

Package ‘RLumCarlo’

November 23, 2019

Type Package

Title Monte-Carlo Methods for Simulating Luminescence Phenomena

Version 0.1.0.9000-130

Date 2019-11-23

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 dosimetric materials 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 7.0.1

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	14
run_MC_LM_OSL_DELOC	17
run_MC_LM_OSL_LOC	18
run_MC_LM_OSL_TUN	20
run_MC_TL_DELOC	23
run_MC_TL_LOC	25
run_MC_TL_TUN	27
Index	31

RLumCarlo-package	<i>Monte-Carlo Methods for Simulating Luminescence Phenomena.</i>
-------------------	---

Description

A collection of functions to simulate luminescence production in dosimetric materials 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 Forschungsgemeinschaft (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 LaS-cArBx (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](https://doi.org/10.1016/j.jlumin.2018.11.024)

plot_RLumCarlo

*Plot RLumCarlo Monte-Carlo Simulation 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 <code>RLumCarlo_Model_Output</code> (required): input object to be plotted, usually the required input object is generated by one of the functions starting with <code>run_</code> . Alternatively a list of such objects can be provided.
plot_uncertainty	logical (<i>with default</i>): type of the displayed uncertainty. Allowed values are <code>range</code> , <code>sd</code> (standard deviation) and <code>var</code> (variance). <code>NULL</code> disables the uncertainty visualisation.
norm	logical (<i>with default</i>): normalise curve to the highest intensity value
add	logical (<i>with default</i>): allows overplotting of results by adding curves 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: <code>xlab</code> , <code>ylab</code> , <code>xlim</code> , <code>ylim</code> , <code>main</code> , <code>lwd</code> , <code>type</code> , <code>pch</code> , <code>lty</code> , <code>col</code> , <code>grid</code> , <code>legend</code> . The arguments <code>lwd</code> , <code>type</code> , <code>pch</code> , <code>lty</code> , <code>col</code> 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

Author(s)

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

run_MC_CW_IRSL_LOC	<i>Monte-Carlo Simulation for CW-IRSL (localized transitions)</i>
--------------------	---

Description

Runs a Monte-Carlo (MC) simulation of continuous wave infrared stimulated luminescence (CW-IRSL) using the generalized one trap (GOT) model. Localized transitions refer to transitions which do not involve the conduction or valence band. These transitions take place between the ground state and an excited state of the trapped charge, and also involve an excited state of the recombination center.

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 the ground state of the 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 retrapping ratio for localized transitions

method	character (<i>with default</i>): Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (<i>with default</i>): output is either the 'signal' (the default) or 'remaining_e' (the remaining charges/electrons in the trap)
...	further arguments

Details

The model

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

where in the function:

A := optical excitation rate from the ground state into the excited state of the trapped charge (s⁻¹)

r := retrapping ratio for localized transitions

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 clusters` and a **numeric** time vector.

Function version

0.1.0

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](https://doi.org/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](https://doi.org/10.1142/2781)

Examples

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

run_MC_CW_IRSL_TUN	<i>Run Monte-Carlo Simulation for CW-IRSL (tunneling transitions)</i>
--------------------	---

Description

Runs a Monte-Carlo (MC) simulation of continuous wave infrared stimulated luminescence (CW-IRSL) using the model for tunneling transitions. Tunneling refers to quantum mechanical tunneling processes from the excited state of the trapped charge, into a 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 the ground state of trap to the excited state of trap (s^{-1}).
rho	numeric (required) : The density of recombination centers (defined as ρ' 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 must be provided if the sample has been thermally and/or optically pretreated. This parameter expresses the fact that electron-hole pairs within a critical radius r_c have already recombined.
delta.r	numeric (with default) : Increments of the unitless distance parameter r
N_e	numeric (with default) : The total number of electron traps available (unitless).
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges/electrons in the trap)
...	further arguments

Details

The model

$$I_{TUN}(r', t) = -dn/dt = A * \exp(-(\rho')^{-1/3} * r') * n(r', t)$$

Where in the function:

