Material parameter estimation and hypothesis testing on a 1D viscoelastic stenosis model: Methodology

H.T. Banks, Shuhua Hu, and Z.R. Kenz

Center for Research in Scientific Computation

Center for Quantitative Sciences in Biomedicine

North Carolina State University, Raleigh, NC 27695-8212

Carola Kruse, Simon Shaw, and J.R. Whiteman

BICOM, Brunel University, Uxbridge, UB8 3PH, England

M.P. Brewin and S.E. Greenwald

Blizard Institute, Barts and the London School of Medicine and Dentistry,

Queen Mary, University of London, England

M.J. Birch

Clinical Physics, Barts and the London National Health Service Trust, England

April 9, 2012

Abstract

Non-invasive detection, location and characterization of an arterial stenosis (a blockage or partial blockage in the artery) continues to be an important problem in medicine. Partial blockage stenoses are known to generate disturbances in blood flow which generate shear waves in the chest cavity. We examine a one-dimensional viscoelastic model that incorporates Kelvin-Voigt damping and internal variables, and develop a proof-of-concept methodology using simulated data. We first develop an estimation procedure for the material parameters. We use that procedure to determine confidence intervals for the parameters found, which indicates the efficacy of finding parameter estimates in practice. Confidence intervals are computed using asymptotic error theory as well as bootstrapping. We then develop a model comparison test to be used in determining if a particular data set came from a low input amplitude or a high input amplitude, which we anticipate will aid in determining when stenosis is present. These two thrusts together will serve as the methodological basis for our future analysis using experimental data currently being collected.

Mathematics Subject Classification: 62F12; 62F40; 65M32; 74D05.

Key words: viscoelastic model; sensitivity analysis; inverse problem; asymptotic theory; bootstrapping; model selection.

1 Introduction

Current methods to detect and locate arterial stenoses (blocked arteries) include somewhat invasive angiography and expensive CT scans. Neither procedure is particularly easy to administer, while the CT scan can localize hard plaques but not soft plaques [Luke]. Accordingly, there is interest in examining other methods to determine the existence and location of stenosed vessels. Previous work [Luke, Sam, BaSam, BaPint, BaMedPint, BaLu, BBETW] focused on developing a sensor device to be used with a physical model of a chest cavity, and then developing a mathematical model to describe the medium in which a stenosis-generated acoustic signal is propagated to the chest surface. After an interregnum of roughly five years between that earlier work and our current efforts, we have returned to the early ideas and have reformulated the problem to some extent. This is motivated by companion experiments being conducted with novel acoustic phantoms built at Queen Mary, University of London (QMUL) and Barts & London NHS Trust (BLT) in England. Our viscoelastic model will be quite general, incorporating Kelvin-Voigt damping and internal variables in a hysteresis formulation, so as to provide maximum flexibility in these early stages of model development and analysis.

In this work, we continue to use mathematical modeling techniques in order to non-invasively determine the existence and location of any arterial stenoses, ideally through sensors placed only on the surface of the chest. To this end, we have developed novel experiments to produce one-dimensional pressure wave data that can be fit using the viscoelastic model developed here. While this work will be informed partly by test experimental devices which have been built, it is important to note that all the data employed in this initial mathematical formulation will be simulated data. Broadly, then, the work here is intended to serve as a proof-of-concept of our mathematical and statistical inverse problem methodology, using simulated data very similar to the experimental data we will subsequently use with these methods along with incorporation of standard viscoelastic models (see, e.g., [BaHuKe]).

Our goals for these initial efforts are twofold. We first focus on developing methodologies for determining material parameters and analyzing data using a viscoelastic model, as well as also quantifying the uncertainty in the estimation procedure through both bootstrapping and asymptotic error theories. We then conduct model comparison testing to examine the viability of determining if data originated from a low-amplitude traction (e.g., resulting from normal blood flow) or high-amplitude traction (e.g., resulting from abnormal blood flow caused by a stenosis). This notion builds off of previous work (see, e.g., [BBETW, Niss, NiYo, OwHu, OHAK, SWKM, Ver, VeVa]) which discussed the compression and shear waves which result from a stenosed vessel and some methods for measuring these waves, in particular the shear waves which experience slower transmission than the pressure waves. An ultimate goal of our wider research project will be a methodology to decide if a vessel is stenosed or not, and if so, possibly the extent of the stenosis. However, at this point we have not carried out experiments to determine these differences in either test devices or live subjects. Thus, in these early efforts we make the (very) tentative assumption of representing the difference between normal vessels and stenosed vessels as a comparison between low and high shear input amplitudes, leaving the specifics of the actual system inputs to future work. Although this might need to be revisited in the future, it is a reasonable first approximation. Overall, then, these two thrusts of material parameter estimation and model comparison tests represent a proof-of-concept for our future data-driven inverse problem work.

1.1 Problem formulation

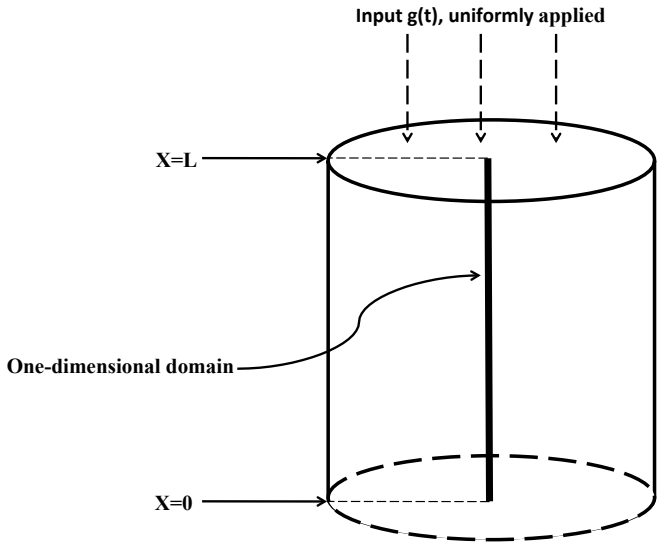

Figure 1: Setup of agar phantom, with sample one-dimensional domain denoted

In this work, we will examine a simplified one-dimensional model for an agar phantom, as pictured in Figure 1. This configuration is an approximation (under certain assumptions) to the novel experimental devices we are using at QMUL to gather experimental data. Development of general viscoelastic equations can be found throughout the literature; in particular, one may refer to [BaHuKe] as a source of the model components discussed in this current work. As in the example in [BaHuKe], we make simplifying assumptions that will result in a one-dimensional wave equation. If we assume a uniform force applied along the top of the phantom and radial symmetry within the phantom (in part to closely match the symmetrically constructed phantoms used at QMUL), then we can simplify the cylindrical physical domain to a one-dimensional domain and to finding the function $u(x,t)$ which represents the material response to, in this case, an applied stenosis-generated like force.

We will use a general acoustic pressure viscoelastic wave equation on a one-dimensional domain $\Omega = [0, L]$. For the purposes of our initial investigation here, all parameters will be considered constant (i.e., a homogeneous medium). This is not necessary but significantly simplifies our initial computations in methodology development. Choosing a material initially at rest with a reflecting boundary at $x = 0$, an applied force $g(t)$ at the $x = L$ boundary, and no additional forcing terms

we obtain the system for the displacement u

$$\begin{aligned} &\rho u_{tt} - \sigma_x = 0 \\ &u(0,t) = 0, \quad \sigma(L,t) = -g(t), \\ &u(x,0) = 0, \quad u_t(x,0) = 0. \end{aligned} \tag{1}$$

A constitutive relationship for stress $\sigma(x,t)$ in terms of strain must be developed. To this end, we assume that the stress is described by a combined model with Kelvin-Voigt damping and Maxwell-Zener internal variables (also known as the generalized standard linear solid model). We begin with the equation

$$\sigma = E_1 u_{xt} + E_0 \left(P(0) u_x - \int_0^t P_s(t-s) u_x(s) ds \right) \tag{2}$$

The Kelvin-Voigt damping is represented by the term involving E_1. The form of the stress relaxation function $P(t)$ is assumed to be a Prony series

$$P(t) = p_0 + \sum_{j=1}^{N_p} p_j e^{-t/\tau_j},$$

where we assume all the p_j are nonnegative numbers and the τ_j values are positive, and with N_p being a positive integer. This series makes the assumption that relaxation in a material can be represented by a discrete number of relaxation times τ_j. Without loss of generality, we will also enforce $P(0) = 1$. A result of this constraint is that $\sum_{j=0}^{N_p} p_j = 1$. If we replace $P_s(t-s)$ in (2) with the s-derivative of the Prony series at $t-s$, we obtain

$$\sigma = E_1 u_{xt} + E_0 \left(u_x - \sum_{j=1}^{N_p} \int_0^t \frac{p_j}{\tau_j} e^{-(t-s)/\tau_j} u_x(s) ds \right).$$

We can reformulate the integrals related to each internal variable as differential equations which we can solve simultaneously with the main system PDE. To this end, we define

$$\epsilon^j = \int_0^t \frac{p_j}{\tau_j} e^{-(t-s)/\tau_j} u_x(s) ds.$$

Then the time derivative of ϵ^j is given by

$$\epsilon^j_t = \frac{p_j}{\tau_j} u_x(t) - \frac{1}{\tau_j} \int_0^t \frac{p_j}{\tau_j} e^{-(t-s)/\tau_j} u_x(s) ds.$$

Relating ϵ^j and ϵ^j_t allows us to model the internal variables dynamically as

$$\begin{aligned} &\tau_j \epsilon^j_t + \epsilon^j = p_j u_x, \\ &\epsilon^j(0) = 0. \end{aligned} \tag{3}$$

for $j = 1, 2, \ldots, N_p$. Then we can write the overall stress-strain relationship as

$$\sigma \quad = \quad E_1 u_{xt} + E_0 \left(u_x - \sum_{j=1}^{N_p} \epsilon^j \right). \tag{4}$$

Note that even though p_0 is an element in the Prony series for $P(t)$, once the series is substituted in the model the constant p_0 no longer appears. However, p_0 is still present in the sum-to-one constraint on *all* p_i values, but we can easily work with the alternate constraint that the remaining p_j terms follow $\sum_{j=1}^{N_p} p_j \leq 1$. Then, E_0 and E_1 along with the relaxation times represent the parameters which affect the response of a particular material to stresses. We incorporate Kelvin-Voigt damping via the $E_1 u_{xt}$ term in (4). Development of these models is described in [BaHuKe], as well as in standard viscoelastic theory [FiLaOn, Lak, GoGr, Fer, CoNo, FaMo]. The damping and internal variables provide us the future flexibility to match the model to data from the experimental device, and also present an interesting question of identifiability of the damping and internal variable parameters which we will later discuss in depth. Note also that the authors in [BBETW, Luke, Sam] give computational results (using essentially an equivalent model from a slightly different conceptual formulation involving a distribution of relaxation mechanisms-see [BaPint]) showing that discrete relaxation times can model well the viscoelastic material responses of the type we consider in this work (namely, attempting to mimic the response of biological soft tissue). In fact, in previous work no more than two discrete relaxation times were used, which has informed our decision to allow a maximum of two relaxation times.

Since the ultimate goal of the wider research project will be examining the traction into the chest cavity that results from a healthy artery experiencing a heartbeat as compared with an artery containing a stenosis, our nonzero boundary input $g(t)$ will here be represented by an approximation to a pulse traction. In order to ensure a smooth, compactly supported input, we implement the input function as a Van Bladel function. This smoothness is useful in order to get the maximum benefit from using high order numerics.

The function used is

$$g(t) = \begin{cases} A \cdot \exp\left(\dfrac{|ab|}{t(t+a-b)} \right) & \text{if } t \in (0, b-a), \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Altogether, this description (1), (3) and (4) along with (5) encompasses the one-dimensional model for the displacement $u(x,t)$ that will be studied in both major sections of this work.

Throughout this work, we solve (1), (3) and (4) for $u(x,t)$ with given input function $g(t)$ and parameter values using a spectral continuous finite element method in space (Gauss-Lobatto nodes) and a discontinuous Galerkin method in time. The numerical scheme is especially tailored to allow for high order space-time discretization in order to control dispersion errors and will be documented fully in [BICOM-2012].

1.2 Parameter values

We pick values for the system parameters which simulate low-amplitude (on the order of $0.1mm$) oscillatory motion, motivated by the experimental data to which we intend to apply this methodology. For data generation, we will use two internal variables ($N_p = 2$). The weights p_i for our two relaxation time model will be as shown below. The baseline material parameter values chosen for this work are as follows:

$$\begin{gathered} E_0 = 2.2 \times 10^5 \text{ Pa}, \quad E_1 = 40 \text{ Pa} \cdot \text{s}, \quad \rho = 1010 \text{ kg/m}^3, \quad L = 0.053 \text{ m} \\ \tau_1 = 0.05 \text{ s}, \quad \tau_2 = 10 \text{ s}, \quad p_1 = 0.3, \quad p_2 = 0.55. \end{gathered} \tag{6}$$

Note that the density $\rho = 1010$ kg/m^3 is the true density of the agar gel that is used in the medium for our experiments at QMUL, and $L = 0.053$ m is the true height of the phantom. These are parameter values which are directly taken to approximate the experimental device. The values for E_0 and E_1 and for the relaxation times are physically reasonable parameters based on a perusal of the viscoelastic materials research literature and are also informed by our early experiments with the agar gels.

In the Van Bladel function, the values of a and b have an effect on pulse width as well as the amplitude. We take $a = 6 \times 10^{-3}$ and $b = 20 \times 10^{-3}$ for an effective pulse application time of 14 ms. We then use A to scale the amplitude, in this case taking $A = 6 \times 10^3$. The Van Bladel function with these parameter values, which we will use as our "true" input function, is depicted in Figure 2.

