Rheologica Acta Rheol Acta 32:322–327 (1993)

Generating line spectra from experimental responses. Part II: Storage and loss functions

N. W. Tschoegl[1] and I. Emri[2]

[1] California Institute of Technology, Pasadena, California, USA
[2] University of Ljubljana, Ljubljana, Slovenia

Abstract: A computer algorithm is described which allows the determination of a discrete distribution of relaxation times from simulated or smoothed storage or loss modulus data, or of retardation times from simulated or smoothed storage or loss compliance data. The distributions faithfully reproduce the input data and are suitable for data storage as well as for generating any other response curves.

Key words: Algorithm – line spectrum – loss compliance – loss modulus – storage compliance – storage modulus – relaxation spectrum – retardation spectrum

Introduction

This paper is a companion to the preceding one (Emri and Tschoegl, 1993a). The latter presented a general background, briefly reviewed other work in this area, and provided the general theory of our algorithm for the determination of respondance (relaxation or retardation) time distributions in linear viscoelastic materials. It then discussed the form of the algorithm suitable for obtaining a discrete distribution of relaxation times from simulated or smoothed relaxation modulus data, or of retardation times from simulated or smoothed creep compliance data. For brevity, we shall refer to this form of the algorithm as the *transient algorithm*. This companion paper discusses the *storage algorithm* and the *loss algorithm*, i.e., the forms of the algorithm in which it applies to simulated or smoothed data on the storage and loss functions. As in the preceding paper, we demonstrate the power of the algorithm here using the standard linear solid and liquid models, and the bimodal form of the Kobeko equation, there given by Eq. (29). Again, the use of simulated data allows us to avoid complications arising from the presence of experimental error. (See the companion paper for details). Another paper (Emri and Tschoegl, 1993b) discusses the application of the algorithm in the presence of scatter in experimental data.

Theoretical

The change from the exponential to the half-Lorentzian and Lorentzian kernels, i.e., from $\exp(-t/\tau)$ to $1/(1+\omega^2\tau^2)$ and $\omega\tau/(1+\omega^2\tau^2)$, necessitates changes in the form of the algorithm that we described in the companion paper (Emri and Tschoegl, 1993a). We begin by discussing the modifications required when adapting the algorithm for use with the storage kernel, $1/(1+\omega^2\tau^2)$, instead of the transient kernel, $\exp(-t/\tau)$. Both kernels are monotone non-increasing functions of their arguments. The modifications to be introduced result primarily from the need to re-evaluate the limits of the two intervals that we have called the Boundary Window, or Window 1, and the Modeling Window, or Window 2, in the companion paper. We recall from that paper that Window 1 determines the maximum allowable width for the region from which the algorithm selects datum points for the calculation of each successive spectrum line. The respective Windows 1 are compared in Fig. 1 together with the two kernels.

The kernel function is capable of modeling the storage modulus effectively only in the transition region where the first derivative is significantly larger than zero. As can be seen from Fig. 1, this condition is well satisfied in the region from about $\log \omega\tau_k = -0.5$ to 0.5 which now defines Window 1. The limits

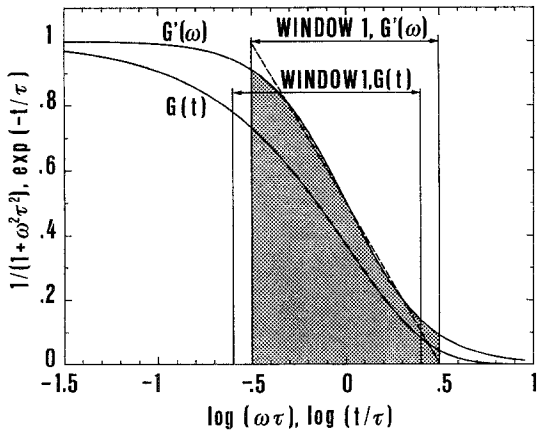

Fig. 1. The exponential and half-Lorentzian kernels and Windows 1 for both kernels

are symmetrical because the half-Lorentzian kernel is self-congruent (Tschoegl, 1989, p. 318). Within Window 1 the kernel function can be well modeled by a straight line when plotted semilogarithmically.