A := excitation rate from ground state of the trap to the excited state (1/s)

r' := the unitless tunneling radius

ρ' := rho' the unitless density of recombination centres (see Huntley (2006))

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

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](https://doi.org/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](https://doi.org/10.1016/j.jlumin.2018.11.024)

Further reading

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

Jain, M., Guralnik, B., Andersen, M.T., 2012. Stimulated luminescence emission from localized recombination in randomly distributed defects. *J. Phys.: Condens. Matter* 24, 385402. doi: [10.1088/09538984/24/38/385402](https://doi.org/10.1088/09538984/24/38/385402)

Chen, R., McKeever, S.W.S., 1997. Theory of Thermoluminescence and Related Phenomena. WORLD SCIENTIFIC. doi: [10.1142/2781](https://doi.org/10.1142/2781)

Examples

```
run_MC_CW_IRSL_TUN(
  A = 0.8,
  rho = 1e-4,
  times = 0:50,
  r_c = 0.05,
  delta.r = 0.1,
  method = "seq",
  clusters = 10,
```

```
output = "signal") %>%
plot_RLumCarlo(norm = TRUE, legend = TRUE)
```

run_MC_CW_OSL_DELOC	<i>Run Monte-Carlo Simulation for CW-OSL (delocalized transitions)</i>
---------------------	--

Description

Runs a Monte-Carlo (MC) simulation of continuous wave optically stimulated luminescence (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 retrapping ratio for delocalized transitions (unitless)
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : Output is either the 'signal' (the default) or 'remaining_e' (the remaining charges, electrons, in the trap)
...	further arguments

Details

The model

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

Where in the function:

t := time

A := the optical excitation rate from trap to conduction band

n := n_filled, the instantaneous number of electrons

N := N_e the available number of electron traps available

R := retrapping ratio for delocalized transitions

Value

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

Function version

0.1.0

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](https://doi.org/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](https://doi.org/10.1142/2781)

Examples

```
## brief example
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 (delocalized transitions)

Description

Runs a Monte-Carlo (MC) simulation of isothermally stimulated luminescence (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 time 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 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : Output is either the 'signal' (the default) or 'remaining_e' (the remaining charges, electrons, in the trap)
...	further arguments

Details

The model

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

Where in the function:

t := time

k_B := Boltzmann constant

T_{ISO} = temperature of the isothermal experiment

n := n_filled, the number of filled electron traps at the beginning of the simulation

E := the trap depth in eV

s := the frequency factor in (s^{-1})

N := N_e, the total number of electron traps available (unitless)

R := the retrapping ratio for delocalized transitions

Value

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

Function version

0.1.0

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](https://doi.org/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](https://doi.org/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 (localized transitions)

Description

Runs a Monte-Carlo (MC) simulation of isothermally stimulated luminescence (ISO-TL or ITL) using the generalized one trap (GOT) model. Localized transitions refer to transitions which do not involve the conduction or valence band. These transitions take place between the ground state and an excited state of the trapped charge, and also involve an excited state of the recombination center.

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 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 retrapping ratio for localized transitions.
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges/electrons in the trap)
...	further arguments

Details

The model

$$I_{LOC}(t) = -dn/dt = (s * \exp(-E/(k_B * T_{ISO}))) * (n^2/(r + n))$$

Where in the function:

t := time (s)

k_B := Boltzmann constant

T_{ISO} := isothermal temperature

n := n_filled

s := frequency factor of the trap (1/s)

E := activation energy of the trap (eV)

r := retrapping ratio for localized transitions

Value

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

Function version

0.1.0

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](https://doi.org/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 ISO-TL (tunneling transitions)

Description

Runs a Monte-Carlo (MC) simulation of isothermally stimulated luminescence (ISO-TL or ITL) using the tunneling (TUN) model. Tunneling refers to quantum mechanical tunneling processes from the excited state of the trapped charge, into the recombination center.

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) : The effective frequency factor for the tunneling process (s^{-1}).
T	numeric (with default) : Constant stimulation temperature (degrees C).
rho	numeric (required) : The dimensionless density of recombination centres (defined as ρ' 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 must be provided if the sample has been thermally and/or optically pretreated. This parameter expresses the fact that electron-hole pairs within a critical radius r_c have already recombined.
delta.r	numeric (with default) : Fractional change of the dimensionless distance of nearest recombination centres (r')
N_e	numeric (with default) : The total number of electron traps available (unitless).
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges/electrons in the trap)
...	further arguments

Details

The model

$$I_{TUN}(r', t) = -dn/dt = (s * \exp(-E/(k_B * T_{ISO}))) * \exp(-(\rho')^{-1/3} * r') * n(r', t)$$

Where in the function:

E := thermal activation energy (eV)

s := the effective frequency factor for the tunneling process (s^{-1})

T_{ISO} := the temperature of the isothermal experiment (degrees C)

k_B := Boltzmann constant

r' := the unitless tunneling radius

ρ := rho the unitless density of recombination centres see Huntley (2006)

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 $\text{length}(\text{times}) \times \text{length}(r) \times \text{clusters}$ and a **numeric** time vector.

Function version

0.1.0

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](https://doi.org/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](https://doi.org/10.1088/09538984/18/4/020)

Jain, M., Guralnik, B., Andersen, M.T., 2012. Stimulated luminescence emission from localized recombination in randomly distributed defects. *J. Phys.: Condens. Matter* 24, 385402. doi: [10.1088/09538984/24/38/385402](https://doi.org/10.1088/09538984/24/38/385402)

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](https://doi.org/10.1016/j.jlumin.2018.11.024)

Examples

