# tgcd: An R package for analyzing thermoluminescence glow curves

Jun Peng [a,*], ZhiBao Dong [a], FengQing Han [b]

[a] *Cold and Arid Regions Environmental and Engineering Research Institute, Chinese Academy of Sciences, 320 Donggang West Road, Lanzhou, China*
[b] *Qinghai Institute of Salt Lakes, Chinese Academy of Sciences, 18 Xinning Road, Xining, China*

## Abstract

Thermoluminescence (TL) glow curves are widely used in dosimetric studies. Many commercial and free-distributed programs are used to deconvolute TL glow curves. This study introduces an open-source R package tgcd to conduct TL glow curve analysis, such as kinetic parameter estimation, glow peak simulation, and peak shape analysis. TL glow curves can be deconvoluted according to the general-order empirical expression or the semi-analytical expression derived from the one trap-one recombination center (OTOR) model based on the Lambert $W$ function by using a modified Levenberg–Marquardt algorithm from which any of the parameters can be constrained or fixed. The package provides an interactive environment to initialize parameters and offers an automated "trial-and-error" protocol to obtain optimal fit results. First-order, second-order, and general-order glow peaks (curves) are simulated according to a number of simple kinetic models. The package was developed using a combination of Fortran and R programming languages to improve efficiency and flexibility.
© 2016 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

*Keywords:* TL glow curve; Deconvolution; Simulation; R package

## Code metadata

| | |
|---|---|
| Current code version | v1.9 |
| Permanent link to code/repository used of this code version | https://github.com/ElsevierSoftwareX/SOFTX-D-15-00066 |
| Legal Code License | GPL-2 or GPL-3 |
| Code versioning system used | None |
| Software code languages, tools, and services used | Fortran 77, Fortran 90, and R |
| Compilation requirements, operating environments & dependences | R software |
| If available Link to developer documentation/manual | https://github.com/ElsevierSoftwareX/SOFTX-D-15-00066/blob/master/README.pdf |
| Support email for questions | pengjun10@mails.ucas.ac.cn |

## 1. Motivation and significance

Thermoluminescence (TL) is the light emitted by a material when heated, which results from emissions of absorbed energy. Most semiconductors or insulators display a TL glow curve that consists of one or more peaks when charge carriers are released. The goal of TL glow curve analysis is to estimate parameters that can then be used to describe TL processes that take place

in the material [1]. A number of computer procedures have been developed to fit TL glow curves. Many use commercial software as running platforms. Using Mathematica software, Pagonis et al. [1] presented comprehensive examples of TL data analysis for glow curves following first-order, second-order, and general-order kinetics. Harvey et al. [2] and El-Hafez et al. [3] used MATLAB software to fit first-order TL glow curves. Afouxenidis et al. [4] employed a spreadsheet program to fit general-order TL glow curves. Kiisk [5] used the Mathcad program to fit a few common analytical approximations of first-order TL glow curves. Some studies have also used the PEAKFIT and MINUIT software programs to fit TL glow

curves [6,7]. Two self-contained and freely accessible packages for TL glow curve deconvolution are TLanal [8], which fits first-order, second-order, and general-order kinetics, and Glowfit [9], which fits first-order kinetics. Given that many researchers do not have access to commercial software, the use of commercial software as running platforms largely results in the restrictive redistribution of a program. Moreover, some popular programs (such as TLanal and Glowfit) are designed to only work under Microsoft Windows operating systems and are therefore black boxes to non-software engineers. Alternatively, R is a free software environment for statistical computing and graphics [10]. Furthermore, R runs on a wide variety of operating systems (Linux, Windows, and MacOS). It is also considered as a well-developed, simple, and effective programming language. R can easily communicate with external programming languages such as Fortran, C++, and C. Acceptable running speeds can be achieved by writing the time-consuming part of a program in Fortran or C++ and linking it to R using an interface [11].

