Numerical solutions to the self-contained soliton model

– and the potential connection between the soliton theory and lithium treatment in bipolar disorder

Written by *Dana Kamp*
July 20, 2021

Supervised by
Prof. Dr. Thomas Heimburg

University of Copenhagen

UNIVERSITY OF COPENHAGEN

NAME OF INSTITUTE:	Niels Bohr Institutet
NAME OF DEPARTMENT:	Biocomplexity, Membranes group
AUTHOR(S):	Dana Kamp
EMAIL:	rhf801@alumni.ku.dk (dana.kamp@protonmail.ch)
TITLE AND SUBTITLE:	Numerical solutions to the self-contained soliton model – and the potential connection between the soliton theory and lithium treatment in bipolar disorder
SUPERVISOR(S):	Prof. Dr. Thomas Heimburg
HANDED IN:	20th of July 2021
DEFENDED:	Between 17th and 20th of August 2021

NAME ______________________________

SIGNATURE __________________________

DATE _______________________________

Abstract

The soliton theory of the nerve pulse is presented with focus on the case of density dependent dispersion. The density dependent dispersion coefficient has been derived previously by Mosgaard [56] and involves the excess heat capacity of the membrane that needs to be expressed in terms of density for the model to be self-contained. The heat capacity is modelled using thermodynamics. Using MATLAB®'s ODE-solvers and table values from DPPC bilayers I solve the self-contained model alongside two cases of constant dispersion and a density-dependent approximation to the self-contained model where the heat capacity is fitted from experimental data of DPPC bilayers. The initial conditions were optimized for solitary solutions using the bisection method. In the case of a constant dispersion coefficient I found a maximum density change of $\approx 20\%$ relative to equilibrium, in agreement with literature [33, 44]. The solitons of the self-contained model had widths of approximately 1 meter and and maximum relative density changes of $\approx 8\%$. In comparison the widths of the density dependent model with experimentally fitted heat capacity are approximately 0.5 m, and maximum relative density changes are $\approx 17\%$. In both cases of density dependent dispersion some velocities had two solutions of similar dimensions. The difference in dimensions between the self-contained theoretical model of the heat capacity and the experimental approximation is due to differences in cooperativity. I attempted to investigate the effect of cooperativity in the self-contained model but only managed to find solitary solutions for a small range of cooperativities due to numerical instability. Suggestions on how to improve my methods are included in the discussion. Through a literature review I bridge a connection between the soliton theory and the therapeutic effect of lithium on a variety of human conditions, in particular bipolar disorder. Studies that have looked into the thermodynamics of lithium ions have found remarkable effects on anionic lipid bilayers and the heat generation of nerves. This suggests lithium's prophylaxis might be connected to the interaction between the ion and the lipids in the nerve membrane. I investigated lithium's effect on zwitterionic lipid bilayers by measuring the heat capacities, thermal volume expansion coefficients and relaxation times of DPPC LUV suspended in water containing either lithium, sodium or potassium. The meaurements were performed by differential scanning calorimetry and pressure pertubation calorimetry. The lithium sample had a simpler heat capacity profile than sodium, potassium and control sample and showed a greater proportionality between the relaxation times and the heat capacity profile. The proportionality constants of all four samples were within the range of literature with values of $5{\cdot}10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}} - 9{\cdot}10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}}$ and standard deviation of $3{\cdot}10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}} - 6{\cdot}10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}}$. From the relationship between relaxation times and heat capacities I derived Onsager's phenomenological constant for each sample, they were all in the magnitude of $10^8\,\frac{\mathrm{J\cdot K}}{\mathrm{s\cdot mol}} - 10^{10}\,\frac{\mathrm{J\cdot K}}{\mathrm{s\cdot mol}}$, which is in agreement with literature. The subtle differences between lithium and sodium, potassium and control is compared to lithium's unique ability among the monovalent cations to induce order in anionic lipid membranes. However, the effect of lithium on lipid membranes in vitro cannot directly be extrapolated to the prophylaxis of lithium since the metabolic response of the human body must be taken into account.

Contents

1 Preface

Like any production made for broad audiences within the last 50 years, I will start with a disclaimer. In literature on the soliton theory, the terms adiabatic and isentropic are used interchangeably when deriving thermodynamic properties, since the nerve pulse is assumed to be a reversible process. In this thesis I try to stick to isentropic just to avoid confusion. Some mathematicians [16] have complained the term 'soliton' is misused since collisions between nerve pulses are not fully elastic. This claim is likely based on a single study by Tasaki from 1949 [74], but Heimburg and associates have recently documented several predominantly elastic nerve pulse collisions [21, 56]. Pulses based on the soliton model has likewise been shown to pass through each other generating only a small amount of noise [4], so I decide it is reasonable to stick to the term, though it might anger some purists.

2 Introduction

2.1 The physics of nerve excitation

Nerve signaling is a commonly used term for our body's ability to pass information along the nerves of the body. It is a crucial part of controlling our bodily functions and its concepts has been the inspiration of technological innovations such as neural networks. Nevertheless many fundamental processes associated with this mechanism has yet to be fully understood.

The most extensively studied phenomenon taking place during the nerve signal is the electrical changes, dicovered already in the late 1800s by Galvani and Volta [31]. During a signaling event the transmembrane voltage increases approximately 100 mV, followed by a small hyperpolarization where the voltage lies slightly below that of the initial resting state. These changes in membrane potential are denoted 'the action potential' and are associated with a decrease in membrane resistance [78] and capacitance (of about 50%) [75, 32] and an increase in transmembrane current [35]. The increase in transmembrane current is caused by an increase in ion permeability for cations such as Na^+, K^+ and Ca^{2+} [78, 79, 35]. These fluxes are very small compared to the concentrations in the external environment [1, 81], but in relative terms quite impressive: sodium and potassium each go through an approximately 100 fold increase in permeability with individual time courses during the signaling event [1]. The influx of Ca^{2+} ions is much less than that of Na^+ and K^+ [79], but Ca^{2+} are by no means less important to the existence of the nerve signal: Removal of external calcium abolishes the nerve signal [79].

A lesser known feature is the mechanical changes accompanying the nerve signal. Swelling and contraction have been extensively documented by Tasaki and associates in a various nervous tissues such as squid axon and olfactory nerves of frogs and fish [39, 77, 80]. The cause of the swelling is not entirely clear. It might partly be caused by altered membrane hydration due to the changing concentrations of metal ions during the nerve pulse [75]. This cannot be the whole story though, since there is a net volume expansion of the nerve during the nerve pulse that cannot be explained by internal movement of water of the system [76]. It might instead indicate that structural changes are taking place in the nerve membrane [81, 33]. Another indicator of this is that the nerve pulse alters the optical properties of the membrane, changing the order and orientation of membrane lipids [81, 32]. These transient alterations are practically adiabatic in the sense that for both myelinated and unmyelinated nerves the majority ($> 80\%$) if not all of the net heat produced during the signal is absorbed back into the system [75, 34, 65]. Since the measured temperature change is in the μK range [65], which is minuscule relative to ambient temperature (≈ 300 K), the signal must furthermore be isentropic: The second law of thermodynamics states that for reversible processes in closed systems $dQ = TdS$. With approximately reversible heat transfer and constant temperature, we then have $\oint dQ \approx 0 \rightarrow \oint dS \approx 0$ [29, chap. 18.2.1].

The many phenomena of the nerve signal are coupled, which can be illustrated by the fact that though studies usually induce nerve signals using electricity, they can equally as well be initiated through mechanical stimulation [31] or local cooling [34]. Trying to explain the workings of nerve signals by focusing on one or two aspects of the mechanism, such as electrical changes and ion conduction [35, 75], would be to disregard the internal coupling between the processes and one would not have a complete understanding of the overall mechanism.

This thesis will concern itself with the soliton theory, a well-established theory that manages to unify the many aspects of nerve signaling. Through the formalism of linear non-equilibrium thermodynamics it explains the couplings of the permeability, electricity and heat changes during the nerve signal. The theory has great predictive abilities, as an example it can explain how anesthesia renders the membrane inexcitable and why this can be reversed by pressure [34, 30, 42]. It is based on the idea that the nerve signal includes a longitudal density wave in the nerve membrane, causing the lipids of the membrane to transition temporarily from a liquid crystal to a gel with a variety of thermodynamical implications. Originally proposed by Kaufmann [37], it inspired Heimburg to formulate a theory for the thermodynamic excitation of nerves, which in collaboration with Jackson was put into mathematical formalism in their 2005 paper [33]. The initial mathematical model was based on hydrodynamics and is a 4th order variation on the wave equation with a density-dependent propagation speed and a constant dispersion coefficient. The equation produces single or repeated localized pulses with permanent wave profiles [83], better known as 'solitons'.

Since 2005 there has been notable theoretical contributions by ex. Villagran-Vargas et al. [83] and Lautrup et al. [44] on the properties and stability of the soliton equation, and more recently by Mosgaard [56] on incorporating density dependence into the dispersion coefficient. I have continued the work of Mosgaard and will present numerical solutions to the soliton model of density dependent dispersion, alongside a motivation on the potentially thermodynamic nature of lithium's therapeutic effect on various human conditions such as bipolar disorder.

2.2 Introduction to the soliton theory

A simpler and more common term for isentropic density waves is simply: 'sound'. In the next chapter I will go through the basics of sound propagation in membranes with brief introductions to the necessary thermodynamics to understand the soliton theory.

2.2.1 Dynamic susceptibilities and sound

Studies on membranes of single cell organisms and mammalian tissues have found their melting temperature to be consistently $10^\circ - 20^\circ$C below physiological temperature of the associated species [33, 60, 58]. The melting profiles are very wide, often with a half width of $\approx 15^\circ$C. A biological membrane is therefore close to phase transition under normal physiological conditions. Membrane phase transition creates non-linear compressibilities and dispersion, which is the criteria for the existence of solitons [29, chap. 18.3.1].

The original Heimburg-Jackson model from 2005 [33] proposes density pertubations travel like mechanical waves along the membrane in the following manner:

$$\frac{\partial^2 \Delta\rho^A}{\partial t^2} = \frac{\partial}{\partial x}\left(\left(c^A\left(\rho^A\right)\right)^2 \frac{\partial \Delta\rho^A}{\partial x}\right) - h\frac{\partial^4 \Delta\rho^A}{\partial x^4} \tag{1}$$

Where the speed of sound, c^A, is dependent on the local area density, and there has been added spatial dispersion with a constant dispersion coefficent h. The soliton theory models the nerve axon as approximately one-dimensional and infinitely long in order to simplify the problem. Since a single motor neuron in the human body has a diameter of approximately 0.1 millimeter and can grow up to 1 meter [19], this is not completely without reason. The superscript 'A' for 'area' is to denote the changes are taking place in the plane.

In fluids, the speed of sound given by the Newton-Laplace equation [84]:

$$c = \frac{1}{\sqrt{\kappa_S \rho}}$$

Here κ_S is the isentropic compressibility and ρ the density of the media. The isentropic compressibility κ_S has great influence on the speed of sound near phase transition. Density decreases monotonically with increasing temperature, more rapidly in transition than outside [13]. The speed of sound is not a monotonic function, it has a marked local minimum at transition [23, 28, 69]. This is due to a local maximum in the isentropic compressibility at the same temperatures [23]. To illustrate why the isentropic compressibility shows a maximum during phase transition I will demonstrate how it can be directly related to the changes in the system's heat capacity. This relation is an important feature of lipid bilayers at phase transition. I begin by relating isothermal and isentropic volume compressibilities of a lipid bilayer in an aqueous reservoir. The derivation is identical to derivation for volume compressibility in Heimburg [29, chap. 4.12].

Volume compressibilities describes the relative rate at which the volume of a system changes when subjected to pressure under certain constraints, such as constant temperature (isothermal) or entropy (isentropic).

$$\kappa_T^V = -\frac{1}{\langle V\rangle}\frac{d\langle V\rangle}{dp}\Big|_T, \quad \kappa_S^V = -\frac{1}{\langle V\rangle}\frac{d\langle V\rangle}{dp}\Big|_S$$

The variables I need to connect between the two equations are volume, entropy, pressure and temperature. These variables can be related by approximating the change in volume and entropy for small changes in pressure and temperature.

$$d\langle V\rangle = \left(\frac{\partial\langle V\rangle}{\partial p}\right)_T dp + \left(\frac{\partial\langle V\rangle}{\partial T}\right)_p dT \quad \text{and} \quad dS = \left(\frac{\partial S}{\partial p}\right)_T dp + \left(\frac{\partial S}{\partial T}\right)_p dT = 0 \tag{2}$$

Since the process is isentropic, the second part of equation 2 is equal to zero. I can use this equation to express dT in terms of entropy:

$$dT = -\frac{\left(\frac{\partial S}{\partial p}\right)_T}{\left(\frac{\partial S}{\partial T}\right)_p} dp \quad \text{and} \quad d\langle V\rangle = \left(\frac{\partial\langle V\rangle}{\partial p}\right)_T dp - \left(\frac{\partial\langle V\rangle}{\partial T}\right)_p \frac{\left(\frac{\partial S}{\partial p}\right)_T}{\left(\frac{\partial S}{\partial T}\right)_p} dp$$

The change in entropy with respect to temperature can be related to the heat capacity of the system:

$$\frac{\partial S}{\partial T} = \frac{\partial Q}{T\partial T} = \frac{c_p}{T}$$

Using Maxwell's relation, $\frac{\partial S}{\partial p}_T = -\frac{\partial\langle V\rangle}{\partial T}_p$, the entropy change with respect to pressure can be rephrased:

$$\left(\frac{\partial\langle V\rangle}{\partial p}\right)_S = \left(\frac{\partial\langle V\rangle}{\partial p}\right)_T + \left(\frac{\partial\langle V\rangle}{\partial T}\right)_p^2 \frac{T}{c_p}$$

Multiplying the above with $-\frac{1}{\langle V\rangle}$ leads to the following expression for the isentropic compressibility:

$$\kappa_S^V = \kappa_T^V - \frac{T}{\langle V\rangle c_p}\left(\frac{\partial\langle V\rangle}{\partial T}\right)_p^2 \tag{3}$$

$\alpha_V = \frac{1}{\langle V\rangle}\left(\frac{\partial\langle V\rangle}{\partial T}\right)_p$ is the thermal volume expansion coefficient of the membrane. This coefficient is proportional to the heat capacity due to an empirical proportionality between the change in volume and change in entalpy with respect to temperature for lipid bilayers close to transition [2, 28]:

$$\frac{1}{\langle V\rangle}\left(\frac{d\langle\Delta V\rangle}{dT}\right)_p = \frac{\gamma_V}{\langle V\rangle}\left(\frac{d\langle\Delta H\rangle}{dT}\right)_p, \quad \text{i.e.} \quad \alpha_V = \frac{\gamma_V}{\langle V\rangle}\Delta c_p \tag{4}$$

Where Δc_p signifies the *excess* heat capacity. In order to insert this into equation 3, I need to justify that the volume change close to transition is dominated by lipid melting, such that I can ignore contributions from the surroundings. Water changes very little at these temperatures, but the pure gel and fluid phases do expand measurably on their own [23, 28]. Yet these expand in a slow, linear manner, which will just cause an small offset in the compressibility that can be corrected for. I set $\left(\frac{d\langle\Delta V\rangle}{dT}\right)_p = \frac{\partial\langle V\rangle}{\partial T}_p$ and insert this into equation 3:

$$\kappa_S^V = \kappa_T^V - \frac{T}{\langle V\rangle c_p}\left(\gamma_V \Delta c_p\right)^2 \tag{5}$$

It is now time to take a look at the isothermal compressibility, κ_T^V. Isothermal compressibility can be related to the variance in volume fluctuations at equilibrium if I express the average volume, $\langle V\rangle$, in the definition of isothermal compressibility as the average of all microstates of the system, [29, chap. 4.9 & 4.10.1]:

$$\begin{aligned}\kappa_T^V &= -\frac{1}{\langle V\rangle}\left(\frac{\partial\langle V\rangle}{\partial p}\right)_T = -\frac{1}{\langle V\rangle}\frac{d}{dp}\sum_i V_i P_i, \quad \text{where} \quad P_i = \frac{\exp\left(-H_i/k_BT\right)}{Z}, \quad H_i = E_i - pV_i \\ &= -\frac{1}{\langle V\rangle}\left(-\frac{1}{k_BT}\sum_i V_i^2 P_i + \frac{1}{k_BT}\left(\sum_i V_i P_i\right)^2\right) \\ &= \frac{\langle V^2\rangle - \langle V\rangle}{\langle V\rangle k_BT}\end{aligned}$$

Variances are additive, which means I can split the isothermal compressibility into volume changes related to lipid melting and reservoir contributions:

$$\kappa_T^V = \kappa_{T,0}^V + \frac{\langle(\Delta V)^2\rangle - \langle\Delta V\rangle^2}{\langle V\rangle k_BT}$$

If one assumes the proportionality between volume change and entalpy in equation 4 holds for all temperatures, it follows from mathematical arguments that the proportionality must be upheld by each microstate of the system [28], i.e.

$$\Delta V_i = \gamma_V \cdot \Delta H_i$$

As a natural consequence, the variances in volume change and entalpy change must also be proportional:

$$\begin{aligned}
\frac{\sum V_i^2 \exp(-H_i/k_BT)}{Z} &= \frac{\sum (\gamma_V H_i)^2 \exp(-H_i/k_BT)}{Z} \\
\text{i.e.} \quad \langle(\Delta V)^2\rangle &= \gamma_V^2 \langle(\Delta H)^2\rangle \\
\text{and} \quad \langle(\Delta V)^2\rangle - \langle\Delta V\rangle^2 &= \gamma_V^2 \left(\langle(\Delta H)^2\rangle - \langle\Delta H\rangle^2\right)
\end{aligned}$$

Where Z is the partition function of the system. I can therefore express the isothermal compressibility as a function of variance in entalpy:

$$\kappa_T^V = \kappa_{T,0}^V + \frac{\gamma_V^2\left(\langle(\Delta H)^2\rangle - \langle\Delta H\rangle^2\right)}{\langle V\rangle k_B T}$$

The heat capacity can be related to the variance in equilibrium entalpy fluctuations in the same manner as isothermal compressibility relates to volume fluctuations:

$$\Delta c_p = \frac{d\langle\Delta H\rangle}{dT} = \frac{\langle(\Delta H)^2\rangle - \langle\Delta H\rangle^2}{k_B T^2} \tag{6}$$

Thus I can use the heat capacity to express the isothermal heat capacity:

$$\kappa_T^V = \kappa_{T,0}^V + \frac{\gamma_V^2 T}{\langle V\rangle}\Delta c_p \tag{7}$$

Where the second term is the contribution of the melting transition, and the first term covers the rest.

