



Original software publication

takos: An R package for thermal analysis calculations

G. Luciano^a, K.H. Liland^{b,*}, M. Marsotto^a, R. Svoboda^c, S. Berretta^d^a CNR - ISMAC, Via De Marini 6, 16149 Genoa, Italy^b Norwegian University of Life Sciences, Department of Science and Technology, NO-1432 Ås, Norway^c Department of Physical Chemistry, Faculty of Chemical Technology, University of Pardubice, CZ-532 10 Pardubice, Czech Republic^d CNR - Imati, Via De Marini 6, 16149 Genoa, Italy

ARTICLE INFO

Article history:

Received 19 October 2020

Received in revised form 19 November 2020

Accepted 23 November 2020

Keywords:

Calorimetry

Thermal analysis

Kinetic analysis

R

ABSTRACT

Thermal analysis consists of a wide range of methodologies that can be applied to assess the composition and properties of materials. This paper describes the basic features of a new package for the R software named *takos* for simulating and analysing calorimetric data sets. The package can simulate data via the Šesták Berggren (SB), Johnson–Mehl–Avrami (JMA) and other common kinetic models used in solid state kinetics (power law, one dimensional diffusion, Mampel, Avrami–Erofeev, three dimensional diffusion, contracting sphere, contracting cylinder, and two-dimensional diffusion). The methodologies included in order to determine the kinetic triplet are the Avrami, Friedman, Kissinger, Ozawa, Ozawa–Flynn and Wall (OFW), Mo, Starink and Vyazovkin methodology (Vyazovkin). The package is under constant development, being improved and extended with new functionalities, as well as being continually tested on real-life data, the analyses of which are peer-reviewed during their respective publication processes.

© 2020 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/>).

Code metadata

Current code version
 Permanent link to code/repository used for this code version
 Legal Code License
 Software code languages, tools, and services used
 Compilation requirements, operating environments & dependencies

v0.2.0
<https://github.com/ElsevierSoftwareX/SOFTX-D-20-00069>
 GPL-2
 R
 R no compilation needed;
 Dependencies: MASS, devEMF, segmented, sfsmisc, smoother, pracma, data.table, broom, colorRamps, minpack.lm, tools, baseline, graphics
<https://cran.r-project.org/web/packages/takos/takos.pdf>
serena.berretta@ge.imati.cnr.it

If available Link to developer documentation/manual
 Support email for questions

1. Motivation and significance

To the authors' knowledge, several non FOSS (free and open source) softwares are able to perform analyses of thermoanalytic data but there is a lack of scriptable, open source solutions for performing a complete thermal analysis on the data acquired via differential scanning calorimetry (DSC). In addition, the availability of a package in the R environment can help researchers to rely on a reproducible work flow [1]. The usefulness of such a tool is evident as it provides qualitative testing of the considered methodology at unprecedented precision. This is further

enhanced by letting users write scripts that can analyse data by means of the most widely adopted methodologies in an efficient, reproducible pipeline. In this context the authors developed the package *takos*.

2. Software framework

As reported in the ICTAC Kinetics Committee recommendations for collecting experimental thermal analysis data for kinetic computations [2]:

Generally, any physical property whose change is measured as a function of time can be a source of kinetic data. In the area of thermal analysis, kinetic data are collected most commonly

* Corresponding author.

E-mail address: kristian.liland@nmbu.no (K.H. Liland).

by measuring changes in either heat (calorimetry) or mass (thermogravimetry).

For any monitored physical property ξ we can write that

$$\alpha = \frac{\xi_i - \xi}{\xi_i - \xi_f}$$

where α is the degree of the process conversion, ξ is the current state of a property, ξ_i its initial and ξ_f its final value. When the process starts, α values are at 0 while at the process end they approach 1. Values of α in the range 0–1 represent the process progress at a given moment of time and is determined via a fractional change in the measured property.

The properties measured can vary depending on the technique adopted. For example, in thermogravimetry (TG), α is the ratio of the mass change to the total mass of the sample during the process, while in differential scanning calorimetry (DSC) the progress is evaluated as a ratio of the current heat change to the total heat released or absorbed in the process.