Another re-evaluation is needed for Window 2 which demarcates the region in which the algorithm works most efficiently when neighboring kernels overlap (Emri and Tschoegl, 1993a). The negative inverse of the logarithmic derivative of the storage kernel, defined by

$$D_{\mathrm{stor}}(\omega\tau) = -\frac{d}{d\ln\omega\tau}\left(\frac{1}{1+\omega^2\tau^2}\right) = \frac{4.605\,\omega^2\tau^2}{(1+\omega^2\tau^2)^2}\,, \tag{1}$$

is plotted in Fig. 2 for three neighboring spectrum lines separated from each other by one half of a logarithmic decade.

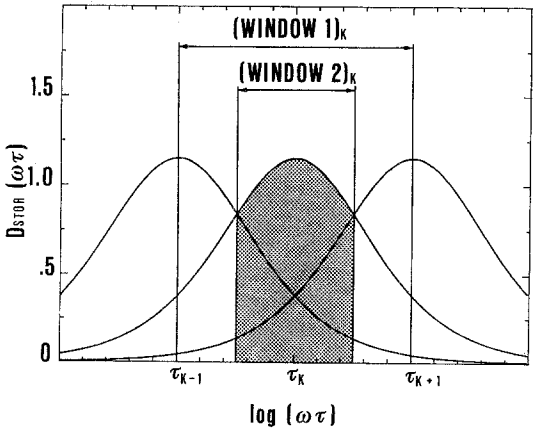

Fig. 2. Three neighboring storage kernels and Windows 1 and 2

The intersections of the derivatives define the width of Window 2. The lower and upper limits of the window, ω_1, and ω_2, are obtained from

$$\omega_1^2 = 1/\tau_{k-1}\tau_k = 10^{-1/r}/\tau_k^2 \tag{2}$$

and

$$\omega_u^2 = 1/\tau_k\tau_{k+1} = 10^{1/r}/\tau_k^2\,, \tag{3}$$

where r, the number of preselected spectrum lines per decade of $\log\omega$, is given by

$$r = \frac{1}{\log(\tau_{k-1}/\tau_k)} = \frac{1}{\log(\tau_k/\tau_{k+1})}\,. \tag{4}$$

The derivation of these equations is analogous to that given in the companion paper (Emri and Tschoegl, 1993a). Values for $r = 1$ to $r = 8$ for the storage algorithm are tabulated in Table 1. As in the case of the transient algorithm, for a single preselected spectrum line per decade Window 1 and Window 2 coincide.

In the transient algorithm, one is likely to encounter modeling and computational problems outside of Window 1. Such problems arise when the numbers the computer is called to handle become exceedingly small. Here, because of the more gradual decline of the kernel, these problems are less likely to occur. If the number of preselected spectrum lines is increased too much, the interval may not contain any datum points. In that case the width of the window must be increased.

We now turn to a description of the loss algorithm. For the Lorentzian loss kernel the situation is a little more complicated than that which we have just discussed. This kernel is not a monotone non-increasing function of its argument but possesses a maximum as shown in Fig. 3. Consequently, to be able to do its work, the loss algorithm needs two Windows 1 and 2, one on each side of the maximum.

The left window is used in modeling the loss modulus to the left of its maximum. Conversely, the right window is used to the right of the peak. The two

Table 1. Limits of Window 2 for the storage algorithm

r	$\log\omega_l\tau_k$	$\log\omega_u\tau_k$	r	$\log\omega_l\tau_k$	$\log\omega_u\tau_k$
1	−0.50	0.50	5	−0.10	0.10
2	−0.25	0.25	6	−0.08	0.08
3	−0.17	0.17	7	−0.07	0.07
4	−0.13	0.12	8	−0.06	0.06

