Optimizing the Sheen Contour for the Inversion of the Laplace Transform

Parol, Axel P.A.

Award date:
2023

Technische Universiteit
Eindhoven
University of Technology

Department of Mechanical Engineering
Mechanics of Materials Research Group

Optimizing the Sheen Contour for the Inversion of the Laplace Transform

Bachelor Thesis

Axel Parol, 1438115

Supervisors:
J.J.C. Remmers
Nelson Sanchez Martinez

Eindhoven, Februari 2023

Abstract

The inverse Laplace transform of some functions can be performed using a hyperbola as an integration contour. An example of this is the Sheen Contour. It is important to set the five parameters of this contour to a proper value to get an accurate inverse Laplace transform. In this thesis, a fitting algorithm is explored while fixing three of the five parameters to see if this would result in an accurate inverse Laplace transform. The specific fixed parameters are chosen by use of a sensitivity analysis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The computational time for the calculation of time-dependent problems can take very long. This is not favorable when working with a digital twin hooked up to a 3D printer since the digital twin would take longer to compute than the experiment is taking for the heat conduction problem. This computational time can be reduced significantly by making use of the inverse Laplace Transform method to take the equations out of the time domain. This gives the possibility to calculate the heat conduction problem while the experiment is performed. It is important to choose the correct method for the inverse Laplace transform so the inverse Laplace transform is performed in the correct way and with as little error as possible. With the method chosen, the parameters have to be chosen carefully. Currently, the problem is that an inverse Laplace transform method used for the heat conduction problem has parameters that have to be tuned according to the specific problem. The goal is to find a method that can tune these parameters in order for a successful inverse Laplace transform.
In this thesis, the parameters of the inverse Laplace transform contour proposed by Sheen in [13] will be studied. This is also called the Sheen Contour. This will be done using a Sobol method, which is a method to analyze the sensitivity of the parameters. The parameters are then optimized by fitting the inversed Laplace transform to simulated experimental data with inspiration from [16] in order to assess if the fitting can be done on already existing experimental data.

Chapter 2

Laplace transform

The Laplace transform is a tool to reduce the computational time significantly for difficult problems. It converts a difficult problem into a simpler one. By applying the Laplace transform, differential operators in time and space can be reduced to algebraic operators in the frequency variable [3]. This allows the solution to be expressed in terms of the original independent variables. However, an increase in the entropy of a system can not be taken into account with the inverse Laplace transform. It is, therefore, important to only apply the inverse Laplace transform to a function f with certain conditions. These conditions will be explained underneath.

The Laplace transform of a real function $f : \mathbb{R} \to \mathbb{R}$ with $f(t) = 0$ for $t < 0$ and its inversion formula are defined as [10]

$$F(z) = L[f(t)] = \int_0^\infty e^{-zt} f(t) dt \tag{2.1}$$

$$f(t) = L^{-1}[F(z)] = \frac{1}{2\pi i} \int_{v-i\infty}^{v+i\infty} e^{zt} F(z) dz \tag{2.2}$$

In this formula $z = v + iw$; $v, w \in \mathbb{R}$.
$v \in \mathbb{R}$ is arbitrary but greater than the real parts of all the singularities of $F(s)$. The integrals in Equation 2.1 and Equation 2.2 exist for $\mathrm{Re}(z) > a \in \mathbb{R}$ if

1. f is locally integrable
2. there exist a $t_0 \geq 0$ and $k, a \in \mathbb{R}$, such that $|f(t)| \leq ke^{at}$ for all $t \geq t_0$
3. for all $t \in (0, \infty)$ there is a neighbourhood in which f is of bounded variation.

In this paper, it is assumed that all functions fulfill the above conditions and that there are no singularities of $F(z)$ to the right of the origin. Especially the second condition is important. This condition namely means that the function must decay over time. If this is not the case, no inversion of the Laplace transform can be applied successfully.
Besides this method, also called the Bromwich contour, there are multiple ways of inverting the Laplace transform [15]. Previous methods involved expansion of the inverse in series of Laugerre functions, as for example Widder, and Tricomi did. A more known method for this is Weeks Method. Salzer in 1955 evaluated the inversion integral by Gaussian quadrature using an appropriate system of orthogonal polynomials. Most of the methods can be divided into two groups, namely orthogonal series expansions, or weighted sums of values of the transform at a set of points, usually complex points. The method described in this paper is a continuation of the work of Talbot. Talbot made the number of points n_z as one of several arbitrary parameters. This made little to none preliminary computational work to be required. Talbot replaced the Bromwich contour with an equivalent contour L for which L enclosed all singularities of $F(z)$. The advantage is that this contour can handle noise in data [4]. This contour is the basis of the Sheen Contour, which will be explained in the next chapter.

Chapter 3

Sheen Contour

In this chapter, the inverse Laplace transform by the Sheen Contour is explained according to [13]. The inverse Laplace transform by the Sheen Contour can, just as the Talbot method, handle noise in data. First, the formulas are explained that will be used to perform the Sheen Contour. Next, The effect of every parameter is shown in order to get a clear picture of what every parameter does. Lastly, it is explained what the goal of selecting the right parameters is in order to get a stable inverse Laplace transform.

3.1 The method of Sheen

As said in the chapter before, the inverse Laplace transform will be performed using the Sheen Contour.

The first step in the approach of the Sheen Contour is to represent the inverse Laplace transform solution $f(t)$ as a contour integral of the form

$$f(t) = \frac{1}{2\pi i} \int_{\Gamma} e^{zt} F(z) dz \tag{3.1}$$

where, for $\mathrm{Re}(z) \geq \sigma$ with σ sufficiently large, $F(z)$ is the Laplace transform of f

$$F(z) = \hat{f}(z) = \int_0^{\infty} e^{-zt} f(t) dt \tag{3.2}$$

In these equations, z is a complex number in the form $z = v + iw$, and t is time. Γ is the deformed contour, which behaves asymptotically as a pair of straight lines in the left half-plane. This means that it can be seen as two lines that go to $\mathrm{Re}(z) \to -\infty$ with an opposite slope when $\mathrm{Im}(z) \to \pm\infty$, forcing the factor e^{zt} to decay exponentially towards both ends of the deformed contour. Γ is permitted to cut the real axis either to the right of the origin or to the left, but sufficiently close to the origin. The formula for the contour Γ is then

$$\Gamma = \{z : z = \varphi(y) + isy, y \in \mathbb{R},\ y \text{ increasing}\} \tag{3.3}$$

In this equation, s is a positive parameter, and $\varphi : \mathbb{R} \to \mathbb{R}$ is a fixed smooth function satisfying

$$\varphi(y) \approx -|y| \quad \text{for large } |y|, \quad \text{and } \varphi(y) \leq \alpha - |y| \quad \text{for } y \in \mathbb{R} \tag{3.4}$$

An example of this contour can be seen in Figure 3.1.

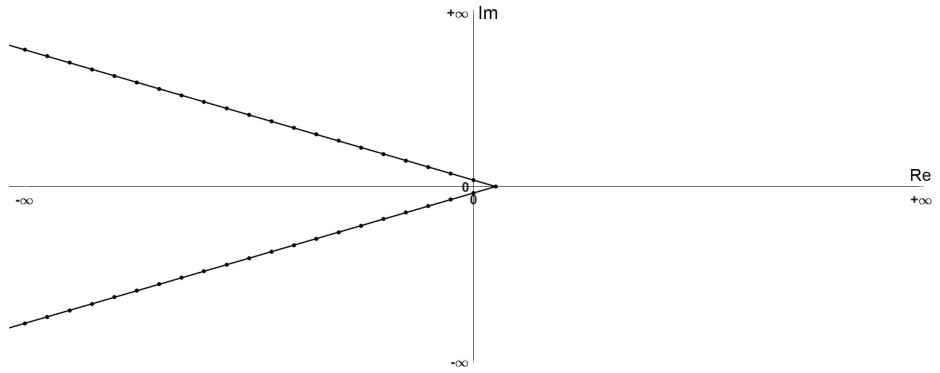

Figure 3.1: Straight line Sheen Contour

In this contour, the two asymptotes of Γ have slope $\pm s$, and from the point of view of enhancing the exponential decay in Equation 3.1 s should be taken as small as possible. Quadrature will now be applied to the contour integral because it was suggested by M. Crouzeix this has benefits over the previous treatment. A choice for the function φ which is used in the calculation is then

$$\varphi(y) = \alpha - \sqrt{y^2 + \beta^2}, \quad \text{for } y \in \mathbb{R} \tag{3.5}$$

In this equation, the two parameters are α and β, with $\alpha \in \mathbb{R}$ and $\beta > 0$. The curve Γ is then the left-hand branch of a hyperbola, with asymptotes having slopes $\pm s$, which crosses the real axis at $\alpha - \beta$. An example of this contour can be seen in Figure 3.2.

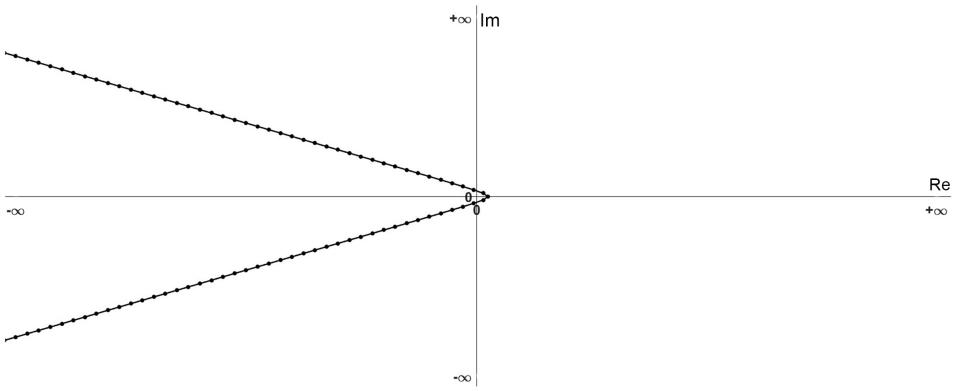

Figure 3.2: Hyperbolic Sheen Contour

It is important that the Laplace transform $\hat{f}(z)$ of a function $f(t)$ must have an analytical continuation, from Γ_0 to the deformed contour Γ. Another requirement is that all singularities of $\hat{f}(z)$ should lie to the left of Γ. With the quadratic contour represented in Equation 3.3, the integral Equation 3.1 can be written as an infinite integral with respect to a real variable:

$$f(t) = \int_{-\infty}^{\infty} v(t,y)dy, \quad \text{with } v(t,y) = \frac{1}{2\pi i} e^{z(y)t} F(z(y)) z'(y), \quad z(y) := \varphi(y) + isy \tag{3.6}$$