Considering a single step process, we have:

$$\frac{d\alpha}{dt} = k(T)f(\alpha)h(P) \quad (1)$$

where t is time, T is temperature and P is pressure. In the case of multiple processes, it is necessary to evaluate the progress of each single step separately. The pressure dependence, $h(P)$, can be a very important variable but it is generally not considered in the majority of kinetic methods in thermal analysis, and it is beyond the scope of the software implemented (more on this topic can be found in [3]). For sake of clarity we will rewrite the previous equation as

$$\frac{d\alpha}{dt} = k(T)f(\alpha). \quad (2)$$

In this equation, we made the assumption that we have the product of two functions independent from each other, where one depends solely on T and the other one ($f(\alpha)$) solely on the conversion process. In other words, the conversion function $f(\alpha)$ at a fixed value of α holds for any temperature or temperature regime, i.e. the mechanism of the process is solely a function of conversion, it is not a function of temperature. These assumptions constitute the base for isoconversional methodologies.

The dependence of α is the reaction model $f(\alpha)$ that can take a variety of mathematical forms as expressed in Table 1, while $k(T)$ is generally represented by the Arrhenius equation reported in (3), where A is the pre-exponential factor (s^{-1}), E is the activation energy (Jmol^{-1}), T is the absolute reaction temperature (K) and R stands for the universal gas constant, $8.314 \text{ Jmol}^{-1}K^{-1}$.

$$k(T) = Ae^{\left(\frac{-E}{RT}\right)} \quad (3)$$

Combining Eqs. (2) and (3) when we have a linear heating temperature program we can obtain:

$$\left(\frac{d\alpha}{dt}\right)_\alpha = \left(\beta \frac{d\alpha}{dT}\right)_\alpha = [A_\alpha f(\alpha)] \exp\left[-\frac{E}{RT_\alpha}\right] = A_\alpha \exp\left[-\frac{B_\alpha}{T_\alpha}\right] \quad (4)$$

where the subscript α refers to a fixed conversion degree and β is the applied heating rate. This is the starting point for **differential methods**. Applying a logarithmic transformation to (4) we have:

$$\ln\left(\beta \frac{d\alpha}{dT}\right)_\alpha = \ln A_\alpha - \frac{B_\alpha}{T} \quad (5)$$

known as Friedman's method.

Integrating (4), we have:

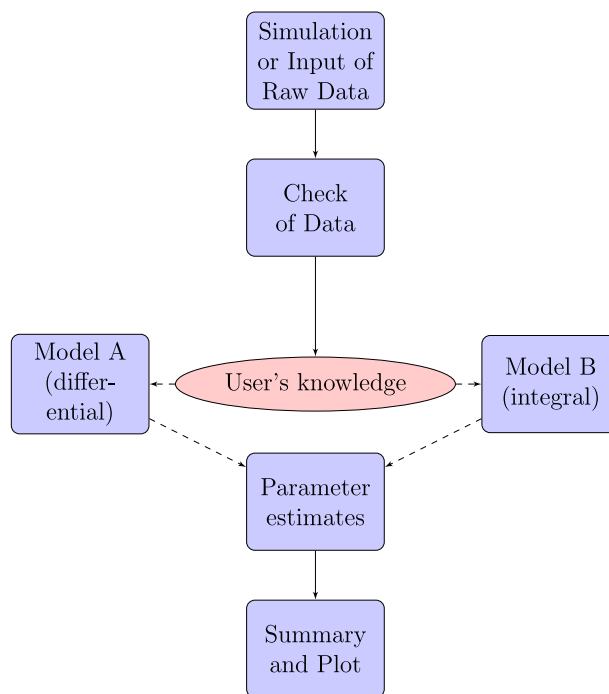
$$\int_0^\alpha \frac{d\alpha}{f(\alpha)} = \int_0^{t_\alpha} k(T)dt \quad (6)$$

which is the starting point for **integral methods**. All the expressions for methods implemented in *takos* are presented in Table 2.