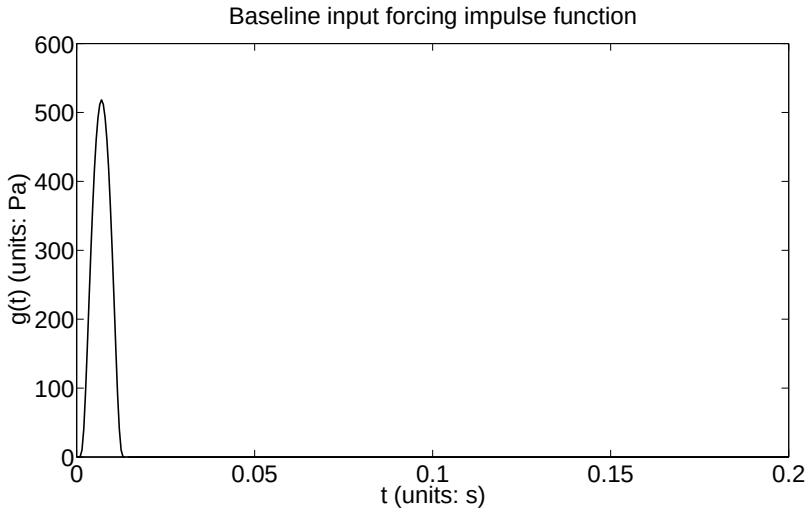

Figure 2: Van Bladel function (5) with $A = 6 \times 10^3$, $a = 6 \times 10^{-3}$, and $b = 20 \times 10^{-3}$.

In reality, one will obtain a set of experimental data and then one needs to determine how many (if any) relaxation times are required to represent well the data. Thus, we will want to compare the performance of three models. In each model, we will always estimate E_0 and E_1 (assuming given values for ρ and L in (6)), but we will vary the number of relaxation times incorporated into the model. The three models are as follows:

1. For a model with no relaxation times, we do not include any τ_i or corresponding p_i in the model. Thus, we estimate only E_0 and E_1.

2. In the case with one relaxation time, we incorporate a single internal variable (i.e., $N_p = 1$), corresponding with relaxation time τ_1, and use the material weights

$$p_1 = 0.3. \tag{7}$$

3. For the case of two relaxation times, we will use the material weights in (6), given by

$$p_1 = 0.3, \quad p_2 = 0.55. \tag{8}$$

Considering this set of models will allow us to follow what one would consider in practice, examining the results of adding/subtracting model features. It is worth noting here that for this particular set of models, the one-relaxation-time model (the second case) is not a special case of the two-relaxation-time model (the third case) as the material weighting p_2 in the two-relaxation-time model is fixed, and that the no-time-relaxation-time model (the first case) is not a special case of the one-time-relaxation model as the material weighting p_1 in the one-relaxation-time model is fixed. However, if we allow the corresponding material weights to be free (i.e., to be estimated along with relaxation times), then the no-relaxation-time model is indeed a special case of the one-relaxation-time model, and the one-relaxation-time model is a special case of the two-relaxation-time model. We will use the sensitivity equations and parameter estimation results as well as model selection criteria to suggest the number of relaxation times to use in practice.

1.3 "True" model

Using the baseline parameters, we present a graph of the noiseless model response, what we will henceforth call the "true" system motion in Figure 3. This is the simulated device motion from which we will generate our data, and also demonstrates motion under the true parameter values which we will use with our inverse problem methodology to attempt to recover from noisy data. Note that the motion is on the 1/10 millimeter scale, which was again motivated by the likely scale of results from the experimental device.

2 Estimation of material parameters

In this section, we examine an inverse problem methodology for estimating material parameters (and thus gain a sense for our ability to characterize an individual's material properties) under various measurement noise conditions. In addition to determining an estimate for material parameters, we also need to determine our confidence in the estimation procedure. To this end, we will compare two techniques for determining confidence intervals, specifically the asymptotic theory discussed in [BaTr, BaHoRo] versus using bootstrapping as discussed in [BaHoRo].

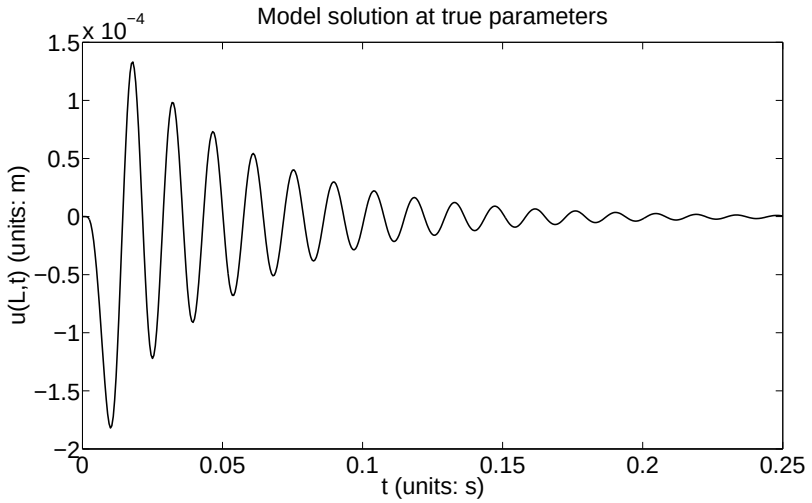

Figure 3: Solution of model (1), (3) and (4) using the "true" parameters (6) and forcing function with parameters $a = 6 \times 10^{-3}$, $b = 20 \times 10^{-3}$, and $A = 6 \times 10^{3}$.

2.1 Study of effects of changing material parameters

Before discussing simulated data and actually solving the inverse problem, we wish to complete some analysis on the model around the "true" material parameter values (6). It is clear that changing the amplitude factor A for the Van Bladel input will change the resulting amplitude of the system. Hence, we consider here changes in the material parameters E_0, E_1, and τ_j values.

We first consider changes in the stiffness E_0 and damping factor E_1. As an example of typical effects of changing parameters, we show the effects of reducing stiffness to $E_0 = 200,000$ in the left pane of Figure 4 and in the right pane the effects of increasing the damping to $E_1 = 60$. Changes

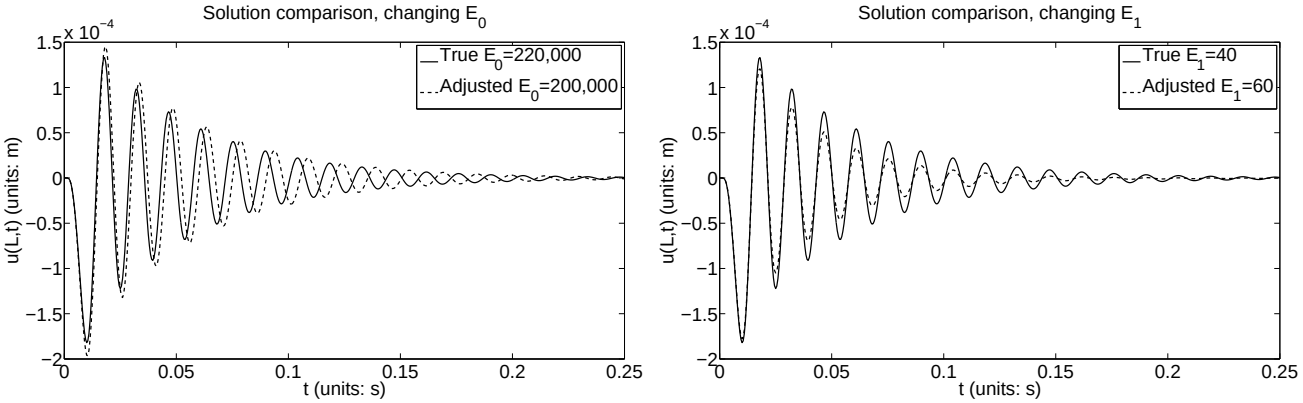

Figure 4: Solution of model (1), (3) and (4) using the "true" parameters (6) and forcing function with parameters $a = 6 \times 10^{-3}$, $b = 20 \times 10^{-3}$, and $A = 6 \times 10^{3}$ (depicted by the solid line), alongside solutions using $E_0 = 2 \times 10^5$ (left pane) and $E_1 = 60$ (right pane) which are represented with dashed lines in their respective graphs.

in E_0 are shown to have a significant effect on the oscillation frequency, as well as a minor effect

on peak heights. This is known – a more stiff material will propagate waves more quickly and will dissipate energy less slowly. Increases in damping, E_1, lead to the expected effects that the energy dissipates faster in the material, so the oscillation peak heights become smaller and the material experiences fewer small oscillations at later simulation times. As one might expect, these two parameters seem to govern the major properties of how the material responds to an impulse response traction.

Relaxation times can allow the model flexibility in matching the periodic local "peaks" and "troughs" in the oscillating solution. For example, if we change from the baseline $\tau_1 = 0.05$ to $\tau_1 = 0.5$, the system experiences the changes shown in Figure 5. This response to changing relaxation

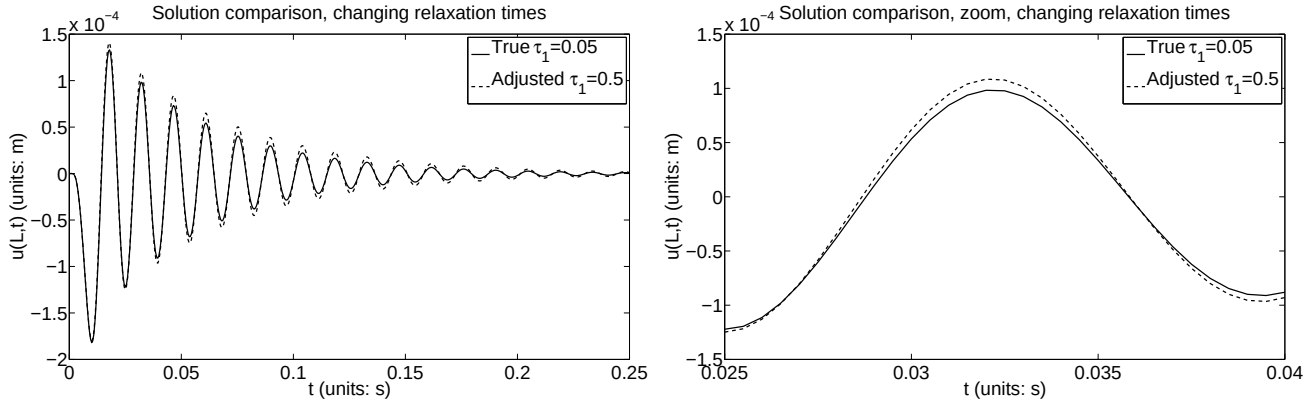

Figure 5: Solution of model (1), (3) and (4) using the "true" parameters (6) and forcing function with parameters $a = 6 \times 10^{-3}$, $b = 20 \times 10^{-3}$, and $A = 6 \times 10^3$ (depicted by the solid line), alongside the dashed line using $\tau_1 = 0.5$ with the remaining parameters the same. The right pane demonstrates the solution zoomed in for $t \in [0.025, 0.04]$.

times represents a typical example of changing either τ_1 or τ_2. However, note the scale of the changes: the maximum difference between the solutions shown in Figure 5 is 1.0996×10^{-5}. As we will see later when adding noise, the noise itself is on the scale of 10^{-5}. This foreshadows the difficulties in properly estimating relaxation times that we will see going forward. This will be evident also both in a discussion on using different optimization routines and in a discussion of model sensitivity with respect to parameters.

2.2 Sensitivity of model output with respect to material parameters

In order to further quantify the model response to changes in parameters around the baseline values of (6), we will examine the sensitivity of the model output with respect to material parameters. Note that since the values of parameters are on such a varying scale, we will actually work with the log-scaled versions of the material parameters we are attempting to estimate. In other words, if $\bar{\theta} = (E_0, E_1, \tau_1, \tau_2)^T$ is the vector of parameters to be estimated, we define $\theta = \log_{10}(\bar{\theta})$.

Sensitivity analysis has been widely used in inverse problem investigations (e.g., see [BaTr] and the references therein for details) to identify the model parameters and/or initial conditions to which the model outputs (observations) are most sensitive and for which one can readily construct confidence intervals when they are estimated (i.e., which are the most reliably estimated values).

To compute the sensitivity of the model output to each parameter, one needs to find sensitivity equations which describe the time evolution of the partial derivatives of the model state with respect to each parameter. Sensitivity equations in terms of the non-log-scaled parameters $\bar{\theta}$ are derived in Appendix A. Since we are only using observations at the $x = L$ position, we will actually consider only the sensitivities with respect to each parameter at this position.

We can use the sensitivity of model output to the non-log-scaled parameters to find the sensitivity of model output with respect to the log-scaled parameters, which will be of interest here. Using the chain rule, we find that

$$\frac{\partial u(L,t;10^{\theta})}{\partial \theta_i} = \bar{\theta}_i \ln(10) \frac{\partial u(L,t;\bar{\theta})}{\partial \bar{\theta}_i},$$

where θ_i and $\bar{\theta}_i$ are the ith element of θ and $\bar{\theta}$, respectively.