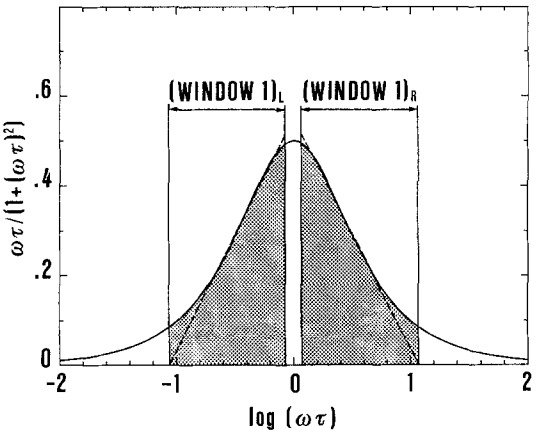

Fig. 3. The Lorentzian kernel at the two Windows 1

Windows 1, as shown in Fig. 3, span the regions from $\log \omega\tau = -1.07$ to -0.07, and from $\log \omega\tau = 0.07$ to 1.07, respectively. These endpoints, as before, limit the region within which the kernel function, when plotted semilogarithmically, can be reasonably approximated by a straight line. In this region the first derivate is significantly larger than zero.

The two Windows 2 are shown in Fig. 4 along with plots of the first logarithmic derivatives of the loss kernel,

$$D_{\text{loss}}(\omega\tau) = \frac{d}{d\ln\omega\tau}\left(\frac{\omega\tau}{1+\omega^2\tau^2}\right)$$

$$= \frac{2.303\,\omega\tau(1-\omega^2\tau^2)}{(1+\omega^2\tau^2)^2} , \qquad (5)$$

three neighboring spectrum lines, again separated by one half of a logarithmic decade.

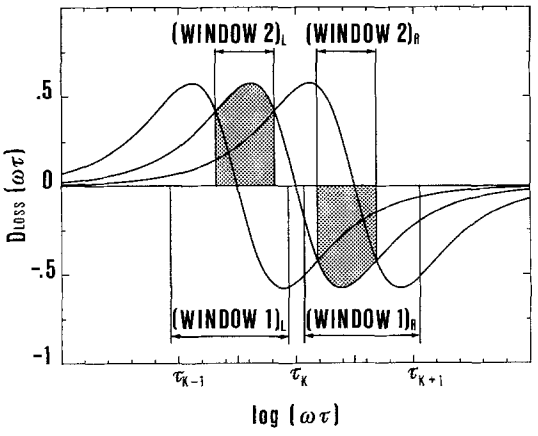

Fig. 4. Three neighboring loss kernels and the two Windows 2

The lower and upper limits for Windows 2 for two preselected spectrum lines per logarithmic decade are obtained by proceeding analogously to the two previous cases. In the present case, however, ω_1^2 must be obtained from solving the equation

$$A\omega^6 + B\omega^4 + C\omega^2 + D = 0 , \qquad (6)$$

a cubic equation in ω^2. For the lower limit the parameters A, B, C, and D are

$$A = \tau_{k-1}^3\tau_k^4 - \tau_{k-1}^4\tau_k^3 \qquad (7a)$$

$$B = \tau_{k-1}^4\tau_k - 2\tau_{k-1}^2\tau_k^3 + 2\tau_{k-1}^3\tau_k^2 - \tau_{k-1}\tau_k^4 \qquad (7b)$$

$$C = \tau_{k-1}^3 - 2\tau_{k-1}\tau_k^2 + 2\tau_{k-1}^2\tau_k - \tau_k^3 \qquad (7c)$$

$$D = \tau_k - \tau_{k-1} . \qquad (7d)$$

Into these we introduce the number, r, of spectrum lines per decade via Eq. (4). Equation (6) then yields the lower limits for both Windows 2. The upper limits are obtained in the same way by replacing τ_{k-1} by τ_{k+1}. The values for $r = 1$ to $r = 8$ for the right Window 1 are tabulated in Table 2 below. Those for the left window are obtained by replacing the positive with the negative sign throughout.

We are now ready to discuss the two algorithms themselves.