In order to determine the kinetic triplets (E , A and $f(\alpha)$) applying the methodologies presented, we need to simulate or collect data sets of kinetic runs from the dependencies of time vs. heating rate (for isothermal measurements), temperature vs. heating rate (for integral and incremental methods with linear heating rate) or from reaction rate vs. temperature (for the differential Friedman method). It is worth noting that *takos* implements each methodology by means of dedicated functions, allowing the user to create scripts able to perform analysis in batches.

It should be noted that the methods listed in Table 2 are not equally adoptable regarding their utilization in kinetic analysis. For example, the Kissinger method (based on the utilization of the temperature corresponding to the maximum of the kinetic peak) is largely insensitive to the process complexity and must not be used during cooling (see e.g. [16,17]). The rigid integral isoconversional methods (such as KAS, OFW or Starink) were also derived for the single-process analysis, and when the activation energy exhibits variability above 20%, these methods result in large systematic errors (see e.g. [18,19]). Although the kinetic analysis is usually performed on data obtained during heating, a similar approach can be also adopted for data obtained during cooling (such as e.g. crystallization of melts). However, in such case only certain methods (such as the Friedman or the Vyazovkin advanced isoconversional method) that reflect the cooling temperature program can be used (see [20] for more detail).

2.1. Software architecture



2.2. Software functionalities

The R package depends on packages MASS, devEMF, segmented, sfsmisc, smoother, pracma, data.table, broom, colorRamps, minpack.lm, tools, baseline and graphics.

Its latest version (0.2), including the User Manual, is downloadable from the Comprehensive R Archive Network (CRAN)

Table 1
Reaction model implemented in **takos**.

Reaction Model	Code	$f(a)$	$g(a) = \int_0^a f(a) da$
Power Law	P4	$4\alpha^{3/4}$	$\int d\alpha / (4\alpha^{3/4})$
Power Law	P3	$3\alpha^{2/3}$	$\alpha^{1/3}$
Power Law	P2	$2\alpha^{1/2}$	$\alpha^{1/2}$
Power Law	P2/3	$2/3\alpha^{-1/2}$	$\alpha^{3/2}$
One-Dimensional diffusion	D1	$1/2\alpha^{-1}$	α^2
Mampel (first order)	F1	$1 - \alpha$	$-\ln(1 - \alpha)$
Avrami-Erofeev	A4	$4(1 - \alpha)[- \ln(1 - \alpha)]^{3/4}$	$[- \ln(1 - \alpha)]^{1/4}$
Avrami-Erofeev	A3	$3(1 - \alpha)[- \ln(1 - \alpha)]^{2/3}$	$[- \ln(1 - \alpha)]^{1/3}$
Avrami-Erofeev	A2	$2(1 - \alpha)[- \ln(1 - \alpha)]^{1/2}$	$[- \ln(1 - \alpha)]^{1/2}$
Three-dimensional diffusion	D3	$3/2(1 - \alpha)^{2/3}[1 - (1 - \alpha)^{1/3}]^{-1}$	$[1 - (1 - \alpha)^{1/3}]^2$
Contracting sphere	R3	$3(1 - \alpha)^{2/3}$	$[1 - (1 - \alpha)]^{1/3}$
Contracting cylinder	R2	$2(1 - \alpha)^{1/2}$	$[1 - (1 - \alpha)]^{1/2}$
Two-dimensional diffusion	D2	$[- \ln(1 - \alpha)]^{-1}$	$[(1 - \alpha)\ln(1 - \alpha)] + \alpha$
Truncated Šesták Berggren	SB	$\alpha^m(1 - \alpha)^n$	$\int d\alpha / (\alpha^m(1 - \alpha)^n)$
Johnson-Mehl-Avrami	JMA	$m(1 - \alpha)[- \ln(1 - \alpha)]^{1-(1/m)}$	$[- \ln(1 - \alpha)]^{1/m}$

Table 2
Kinetics methods implemented in **takos**.