Following the argumentation made by Mosgaard et al. [57] I split c_p in equation 5 into the excess heat capacity Δc_p and the reservoir heat capacity c_p^R (c_p^R also contains contributions from other lipid degrees of freedom than melting). Since the heat capacity of the aqueous environment greatly exceeds that of the lipid membrane, $\left(\Delta c_p + c_p^R\right)^{-1} \approx \left(c_p^R\right)^{-1}$.

$$\begin{aligned}
\kappa_S^V &= \left(\kappa_{T,0}^V + \frac{\gamma_V^2 T}{V}\Delta c_p\right) - \frac{T}{V\left(\Delta c_p + c_p^R\right)}\left(\gamma_V \Delta c_p\right)^2 \\
&= \left(\kappa_{T,0}^V + \frac{\gamma_V^2 T}{V}\Delta c_p\right) - \frac{T\gamma_V^2}{V}\frac{\left(\Delta c_p\right)^2}{c_p^R} \\
&= \kappa_{T,0}^V + \frac{T\gamma_V^2}{V}\Delta c_p \cdot \left(1 - \frac{\Delta c_p}{c_p^R}\right)
\end{aligned} \tag{8}$$

This equation was derived in Mosgaard et al. [57]. It relates the isentropic compressibility directly to the excess heat capacity for lipid bilayers in phase transition, thus making it possible to obtain information about elastic properties of the system by measuring thermodynamic quantities. It also shows how the ratio between the membrane and reservoir heat capacities influence the compressibility of the membrane. In general the isentropic compressibility is lower than the the isothermal compressibility, which constitutes the upper limit to the isentropic compressibility in the case of an infinite reservoir. In the limiting case, both compressibilities follow the excess heat capacity closely, and since the excess heat capacity of lipid bilayers changes many orders of magnitude [23], so does the compressibility, predicting a minimum in the speed of sound [28]. If instead the reservoir heat capacity is finite and comparable in size to the membrane heat capacity, the compressibility will be strongly reduced. High frequency of sound has a similar effect as a limited reservoir [57]. This is related to the relaxation time of fluctuations in the lipid membrane and surrounding water, and to explain this well I need to give a short introduction to the field of linear, non-equilibrium mechanics.

Using the general expression for internal energy of a thermodynamic system, one can express the entropy of the system as the sum of contributions from each of the system variables divided by T. These terms are referred to as 'reaction variables' [29, chap. 4.3]:

$$dS = \frac{dE}{T} + \frac{p}{T}dV + \left(\frac{\Pi}{T}dA + \frac{F}{T}dx + \frac{\Psi}{T}dq + \frac{1}{T}\sum_i \mu_i dn_i ...\right)$$

Linear non-equilibrium mechanics builds on the approximation that close to equilibrium, entropy can be treated as a quadratic potential of these reaction variables, denoted ξ_i, with thermal equilibrium at the top (subscript 0) [34, chap. 4.13]:

$$S \approx S_0 + \frac{1}{2}\sum_i \sum_j \left(\frac{\partial^2 S}{\partial \xi_i \partial \xi_j}\right)\Big|_{\xi_{i,0},\xi_{j,0}} (\xi_i - \xi_{i,0})(\xi_j - \xi_{j,0}) \tag{9}$$

The slopes of the entropy potential, $X_i = \frac{\partial S}{\partial \xi_i}$, are named thermodynamic forces (notice the unconventional sign of the potential), and the changes in reaction variables over time, $J_i = \frac{\partial \xi_i}{\partial t}$, are named thermodynamic fluxes. The use of classic terminology gives a more intuitive feel for the system: Small displacements in a given variable induces a restoring force towards the maximum of the potential, equilibrium. Close to equilibrium, the fluxes are linearly related to the thermodynamic forces, following Onsagers reciprocal relations:

$$J_i = \sum_j L_{i,j} X_j, \quad L_{i,i} > 0, \quad L_{ij} = L_{j,i}, \quad \text{and} \quad \frac{L_{ij}}{L_{i,i}L_{jj}} \leq 1 \tag{10}$$

Where L_{ij} are phenomenological coupling constants between reaction variables.

I will now apply this formalism to investigate how a small fluctuation in a lipid bilayer relaxes back to equilibrium. In the case of melting lipid bilayer membranes, area and volume changes are linearly related to entalpy fluctuations, such that there is essentially only one reaction variable, entalpy. Any fluctuation is reversible, so the total entalpy of the system $\langle H \rangle$, is conserved. The distribution of possible system states depend only the entropy, which in the case of maximum entropy can be related to the fluctuations of the system using equation 9:

$$P_{system} = \exp\left(\frac{-(H - TS)}{k_B T}\right) \propto \exp\left(\frac{S}{k_B}\right) = \exp\left(\left(\frac{\partial^2 S}{\partial (\Delta H)^2}\right)\Big|_{H_0} \frac{(\Delta H)^2}{2k_B}\right) \quad \text{(in case of maximal entropy)} \tag{11}$$

Where $\Delta H = H - \langle H \rangle$ is a fluctuation in entalpy.

If the system is large enough, there is a sufficient number of possible states to approximate this distribution with a Gaussian:

$$P_{system}\left(\Delta H\right) \approx \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\left(\Delta H\right)^2}{2\sigma}\right) \propto \exp\left(-\frac{\left(\Delta H\right)^2}{2c_p \cdot k_B T^2}\right) \quad \text{(close to maximal entropy)} \qquad (12)$$

I mentioned earlier (equation 6) that the variance, σ^2, in entalpy fluctuations in equation 11 is equal to $c_p \cdot k_B T^2$. Comparing the exponents of equation 11 I conclude:

$$\left(\frac{\partial^2 S}{\partial\left(\Delta H\right)^2}\right)\Big|_{H_0} \approx -\frac{1}{T^2 c_p} \qquad (13)$$

Multiplying equation 13 with ΔH, the thermodynamic force, $\frac{\partial S}{\partial \Delta H}$, is found:

$$-\frac{\Delta H}{T^2 c_p} = \frac{\partial S}{\partial \Delta H} = \left(\frac{\partial^2 S}{\partial\left(\Delta H\right)^2}\right)\Big|_{H_0} \Delta H = \frac{\partial S}{\partial \Delta H}$$

Using Onsager's relations (equation 10) I relate the entalpy flux and force and arrive at a final expression for the change in entalpy over time:

$$\begin{aligned} &\frac{\partial \Delta H}{\partial t} = L \cdot \frac{\partial S}{\partial \Delta H} = -L\frac{\Delta H}{T^2 c_p} \quad \text{(1st order ODE)} \\ &\rightarrow \Delta H\left(t\right) = \Delta H_0 \exp\left(-\frac{L}{T^2 c_p} t\right) \end{aligned}$$

I see that the time scale of equilibration, or, 'relaxation time', is proportional to the heat capacity of the system:

$$\tau = \frac{T^2}{L} c_p \approx \frac{T^2}{L} \Delta c_p \quad \text{(at phase transition)} \qquad (14)$$

.

When the heat capacity increases during phase transition, the time it takes for the membrane to equilibrate fluctuations are similarly prolonged. The approximation is made on the basis that the entalpy changes related to melting are magnitudes larger than contributions from the reservoir. By looking at Monte Carlo simulations of relaxation processes, Mosgaard et al. [57] finds that when a system doesn't have the time to relax properly because it is pertubed at a frequency much faster than the time scale of the equilibrium relaxation, the process does not access the full heat capacity of the system as if the size of the reservoir was reduced. Thus the isentropic compressility is reduced and the speed of sound is coupled to frequency of sound, causing dispersion.

Returning to the soliton model in equation 1, it must be mentioned that the density of the membrane is described by the distance between lipid headgroups at the membrane surface, and the dimension perpendicular to the membrane surface is essentially ignored. The speed of sound depends on the *area* density, ρ^A, and isentropic *area* compressibility, κ_S^A, instead of volume density and isentropic volume compressibility. Currently a linear empirical relation for area and entalpy haven't been documented, $\langle \Delta A \rangle = \gamma_A \cdot \langle \Delta H \rangle$, since area is difficult to measure accurately alongside changes in entalpy, but it is generally assumed to exist [28], such that isentropic area compressibility can be related to the heat capacity of the system following the same line of argument as for the isentropic volume compressibility.

To simplify things, Heimburg and Jackson [33] leaves out the influence of dispersion in the speed of sound, c of the soliton model. The dispersion is instead taken care of by adding the term $-h\frac{\partial^4 \Delta\rho^A}{\delta x^4}$ and adjusting the parameter h. They fit a quadratic function to empirical measurements of the speed of sound as a function of density in the low frequency limit [33]:

$$c^2 = c_0^2 + p \cdot \Delta\rho^A + q \cdot \left(\Delta\rho^A\right)^2 + \mathcal{O}\left(\Delta\rho^A\right)^3 \tag{15}$$

$$\begin{aligned}\frac{\partial^2 \Delta\rho^A}{\partial t^2} &= \frac{\partial}{\partial x}\left(\left(c_0^2 + p \cdot \Delta\rho^A + q \cdot \left(\Delta\rho^A\right)^2\right)\frac{\partial \Delta\rho^A}{\partial x}\right) - h\frac{\partial^4 \Delta\rho^A}{\partial x^4} \\ &= \left(p + 2 \cdot q \cdot \Delta\rho^A\right)\left(\frac{\partial \Delta\rho^A}{\partial x}\right)^2 + \left(c_0^2 + p \cdot \Delta\rho^A + q\left(\Delta\rho^A\right)^2\right)\frac{\partial \Delta^2\rho^A}{\partial x^2} - h\frac{\partial^4 \Delta\rho^A}{\partial x^4}\end{aligned} \tag{16}$$

The coefficients in the following expression for the speed of sound is therefore dependent on the reservoir size of the experimental setup used. The fit coefficients used in the work presented in this thesis can be found in the list of table values in appendix D.

2.2.2 Modelling dispersion from thermodynamic properties

Though the constant dispersion coefficient h in equation 1 is shown to only influence the width of the pulse [33], it is still a free parameter. In order to achieve a self-contained, thermodynamical model, Mosgaard [56] has managed to express the dispersion in terms of thermodynamic variables of the system close to phase transition in the limit of small amplitude pertubations, drawing on many of the equations I've just introduced in this chapter. He expresses the new h in terms of limiting velocities,

$$\begin{aligned}\left(c_1(\rho)\right)^2 &\equiv \frac{1}{\rho^A \kappa_T^{A,0}} \quad \text{High frequency limit} \\ \left(c_2(\rho)\right)^2 &\equiv \frac{1}{\rho^A \frac{\gamma_A^2 T}{A}\Delta c_p} \quad \text{Component related to lipid melting} \\ \left(c_0(\rho)\right)^2 &\equiv \frac{1}{\left(c_1(\rho)\right)^{-2} + \left(c_2(\rho)\right)^{-2}} \quad \text{Low frequency limit}\end{aligned} \tag{17}$$

Here $\kappa_T^{A,0}$ is the isothermal area compressibility without contribution from membrane melting, and A is the area of the membrane. The new dispersion coefficient is then:

$$h = \Delta c_p^2 \left(\frac{T^2 \left(c_0\left(\rho\right)\right)^3}{2L}\right)^2 \left[\frac{4c_2^2 + 3c_1^2}{c_2^2(c_2^2 + c_1^2)}\right] \tag{18}$$

Where T is the temperature of the system and L is Onsager's phenomenological constant. For full derivation, see appendix A.1. Instead of deriving a relation between area density and $\kappa_T^{A,0}$, the high frequency speed of sound is approximated by fitting a quadratic function to experimental data similarly to the approximation of the low frequency speed of sound in equation 15 in the previous section:

$$(c_1(\rho))^2 \approx k \cdot c_0^2 + f \cdot \Delta\rho^A + g \cdot \left(\rho^A\right)^2 + \mathcal{O}\left(\rho^A\right)^3$$

These fit coefficients can also be found in the list of table values in appendix D.

For the dispersion coefficient to be exclusively dependent on density, ρ_A, the excess heat capacity must be expressed in terms of ρ_A. In the following I will derive the approximated relationship between heat capacity and membrane density that I've implemented in the numerical work of this thesis. To make the derivation less messy, I abbreviate ρ_A with ρ.

ρ, or more specifically, $\langle\rho(T)\rangle$, can be written as:

$$\langle\rho(T)\rangle = f(T) \cdot \langle\rho_{\text{fluid}}(T)\rangle + (1 - f(T))\langle\rho_{\text{gel}}(T)\rangle$$

Where f is the fluid fraction. Since the membrane is close to melting transition, the initial state of the nerve isn't likely to be completely fluid. If you add a small change in the fluid fraction, Δf, you get:

$$\begin{aligned}
&(f + \Delta f) \cdot \langle\rho_{\text{fluid}}(T)\rangle + (1 - (f + \Delta f))\,\langle\rho_{\text{gel}}(T)\rangle \\
&\quad = f \cdot \langle\rho_{\text{fluid}}(T)\rangle + (1 - f)\langle\rho_{\text{gel}}(T)\rangle + \Delta f\left(\langle\rho_{\text{fluid}}(T)\rangle - \langle\rho_{\text{gel}}(T)\rangle\right) \\
&\quad = \langle\rho(T)\rangle + \langle\Delta\rho(T)\rangle \quad \text{where} \quad \langle\Delta\rho(T)\rangle = \Delta f\left(\langle\rho_{\text{fluid}}(T)\rangle - \langle\rho_{\text{gel}}(T)\rangle\right) \\
&\quad \rightarrow \Delta f = \frac{\langle\Delta\rho(T)\rangle}{\langle\rho_{\text{fluid}}(T)\rangle - \langle\rho_{\text{gel}}(T)\rangle}
\end{aligned}$$

Assuming that the molar mass of a lipid doesn't change from fluid to gel phase (ex. due to changes in water content), the molar mass is constant, M.