The storage algorithm

Our starting point is the equation

$$G'(\omega) = G_g - \sum_{i=1}^{i=N} G_i \frac{1}{1+\omega^2\tau_i^2}$$

$$= \{G_e\} + \sum_{i=1}^{i=N} G_i \frac{\omega^2\tau_i^2}{1+\omega^2\tau_i^2} , \qquad (8)$$

where $\tau_i = \eta_i/G_i$, and G_i and η_i represent the modulus and the viscosity, respectively, of the i^{th} Maxwell unit. G_g is the glassy, and G_e is the equilibrium modulus. The braces signify that

Table 2. Limits of right Window 2 for the loss algorithm

n	$\log \omega_l\tau_k$	$\log \omega_u\tau_k$	n	$\log \omega_l\tau_k$	$\log \omega_u\tau_k$
1	0.07	1.07	5	0.29	0.49
2	0.18	0.68	6	0.31	0.47
3	0.24	0.57	7	0.32	0.46
4	0.27	0.52	8	0.32	0.45

$\{G_e\} = G_e$ when the modulus describes an arrheodictic[1]) material, and that $\{G_e\} = 0$ when the material is rheodictic[1]). We will use the form having the kernel $1/(1+\omega^2\tau^2)$ because its general characteristics parallel those of the exponential kernel, $\exp(-t/\tau)$.

Normalizing by the difference $G_g - \{G_e\}$ leads to

$$g'(\omega) = g_g - \sum_{i=1}^{i=N} g_i \frac{1}{1+\omega^2\tau_i^2} , \tag{9}$$

where $g'(\omega)$ and g_g are the normalized storage and glassy modulus, respectively, and the g_i's may be seen to represent the strengths of the delta functions in the normalized discrete relaxation spectrum

$$h(\tau) = \sum_{i=1}^{i=N} g_i \tau_i \delta(\tau - \tau_i) . \tag{10}$$

The source data are assumed to be available as a discrete set of M datum points $\{G'(\omega_j)\}$ where $j = 1,2,\ldots,M$. Each of these datum points can be normalized by the difference between the largest point, G'_M, and the smallest point, G'_1, to yield the set $\{\hat{g}'(\omega_j)\}$. G'_1 replaces G_g, and G'_M replaces $\{G_e\}$. The modulus can then be expressed alternatively by a discrete set of (normalized) spectrum lines, $\{\hat{g}_i\}$. In terms of these each datum point becomes

$$\hat{g}'(\omega_j) = \hat{g}'(\omega_M) - \sum_{i=1}^{i=N} \hat{g}_i \frac{1}{1+\omega_j^2\tau_i^2} . \tag{11}$$

We intend to determine, from the set of source data, $\{\hat{g}(\omega_j)\}$, a set of spectrum lines, $\{\hat{g}_i\}$, which will faithfully reproduce the modulus, $G'(\omega)$. We split the sum in Eq. (11) into three parts and write

$$\hat{g}'(\omega_j) = \hat{g}'(\omega_M) - \sum_{i=1}^{i=k-1} \hat{g}_i \frac{1}{1+\omega_j^2\tau_i^2} - \hat{g}_k \frac{1}{1+\omega_j^2\tau_i^2} - \sum_{i=k+1}^{i=N} \hat{g}_i \frac{1}{1+\omega_j^2\tau_i^2} + \Delta_j , \tag{12}$$

where we have separated out the k^{th} line, and added the term $\Delta_j = \Delta_j^e + \Delta_j^a$ to allow us to account for the experimental error in the source data, Δ_j^e, and the approximation error, Δ_j^a. The latter must be considered because the calculated spectral distribution is at best an approximation to the true spectrum.

We may reduce the effort (Emri and Tschoegl, 1993a) required to calculate $\hat{g}'(\omega_j)$ by beginning the first summation in Eq. (12) with $1 \le m = k - 4n - 1$. This yields

$$\hat{g}'(\omega_j) = \hat{g}'(\omega_M) - \sum_{i=m}^{i=k-1} \hat{g}_i \frac{1}{1+\omega_j^2\tau_i^2} - \hat{g}_k \frac{1}{1+\omega_j^2\tau_i^2} - \sum_{i=k+1}^{i=N} \hat{g}_i \frac{1}{1+\omega_j^2\tau_i^2} + \Delta_j . \tag{13}$$

We take four times the number of spectrum lines per decade, n, to cut off contributions from lines four decades downscale where their contribution has decreased to 10^{-8}. This value for m has been chosen for convenience. If it is considered to be too large, the value of m should be lowered.

