



Original software publication

RelaxPy: Python code for modeling of glass relaxation behavior

Collin J. Wilkinson*, Yihong Z. Mauro, John C. Mauro

Department of Materials Science and Engineering, The Pennsylvania State University, University Park, PA 16802, USA



ARTICLE INFO

Article history:

Received 19 June 2018

Received in revised form 26 July 2018

Accepted 26 July 2018

Keywords:

Glass

Thermodynamics

Viscosity

Relaxation

Fictive temperature

Kinetics

ABSTRACT

The degree of relaxation in any glass sample is a governing property in every property of the glass. It plays an important role in every major glass product commercially available, but has required individual groups to develop their own relaxation codes. RelaxPy is a Python-based script designed to be used in a Python interpreter or Linux terminal. Given an input temperature path and set of material properties including the nonequilibrium viscosity parameters, RelaxPy returns the evolution of the composite fictive temperature, viscosity, and relaxation time. Optionally, the software can also return the individual values of the fictive temperature components using a Prony series fit to approximate the stretched exponential relaxation form. RelaxPy aims to provide a flexible, open-source framework for glass relaxation modeling where new advances in glass physics can be easily incorporated and shared with the community.

© 2018 The Authors. 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 | v0.1 |
| Permanent link to code/repository used for this code version | https://github.com/ElsevierSoftwareX/SOFTX_2018_90 |
| Legal Code License | GNU GPL v3 |
| Code versioning system used | none |
| Software code languages, tools, and services used | Python, numpy, matplotlib, and scipy |
| Compilation requirements, operating environments | [NA] |
| If available Link to developer documentation/manual | [NA] |
| Support email for questions | cxw824@psu.edu |

1. Motivation and significance

Glasses are inherently nonequilibrium materials that spontaneously relax toward the metastable supercooled liquid state [1]. The functional form for glass relaxation dates back to 1854 when Kohlrausch [2] fit the residual charge of a Leyden jar with

$$g(t) = g(0) \exp[-(t/\tau)^\beta], \quad (1)$$

where t is time, τ is the relaxation time constant, β is the stretching exponent, and $g(t)$ is some property that is effected by relaxation (e.g., volume, conductivity, viscosity, etc.); Eq. (1) is referred to as the stretched exponential relaxation function (SER). The stretching exponent is bounded from $0 < \beta \leq 1$, where the upper limit ($\beta = 1$) represents a simple exponential decay and fractional values of β represent stretched exponential decay with a fat tail [3,4].

For over a century the physical origin of the stretched exponential relaxation form was one of the greatest unsolved problems in

physics, until 1982 when Grassberger and Proccacia were able to derive the general form based on a model of randomly distributed traps that annihilate excitations and the diffusion of the excitations through a network [5]. This model, however, provided no physical meaning for β , which was still treated as an empirical fitting parameter. In 1994, Phillips was able to extend the diffusion trap model by showing that at high temperatures $\beta = 1$ and at low temperatures the stretching exponent can be derived based on the effective dimensionality of available relaxation pathways [6]. He in turn expressed the stretching exponent

$$\beta = \frac{fd}{fd + 2}, \quad (2)$$

where d is the dimensionality of the network and f is the fraction of relaxation pathways available. He then proposed a set of 'magic' numbers for common scenarios. Assuming a three dimensional network with all pathways activated ($d = 3, f = 1$), a value of $\beta = 3/5$ is obtained. The second case is a three dimensional network with only half of the relaxation pathways activated ($d = 3, f = 1/2$) yielding a value of $\beta = 3/7$. A third magic value $\beta =$

* Corresponding author.

E-mail address: collin.wilkinson1123@gmail.com (C.J. Wilkinson).

1/2 was found for a two dimensional technique with all pathways activated ($d = 2, f = 1$). $\beta = 3/5$ occurs for stress relaxation of glasses under a load, because both long and short range activation pathways are activated, whereas a value of $\beta = 3/7$ is obtained for structural relaxation of a glass without an applied stress. The value was confirmed by Welch et al. [7] when the value was measured over a period of 1.5 years at more than 600 °C below the glass transition temperature of Corning Gorilla Glass®.

Due to the fundamental nature of glass being a nonequilibrium material, classical reversible thermodynamics cannot describe systems of glass networks [3,7–13]. To account this problem, an extra thermodynamic variable (or order parameter), called fictive temperature (T_f) is used [4,8,14,15]. Early research on relaxation in glasses dates back to the work of Tool and Eichlin in 1932, and Tool in 1946 whose works originally proposed a temperature at which a glass system could be in equilibrium without any atomic rearrangement [16,17]. Tool suggested that this fictive temperature (T_f) was sufficient to understand the thermodynamics of a glassy system. Originally the fictive temperature was treated as a single value that was some function of thermal history ($T_f[T(t)]$) [16,17]; however, subsequent experiments have shown that the concept of a singular fictive temperature is inadequate [4].