Now the question is, can I express $\langle\rho_{\text{fluid}}(T)\rangle = \langle\frac{2\cdot M}{A_{\text{fluid}}(T)}\rangle$ as $= \frac{2\cdot M}{\langle A_{\text{fluid}}(T)\rangle}$ (factor of 2 due to BI-layer) and likewise for the gel-phase? If I Taylor-expand $\rho = \frac{2\cdot M}{A}$ around the mean, $\langle A\rangle$ I get:

$$\begin{aligned}
\rho &= 2 \cdot M\left(\frac{1}{\langle A\rangle} + \frac{-1}{\langle A\rangle^2} \cdot \Delta A + \mathcal{O}(\Delta A)^2\right) \\
\implies \rho &= \frac{2 \cdot M}{\langle A\rangle}\left(1 - \frac{\Delta A}{\langle A\rangle}\right)
\end{aligned}$$

As long as the relative change in area with respect to temperature is low for each type of lipid state, such that the majority of the area change is due to the change in distribution of fluid and gel lipids, this is approximately equal to $\rho = \frac{2M}{\langle A\rangle}$. Is this the case? Outside of transition there is an observed proportional relationship between temperature and relative area change in both fluid and gel phase, so the thermal area expansion coefficients are constant [28] and approximately equal to $\alpha_{\text{gel}} = 0.0026/\text{K}$, $\alpha_{\text{fluid}} = 0.0042/\text{K}$ for DPPC (MLV, [28]). Measuring the thermal area expansion coefficients inside of transition is much more difficult. In the case of volume expansion coefficients, the coefficients for pure lipid and gel phase are are a magnitude less than the largest coefficients during transition, $\alpha_{\text{gel}} = 0.00088/\text{K}$, $\alpha_{\text{gel}} = 0.001/\text{K}$ for DPPC (MLV, [28]) versus $\alpha_{\text{transition}} = 0.03/\text{K}$ (LUV, [27]) and $\alpha_{\text{transition}} = 0.17/\text{K}$ (MLV [27]), so the expansion of the pure phases are relatively small. Since the area change during phase transition in DPPC is estimated to be much larger than the volume change, 24.6% vs. 4.7% [33], I assume the area changes inside and outside transition to differ similarly or more. Furthermore the full melting transition is quite wide, including a small pre-transition called the 'ripple' phase several degrees below the main transition. There might therefore be contributions from lipid transitions in the experimentally determined thermal expansion coefficients for the fluid and gel phases, such that the true expansion of each phase is even smaller than expected. This argument is similar to the argument in the previous section of the pure lipid phase contribution to volume expansion close to phase transition.

I then have an expression for the change in fluid fraction, which I need in the following derivation of the average molar area:

First I use the same procedure as for the molar density to derive an expression for the molar area:

$$\begin{aligned}
\langle A(T)\rangle &= f \cdot \langle A_{\text{fluid}}(T)\rangle + (1-f)\langle A_{\text{gel}}(T)\rangle \\
&= (f+\Delta f)\cdot \langle A_{\text{fluid}}(T)\rangle + (1-(f+\Delta f))\,\langle A_{\text{gel}}(T)\rangle \\
&= (f\cdot\langle A_{\text{fluid}}(T)\rangle + (1-f)\langle A_{\text{gel}}(T)\rangle) + \Delta f\,(\langle A_{\text{fluid}}(T)\rangle - \langle A_{\text{gel}}(T)\rangle) \\
&= \langle A(T)\rangle + \langle \Delta A(T)\rangle \quad \text{where} \quad \langle \Delta A(T)\rangle = \Delta f\,(\langle A_{\text{fluid}}(T)\rangle - \langle A_{\text{gel}}(T)\rangle) \\
&\to \langle \Delta A(T)\rangle = \Delta f\,(\langle A_{\text{fluid}}(T)\rangle - \langle A_{\text{gel}}(T)\rangle)
\end{aligned}$$

Here I do not divide the area by a factor of 2 since the bilayer of the membrane is accounted for in the area. Then I insert the expression for the change in fluid fraction:

$$\begin{aligned}
\langle \Delta A(T)\rangle &= \frac{\langle \Delta\rho(T)\rangle}{2\cdot M}\cdot\left(\frac{1}{\langle A_{\text{fluid}}(T)\rangle} - \frac{1}{\langle A_{\text{gel}}(T)\rangle}\right)^{-1}(\langle A_{\text{fluid}}(T)\rangle - \langle A_{\text{gel}}(T)\rangle) \\
&= \frac{\langle \Delta\rho(T)\rangle}{2\cdot M}\cdot\left(\frac{\langle A_{\text{gel}}(T)\rangle - \langle A_{\text{fluid}}(T)\rangle}{\langle A_{\text{fluid}}(T)\rangle\cdot\langle A_{\text{gel}}(T)\rangle}\right)^{-1}(\langle A_{\text{fluid}}(T)\rangle - \langle A_{\text{gel}}(T)\rangle) \\
&= \frac{\langle \Delta\rho(T)\rangle}{2\cdot M}\cdot(-1)\cdot\langle A_{\text{fluid}}(T)\rangle\cdot\langle A_{\text{gel}}(T)\rangle
\end{aligned}$$

To determine $\langle A_{\text{fluid}}(T)\rangle$ and $\langle A_{\text{gel}}(T)\rangle$ I exploit the approximate proportionality between relative change in area and change in temperature (per Kelvin) described by the intrinsic thermal area expansion coefficient for gel and for fluid (denoted with X's).

$$\begin{aligned}
&\langle A_{\text{fluid}}(T)\rangle = [X_{A,\text{fluid}}\cdot\Delta T_1 + 1]\cdot\langle A_{\text{fluid}}(T_1)\rangle \ \text{ and,} \\
&\langle A_{\text{gel}}(T)\rangle = [X_{A,\text{gel}}\cdot\Delta T_2 + 1]\cdot\langle A_{\text{gel}}(T_2)\rangle
\end{aligned}$$

Here $\Delta T_1 = T - T_1$, where T_1 is the temperature far from transition where $\langle A_{\text{fluid},1}(T)\rangle$ is obtained, similarly with T_2.

Summarizing, I get:

$$\langle \Delta A(T)\rangle = \frac{\langle \Delta\rho(T)\rangle}{2\cdot M}\cdot(-1)\cdot[X_{A,\text{fluid}}\cdot\Delta T_1+1]\cdot\langle A_{\text{fluid}}(T_1)\rangle\cdot[X_{A,\text{gel}}\cdot\Delta T_2+1]\cdot\langle A_{\text{gel}}(T_2)\rangle \tag{19}$$

Following the argument from previous section, let us assume that area and entalpy fluctuations are proportional:

$$\langle \Delta A(T)\rangle = \gamma_A\langle \Delta H(T)\rangle \tag{20}$$

I model the melting transition as a discrete system with two possible states, each lipid is independent and identically distributed, and I have the probabilities:

$$P_{\text{fluid}} = \frac{K(T)}{K(T)+1}\quad\text{for a fluid lipid, and}\quad P_{\text{gel}} = \frac{1}{K(T)+1}\quad\text{for a gel lipid}$$
$$\text{where}\quad K(T) = \exp(-\frac{n\Delta G}{k_BT}) = \exp(-n\frac{\Delta H_0 - T\Delta S}{k_BT})$$

H_0 is the entalpy difference of the fluid (reference state) and gel state of a lipid, $H_{\text{fluid}} - H_{\text{gel}}$. For a sharp transition I can write $\Delta S = \frac{\Delta H_0}{T_m}$ (set $\Delta G = 0$ in K).

$\langle \Delta H(T)\rangle$ for one lipid is calculated by $\sum_i \Delta H_i P_i$. The reference state for the density change is the fluid state, so $\Delta H = -\Delta H_0$ for gel, $\Delta H = 0$ for fluid:

$$\langle \Delta H(T)_{\text{lipid}}\rangle = -\Delta H_0\cdot\frac{1}{K(T)+1} + 0\cdot\frac{K(T)}{K(T)+1} \tag{21}$$

By plugging formula 19 into formula 20 and subsequently into formula 21 I can obtain the temperature of the system, T:

$$\begin{aligned}\text{solve}\Big(-\gamma_A\cdot\Delta H_0\cdot\frac{1}{\exp(\frac{n\Delta H_0}{k_B}(\frac{1}{T}-\frac{1}{T_m}))+1} &= \frac{\langle\Delta\rho(T)\rangle}{2\cdot M}\cdot(-1)\\ \cdot[X_{A,\text{fluid}}\cdot\Delta T_1+1]\cdot\langle A_{\text{fluid}}(T_1)\rangle\cdot[X_{A,\text{gel}}\cdot\Delta T_2+1]&\cdot\langle A_{\text{gel}}(T_2)\rangle,\, T\Big)\end{aligned} \tag{22}$$

This equation can be solved numerically using MATLAB®'s variable precision solver. The temperature can then be inserted into the following equation based on Van't Hoffs law [29, chap. 6.2] to obtain the single lipid contribution to the excess heat capacity:

$$
\begin{aligned}
\Delta c_{\mathrm{p,lipid}} &= \left(\frac{d\langle \Delta H(T)_{\mathrm{lipid}}\rangle}{dT}\right)_P \\
&= -\Delta H_0 \cdot \frac{-K(T)}{(K(T)+1)^2} \cdot \frac{n\Delta H_0}{k_B T^2} \\
&= \frac{n(\Delta H_0)^2}{k_B T^2} \frac{K(T)}{(K(T)+1)^2}
\end{aligned} \tag{23}
$$

The heat capacity of a mole of lipids is then Avogadro's number times the lipid heat capacity, $N_A \cdot \Delta c_{\mathrm{p,lipid}}$. It should be mentioned that the cooperativity, n, has a rather strange unit since it is expressed for a single lipid. It is the cooperativity of a single lipid as if it were 'in a mole of lipids'.

2.2.3 Membrane permeability and electricity

So how does the soliton theory connect density pulses to the other processes taking place during nerve signaling, as mentioned in section 2.1? To give a few examples, I will briefly go through the case of voltage changes and and altered permeability.

Density change is coupled to the change in trans-membrane voltage and capacitance through the alteration of the charge density of the bilayer. The electrical potential of a monolayer is approximately proportional to its charge density, $\Psi \propto \sigma$, which then is proportional to area density, $\sigma \propto \rho_A$ [29, chap. 18.3.2.2]. The charge density covers lipids, proteins and other charged constituents of the membrane and can be strongly modulated by ions in the environment.

In case of a constant dispersion coefficient, the soliton model predicts an $\approx 20\%$ increase in area density in case of a pure DPPC bilayer [33], transitioning from fluid to solid and going through 85% of the phase transition [29, chap. 18.3.2.2]. Since area density is proportional to the charge density, one would expect a 20% increase in charge density as well, which in case of charged membranes can have a considerable effect on membrane potential [29, chap. 18.3.2.2]. To illustrate this, Heimburg and Jackson measured the voltage and area changes of charged lipid bilayers undergoing phase transition. The outer monolayer consisted purely of uncharged lipids, whereas the inner monolayer contained 40% charged lipids. This asymmetrical charge distribution is common for biological membranes such as mitochondria [36]. They measured ≈ 50 mV voltage changes across the membrane, which is in the same order of magnitude as action potentials [29, chap. 18.3.2.2].

The relative change in membrane thickness during phase transition is also quite large. As an example, the thickness of pure DPPC bilayer decreases $\approx 16\%$. Furthermore area changes ≈ 24.6 % [29, chap. 18.3.2.2]. If the nerve membrane is approximated as a simple plate capacitor, the capacitance is directly proportional to area and inversely proportional to the distance between charges on either side of the membrane: $C \propto \frac{A}{d}$. Changes in membrane capacity is therefore a natural consequence of the altered area and thickness during the nerve pulse.

Increases in permeability during nerve pulses are likewise related to the impact of phase transition on the membrane. Permeability is linearly related to isothermal compressibility, κ_T [29, chap. 17], and isothermal compressibility follows the heat capacity, also in a linear manner (see equation 5). Thus phase transition should have a strong effect on membrane permeability, which is affirmed by empricial studies [29, chap. 17].

According to Heimburg [30], sodium permeability increased factor 100 in DPPG and DPPC membranes near phase transition [30], matching the 100-fold increase in permeability of sodium during the nerve signal reported by Abbott et al. [1]. It is therefore likely that the altered membrane permeability during the nerve pulse can be fully explained by soliton induced lipid phase transition. Antonov et al. [3] has among others shown how phase transition can induce discrete, ion-channel-like events in pure lipid membranes. It is unlikely for strongly hydrated ions to diffuse directly through the hydrophobic lipid core, instead one can imagine the spontaneous formation of lipid pores due to increased thermal motion at phase transition [29, chap. 17], exacerbated by the presence of domain interfaces, protein clusters and other defects in the lipid bilayer.

2.3 Summary

In the previous chapter I have briefly listed some of the characteristics of the nerve pulse such as electrical, mechanical and permeability changes. I have given a short introduction to the thermodynamics of sound propagation in lipid membranes and derived an expression for the dispersion of the soliton model through the thermodynamic quantities adiabatic compressibility and heat capacity. These results will be applied in the following chapters where I present my work.

3 Numerical endeavors

The initial objective of this thesis was to solve the self-contained soliton equation presented in the previous chapter for as wide a range of propagation speeds as possible. This version has been documented to be difficult to solve by [61, chap. 4.5] (and by Heimburg and Mosgaard though not in print). I intended to approach the problem in steps, first solving the cases of constant dispersion coefficient (in two steps), then an approximation to the density dependent problem (Gaussian approximation for density-dependent heat capacity) and lastly the self-contained model itself (thermodynamic model for density-dependent heat capacity). In reality the process was a lot more muddled, but I will nevertheless present it in this order for clarity's sake.

3.1 Solving ordinary differential equations in MATLAB®

All numerical work has been done using MATLAB® tools. I saw no reason to spend time trying to implement an algorithm on my own before knowing the problem better. MATLAB is one of the most widely used if not *the* most widely used environment for solving differential equations numerically [26, chap. 1.4.2]. It has an extensive catalog of powerful solving procedures based on well-established algorithms, with the caveat that one has to go through the source code to gain insight into their actual implementation. The major obstacles I encountered while working on the self-contained soliton equation were:

Stiffness: Typically occurs when there is a mismatch between time scales of the different terms in the differential equation [26, chap. 9.3.4] such that there comes a point where a slowly varying solution is overpowered by divergence and 'runs wild'. Very small changes in initial conditions or error can lead the algorithm from a well-behaved solution into an adjacent, divergent solution. In this case the density dependent problems show stiffness by presenting with smooth, well-behaving solutions within a limited variable interval, but for certain points tiny changes in initial conditions or numerical error can introduce large variations in the solution behaviour and manner of divergence (e.g. divergence from - ∞ to + ∞, see figure 7). This is very demanding for the ODE solver that needs to recognize these problematic areas and implement algorithms of increased accuracy. Non-stiff numerical methods involving adaptive step sizes usually get stuck at these points. I tried patching several methods together in order to tailor the numerical method to the problem, but the intermediary steps of the ODE solvers are not easily accessible to the user, so it led to quite a loss of information at the intersection of the two methods.

Solving the heat capacity: For intermediate or true steps of solving the self-contained problem, the ODE-solver needs to obtain the heat capacity using a numerical solver.

Systematic storage and retrieval of data: Due to the nature of the initial condition optimization, large amounts of data are generated. Because of the stiffness of the problem, it is very important to store information in a way least prone to machine error. MATLAB® has its own '.mat' files for this purpose.

Optimizing for initial conditions: I do not have adequate information to obtain a soliton-like solution up front in any of the cases.

The soliton equation is a 4th order PDE both in the case of constant and density-dependent dispersion coefficient. I followed the footsteps of Heimburg and Jackson [33] and transformed it into an ODE by changing the coordinate system to follow the pulse, $z = x - vt$:

$$v^2 \frac{\partial^2 \Delta\rho^A}{\partial z^2} = \frac{\partial}{\partial z}\left(\left(c_0^2 + p \cdot \Delta\rho^A + q \cdot \left(\Delta\rho^A\right)^2\right)\frac{\partial \Delta\rho^A}{\partial z}\right) - h\frac{\partial^4 \Delta\rho^A}{\partial z^4} \tag{24}$$

$$= \left(p + 2 \cdot q \cdot \Delta\rho^A\right)\left(\frac{\partial \Delta\rho^A}{\partial z}\right)^2 + \left(c_0^2 + p \cdot \Delta\rho^A + q\left(\Delta\rho^A\right)^2\right)\frac{\partial \Delta^2\rho^A}{\partial z^2} - h\frac{\partial^4 \Delta\rho^A}{\partial z^4} \tag{25}$$

I've tried different MATLAB®'s ODE-solvers for each problem to see which works the best. The following list includes the solvers I chose to work with:

ODE solvers:

All solvers incorporate multistep methods starting from a single initial point.

ODE 15s MATLAB®'s ODE15s is a multistep method using variable order numerical differential formulas (1st - 5th order) and adaptive step sizes [51]. That is the rather non-descriptive information given in the MATLAB documentation, I did not take the time to read the source code and cannot disclose more information. It is the only solver of theirs by which I was able to obtain solutions for the self-contained model, and concurrently their stiff solver of highest accuracy [50].

