modulated optically stimulated luminesence (LM-OSL) using the generalized one trap (GOT) model. Localized refers to excitation of an electron before it recombines, but without the involvement of the conduction band.

Usage

```
run_MC_LM_OSL_LOC(A, times, clusters = 10, n_filled = 100, r,
  method = "par", output = "signal", ...)
```

Arguments

`A`	numeric (**required**): The optical excitation rate from trap to conduction band (s^-1).
`times`	numeric (*with default*): The sequence of temperature steps within the simulation (s).
`clusters`	numeric (*with default*): The number of MC runs (unitless).
`n_filled`	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
`r`	numeric (*with default*): The localized retrapping ratio (unitless)
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$I_{LOC}(t) = -dn/dt = p(t) * (n^2/(r+n))$$

Where in the function:
t := Time (s)
p(t) := The experimental stimulation mode
n := The Instantaneous number of electrons
r := localized retrapping ratio (unitless)

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_LM_OSL_LOC(): Run Monte-Carlo Simulation for LM-OSL for Localized Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Examples

```
run_MC_TL_LOC(
 s = 1e8,
 E = 0.5,
 times = 0:40,
 clusters = 10,
 n_filled = 10,
 r = 1e-7,
 method = "seq",
 output = "signal") %>%
plot_RLumCarlo(legend = TRUE)
```

`run_MC_LM_OSL_TUN` *Run Monte-Carlo Simulation for LM-OSL for Tunneling Transition*

Description

Runs a Monte-Carlo (MC) simulation of linearly modulated optically stimulated luminesence (LM-OSL) using the tunneling (TUN) model. Tunneling refers to the direct movement of electrons from a trap directly to the recombination center

Usage

```
run_MC_LM_OSL_TUN(A, rho, times, clusters = 10, r_c = 0,
  delta.r = 0.1, N_e = 200, method = "par", output = "signal", ...)
```

Arguments

`A`	numeric (**required**): The optical excitation rate from ground state of trap to excited state of trap (s^-1).
`rho`	numeric (**required**): The density of recombination centers (defined as rho' in Huntley 2006) (unitless).
`times`	vector (*with default*): The sequence of temperature steps within the simulation (s).
`clusters`	numeric (*with default*): The number of clusters.
`r_c`	numeric (*with default*): Critical distance (>0) that is to be inserted if the sample has 1 been thermally and/or optically pretreated, so that the electron-hole pairs within `r_c` have already recombined
`delta.r`	numeric (*with default*): Increments of r_c (unitless).
`N_e`	numeric (*width default*): The total number of electron traps available (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$p(t) = A * (t/P) * e^{(-r'/\rho'^{(-1/3)})}$$

$$I_{TUN}(t) = 3 * n * p(t) * (r')^2 * e^{(-r'^3)}$$

Where in the function:
p(t) := The experimental stimulation mode
t := `Time`
P := Maximum stimulation time
r' := `r`
ρ' := rho
n := The instantaneous number of electrons

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Friedrich, J., Kreutzer, S., 2019. run_MC_LM_OSL_TUN(): Run Monte-Carlo Simulation for LM-OSL for Tunneling Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Johannes Friedrich, University of Bayreuth (Germany), Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS-Université Bordeaux Montaigne (France)

References

Huntley, D.J., 2006. An explanation of the power-law decay of luminescence. Journal of Physics: Condensed Matter, 18(4), 1359.doi: 10.1088/09538984/18/4/020

Pagonis, V. and Kulp, C., 2017. Monte Carlo simulations of tunneling phenomena and nearest neighbor hopping mechanism in feldspars. Journal of Luminescence 181, 114–120. doi: 10.1016/j.jlumin.2016.09.014

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading Aitken, M.J., 1985. Thermoluminescence dating. Academic Press.