In this equation, $F(z)$ is the Laplace transform of $f(t)$. The integrand in this integral decays exponentially for large t, because of the assumed behavior of $\mathrm{Re}(z(y)) = \varphi(y)$. The approximate solution will then be of the form:

$$\tilde{F}_N(t) = \sum_{j=-n_z+1}^{n_z-1} \omega_j v(t, y_j) = \sum_{j=-n_z+1}^{n_z-1} \hat{\omega}_j e^{z_j t} F(z_j), \quad z_j = z(y_j), \quad \hat{\omega}_j = \frac{1}{2\pi i} z'(y_j)\omega_j \tag{3.7}$$

In this equation, y_j are the quadrature points with $y_j \in \mathbb{R}$ and the nonnegative weights ω_j depending on the particular quadrature scheme. The parameter n_z is the number of quadrature points. The particular quadrature scheme that now will be considered is obtained by mapping the infinite integral of Equation 3.6 to (-1,1) and then applying the trapezoidal rule to the resulting finite integral. Under appropriate conditions, the resulting quadrature formula has a high order of accuracy. This is done by first changing the variables $y = y(\eta)$ where $y(\eta)$ is a smooth increasing function mapping (-1,1) to $\mathbb{R}$, to obtain, for $v \in C(\mathbb{R}; \mathbb{B})$ with $\mathbb{B}$ the Banach space. This results in

$$I(v) := \int_{-\infty}^{\infty} v(y)dy = \int_{-\infty}^{\infty} V(\eta)d\eta, \quad \text{where } V(\eta) = v(y(\eta))y'(\eta) \tag{3.8}$$

Specifically, with τ a positive parameter, $y(\eta)$ is chosen to be the odd function

$$y(\eta) = \tau^{-1}\chi(\eta), \quad \text{where } \chi(\eta) = \log((1+\eta)/(1-\eta)) \tag{3.9}$$

In the above equation, τ will play the role of a threshold in t, in that the approximate solution will be accurate of order essentially t/τ for $t > \tau$. An example of this quadrature contour can be seen in Figure 3.3.

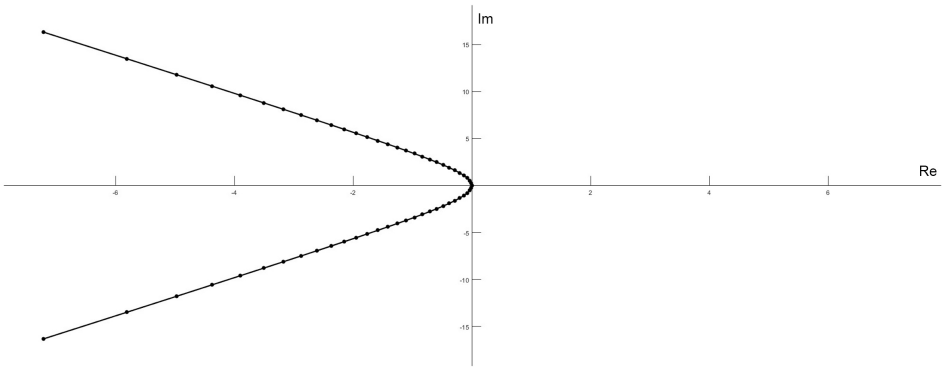

Figure 3.3: Sheen Contour after implementing τ

The difference between the previous contour, as displayed in Figure 3.2, and this contour is the spacing of the quadrature points. In Figure 3.2, the quadrature points are all evenly spaced apart, while in Figure 3.3 the quadrature points are more placed near the top of the hyperbola. This is for the reason that it is found that these points are the most important for an accurate Laplace transform [16]. In this paper, it was shown that no accuracy is lost by including only the middle 75% of the nodes and discarding the outlying 25%.
The composite trapezoidal rule can now be applied with spacing $1/n_z$ to the integral over (-1,1) and assuming $V(\pm 1) = 0$, the following equation is defined

$$Q_{N,\tau}(v) = \frac{1}{n_z} \sum_{j=-n_z+1}^{n_z-1} V(\eta_j) = \frac{1}{n_z\tau} \sum_{j=-n_z+1}^{n_z-1} \mu_j v(y_j)$$

where

$$\eta_j = j/n_z,\ \mu_j = \chi'(\eta_j) = 2/(1-\eta_j^2),\ y_j = y(\eta_j) = \tau^{-1}\chi(\eta_j) \tag{3.10}$$

The quadrature points are thus distributed over an interval with endpoints $\pm\tau^{-1}\log(2n_z-1)$. Provided $v(y)$ in Equation 3.8 vanishes appropriately fast at infinity, this formula is then of an arbitrarily high order of accuracy.
The quadrature scheme can now be applied as an approximation of the exact solution $f(t) = I(v(t,\cdot))$, where $v(t,y)$ is given in terms of w by Equation 3.6. Explicitly, the approximation of $f(t)$ is given by

$$\tilde{F}_{n_z,\tau}(t) = Q_{N,\tau}(v(t,\cdot)) = \frac{1}{n_z\tau}\sum_{j=-n_z+1}^{n_z-1}\tilde{\mu}_j e^{z_j t}F(z_j), \quad \tilde{\mu}_j = \frac{1}{2\pi i}z'(y_j)\mu_j \tag{3.11}$$

where $z_j = z(y_j) = \varphi(y_j) + isy_j$, $z'(y_j) = \frac{-y_j}{\sqrt{y_j^2+\beta^2}} + is$, and where $\tau > 0$ is the parameter in the transformation in Equation 3.9. In cases when $F(\overline{z}) = \overline{F(z)}$ (what will be the case in the used functions), and the function φ is even, it is given that $\tilde{\mu}_{-j} = \overline{\tilde{\mu}_j}$ and $z_{-j} = \overline{z}_j$, for which Equation 3.11 can be rewritten as

$$U_{n_z,\tau}(t) = 2\mathrm{Re}\left(\frac{1}{n_z\tau}{\sum_{j=0}^{n_z-1}}'\tilde{\mu}_j e^{z_j t}F(z_j)\right) \tag{3.12}$$

where the prime after the summation indicates that the term with $j = 0$ is to be halved.

3.2 Effect of the parameters

After giving every equation to perform the inverse Laplace transform by the Sheen Contour, the effect of each of the five parameters α, β, n_z, s, and τ on the Sheen Contour will be shown. This will give a clearer insight into how the Sheen Contour, as seen in Figure 3.3, reacts to parameter changes.

3.2.1 Effect of α

The parameter α is part of $z_j = z(y_j) = \alpha - \sqrt{y_j^2+\beta^2} + isy_j$. The effect of changing α can be seen in Figure 3.4.

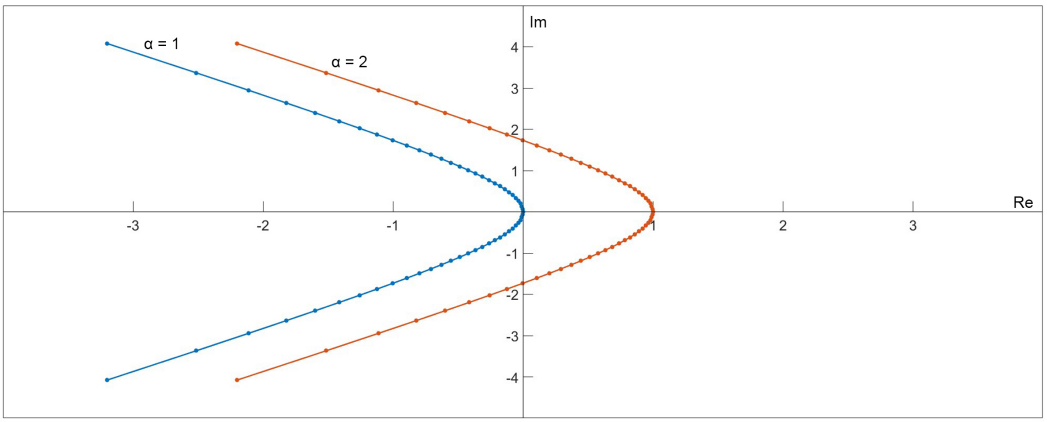

Figure 3.4: Sheen Contour for $\alpha = 1$ and $\alpha = 2$

In the above figure, it clearly can be seen that an increase in α will result in a shift to the right of the Sheen Contour. It was already explained above that the Sheen Contour crosses the real axis at $\alpha - \beta$. In the case of the above figure $\beta = 1$, which means for $\alpha = 1$ the real axis is crossed at 0, and for $\alpha = 2$ the real axis is crossed at 1.

3.2.2 Effect of β

The parameter β is also part of $z_j = z(y_j) = \alpha - \sqrt{y_j^2 + \beta^2} + isy_j$. The effect of changing β can be seen in Figure 3.5.

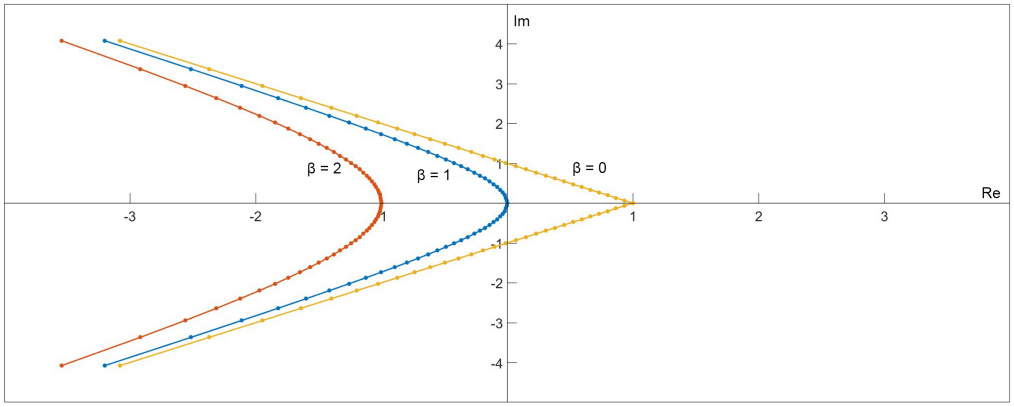

Figure 3.5: Sheen Contour for $\beta = 0$, $\beta = 1$, and $\beta = 2$

In the above figure, it can be seen that an increase in β will result in a shift to the left of where the Sheen Contour crosses the real axis. However, only the integration points that are close to the real axis are shifted drastically, while the integration points further from the real axis only show a small shift. A special case arises when $\beta = 0$. The integration contour is then the same as the straight line Sheen Contour in Figure 3.1.