A key experiment was performed by Ritland in 1956 [18]. Ritland took several samples with different thermal histories but the same measured fictive temperature; thus, based on Tool's equation, both should have had identical relaxation properties. Ritland, however, showed that the refractive index evolved differently between samples. To account for these differences, Ritland proposed the use of multiple fictive temperatures [18]. This experiment showed that materials held isothermally at their fictive temperatures with varying thermal histories can give varying results for the relaxation of a given property ($T_f(T)$) [16–19]. Thus as a glass network relaxes (atoms shifting toward equilibrium positions) the fictive temperature will also shift [4,7–9]. Narayanaswamy was able to apply Ritland's concept in a highly successful engineering model, but the physical meaning of multiple fictive temperatures remained elusive [19].

The current state-of-the-art in modeling of glass relaxation behavior considers a weighted average of the fraction of the network (w_i) that is associated with a given fictive temperature component (T_{fi}) [8] given by

$$T_f = \sum_{i=1}^N w_i T_{fi}. \quad (3)$$

The values for w_i are defined by a Prony series approximation [4] of Eq. (1)

$$\exp(-x^b) \simeq \sum_{i=1}^N w_i \exp(-K_i x), \quad (4)$$

in order to make the differentials analytically solvable (the fitting method is described by Mauro and Mauro [4]. This leads to a set of N coupled differential equations that represent the change in fictive temperature as a function of thermal history as expressed by:

$$\frac{dT_{fi}}{dt} = \frac{T(t) - T_{fi}(t)}{\tau_i(T(t), T_f(t))}, \quad i = 1 \dots N. \quad (5)$$

The relaxation time of each component ($\tau_i[T(t), T_f(t)]$) is a function composition, temperature, and thermal history and is given by

$$\tau_i(T, T_f) = \frac{\tau_k(T, T_f)}{K_i}. \quad (6)$$

K_i is a factor that comes from the Prony series approximation of the SER function [4] and $\tau_k(T, T_f)$ is the common relaxation factor

scaled by the Maxwell relation,

$$\tau_k(T, T_f) = \frac{\eta(T, T_f)}{K_s} \quad (7)$$

where K_s is the shear modulus and is typically treated as a fitting parameter. The $\eta(T, T_f)$ function from Eq. (7) is the non-equilibrium viscosity [13] as given by the Mauro–Allan–Potuzak non-equilibrium viscosity model (MAP equation) and is given by

$$\log_{10}\eta(T, T_f) = x \log_{10}\eta_{eq}(T_f) + (1 - x) \log_{10}\eta_{ne}(T, T_f) \quad (8)$$

where $\eta_{eq}(T_f)$ is the equilibrium viscosity given by the MYEGA equation for equilibrium viscosity, $\eta_{ne}(T, T_f)$ is the nonequilibrium contribution to viscosity, and x is an ergodicity parameter defined by

$$x = \left(\frac{\min(T, T_f)}{\max(T, T_f)} \right)^p. \quad (9)$$

Here, p is a constant related to the glass transition range and represents how sharply the ergodicity breaks down; it scales linearly with fragility. The final part of the equation is the nonequilibrium contribution to viscosity [13],

$$\log_{10}\eta_{ne}(T, T_f) = A + \frac{\Delta H}{k \ln 10} - \frac{S_\infty}{k \ln 10} \exp\left[-\frac{T_g}{T_f} \left(\frac{m}{12 - \log_{10}\eta_\infty}\right)\right]. \quad (10)$$

The values for A , $\frac{\Delta H}{k \ln 10}$, η_∞ , and p are all default values (that can also be specified if desired) while the fragility (m), the glass transition temperature (T_g), and $\frac{S_\infty}{k \ln 10}$ are all inputs to the code. The values and the calculations of these parameters are further discussed in Guo et al. [8].

2. Software description

RelaxPy script is written using Python syntax and relies on the numpy, scipy, and matplotlib libraries. It is designed to be run within a Linux terminal with command line options or inside a Python interpreter. It can be used by calling the name of the script, followed by the input file, the desired output file name, and finally an optional tag for all fictive components to be displayed as well as printed to a specified output file (formatted as comma separated values).

2.1. Software architecture

The RelaxPy package consists of an algorithm that iterates over time in order to determine the values of viscosity (η), relaxation time (τ), and the current fictive temperature (T_f) given the user-supplied thermal history and material property values. The purpose of this code is to enable calculation of the relaxation behavior's change over time by using the set of differential equations shown in Eq. (5), using only easily-attainable parameters.

Eq. (5) is solved for each term in the Prony series finding the change in each T_{fi} component for a given time (t). The program then iterates over the entire thermal history of the sample using a user-specified time step dt , allowing for calculation of the overall fictive temperature over time, as shown in Eq. (3). The values for the w_i and K_i are taken from the database originally generated by Mauro and Mauro [4] with their hybrid fitting method based on the number of terms in the desired Prony series and the magic value chosen from Phillips [6]. The thermal history is also defined in the original input file using linear interpolation (an example input file can be found the illustrative examples section); an example is shown in Fig. 1. The initial fictive temperature components are assumed to be at equilibrium ($T_{fi} = T(0)$).

Download English Version:

<https://daneshyari.com/en/article/6964944>

Download Persian Version:

<https://daneshyari.com/article/6964944>

[Daneshyari.com](https://daneshyari.com)