The sum of squares of Δ_j within Window 2 is given by

$$E_k = \sum_{j=s_{k,l}}^{j=s_{k,u}} \Delta_j^2 , \tag{14}$$

where $s_{k,l}$ and $s_{k,u}$ are the first and the last discrete points in the window that belong to the k^{th} spectrum line. Minimizing the error according to $\partial E_k / \partial \hat{g}_k = 0$, leads to

$$\hat{g}_k = \frac{\sum_{j=s_{k,l}}^{j=s_{k,u}} \Phi'(\omega_j) \frac{1}{1+\omega_j^2\tau_k^2}}{\sum_{j=s_{k,l}}^{j=s_{k,u}} \left(\frac{1}{1+\omega_j^2\tau_k^2}\right)^2} , \tag{15}$$

where

$$\Phi'(\omega_j) = \hat{g}'(\omega_M) - \hat{g}'(\omega_j) - \sum_{i=m}^{i=k-1} \hat{g}_i \frac{1}{1+\omega_j^2\tau_i^2} - \sum_{i=k+1}^{i=N} \hat{g}_i \frac{1}{1+\omega_j^2\tau_i^2} . \tag{16}$$

Equation (15) is the expression from which the strength of the k^{th} spectrum line is to be obtained.

We start the computation with the Nth, i.e., the farthest right, spectrum line, $\hat{g}_N$, because in this way the first sum in Eq. (16) vanishes. The $\hat{g}_i$, $(i = m, \ldots, k-1)$, are set to zero in the first pass over the source data. In succeeding sweeps, we then substitute any newly found *non-negative* value for the corresponding previous one and set any negative value again to

[1]) The term *rheodictic* refers to a material showing steady-state flow. *Arrheodictic* then denotes a material which does not (Tschoegl, 1989, p. 93).

zero. The iteration is broken off when the difference between the previously found and the newly computed spectrum lines is smaller than a preset error.

The algorithm remains the same for the storage compliance,

$$J'(\omega) = J_g + \sum_{i=1}^{i=N} J_i \frac{1}{1+\omega^2\tau_i^2} , \tag{17}$$

since its kernel is identical with that in Eq. (1). For the normalization, we now use $J'_1 - J'_M$.

The loss algorithm

The algorithm to be applied to the loss modulus,

$$G''(\omega) = \sum_{i=1}^{i=N} G_i \frac{\omega\tau}{1+\omega^2\tau_i^2} , \tag{18}$$

is similar. Equations (10) and (14) remain the same. Equations (9) and (11) become

$$g''(\omega) = \sum_{i=1}^{i=N} g_i \frac{\omega\tau_i}{1+\omega^2\tau_i^2} \tag{19}$$

and

$$\hat{g}''(\omega_j) = \sum_{i=1}^{i=N} \hat{g}_i \frac{\omega_j\tau_i}{1+\omega_j^2\tau_i^2} , \tag{20}$$

the normalization factor now being $G''_{\max} = \max[G''(\omega)]$. For Eqs. (12) and (13), we now have

$$\hat{g}''(\omega_j) = \sum_{i=1}^{i=k-1} \hat{g}_i \frac{\omega_j\tau_i}{1+\omega_j^2 t_i^2} + \hat{g}_k \frac{\omega_j\tau_i}{1+\omega_j^2\tau_i^2} + \sum_{i=k+1}^{i=N} \hat{g}_i \frac{\omega_j\tau_i}{1+\omega_j^2\tau_i^2} + \Delta_j , \tag{21}$$

and Eq. (15) becomes

$$\hat{g}_k = \frac{\sum_{j=s_{k,l}}^{j=s_{k,u}} \Phi''(\omega_j) \frac{\omega_j\tau_k}{1+\omega_j^2\tau_k^2}}{\sum_{j=s_{k,l}}^{j=s_{k,u}} \left(\frac{\omega_j\tau_k}{1+\omega_j^2\tau_k^2}\right)^2} , \tag{22}$$

where

$$\Phi''(\omega_j) = \hat{g}''(\omega_j) - \sum_{i=1}^{i=k-1} \hat{g}_i \frac{\omega_j\tau_i}{1+\omega_j^2 t_i^2}$$