ODE 23tb For the Gaussian approximation of the heat capacity I have worked with ODE23tb [52] in parallel to ODE15s. It is a method of two stages: An implicit Runge-Kutta with trapezoidal rule governed step size in the beginning and later a switch to some second order backwards differentation method when it sees fit.

ODE 45 The constant dispersion coefficient cases did not have these difficulties, here the classical, explicit Runge-Kutta (4,5) with adaptive step sizes (ODE45, [53]) did well, alongside the stiff equation solvers ODE15s and ODE 23tb.

3.1.1 Analytical integration of the soliton model in case of a constant dispersion coefficient

I did not implement the constant dispersion coefficient case directly (a 4th order ODE). Instead I integrated the equation twice and thrice analytically before I solved the resulting 2nd and 1st order equations numerically. I'll briefly go through the analytical preparations underneath, which can also be found in Lautrup et al. [44] and Villagran-Vargas et al. [83]. While integrating I impose conditions such that the original 4th order problem requiring 4 initial conditions transforms into a 2nd or 1st order problem requiring 2 and 1 initial condition(s), respectively.

Integrating equation 24 from the previous section once, I get:

$$\begin{aligned}
&\int v^2 \frac{\partial^2 \Delta\rho^A}{\partial z^2} dz = \int \frac{\partial}{\partial z}\left[\left(c_0^2 + p \cdot \Delta\rho^A + q(\Delta\rho^A)^2\right)\frac{\partial \Delta\rho^A}{\partial z}\right] dz - h \int \frac{\partial^4 \Delta\rho^A}{\partial z^4} \\
&\to v^2 \frac{\partial \Delta\rho^A}{\partial z} = \left(c_0^2 + p \cdot \Delta\rho^A + q(\Delta\rho^A)^2\right)\frac{\partial \Delta\rho^A}{\partial z} + h \frac{\partial^3 \Delta\rho^A}{\partial z^3} + K_1
\end{aligned}$$

K_1 can be set to 0 since I assume $\Delta\rho^A$ and its derivatives to tend to 0 at $\pm\infty$ [83]. Integrating once more, I get:

$$\begin{aligned}
&\int v^2 \frac{\partial \Delta\rho^A}{\partial z} dz = \int \left(c_0^2 + p \cdot \Delta\rho^A + q(\Delta\rho^A)^2\right)\frac{\partial \Delta\rho^A}{\partial z} dz + \int h \frac{\partial^3 \Delta\rho^A}{\partial z^3} dz \\
&\to v^2 \Delta\rho^A = c_0^2 \Delta\rho^A + \frac{p}{2}(\Delta\rho^A)^2 + \frac{q}{3}(\Delta\rho^A)^3 - h \frac{\partial^2 \Delta\rho^A}{\partial z^2} + K_2 \\
&\to h \frac{\partial^2 \Delta\rho^A}{\partial z^2} = \left(c_0^2 - v^2\right)\Delta\rho^A + \frac{p}{2}(\Delta\rho^A)^2 + \frac{q}{3}(\Delta\rho^A)^3 \qquad (26)
\end{aligned}$$

Likewise K_2 is set to 0. If $K_2 > 0$, the solutions become periodic [83]. Multiplying with $\frac{\partial \Delta\rho^A}{\partial z}$ and integrating leaves:

$$\begin{aligned}
\int v^2 \Delta\rho^A \frac{\partial \Delta\rho^A}{\partial z} dz =& \int \left(c_0^2 \Delta\rho^A + \frac{p}{2}(\Delta\rho^A)^2 + \frac{q}{3}(\Delta\rho^A)^3\right)\frac{\partial \Delta\rho^A}{\partial z} dz \\
&- h \int \frac{\partial^2 \Delta\rho^A}{\partial z^2}\frac{\partial \Delta\rho^A}{\partial z} dz + \int K_2 \frac{\partial \Delta\rho^A}{\partial z} dz \\
\to \frac{v^2}{2}(\Delta\rho^A)^2 =& \frac{1}{2}c_0^2(\Delta\rho^A)^2 + \frac{p}{2 \cdot 3}(\Delta\rho^A)^3 + \frac{q}{3 \cdot 4}(\Delta\rho^A)^4 - \frac{h}{2}\left(\frac{\partial \Delta\rho^A}{\partial z}\right)^2 + K_3 \\
\to h \left(\frac{\partial \Delta\rho^A}{\partial z}\right)^2 =& \left(c_0^2 - v^2\right)\Delta\rho^A + \frac{p}{3}\left(\Delta\rho^A\right)^3 + \frac{q}{6}\left(\Delta\rho^A\right)^4 \qquad (27)
\end{aligned}$$

K_3 is set to zero too, though it has no influence on the solutions [83].

The equation of 1st order has a squared derivative, meaning the solution is constrained to be either even or odd. This was not the case for the second order equation. None of the equations above has an explicit dependence on z, see appendix A.2. Since this is the case for the density-dependent dispersion coefficient equation as well, I center all solutions around zero by locating the maximum amplitude of the non-centered solitons and translating the z-axis. Lautrup et al. [44] has found an analytical solution to the soliton model in the case of constant dispersion coefficient. This is convenient, since I can compare shape and behaviour of my solutions to the true soliton for different ODE-solvers.

3.1.2 Linearization and ODE solvers

The ODE solvers of MATLAB® require the differential equation to be expressed as a first order linear system [50] . A brief description of this can be found in Heath [26, chap. 9]. Initial conditions are then expressed as a vector of the system. The number of initial conditions I need to specify is then equal to the size of the linearized system.

For the twice integrated constant dispersion coefficient equation, the linearization results in a system of size 2:

$$\mathbf{U} = \begin{bmatrix} y \\ y' \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad \text{and} \quad \mathbf{U}' = \begin{bmatrix} y' \\ y'' \end{bmatrix} = \begin{bmatrix} u_2 \\ g(u_1, u_2) \end{bmatrix} \tag{28}$$

Where the density change y is differentiated with respect to $z = x - vt$, and g is the equation describing the system (adapted from equation 26 in the previous section):

$$y'' = h^{-1}\left((c_0^2 - v^2)y + \frac{p}{2}y^2 + \frac{q}{3}y^3\right) \to g(u_1, u_2) = h^{-1}\left((c_0^2 - v^2)u_1 + \frac{p}{2}(u_1)^2 + \frac{q}{3}(u_1)^3\right)$$

For the thrice integrated equation, I just have a single equation:

$$\mathbf{U} = y \quad \text{and} \quad \mathbf{U}' = y' = g(u_1) \tag{29}$$

The system is described by the following equation (adapted from equation 27 in the previous section):

$$g(u_1) = \begin{cases} \sqrt{h^{-1}\left((c_0^2 - v^2)u_1^2 + \frac{p}{3}u_1^3 + \frac{q}{6}u_1^4\right)} & \text{if } z \leq 0 \\ -\sqrt{h^{-1}\left((c_0^2 - v^2)u_1^2 + \frac{p}{3}u_1^3 + \frac{q}{6}u_1^4\right)} & \text{if } z > 0 \end{cases} \tag{30}$$

After taking the square root, I need to consider the sign of the derivative. A symmetrical derivative leads to sigmoid solutions, whereas alternating signs leads to pulses. Since I want the pulse to be positive and centered around $z = 0$, I set $y' > 0$ below $z = 0$ and vice versa.

For the density dependent dispersion coefficient the system is of size four:

$$\mathbf{U} = \begin{bmatrix} y \\ y' \\ y'' \\ y''' \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \quad \text{and} \quad \mathbf{U}' = \begin{bmatrix} y' \\ y'' \\ y''' \\ y'''' \end{bmatrix} = \begin{bmatrix} u_2 \\ u_3 \\ u_4 \\ g(u_1, u_2, u_3) \end{bmatrix} \tag{31}$$

Where g is now including the density dependent dispersion coefficient term $h(y) = h(u_1)$ from equation 18. The system equation is adapted from equation 25 from the previous section:

$$g(u_1, u_2, u_3) = h(u_1)^{-1} \left(\left(c_0^2 - v^2 + p \cdot u_1 + q \cdot (u_1)^2 \right) \cdot u_3 + (p + 2 \cdot q \cdot u_1) \cdot (u_2)^2 \right)$$

3.1.3 Solving the heat capacity

For the case of the Gaussian approximation of the density dependent excess heat capacity, I estimated the parameters for a scaled Gaussian from sonicated LUV DPPC data in Mosgaard et al. [58] using guides in GIMP as rulers. The result can be seen in fig. 1. A Gaussian shape was chosen simply because the data was bell shaped and parameters can be easily determined. The non-zero intersection with the y-axis indicates that the lipids are partly in transition in the membrane's ground state, so that the the initial state of the membrane isn't fully fluid. One has to bear in mind that the direction of the transition in a plot of heat capacity versus density change is flipped compared to the usual plot of heat capacity versus temperature. The membrane is in a fluid state at low density changes to the left and solid at large density changes to the right.

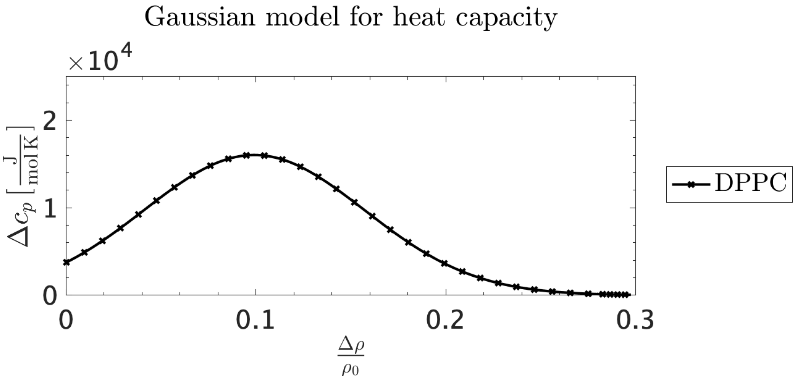


Figure 1: Heat capacity as a Gaussian function of relative density change based on sonicated LUV DPPC data from Mosgaard et al. [58].

As for the thermodynamic model for the heat capacity I needed to solve equations equations 23 and 22 in section 2.2.2. Since the table values for the average areas are per lipid and not per mole, I calculate everything per lipid until the end, where I scale the system to per mole. I could not find an analytical solution to the heat capacity equations 23 and 22. Attempts with symbolic solvers in MATLAB®, Maple and Wolfram-Alpha were unsuccesful. MATLAB's numerical solver fsolve (floating point arithmetic) also didn't return a solution. MATLAB's vpasolve (variable precision arithmetic) yielded results but is slow and not always succesful. Since I incorporated vpasolve into the input of the ODE solver, the latter would crash if vpasolve returned empty-handed. To prevent loss of information I encapsuled the ODE solver in a 'try-catch' maneuvre, saving the workspace for each run of the ODE solver, succesful or not, alongside a log of error messages (the messages I had implemented myself, that is). A log file is generated in txt-format in the same directory as the data, documenting the approximate initial condition of the problem and whether the solving was succesful or not. This feature cannot be turned off since it has many dependants in the rest of the code, and unfortunately it can accumulate a lot of data.

In figure 2 is a plot of the calculated heat capacity curves for DPPC and e. coli membranes as a function of both relative density change and system temperature. The parameters for the DPPC curve are included in appendix D. I mainly use table values from Heimburg [28]. The cooperativity $n = 80$ is a choice I made since I can obtain solutions for a decent range of velocities for this value. In the case of e. coli the cooperativity n, melting entalpy, ΔH_0, and melting temperature, T_m, were obtained using a Van't Hoff fit (see equation 23) of e. coli mealting profiles from Mužić et al. [60, fig.3], these parameters are also included in appendix D. The rest of the parameters for the e. coli run were set to DPPC values. In figure 2 the melting profile of DPPC membranes matches fairly well with melting profiles of extruded DPPC LUVs in literature [28, 22], so the choice of $n = 80$ might not be too unrealistic. The widths of the curves of the self-contained model are smaller, and the heights larger, than the Gaussian approximation based on sonicated LUVs, which corresponds well to extruded DPPC LUVs having more cooperative transitions than sonicated LUV's [28]. Because of this the thermodynamically modelled heat capacity might not be directly comparable to the Gaussian approximation.

The DPPC transition peak is approximately 12 times larger than peak of the e. coli membrane. Generally the cooperativities of pure lipid membranes are much higher than in mixed membranes [23] and biological membranes [60]. The structure of the vesicles can also alter the cooperativity greatly, a good example is the differences between melting peaks of MLVs, extruded LUVs, sonicated LUVs of DPPC seen in the 1998 paper by Heimburg [28, fig. 2].

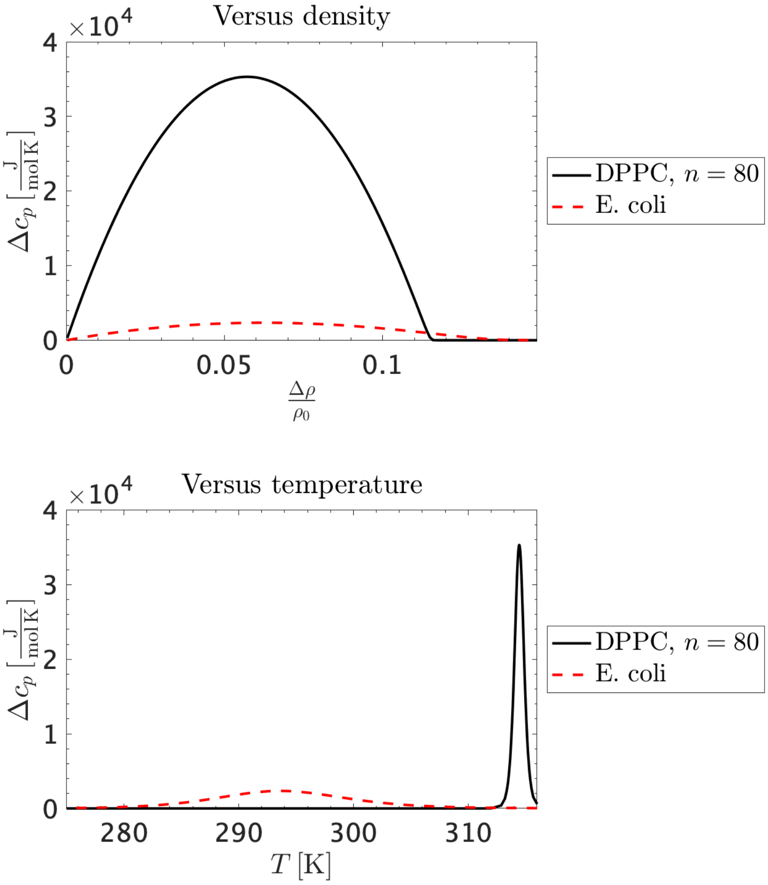


Figure 2: Calculated heat capacity curves as a function of relative change in density and of temperature for DPPC and e.coli membranes.

Finally I calculated the density-dependent dispersion coefficient for DPPC using both the Gaussian approximation and the thermodynamical model of the heat capacity for different values of the phenomenological constant L. The results are displayed in figure 3. The Gaussian approximation yields dispersion coefficient curves with similar size and shape as previous results by Panagiotis [61, chap. 4.4, fig. 4.3], though my curves do not intersect the y-axis at zero, which is a reflection of the non-zero excess heat capacity of the ground state, see figure 1. The curves of the thermodymic model are noticeably taller and does not have any intersection worth mentioning, due to the increased cooperativity. The abrupt increment is likely contributing to the stiffness of the problem.

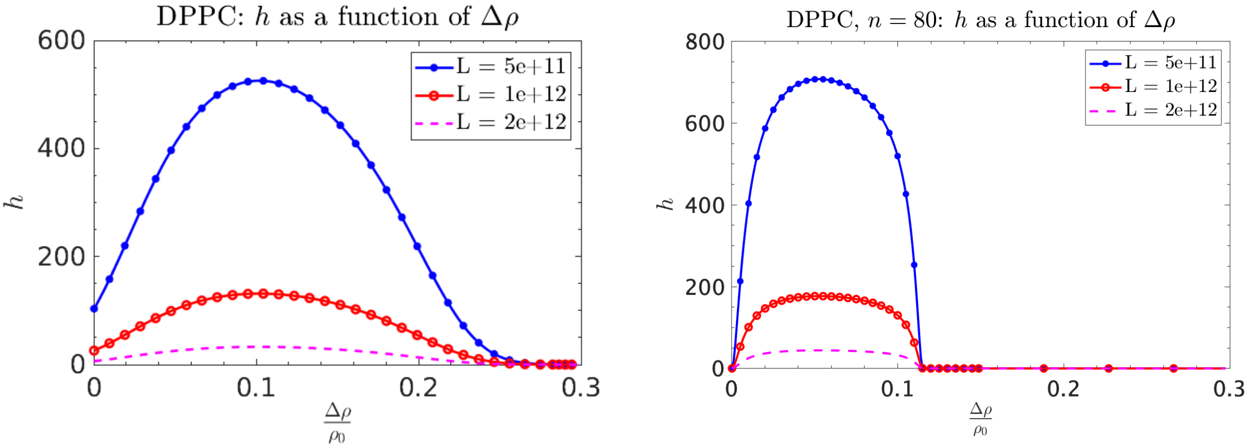


Figure 3: Dispersion coefficients as a function of relative density. **LEFT:** Gaussian approximation of density-dependent heat capacity. **RIGHT:** Thermodynamic model for density-dependent heat capacity (bottom).