Method	Expression	Ref.
Johnson-Mehl-Avrami	$\frac{d\alpha}{dt} = kn(1 - \alpha)[- \ln(1 - \alpha)]^{1-1/n}$	[4,5]
Differential Kissinger	$\ln\left(\frac{\beta}{T_p}\right) = \ln\frac{A_R}{E} - \frac{E}{RT_p}$	[6]
Differential Friedman	$\ln\left(\frac{d\alpha}{dt}\right)_\alpha = \ln\left(\beta \frac{d\alpha}{dt}\right)_\alpha = \ln f(\alpha)(A_\alpha) - \frac{E}{RT_\alpha}$	[7]
Integral Avrami	$\log(-\ln(1 - \alpha(T, t))) = \beta \log(k') + n \log t$	[8]
Integral Ozawa	$\log(-\ln(1 - \alpha(T, t))) = \log K^*(T) + m \log \beta$	[9]
Integral Flynn-Wall-Ozawa	$\log(\beta) = \log\left(\frac{A_E}{R}\right) - \log(g(\alpha(T))) - 5.331 - 1.052 \frac{E}{RT}$	[10]
Integral Kissinger-Akahira-Sunose	$\log\left(\frac{\beta}{T^2}\right) = \log\left(\frac{A_E}{Rg(\alpha(T))}\right) - \frac{E}{RT}$	[11]
Integral Mo	$\log(\beta) = \log(F(T)) - b \log(t)$	[12]
Integral Advanced Isoconversional Method	$J(E_\alpha, t_\alpha) = \int_0^{t_\alpha} \exp\left[-\frac{E}{RT}\right] dt$	[13]
Integral Modified Advanced Isoconversional Method	$J(E_\alpha, t_\alpha) = \int_{t_\alpha - \Delta\alpha}^{t_\alpha} \exp\left[-\frac{E}{RT}\right] dt$	[14]
Integral Starink	$\log\left(\frac{\beta}{T^{1.92}}\right) = -1.0008 \frac{E}{RT} + \text{Const}$	[15]

(<http://CRAN.R-project.org/package=takos>).

The package is installed into the R software by issuing the following commands in an R terminal:

```
install.packages('takos')
library(takos)
```

The package includes a generic function that can simulate thermoanalytic data according to the reaction models presented in Table 1 that can be invoked using the command `gAC`. It performs calculations using the function `LSODA`, which is a part of the `deSolve` package [21]. The `LSODA` function provides an interface to the Fortran ordinary differential equation (ODE) solver of the same name. It switches automatically between stiff and non-stiff methods and employs the following families of linear multi-step methods: Adams methods (interpolation of the past dy/dx values) for solving the non-stiff problems and the backward differentiation formulae (interpolation of the past y values) for solving the stiff problems.

For this function the user needs to input:

- the time of starting of the simulation (`time.start`)
- the desired starting temperature (`T0`) and the desired ending temperature (`T.end`) for the simulation

- the heating rate (expressed in degrees Celsius/min or in Kelvin/min) (`qqq`)
- the pre-exponential parameter A expressed in $1/s$ (`A`)
- E_α , the activation energy expressed in J/mol (`Ea`)
- the parameters m, n, K of the chosen model (`m, n, K`)
- the number of points for the simulated thermoanalytic curve (`npoints`)
- the starting 0 value for the simulation, that will affect the precision of calculations [22] (`prec`)
- the user selected model (`rmod`)

When we invoke the command `gAC` with the default parameters we have

```
gAC(time.start=0,
     T0=0, T.end=500,
     qqq=50,
     A=10^(6.3),
     Ea=80000,
     m=1, n=2, K=0,
     npoints=10000,
     prec=10^(-4.30095790876), rmod="SB",
     ...)
```

Also, due to the frequent use of the Johnson–Mehl–Avrami and truncated Šesták Berggren, these models have two dedicated functions called JMA and SB.

After the importing or simulation of a series of thermoanalytic curves, the data need to be converted to a data table using the command `data.table`, then checked via the function `checkmat`, included in *takos*, before being able to be analysed.

