Package 'tgcd'

May 23, 2016

Type Package

Title Thermoluminescence Glow Curve Deconvolution

Version 1.9

Date 2016-05-23

Author Jun Peng [aut, cre],
Jorge More [ctb],
Burton Garbow [ctb],
Kenneth Hillstrom [ctb],
John Burkardt [ctb],
Linda R. Petzold [ctb],
Alan C. Hindmarsh [ctb],
R. Woodrow Setzer [ctb],
Andrew Horchler [ctb]

Maintainer Jun Peng <pengjun10@mails.ucas.ac.cn>

Description Deconvolving thermoluminescence glow curves according to the general-order empirical expression or the semi-analytical expression derived from the one trap-one recombination (OTOR) model based on the Lambert's W function using a modified Levenberg-Marquardt algorithm. It provides the possibility of setting constraints or fixing any of parameters. It offers an interactive way to initialize parameters by clicking with a mouse on a plot at positions where peak maxima should be located. The optimal estimate is obtained by ``trial-and-error''. It also provides routines for simulating first-order, second-order, and general-order glow peaks (curves).

License GPL-2 | GPL-3

Depends R (>= 3.0.1)

Imports graphics, stats, utils

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

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-05-23 12:42:54

R topics documented:

`Refglow` *Reference glow curves*

Description

Two thermoluminescence glow curves.

Usage

```
data(Refglow)
```

Format

A list that contains two thermoluminescence glow curves:

Refglow002 a synthetic glow curve from the GLOCANIN project (Bos et al., 1993)

Refglow009 an experimental glow-curve from the GLOCANIN project (Bos et al., 1994)

Details

Refglow contains two thermoluminescence glow curves (Refglow002 and Refglow009) from the GLOCANIN project. The author is grateful to Adrie J.J. Bos for providing the reference glow curves of the GLOCANIN project to test function tgcd.

References

Bos AJJ, Piters TM, Gomez Ros JM, Delgado A, 1993. An intercomparison of glow curve analysis computer programs: I. Synthetic glow curves. Radiation Protection Dosimetry, 47(1-4), 473-477.

Bos AJJ, Piters TM, Gomez Ros JM, Delgado A, 1994. An intercomparison of glow curve analysis computer programs: II. Measured glow curves. Radiation Protection Dosimetry, 51(4): 257-264.

Examples

```
# Load package "tgcd".
 require(tgcd)

 data(Refglow)
 names(Refglow)
```

simPeak *Thermoluminescence glow peak simulation*

Description

Simulating first-order, second-order, or general-order glow peaks.

Usage

```
simPeak(temps, n0, Nn = NULL, bv = NULL, ff,
        ae, hr, typ = c("f", "s", "g"),
        outfile = NULL, plot = TRUE)
```

Arguments

temps	vector(**required**): temperature values (K) where the values of the thermoluminescence intensity will be computed. It needs to be sorted increasingly. A vector of temperature values may be generated using the internal function seq
n0	numeric(**required**): initial concentration of trapped electrons (1/cm^3)
Nn	numeric(**required**): total concentration of the traps in the crystal (1/cm^3)
bv	numeric(**required**): order number for the general order glow peak
ff	numeric(**required**): the frequency factor (1/s)
ae	numeric(**required**): the activation energy (eV)
hr	numeric(with default): the linear heating rate (K/s)
typ	character(with default): the type of a glow peak, `typ="f"` means first-order, `typ="s"` means second-order, `typ="g"` means general-order, default `typ="f"`
outfile	character(optional): if specified, simulated intensities of glow peaks will be written to a file named `"outfile"` in `CSV` format and saved to the current work directory
plot	logical(with default): draw a plot according to the simulated glow peak or not

Details

Function *simPeak* simulates glow peaks of various orders. The first-, second-, and general-order glow peak can be simulated using the following three ordinary equations, respectively (Pagonis et al., 2006):

$\frac{dn}{dT} = -n * S * e^{-E/(k*T)}/hr$

$\frac{dn}{dT} = -n^2 * S * e^{-E/(k*T)}/(Nn * hr)$

$\frac{dn}{dT} = -n^{bv} * S * e^{-E/(k*T)}/(Nn * hr)$

where n is the concentration of trapped electrons, $\frac{dn}{dT}$ the rate of change of the concentration of trapped electrons, S the frequency factor, E the activation energy, T the absolute temperature, k the Boltzmann constant, Nn the total concentration of the traps in the crystal, bv the b value (kinetic order), and hr the linear heating rate.