3.1.4 The initial value problem

At this point I have described the full implementation of the 4 different ODEs in MATLAB®. What remains is to find initial conditions that produces a single, localized pulse. I decided not to complicate things more than necessary, so my methods are brute force and based on observation.

For each problem I start out by producing solutions for a broad range of initial conditions and manually deciding what features indicate that a solution is close to solitary behavior. I then create a procedure that calculates all solutions for a manually defined interval of initial values and automatically picks out the solutions closest to solitary behavior. If it manages to find two solutions whose initial conditions presumably constitutes an interval around the initial condition of a soliton, the procedure stores their associated initial conditions and further minimizes the interval using the bisection method. If only one boundary of the interval can be determined, the procedure determines in which direction to search for the second. If it is succesful, the interval of initial values is subsequently minimized likewise using the bisection method. As previously mentioned, vpasolve causes the ODE-solver to crash quite often, so for each step in the bisection method the bisection interval is divided into 10 steps, and the procedure searches for a solution among these steps before assuming no solution can be found and stopping the optimization. The steps alternate across the middle of the initial value interval in order to preserve as much of the bisection as possible.

Each case of constant dispersion coefficients have their individual optimization procedure, whereas the same optimization procedure is used for both of the density dependent cases. In the following I will briefly go through the initial values and the optimization criteria.

The case of constant dispersion coefficients, twice integrated (equation 28):
I start the solver from what I assume is the maximum of the density amplitude, since the maximum point is easy to estimate. The caveat of this strategy is in case of flat amplitudes, such as the ones obtained by Lautrup et al. [44, fig. 1], the solution changes too little for the precision of the algorithm. It has not been a major issue here. The system is size 2, see equation 28, but since the second derivative at maximum is zero always, I effectively have only 1 initial value to define, which is the amplitude, $\Delta\rho_{\max}$:

$$\mathbf{U_0} = \begin{bmatrix} y_0 \\ y_0' \end{bmatrix} = \begin{bmatrix} \Delta\rho_{\max} \\ 0 \end{bmatrix}$$

In Heimburg and Jackson [33] all the solutions have a maximum relative density change around $\frac{\Delta\rho_{\max}}{\rho_0} = [0.04, 0.20]$, a depending on the velocity in question. This is the interval I used for optimization.

Figure 4 is a representation of the two types of solutions I get from the twice integrated constant dispersion coefficient case when iterated from the maximum. Too small amplitudes leads to oscillatory solutions, whereas too large amplitudes causes the solution to diverge. The largest amplitude of the oscillatory solutions and the smallest amplitude of the divergent solutions constitutes lower and upper boundary for a soliton solution. I have included the full optimization scheme in appendix F.7.

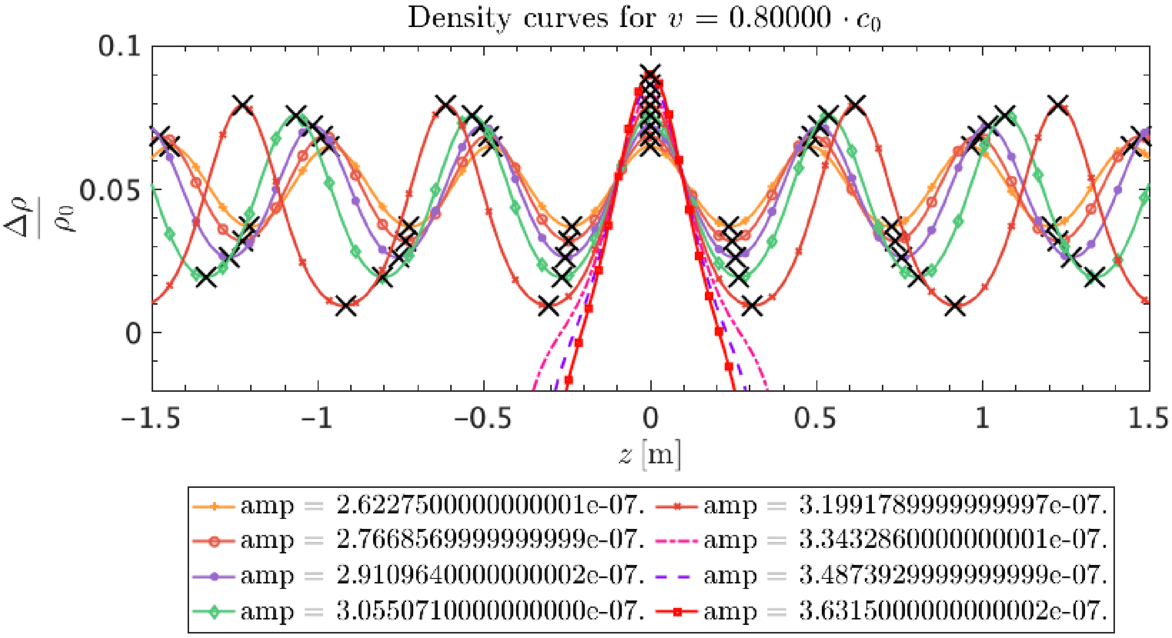


Figure 4: Range of solutions to illustrate the twice integrated bisection method, ODE23tb, $h = 50\frac{\text{m}^4}{\text{s}^2}$. The upper boundary for bisection would for this velocity be an amplitude of ≈ 3.20 and the lower amplitude ≈ 3.34.

The case of constant dispersion coefficient, thrice integrated (equation 29):

In this case the system is size 1, see equation 29, so I only need to define the amplitude, $\Delta\rho_{\max}$, start my iteration:

$$\mathbf{U_0} = y_0 = \Delta\rho_{\max}$$

I search for solitons using the same range of amplitudes as for the twice integrated case of constant dispersion coefficient. In figure 5 we see a different picture than in figure 4. The small amplitudes now lead to localized pulses that are only distinguishable from the localized pulses at larger amplitudes by the lack of oscillating tails. The oscillations are the real part of a complex oscillation created by the square root in the system, see equation 29. The bisection interval is upwards bounded by the smallest amplitude with oscillating tails, while the adjacent lower amplitude constitutes the lower bound.

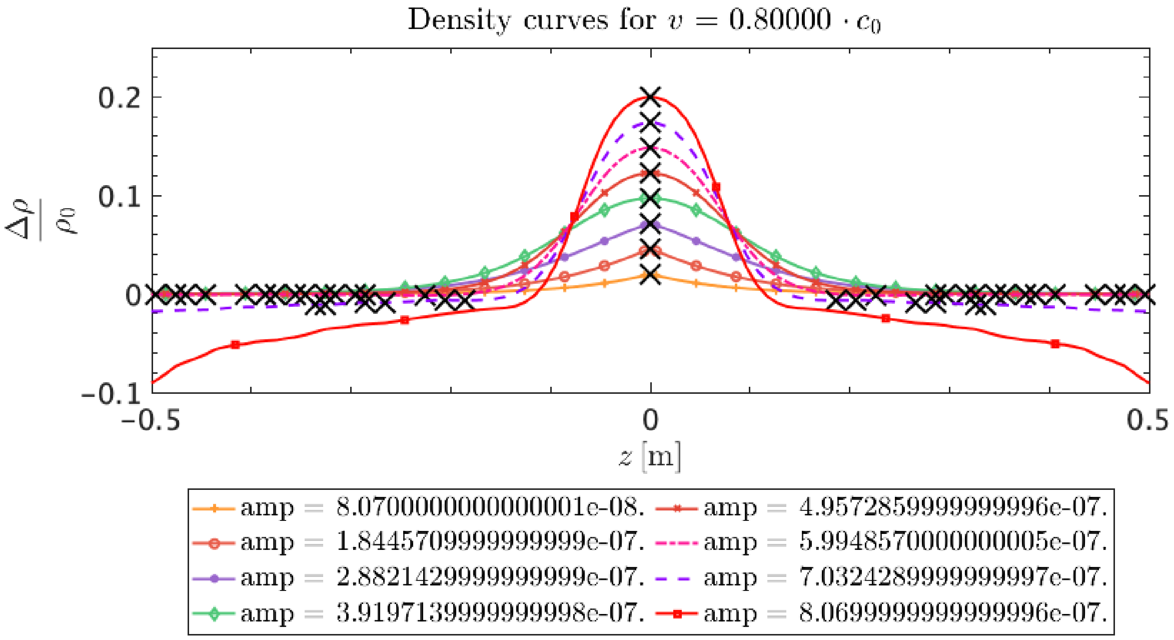


Figure 5: Range of solutions to illustrate the thrice integrated bisection method, ODE23tb, $h = 50\frac{\mathrm{m}^4}{\mathrm{s}^2}$. The oscillating tails of the high amplitude solutions are not easy to tell from the straight tails of lower amplitude, here the black crosses indicate minima and maxima does exist. One sees how increasing the amplitude causes the ends to diverge downwards (amplitude 7.03, 8.07). The divergent tails also oscillate, though not depicted here.

Density dependent dispersion coefficient cases (equation 31):

The density dependent dispersion coeffcicient cases are systems of size 4 and therefore needs 1st, 2nd and 3d derivative defined in addition to the density. I do not know the size of the third and fourth derivative at the maximum. What I do know is that I want the solution to be a localized pulse. If I assume the soliton is exponentially localized as in the original paper by Heimburg and Jackson [33], I can start the solver at the asymptote, which is modelled as an increasing exponential. If we write the exponential as $f(x) = \exp(a \cdot x)$, the general expression for the n'th derivative is given by $f^{(n)}(x) = a^n \exp(a \cdot x) = a^n f(x)$, where a in this thesis is denoted 'the amplitude' of the exponential (This amplitude is different from the amplitude of the solitons). As long as I am on an exponential tail, I can express all derivatives of the initial conditions in the terms of amplitude and initial density:

$$\mathbf{U_0} = \begin{bmatrix} y_0 \\ y_0' \\ y_0'' \\ y_0''' \end{bmatrix} = \begin{bmatrix} \Delta\rho_{\text{end}} \\ a \cdot \Delta\rho_{\text{end}} \\ a^2 \Delta\rho_{\text{end}} \\ a^3 \Delta\rho_{\text{end}} \end{bmatrix}$$

To make sure I stay as far out on the asymptote as possible, I am keeping $\Delta\rho_{\text{end}}$ at a minimum, $10^{-2.2} \cdot \Delta\rho_0$

as default. I optimize boundary conditions by adjusting the amplitude, a.

In figure 6 and figure 7 I've included representative ranges of solutions for the self-contained model and the density dependent dispersion coefficient with a Gaussian approximation for the heat capacity. Both showed a wider range of behavior that was less consistent than the constant dispersion coefficient cases, which is illustrated in the Gaussian approximation case, see figure 6. It seemed that all the interesting behavior is concentrated around an area of negatively diverging solutions with a single maximum. I dubbed this area 'The valley' due to the behavior of its maximum (see figure 7). On either side of the valley is an area showing potential for a localized solution, where the oscillating, upwards divergent solutions meet the single maximum, downwards divergent solutions of the valley. These sides are referred to as 'left' and 'right' side, corresponding to large and small amplitudes.

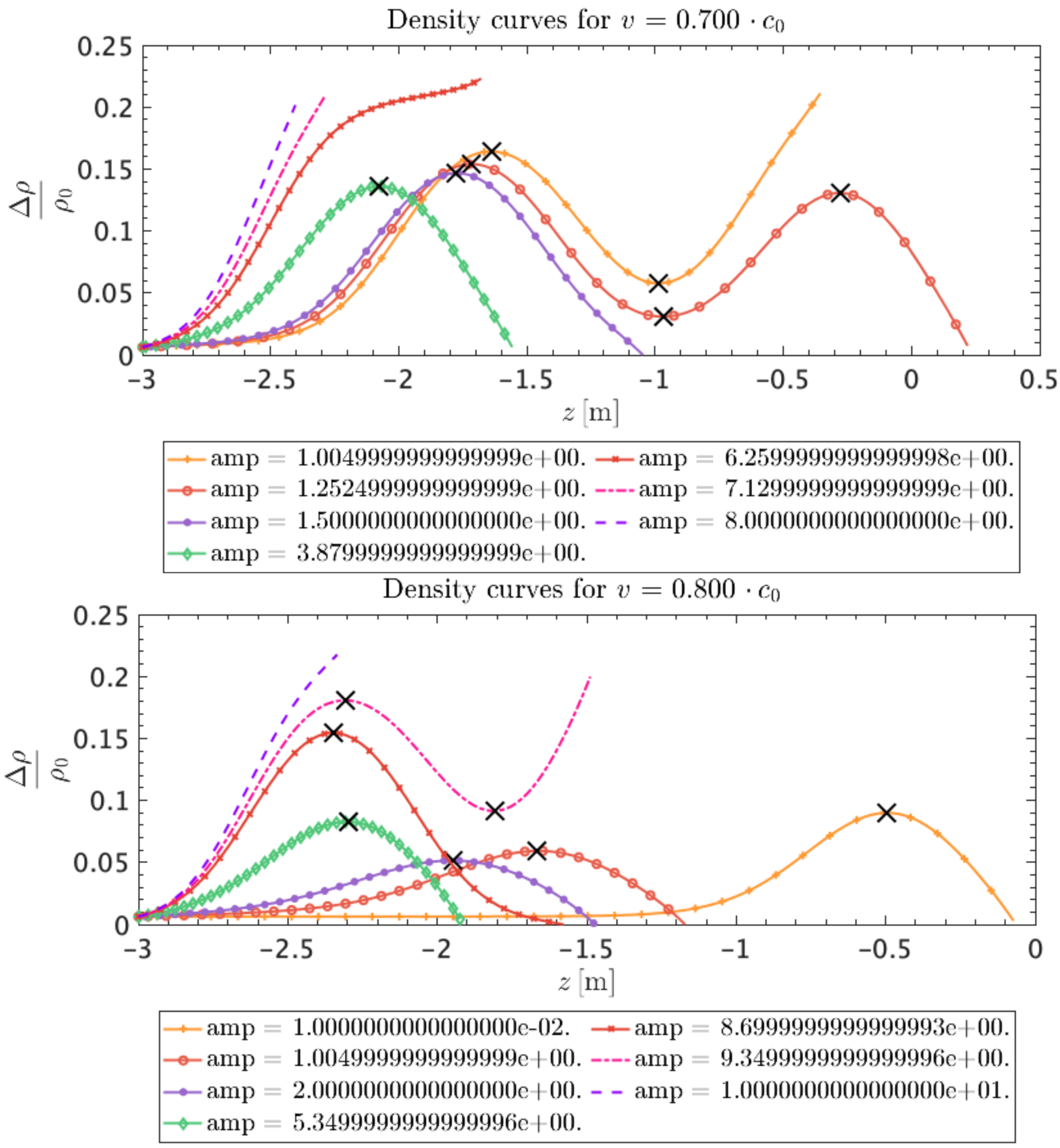


Figure 6: Range of solutions to illustrate the Gaussian bisection method, ODE15s. **Example of inconsistent behaviour:** For $0.70\,c_0$ (top), the valley, left and right sides can all be determined. In the case of $0.80\,c_0$ (bottom), right side oscillatory solutions has not been found. These are not found by further optimization either, and the final result in fig. 13 does not have a right side boundary.