In both cases the imported files must include a column with the time of acquisition (expressed in seconds or minutes), a column for the sample temperature (in degrees Celsius or Kelvin) and one column with the value of the signal (expressed in mW).

Each methodology for finding the kinetic triplet can be invoked using:

`METHOD(mat, id, degree)` where the arguments are defined as:

- the matrix of all the thermoanalytic data, `mat`, checked using the function `mat.check`
- the variable chosen for subsetting the matrix (default = "rate"), `id`, alternatively the number of the thermoanalytic data-curve or another numeric value
- `degree`, selected degrees of crystallinity for performing the analysis.

The name of the methods implemented in *takos* are the following: FRI (Friedman), KAS (Kissinger–Akahira–Sunose), Kiss (Kissinger), MO (Mo), OFW (Ozawa–Flynn and Wall), OZ (Ozawa), VY (Vyazovkin isoconversional method).

As output we obtain a model `mod`, a `data.table`, a table `xy` that let us construct the default plot for each model, the activation Energy, `Ea`, and a `data.table`, `DT`, built with the values of `mat` according to the specified degree used for calculating the model.

The package has a methodology for calculating the baseline in differential thermal scans (tangent area proportional baseline) that was also later included in the package *baseline* due to its feasibility of use in signals originating from other physical chemical analytical techniques (i.e. X-ray photoelectron spectroscopy XPS, Mossbauer spectroscopy etc.).

Finally, functions for creating summaries with all the values of each model and to create standard plots for each model calculated are available in the package.

3. Illustrative example

As a demonstration we will report a generic script used for creating part of the data sets and the analysis reported in [23]. Briefly, theoretically simulated kinetic data were used to evaluate the performance of the most common isoconversional methods of kinetic analysis in complex-process scenarios with two independent overlapping processes exhibiting nucleation-growth kinetics, and further expand the conclusions for the autocatalytic kinetic processes with positive asymmetry.

The performance was evaluated by means of the Friedman, KAS, Starink, OFW, Vyazovkin and incremental modified Vyazovkin methods.

After loading the required libraries, we can launch the following script that will perform the analysis:

```
rates=c(0.5,1,2,5,10,20,50)
# The rates chosen for the simulation

# Application of the model JMA with the selected parameters:
a1 <- lapply(rates, function(x) JMA(A=exp(35), Ea=120000,
  T0=-100, T.end=300,
  q=x, npoints=10000, n=2))

a2 <- lapply(rates, function(x) JMA(A=exp(14), Ea=60000,
```

```
T0=-100, T.end=300,
q=x, npoints=10000, n=1))
```

Conversion to data.table and merging of the curves:

```
a1 <- lapply(seq(1,length(a1)),
  function(x) data.table(a1[[x]]$time.s,
    a1[[x]]$T.C,
    a1[[x]]$dadT,
    rates[[x]]))
```

```
lapply(seq(1,length(a1)),
  function(x) setnames(a1[[x]],
    c("time.seconds",
      "temperature.s",
      "heat.flow",
      "rates"))) )
```

```
a2 <- lapply(seq(1,length(a2)),
  function(x) data.table(a2[[x]]$time.s,
    a2[[x]]$T.C,
    a2[[x]]$dadT,
    rates[[x]]))
```

```
lapply(seq(1,length(a2)),
  function(x) setnames(a2[[x]],
    c("time.seconds",
      "temperature.s",
      "heat.flow",
      "rates"))) )
```

```
a <- lapply(seq(1,length(a1)),
  function(x) data.table(a1[[x]]$time.seconds,
    a1[[x]]$temperature.s,
    (a1[[x]]$heat.flow*0.66)+
    (a2[[x]]$heat.flow*0.33),
    rates[[x]]))
```

```
lapply(seq(1,length(a)),
  function(x) setnames(a[[x]],
    c("time.seconds",
      "temperature.s",
      "heat.flow",
      "rates"))) )
```