The ordinary equation is solved by the Fortran 77 subroutine *lsoda* (original version written by Linda R. Petzold and Alan C. Hindmarsh available at Netlib: http://www.netlib.org/odepack/, modified version by R. Woodrow Setzer from the R package *deSolve* (Soetaert et al., 2010) available at CRAN: http://CRAN.R-project.org/package=deSolve).

Value

Return an invisible list containing the following elements:

`temps`	a vector of temperature values
`tl`	values of the thermoluminescence intensity
`n`	variation of concentration of trapped electrons with temperature
`sp`	parameters used for describing the shape of a glow peak (Pagonis et al., 2006): the temperature corresponding to half intensity on the left side of the peak (`T1`); the temperature corresponding to half intensity on the right side of the peak (`T2`); the temperature corresponding to maximum intensity (`Tm`); the half-width at the left side of the peak (`d1=Tm-T1`); the half-width at the right side of the peak (`d2=T2-Tm`); the total half-width (`thw=d1+d2`); the symmetry factor (`sf=d2/thw`)

References

Pagonis V, Kitis G, Furetta C, 2006. Numerical and practical exercises in thermoluminescence. Springer Science & Business Media.

Soetaert K, Petzoldt T, Setzer RW, 2010. Solving Differential Equations in R: Package deSolve. Journal of Statistical Software, 33(9): 1-25.

See Also

tgcd; simqOTOR

Examples

```
### Effect of initial electron trap concentration (n0)
### on second-order glow peaks.
 temps <- seq(400, 600, by=0.5)
 peak1 <- simPeak(temps, n0=0.2e10, Nn=1e10,
   ff=1e19, ae=2.0, hr=1, typ="s")
 peak2 <- simPeak(temps, n0=0.4e10, Nn=1e10,
   ff=1e19, ae=2.0, hr=1, typ="s")
 peak3 <- simPeak(temps, n0=0.6e10, Nn=1e10,
   ff=1e19, ae=2.0, hr=1, typ="s")
 peak4 <- simPeak(temps, n0=0.8e10, Nn=1e10,
   ff=1e19, ae=2.0, hr=1, typ="s")
 peak5 <- simPeak(temps, n0=1.0e10, Nn=1e10,
   ff=1e19, ae=2.0, hr=1, typ="s")
 peaks <- cbind(peak1$tl, peak2$tl, peak3$tl, peak4$tl, peak5$tl)
 matplot(temps, peaks, type="l", lwd=2, lty="solid",
   xlab="Temperature (K)", ylab="TL intensity (counts)")
```

simqOTOR *Thermoluminescence glow peak simulation*

Description

Simulating glow peaks according to the one trap-one recombination center (OTOR) model using the quasi-equilibrium approximation.

Usage

```
simqOTOR(temps, n0, Nn, Ah, An, ff, ae,
         hr, outfile = NULL, plot = TRUE)
```

Arguments

temps	vector(**required**): temperature values (K) where the values of the thermoluminescence intensity will be computed, it needs to be sorted increasingly
n0	numeric(**required**): initial concentration of trapped electrons (1/cm^3)
Nn	numeric(**required**): total concentration of the traps in the crystal (1/cm^3)
Ah	numeric(optional): probability coefficient of electron recombining with holes in the recombination center (cm^3/s)
An	numeric(optional): probability coefficient of electron retrapping in the traps (cm^3/s)
ff	numeric(**required**): the frequency factor (1/s)
ae	numeric(**required**): the activation energy (eV)
hr	numeric(with default): the linear heating rate (K/s)
outfile	character(optional): if specified, simulated intensities of glow peaks will be written to a file named "outfile" in CSV format and saved to the current work directory
plot	logical(with default): draw a plot according to the simulated glow peak or not

Details

Function *simqOTOR* simulates a synthetic glow peak according to the OTOR model using the quasi-equilibrium approximation. This function may be used to simulating glow peaks of first-, second-, and general-order, depending on the given kinetic parameters. The approximate equation of the OTOR model derived using the quasi-equilibrium approximation can be described by (Pagonis et al., 2006):

$\frac{dn}{dT} = -Ah * n^2 * S * e^{-E/(kT)}/((n * Ah + (Nn - n) * An) * hr)$

where n is the concentration of trapped electrons, $\frac{dn}{dT}$ the rate of change of the concentration of trapped electrons, S the frequency factor, E the activation energy, T the absolute temperature, k

the Boltzmann constant, Nn the total concentration of the traps in the crystal, Ah the probability coefficient of electron recombining with holes in the recombination center, An the probability coefficient of electron retrapping in the traps, and hr the linear heating rate.