The sensitivities of model output with respect to parameters $(\log_{10}(E_0), \log_{10}(E_1), \log_{10}(\tau_1), \log_{10}(\tau_2))$ are shown in Figure 6. From this figure we see that model output is most sensitive to

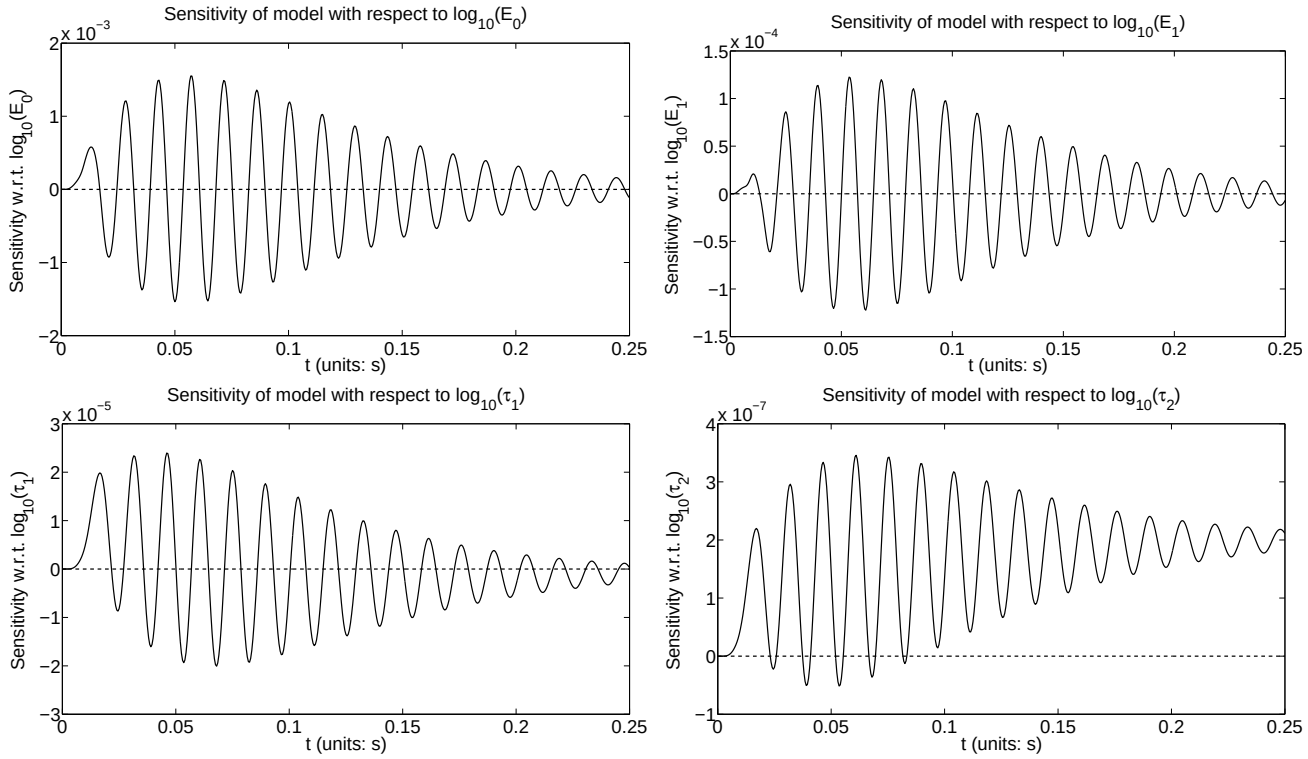

Figure 6: (upper left pane) Sensitivity of model output with respect to $\log_{10}(E_0)$; (upper right pane) Sensitivity of model output with respect to $\log_{10}(E_1)$; (bottom left pane) Sensitivity of model output with respect to $\log_{10}(\tau_1)$; and (bottom right pane) Sensitivity of model output with respect to $\log_{10}(\tau_2)$. All sensitivities are around the baseline parameters (6) and forcing function with parameters $a = 6 \times 10^{-3}$, $b = 20 \times 10^{-3}$, and $A = 6 \times 10^{3}$.

$\log_{10}(E_0)$, sensitive to $\log_{10}(E_1)$, less sensitive to $\log_{10}(\tau_1)$, and least sensitive to $\log_{10}(\tau_2)$. The most interesting feature related to our study is the fact that the scale of sensitivity of model output to the first relaxation time is on the order of 10^{-5} whereas the sensitivity of model output to the second

relaxation time is roughly two orders of magnitude smaller on the order of 10^{-7}. We will later see that, while we have difficulty estimating both relaxation times due to the small changes they induce in the model solution (as previously discussed), we especially have difficulty obtaining a reasonable estimate for τ_2 because the model is so much less sensitive to the second relaxation time than to the first. In addition, we observe from Figure 6 that at later times the model output is not particularly sensitive to all the material parameters except the second relaxation time. This further indicates that we may have trouble in estimating the second relaxation time with high additive noisy data (which will be discussed later).

Figure 7 demonstrates the sensitivities of model output with respect to material weights $\log_{10}(p_1)$ and $\log_{10}(p_2)$. From this figure we see that the model is less sensitive to the second weighting than to

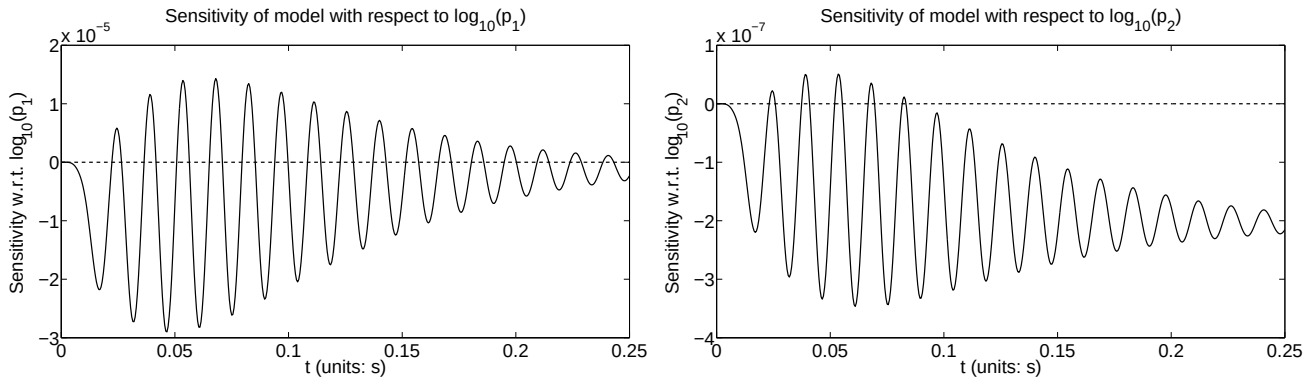

Figure 7: (left pane) Sensitivity of model output with respect to $\log_{10}(p_1)$; and (right pane) Sensitivity of model output with respect to $\log_{10}(p_2)$. Again, both sensitivities around the baseline parameters (6) and forcing function with parameters $a = 6 \times 10^{-3}$, $b = 20 \times 10^{-3}$, and $A = 6 \times 10^3$.

the first one. In future work when we turn to consider estimating the weights (instead of specifying them as we do in this work), we can predict that accurately estimating the second material weighting will be much more difficult than estimating the first one.

Armed with our knowledge of sensitivities of model output with respect to the material parameters around the true parameter values (6), and our knowledge of effects on the model solution of changing the parameters, we can describe the generation of our simulated data and discuss solution of the inverse problem itself.

2.3 Statistical model and inverse problem

We will work with simulated data for various noise levels generated at position $x = L$, namely data u_j corresponding with model solution $u(L, t_j)$ at measurement time points $t_j = 0.001j$, $j = 0, 1, \ldots, 250$. Thus, there are a total of $n = 251$ data points. For the current proof of concept discussion, we will assume the measurement errors $\mathcal{E}_j$ are independent, identically, normally distributed with mean zero ($E(\mathcal{E}_j) = 0$) and constant variance $\text{var}(\mathcal{E}_j) = \sigma_0^2$. We thus are assuming absolute additive error; this is necessary in order to use the hypothesis testing methodology later, and is reasonable as an initial error model for our proof of concept investigations. The error

assumptions correspond with the error process

$$U_j = u(L, t_j; 10^\theta) + \mathcal{E}_j, \quad j = 0, 1, \ldots, n-1, \tag{9}$$

where $u(L, t_j; 10^\theta)$ is the solution to (1), (3) and (4) along with (5) at $x = L$ with a given set of material parameters θ. For realizations ϵ_j of $\mathcal{E}_j$, corresponding realizations of U_j are given by

$$u_j = u(L, t_j; 10^\theta) + \epsilon_j, \quad j = 0, 1, \ldots, n-1. \tag{10}$$

Under the error assumption framework (9), we now define the estimator $\hat{\Theta}$ to be

$$\hat{\Theta} = \arg\min_{\theta \in Q} \sum_{j=0}^{n-1} \left[U_j - u(L, t_j; 10^\theta)\right]^2, \tag{11}$$

where $Q \subset \mathbb{R}^\kappa$ is some viable admissible parameter set, assumed compact in $\mathbb{R}^\kappa$ with κ being the number of parameters requiring estimation. Note that under different error assumptions, one would need to modify the cost function in (11) (a topic discussed in [BaTr]) for an appropriate asymptotic parameter distribution theory to be valid. Since $\hat{\Theta}$ is a random variable (inherited from the fact that the errors $\mathcal{E}_j$ are random variables), we can define its corresponding realizations by (using data realizations (10), either simulated or from an experiment)

$$\hat{\theta} = \arg\min_{\theta \in Q} \sum_{j=0}^{n-1} \left[u_j - u(L, t_j; 10^\theta)\right]^2. \tag{12}$$

Estimating material parameters $\hat{\theta}$ from given sets of data with different noise levels, as well as quantifying uncertainty in our estimate, will be the key focus of our work in this section. For these purposes, we will assume that the material density and the weights p_i for each relaxation time are known and given in either (7) or (8) (depending on the number of relaxation times to be estimated). Though in reality one would certainly need to estimate the weights p_i, we take the liberty here of assuming them to be known so we can focus on the general methodology and in particular the identifiability of the relaxation times. We also use the values for E_0, E_1, τ_1, and τ_2 in (6) in their log scaled form as the baseline values θ_0 used in simulating data. That is,

$$\theta_0 = (5.3424, 1.6021, -1.3010, 1)^T = (\log_{10}(2.2 \times 10^5), \log_{10} 40, \log_{10} .05, \log_{10} 10)^T$$

and these are the values we are seeking to estimate at different noise levels. As previously discussed, we will find parameter estimates for models with zero, one, and two relaxation times in the model itself (and thus the number of parameters estimated changes). In all cases, the parameters belong to a viable compact set Q with the upper and lower bounds on parameters being taken (in educated guesses) as $\theta_{lb} = (-15, -15, -15, -15)^T$, $\theta_{ub} = (7.3010, 2.3010, 2, 2)^T$ for two relaxation times estimation, $\theta_{lb} = (-15, -15, -15)^T$, $\theta_{ub} = (7.3010, 2.3010, 2)^T$ for the one relaxation time estimation, and $\theta_{lb} = (-15, -15)^T$, $\theta_{ub} = (7.3010, 2.3010)^T$ for the no relaxation time estimation.

2.3.1 Data generation

As noted previously, we will simulate data using two relaxation times (and a question of interest later will be how many of those relaxation times we can recover). With the values of parameters given in (6), noiseless data has maximum amplitude on the order of 10^{-4} (shown in Figure 3). This level informs the magnitude we choose for the additive noise. We represent a "low" noise level with $\sigma_0^2 = 5 \times 10^{-6}$, a "medium" noise level by $\sigma_0^2 = 10 \times 10^{-6}$, and a "high" noise level by taking $\sigma_0^2 = 20 \times 10^{-6}$. In Figure 8, we show plots corresponding to the three levels of noisy simulated

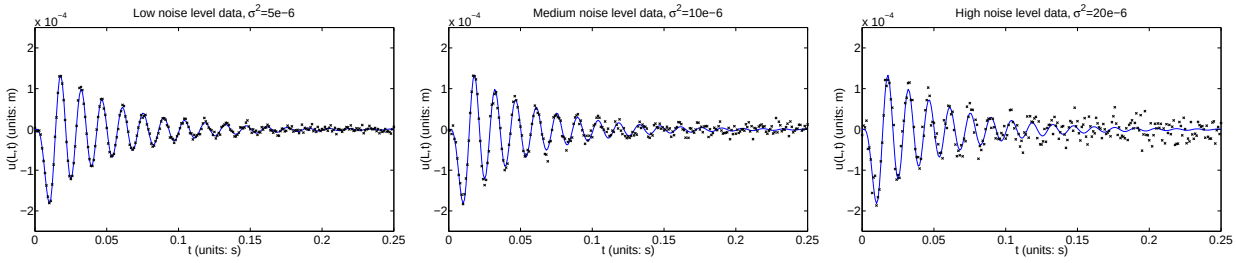

Figure 8: Simulated noisy data around the true parameter values. (left pane) Low noise level, $\sigma_0^2 = 5 \times 10^{-6}$. (middle pane) Medium noise, $\sigma_0^2 = 10 \times 10^{-6}$. (right pane) High noise level, $\sigma_0^2 = 20 \times 10^{-6}$.

data against the system dynamics corresponding to the true parameters. Noise is assumed absolute for our initial investigations (though we may ultimately need to explore relative noise once an error model is developed for our experimental data), and is added according to the error model (10). Low noise results in data mostly along the trajectory of the true model. Medium noise begins to obfuscate the later-time oscillations which have lost much of their earlier energy. High noise significantly affects the level of peaks and troughs from $t = 0.05$ forward. We thus obtain a series of increasingly difficult problems in obtaining material parameter estimates, though entirely expected since higher noise tends to significantly affect data features and presents a more difficult parameter estimation problem.

2.3.2 Inverse problem, different optimization routines

In this section, we discuss different options for the optimization routine used to solve (12), and begin to gain a sense of the robustness of parameter estimation with respect to the optimization routine. Note that we expect to have some difficulty in relaxation time estimation, based on our earlier discussion on the model response to changes in relaxation times as well as model sensitivities. We do expect to obtain more accurate estimates for E_1, and very good estimates for E_0. To begin this discussion, we will examine parameter estimates for a model which incorporates two relaxation times.