In order to visually inspect the simulated data, we can write:

```
windows() # to open a new window

# To calculate the maximum value of y:
amaxH <- max(sapply(a, function(x) max(x$heat.flow)))

# To plot the mixed curves:
thermo(c(-100,300), c(0,amaxH),
  mytitle="Dataset A 120/60 0.66/0.33")
lapply(a, function(x) lines(x$temperature.s,
  x$heat.flow, lwd=3))
lapply(a1, function(x) lines(x$temperature.s,
  x$heat.flow*0.66, col="red"))
lapply(a2, function(x) lines(x$temperature.s,
  x$heat.flow*0.33, col="blue"))
```

After preparing the matrix we can apply the methodologies.

```
# Setting the parameters for the analysis:
mystep <- 0.01 # chosen step size for the alpha to be investigated
mydegree <- seq(0.005, 0.995, by = mystep)

aF <- FRI(ap, degree=mydegree) # Friedman method
aKAS <- KAS(ap, degree=mydegree) # Kissinger-Akahira-Sunose method
```

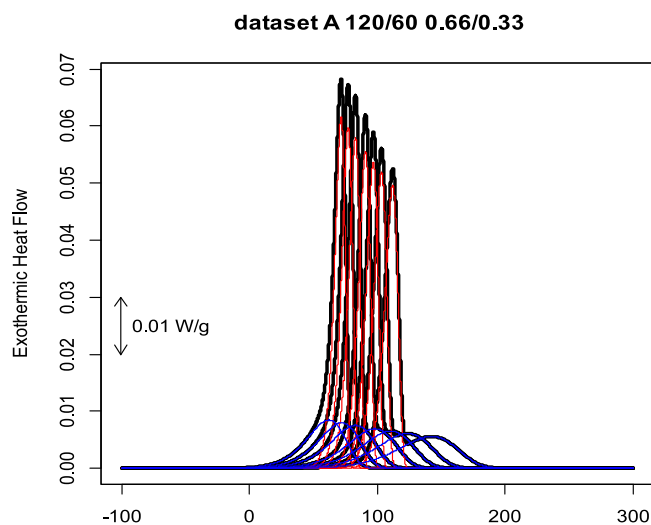


Fig. 1. Activation Energy vs degree of transformation (i.e. crystallization).

```
aStarink <- Starink(ap,degree=mydegree) # Starink method
aOFW <- OFW(ap,degree=mydegree) # Ozawa-Flynn and Wall method
aOZ <- OZ(ap,n.step=1, spks=1, eps=0.001) # Ozawa method
```

The summary for each model can be presented with the following commands:

```
aF.Table <- summaryTableF(aF)
aKAS.Table <- summaryTableF(aKAS)
aStarink.Table <- summaryTableF(aStarink)
aOFW.Table <- summaryTableF(aOFW)
aOZ.Table <- summaryTableOz(aOZ)
aKiss.Table <- data.frame(summaryTableK(aKiss))
```

As an example the table for the model according to Friedman is reported:

rit	term	estimate	std.error	statistic	p.value	Ea.dup
1:	0.01 (Intercept)	38.08	0.61	62.08	2.59e-04	125.25
2:	0.01 x	-15.07	0.22	-69.70	2.06e-04	
3:	0.02 (Intercept)	36.55	0.34	106.29	1.84e-06	119.88
4:	0.02 x	-14.42	0.12	-118.61	1.32e-06	
5:	0.03 (Intercept)	34.97	0.20	172.03	3.38e-05	114.88

194:	0.97 x	-11.64	1.52	-7.67	6.02e-04	
195:	0.98 (Intercept)	24.44	4.18	5.85	2.07e-03	96.75
196:	0.98 x	-11.64	1.52	-7.67	6.02e-04	
197:	0.99 (Intercept)	24.44	4.18	5.85	2.07e-03	96.75
198:	0.99 x	-11.64	1.52	-7.67	6.02e-04	

Results can be easily plotted with standard R commands (Fig. 1).

As can be seen in the reported results, the simulation of the data was successful in obtaining the curves with the given parameters.