The ordinary equation is solved by the Fortran 77 subroutine *lsoda* (original version written by Linda R. Petzold and Alan C. Hindmarsh available at Netlib: http://www.netlib.org/odepack/, modified version by R. Woodrow Setzer from the R package *deSolve* (Soetaert et al., 2010) available at CRAN: http://CRAN.R-project.org/package=deSolve).

Value

Return an invisible list containing the following elements:

`temps`	a vector of temperature values
`tl`	values of the thermoluminescence intensity
`n`	variation of concentration of trapped electrons with temperature
`sp`	parameters used for describing the shape of a glow peak, see function simPeak for details

References

Pagonis V, Kitis G, Furetta C, 2006. Numerical and practical exercises in thermoluminescence. Springer Science & Business Media.

Soetaert K, Petzoldt T, Setzer RW, 2010. Solving Differential Equations in R: Package deSolve. Journal of Statistical Software, 33(9): 1-25.

See Also

tgcd; simPeak

Examples

```
# Synthesizing a glow curve consisting of five glow peaks.
temps <- seq(330, 730, by=0.5)
peak1 <- simqOTOR(temps, n0=0.7e10, Nn=1e10, Ah=1e-3, An=1e-7,
  ff=1e14, ae=1.5, hr=1, outfile = NULL, plot = TRUE)
peak2 <- simqOTOR(temps, n0=0.5e10, Nn=1e10, Ah=1e-7, An=1e-7,
  ff=1e17, ae=1.9, hr=1, outfile = NULL, plot = TRUE)
peak3 <- simqOTOR(temps, n0=0.2e10, Nn=1e10, Ah=1e-5, An=1e-7,
  ff=1e15, ae=1.45, hr=1, outfile = NULL, plot = TRUE)
peak4 <- simqOTOR(temps, n0=0.2e10, Nn=1e10, Ah=1e-5, An=1e-7,
  ff=1e9, ae=0.85, hr=1, outfile = NULL, plot = TRUE)
peak5 <- simqOTOR(temps, n0=0.3e10, Nn=1e10, Ah=1e-7, An=1e-7,
  ff=1e11, ae=1.4, hr=1, outfile = NULL, plot = TRUE)
peaks <- cbind(peak1$tl, peak2$tl, peak3$tl, peak4$tl, peak5$tl,
           peak1$tl+peak2$tl+peak3$tl+peak4$tl+peak5$tl)
matplot(temps, y=peaks, type="l", lwd=2, lty="solid",
  xlab="Temperature (K)", ylab="TL intensity (counts)")
```

tgcd *Thermoluminescence glow curve deconvolution (TGCD)*

Description

Thermoluminescence glow curve deconvolution according to the general-order empirical expression or the semi-analytical expression derived from the one trap-one recombination (OTOR) model based on the Lambert's W function.

Usage

```
tgcd(Sigdata, npeak, inis = NULL, mwt = 90,
     mdt = 3, nstart = 30, model=c("g", "lw"),
     elim = NULL, logy = FALSE, hr = NULL,
     outfile = NULL, plot = TRUE)
```

Arguments

`Sigdata`	matrix(**required**): a `two-column` matrix, temperature values and thermoluminescence signal values are stored in the first and second column, respectively
`npeak`	integer(**required**): number of glow peaks, the allowed maximum number of glow peaks is set equal to 13
`inis`	matrix(optional): a `npeak-row 4-column` matrix used for storing starting parameters `Im`, `E`, `Tm`, `b` (or `R`). If `inis=NULL`, the user has to click with a mouse on a plot to locate each glow peak maximum
`mwt`	numeric(with default): allowed maximum total half-width of deconvoluted glow peaks. A smaller `mwt` prevents the appearance of glow peaks with large total half-width. A conservative `mwt` is adopted by default
`mdt`	numeric(with default): allowed minimum distance between each optimized temperature at maximum thermoluminescence intensity. A larger `mdt` prevents the appearance of strongly overlapping peaks. A conservative `mdt` is adopted by default
`nstart`	integer(with default): number of trials, if `nstart>1`, a "try-and-error" protocol will be performed, the upper limit for `nstart` is set equal to 10000
`model`	character(with default): `"g"` means fitting a general-order model, `"lw"` means fitting the Lambert W-function, default `model="g"`
`elim`	vector(with default): lower and upper limits for the activation energy, default `elim=c(0.5, 5)`
`logy`	logical(with default): draw the y-axis of the plot used for locating peak maxima with a logarithmic scale or not
`hr`	numeric(optional): the linear heating rate used for calculating the frequency factor
`outfile`	character(optional): if specified, fitted signal values for each glow peak will be written to a file named `"outfile"` in `CSV` format and saved to the current work directory
`plot`	logical(with default): draw a plot according to the fitting result or not