3.2.3 Effect of n_z

The parameter n_z is the number of integration points. n_z is part of $\eta_j = j/n_z$ which comes back in two parts of the Sheen Contour. The effect of changing n_z can be seen in Figure 3.6.

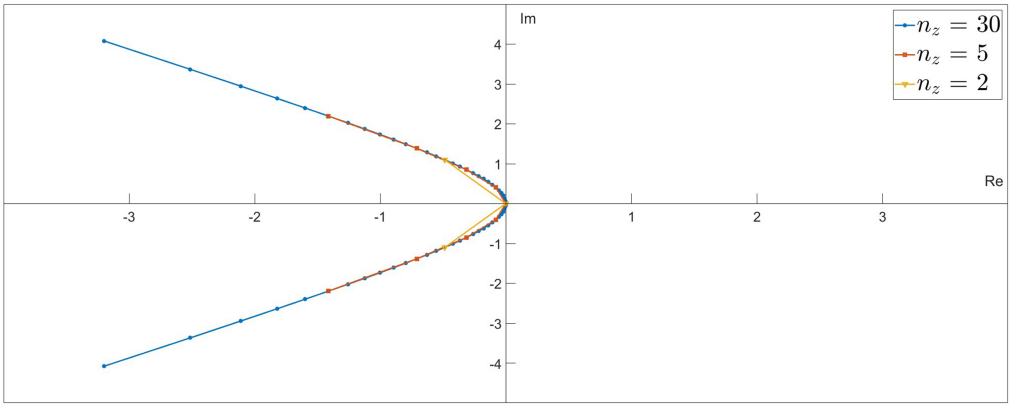

Figure 3.6: Sheen Contour for $n_z = 2$, $n_z = 5$, and $n_z = 30$

In the above figure, it can be seen that the Sheen Contour will not change in shape for a change in n_z. The value at which the contour crosses the real axis always stays the same. However, what does change is of course the number of integration points itself. Another thing that changes with this is how far the Sheen Contour will go on the negative real axis. If n_z becomes smaller, the Sheen Contour will reach less far on the negative real axis. This is for the reason that a lower n_z will result in a lower value for η_j in Equation 3.9 and Equation 3.10 which results in a lower real part.

3.2.4 Effect of s

The parameter s is again part of $z_j = z(y_j) = \alpha - \sqrt{y_j^2 + \beta^2} + isy_j$. The effect of changing s can be seen in Figure 3.7.

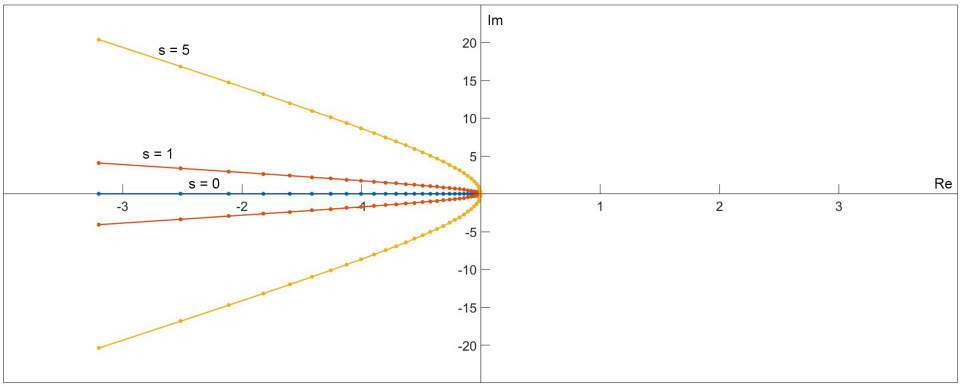

Figure 3.7: Sheen Contour for $s = 0$, $s = 1$, and $s = 5$

In the above figure, it clearly can be seen that an increase in s will result that the Sheen Contour will go further on the imaginary axis. For $s = 0$, there is no imaginary part of the Sheen Contour and for a higher s, the maximum value of the imaginary part of the Sheen Contour will increase. The maximum value for the imaginary part scales linearly with s and $y_j = \tau^{-1}\chi(\eta_j)$. This means that in this case it only scales linearly with s because y_j is kept constant.

3.2.5 Effect of τ

The parameter τ is again part of $z_j = z(y_j) = \alpha - \sqrt{y_j^2 + \beta^2} + isy_j$. The parameter τ comes back in y_j in $\sqrt{y_j^2 + \beta^2}$ and in the imaginary part isy_j. The effect of changing τ can be seen in Figure 3.8.

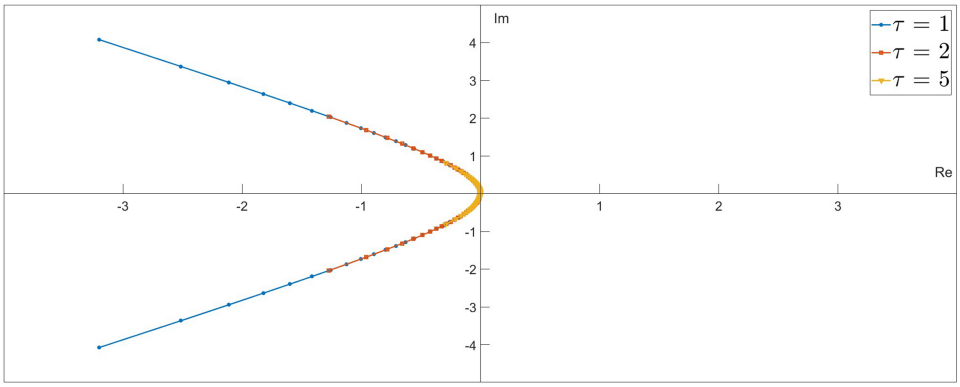

Figure 3.8: Sheen Contour for $\tau = 1$, $\tau = 2$, and $\tau = 5$

In the above figure, it can be seen that increasing τ will change the distribution of the integration points to the top of the hyperbola. An increase in τ will also decrease how far the Sheen Contour will go on the negative real axis. This is the opposite for n_z since a decrease in n_z will have a similar result as increasing τ of how far the Sheen Contour will go on the negative real axis. The reason for this trend is Equation 3.10 since a higher τ will mean a lower y_j.

3.3 Goal of the Sheen Contour

As said before, all singularities of the Laplace transform $F(z)$ of a function $f(t)$ should lie to the left of the integration contour Γ. The goal for a good inverse Laplace transform is to fit the contour as close to these singularities as possible. This is done by tuning the five parameters α, β, n_z, s, and τ. An example of a Sheen Contour that follows this rule on the function $f(t) = e^{-t^2}$ with $F(z) = \frac{\sqrt{\pi}}{2} e^{z^2/4}(1 - erf(\frac{z}{2}))$ can be seen in Figure 3.9 together with the result of this Sheen Contour in Figure 3.10.

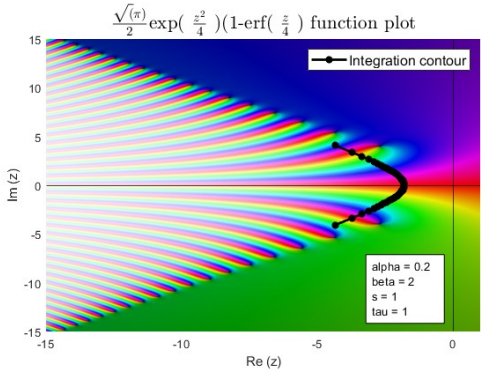

Figure 3.9: Sheen Contour for $F(z) = \frac{\sqrt{\pi}}{2} e^{z^2/4}(1 - erf(\frac{z}{2}))$ and $n_z = 30$

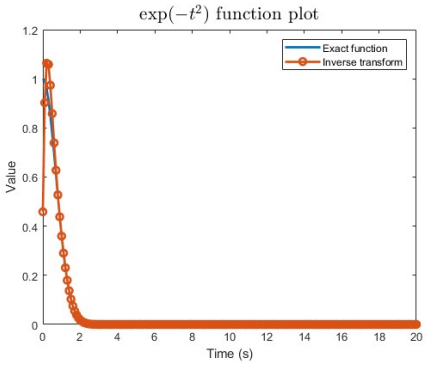

Figure 3.10: Result of this Sheen Contour

A plot like Figure 3.9 is known as domain coloring [6]. Each color is given to a point of the complex plane based on the argument. The brightness is based on the modulus. In this paper, particular attention is paid to the white color. This indicates the singularities. The black color indicates a pole. As can be seen in the figure, all singularities (white color) are to the left of the Sheen integration Contour, which makes it a proper Contour. This Sheen Contour would result in an absolute mean error of 0.006 in the first 20 seconds compared to the real function $f(t) = e^{-t^2}$. It even is stable after 1000 seconds, which makes it a good integration Contour.

An example of a bad Sheen Contour can be seen in Figure 3.11 together with the result in Figure 3.12. The singularities of the function are lying to the right of the Sheen Contour. This results in a very unstable inversed Laplace transform in which the absolute mean error equal to $1.8{\cdot}10^{17}$ is for the first 20 seconds compared to the real function $f(t) = e^{-t^2}$.