```
## short example
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)

## Not run:
## long (meaningful) example
results <- run_MC_ISO_TUN(
  E = .8,
  s = 1e16,
  T = 50,
  rho = 1e-4,
  times = 0:100,
  clusters = 1000,
  N_e = 200,
  r_c = 1e-4,
  delta.r = 0.5,
  method = "par")

plot_RLumCarlo(results, legend = TRUE)

## End(Not run)
```

run_MC_LM_OSL_DELOC *Run Monte-Carlo Simulation for LM-OSL (delocalized transitions)*

Description

Runs a Monte-Carlo (MC) simulation of linearly modulated optically stimulated luminescence (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 time 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 retrapping ratio for delocalized transitions
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges/electrons in the trap)
...	further arguments

Details

The model

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

Where in the function:

t := time (s)

A := the optical excitation rate from trap to conduction band
 n := n_filled, the instantaneous number of electrons
 R := the retrapping ratio for delocalized transitions
 N := N_e, the total number of electron traps available (unitless)
 P := total stimulation time"

Value

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

Function version

0.1.0

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](https://doi.org/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](https://doi.org/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 (localized transitions)

Description

Runs a Monte-Carlo (MC) simulation of linearly modulated optically stimulated luminescence (LM-OSL) using the generalized one trap (GOT) model. Localized transitions refer to transitions which do not involve the conduction or valence band. These transitions take place between the ground state and an excited state of the trapped charge, and also involve an excited state of the recombination center.

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 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 retrapping ratio for localized transitions
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges, electrons, in the trap)
...	further arguments

Details**The model**

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

Where in the function:

A := optical excitation rate from the trap to the conduction band

P := total excitation time

t := time (s)

n := n_filled, the instantaneous number of electrons

r := the retrapping ratio for localized transitions

P := the total stimulation period (s)

Value

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

Function version

0.1.0

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](https://doi.org/10.1016/j.jlumin.2018.11.024)

Examples

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

## Not run:
## the long (meaningful) example
results <- run_MC_LM_OSL_LOC(
  A = 1,
  s = 1e8,
  E = 0.6,
  times = 0:100,
  clusters = 100,
  n_filled = 100,
  r = 1e-7,
  method = "par",
  output = "signal")

## plot
plot_RLumCarlo(results, legend = TRUE)

## End(Not run)
```

run_MC_LM_OSL_TUN

Run Monte-Carlo Simulation for LM-OSL (tunneling transitions)