It has been previously reported that TL glow curve deconvolution can yield unreliable fit results while being mathematically correct [9]. The analysis of TL glow curves relies on using carefully measured data sets, given that any errors in measurements can lead to incorrect results in computerized programs [1]. Curve fitting methods based on a particular theoretical model should be applied with the utmost care, and extreme caution should be taken when drawing conclusions [12]. Furthermore, the deconvolution of complex glow curves may not converge to unique results unless sufficiently stringent constraints are imposed [5]. Complex glow curves consisting of strongly overlapping peaks or having very poor statistics can be resolved by setting constraints or by fixing particular parameters [9]. In the R package tgcd, the Levenberg–Marquardt algorithm, implemented using the Fortran subroutine MINPACK [13], was modified to allow the fixing or constraining of any parameter in order to enable the user to obtain meaningful fit results from complex glow curves. Fixed parameters remain unchanged during the fitting process.

May and Partridge [14] first proposed a general-order kinetic equation with the kinetic order $b$, ranging from 1 to 2 for intermediate cases between first-order and second-order kinetics. For over two decades, general-order kinetics have most commonly been applied to analyze TL glow curves. Recently, Kitis and Vlachos [15] developed new semi-analytical expressions for TL glow peaks based on the one trap-one recombination center (OTOR) model using the Lambert $W$ function. Sadek et al. [16,17] found that the developed OTOR TL expressions can accurately describe TL glow peaks even in cases where other TL expressions have failed. Sadek et al. [18] suggested that OTOR TL expressions are the most suitable expressions to use in deconvolution of experimental glow curves given that the peak fitting method applying these expressions can provide accurate values for activation energy of glow peaks whether re-trapping or recombination is dominated. In the R package tgcd, glow curves can be deconvoluted according to the general-order empirical expression provided by Kitis et al. [19] or the semi-analytical expression derived

from the OTOR model using the Lambert $W$ function outlined by Sadek et al. [20] and Kitis et al. [21]. Fitting equations of the two different types of models are presented in detail in the User Manual of the R package (for the permanent link, see row 7 of the Code Metadata Table) and therefore will not be repeated here. The quality of fit is measured by the figure-of-merit (FOM) [22].

Mathematically speaking, fitting TL glow curves is an ill-conditioned problem, and fit results are very sensitive to the choice of initial parameters. To obtain reasonable fit results, it is essential to rerun procedures with various initial parameters. In the MATLAB program developed by Harvey et al. [2], for example, parameters were repeatedly optimized using previously gathered knowledge related to peak positions and activation energies taken from literature as starting points until the best possible fits were obtained. A differential evolution algorithm was also suggested to automatically initialize starting parameters of an optically stimulated luminescence (OSL) decay curve [23]. This algorithm has been successfully implemented in R package numOSL [24]. For fitting TL glow curves, however, stochastic algorithms of this kind are impracticable as the number of unknown parameters is very large compared to those in OSL decay curves. Accordingly, user-provided initial parameters are generally required by most existing programs. The glow curve analysis (GCA) program [25] provides an algorithm to automatically search glow peak positions, but some peaks initialized by this method may require manual adjustment before running the program [9].

It demonstrates that quality of fit is much more dependent on the position of glow peaks than on other parameters [26]. Both TLanal [8] and Glowfit [9] software offers a convenient and intuitive way to initialize parameters by using a mouse to click on glow curves at positions where peak maxima should be located. The R package tgcd offers a similar way to locate peak temperature ($T_m$) and peak intensity ($I_m$) for each glow peak with the aid of a mouse. On execution, a graph of the original glow curve data is automatically drawn to prompt the user to locate the coordinate ($x = T_m$, $y = I_m$) of each glow peak. The R package also offers an alternative to initialize parameters by using a matrix to store initial parameter guess values [5]. However, this method is only feasible if the peak position and activation energy of each glow peak have been well determined for the type of TL dosimeter being analyzed [2]. For complex glow curves that consist of a number of glow peaks, this parameter initialization method can be tedious.

To enable the user to obtain an optimal fit result, the R package provides an automated "trial-and-error" protocol using random parameters generated around starting values to repeatedly fit the glow curve at certain times (the allowed maximum number of random trials *nstart* is specified by the user). Given that the procedure was programmed using the Fortran language, the "trial-and-error" protocol is very efficient. Fitting parameters that yield the lowest FOM values will be treated as optimal fit results. However, "optimal" parameters with very low FOM values that are minimized using the automated "trial-and-error" protocol may be characterized by glow peaks with large total half-widths