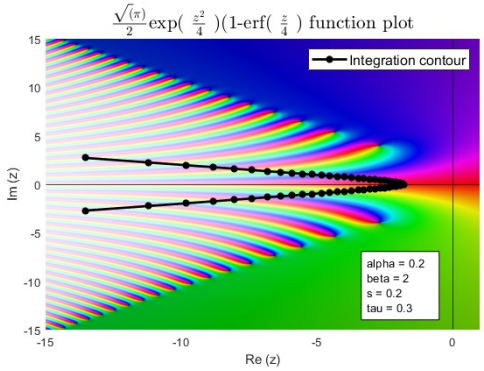

Figure 3.11: Bad Sheen Contour for $F(z) = \frac{\sqrt{\pi}}{2} e^{z^2/4}(1 - erf(\frac{z}{2}))$ and $n_z = 30$

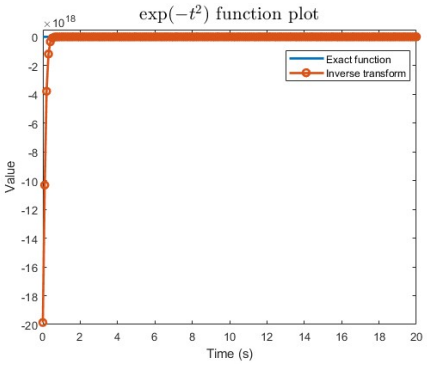

Figure 3.12: Result of this bad Sheen Contour

Chapter 4

Sensitivity analysis

In this chapter, the sensitivity analysis of the parameters α, β, n_z, s, and τ is explored. This is done using a variance-based sensitivity analysis named Sobol method. First, the Sobol method is explained together with the parameter spaces. Next, the sensitivity analysis is done on the set parameters and a conclusion is drawn.

4.1 Different types of sensitivity analysis

Before starting with a sensitivity analysis of any problem set, there needs to be determined which sensitivity analysis is going to be used. Each problem set faces different challenges. In the case of the inverse Laplace transform, there is known that it is nonlinear from Equation 3.9. Then there is also known that there are interactions between parameters according to Equation 3.10. It is, therefore, preferable to have a sensitivity analysis that captures these interactions. Now we set the above constraints, four different methods will be discussed in order to assess if they are suitable for the inverse Laplace transform [7]. It is important to notice that these are not all sensitivity analysis methods. Some methods have not been taken into consideration at all for the reason that they were directly seen as not suitable.

4.1.1 Nominal Range Sensitivity

The nominal range sensitivity method, also called the one-at-a-time method, is the most straightforward approach to performing a sensitivity analysis. This method is also known as a local sensitivity analysis. In this method, all the parameters are fixed at one specific value except for one. This is also where the term 'local' comes from. This parameter will be changed in a certain set range and the effect on the output will be analyzed. This method can be used for an unlimited number of parameters. There are multiple ways to analyze the output. One way is to calculate the sensitivity by dividing the percentage change in output by the percentage change in input for each of the parameters. Another way is to do linear regression to analyze the output. A downside of the nominal range sensitivity method is that it is unsuitable for nonlinear models. Besides, with five input parameters that have a volume fraction of $\frac{1}{5!}$, it means approximately only 0.8% of the total parameter space is explored.

4.1.2 Regression analysis

Regression analysis can be approached as a probabilistic sensitivity analysis technique (Iman *et al*, 1985 in [7]). Before doing this sensitivity analysis, a relation between inputs and output should be determined. For this, understanding the functional form of the model or scatter plots is needed. Generally, a relationship between inputs and an output is fitted. This relationship is assumed, which means that if the relationship is not met it lacks robustness. Another limitation is that

regression analysis will not be optimal for input parameters with interactions (Devore and Peck, 1996 in [7]).

4.1.3 Scatter plots

The method of using scatter plots is a graphical sensitivity analysis. The method is to get scatter plots of the output against the individual input variables. The correlation can then be measured between the input and output from which a conclusion can be drawn about which parameter has the most impact. As said in the subsection above, scatter plots must be used to do the regression analysis. Scatter plots are, therefore, considered to often be the first step in the sensitivity analysis. There is also a personal interpretation of scatter plots, which can result in different conclusions retrieved from them from one person to another.

4.1.4 Variance-based analysis (Sobol method)

In the variance-based sensitivity analysis, the variance of the output is divided into fractions that are attributed to individual inputs or sets of inputs. These fractions can be directly interpreted. One way to perform the variance-based analysis is the Sobol method. This method will be explained later. One advantage of variance-based analysis is that no assumption is needed regarding the underlying model, only the parameter spaces are needed. Another advantage of this method is that it is useable for nonlinear systems and it deals with interactions. However, compared to other sensitivity analyses, it can be computationally heavy for a large set of parameters.

4.2 Selecting sensitivity analysis

As stated before, the sensitivity analysis used needs to deal with the nonlinearity in the system. The inverse Laplace transform is highly non-linear due to the spacing function for the quadrature points. In Equation 3.9, it can clearly be seen from the logarithms. Therefore, the sensitivity analysis must deal with very large nonlinearities. For this reason, the nominal range sensitivity is not suitable. Another reason is that just 0.8% of the parameter space is explored with this method, as calculated in the subsection. This means that a large part keeps being undiscovered. The second condition for the sensitivity analysis of the inverse Laplace transform is that the sensitivity analysis must capture interactions between parameters. This means that just using scatter plots is not applicable to this system since these interactions will not be captured. Due to this, it will also be difficult to perform a regression analysis on the system. This is for the reason that understanding the functional form of the model or scatter plots are needed for the regression analysis. The functional form is too hard to obtain for the inverse Laplace transform in the scope of this project. Besides this, the regression analysis is not optimal for input parameters with correlations. From this, it is concluded that regression analysis is not suitable as well. The last method considered was the variance-based analysis. This method does deal with nonlinearity in the system and it can handle interactions between the parameters [12]. The interactions between parameters can even be retrieved from the calculated fractions. Therefore, this seems the best sensitivity analysis method to be used on the inverse Laplace transform. The only downside is that it can be computationally heavy for a large set of parameters. However, there are only dealt with five parameters, which are not considered a 'large' number.

4.3 Sobol method

As concluded above, the used sensitivity analysis will be a variance-based analysis. This is a global sensitivity analysis, i.e. the entire parameter space is explored. This method is often to referred as the Sobol method since this is the most used method of variance-based analysis. Another method is the extended Fourier Amplitude Sensitivity Test method but this method was not considered. In the Sobol method, two indices are calculated for every parameter [2]. The calculated indices are

used for estimating the influence of variables on the model output. The indices are then ranked to find which input parameter has the most influence on the output. These indices are computed by Monte Carlo simulations [14]. Monte Carlo simulation uses random sampling in order to solve a problem that might be deterministic. This should be the case for the Sobol indices if the parameter spaces are kept constant. A schematic for global sensitivity is given in Figure 4.1.

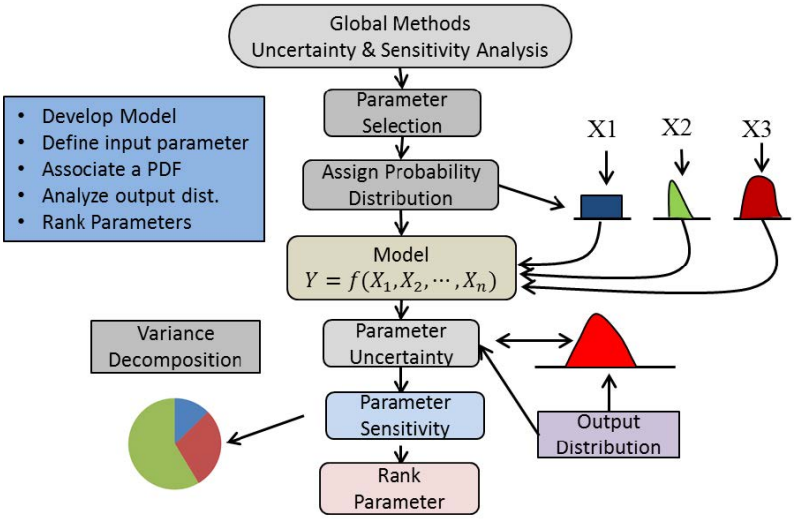

Figure 4.1: Schematic for global sensitivity analysis

As seen in the figure above, global sensitivity analysis consists of five main steps in order to perform the sensitivity analysis. These five steps for the Sobol method are as follows:

1. The model is known and developed and the individual parameters that want to be investigated are selected. The parameter spaces are defined with a probability distribution.

2. From this parameter space together with the probability distribution, a set of random input vectors is generated.

3. The set of random input vectors is inserted in the model and each solution is calculated.

4. The output distribution is analyzed and determined.

5. Variance decomposition according to the Sobol method to rank the input parameters followed from the Sobol indices.

The first four steps listed above are the same for every global sensitivity analysis method. Only the last step is different for the global sensitivity method chosen. As stated before, two Sobol indices will be calculated. These indices are the first order effects (direct effects) and total effects. In the end, especially the total effects are important since these include all interactions of that parameter. With the parameters α, β, n_z, s, and τ, the total effect of parameter α on the output is:

$$TS(\alpha) = S(\alpha) + S(\alpha\beta) + S(\alpha n_z) + S(\alpha s) + S(\alpha\tau) \tag{4.1}$$

In this formula, $TS(\alpha)$ is the total sensitivity index of α, $S(\alpha)$ is the first order sensitivity index for α, and $S(\alpha\beta)$ is the second-order sensitivity index between α and β, i.e. the interaction between α and β. This is also for the interactions between α and all other parameters. In order to calculate the first order indices and total indices, the analysis of variances has to be computed. Assume that the model is described by a function $u = f(x)$, where the input $x = (x_1, ..., x_n)$ is a point

inside an n-dimensional box and u is a scalar output. The function $f(x)$ can be represented into summands of different dimensions by decomposition of the function where $1 \leq i_1 < ... < i_s \leq n$:

$$f(x) = f_0 + \sum_{i=1} f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j) + ... + f_{1,2...n}(x_1, x_2, .., x_n) \tag{4.2}$$

Equation 4.2 must satisfy

$$\int_0^1 f_{i_1...i_s}(x_{i_1}, ..., x_{i_s})dx_k = 0 \quad \text{for} \quad k = i_1, ..., i_s \tag{4.3}$$

It follows from Equation 4.3 that all members in Equation 4.2 are orthogonal and can be expressed as integrals of $f(x)$:

$$f_0 = \int f(x)dx \tag{4.4}$$

$$f_i(x_i) = -f_0 + \int f(x)dx \prod_{k \neq i} dx_k \tag{4.5}$$

$$f_{ij}(x_i, x_j) = -f_0 - f_i(x_i) - f_j(x_j) + \int f(x)dx \prod_{k \neq i,j} dx_k \tag{4.6}$$