Description

Runs a Monte-Carlo (MC) simulation of linearly modulated optically stimulated luminescence (LM-OSL) using the tunneling (TUN) model. Tunneling refers to quantum mechanical tunneling processes from the excited state of the trapped charge, into a 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 effective optical excitation rate for the tunneling process
rho	numeric (required) : The dimensionless density of recombination centers (defined as ρ' 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
r_c	numeric (with default) : Critical distance (>0) that is to be used if the sample has 1 been thermally and/or optically pretreated. This parameter expresses the fact that electron-hole pairs within a critical radius r_c have already been recombined.
delta.r	numeric (with default) : Increments of r_c (unitless)
N_e	numeric (with default) : The total number of electron traps available (unitless)
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges, electrons, in the trap)
...	further arguments

Details

The model

$$I_{TUN}(r', t) = -dn/dt = (A * t/P) * \exp(-(\rho')^{-1/3} * r') * n(r', t)$$

Where in the function:

A := the optical excitation rate for the tunneling process

t := time (s)

P := maximum stimulation time

r' := the unitless tunneling radius

ρ := rho the unitless density of recombination centres see Huntley (2006)

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

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](https://doi.org/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](https://doi.org/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](https://doi.org/10.1016/j.jlumin.2018.11.024)

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

Jain, M., Guralnik, B., Andersen, M.T., 2012. Stimulated luminescence emission from localized recombination in randomly distributed defects. *J. Phys.: Condens. Matter* 24, 385402. doi: [10.1088/09538984/24/38/385402](https://doi.org/10.1088/09538984/24/38/385402)

Examples

```
##the short example
run_MC_LM_OSL_TUN(
  A = 1,
  rho = 1e-7,
  times = 0:10,
  clusters = 10,
  N_e = 100,
  r_c = 0.001,
  delta.r = 1e-1,
  method = "seq",
  output = "signal") %>%
plot_RLumCarlo(norm = TRUE)

## Not run:
## the long (meaningful) example
results <- run_MC_LM_OSL_TUN(
  A = 1,
  rho = 1e-3,
  times = 0:1000,
  clusters = 30,
  N_e = 100,
  r_c = 0.1,
```

```

    delta.r = 1e-1,
    method = "par",
    output = "signal")

plot_RLumCarlo(results, norm = TRUE)

## End(Not run)

```

run_MC_TL_DELOC

Run Monte-Carlo Simulation for TL (delocalized transitions)

Description

Runs a Monte-Carlo (MC) simulation of thermoluminescence (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) : Retrapping ratio for delocalized transitions
method	character (with default) : Sequential 'seq' or parallel 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges/electrons in the trap)
...	further arguments

Details

The model

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

Where in the function:

E := the thermal activation energy (eV)

s := the frequency factor in (s⁻¹)

t := time

k_B := Boltzmann constant

T := temperature

n := n_{filled}, the instantaneous number of electrons

N := N_e, the 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 clusters` and a [numeric](#) time vector.

Function version

0.1.0

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](https://doi.org/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](https://doi.org/10.1142/2781)

Examples

```
## the short example
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:
## the long (meaningful) example
# 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 <- 300
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)

```

Description

Runs a Monte-Carlo (MC) simulation of thermo-luminescence (TL) using the generalized one trap (GOT) model. Localized transitions refer to transitions which do not involve the conduction or valence band. These transitions take place between the ground state and an excited state of the trapped charge, and also involve a state of the recombination center.

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 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges, electrons, in the trap)
...	further arguments

Details

The model

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

Where in the function:

E := the thermal activation energy (eV)

s := the frequency factor for the trap (1/s)

t := time (s)

k_B := Boltzmann constant

T := temperature (degrees C)

n := the Instantaneous number of electrons

r := the retrapping ratio for localized transitions

Value

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

Function version

0.1.0

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](https://doi.org/10.1016/j.jlumin.2018.11.024)

Examples

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

## Not run:
## the long (meaningful) example
run_MC_TL_LOC(
  s = 1e14,
  E = 0.9,
  times = 50:100,
  method = "par",
  clusters = 100,
  r = 1e4) %>%
plot_RLumCarlo()