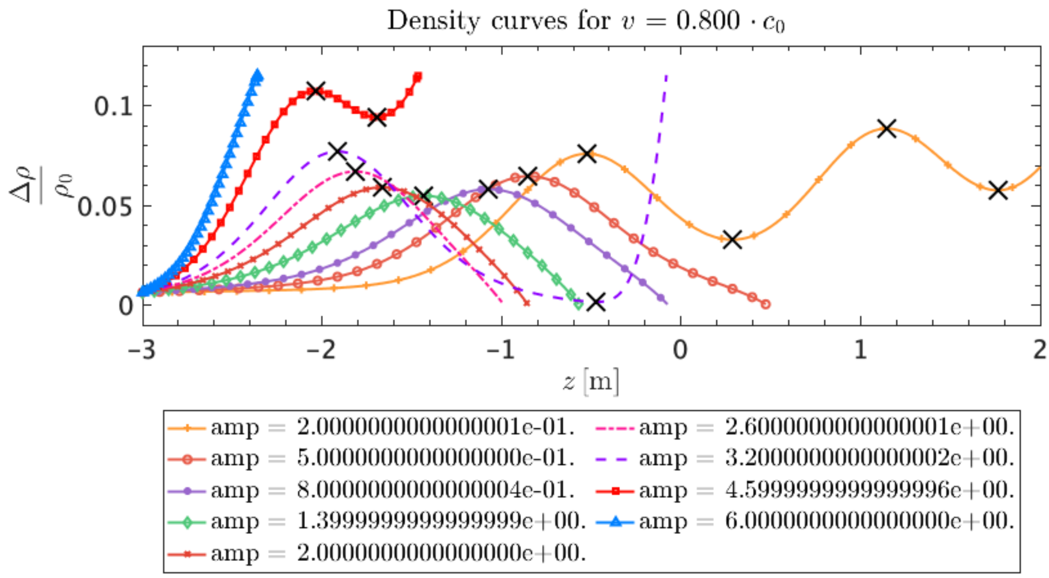


Figure 7: Range of solutions to illustrate the thermodynamic bisection method, ODE15s, cooperativity = 80. The left side is here presented by amplitudes $\in [3.2, 6]$, the valley by amplitudes $\in [0.5, 2.6]$ and the right side for amplitude < 0.5. If one looks at the first maximum of each solution, it obtains a minimum value inside of the valley, hence the name.

3.2 Results

Solutions differ depending on the ODE-solver used, and therefore the initial values are specific to the ODE-solver. The differences in solutions can be quite large. In appendix E.1 I've included a comparison of three different ODE-solvers solving the case of the thrice integrated soliton model with a constant dispersion coefficient. The solutions are similar at very high velocities, $\gtrapprox 0.95 \cdot c_0$ (figure 35) and large dispersion coefficients, $h \gtrapprox 50 \frac{\mathrm{m}^4}{\mathrm{s}^2}$ (figure 36). In order to be able to compare results I stick to the same ODE-solver. For the constant dispersion coefficient I use ODE23tb unless otherwise notified and for the density dependent dispersion coefficient ODE15s proves to be the best.

I solve each dispersion case for a handful of velocities in the range $[0.65 \cdot c_0, 1 \cdot c_0]$, where $c_0 = 176.6\ \frac{\mathrm{m}}{\mathrm{s}}$ (see appendix D). Some solutions do not have both boundaries of the initial value defined. I do not regard these solutions as well-defined, but include them for intuition. To give the reader a picture of how slow the convergence of each problem is, I calculate at what decimal the upper and lower boundary of the bisection interval differs at the end of the optimization, Δamp. Neither the soliton equation nor the dispersion coefficients have an explicit z-dependency (sketch of proof in appendix A.2), so I choose to center all solutions around zero.

I also attempted to run the e. coli heat capacity in figure 2. Unfortunately I did not obtain any wave-like solutions, see fig. 39 in appendix E.3. This does not mean a solution doesn't exist. In case of a constant dispersion coefficient there is a direct relationship between the width of the pulse and the size of the dispersion coefficient, see figure 12). This implies that small dispersion values due to broader heat capacity profiles will result in narrower waves. Since I'm solving from the asymptotic end, the stark increment between tail and amplitude is likely hard for the ODE solver to handle. Instead I chose to explore the role of cooperativity at values close to the default cooperativity of $n = 80$.

3.2.1 Determining the width of the pulses

In order to compare solitons of different velocities or parameter setups, I need some rough measures of shape and size. I decided to use FWHM as a simple measure of the width of localized solutions, maximum being the amplitude of the soliton. For density dependent dispersion coefficient cases, there is an offset in the solutions caused by the initial density $y_0 = \Delta\rho_{\text{end}}$ being too large, see fig. 13 and 14. For the constant dispersion coefficient cases the ends tend to 0, $\Delta\rho_{\text{end}} = 0$, so there is no offset (fig. 9 and 10). Half max is taken to be half way between the asymptotic ends and the maximum density of the pulse, disregarding any offset in the data.

Since all methods I've worked with use adaptive step size, I cannot be sure that there are data points close to half max. I therefore define a tolerance margin around half max. For the twice and thrice integrated constant dispersion coefficient case it is 0.05 times the maximum density relative to the asymptotic ends and the density dependent cases both have a margin of 0.025 times their relative maximum density.

Finding the maximum of the pulse: Though it is a localized solution, the ends might oscillate or diverge such that they contain points exceeding that of the maximum I'm interested in. In the constant dispersion coefficient case, the maximum I am looking for happens to be at the initial point, $y_0 = \Delta\rho_{\max}$. In the density dependent cases, I search for the first maximum in the direction of evaluation.

Find all points within the margin of FWHM using the criteria:

$$\text{abs}(\Delta\rho - \frac{1}{2}\Delta\rho_{\max}) < \text{margin}$$

If I have a perfectly localized solution, I end up with two intervals of points, one for each side of the pulse. But as with finding a maximum, the tails of the solution might contain points that fit this criteria, causing more intervals to appear in the output. Since I centered the solution around 0, the intervals belonging to the left and right side of the pulse will be the intervals of lowest z-values.

The points inside the interval are often irregularly distributed, so I decide to make a linear extrapolation between the boundaries of the interval and determine the z-value belonging to the value $y_{\text{center}} = \frac{y_1+y_2}{2}$, where y is the density relative to the asymptotic ends and subscript 1 and 2 denotes the two boundaries.

$$z_{\text{center}} = z_1 + \frac{y_{\text{center}} - y_1}{\Delta y/\Delta z}$$

This is done for both sides of the pulse. FWHM is then 2 times the mean value of z_{center} for the left and right side. If a localized, symmetric pulse hasn't been found, such as in the case of velocities $\{0.80 \cdot c_0, 0.85 \cdot c_0\}$ in fig. 13, the FWHM obtained using this method does not align with the graphs, since one side is wider than the other and the FWHM is an average of the two.

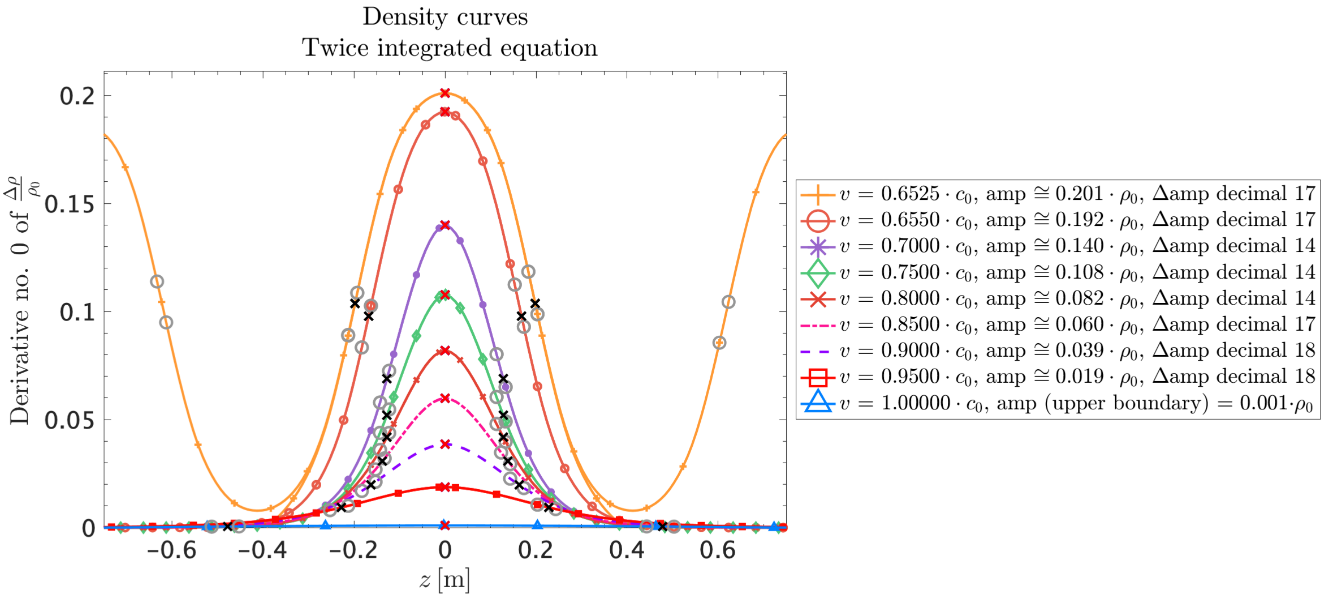


Figure 8: Twice integrated, ODE15s, $h = 50\frac{\mathrm{m}^4}{\mathrm{s}^2}$. Red crosses mark the first maximum, in the constant dispersion coefficient case coincidentally the initial point. Grey circles mark the boundaries of the intervals where points are inside the tolerance margin of half max. The black crosses mark the final FWHM.

3.2.2 Constant dispersion coefficient

The solutions for the twice and thrice integrated constant dispersion coefficients are displayed in figure 9 and figure 10, respectively. Both cases have a broad range of well-defined solitary solutions. My solutions have dimensions of similar to what is found in literature [33, 44] and [56, chap. 1.2.2, fig. 1.9], though these studies have chosen $h = 2\frac{\mathrm{m}^4}{\mathrm{s}^2}$ and therefore have a slightly smaller width. The amplitudes decrease as the speed of sound increases, and the widths obtained by the method in section 3.2.1 display a U-shaped velocity dependence with a minimum in the low end of the velocity range, see figure 11. The lowest velocity with a well-defined, solitary solution is $0.6499 \cdot c_0$ for the twice integrated case and $0.6500 \cdot c_0$ for the thrice integrated case. The highest velocity with a well-defined, solitary solution is $0.9500 \cdot c_0$ for the twice integrated case and $1.0000 \cdot c_0$ for the thrice integrated case. The limit velocity for solitary solutions is in the twice-integrated case in the range of $[0.6498 \cdot c_0, 0.6499 \cdot c_0]$, and the thrice-integrated limit velocity is somewhere in the range of $[0.6450 \cdot c_0, 0.65 \cdot c_0]$, which fits well with the analytically derived limit velocity of $\approx 0.649851 \cdot c_0$ found by Lautrup et al. [44], using the same parameters from Heimburg and Jackson [33]. Lautrup et al. [44] find the width of the analytical solution diverges for both low and high velocities with a minimum at $\approx 0.734761 \cdot c_0$, which matches well with the U-shaped velocity-dependence of the widths in figure 11. The thrice integrated solutions are generally taller and more narrow than the twice integrated solutions, this is very evident for the velocity of $0.70 \cdot c_0$ in figures 9 and 10. Both the solutions of the upper and lower bound on the initial value are represented in these plots, and these are indistinguishable. I therefore think it is unlikely that the difference between twice and thrice integrated solutions has to do with inadequate convergence. This issue also appeared when I varied the dispersion coefficient h for a fixed velocity, see figure 12. The twice integrated case looks similar to results obtained by Panagiotis [61, chap. 3, fig. 3.3] and Mosgaard [56, chap. 5 fig. 5.1], where the dispersion term influences the width but not the amplitude of the solution. The thrice integrated case on the other hand doesn't adhere to this and ODE23tb doesn't even reach a proper solution for small h. I believe the ODE-solver introduces the variation because it has a difficult time with the square root in equation 30.

I fitted the tails of the solutions to see if they decayed exponentially. Since the solutions are obtained by running the ODE-solver from maximum amplitude towards the asymptote and then mirroring the data, there only exists one unique tail. Plots of the fitted tails are included in figure 42, appendix E.5. I didn't find a way to create a good test statistic for data generated by an analytical algorithm, so I just assert that by eye the fits seem to follow the data well.

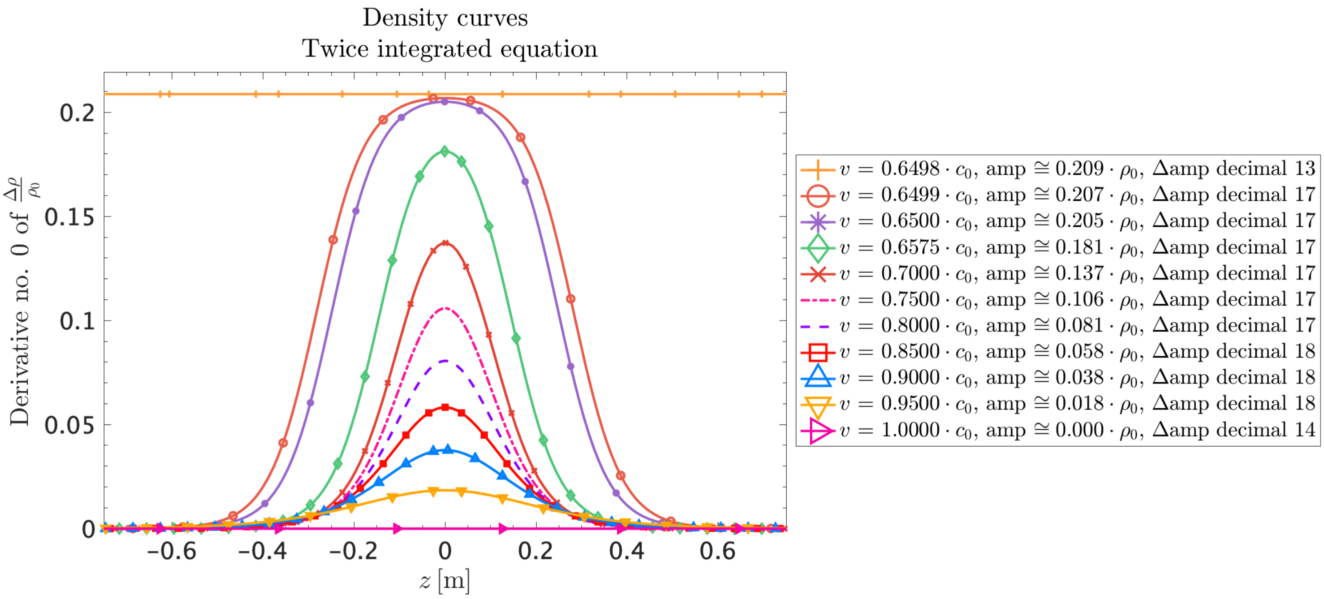


Figure 9: Localized density pulses for constant dispersion, twice integrated, ODE23tb, $h = 50$.

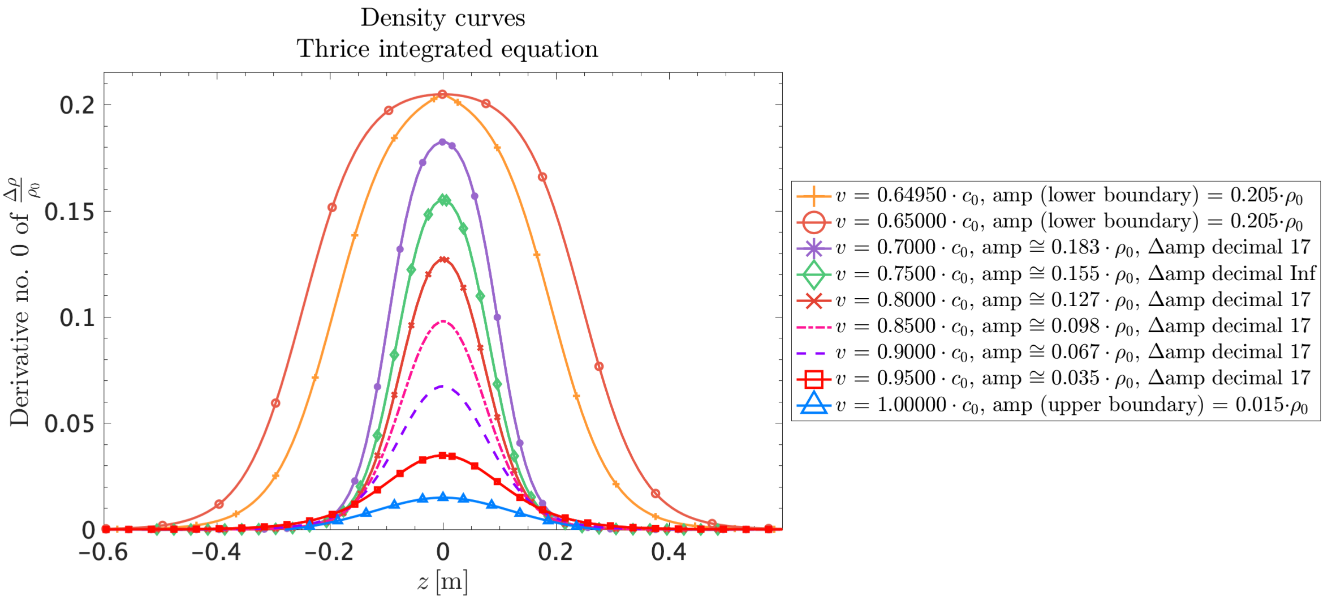


Figure 10: Localized density pulses for constant dispersion, thrice integrated, ODE23tb, $h = 50$.