Similar formulae can be obtained for higher-order terms. Assuming that f(x) is square integrable means that all terms in Equation 4.2 are also square integrable. Following from this the total variance D of $f(x)$ is calculated:

$$D = \int f^2 dx - f_0^2 \tag{4.7}$$

The partial variances for each of the terms in Equation 4.1 are then:

$$D_{i_1,...,i_s} = \int f^2_{i_1,...,i_s} dx_{i_1} \cdots dx_{i_s} \tag{4.8}$$

In the above equation, $1 \leq i_1 < ... < i_s \leq n$ and $s = 1, ..., n$. From this, the global sensitivity indices are calculated

$$S_{i_1,...,i_s} = \frac{D_{i_1,...,i_s}}{D} \quad \text{for} \quad i \leq i_1 < ... < i_s \leq k \tag{4.9}$$

In this equation, S_i is the first-order sensitivity index for factor x_i. This will measure the direct effects of x_i on the output. There is also the possibility to calculate second-order sensitivity indices, but this will not be looked at. The direct estimation of the global sensitivity indices is calculated using values of $f(x)$ only. This is done with the Monte Carlo algorithm. There are crude Monte Carlo estimates obtained for the first-order effects and total effects. These are as follows:

$$D = \int f^2 dx - f_0^2 \approx \frac{1}{N} \sum_{j=1}^{N} f^2(x_j) - f_0^2 \tag{4.10}$$

$$D_i = D - \frac{1}{2}[f(x) - f(x_i, x'_{-i}]^2 dx dx'_{-i} \approx D - \frac{1}{2N} \sum_{j=1}^{N} [f(x_j) - f(x_{ij} - x'_{-ij})]^2 \tag{4.11}$$

$$D_i^{tot} = \frac{1}{2} \int [f(x) - f(x_i, x'_{-i}]^2 dx dx'_{-i} \approx \frac{1}{2N} \sum_{j=1}^{N} [f(x_j) - f(x_{ij} - x'_{-ij})]^2 \tag{4.12}$$

In these equations, N is the sampling size for Monte Carlo discretization, and $x_{-1} = (x_1, ..., x_{i-1}, x_{i+1}, ..., x_m)$ is the parameter combination complementary to x_i.

Now the formulas are given and explained, the sensitivity analysis can be performed on the existing model.

4.4 Performing the Sobol method

The sensitivity analysis will be performed on the following function:

$$f(t) = e^{-t^2} \tag{4.13}$$

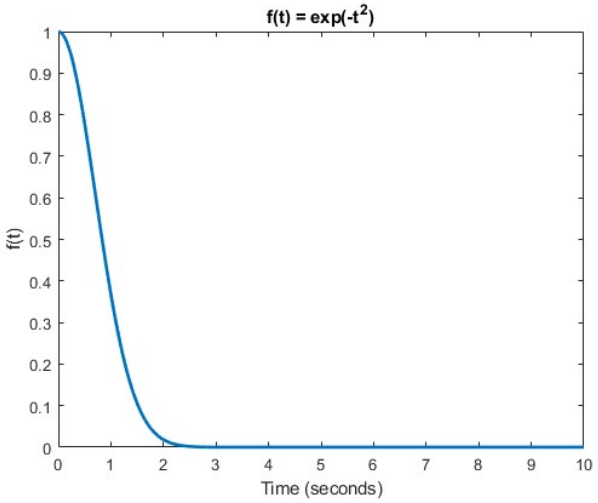

Figure 4.2: Function plot of $f(t) = e^{-t^2}$

The plot of this function can be seen in Figure 4.2 above. The Laplace transform $F(z)$ for this function is:

$$F(z) = \frac{1}{2}\sqrt{\pi}e^{z^2/4}\left(1 - erf(\frac{z}{2})\right) \tag{4.14}$$

As discussed in chapter 3, the sensitivity analysis of the above function will be done on the Sheen Contour Laplace transform.

4.4.1 Defining parameter spaces

The first thing to do when performing the Sobol method is to define the parameter spaces used. Every parameter space consists of a uniform distribution with a lower and upper bound. Only the variable n_z is set to be an integer always. The parameter space is defined in Table 4.1 below.

	Lower bound	Upper bound	Distribution
α	0.00	2.00	'Uniform'
β	0.00	2.00	'Uniform'
s	0.00	7.00	'Uniform'
n_z	2	50	'Uniform'
τ	0.5	5.00	'Uniform'

Table 4.1: Parameter spaces used for the sensitivity analysis

The upper bounds for the parameter spaces are estimated out of literature on the Sheen Contour. Now that the parameter space is defined, the number of model evaluations n must be determined.

4.4.2 Determining n

The number of model evaluations for the sensitivity analysis according to the Sobol method must be determined in order to get an accurate result. The number of evaluations is preferred to be as high as possible, but this would cost a lot of computational time. Therefore, the number of evaluations has to be chosen such that the result is accurate with as less computational time as possible. The formula used for the number of model evaluations (N) is as followed [9]

$$N = n \cdot (M + 2) \tag{4.15}$$

In this formula, n is the sample size and M is the number of uncertain inputs. This N is not the same N as in Equation 4.10, since in Equation 4.10, it is also divided by the number of parameters. For each sensitivity analysis, the number of uncertain inputs is known. Therefore, only factor n needs to be determined. The factor n depends on the complexity of the model and since we are dealing with a nonlinear model, the expected value is high. To determine the sample size n, first, a sensitivity analysis is run three times with $n = 100000$ and $M = 3$ at $t = 0$ s. The value for n was chosen this high to make sure convergence was reached and the value of M was chosen to have it assessed on three parameters. The summation of the Sobol indices for each parameter will be assessed with an accuracy of two decimals ranging in values from 0.00-1.00. For this reason, the maximum allowed difference is 0.01 which coincides with an error of 1%. The result for the first-order effects and total effects at $t = 0$ seconds can be seen in Table 4.2 for a large value of n ($n = 100000$).

	First-order effects indices	Total effects indices
Run 1	0.94	1.08
Run 2	0.93	1.09
Run 3	0.93	1.08

Table 4.2: First-order effects indices and total effects indices for $n = 100000$ at $t = 0$ s

The value of n will now be reduced until the solution does not satisfy the accuracy that is set above. This can be seen in Table 4.3.

	First order effects indices	Total effects indices
$n = 10000$	0.94	1.09
$n = 8000$	0.92	1.09
$n = 7000$	0.91	1.10
$n = 6000$	0.94	1.07

Table 4.3: First-order effects indices and total effects indices for different values of n at $t = 0$ s

As can be seen in the table above, the value of n will be set at 8000 in order to keep the results within an error of 1%. The error at $n = 7000$ is namely 2% compared to the converged first order effects.

4.5 Results

The results of the performed Sobol method on the four parameters α, β, n_z, and s can be seen in Figure 4.3 and Figure 4.4. The reason τ is not included in the Sobol method is that for certain combinations of s and τ, there would be no solution and, therefore, no Sobol index could be calculated. For this reason, it was decided to leave τ out of the sensitivity analysis. It was first tried to run the sensitivity analysis with τ included, but this did not work. Afterward, it was indeed found that removing τ from the sensitivity analysis resolved the problem. The reason only the Sobol indices at the first 5 seconds of the function $f(t) = e^{-t^2}$ are given, is that the value of $f(5) = 1.4 \cdot 10^{-11}$. This value is already so close to 0, that it is not interesting anymore to look at the influence of the parameters on the Laplace transform after this time. This convergence to 0 after 5 seconds can also be seen in Figure 4.2.

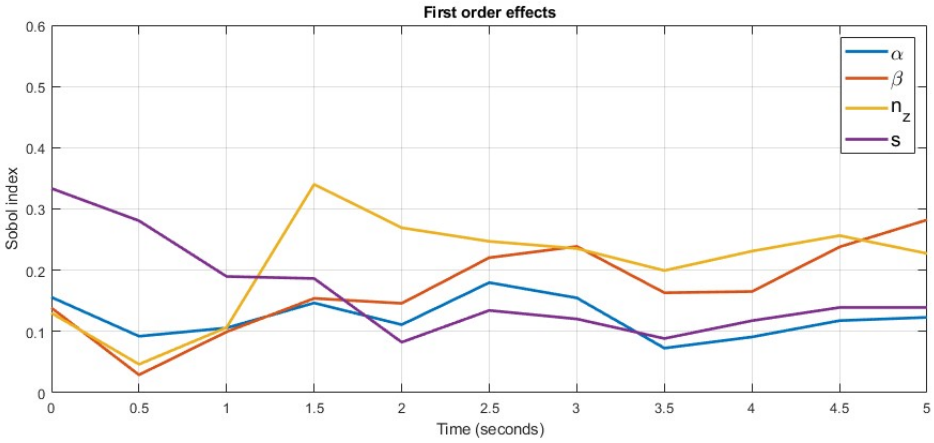

Figure 4.3: First order Sobol indices of the Laplace transform of the function $f(t) = e^{-t^2}$

As seen in Figure 4.3 above, the first order Sobol indices of the parameters reach a maximum value of 0.35. This means that there is no parameter that has a large direct effect. A large direct effect would namely mean a first order Sobol index that is close to 1. Furthermore, it can be seen that for the first second, s has an approximately twice as large first order Sobol index than α, β, and n_z. After the first second, the first order Sobol index of n_z increases three times in 0.5 seconds, while α, and β are only increasing by a factor of 1.5. It can also be seen that the first order Sobol index of s decreases for the first two seconds, after which it slowly increases a bit again. This increase is, however, very small. At t=5 seconds, β, and n_z have the largest first order Sobol indices, and α, and s the lowest. From the results of the first order Sobol indices, no real conclusions can be made. This is for the reason that, as stated before, the interactions between the parameters are not taken into account. It is, therefore, more interesting to look at the total Sobol indices in order to draw clear conclusions. This result can be seen in Figure 4.4.