Examples

```
run_MC_LM_OSL_TUN(
 A = 1,
 rho = 1e-7,
 times = 0:10,
 clusters = 3,
 N_e = 2,
 r_c = 0.001,
 delta.r = 1e-1,
 method = "seq",
 output = "signal") %>%
plot_RLumCarlo(norm = TRUE)
```

`run_MC_TL_DELOC` *Run Monte-Carlo Simulation for TL for Delocalized Transition*

Description

Runs a Monte-Carlo (MC) simulation of thermo-luminesence (TL) using the one trap one recombination center (OTOR) model. Delocalized refers to involvement of the conduction band.

Usage

```
run_MC_TL_DELOC(s, E, times, clusters = 10, N_e = 200,
  n_filled = N_e, R, method = "par", output = "signal", ...)
```

Arguments

`s`	numeric (**required**): The frequency factor of the trap (s^-1).
`E`	numeric (**required**): Thermal activation energy of the trap (eV).
`times`	numeric (*with default*): The sequence of temperature steps within the simulation (s).
`clusters`	numeric (*with default*): The number of MC runs (unitless).
`N_e`	integer (*with default*): The total number of electron traps available (unitless).
`n_filled`	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
`R`	numeric (*with default*): The delocalized retrapping ratio (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$I_{DELOC}(t) = -dn/dt = (s * e^{-}E/k_b * T) * (n^2/(N * R + n(1 - R))))$$

Where in the function:
`t` := `Time`
k_B := Boltzmann constant
`T`= Temperature
`n` := The Instantaneous number of electrons
`N` := `N_e`

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_TL_DELOC(): Run Monte-Carlo Simulation for TL for Delocalized Transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading

Chen, R., McKeever, S.W.S., 1997. Theory of Thermoluminescence and Related Phenomena. WORLD SCIENTIFIC. doi: 10.1142/2781

Examples

```
run_MC_TL_DELOC(
 s = 3.5e12,
 E = 1.45,
 R = 0.1,
 method = 'seq',
 clusters = 3,
 times = 150:350) %>%
plot_RLumCarlo(legend = TRUE)

## Not run:
#' ##============================================================================##
## Example 2: Plot multiple TL stimulation TL curves in R with varying params
##============================================================================##
# define your parameters
times <- seq(100, 450, 1)
s <- rep(3.5e12, 4)
E <- rep(1.45, 4)
R <- c(0.7e-6, 1e-6, 0.01, 0.1)
clusters <- 1000
N_e <- c(400, 500, 700, 400)
n_filled <- c(400, 500, 300, 70)
method <- "par"
output <- "signal"
col <- c(1, 2, 3, 4) # different colours for the individual curves
plot_uncertainty <- c(TRUE, TRUE, TRUE, TRUE)  # do you want to see the uncertainty?
add_TF <- c(FALSE, rep(TRUE, (length(R) - 1)))

# loop to plot different curves into one plot
for (u in 1:length(R)){
 results <- run_MC_TL_DELOC(
  times=times,
  s = s[u],
  E = E[u],
```

```
  clusters = clusters,
  N_e = N_e[u],
  n_filled = n_filled[u],
  R = R[u],
  method = method,
  output = output)

plot_RLumCarlo(
 results,
 add = add_TF[u],
 legend = FALSE,
 col=col[u],
 main = " your plot",
 ylim=c(0,20))
}
#add your legend with your parameters
legend("topright",
  ncol = 5,
  cex = 0.55,
  bty = "n",
  title = "parameters",
  legend = c(
   paste0("E = ", E),
   paste0("s = ", s),
   paste0("n_filled = ", n_filled),
   paste0("N_e = ", N_e), paste0("R = ", R)),
   text.col = col)

## End(Not run)
```

`run_MC_TL_LOC` *Run Monte-Carlo Simulation for TL for Localized transition*

Description

Runs a Monte-Carlo (MC) simulation of thermo-luminesence (TL) using the generalized one trap (GOT) model. Localized refers to excitation of an electron before it recombines, but without the involvement of the conduction band.