These curves were generated using the “JMA” methodology. For the paper previously cited, we performed different simulations in batches changing the parameters at the beginning of the process while leaving the remaining workflow the same. This had the double advantage to decreasing the time needed for the study (in comparison with a less automated approach) and to avoid any mistake that could have arisen from an operator while conducting tedious repetitive tasks. Furthermore, for a quick batch comparison with other methodology of simulations, we would have just needed to change the method while invoking the `gAC` function.

4. Impact

The package *takos* will let researcher perform calculations with unprecedented precision and so study in depth kinetics processes on thermograms already collected. It will speed up the workflow letting users take advantage of a scripting language. Once implemented, scripts can be used in a laboratory and routinely applied to collected data, shifting the focus to the analysis of the results obtained instead on their calculation.

We have received positive feedback while presenting our work in congresses and several colleagues, that actually rely on proprietary software, are interested in trying the software in order to compare it with their present work flow. The package *takos* was also used by the authors for performing all the data analysis for the following peer reviewed published articles [22–26] and it is routinely used as a diagnostic tool at the SCITEC laboratory. The diagnostic available on CRAN reports that it has been downloaded almost 7000 times.

5. Conclusions

In this paper, we have presented *takos*, an R-based, flexible, and powerful collection of functions for differential and integral methods dedicated to the kinetic analysis of data obtained by Differential Scanning Calorimetry.

Since the software R allows for creation of complex and flexible data processing routines that are not available in existing programs, it gives the user the possibility to perform calculation with unprecedented precision and to avoid pitfalls due to the repetitive task needed while following a workflow which includes a non-scriptable approach.

The kinetic models provided in the package may prove useful in characterizing a wide range of processes, e.g. curing, polymerization, and effects on changing of synthesis parameters of a material, which are the fundamental aspects of thermal kinetic analysis.

The package *takos* is an ongoing project. We intend to improve its performance using specific architectures (Compute Unified Device Architecture) and to extend its features and functionality by implementing pre-processing functions for the thermoanalytic data and implementing additional methodologies for kinetic analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] Stodden V, Leisch F, Peng RD. Implementing reproducible research. Chapman and Hall/CRC; 2014. <http://dx.doi.org/10.1201/b16868>.
- [2] Vyazovkin S, Chrissafis K, Lorenzo MLD, Koga N, Pijolat M, Roduit B, et al. ICTAC Kinetics Committee recommendations for collecting experimental thermal analysis data for kinetic computations. *Thermochim Acta* 2014;590:1–23. <http://dx.doi.org/10.1016/j.tca.2014.05.036>.
- [3] Vyazovkin S. Kinetic effects of pressure on decomposition of solids. *Int Rev Phys Chem* 2020;39(1):35–66. <http://dx.doi.org/10.1080/0144235X.2019.1691319>.
- [4] Johnson WA, Mehl RF. Reaction kinetics in processes of nucleation and growth. *Trans Am Inst Min Metall Eng* 1939;135:416–42.
- [5] Kolmogorov A. On the statistical theory of the crystallization of metals. *Bull Acad Sci USSR Math Ser* 1937;3:355.
- [6] Kissinger HE. Reaction kinetics in differential thermal analysis. *Anal Chem* 1957. <http://dx.doi.org/10.1021/ac60131a045>.
- [7] Friedman HL. Kinetics of thermal degradation of char-forming plastics from thermogravimetry. Application to a phenolic plastic. *J Polym Sci C* 1964;6(1):183–95. <http://dx.doi.org/10.1002/polc.5070060121>.