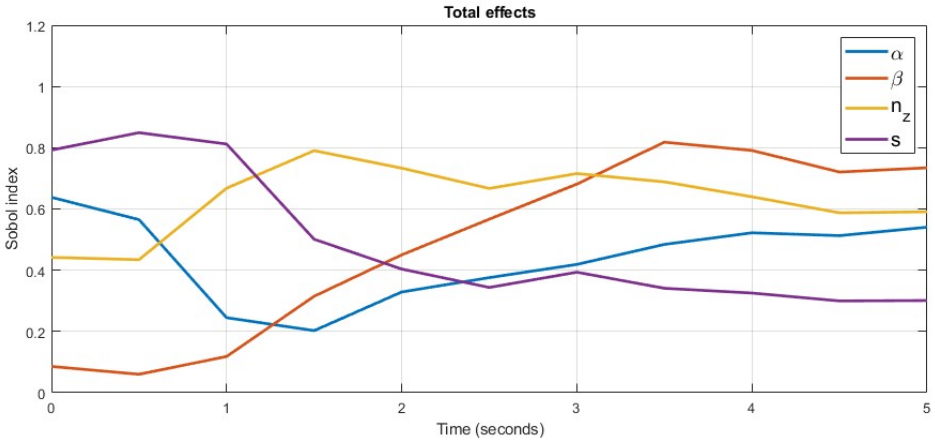

Figure 4.4: Total Sobol indices of the Laplace transform of the function $f(t) = e^{-t^2}$

As seen in the figure above, the total Sobol indices of the parameters reach a maximum value of 0.85. This value means a large total effect because the maximum total Sobol index is 1. The sum of all of the total Sobol indices is equal to or grater than 1 [8]. It can be seen in the figure above that s has the largest total effect for the first second. After the first second, the total Sobol index of s decreases until it reaches a value of 0.3 after five seconds. For the first 0.5 seconds, the total Sobol index of α is around 0.6, after which it drops down to 0.25 at t=1 second. The total Sobol index of α then slowly increases until it reaches a value of 0.54 at t=5 seconds. For parameter n_z, the total Sobol index is at 0.44 for the first 0.5 seconds, after which it reaches a maximum total Sobol index of 0.79 at t=1.5 seconds. It then slowly decreases till it reaches a value of 0.59 at t=5 seconds. The total Sobol index of parameter β is significantly lower than the rest for the first second. It then increases until it reaches a maximum total Sobol index of 0.82 at t=3.5 seconds. It then decreases slightly.

The main conclusion of this figure that can be made, is that for the first second, s has the largest influence on the variance of the output. After the first second, n_z will have the largest influence until the influence of β is almost equal at t=3 seconds. After 3 seconds, the influence of β is the largest. Since the parameter optimization will be conducted, it will be investigated fixing and leaving α, and s out of the optimization. Furthermore, it has been concluded from experience that τ behaved similarly to s. For this reason, τ will also be left out of the optimization and is fixed to an arbitrary value.

Chapter 5

Parameter optimization

In this chapter, the optimization of the parameters α, β, n_z, s, and τ is investigated. The sensitivity analysis will be taken into account to only fit two of the parameters for optimization. First, it is explained how the optimization will be performed and on which functions. After this, different options for the optimization of the parameters are explored, and in the end, a guideline will be presented to optimize the numerical inversion of the Laplace transform by the Sheen contour.

5.1 Optimization method

The parameters will be optimized in Matlab using the nonlinear least-squares solver on three different functions. With the nonlinear least-squares solver, simulated experimental data with noise will be compared to the already performed inverse Laplace transform by the Sheen contour. The nonlinear least-squares solver will try to find the best set of parameters that will fit the simulated experimental data the best. The first reason a nonlinear least-squares solver is chosen, is that the inverse Laplace transform by the Sheen Contour is nonlinear because of Equation 3.9. The second reason is that this type is a solver-based optimization problem setup, which is suitable for functions or matrices. The third reason is that the least-squares solver can handle complex data, which is the case in the inverse Laplace transform. A nonlinear least-squares solver solves fitting problems of the form [5]

$$\min_x f(x) = ||f(x)||_2^2 = \min_x(f_1(x)^2 + f_2(x)^2 + ... + f_n(x)^2) \tag{5.1}$$

In this equation, $f(x)$ is the function and n is the number of data points. It is then important to find the nonlinear least-squares solver in Matlab that is most suitable for the numerical inversion of the Laplace transform. The nonlinear least-squares solver that will be used is the function '*lsqnonlin*'. This function can handle differences in the number of data points between simulated experimental data and the data of the performed inverse Laplace transform. Just as dealing with complex values.

The fitting algorithm used inside '*lsqnonlin*' is the Levenberg-Marquardt method [11]. This method uses a search direction d_k that is a solution of the linear set of equations

$$(J(x_k)^T J(x_k) + \lambda_k diag\Big(J(x_k)^T J(x_k)\Big))d_k = -J(x_k)^T F(x_k) \tag{5.2}$$

In this equation, $J(x_k)$ is the Jacobian of the function $f(x)$,λ_k is a scalar, and $diag(A)$ is the matrix of diagonal terms in A. The algorithm tries to find a minimum. If the step is successful and a lower value is found, the algorithm sets $\lambda_{k+1} = \lambda_k/10$, if not the algorithm sets $\lambda_{k+1} = \lambda_k \cdot 10$. This way, d_k is changed until it will find a minimum unless a stopping criterion is met. An example of the visualization of the algorithm with two parameters can be seen in Figure 5.1 (taken from [1]).

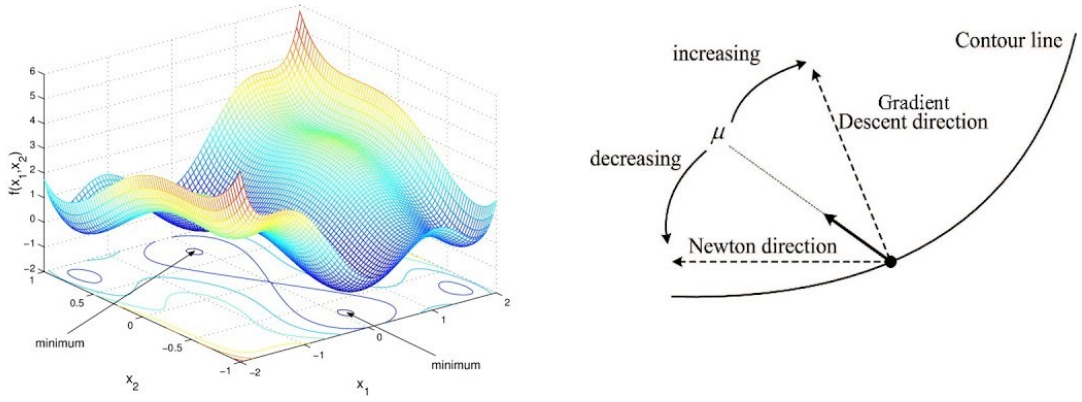

Figure 5.1: Example of the Levenberg-Marquardt method

In the above figure, it can be seen that the Levenberg-Marquardt method on two parameters can be visualized as a 3D space in which the algorithm will find a minimum. In the image to the right of the 3D space, μ is equal to d_k. The upper and lower bounds of the parameters that will be fitted are the same as in Table 4.1.

5.2 Functions

The fitting technique will be applied to three functions to assess the error of the inverse Laplace transform by the Sheen Contour. Each function will have noise added to the data in order to simulate experimental data. The maximum noise is set to be approximately 10 % of the maximum value of the functions. For all functions, this results in a maximum noise of 0.1 since the maximum value of the functions is 1.0. The functions described below all satisfy the conditions described in chapter 2.

5.2.1 Function 1: $f(t) = e^{-t^2}$

This function is the same on which the Sobol method is performed. The function plot can be seen in Figure 4.2. This function is decaying rather quickly since the value of $f(5) = 1.4 \cdot 10^{-11}$. The Laplace transform of this function is already described in Equation 4.14.

5.2.2 Function 2: $f(t) = e^{-(t/5)^2}$

This function is a variation of the above function. The reason a value of $t/5$ is chosen instead of t is that this function does not decay so quickly. The function plot can be seen in Figure 5.2.

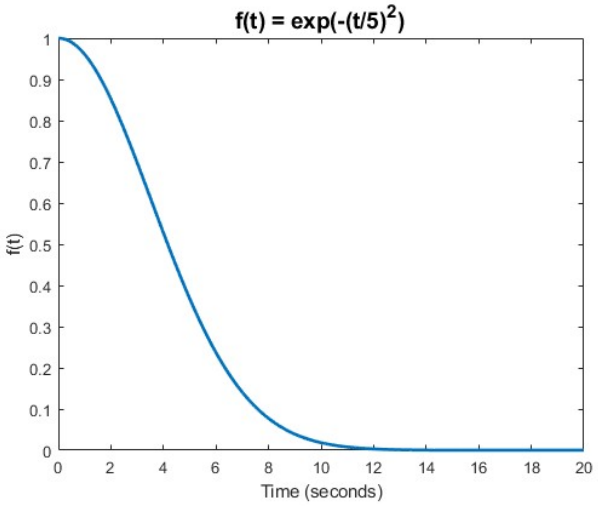

Figure 5.2: Function plot of $f(t) = e^{-(t/5)^2}$

The Laplace transform $F(z)$ of this function is as follows

$$F(z) = \frac{5\sqrt{\pi}}{2} e^{(25z^2)/4} \left(1 - erf\left(\frac{5z}{2}\right)\right) \tag{5.3}$$

5.2.3 Function 3: $f(t) = e^{-t}\cos(10t)$

The last function implements a cosine multiplied with an exponential function. The exponential function is added to satisfy the conditions in chapter 2. The function plot can be seen in Figure 5.3.