Details

Function *tgcd* is used for deconvolving thermoluminescence glow curves according to the general-order empirical expression (Kitis et al., 1998; Pagonis et al., 2006) or the semi-analytical expression derived from the one trap-one recombination (OTOR) model based on the Lambert's W function (Kitis and Vlachos, 2013; Sadek et al., 2015; Kitis et al., 2016) using the Levenberg-Marquardt algorithm (plus supports for constraining and fixing parameters).

The general-order empirical expression for a glow peak is:

```
I(T)=Im*b^(b/(b-1))*expv*((b-1)*(1-xa)*(T/Tm)^2*expv+Zm)^(-b/(b-1))
xa=2*k*T/E
xb=2*k*Tm/E
expv=exp(E/(k*T)*(T-Tm)/Tm)
Zm=1+(b-1)*xb
```

where `b` is the kinetic parameter (lies between 1 and 2), `I` is the glow peak intensity, `E` the activation energyin ev, `k` the Boltzmann constant in eV/k, T the temperature in K with constant heating rate K/s, `Tm` the temperature at maximum thermoluminescence intensity in K, and `Im` the maximum intensity. The four parameters for this model are: `Im`, `E`, `Tm`, and `b`.

The semi-analytical expression derived from the one trap-one recombination (OTOR) model based on the Lambert's W function is:

```
I(T)=Im*exp(-E/(k*T)*(Tm-T)/Tm)*(W(Zm)+W(Zm)^2)/(W(Z)+W(Z)^2)
Zm=R/(1-R)-log((1-R)/R)+E*exp(E/(k*Tm))/(k*Tm^2*(1-1.05*R^1.26))*F(Tm,E)
Z=R/(1-R)-log((1-R)/R)+E*exp(E/(k*Tm))/(k*Tm^2*(1-1.05*R^1.26))*F(T,E)
F(Tm,E)=Tm*exp(-E/(k*Tm))+E/k*Ei(-E/(k*Tm))
F(T,E)=T*exp(-E/(k*T))+E/k*Ei(-E/(k*T))
```

where `W(x)` is the wright Omega function for variable x, `Ei(x)` is the exponential integral function for variable x, `I` is the glow peak intensity, `E` the activation energy in eV, `k` the Boltzmann constant in eV/K, `T` the temperature in K with constant heating rate in K/s, `Tm` the temperature at maximum thermoluminescence intensity in K, and `Im` the maximum intensity. The four parameters for this model are: `Im`, `E`, `Tm`, and `R`. The Fortran 90 subroutine used for evaluating the Wright Omega function is transformed from the Matlab code provided by Andrew Horchler available at https://github.com/horchler/wrightOmegaq.

The procedure minimizes the objective:

$fcn = \sum_{i=1}^{n} |y_i^o - y_i^f|, i = 1, ..., n$

where y_i^o and y_i^f denote the *i-th* observed and fitted signal value, respectively, and n indicates the number of data points.

Starting parameters (`inis`) can be specified by the user through argument `inis` or by clicking with a mouse on the plot of the thermoluminescence glow curve showing peak maxima if `inis=NULL`.The Levenberg-Marquardt algorithm (More, 1978) (*minpack*: Original Fortran 77 version by Jorge More, Burton Garbow, Kenneth Hillstrom. Fortran 90 version by John Burkardt freely available at http://people.sc.fsu.edu/~jburkardt/f_src/minpack/) was modified so as to supports constraints and fixes of parameters. If argument `nstart>1`, a "try-and-error" protocol with starting values generated uniformly around the given starting values `inis` will be performed repeatedly to search the optimal parameters that give a minimum Figure Of Merit (FOM) value.