Usage

```
run_MC_TL_LOC(s, E, times, clusters = 10, n_filled = 100, r,
  method = "par", output = "signal", ...)
```

Arguments

`s`	numeric (**required**): The frequency factor of the trap (s^-1).
`E`	numeric (**required**): Thermal activation energy of the trap (eV).
`times`	numeric (*with default*): The sequence of temperature steps within the simulation (s).
`clusters`	numeric (*with default*): The number of MC run (unitless).

`n_filled`	integer (*with default*): The number of filled electron traps at the beginning of the simulation (unitless).
`r`	numeric (*with default*): The localized retrapping ratio (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap)
`...`	further arguments

Details

$$I_{LOC}(t) = -dn/dt = (s * e^{-}E/k_b * T) * (n^2/(r+n))$$

Where in the function:
`t` := `Time`
k_B := Boltzmann constant
T := Temperature
E := the trap depth (eV)
n := The Instantaneous number of electrons
r := the localised retrapping ratio (unitless)

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Kreutzer, S., 2019. run_MC_TL_LOC(): Run Monte-Carlo Simulation for TL for Localized transition. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, CNRS - Université Bordeaux Montaigne (France)

References

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., reutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Examples

```
run_MC_TL_LOC(
 s = 1e14,
 E = 0.9,
 times = 50:100,
 method = "seq",
 clusters = 2,
 r = 1e4) %>%
plot_RLumCarlo()
```

`run_MC_TL_TUN` *Run Monte-Carlo Simulation for TL for Tunnelling Transitions*

Description

Runs a Monte-Carlo (MC) simulation of thermo-luminesence (TL) caused by tunnelling (TUN) transitions. Tunneling refers to the direct recombination of electrons from a trap directly from the excited state of the trap, without involvement of the conduction band.

Usage

```
run_MC_TL_TUN(s, E, rho, r_c = 0, times, clusters = 10, N_e = 200,
  delta.r = 0.1, method = "par", output = "signal", ...)
```

Arguments

`s`	list (**required**): The frequency factor of the trap (s^-1).
`E`	numeric (**required**): Thermal activation energy of the trap (eV).
`rho`	numeric (**required**): The density of recombination centers (defined as rho' in Huntley 2006) (unitless).
`r_c`	numeric (*with default*): Critical distance (>0) that is to be inserted if the sample has been thermally and/or optically pretreated, so that the electron-hole pairs within 'r_c" have already recombined (unitless).
`times`	vector (*wih default*): The sequence of time steps within the simulation (s).
`clusters`	numeric (*with default*): The number of MC runs (unitless).
`N_e`	numeric (*with default*): The total number of electron traps available (unitless).
`delta.r`	numeric (*with default*): The increments of r_c (unitless).
`method`	character (*with default*): sequential `'seq'` or parallel processing `'par'`.
`output`	character (*with default*): output is either the `'signal'` (the default) or `'remaining_e'` (the remaining charges, electrons, in the trap).
`...`	further arguments

Details

$$p(t) = (s * exp(-E/(k_B * T))) * e^{(-r'/\rho'^{1/3})}$$

$$I_{TUN}(t) = 3 * n * p(t) * r'^{2*e^{(-r'^3)}}$$

Where in the function:
p(t) := The experimental stimulation mode
k_B := Boltzmann constant
T := Temperature
r' := r electron-hole distance (unitless)
ρ' := Density of recombination centers
t := Time
n := The instantaneous number of electrons

Value

This function returns an object of class `RLumCarlo_Model_Output` which is a list consisting of an array with dimension length(times) x length(r) x clusters and a numeric time vector.

Function version

0.1.0

How to cite

Friedrich, J., Kreutzer, S., 2019. run_MC_TL_TUN(): Run Monte-Carlo Simulation for TL for Tunnelling Transitions. Function version 0.1.0. In: Friedrich, J., Kreutzer, S., Pagonis, V., Schmidt, C., 2019. RLumCarlo: Monte-Carlo Methods for Simulating Luminescence PhenomenaR package version 0.1.0.9000-122.