The optimization routines we compare are all built-in Matlab routines. We use `fmincon` with active-set optimization, which treats the optimization as constrained nonlinear programming with our cost function (12). We also examine the use of `lsqnonlin`, which is designed for nonlinear least squares data-fitting problems; our cost function is exactly the form of a nonlinear least squares function. We test both the Levenburg-Marquardt (LM) option and the trust-region-reflective (TRR)

option. Note that the Levenburg-Marquardt algorithm does *not* allow bound constraints; we tried the routine out of curiosity, to see if it would produce unrealistic estimates of any parameters (it does at high noise levels).

Results from optimizing for θ are shown in Tables 1-3. All optimization runs used the initial guess

$$\theta_{init} = (\log_{10}(1.8 \times 10^5), \log_{10}(60), \log_{10}(0.5), \log_{10}(20))^T = (5.2553, 1.7782, -0.3010, 1.3010)^T.$$

All these tables include the parameter estimates for $\hat{\theta}$, computation time for that particular optimization run, and the residual sum of squares (RSS) defined as

$$\text{RSS} = \sum_{j=0}^{n-1} \left[u_j - u(L, t_j; 10^{\hat{\theta}}) \right]^2 .$$

Overall, the routines do a good job (Tables 1-3) of estimating E_0, as we expected. The `lsqnonlin`

Table 1: Estimation of material parameters in the case of low noise $\sigma_0^2 = 5 \times 10^{-6}$: Comparison between optimization routines (TRR=trust-region-reflective, LM=Levenburg-Marquardt).

	estimated parameter values $\hat{\theta}$	Comp. Time (in sec.)	RSS
`fmincon`:	$(5.3422, 1.6581, -0.3000, 1.3012)^T$	194.16	6.7618×10^{-9}
`lsqnonlin`, TRR:	$(5.3425, 1.6046, -1.2297, 1.0046)^T$	347.09	6.2458×10^{-9}
`lsqnonlin`, LM:	$(5.3425, 1.6044, -1.2309, 1.0316)^T$	613.47	6.2458×10^{-9}
true values θ_0:	$(5.3424, 1.6021, -1.3010, 1.0000)^T$		

Table 2: Estimation of material parameters with medium noise $\sigma_0^2 = 10 \times 10^{-6}$: Comparison between optimization routines (TRR=trust-region-reflective, LM=Levenburg-Marquardt).

	estimated parameter values $\hat{\theta}$	Comp. Time (in sec.)	RSS
`fmincon`:	$(5.3430, 1.6583, -0.2998, 1.3012)^T$	203.75	2.4435×10^{-8}
`lsqnonlin`, TRR:	$(5.3433, 1.5889, -1.3269, 2.0000)^T$	241.51	2.3647×10^{-8}
`lsqnonlin`, LM:	$(5.3433, 1.5893, -1.3252, 5.9303)^T$	608.27	2.3646×10^{-8}
true values θ_0:	$(5.3424, 1.6021, -1.3010, 1.0000)^T$		

Table 3: Estimation of material parameters in the case of high noise $\sigma_0^2 = 20 \times 10^{-6}$: Comparison between optimization routines (TRR=trust-region-reflective, LM=Levenburg-Marquardt).

	estimated parameter values $\hat{\theta}$	Comp.Time(s.)	RSS
`fmincon`:	$(5.3433, 1.6380, -0.2995, 1.3012)^T$	238.58	1.03291×10^{-7}
`lsqnonlin`, TRR:	$(5.3433, 1.6361, -1.990, 0.2496)^T$	606.36	1.03257×10^{-7}
`lsqnonlin`, LM:	$(5.3433, 1.6351, -0.02324, 3.6112 \times 10^{-4})^T$	1110.83	1.03248×10^{-7}
true values θ_0:	$(5.3424, 1.6021, -1.3010, 1.0000)^T$		

routines tend to better estimate E_1. As for relaxation times, we begin to see a major flaw in the `fmincon` routine. It does not seem particularly sensitive to the relaxation times, and the resulting estimates of the relaxation times stay near the initial guess. The `fmincon` routine produced similar non-responsive results for different initial guesses. The `lsqnonlin` routines estimate the relaxation times well in the presence of low noise (Table 1). At medium noise (Table 2), the routines estimate τ_1 well but not τ_2. At high noise (Table 3), relaxation time estimation is poor. This will be quantified further in the following sections on error analysis.

Even though there might be some spurious computation times on desktop machines (due to other background programs), we still include them here to demonstrate typical optimization routine performance. Consistently, `fmincon` was the fastest routine. This is in part due to the fact that this routine alone of the three supports parallel computation, so on our multi-core desktop machines we were able to see a speed-up. However, the computation times for the trust-region-reflective `lsqnonlin` algorithm are reasonable. Using Levenburg-Marquardt consistently is the slowest method, and the results are not better than those using trust-region-reflective `lsqnonlin` algorithm.

As a result, we recommend using the trust-region-reflective `lsqnonlin` algorithm when trying to estimate relaxation times. If the model does not contain relaxation times (i.e., only estimating E_0 and E_1), the speedup afforded by using `fmincon` may make that algorithm the one of choice. Figure 9 illustrates model fits to the data at different noise levels, where the model solution is calculated with the values of model parameters obtained through `lsqnonlin` TRR routine. We see in all cases that the model solution provides reasonable fits to the data.

2.3.3 Asymptotic error analysis

Most asymptotic error theory [BaTr, BaHoRo] is described in the context of an ODE model example $\dot{z}(t) = g(z(t;\theta);\theta)$. However, we can use the PDE sensitivities of the model state with respect to each parameter in θ, namely $\dfrac{\partial u(x,t;10^{\theta})}{\partial \theta_i}$, in a similar manner to the ODE sensitivities in the asymptotic theory. The steps of the asymptotic theory error analysis are as follows (the theory for the following steps is described in [BaTr, BaHoRo]).

1. Determine $\hat{\theta}$ by computing (12).

2. Compute the sensitivity equations to obtain $\dfrac{\partial u(x,t;10^{\hat{\theta}})}{\partial \theta_i}$ (as discussed in Section 2.2) for $i = 1, \ldots, \kappa$ where κ is the number of parameters being estimated. Since data is only taken at position $x = L$, only the values at $x = L$ will be used to form the sensitivity matrix $\chi(\hat{\theta})$. This matrix then has entries

$$\chi_{j,i}(\hat{\theta}) = \frac{\partial u(L,t_j;10^{\hat{\theta}})}{\partial \theta_i}, \quad j = 0, 1, \ldots, n-1, \text{ and } i = 1, \ldots, \kappa.$$

   Note that $\chi(\hat{\theta})$ is then an $n \times \kappa$ matrix. We can also obtain an estimate for the constant variance σ_0^2 as

$$\hat{\sigma}^2 = \frac{1}{n-\kappa} \sum_{j=0}^{n-1} \left(u_j - u(L,t_j;10^{\hat{\theta}}) \right)^2.$$

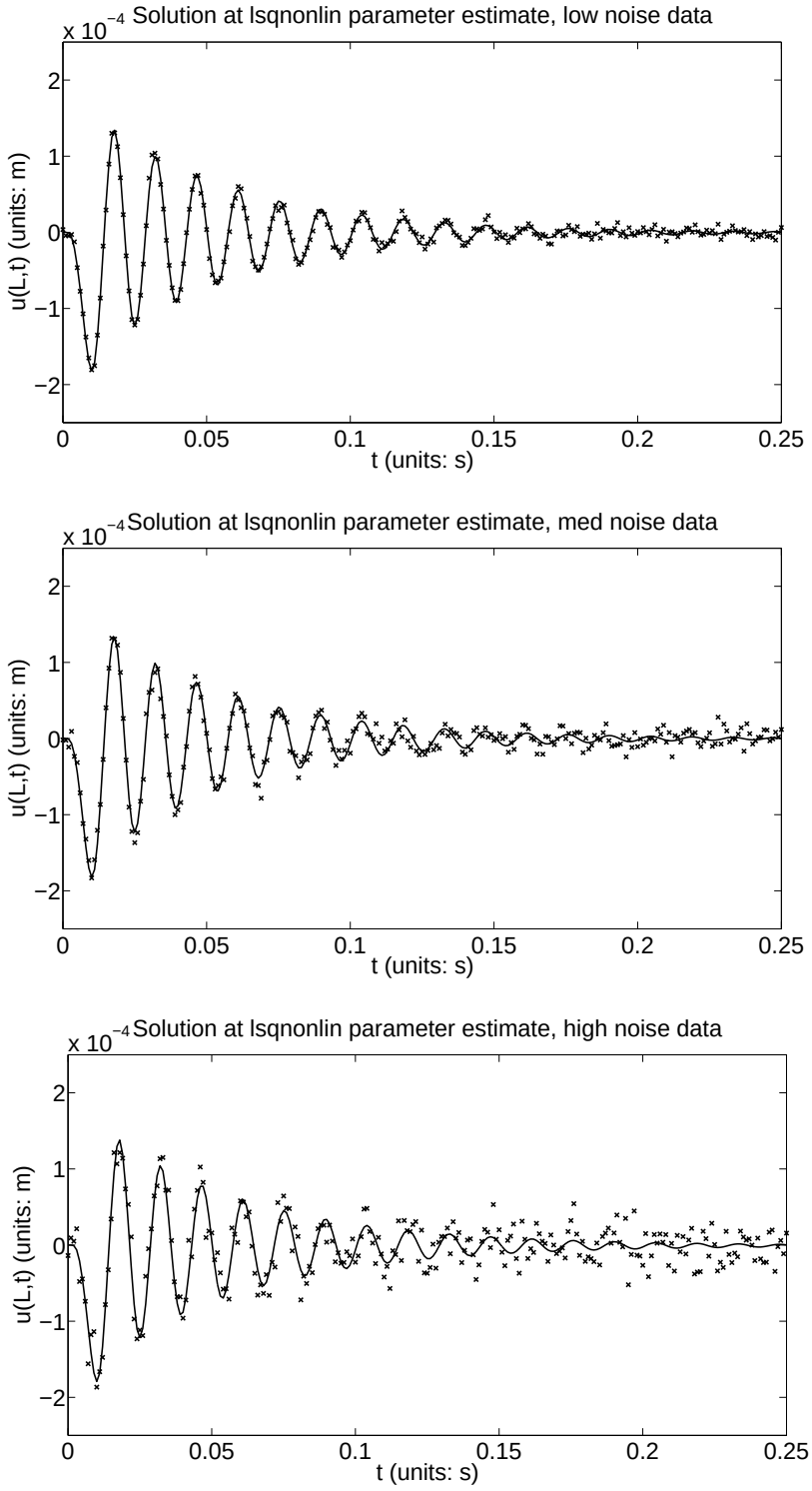

Figure 9: Data and two-relaxation-time model solutions at parameter estimates obtained using `lsqnonlin`, trust-region-reflective method, at different noise levels (see Tables 1-3). (upper pane) Low noise. (middle pane) Medium noise. (bottom pane) High noise.

3. Asymptotic theory yields that the estimator $\hat{\Theta}$ is asymptotically (as sample size $n \to \infty$) normal with mean approximated by $\hat{\theta}$ and the covariance matrix approximated by

$$Cov(\hat{\Theta}) \approx \hat{\Sigma} = \hat{\sigma}^2[\chi^T(\hat{\theta})\chi(\hat{\theta})]^{-1}.$$

4. The standard errors for each element in the parameter estimator $\hat{\Theta}$ can be approximated by

$$SE(\hat{\Theta}_i) = \sqrt{\hat{\Sigma}_{ii}}, \quad i = 1, 2, \ldots, \kappa,$$

where $\hat{\Theta}_i$ is the ith element of $\hat{\Theta}$, and $\hat{\Sigma}_{ii}$ is the (i,i)th entry of the matrix $\hat{\Sigma}$. Hence, the endpoints of the confidence intervals for $\hat{\Theta}_i$ are given by

$$\hat{\theta}_i \pm t_{1-\alpha/2} SE(\hat{\Theta}_i)$$

for $i = 1, 2, \ldots, \kappa$. Here $t_{1-\alpha/2}$ is a distribution value that is determined from a statistical table for Student's t-distribution based on the level of significance α (i.e., $\alpha = .05$ for a 95% confidence interval).

We will present results below in Tables 4-9 on the low, medium, and high noise data sets using zero, one, and two relaxation times, and using the routines `fmincon` and `lsqnonlin` (trust-region-reflective only, as we cannot enforce the bound constraints with the Levenburg-Marquardt algorithm). We see throughout the tables that the problem of estimating the second relaxation time is fraught with difficulty (the standard error is significantly higher than its estimated value), even though we know the simulated data came from a model incorporating two relaxation times. This was foreshadowed in our earlier examination of the sensitivities with respect to the second relaxation time, as well as the results for relaxation times seen when using different optimization routines. In addition, when estimating two relaxation times using `lsqnonlin` on high noise data (shown in Table 9(c)) we see that the estimates for τ_1 and τ_2 are not close to the true parameter values; also, the standard error for τ_1 is much larger than in any other case. Thus, instead of merely having difficulty estimating a second relaxation time, in this estimation we now additionally have a less confidence on the estimate of τ_1.

From Tables 4-6 we see that standard errors for E_0 and E_1 for the model with no relaxation times are comparable with those for the model with one relaxation time when using `fmincon`, but the standard error for τ_1 is fairly large (around 7, as compared with the estimated parameter value $\log_{10}(\hat{\tau_1}) \approx -0.3$). When using `lsqnonlin` (see Tables 7-9), the standard errors for E_0 and E_1 increase significantly at all noise levels when moving from the no relaxation time model to the one relaxation time model, but the standard error for τ_1 is closer to 2 rather than the 7 for `fmincon`. This may not seem significant, but if we recall that these are log-scaled parameter values, then the difference between standard errors of 2 and 7 is significant and an indication that `lsqnonlin` may be the better procedure.