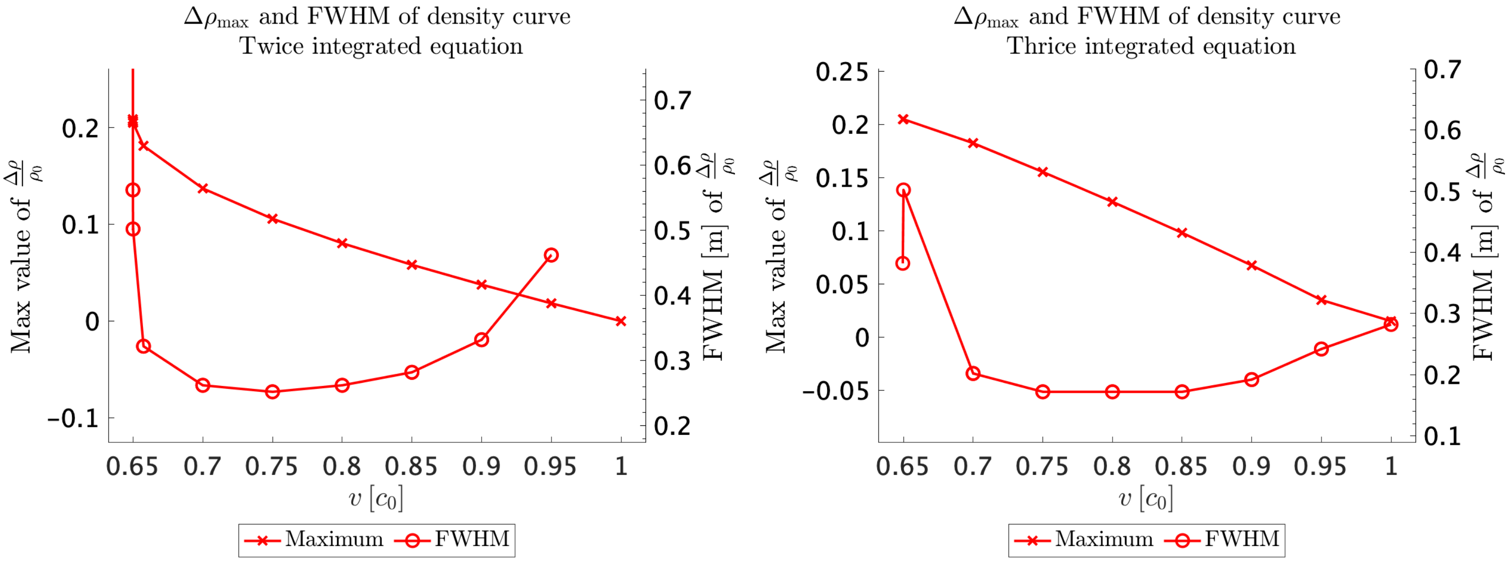


Figure 11: Velocity dependence of widths and amplitudes for constant dispersion coefficient. In both cases, solutions were obtained using ODE23tb. These plots include the widths and amplitudes of all solutions, also the ones that are not well-defined, with the purpose of giving a qualitative idea of the behaviour.

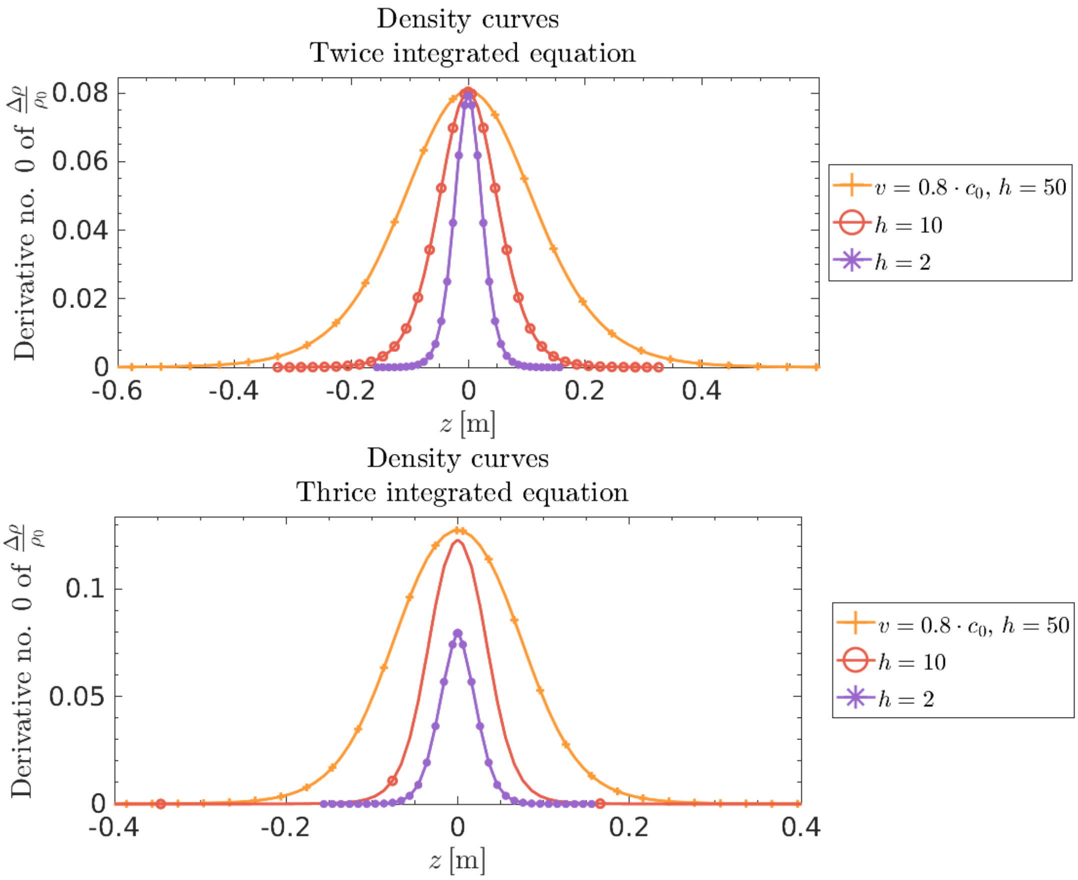


Figure 12: Varying the constant dispersion coefficient h, ODE23tb. **ABOVE:** Twice integrated. **BELOW:** Thrice integrated.

3.2.3 Density-dependent dispersion coefficient

In figure 13 and 14 I present the solutions to the density dependent dispersion coefficient in case of the Gaussian approximation and the thermodynamic model of the heat capacity. In both cases the tails are modelled by an exponential below the initial density threshold. The solutions therefore appear much more well-behaving than the raw data from the ODE-solver with wildly diverging ends. I have included the raw curves with divergences in appendix E.2, figure 37 and 38. Two things immediately strikes the eye when looking at figure 13 and 14: The widths of the solutions are quite large, and some velocities have two solitary solutions. The full wave profiles in case of the self-contained model are about 2 meters long, which is a bit larger than what can be estimated from experiment. The nerve pulse takes about 10 milliseconds to pass when measuring the action potential. In case of a myelinated nerve, the pulse travels at approximately 100 $\frac{\mathrm{m}}{\mathrm{s}}$, which is equivalent of 1 meter nerve profiles. Since the speed of sound is much smaller in non-myelinated nerves, $1\frac{\mathrm{m}}{\mathrm{s}}$-$5\frac{\mathrm{m}}{\mathrm{s}}$, about 1 meter is the largest one would expect a nerve pulse to be (values from [29, chap. 18.1]). As mentioned previously, the disparity is most likely caused by difference in cooperativity between biological and pure lipid membranes.

The maximum area density change in case of the self-contained is $\approx 8\%$ relative to ρ_0. This is quite small compared with the estimated 25% change in area density for pure DPPC bilayers undergoing phase transition. It is also much less than in the case of the Gaussian approximation for the heat capacity and both cases of constant dispersion, resulting in $\approx 17\%$ and $\approx 20\%$ relative changes. Since the widths of the solitons in case of the Gaussian approximation are about half the size of the widths in case of the self-contained model, this difference in maximum density change is likely a result of the previously stated difference in cooperativity between the thermodynamic modelled heat capacity and the Gaussian approximation from experimental data.

Double solutions can be seen for the velocities $\{0.7000 \cdot c_0, 0.7250 \cdot c_0\}$ in case of the Gaussian approximantion and $\{0.7500 \cdot c_0, 0.7750 \cdot c_0, 0.800 \cdot c_0, 0.8250 \cdot c_0\}$ in case of the thermodynamical model. Overall the doubles are very similar. By the look of figure 15, it would seem that the left side amplitudes constitute an upper bound to the amplitudes on the right side, and the left widths are a lower bound on the right side widths. This claim is largely based on the lower half of the plot for self-contained model, since the right side solutions belonging to the self-contained of velocities larger than $0.8500 \cdot c_0$ and almost all of the right side solutions of the Gaussian model aren't well-defined solitary solutions and therefore can't be trusted too much. Judging only from left side data the solutions seem to get wider and amplitudes smaller as velocity goes up, but the relative changes are very small except for the amplitudes of the self-contained model.

In both cases the right side solutions show an offset, so that they do not converge to 0. It is most prominent in case of the self-contained model, which coincidentally has larger number of double solutions. I suspect these issues might be related. I attempted to decrease the initial density from $10^{-2.2} \cdot \rho_0$ to $10^{-2.6} \cdot \rho_0$ by increasing the tolerance an order of magnitude to see if that would lead to convergence in the tails. I was unable to obtain any solutions in the case of the self-contained model. The Gaussian approximation led to a few solitary solutions, both in case of ODE15s and ODE23tb, see appendix E.4, figure 40 and 41. The solvers yielded different results, for ODE15s I was only able to evaluate two almost identical velocities and found an almost identical doublet. In the case of ODE23tb there are no doublets and I was able to evaluate solutions for a much larger range of velocities. Common for both solvers is the fact that left and right solitary solutions converge nicely towards zero, so lowering the initial condition did solve the issue of convergence. Whether it also fixes the issue of double solutions is difficult to say, since there hardly seem to be any solutions at all.

I fit the tails of the raw ODE-data in appendix E.2, figure 37 and 38, to an exponential, and the plots are included in appendix E.5, figure 43 and 44. The fits of the decreasing ends do not converge as well as for the increasing ends, which is not surprising if one takes a look at how quickly the raw solutions from the ODE-solver diverges in appendix E.2.

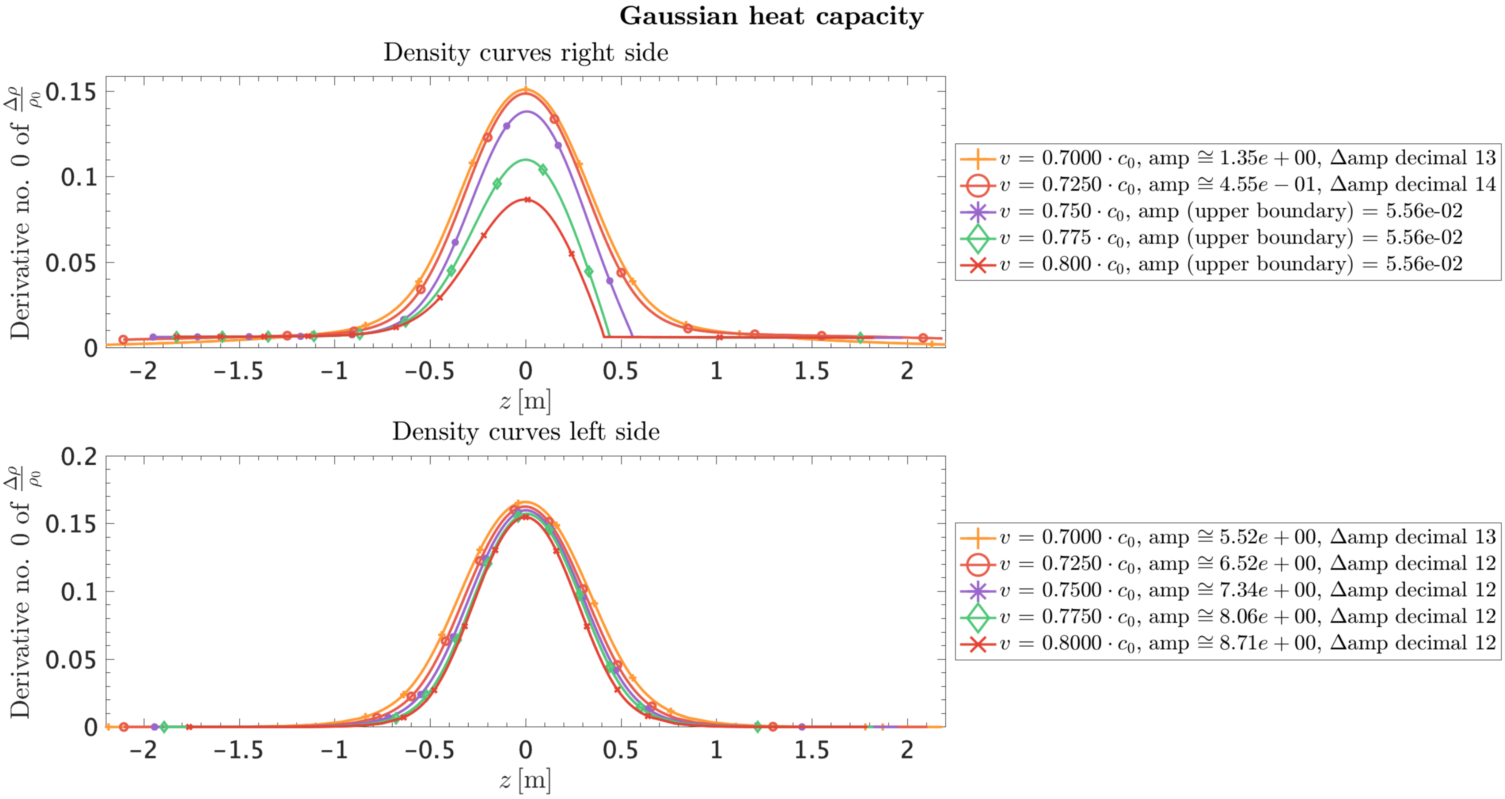


Figure 13: ODE 15s, this plot contains a selection of the solution for the density dependent dispersion coefficient in case of Gaussian approximation of the heat capacity. For all solutions see appendix E.2, figure 37.