Parameters can be interactively constrained and fixed by modifying the following elements in a automatically generated **Dialog Table** if `inis=NULL`:
(1) *INTENS(min, max, ini, fix)*: lower and upper bounds, starting and fixing values of `Im`
(2) *ENERGY(min, max, ini, fix)*: lower and upper bounds, starting and fixing values of `E`
(3) *TEMPER(min, max, ini, fix)*: lower and upper bounds, starting and fixing values of `Tm`
(4) *bValue(min, max, ini, fix)*: lower and upper bounds, starting and fixing values of `b`

Value

Return a list containing the following elements:

`pars`	optimized parameters stored in a matrix
`ff`	calculated frequency factor if `hr!=NULL`
`sp`	parameters used for describing the shape of a glow peak, see function simPeak for details
`FOM`	minimized Figure Of Merit

Note

The model to be optimized should not be underdetermined. This means that the number of data points should exceed the number of parameters. A lack of background counts in the analyzed data is assumed. To obtain reliable estimate, the presented background may be accounted for by subtracting from measured data before analysis.

References

Kitis G, Gomes-Ros JM, Tuyn JWN, 1998. Thermoluminescence glow curve deconvolution functions for first, second and general orders of kinetics. Journal of Physics D: Applied Physics, 31(19): 2636-2641.

Kitis G, Polymeris GS, Sfampa IK, Prokic M, Meric N, Pagonis V, 2016. Prompt isothermal decay of thermoluminescence in MgB4O7:Dy, Na and LiB4O7:Cu, In dosimeters. Radiation Measurements, 84: 15-25.

Kitis G, Vlachos ND, 2013. General semi-analytical expressions for TL, OSL and other luminescence stimulation modes derived from the OTOR model using the Lambert W-function. Radiation Measurements, 48: 47-54.

More JJ, 1978. "The Levenberg-Marquardt algorithm: implementation and theory," in Lecture Notes in Mathematics: Numerical Analysis, Springer-Verlag: Berlin. 105-116.

Pagonis V, Kitis G, Furetta C, 2006. Numerical and practical exercises in thermoluminescence. Springer Science & Business Media.

Sadek AM, Eissa HM, Basha AM, Carinou E, Askounis P, Kitis G, 2015. The deconvolution of thermoluminescence glow-curves using general expressions derived from the one trap-one recombination (OTOR) level model. Applied Radiation and Isotopes, 95: 214-221.

Further reading

Bos AJJ, Piters TM, Gomez Ros JM, Delgado A, 1993. An intercomparison of glow curve analysis computer programs: I. Synthetic glow curves. Radiation Protection Dosimetry, 47(1-4), 473-477.

Chung KS, Choe HS, Lee JI, Kim JL, Chang SY, 2005. A computer program for the deconvolution of thermoluminescence glow curves. Radiation Protection Dosimetry, 115(1-4): 345-349. Software is freely available at http://physica.gsnu.ac.kr/TLanal.

Harvey JA, Rodrigues ML, Kearfott JK, 2011. A computerized glow curve analysis (GCA) method for WinREMS thermoluminescent dosimeter data using MATLAB. Applied Radiation and Isotopes, 69(9):1282-1286. Source codes are freely available at http://www.sciencedirect.com/science/article/pii/S0969804311002685.

Kiisk V, 2013. Deconvolution and simulation of thermoluminescence glow curves with Mathcad. Radiation Protection Dosimetry, 156(3): 261-267. Software is freely available at http://www.physic.ut.ee/~kiisk/mcadapps.htm.

Puchalska M, Bilski P, 2006. GlowFit-a new tool for thermoluminescence glow-curve deconvolution. Radiation Measurements, 41(6): 659-664. Software is freely available at http://www.ifj.edu.pl/dept/no5/nz58/deconvolution.htm.

See Also

simPeak; simqOTOR

Examples

```
# Load the data.
  data(Refglow)

# Deconvolve Refglow002 with 4 peaks using the Lambert W-function.
  startingPars <-
  cbind(c(400, 550, 850, 1600), # Im
        c(1.4, 1.5, 1.6, 2),    # E
        c(420, 460, 480, 510),  # Tm
        c(0.1, 0.1, 0.1, 0.1))  # R
  tgcd(Refglow$Refglow002, npeak=4, model="lw",
       inis=startingPars, nstart=10)

# Do not run.
# Deconvolve Refglow009 with 9 peaks using the general-order equation.
# startingPars <-
# cbind(c(9824, 21009, 27792, 50520, 7153, 5496, 6080, 1641, 2316), # Im
#       c(1.24, 1.36, 2.10, 2.65, 1.43, 1.16, 2.48, 2.98, 2.25),    # E
#       c(387, 428, 462, 488, 493, 528, 559, 585, 602),             # Tm
#       c(1.02, 1.15, 1.99, 1.20, 1.28, 1.19, 1.40, 1.01, 1.18))    # b
# tgcd(Refglow$Refglow009, npeak=9, model="g",
#      inis=startingPars, nstart=10)
```

Index