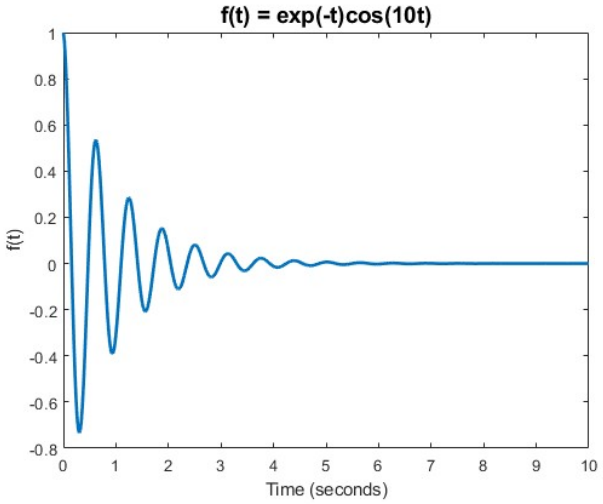

Figure 5.3: Function plot of $f(t) = e^{-t}\cos(10t)$

The Laplace transform $F(z)$ of this function is as follows

$$F(z) = \frac{z+1}{(z+1)^2 + 100} \tag{5.4}$$

5.3 Fixing parameters

As discussed in the result of the Sobol method, it will be investigated to fix parameters α, s, and τ before performing the parameter optimization to see if every parameter is necessary to fit. Fitting fewer parameters will namely mean less computational time is needed in order to fit the parameters and the outcome of fitting with two parameters will be more accurate than fitting on five parameters. This is for the reason that the Levenberg-Marquardt Algorithm only needs to find the minima in a 3D space. With 5 parameters, this would be a 6D space. The possibility that a global minimum is found in a 6D space is much smaller than in a 3D space. In real life, there will not be much experimental data available to fit the functions onto. It is, therefore, chosen to fit the parameters to the first 10 seconds. The mean absolute error will then be calculated for the first second, the first 10 seconds, the first 50 seconds, and the first 100 seconds while fixing the three parameters. It is also stated if the fitted parameter values differ from the initial guesses during the fit since this could result in a wrong interpretation of the fit. Before going to the results, the technique of selecting the fixed values for the other parameters α, s, and τ is explained. This is done by making use of the same technique described in section 3.3. For every function, the singularities can be plotted. It is then important to first make a Sheen Contour that complies with the most important aspect of the Sheen Contour. Namely, all singularities of the Laplace transform $\hat{f}(z)$ of a function $f(t)$ should lie to the left of the integration contour Γ. Besides this, it is preferable to have the contour as close to the singularities as possible.

5.3.1 Result of $f(t) = e^{-t^2}$

For the fitting of this function, the same parameter values were used that were found in section 3.3 in Figure 3.11. The Sheen Contour in this figure was already close to the singularities with all the singularities to the left of the contour. The result of the mean absolute error for different t without the fitting can be found in Table 5.1 below.

	t=1 s	t=10 s	t=50 s	t=100 s
$f(t) = e^{-t^2}$	0.082	0.009	0.002	0.001

Table 5.1: Mean absolute error for different t for the initial guess of $f(t) = e^{-t^2}$

The parameters β and n_z are now fitted with the Levenberg-Marquardt algorithm. The new value for $\beta = 0.0158$ and for $n_z = 31$. The results of the fit can be seen in Table 5.2 below.

	t=1 s	t=10 s	t=50 s	t=100 s
$f(t) = e^{-t^2}$	0.030	0.005	6.705	$4.6{\cdot}10^4$

Table 5.2: Mean absolute error for different t for the fitted parameters of $f(t) = e^{-t^2}$

It clearly can be seen that the inverse Laplace transform with the fitted parameters performs better for the first 10 seconds. The reason for this is that the parameters are fitted to data of 10 seconds. However, after 10 seconds the solution of the inverse Laplace transform with the fitted parameters blows up. This can be seen in Figure 5.4.

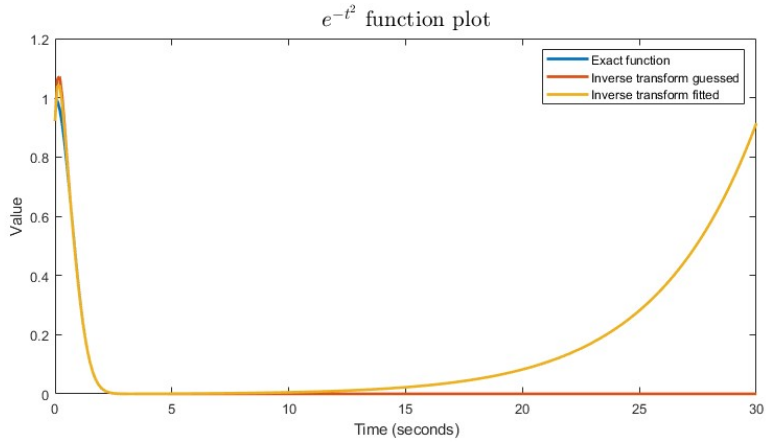

Figure 5.4: Function plot of $f(t) = e^{-t^2}$ with the initial guess and fitted parameters

Before making conclusions, the next function $f(t) = e^{-(t/5)^2}$ is considered.

5.3.2 Result of $f(t) = e^{-(t/5)^2}$

There is again looked at the singularities of this function to come up with an initial guess for the parameters. After trying different values, the contour was made that can be seen in Figure 5.5.

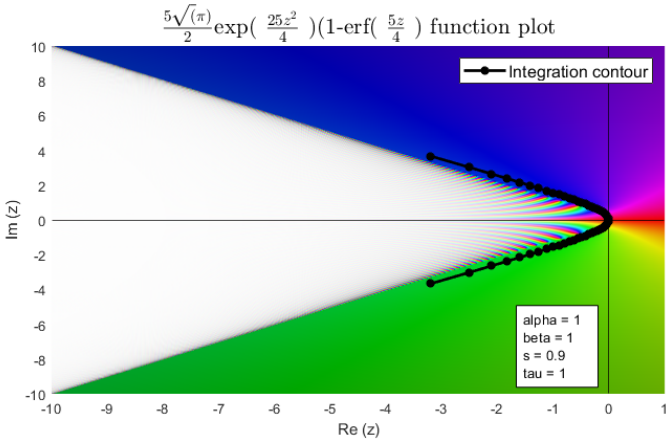

Figure 5.5: Sheen Contour of $f(t) = e^{-(t/5)^2}$ with $n_z = 30$

The result of the mean absolute error of the above Sheen Contour with the parameters for different t can be found in Table 5.3 below.

	t=1 s	t=10 s	t=50 s	t=100 s
$f(t) = e^{-(t/5)^2}$	0.043	0.005	0.001	0.013

Table 5.3: Mean absolute error for different t for the initial guess of $f(t) = e^{-(t/5)^2}$

The parameters β and n_z are now fitted with the Levenberg-Marquardt algorithm. However, the fitted values for $\beta = 1$ and for $n_z = 30$, which is no different from the already guessed values. This can also be seen in Figure 5.6. This means that fitting has not made a difference. In the figure, there can also be seen that the larger mean absolute error for $t = 100$ s comes from the solution of the inverse Laplace transform that blows up around $t = 80$ s.

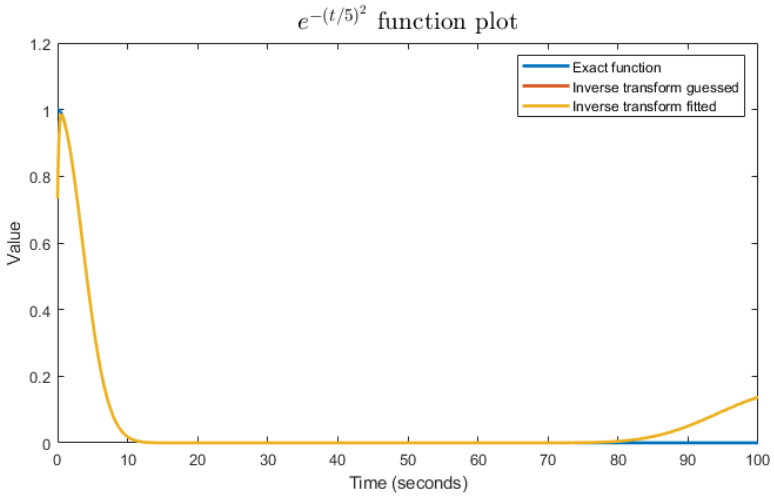

Figure 5.6: Function plot of $f(t) = e^{-(t/5)^2}$ with the initial guess and fitted parameters

5.3.3 Result of $f(t) = e^{-t}\cos(10t)$

For this function, it is more difficult to come up with an initial guess for the parameters. When plotting the Laplace transform of the function, no singularities can be retrieved. However, when looking at the function of the Laplace transform, it can be calculated that there is a singularity at $z = -1$. This is namely a pole of the function, which is a special type of singularity. This means that $\beta - \alpha > -1$, as this is the point on which the Sheen Contour crosses the real axis. Besides this, the initial guess of the parameters can not be done on the Laplace transform itself. Therefore, the initial guess of the parameters will be done on the simulated experimental data. The result of trial and error with the initial guess can be found in Figure 5.7.

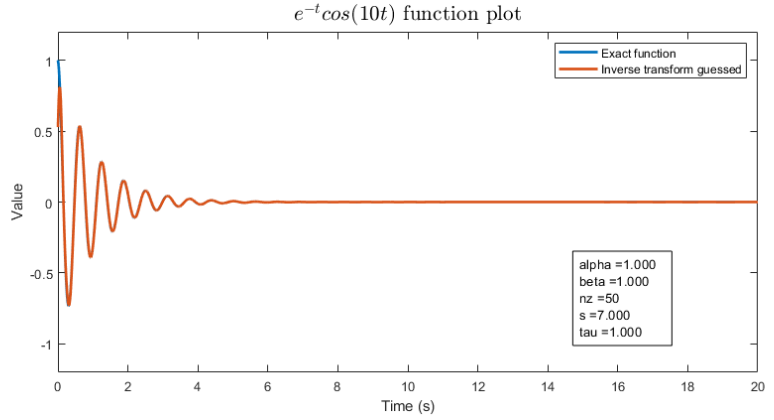

Figure 5.7: Function plot of $f(t) = e^{-t}\cos(10t)$ with the initial guess

The result of the mean absolute error of the above Sheen Contour with the parameters for different t can be found in Table 5.4 below.

	t=1 s	t=10 s	t=50 s	t=100 s
$f(t) = e^{-t}\cos(10t)$	0.025	0.003	0.001	0.001

Table 5.4: Mean absolute error for different t for the initial guess of $f(t) = e^{-t}\cos(10t)$

This fit is already pretty accurate. The parameters β and n_z are now fitted using the Levenberg-Marquardt algorithm. The new value for $\beta = 1.1445$ and n_z stays at 50. The result of the fit can be seen in Table 5.5 below.

	t=1 s	t=10 s	t=50 s	t=100 s
$f(t) = e^{-t}\cos(10t)$	0.049	0.007	0.002	0.001

Table 5.5: Mean absolute error for different t for the fitted parameters of $f(t) = e^{-t}\cos(10t)$

As can be seen in the table above, the inverse Laplace transform with the fitted parameters performs worse than the initial guess. However, when plotting both inverse Laplace transforms, there can be seen that the fitted parameters remove the random oscillation at $t = 22$ s. This can be seen in Figure 5.8. The differences between the fitted inverse Laplace transformation and the guessed one are for the oscillating part is very small.