- [8] Avrami M. Kinetics of phase change. I general theory. *J Chem Phys* 1939;7(12):1103–12. <http://dx.doi.org/10.1063/1.1750380>.
- [9] Ozawa T. A new method of analyzing thermogravimetric data. *Bull Chem Soc Japan* 1965;38(11):1881–6. <http://dx.doi.org/10.1246/bcsj.38.1881>.
- [10] Flynn JH, Wall LA. General treatment of the thermogravimetry of polymers. *J Res Natl Bur Stand A: Phys Chem* 1966. <http://dx.doi.org/10.6028/jres.070a.043>.
- [11] Akahira T, Sunose T. Joint convention of four electrical institutes. *Res Rep Chiba Inst Technol (Sci Technol)* 1971;16:22–31.
- [12] Liu T, Mo Z, Wang S, Zhang H. Nonisothermal melt and cold crystallization kinetics of poly(ary1 ether ether ketone ketone). Tech. rep., 1997.
- [13] Vyazovkin S. Evaluation of activation energy of thermally stimulated solid-state reactions under arbitrary variation of temperature. *J Comput Chem* 1997;18(3):393–402.
- [14] Vyazovkin S. Modification of the integral isoconversional method to account for variation in the activation energy. *J Comput Chem* 2001. [http://dx.doi.org/10.1002/1096-987x\(20010130\)22:2<178::aid-jcc5>3.0.co;2-%23](http://dx.doi.org/10.1002/1096-987x(20010130)22:2<178::aid-jcc5>3.0.co;2-%23).
- [15] Starink MJ. Kinetic equations for diffusion-controlled precipitation reactions. *J Mater Sci* 1997. <http://dx.doi.org/10.1023/A:1018649823542>.
- [16] Svoboda R, Málek J. Is the original Kissinger equation obsolete today? *J Therm Anal Calorim* 2014;115(2):1961–7. <http://dx.doi.org/10.1007/s10973-013-3486-4>, URL <https://link.springer.com/article/10.1007/s10973-013-3486-4>.
- [17] Vyazovkin S. Kissinger method in kinetics of materials: Things to beware and be aware of. *Molecules* 2020;25(12):2813. <http://dx.doi.org/10.3390/molecules25122813>, URL <https://www.mdpi.com/1420-3049/25/12/2813>.
- [18] Vyazovkin S, Burnham AK, Criado JM, Pérez-Maqueda LA, Popescu C, Sbirrazzuoli N. ICTAC kinetics committee recommendations for performing kinetic computations on thermal analysis data. 2011, <http://dx.doi.org/10.1016/j.tca.2011.03.034>.
- [19] Svoboda R, Luciano G. Complex process activation energy evaluated by combined utilization of differential and integral isoconversional methods. *J Non-Crystall Solids* 2020;535:120003. <http://dx.doi.org/10.1016/j.jnoncrysol.2020.120003>.
- [20] Vyazovkin S. Modern isoconversional kinetics: From misconceptions to advances. In: *Handbook of thermal analysis and calorimetry*, vol. 6. Elsevier B.V.; 2018, p. 131–72. <http://dx.doi.org/10.1016/B978-0-444-64062-8.00008-5>.
- [21] Soetaert KE, Petzoldt T, Setzer RW. Solving differential equations in R: Package deSolve. *J Stat Softw* 2010;33(9). <http://dx.doi.org/10.18637/jss.v033.i09>.
- [22] Luciano G, Soetaert K, Svoboda R. Simulation and non-linear optimization of Šesták–Berggren kinetics. *J Non-Crystall Solids* 2020;550:120391. <http://dx.doi.org/10.1016/j.jnoncrysol.2020.120391>.
- [23] Svoboda Luciano. Activation energy determination in case of independent complex kinetic processes. *Processes* 2019;7(10):738. <http://dx.doi.org/10.3390/pr7100738>, URL <https://www.mdpi.com/2227-9717/7/10/738>.
- [24] Ahmed J, Luciano G, Schizzi I, Arfat YA, Maggiore S, Thai TLA. Non-isothermal crystallization behavior, rheological properties and morphology of poly (ϵ -caprolactone)/graphene oxide nanosheets composite films. *Thermochim Acta* 2018;659:96–104.
- [25] Ahmed J, Luciano G, Maggiore S. Nonisothermal crystallization behavior of polylactide/polyethylene glycol/graphene oxide nanosheets composite films. *Polym Compos* 2020;41(5):2108–19.
- [26] Svoboda R, Luciano G. Complex process activation energy evaluated by combined utilization of differential and integral isoconversional methods. *J Non-Crystall Solids* 2020;535:120003.