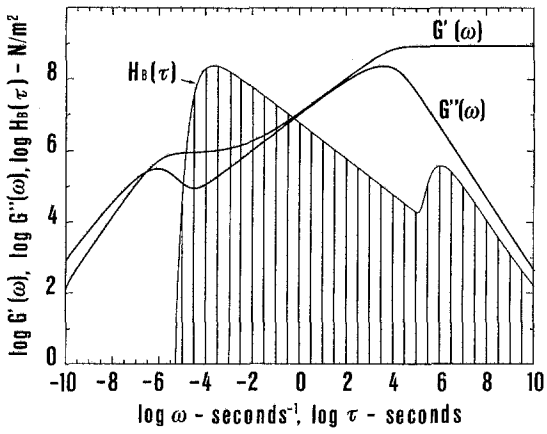

Fig. 5. $\log G'(\omega)$, $\log G''(\omega)$, $\log H_B(\tau)$, and the line spectrum obtained from the moduli

$$+ \sum_{i=k+1}^{i=N} \hat{g}_i \frac{\omega_j\tau_i}{1+\omega_j^2\tau_i^2} . \tag{23}$$

Here, too, we start the computation with the N^{th} spectrum line.

For the loss compliance, we apply the algorithm to

$$J''(\omega) - \{\phi_f/\omega\} = \sum_{i=1}^{i=N} J_i \frac{\omega\tau}{1+\omega^2\tau_i^2} , \tag{24}$$

where ϕ_f is the steady-state fluidity (the reciprocal of the steady-state viscosity), and the braces signify that the term is present when $J''(\omega)$ describes a rheodictic material, and is absent otherwise.

The normalization is carried out with respect to $J''_{\max} = \max[J''(\omega)]$.

Results

To demonstrate the applicability of these algorithms, we made use of the same line spectrum

$$H(\tau) = G_i\tau_i\delta(\tau_i - \tau) \quad i = 1,2,\ldots 32 \tag{25}$$

that we have used in the preceding paper (Emri and Tschoegl, 1993a). From this spectrum we obtained $G'(\omega)$ and $G''(\omega)$ using Eqs. (8) and (18). The curves are plotted in Fig. 5. We then applied the algorithm to both moduli. The line spectra obtained in both cases were virtually identical with one another, with the spectrum obtained in the preceding paper (Emri and Tschoegl, 1993a) from $G(t)$, and with the "original" spectrum $H(\tau)$. The latter is demonstrated by the fact that $H_B(\tau)$, from which $H(\tau)$ was derived (Emri and

Tschoegl, 1993a), envelops the line spectrum. This is shown in Fig. 5.

We conclude that our algorithm works just as well with the storage and loss functions as it does with the relaxation modulus and the creep compliance.

Acknowledgements

The authors gratefully acknowledge support of this work by the Slovene Ministry of Science under Grant P2-1131-782/91, and partial support by the California Institute of Technology.

References

1. Emri I, Tschoegl NW (1993a) Generating line spectra from experimental responses. I. Relaxation modulus and creep compliance. Rheol Acta 32:311–321
2. Emri I, Tschoegl NW (1993b) Generating line spectra from experimental responses. IV. Application to experimental data. To be submitted to Rheol Acta
3. Tschoegl NW (1989) The phenomenological theory of linear viscoelastic behavior. Springer, Berlin
4. Tschoegl NW, Emri I (1992) Generating line spectra from experimental responses. III. Interconversion between relaxation and retardation behaviour. Intern J Polym Mater 18:117–127

(Received August 24, 1992;
accepted April 1, 1993)

Correspondence to:

Prof. Igor Emri
Mechanical Engineering Department
University of Ljubljana
Murnikova 2
61000 Ljubljana, Slovenia

Prof. N.W. Tschoegl
California Institute of Technology
1201 East California Boulevard
Pasadena, CA 91125, USA