We also found that at all noise levels the difference for the residual sum of squares is small among the no-relaxation-time, one-relaxation-time, and two-relaxation-time models using either `fmincon` or `lsqnonlin` (see the third columns of Tables 10 and 11). In addition, for each level noise data set, when we plot the model solutions corresponding to the no, one, and two relaxation time models, we found that they approximately lie on top of each other, and give good fits to the data. To gain further insight into which model should be chosen, we turned to some model selection criterion analysis.

Table 4: Low noise, `fmincon`: Parameter estimates, asymptotic standard errors (SE) and confidence intervals

(a) No relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3422	0.01373	(5.3223, 5.3622)
$\log_{10}(E_1)$	1.6021	1.6651	0.1052	(1.4579, 1.8723)

(b) One relaxation time

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3422	0.01012	(5.3223, 5.3622)
$\log_{10}(E_1)$	1.6021	1.584	0.1436	(1.3757, 1.9412)
$\log_{10}(\tau_1)$	-1.3010	-0.3002	6.9746	(-1.4037, 1.3437)

(c) Two relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3422	0.1014	(5.3223, 5.3622)
$\log_{10}(E_1)$	1.6021	1.6581	0.1463	(1.3700, 1.9463)
$\log_{10}(\tau_1)$	-1.3010	-0.3000	8.2619	(-16.5727, 15.9828)
$\log_{10}(\tau_2)$	1	1.3012	51.0143	(-99.1775, 101.7798)

Table 5: Medium noise, `fmincon`: Parameter estimates, asymptotic standard errors (SE) and confidence intervals

(a) No relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3264	0.01409	(5.3087, 5.3542)
$\log_{10}(E_1)$	1.6021	1.7641	0.01165	(1.5346, 1.9936)

(b) One relaxation time

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3430	0.01032	(5.3226, 5.3633)
$\log_{10}(E_1)$	1.6021	1.6586	0.1462	(1.3706, 1.9467)
$\log_{10}(\tau_1)$	-1.3010	-0.3001	7.1076	(-1.4299, 1.3699)

(c) Two relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3430	0.01034	(5.3226, 5.3633)
$\log_{10}(E_1)$	1.6021	1.6583	0.1490	(1.3648, 1.9519)
$\log_{10}(\tau_1)$	-1.3010	-0.2998	8.4206	(-16.8851, 16.2855)
$\log_{10}(\tau_2)$	1	1.3012	51.9907	(-101.1004, 103.7028)

Table 6: High noise, `fmincon`: Parameter estimates, asymptotic standard errors (SE) and confidence intervals

(a) No relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	0.01046	(5.3227, 5.3639)
$\log_{10}(E_1)$	1.6021	1.6452	0.1136	(1.4214, 1.8691)

(b) One relaxation time

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	0.01045	(5.3227, 5.3639)
$\log_{10}(E_1)$	1.6021	1.6382	0.1600	(1.3232, 1.9534)
$\log_{10}(\tau_1)$	-1.3010	-0.2995	7.7134	(-15.4916, 14.8927)

(c) Two relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	0.01047	(5.3227, 5.3639)
$\log_{10}(E_1)$	1.6021	1.6380	0.1633	(1.3164, 1.9596)
$\log_{10}(\tau_1)$	-1.3010	-0.2995	9.1380	(-18.2979, 17.6989)
$\log_{10}(\tau_2)$	1	1.3012	56.4222	(-109.8289, 112.4314)

Table 7: Low noise, TRR `lsqnonlin`: Parameter estimates, asymptotic standard errors (SE) and confidence intervals

(a) No relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3422	4.9498×10^{-4}	(5.3413, 5.3432)
$\log_{10}(E_1)$	1.6021	1.6651	0.005434	(1.6544, 1.6758)

(b) One relaxation time

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3425	0.01011	(5.3226, 5.3624)
$\log_{10}(E_1)$	1.6021	1.6050	0.3167	(0.9811, 2.2288)
$\log_{10}(\tau_1)$	-1.3010	-1.2317	2.2200	(-5.6041, 3.1407)

(c) Two relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3425	0.0101	(5.3225, 5.3624)
$\log_{10}(E_1)$	1.6021	1.6046	0.3202	(0.9738, 2.2353)
$\log_{10}(\tau_1)$	-1.3010	-1.2297	2.2369	(-5.635, 3.1761)
$\log_{10}(\tau_2)$	1	1.0046	16.0237	(-30.5560, 32.5651)

Table 8: Medium noise, TRR `lsqnonlin`: Parameter estimates, asymptotic standard errors (SE) and confidence intervals

(a) No relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3429	9.2836×10^{-4}	(5.3411, 5.3448)
$\log_{10}(E_1)$	1.6021	1.6653	0.0102	(1.6452, 1.6854)

(b) One relaxation time

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	0.01042	(5.3228, 5.3638)
$\log_{10}(E_1)$	1.6021	1.6050	0.3167	(0.9811, 2.2288)
$\log_{10}(\tau_1)$	-1.3010	-1.2317	2.2200	(-5.6041, 3.1407)

(c) Two relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	0.01045	(5.3227, 5.3639)
$\log_{10}(E_1)$	1.6021	1.5889	0.3717	(0.8567, 2.3211)
$\log_{10}(\tau_1)$	-1.3010	-1.3269	2.0383	(-5.3415, 2.6878)
$\log_{10}(\tau_2)$	1	2.0000	156.5630	(-306.369, 310.369)

Table 9: High noise, TRR `lsqnonlin`: Parameter estimates, asymptotic standard errors (SE) and confidence intervals

(a) No relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	1.7767×10^{-3}	(5.3398, 5.3468)
$\log_{10}(E_1)$	1.6021	1.6452	0.0204	(1.6050, 1.6855)

(b) One relaxation time

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	0.01046	(5.3227, 5.3639)
$\log_{10}(E_1)$	1.6021	1.6397	0.1526	(1.3391, 1.9403)
$\log_{10}(\tau_1)$	-1.3010	-1.3253	2.0328	(-5.3291, 2.6785)

(c) Two relaxation times

	True Value	Estimate	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3433	0.01046	(5.3227, 5.3639)
$\log_{10}(E_1)$	1.6021	1.6361	0.1591	(1.3227, 1.9496)
$\log_{10}(\tau_1)$	-1.3010	-0.1990	15.8821	(-31.4806, 31.0827)
$\log_{10}(\tau_2)$	1	0.2496	12.0051	(-23.3958, 23.8951)

Model selection criteria There are numerous model selection criteria in the literature that can be used to select a best approximating model from a prior set of candidate models. These criteria are based either on hypothesis testing or mean squared error or Bayes factors or information theory, and they all are based to some extent on the principle of parsimony (see [BuAn]). It should be noted that some of these criteria can only be used for nested models (e.g., two models are said to be nested if one model is a special case of the other), but others can be used for both nested models and non-nested models. Here we employ one of the most widely used model selection criteria – the Akaike information criterion (AIC). The AIC was developed by Akaike (in 1973) who formulated a relationship between the Kullback-Leibler information (used to measure the information lost when a model is used to approximate the full reality) and the maximum value of the log likelihood function of the approximating model. As might be expected we find that the AIC value depends on the data set used. Thus, when we try to select a best model from a set of candidate models, we must use the same data set to calculate AIC values for each of the models. One of the advantages of the AIC is that it can be used to compare non-nested models (which is our case here). For the least squares case, it can be found (e.g., see [BuAn, Section 2.2]) that if the observation errors are i.i.d normally distributed, then the AIC is given by

$$AIC = n \log\left(\frac{\text{RSS}}{n}\right) + 2(\kappa + 1). \tag{13}$$

Here $\kappa + 1$ is the total number of estimated parameters including θ and the observation error variance. Given a prior set of candidate models, we can calculate the AIC value for each model, and the best approximating model is the one with minimum AIC value. It should be noted that the AIC may perform poorly if the sample size n is small relative to the total number of estimated parameters (it is suggested in [BuAn] that the sample size n should be at least 40 times the total number of estimated parameters $(\kappa + 1)$; note this is true for our investigations).

In practice, the absolute size of the AIC value may have limited use in supporting the chosen best approximating model, and one may often employ other related values such as Akaike differences and Akaike weights to further compare models. The Akaike difference is defined by

$$\Delta_i = AIC_i - AIC_{\min}, \quad i = 1, 2, \ldots R, \tag{14}$$

where AIC_i is the AIC value of the ith model in the set, $AIC_{\min}$ denotes the AIC value for the best model in the set, and R is the total number of models in the set. The larger Δ_i, the less plausible it is that the ith model is a good approximating model for given the data set. The Akaike weights are defined by

$$w_i = \frac{\exp(-\frac{1}{2}\Delta_i)}{\sum_{r=1}^{R} \exp(-\frac{1}{2}\Delta_r)}, \quad i = 1, 2, \ldots R. \tag{15}$$

These Akaike weights w_i can then be interpreted as the probability that ith model is the best approximating model (see [BuAn]).

Tables 10 and 11 present residual sum squares (RSS), AIC values, AIC differences, and AIC weights obtained for the two-relaxation-time model, the one-relaxation-time model and the no-relaxation-time model at low, medium and high noise levels using `fmincon` and `lsqnonlin`, respectively. From these two tables, we see that on low and medium level noise data sets the

Table 10: `fmincon`: Residual sum of squares (RSS), AIC values, AIC differences (Δ) and AIC weights for zero-relaxation-time model (model 0), one-relaxation-time model (model 1) and two-times-relaxation model (model 2) obtained at low, medium and high noise levels.

noise level	model	RSS	AIC	Δ	AIC weights
low noise	0	7.0368×10^{-9}	-6.0927×10^{3}	7.5867	1.5257×10^{-2}
	1	6.7731×10^{-9}	-6.1003×10^{3}	0	6.7748×10^{-1}
	2	6.7618×10^{-9}	-6.0987×10^{3}	1.5814	3.0726×10^{-1}
medium noise	0	3.6421×10^{-8}	-5.6800×10^{3}	98.1093	3.5908×10^{-22}
	1	2.4442×10^{-8}	-5.7782×10^{3}	0	7.2334×10^{-1}
	2	2.4435×10^{-8}	-5.7762×10^{3}	1.9222	2.7666×10^{-1}
high noise	0	1.0337×10^{-7}	-5.4182×10^{3}	0	6.4532×10^{-1}
	1	1.0330×10^{-7}	-5.4164×10^{3}	1.8267	2.5889×10^{-1}
	2	1.0329×10^{-7}	-5.4144×10^{3}	3.8151	9.5794×10^{-2}

Table 11: `lsqnonlin`: Residual sum of squares (RSS), AIC values, AIC differences (Δ) and AIC weights for zero-relaxation-time model (model 0), one-relaxation-time model (model 1) and two-times-relaxation model (model 2) obtained at low, medium and high noise levels.

noise level	model	RSS	AIC	Δ	AIC weights
low noise	0	7.0368×10^{-9}	-6.0927×10^{3}	27.8791	6.4125×10^{-7}
	1	6.2470×10^{-9}	-6.1206×10^{3}	0	7.2595×10^{-1}
	2	6.2458×10^{-9}	-6.1186×10^{3}	1.9483	2.7405×10^{-1}
medium noise	0	2.4674×10^{-8}	-5.7778×10^{3}	8.6863	9.4255×10^{-3}
	1	2.3646×10^{-8}	-5.7865×10^{3}	0	7.2527×10^{-1}
	2	2.3647×10^{-8}	-5.7845×10^{3}	2.0113	2.6531×10^{-1}
high noise	0	1.0337×10^{-7}	-5.4182×10^{3}	0	6.4303×10^{-1}
	1	1.0330×10^{-7}	-5.4164×10^{3}	1.8299	2.5756×10^{-1}
	2	1.0326×10^{-7}	-5.4145×10^{3}	3.7340	9.9406×10^{-2}

one-relaxation-time model is the best with the probability to be chosen as the best model being more than 0.6 (see the Akaike weights in the last column of these two tables), and the no-relaxation time model has almost no chance of being selected as the best. For the high noise data set, the no-relaxation-time model is the best, with the probability of being chosen as the best being more than 0.6, while the two-relaxation-time model has little chance of being selected as the best model.

Summary remark Based on our analysis to this point, we can conclude that estimating two relaxation times is likely to be difficult. Adopting a model with zero or one relaxation times may be the most feasible approach. However, until we can confirm this approach by examining these methods on experimental data we believe that attempting all three options for including relaxation times in the viscoelastic model (zero, one, or two times) is advisable.

2.3.4 Bootstrapping error analysis

For ease of presentation, we reiterate here the algorithm described in [BaHoRo], in the context of the current viscoelastic model under study.

1. Determine $\hat{\theta}^0$ by computing (12).

2. Define the *standardized residuals* (recall n is the number of data points, and κ is the number of parameters under consideration) to be

$$\bar{r}_j = \sqrt{\frac{n}{n-\kappa}}\left(u_j - u(L, t_j; 10^{\hat{\theta}^0})\right)$$

   for $j = 0, 1, \ldots, n-1$. Set $m = 0$.

3. Create a sample of size n by randomly sampling, with replacement, from the standardized residuals $\bar{r}_j$ to form a bootstrap sample $\{r_0^m, \ldots r_{n-1}^m\}$.

4. Create bootstrap sample points $u_j^m = u(L, t_j; 10^{\hat{\theta}^0}) + r_j^m$ for $j = 0, 1, \ldots, n-1$.

5. Solve the OLS minimization problem (12) with the bootstrap-generated data $\{u_j^m\}$ to obtain a new estimate $\hat{\theta}^{m+1}$ which we store.