## End(Not run)
```

run_MC_TL_TUN

Run Monte-Carlo Simulation for TL (tunneling transitions)

Description

Runs a Monte-Carlo (MC) simulation of thermo-luminescence (TL) caused by tunnelling (TUN) transitions. Tunneling refers to quantum mechanical tunneling processes from the excited state of the trapped charge, into a state of the recombination center.

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 effective frequency factor for the tunneling process (1/s)
E	numeric (required) : Thermal activation energy of the trap (eV)
rho	numeric (required) : The dimensionless density of recombination centers (defined as ρ' in Huntley 2006)
r_c	numeric (with default) : Critical distance (>0) that is to be used if the sample has been thermally and/or optically pretreated. This parameter expresses the fact that electron-hole pairs within a critical radius r_c have already been recombined.
times	vector (with 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 'par' processing. In the parallel mode the function tries to run the simulation on multiple CPU cores (if available) with a positive effect on the computation time.
output	character (with default) : output is either the 'signal' (the default) or 'remaining_e' (the remaining charges, electrons, in the trap)
...	further arguments

Details

The model

$$I_{TUN}(r', t) = -dn/dt = (s * \exp(-E/(k_B * T))) * \exp(-(\rho')^3 - 1/3) * r' * n(r', t)$$

Where in the function:

s := effective frequency for the tunneling process (1/s)

E := thermal activation energy (eV)

k_B := Boltzmann constant

T := temperature

r' := r electron-hole distance (unitless)

ρ' := Density of recombination centers

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

0.1.0

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](https://doi.org/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](https://doi.org/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](https://doi.org/10.1016/j.jlumin.2018.11.024)

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

Jain, M., Guralnik, B., Andersen, M.T., 2012. Stimulated luminescence emission from localized recombination in randomly distributed defects. *J. Phys.: Condens. Matter* 24, 385402. doi: [10.1088/09538984/24/38/385402](https://doi.org/10.1088/09538984/24/38/385402)

Examples

```
## the short example
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()

## Not run:
## the long (meaningful example)
run_MC_TL_TUN(
  s = 1e12,
  E = 0.9,
  rho = 0.01,
  r_c = 0.1,
  times = 80:220,
  clusters = 100,
  method = 'par',
```

```
    delta.r = 1e-1) %>%  
plot_RLumCarlo()  
  
## End(Not run)
```

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, [14](#)
run_MC_LM_OSL_DELOC, [17](#)
run_MC_LM_OSL_LOC, [18](#)
run_MC_LM_OSL_TUN, [20](#)
run_MC_TL_DELOC, [23](#)
run_MC_TL_LOC, [25](#)
run_MC_TL_TUN, [27](#)

* 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, [14](#)
run_MC_LM_OSL_DELOC, [17](#)
run_MC_LM_OSL_LOC, [18](#)
run_MC_LM_OSL_TUN, [20](#)
run_MC_TL_DELOC, [23](#)
run_MC_TL_LOC, [25](#)
run_MC_TL_TUN, [27](#)

* package

RLumCarlo-package, [2](#)

array, [5](#), [7](#), [9](#), [12](#), [13](#), [15](#), [18](#), [19](#), [22](#), [24](#), [27](#), [29](#)

character, [5](#), [6](#), [8](#), [11](#), [13](#), [15](#), [17](#), [19](#), [21](#), [23](#),
[26](#), [28](#)

graphics::plot.default, [3](#)

integer, [4](#), [8](#), [11](#), [13](#), [17](#), [19](#), [23](#), [26](#)

khroma::khroma-package, [4](#)

list, [3](#), [5](#), [7](#), [9](#), [12](#), [13](#), [15](#), [18](#), [19](#), [22](#), [24](#),
[27–29](#)

logical, [3](#)

numeric, [4–9](#), [11–13](#), [15](#), [17–19](#), [21–24](#), [26–29](#)

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, [14](#)

run_MC_LM_OSL_DELOC, [17](#)

run_MC_LM_OSL_LOC, [18](#)

run_MC_LM_OSL_TUN, [20](#)

run_MC_TL_DELOC, [23](#)

run_MC_TL_LOC, [25](#)

run_MC_TL_TUN, [27](#)

vector, [28](#)