Author(s)

Johannes Friedrich, University of Bayreuth (Germany), Sebastian Kreutzer, IRAMAT-CRP2A, UMR 5060, Université Bordeaux Montaigne (France)

References

Huntley, D.J., 2006. An explanation of the power-law decay of luminescence. Journal of Physics: Condensed Matter, 18(4), 1359.doi: 10.1088/09538984/18/4/020

Pagonis, V. and Kulp, C., 2017. Monte Carlo simulations of tunneling phenomena and nearest neighbor hopping mechanism in feldspars. Journal of Luminescence 181, 114–120. doi: 10.1016/j.jlumin.2016.09.014

Pagonis, V., Friedrich, J., Discher, M., Müller-Kirschbaum, A., Schlosser, V., Kreutzer, S., Chen, R. and Schmidt, C., 2019. Excited state luminescence signals from a random distribution of defects: A new Monte Carlo simulation approach for feldspar. Journal of Luminescence 207, 266–272. doi: 10.1016/j.jlumin.2018.11.024

Further reading Aitken, M.J., 1985. Thermoluminescence dating. Academic Press.

Examples

```
run_MC_TL_TUN(
 s = 1e12,
 E = 0.9,
 rho = 1,
 r_c = 1,
 times = 80:120,
 clusters = 2,
 method = 'seq',
 delta.r = 1e-1) %>%
plot_RLumCarlo()
```

Index

∗ **data**
  run_MC_CW_IRSL_LOC, 4
  run_MC_CW_IRSL_TUN, 6
  run_MC_CW_OSL_DELOC, 8
  run_MC_ISO_DELOC, 10
  run_MC_ISO_LOC, 12
  run_MC_ISO_TUN, 13
  run_MC_LM_OSL_DELOC, 15
  run_MC_LM_OSL_LOC, 17
  run_MC_LM_OSL_TUN, 19
  run_MC_TL_DELOC, 21
  run_MC_TL_LOC, 23
  run_MC_TL_TUN, 25
∗ **hplot**
  plot_RLumCarlo, 3
∗ **models**
  run_MC_CW_IRSL_LOC, 4
  run_MC_CW_IRSL_TUN, 6
  run_MC_CW_OSL_DELOC, 8
  run_MC_ISO_DELOC, 10
  run_MC_ISO_LOC, 12
  run_MC_ISO_TUN, 13
  run_MC_LM_OSL_DELOC, 15
  run_MC_LM_OSL_LOC, 17
  run_MC_LM_OSL_TUN, 19
  run_MC_TL_DELOC, 21
  run_MC_TL_LOC, 23
  run_MC_TL_TUN, 25
∗ **package**
  RLumCarlo-package, 2

array, *5*, *7, 8*, *11*, *13, 14*, *16*, *18*, *20, 21*, *24*, *26*

character, *4*, *6*, *8*, *11, 12*, *14*, *16, 17*, *19*, *21*, *24, 25*

graphics::plot.default, *3*

integer, *4*, *8*, *10–12*, *16, 17*, *21*, *24*

khroma::khroma-package, *3*

list, *3*, *5*, *7, 8*, *11*, *13, 14*, *16*, *18*, *20, 21*, *24–26*
logical, *3*

numeric, *4–8*, *10–14*, *16–21*, *23–26*

plot_RLumCarlo, 3

RLumCarlo *(*RLumCarlo-package*)*, 2
RLumCarlo-package, 2
run_MC_CW_IRSL_LOC, 4
run_MC_CW_IRSL_TUN, 6
run_MC_CW_OSL_DELOC, 8
run_MC_ISO_DELOC, 10
run_MC_ISO_LOC, 12
run_MC_ISO_TUN, 13
run_MC_LM_OSL_DELOC, 15
run_MC_LM_OSL_LOC, 17
run_MC_LM_OSL_TUN, 19
run_MC_TL_DELOC, 21
run_MC_TL_LOC, 23
run_MC_TL_TUN, 25

vector, *19*, *25*