6. Increase the index m by 1 and repeat steps 3-5. This iterative process should be carried out for M times where M is large (we used $M = 1000$, as suggested in [BaHoRo]). This will give M estimates $\{\hat{\theta}^m\}_{m=1}^M$.

Upon completing all M simulation runs, the following will give the mean, covariance, and standard errors for the bootstrap estimator $\hat{\Theta}_{boot}$:

$$\begin{aligned}
\hat{\theta}_{boot} &= \frac{1}{M}\sum_{m=1}^{M}\hat{\theta}^m, \\
\hat{\Sigma}_{boot} &= \frac{1}{M-1}\sum_{m=1}^{M}(\hat{\theta}^m - \hat{\theta}_{boot})(\hat{\theta}^m - \hat{\theta}_{boot})^T, \\
(SE_{boot})_i &= \sqrt{\left(\hat{\Sigma}_{boot}\right)_{ii}}, \quad i = 1, 2, \ldots, \kappa,
\end{aligned} \tag{16}$$

where $\left(\hat{\Sigma}_{boot}\right)_{ii}$ is the (i,i)th entry of covariance matrix $\hat{\Sigma}_{boot}$. Hence, the endpoints of the confidence intervals for $(\hat{\Theta}_{boot})_i$ (the ith element of $\hat{\Theta}_{boot}$) are given by

$$(\hat{\theta}_{boot})_i \pm t_{1-\alpha/2}(SE_{boot})_i$$

for $i = 1, 2, \ldots, \kappa$.

Note that bootstrapping requires solving the inverse problem 1000 times. Even for a model that is solved in a short time (e.g., less than one minute), bootstrapping takes a significant time to compute (as we must solve the inverse problem many times and each inverse problem involves solving

the model multiple times). Due to long computational times (e.g., one week for bootstrapping versus minutes for the asymptotic theory), we report here results only for a case using `fmincon` to estimate E_0 and E_1 in a zero-relaxation-time model and a case using `lsqnonlin`, the trust-region-reflective method, to estimate E_0, E_1, and τ_1 in a one-relaxation-time model. It is worth noting that even though the bootstrapping algorithm can be implemented in parallel, this requires a considerable amount of computing resources (unavailable to most investigators) to achieve computational times comparable to that attained in using the asymptotic theory. For our purposes, the bootstrap results we provide are sufficient to indicate that the less conservative asymptotic error analysis yields a reasonable uncertainty measure in the inverse problem we investigate.

For the model with no relaxation times, we see from Table 12 that the confidence intervals for E_0 and E_1 at all noise levels are more conservative (as expected-see [BaHoRo]) than those obtained using the asymptotic theory (shown in Tables 4(a), 5(a) and 6(a)), especially for the cases of medium and high noise level. However, this table still indicates that reasonable parameter estimates are obtained.

Table 12: Bootstrapping results, no relaxation times, using `fmincon`: Parameter estimates, bootstrap standard errors (SE) and confidence intervals

(a) Low noise level

	True Value	$\hat{\theta}_{boot}$	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3422	0.01557	(5.3115, 5.3729)
$\log_{10}(E_1)$	1.6021	1.6649	0.1716	(1.3270, 2.0029)

(b) Medium noise level

	True Value	$\hat{\theta}_{boot}$	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3429	0.03038	(5.2831, 5.4028)
$\log_{10}(E_1)$	1.6021	1.6653	0.3155	(1.0438, 2.2867)

(c) High noise level

	True Value	$\hat{\theta}_{boot}$	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3432	0.05714	(5.2306, 5.4557)
$\log_{10}(E_1)$	1.6021	1.6451	0.6603	(0.3446, 2.9456)

In Figure 10, we depict the bootstrap estimates obtained for E_0 and E_1 for this no relaxation time model. Note that each empirical distribution (a representation for the estimator distribution) tends to have the shape of a normal distribution, which we would expect if bootstrapping is working properly.

We also present bootstrapping results for a model incorporating one relaxation time. The inverse problem solved during the bootstrap procedure was computing using `lsqnonlin`, the trust-region-reflective algorithm. The results are summarized in Table 13. We see that confidence intervals for all parameters are wider than those obtained using the asymptotic error theory, especially for the cases of medium and high noise level. However, this is expected. At the low noise level, we obtained fairly good results for E_0 and E_1 but the standard error for the relaxation time is larger in

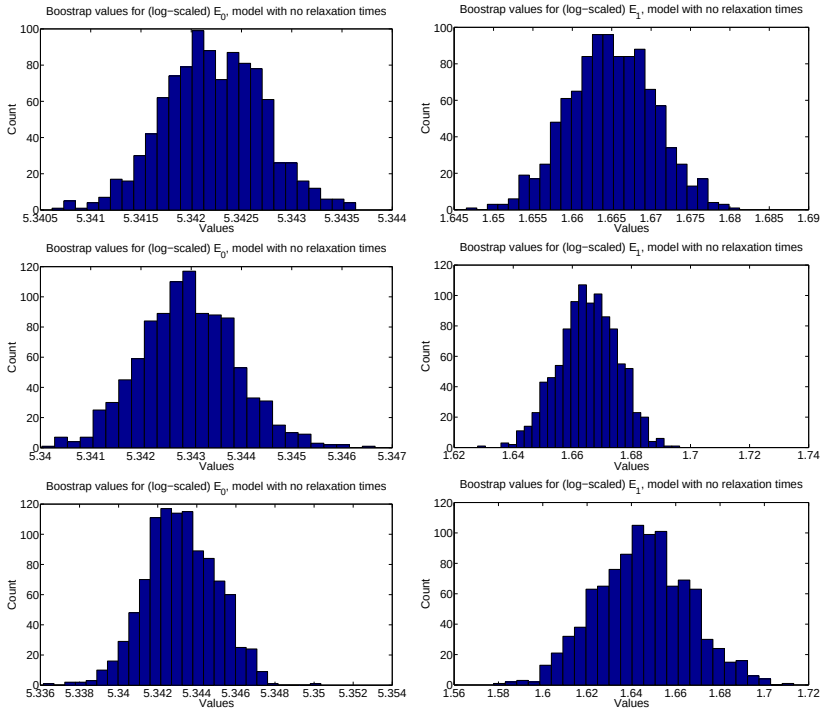

Figure 10: Histograms of bootstrap estimates $\hat{\theta}^m$ for a model with no relaxation times in the case of low noise data set (upper row), medium noise data set (middle row) and high noise data set (bottom row). (left column) Estimates for $\log_{10}(E_0)$; (right column) Estimates for $\log_{10}(E_1)$.

magnitude than the relaxation time value itself. This is even more prominent at higher noise levels – the results in the table indicate that on medium and high noise data sets, the estimation of τ_1 is not very robust. Note also that the estimation of E_1 begins to suffer as well, resulting in a higher standard error than its own value on the high noise data set. This is a further indication that we may have problems in the future estimating even the single relaxation time.

We depict histograms of the estimates in Figure 11. We see on a low noise data set that each parameter estimator appears to be mostly normally distributed. This begins to break down for the case of middle noise level data set (shown in the middle row of Figure 11), where we begin to see some outliers at the $\log_{10}(\hat{\tau}_1) = 2$ level (which means the estimates were converging to our upper bound on that parameter) and also some more pronounced skewness in the count levels. Finally, on the high noise level (shown in bottom row of Figure 11) we have the distribution for the E_1 estimates skewed, and we also observe a clear proliferation of estimates of the first relaxation time approaching the value 2. This further supports the expectation of difficulty in estimating relaxation times, particularly when the noise level is high.

Table 13: Bootstrapping results, one relaxation time, using `lsqnonlin`: Parameter estimates, bootstrap standard errors (SE) and confidence intervals

(a) Low noise level

	True Value	θ_{boot}	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3425	0.01547	(5.3120, 5.3730)
$\log_{10}(E_1)$	1.6021	1.6025	0.5937	(0.4332, 2.7719)
$\log_{10}(\tau_1)$	-1.3010	-1.2294	3.8697	(-8.8510, 6.3923)

(b) Medium noise level

	True Value	θ_{boot}	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3434	0.03136	(5.2816, 5.4052)
$\log_{10}(E_1)$	1.6021	1.5852	1.4590	(-1.2884, 4.4589)
$\log_{10}(\tau_1)$	-1.3010	-1.2079	16.9971	(-34.6849, 32.2692)

(c) High noise level

	True Value	θ_{boot}	SE	95% Confidence Interval
$\log_{10}(E_0)$	5.3424	5.3434	0.061762	(5.2218, 5.4651)
$\log_{10}(E_1)$	1.6021	1.6029	2.4545	(-3.2314, 6.4372)
$\log_{10}(\tau_1)$	-1.3010	-0.1592	40.8381	(-80.5930, 80.2746)

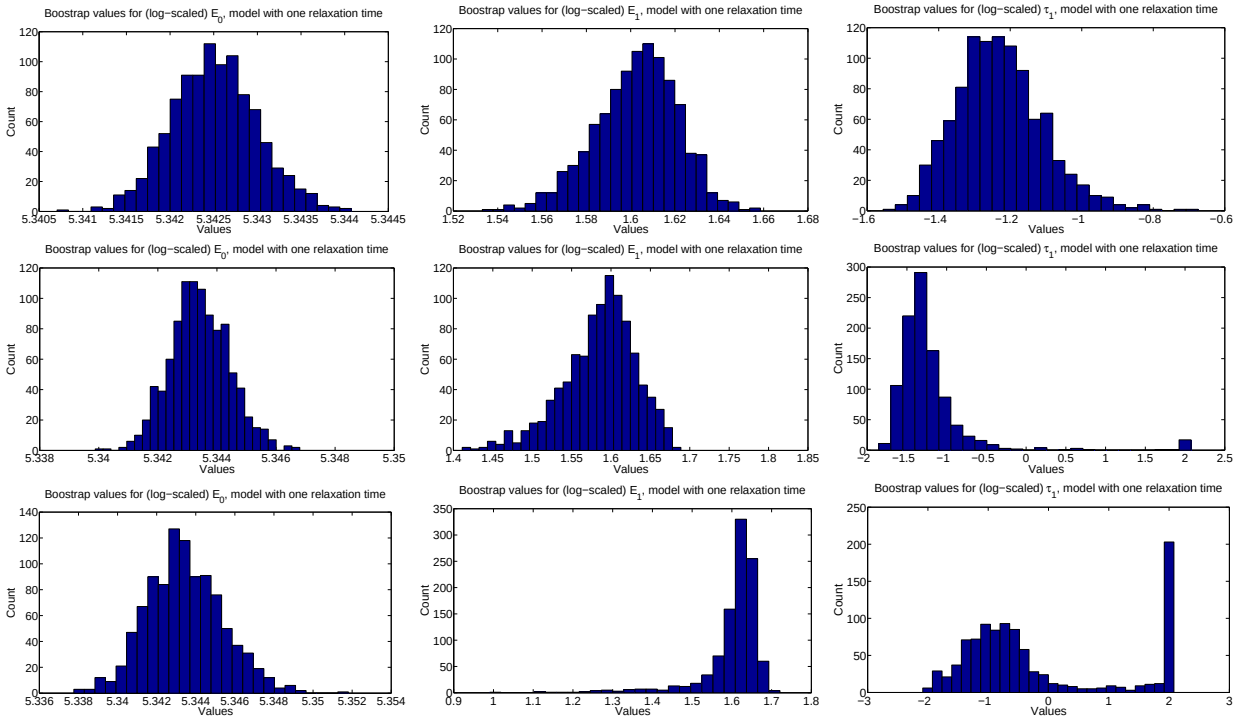

Figure 11: Histograms of bootstrap estimates $\hat{\theta}^m$ for a model with one relaxation time in the case of low noise data set (upper row), medium noise data set (middle row) and high noise data set (bottom row). (left column) Estimates for $\log_{10}(E_0)$; (middle column) Estimates for $\log_{10}(\tau_1)$; (right column) Estimates for $\log_{10}(E_1)$.

3 Model comparison and hypothesis testing on amplitude

In this section, we develop a methodology for determining whether or not data came from a low-amplitude input traction. This corresponds with determining if the data came from a vessel experiencing a normal heartbeat or not. We will ultimately run the inverse problem without amplitude restrictions and use a scoring function to compare results with the score of the model solved at a low amplitude. A model comparison test will be implemented to determine if there is a statistical significance in the differences between the model solved with the unrestricted estimate and the model solved using the restricted amplitude value.

3.1 Setup

We first examine the sensitivity of the model with respect to the Van Bladel input amplitude parameter A, to be sure that an estimation procedure is reasonable (if the model were insensitive to A then the results from the optimization routine would be suspect). The form of the sensitivity equation is nearly identical to that of the actual model, just with a lower amplitude. This is seen in Figure 12, which has a form similar to that of the model solution (Figure 3). In both the low and high amplitude cases, the sensitivity with respect to amplitude is most marked during early times and less so at later times; this makes perfect sense, as the amplitude is greater early on before being damped out. In the problem below, we will take data throughout the full time frame $t \in [0, 0.25]$ so with our sensitivity results we can be assured that the early data will drive estimation of the amplitude parameter.