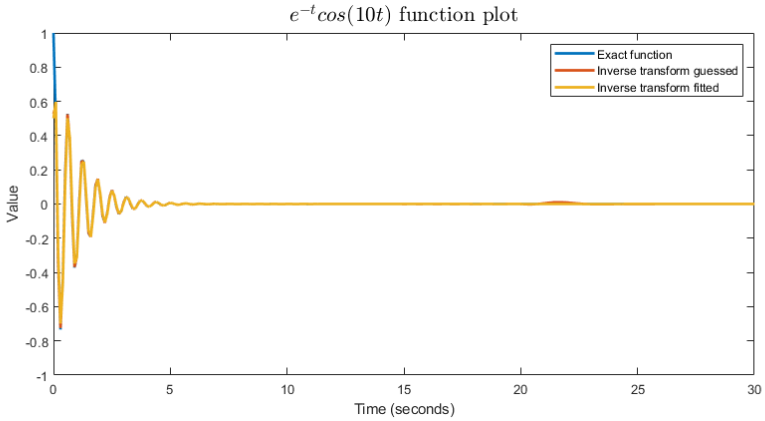

Figure 5.8: Function plot of $f(t) = e^{-t}\cos(10t)$ with the initial guess and fitted parameters

5.3.4 Conclusion of the fitting algorithm

From the above three functions, it can be seen that the fitting algorithm is reducing the absolute mean error compared to the inital guess. For the function $f(t) = e^{-t^2}$ this is only for the first ten seconds on which the inverse Laplace transform is fitted, and for the function $f(t) = e^{-t}\cos(10t)$ the fitted parameters are more stable as $t \to \infty$. The function $f(t) = e^{-(t/5)^2}$ does not show any difference between the initial guess and the fitted parameters. From this, it is concluded that it depends on the function if the fitted parameters are better. Stability as $t \to \infty$ is not guaranteed when fitting the parameters. More complex functions should be explored to see if this fitting algorithm works or if too many parameters are fixed.

5.4 Fitting technique

The next thing that is investigated, is the fitting technique. Currently, the inverse Laplace transform by the Sheen contour is fitted to the entire experimental data set at once. This often results in values for α, β, n_z, s, and τ that are not fitting the experimental data well in the first few seconds. It is, therefore, interesting to see if splitting the fit into multiple parts is beneficial. The values retrieved from the first fit will then be the initial guesses for the second fit. It is namely conducted out of experiments that the fitted parameter values returned by '*lsqnonlin*' depends highly on the initial guesses. The investigation of the fitting technique will be done on the same functions, as described in section 5.2. For the experimental data set, there is data from t=0 seconds to t=10 seconds. The data will be split into two different parts, namely from $t = 0$ to

$t = 10$ seconds, and from $t = 0$ to $t = 1$ seconds. The fit will first be done on the data from $t = 0$ to $t = 10$ seconds, just like in section 5.3. Afterward, one more fit will be performed on the data from $t = 0$ to $t = 1$ seconds with the initial guess taken from the previous fit. This will be called the piece-wise method.

5.4.1 Result of $f(t) = e^{-t^2}$

The result of the first fit is exactly the same as Table 5.2 since it was the same fit from $t = 0$ to $t = 10$ seconds. The result of the second fit on the simulated experimental data from $t = 0$ to $t = 1$ seconds can be seen in Table 5.6. It can be seen in this table that the newly retrieved fit is performing worse than the first fit. Even for the first second, the fit underperforms. The result of the difference between the two fits and the exact function can be seen in Figure 5.9. In this figure, little difference can be seen between the old fit and the new fit. However, from Table 5.6 it can be concluded that the new fitting technique is not better.

	t=1 s	t=10 s	t=50 s	t=100 s
$f(t) = e^{-t^2}$	0.032	0.006	14.014	$1.9{\cdot}10^5$

Table 5.6: Mean absolute error for different t for the piece-wise fit of $f(t) = e^{-t^2}$

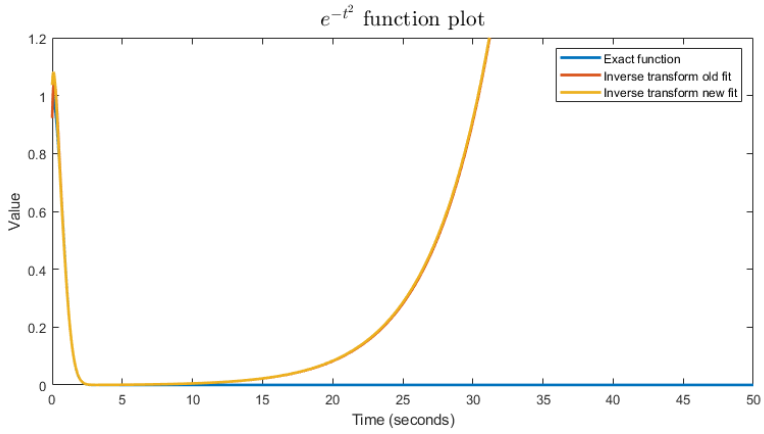

Figure 5.9: Function plot of $f(t) = e^{-t^2}$ with the old fit and newly fitted parameters

5.4.2 Result of $f(t) = e^{-(t/5)^2}$

The result of the first and second fit kept the same as Table 5.3. Apparently, the algorithm already finds a possible local minimum at the initial guess for both time frames, which results in getting back exactly the same initial guess. Fitting the inverse Laplace transform twice would only result in more time, from which it is concluded that the new fitting technique is not worth it.

5.4.3 Result of $f(t) = e^{-t}\cos(10t)$

The result of the piece-wise fit for $f(t) = e^{-t}\cos(10t)$ did not make a significant difference. The value of β went from $\beta = 1.1445$ to $\beta = 1.1438$. For the absolute mean error, it did not make any difference at all and the figure still looked exactly the same as in Figure 5.8. From this, it can be concluded that a piece-wise fitting method does not make any improvements to the fitted parameters of the inverse Laplace transform by the Sheen Contour.

Chapter 6

Conclusion and Future work

The inverse Laplace transform of a function $f(t)$ that is locally integrable and has a decay over time is performed using Equation 2.2. Talbot devised a method for the integration bounds that also can handle noise in data, on which the Sheen Contour is based. The Sheen Contour replaces the bounds of the integral, as in Equation 3.1. The Sheen Contour itself is given by Equation 3.3 which makes a hyperbolic shape. The inverse Laplace transform of a function $f(t)$ can then be calculated entirely using Equation 3.11. This formula has five different parameters that must be tuned to the specific problem. These parameters are α, β, n_z, s, and τ and each parameter has a different effect on the hyperbola. The goal of this hyperbola is to have all singularities of the Laplace transform $F(z)$ of $f(t)$ lay on the left of the Sheen Contour and to fit the contour as close to the singularities as possible.

By making use of the Sobol method, the importance of the parameters α, β, n_z, and s from $t = 0$ to $t = 5$ seconds for an arbitrary function have been explored. The Sobol method gives each parameter an index from 0-1 based on the influence on the variance of the output. The results can be found in Figure 4.3 and Figure 4.4 from which it is retrieved that α and s will have the lowest influence on the output for a longer time. Furthermore, τ behaved similar to s. For this reason, the parameters α, s, and τ will be left out of the parameter optimization.

The parameter optimization has been done using a nonlinear least-squares solver using the Levenberg-Marquardt algorithm on three different functions. It was found that the optimized parameters in one case performed better only in the first ten seconds and in one case the optimized parameters performed better as $t \to \infty$ than the guessed fit. From this, it is conducted that it depends on the function if the fit is better. The stability of the inverse Laplace transform is also not guaranteed with the fitting algorithm. However, fitting the parameters two times while taking the parameters from the first time as an initial guess for the second time will not make the fit better.

This gives still future work to do. One can test the fitting algorithm on even more complex functions, such as the heat equation. It can then be really seen if this fitting algorithm can work in a real-life setting, such as a 3D printer hooked up to a digital twin. Other possibilities are to see if the fitting algorithm works if fewer parameters are fixed.

Bibliography

[1] Behnam Asadi. Levenberg-Marquardt algorithm explained, 10 2019. 18

[2] Nasir Bilal. Implementation of Sobol' s Method of Global Sensitivity Analysis to a Compressor Simulation Model. Technical report, Purdue University, 2014. 11

[3] I. J. D. Craig, A. M. Thompson, and William J. Thompson. Practical Numerical Algorithms Why Laplace Transforms Are Difficult To Invert Numerically. *Computers in Physics*, 8(6):648, 1994. 2

[4] Colin L. Defreitas and Steve J. Kane. The noise handling properties of the Talbot algorithm for numerically inverting the Laplace transform. *Journal of Algorithms and Computational Technology*, 13, 2019. 2

[5] John E. Dennis Jr. Nonlinear Least Squares Data Fitting. In D. Jacobs, editor, *State of the Art in Numerical Analysis*, pages 269–312. Academic Press. 18

[6] Frank A. Farris. Visualizing complex-valued functions in the plane. 9

[7] H. Christopher Frey and Sumeet R. Patil. Identification and review of sensitivity analysis methods. In *Risk Analysis*, volume 22, pages 553–578, 2002. 10, 11

[8] Graham Glen and Kristin Isaacs. Estimating Sobol sensitivity indices using correlations. *Environmental Modelling & Software*, 37:157–166, 11 2012. 17

[9] Antonia Hadjimichael. Determining the appropriate number of samples for a sensitivity analysis, 3 2020. 15

[10] G Honig and U Hirdes. A method for the numerical inversion transforms of Laplace. Technical report, 1984. 2

[11] MathWorks. Least-Squares (Model Fitting) Algorithms, 2022. 18

[12] A. Saltelli and P. Annoni. How to avoid a perfunctory sensitivity analysis. *Environmental Modeling and Software*, 25:1508–1517, 2010. 11

[13] Dongwoo Sheen, Ian H Sloan, Vidar Thomée, and Thom´ Thomée. A parallel method for time discretization of parabolic equations based on Laplace transformation and quadrature. Technical report, 2003. 1, 3

[14] I M Sobol. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Technical report, 2001. 12

[15] A Talbot. The Accurate Numerical Inversion of Laplace Transforms. Technical report, 1979. 2

[16] J. A.C. Weideman. Optimizing Talbot's contours for the inversion of the Laplace transform. *SIAM Journal on Numerical Analysis*, 44(6):2342–2362, 2006. 1, 5