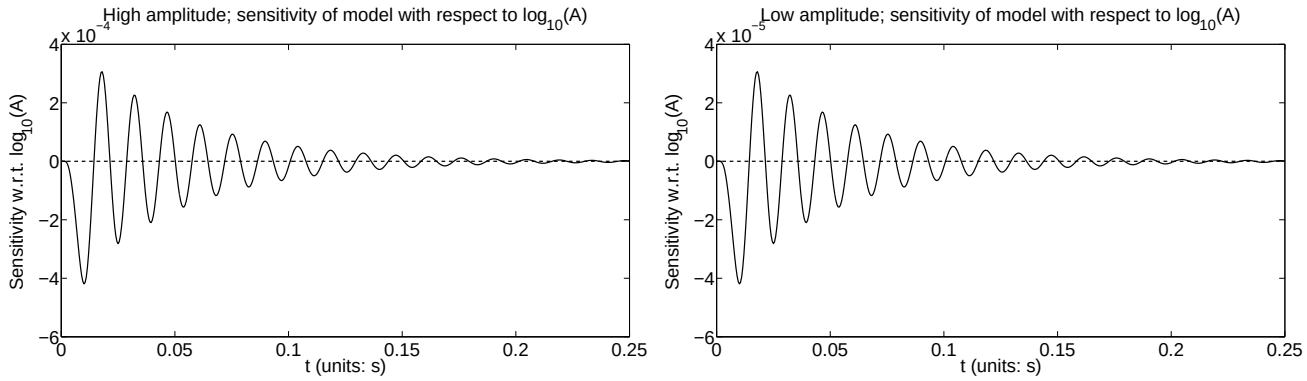

Figure 12: Sensitivity of model with respect to Van Bladel input parameter A around the baseline parameters (6). (left pane) High forcing function amplitude $A = 6 \times 10^3$. (right pane) Low forcing function amplitude $A = 6 \times 10^2$.

3.2 Data generation

For the high amplitude data, we use the same low, medium, and high noise data sets as described in Section 2.3.1 and shown above in Figure 8. We form the low amplitude data by taking $A_{low} = A/10$ as our Van Bladel input amplitude parameter. Thus, the dynamics are roughly 10% the magnitude of the high amplitude data. This means the corresponding noise for the low noise, low amplitude

data set will be generated with variance $\sigma^2 = 5 \times 10^{-7}$, medium noise with $\sigma^2 = 10 \times 10^{-7}$, and high noise with $\sigma^2 = 20 \times 10^{-7}$. The low amplitude input data set then is supposed to represent a normal heartbeat and the high amplitude data set then is meant to represent the input shear for a heartbeat in the presence of a stenosis in the vessel. Note that we are not yet exactly certain regarding the difference between these effects in an actual patient, so the data sets here are truly for a proof-of-concept investigation. The low amplitude data sets are depicted in Figure 13.

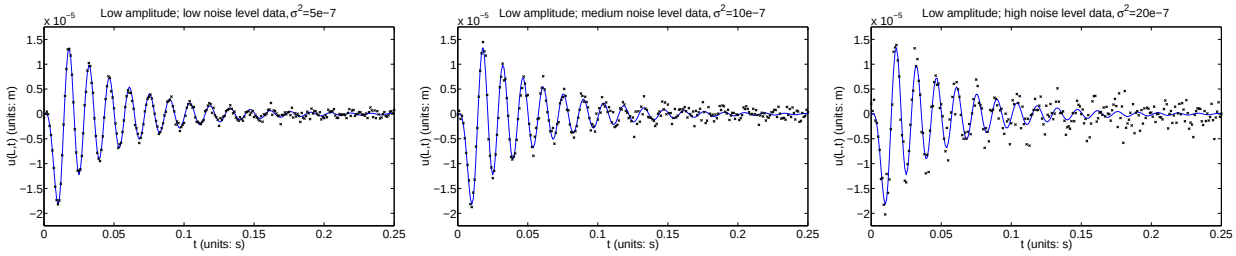

Figure 13: Simulated low amplitude noisy data around the true parameter values. (left pane) Low noise level, $\sigma_0^2 = 5 \times 10^{-7}$. (middle pane) Medium noise, $\sigma_0^2 = 10 \times 10^{-7}$. (right pane) High noise level, $\sigma_0^2 = 20 \times 10^{-7}$.

3.3 Hypothesis testing methodology

We can now begin to discuss the approach to model comparison and hypothesis testing that we will use by defining a model comparison test statistic. The work here follows the development in [BaTr]. Our performance criterion for hypothesis testing will be

$$J(\vec{U}, \theta) = \sum_{j=0}^{n-1} [U_j - u(L, t_j; \theta)]^2.$$

For the purposes of this paper, we postulate that a normal (non-stenosed) vessel corresponds with a low amplitude input parameter $A \leq 6 \times 10^2$. Then, a stenosed vessel would have a high input amplitude parameter with $A > 6 \times 10^2$. The hypothesis test we use requires a set benchmark value for A, so we choose that benchmark to be $A_0 = 6 \times 10^2$. Then, we define the restricted parameter set

$$\mathbb{A}_H = \{A \in \mathbb{A} | A = A_0 = 6 \times 10^2\},$$

where $\mathbb{A} = [A_0, \infty)$ is the larger set of unrestricted admissible amplitudes.

Our null hypothesis H_0 is that the amplitude is a low amplitude, represented by $A \in \mathbb{A}_H = \{A_0\}$. The unrestricted amplitude model would then represent the amplitude parameter as $A = A_0 + \tilde{A}$ where $\tilde{A} \in [0, \infty)$. This framework will allow us to develop a test statistic to determine the confidence level of accepting or rejecting H_0 for a given data set. In other words, we will develop a test to determine if the data is statistically better represented by the benchmark A_0 than the unrestricted amplitude.

The first step is to determine the performance criterion at the benchmark amplitude $\hat{\theta}_H = 6 \times 10^2$, which we will denote $J(\vec{u}, \hat{\theta}_H)$ (Since the value $\hat{\theta}_H$ is fixed in our case, no optimization problem is

needed to compute these values). We then run an optimization routine to determine an unrestricted input amplitude parameter estimate $\hat{\theta}$, which we then use to compute $J(\vec{u}, \hat{\theta})$. The value for $\hat{\theta}$ comes from solving the unrestricted optimization problem (12). As discussed in [BaTr, BaFi], the model comparison statistic is defined as

$$\hat{V} = n\frac{J(\vec{U}, \hat{\Theta}_H) - J(\vec{U}, \hat{\Theta})}{J(\vec{U}, \hat{\Theta})}$$

with realization

$$\hat{v} = n\frac{J(\vec{u}, \hat{\theta}_H) - J(\vec{u}, \hat{\theta})}{J(\vec{u}, \hat{\theta})}. \tag{17}$$

If our null hypothesis H_0 were true, the model comparison statistic $\hat{V}$ converges in distribution to V as $n \to \infty$ where $V \sim \chi^2(r)$ is a chi-square distribution with r degrees of freedom (r is the number of constraints in $\mathbb{A}_H$). For our problem, $r = 1$. Given the significance level α, we can obtain a threshold value ν such that the probability that V will take on a value greater than ν is α. In other words, $\text{Prob}(V > \nu) = \alpha$. In our context, if the test statistic $\hat{v} > \nu$ we reject H_0 as false with confidence level $(1 - \alpha)100\%$. Otherwise we do not reject H_0 as false, at the specified confidence level. In Table 14 we include sample values from the $\chi^2(1)$ distribution for reference (table repeated from [BaTr]).

Table 14: Sample $\chi^2(1)$ values.

α	ν	confidence
0.25	1.32	75%
0.1	2.71	90%
0.05	3.84	95%
0.01	6.63	99%
0.001	10.83	99.9%

We summarize in Table 15 the results of computing the OLS performance criterion for the low amplitude and high amplitude data each with the restricted/unrestricted parameters. Based on this table and Table 14, we see for both the low and medium noise cases for data generated with a

Table 15: Model comparison test results using (17) on low, medium, and high noise data sets generated with both high and low input amplitude parameter A values.

	$J(\vec{u}, \hat{\theta})$	$J(\vec{u}, \hat{\theta}_H)$	$\hat{v}$
Low A, low noise	6.3846e-11	6.3887e-11	0.1609
Low A, medium noise	2.6872e-10	2.6896e-10	0.2258
Low A, high noise	9.8836e-10	9.9658e-10	2.0878
High A, low noise	6.6812×10^{-9}	3.5229×10^{-7}	1.2984e+04
High A, medium noise	3.1016×10^{-8}	3.4730×10^{-7}	2.5596e+03
High A, high noise	9.9737×10^{-8}	4.6015×10^{-7}	907.0283

low A value that we do not reject H_0 with high degrees of confidence. However, the case with high noise is somewhat less certain, though we would still likely not reject H_0 with a fairly high degree of confidence. The results are more stark in the cases where the data was generated from a high amplitude. Given that the magnitude of $\hat{v}$ is greater than 900 at all noise levels, we would reject H_0 as false on these data sets with confidence level more than 99.9%. Altogether, these results suggest robustness in our methodology for determining whether the data came from a normal vessel experiencing a heartbeat (low input amplitude) or not.

4 Conclusion

In this work we have carried out proof-of-concept investigations for estimating material parameters and created a model comparison test as a basis for distinguishing between data that comes from a normal or from a stenosed blood vessel. We found that the model was less sensitive to a second viscoelastic relaxation time than to the other parameters, and this was manifested as a difficulty in recovering two relaxation times. On the other hand, models with zero or one relaxation time allowed for more confidence in the estimation procedure (i.e., smaller standard errors). We compared asymptotic error theory with bootstrapping error theory, and found (as expected) that bootstrapping gives more conservative confidence intervals but not so much so that the asymptotic theory cannot be profitably used for uncertainty quantification in models with large computational costs rendering bootstrapping less desirable. In terms of the model comparison on the input amplitude parameter A, we were able to develop a successful methodology for statistically determining whether or not data came from a low amplitude input force. This will form the basis of a model comparison test we can use on experimental data sets.

In future efforts, as already mentioned we will need to develop a procedure for estimating the material weights p_i. This may involve iterating optimization routines to take advantage of features of `fmincon` for estimating the weights, while using the routine `lsqnonlin` for the remaining material parameters. We also plan to examine the possibility of relative error instead of absolute error, which will necessitate a generalized least squares (GLS) cost function in our inverse problems due to changes in the error process. This will be coupled with a study of a statistical model for the measurement processes being used in the experiments at QMUL. The changes needed are discussed in [BaTr]. If we require a GLS framework, we will need to derive a proper model comparison framework for GLS problems. The most important immediate efforts will be to apply the methods presented in this paper to the data from the QMUL experiments.

5 Acknowledgements

This research was supported in part (HTB,SH,ZRK) by Grant Number R01AI071915-09 from the National Institute of Allergy and Infectious Diseases, in part (HTB,SH,ZRK) by the Air Force Office of Scientific Research under grant number FA9550-09-1-0226, in part (ZRK) by the Department of Education with a GAANN Fellowship under grant number P200A070386, in part (CK,SS,JW) by the Engineering and Physical Sciences Research Council EP/H011072/1, and in part (MBr,SG,MBi) by the Engineering and Physical Sciences Research Council EP/H011285/1.

References

[BBETW] H.T. Banks, J.H. Barnes, A. Eberhardt, H. Tran, and S. Wynne, Modeling and computation of propagating waves from coronary stenoses, *Computation and Applied Mathematics*, **21** (2002), 767–788.

[BaFi] H.T. Banks and B.G. Fitzpatrick, Inverse problems for distributed systems: statistical tests and ANOVA, LCDS/CCS Rep. 88-16, July, 1988, Brown University; *Proc. International Symposium on Math. Approaches to Envir. and Ecol. Problems*, Springer Lecture Notes in Biomath., **81** (1989), 262–273.

[BaHoRo] H.T. Banks, K. Holm and D. Robbins, Standard error computations for uncertainty quantication in inverse problems: Asymptotic theory vs. Bootstrapping, CRSC-TR09-13, North Carolina State University, June 2009; Revised May 2010; *Mathematical and Computer Modelling*, **52** (2010), 1610–1625.

[BaHuKe] H.T. Banks, S. Hu, Z.R. Kenz, A brief review of elasticity and viscoelasticity for solids, *Advances in Applied Mathematics and Mechanics*, **3** (2011), 1–51.

[BaLu] H.T. Banks and N. Luke, Modeling of propagating shear waves in biotissue employing an internal variable approach to dissipation, *Communication in Computational Physics*, **3** (2008), 603–640.

[BaMedPint] H.T. Banks, N. Medhin, and G. Pinter, Multiscale considerations in modeling of nonlinear elastomers, *International Journal for Computational Methods in Engineering Science and Mechanics*, **8** (2007), 53–62.

[BaPint] H.T. Banks and G.A. Pinter, A probabilisitic multiscale approach to hysteresis in shear wave propagation in biotissue, *Multiscale Modeling and Simulation*, **3** (2005), 395–412.

[BaSam] H.T. Banks and J.R. Samuels, Jr., Detection of cardiac occlusions using viscoelastic wave propagation, *Advances in Applied Mathematics and Mechanics*, **1** (2009), 1–28.

[BaTr] H.T. Banks and H.T. Tran, *Mathematical and Experimental Modeling of Physical and Biological Processes*, CRC Press, Boca Raton, FL, 2009.

[BuAn] K.P. Burnham and D.R. Anderson, *Model Selection and Inference: A Practical Information-Theoretical Approach*, Springer-Verlag, New York, 1998.

[CoNo] B.D. Coleman and W. Noll, Foundations of linear viscoelasticity, *Reviews of Modern Physics*, **33** (1961), 239–249.

[FaMo] M. Fabrizio and A. Morro, *Mathematical Problems in Linear Viscoelasticity*, Studies in Applied Mathematics, Vol. 12, SIAM, Philadelphia, 1992.

[Fer] J.D. Ferry, *Viscoelastic Properties of Polymers*, John Wiley and Sons Inc., New York, 1970.

[FiLaOn] W.N. Findley, J.S. Lai, and K. Onaran, *Creep and Relaxation of Nonlinear Viscoelastic Materials (With an Introduction to Linear Viscoelasticity)*, Dover Publications Inc., New York, 1989.

[GoGr] J.M. Golen and G.A.C. Graham, *Boundary Value Problems in Linear Viscoelasticity*, Springer-Verlag, New York, 1988.

[BICOM-2012] Carola Kruse, Matthias Maischak, Simon Shaw, J.R. Whiteman, et al., High order space-time finite element schemes for acoustic and viscodynamic wave equations with temporal decoupling, BICOM Technical Report 12/3, 2012, in preparation.

[Lak] R. Lakes, *Viscoelastic Materials*, Cambridge Univ. Press, Cambridge, 2009.

[Luke] N. Luke, *Modeling Shear Wave Propagation in Biotissue: An Internal Variable Approach to Dissipation*, PhD Thesis, NCSU, Raleigh, 2006.

[Niss] S.E. Nissen, Application of intravascular ultrasound to characterize coronary artery disease and assess the progression or regression of atherosclerosis, *Am. J. Cardiol.*, **89** (2002), 24B–31B.

[NiYo] S.E. Nissen and P. Yock, Intravascular ultrasound: novel pathophysiological insights and current clinical applications, *Circulation*, **103** (2001), 604–616.

[OwHu] N.L. Owsley and A.J. Hull, Beamformed nearfield imaging of a simulated coronary artery containing a stenosis, *IEEE Trans. on Medical Imaging*, **17** (1998), 900–909.

[OHAK] N.L. Owsley, A.J. Hull, M.H. Ahmed, and J. Kassal, A proof of concept experiment for the detection of occluded coronary arteries using array sensor technology, *Engr. in Medicine and Biol. Society*, IEEE 17th Annual Conf., **1** (1995), 145–146.

[Sam] J.R. Samuels, Jr., *Inverse Problems and Post Analysis Techniques for a Stenosis-driven Acoustic Wave Propagation Model*, PhD Thesis, NCSU, Raleigh, 2008.

[SWKM] J. Semmlow, J. Welkowtiz, W. Kostis, and J.W. Mackenzie, Coronary artery disease – correlates between diastolic auditory characteristics and coronary artery stenoses, *IEEE Trans. on Biomed. Engr.*, **30** (1983), 136–139.

[Ver] J. Verburg, Transmission of vibrations of the heart to the chestwall, *Adv. Cardiovasc. Phys.*, **5** (1983), 84–103.

[VeVa] J. Verburg and E. van Vollenhoven, Phonocardiography: Physical and technical aspects and clinical uses, in *Noninvasive Physiological Measurements* (ed. P Rolfe), Academic Press, London (1979).

A Sensitivity equations

In this section, we present the computed sensitivity equations used in the standard error calculations in this effort. Let $\bar{\theta} = (\rho, E_0, E_1, \tau_1, \ldots, \tau_{N_p}, p_1, \ldots, p_{N_p}, A)^T$ be the vector for the parameters requiring estimation, and $s^{\bar{\theta}_i}(x,t;\bar{\theta}) = \dfrac{\partial u(x,t;\bar{\theta})}{\partial \bar{\theta}_i}$, $i = 1, 2, \ldots$ with $\bar{\theta}_i$ being the ith component of $\bar{\theta}$. We first find the equation for $s^{\rho}(x,t;\bar{\theta})$. To do that, we take the partial derivatives of the system (1) with respect to ρ and obtain

$$\begin{aligned} &\rho u_{\rho tt} + u_{tt} - \sigma_{\rho x} = 0 \\ &u_{\rho}(0,t) = 0, \quad \sigma_{\rho}(L,t) = 0, \\ &u_{\rho}(x,0) = 0, \quad u_{\rho t}(x,0) = 0. \end{aligned} \tag{18}$$

(Note that the chain rule on the first term resulted in $\dfrac{\partial}{\partial \rho}(\rho u_{tt}) = \rho u_{\rho tt} + u_{tt}$, which means the sensitivity partial differential equation (PDE) will, not surprisingly, be driven by the original system values.) By changing the order of differentiation and then substituting s^{ρ} for $\dfrac{\partial u}{\partial \rho}$, equation (18) can be rewritten as

$$\begin{aligned} &\rho (s^{\rho})_{tt} - \frac{\partial}{\partial x}\sigma_{\rho} = -u_{tt} \\ &s^{\rho}(0,t) = 0, \quad \sigma_{\rho}(L,t) = 0, \\ &s^{\rho}(x,0) = 0, \quad (s^{\rho})_t(x,0) = 0, \end{aligned} \tag{19}$$

In the above equation, the sensitivity of stress with respect to ρ (i.e., σ_{ρ}) can be obtained by differentiating both sides of (4) with respect to ρ

$$\sigma_{\rho} \;=\; E_1 (s^{\rho})_{xt} + E_0 \left((s^{\rho})_x - \sum_{j=1}^{N_p} \epsilon_{\rho}^{j} \right). \tag{20}$$

Here the sensitivity of internal variable ϵ^j with respect to ρ (i.e., ϵ_{ρ}^{j}) satisfies the following equation

$$\begin{aligned} &\tau_j (\epsilon_{\rho}^{j})_t + \epsilon_{\rho}^{j} = p_j (s^{\rho})_x, \\ &\epsilon_{\rho}^{j}(0) = 0, \end{aligned} \tag{21}$$

where $j = 1, \ldots, N_p$. The above equation is obtained by differentiating both sides of (3) with respect to ρ. Thus, we see the sensitivity PDE (19)-(21) for ρ has the same form as the original system PDE (1), (3), and (4), except with zero initial/boundary conditions and being driven by the solution to the original system PDE (i.e., along the original system solution). Thus, as usual the sensitivity PDE is coupled to the original system PDE.

We now state the remaining sensitivity PDEs.

1. Sensitivity PDE for $s^{E_0} = \dfrac{\partial u}{\partial E_0}$:

As we shall see, solving this sensitivity PDE requires knowledge of the internal variables, in particular the first derivative with respect to x (i.e, we need to know ϵ_x^j since we have the $(\sigma_{E_0})_x$ term in (22), and σ_{E_0} incorporates ϵ^j). Specifically, s^{E_0} satisfies

$$\begin{aligned} &\rho(s^{E_0})_{tt} - (\sigma_{E_0})_x = 0 \\ &s^{E_0}(0,t) = 0, \quad \sigma_{E_0}(L,t) = 0, \\ &s^{E_0}(x,0) = 0, \quad (s^{E_0})_t(x,0) = 0. \end{aligned} \tag{22}$$

In the above equation, the sensitivity of stress with respect to E_0 (i.e., σ_{E_0}) is given by

$$\sigma_{E_0} = E_1(s^{E_0})_{xt} + E_0\left((s^{E_0})_x - \sum_{j=1}^{N_p} \epsilon_{E_0}^j\right) + \left(u_x - \sum_{j=1}^{N_p} \epsilon^j\right). \tag{23}$$

Here the sensitivity of internal variables ϵ^j with respect to E_0 (i.e., $\epsilon_{E_0}^j = \dfrac{\partial \epsilon^j}{\partial E_0}$) satisfies the following equation

$$\begin{aligned} \tau_j(\epsilon_{E_0}^j)_t + \epsilon_{E_0}^j &= p_j(s^{E_0})_x, \\ \epsilon_{E_0}^j(0) &= 0, \end{aligned} \tag{24}$$

where $j = 1, \ldots, N_p$.

2. Sensitivity PDE for $s^{E_1} = \dfrac{\partial u}{\partial E_1}$:

$$\begin{aligned} &\rho(s^{E_1})_{tt} - (\sigma_{E_1})_x = 0 \\ &s^{E_1}(0,t) = 0, \quad \sigma_{E_1}(L,t) = 0, \\ &s^{E_1}(x,0) = 0, \quad (s^{E_1})_t(x,0) = 0. \end{aligned} \tag{25}$$

In the above equation, the sensitivity of stress with respect to E_1 (i.e., σ_{E_1}) is given by

$$\sigma_{E_1} = E_1(s^{E_1})_{xt} + E_0\left((s^{E_1})_x - \sum_{j=1}^{N_p} \epsilon_{E_1}^j\right) + u_{xt}. \tag{26}$$

Here the sensitivity of internal variables ϵ^j with respect to E_1 (i.e., $\epsilon_{E_1}^j = \dfrac{\partial \epsilon^j}{\partial E_1}$) satisfies the following equation

$$\begin{aligned} \tau_j(\epsilon_{E_1}^j)_t + \epsilon_{E_1}^j &= p_j(s^{E_1})_x, \\ \epsilon_{E_1}^j(0) &= 0, \end{aligned} \tag{27}$$

where $j = 1, \ldots, N_p$.

3. Sensitivity PDE for $s^{\tau_k} = \dfrac{\partial u}{\partial \tau_k}$, $k = 1, \ldots, N_p$:

$$
\begin{aligned}
&\rho(s^{\tau_k})_{tt} - (\sigma_{\tau_k})_x = 0 \\
&s^{\tau_k}(0,t) = 0, \quad \sigma_{\tau_k}(L,t) = 0, \\
&s^{\tau_k}(x,0) = 0, \quad (s^{\tau_k})_t(x,0) = 0.
\end{aligned} \tag{28}
$$

In the above equation, the sensitivity of stress with respect to τ_k (i.e., σ_{τ_k}) is given by

$$
\sigma_{\tau_k} \;=\; E_1(s^{\tau_k})_{xt} + E_0\left((s^{\tau_k})_x - \sum_{j=1}^{N_p} \epsilon^j_{\tau_k}\right). \tag{29}
$$

Here the sensitivity of internal variables ϵ^j with respect to τ_k (i.e., $\epsilon^j_{\tau_k} = \dfrac{\partial \epsilon^j}{\partial \tau_k}$) is given as follows:

- if $j = k$, then we have

$$
\begin{aligned}
&\tau_k(\epsilon^k_{\tau_k})_t + \epsilon^k_t + \epsilon^k_{\tau_k} = p_k(s^{\tau_k})_x, \\
&\epsilon^k_{\tau_k}(0) = 0.
\end{aligned} \tag{30}
$$

- if $j \neq k$, then we have

$$
\begin{aligned}
&\tau_j(\epsilon^j_{\tau_k})_t + \epsilon^j_{\tau_k} = p_j(s^{\tau_k})_x, \\
&\epsilon^j_{\tau_k}(0) = 0.
\end{aligned} \tag{31}
$$

4. Sensitivity PDE for $s^{p_k} = \dfrac{\partial u}{\partial p_k}$, $k = 1, \ldots, N_p$:

$$
\begin{aligned}
&\rho(s^{p_k})_{tt} - (\sigma_{p_k})_x = 0 \\
&s^{p_k}(0,t) = 0, \quad \sigma_{p_k}(L,t) = 0, \\
&s^{p_k}(x,0) = 0, \quad (s^{p_k})_t(x,0) = 0.
\end{aligned} \tag{32}
$$

In the above equation, the sensitivity of stress with respect to p_k (i.e., σ_{p_k}) is given by

$$
\sigma_{p_k} = E_1(s^{p_k})_{xt} + E_0\left((s^{p_k})_x - \sum_{j=1}^{N_p} \epsilon^j_{p_k}\right). \tag{33}
$$

Here the sensitivity of internal variables ϵ^j with respect to p_k (i.e., $\epsilon^j_{p_k}$) is given as follows:

- if $j = k$, then we have

$$
\begin{aligned}
&\tau_k(\epsilon^k_{p_k})_t + \epsilon^k_{p_k} \;=\; p_k(s^{p_k})_x + u_x, \\
&\epsilon^k_{p_k}(0) = 0.
\end{aligned} \tag{34}
$$

- if $j \neq k$, then we have

$$
\begin{aligned}
&\tau_j(\epsilon^j_{p_k})_t + \epsilon^j_{p_k} = p_j(s^{p_k})_x, \\
&\epsilon^j_{p_k}(0) = 0.
\end{aligned} \tag{35}
$$

5. Sensitivity PDE for $s^A = \dfrac{\partial u}{\partial A}$:

   To some extent, this is an "outlier" as compared with the preceding sensitivity PDEs. The coefficient A only appears in the (right) boundary condition and not explicitly in the PDE itself; thus, the sensitivity PDE for A will be nearly the same as the original system PDE except with a different (right) boundary condition. The sensitivity equation for s^A is given by

$$\begin{aligned} &\rho(s^A)_{tt} - (\sigma_A)_x = 0, \\ &s^A(0,t) = 0, \quad \sigma_A(L,t) = g_A(t), \\ &s^A(x,0) = 0, \quad (s^A)_t(x,0) = 0. \end{aligned} \tag{36}$$

   In the above equation, g_A is given by

$$g_A(t) = \begin{cases} \exp\left(\dfrac{|ab|}{t(t+a-b)}\right) & \text{if } t \in (0, b-a) \\ 0 & \text{otherwise} \end{cases}.$$

   The sensitivity of stress with respect to A (i.e., σ_A) is given by

$$\sigma_A = E_1(s^A)_{xt} + E_0\left((s^A)_x - \sum_{j=1}^{N_p} \epsilon_A^j\right). \tag{37}$$

   Here the sensitivity of internal variables ϵ^j with respect to A (i.e., $\epsilon_A^j = \dfrac{\partial \epsilon^j}{\partial A}$) is given by

$$\begin{aligned} &\tau_j(\epsilon_A^j)_t + \epsilon_A^j = p_j(s^A)_x, \\ &\epsilon_A^j(0) = 0, \end{aligned} \tag{38}$$

   where $j = 1, \ldots, N_p$.

Based on the above discussions, we see that all the sensitivity equations are coupled to the original system except the one for A. However, it is important to note that the sensitivity PDEs are *not* coupled to each other.

e-mail:
{htbanks, shu3, zrkenz}@ncsu.edu
{carola.kruse, simon.shaw, john.whiteman}@brunel.ac.uk
{s.e.greenwald, m.p.brewin,m.j.birch}@qmul.ac.uk