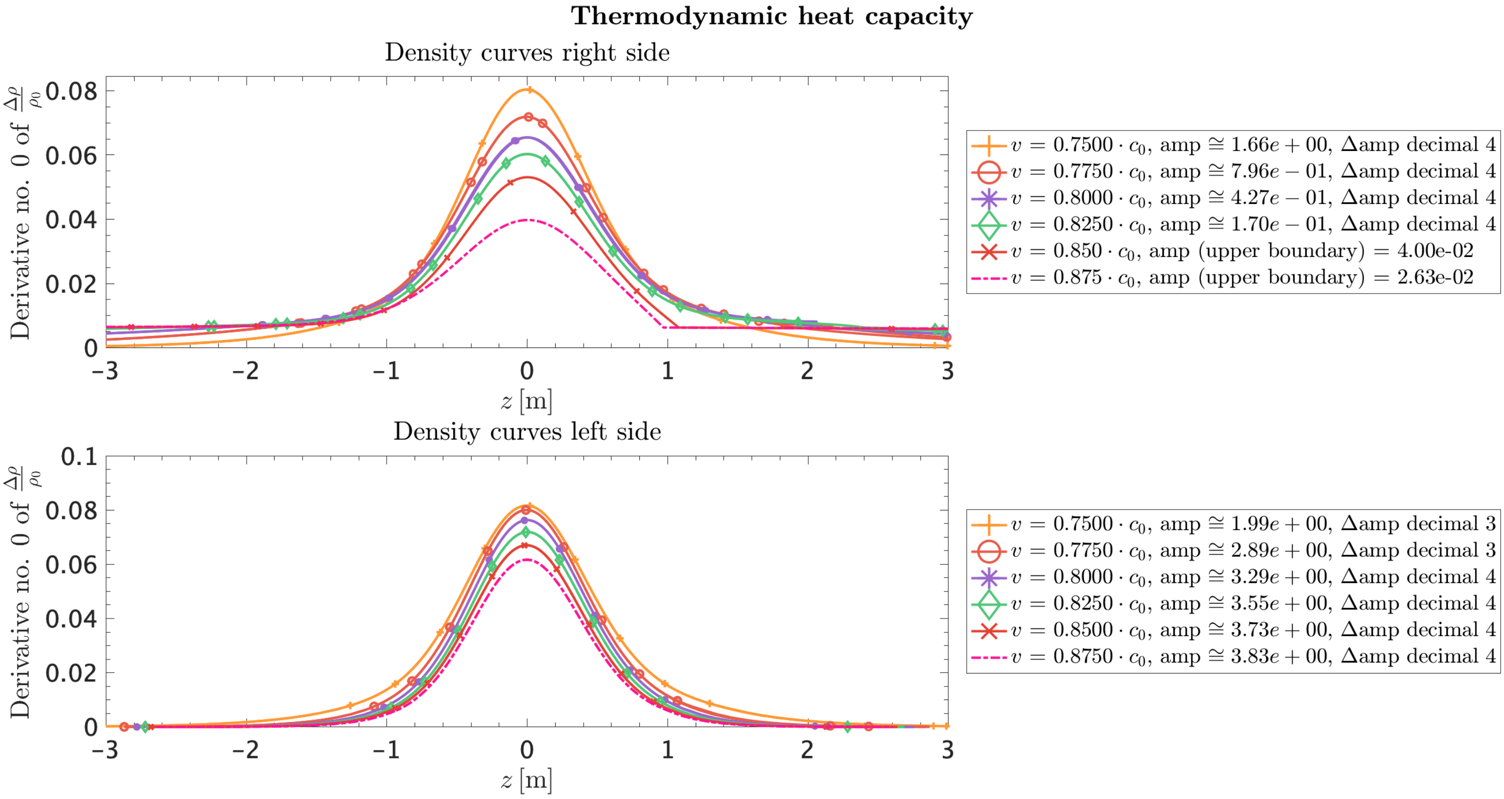


Figure 14: ODE 15s, cooperativity of 80. This plot contains a selection of the solution for the density dependent dispersion coefficient case in case of the self-contained model for the heat capacity. For all solutions see E.2, figure appendix 38.

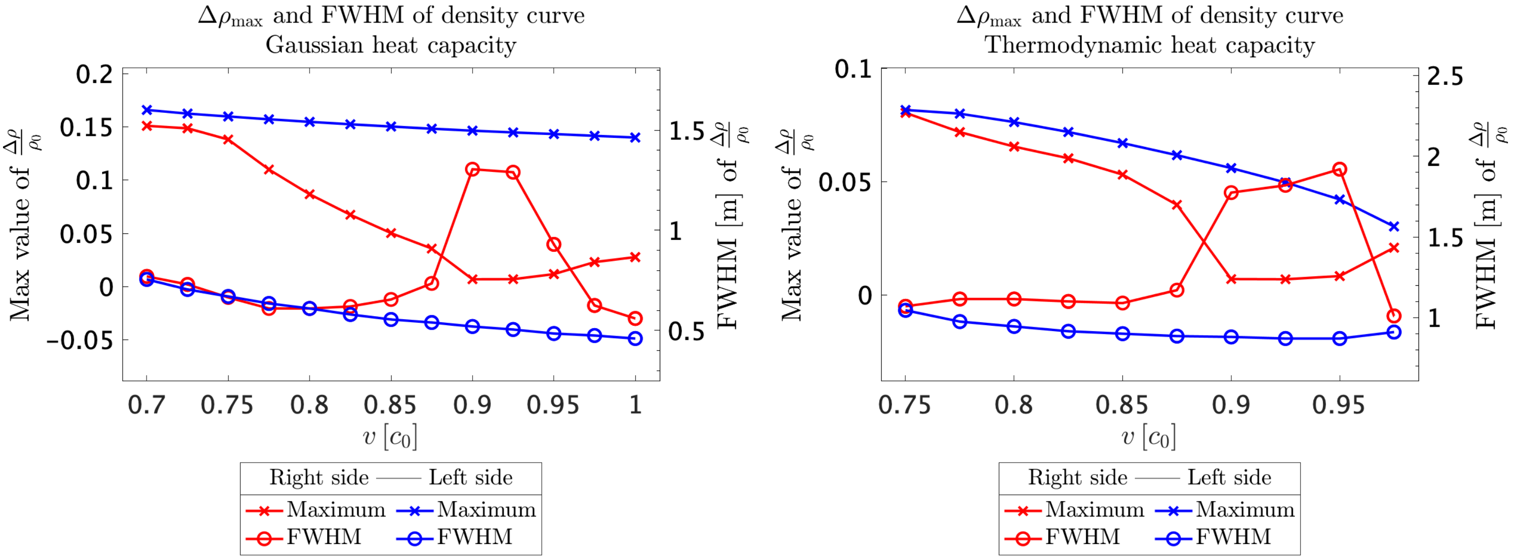


Figure 15: Velocity dependence of widths and amplitudes for the density dependent dispersion coefficient. In both cases, solutions were obtained using ODE15s. These plots include the widths and amplitudes of all solutions displayed in appendix E.2, not just the ones in the sparse plots in figure 13 and 14, and also the ones that are not well-defined or fully solitary, with the purpose of giving a qualitative idea of the behaviour. Not fully solitary solutions are: All right side velocities *but* velocities of $\{0.7000 \cdot c_0, 0.7250 \cdot c_0\}$ in case of the Gaussian approximation, and all right side velocities $\{0.8500 \cdot c_0, 0.8750 \cdot c_0, 0.9250 \cdot c_0, 0.9500 \cdot c_0, 0.9750 \cdot c_0\}$ in case of the self-contained model.

3.2.4 The role of cooperativity in the self-contained model

Last but not least I investigated the role of cooperativity in the self-contained model for the velocity $v = 0.80 \cdot c_0$. I wasn't able to alter the default cooperativity of $n = 80$ much, so the resulting heat capacity profiles all have similar widths and maximum heat capacities within the same order of magnitude, see figure 16. The solutions are very close and can hardly be distinguished from each other in figure 17. I note that all solutions are solitary doublets and the left side tails converge badly. The widths and amplitudes are displayed in figure 18. As opposed to figure 15, these solutions are all well-defined. The widths and amplitudes of the left side solution follow a similar cooperativity dependence as the right side solutions. Again the left side amplitudes constitutes the upper boundary to the right side amplitudes, and the left side widths constitutes the lower boundary to the right side widths. For both sides, there is a transient decrease in the widths and increase of the amplitudes at low cooperativities, followed by a minor increase in widths and decrease in amplitudes. It would be interesting to compare cooperativity plots for different velocities to see if the transient behaviour is a persistent feature. Overall the range of cooperativities is too small for there to be enough variation in the widths and amplitudes to claim that the solutions reflect the inverse relationships between the width and height of the melting profile and the width and amplitude of the soliton.

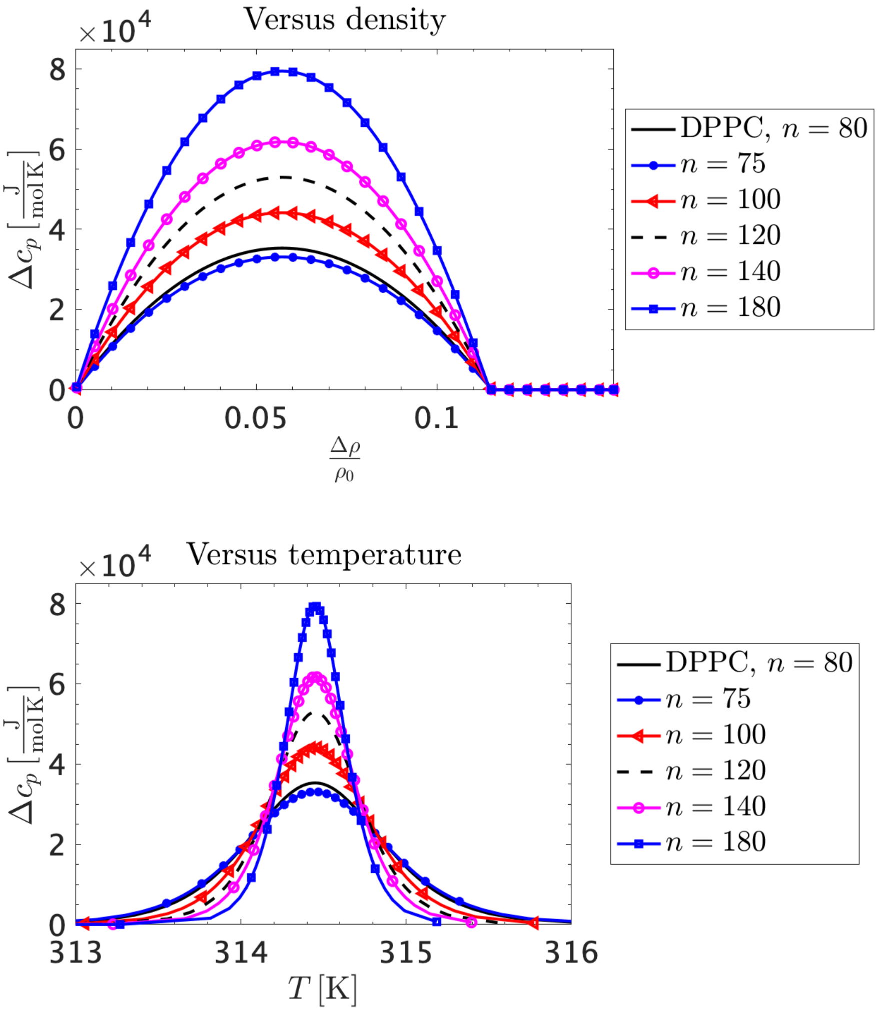


Figure 16: Thermodynamically modelled heat capacities for DPPC bilayers at different cooperativities, n.

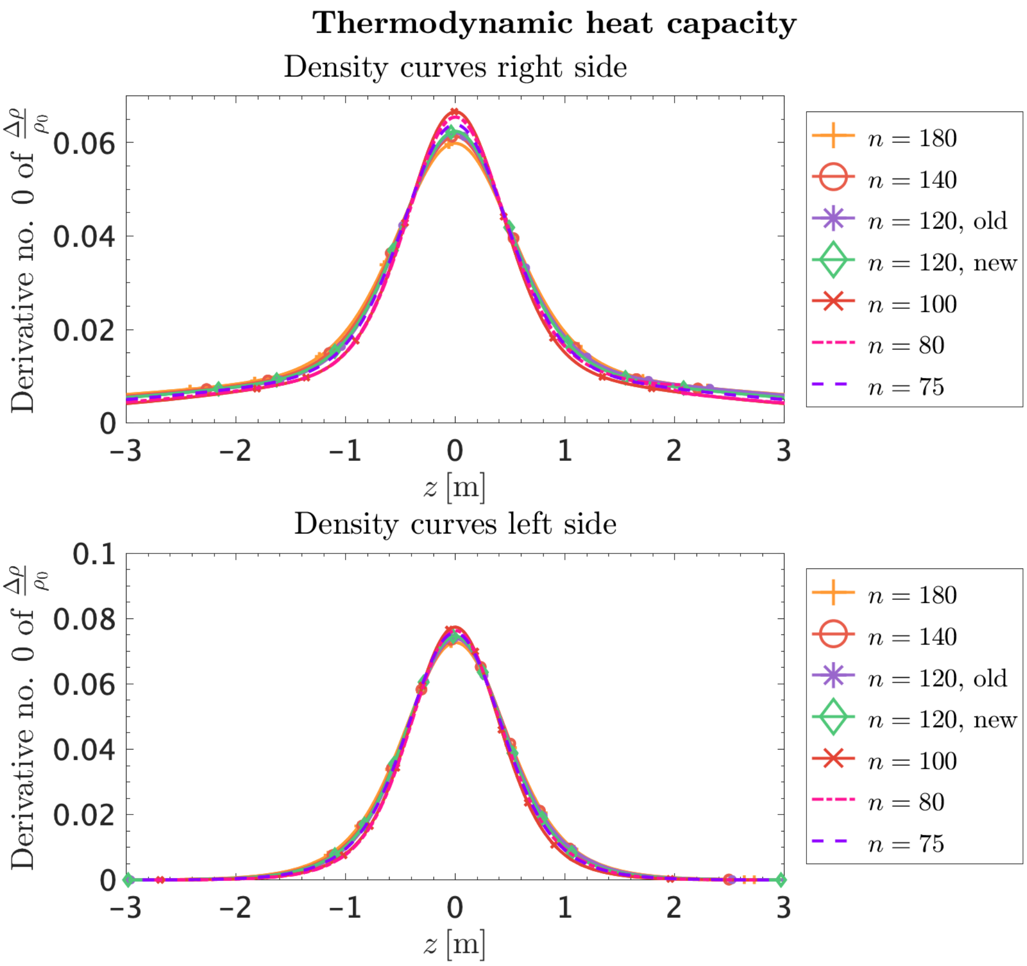


Figure 17: Solitary solutions for the self-contained soliton model at different cooperativities n at propagation speed $v = 0.80c_0$, ODE 15s. The corresponding heat capacity plots are displayed in fig. 16.

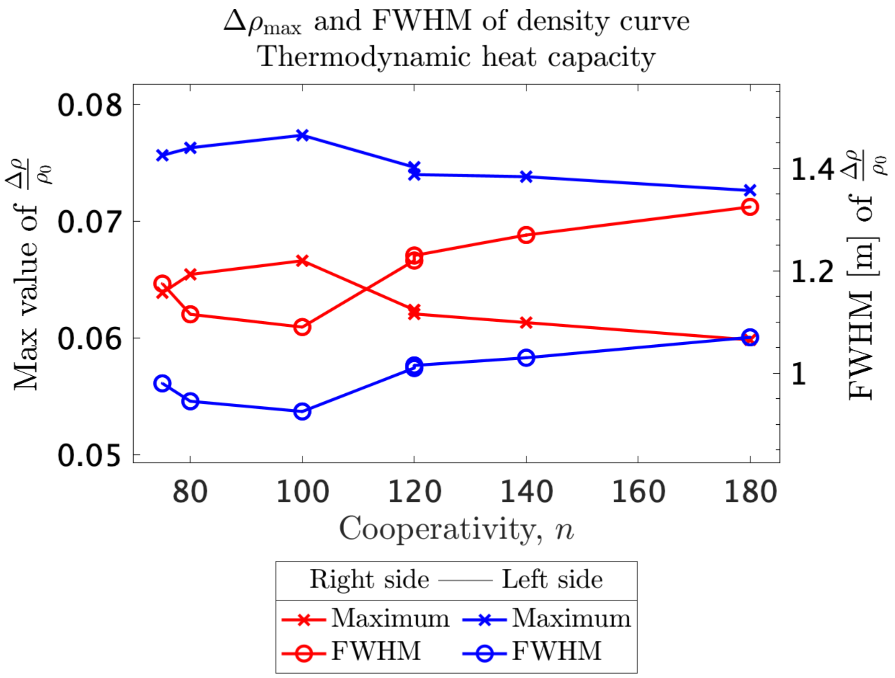


Figure 18: Widths and amplitudes belonging to the solitons in fig. 17. Varying cooperativity n at $v = 0.80c_0$, ODE 15s.

3.3 Discussion

I have reproduced the solutions to the soliton equation in case of a constant dispersion coefficient reported by Heimburg and Jackson [33] and Lautrup et al. [44]. The twice integrated version seem to have most reliable solutions, since ODE solvers differ at small h or medium to low sound velocities in case of the thrice integrated constant dispersion coefficient. The amplitudes and widths show the same velocity dependence as previously found for the analytical solution [44].

I managed to find solitary solutions for a narrow interval of parameters for the density dependent dispersion coefficient in case of both the Gaussian heat capacity approximation from experimental data and the self-contained model. The numerical framework for the two cases is the same and can therefore be applied to other models of heat capacity if needed. I found double solutions that are almost identical in size and shape and could be taken as the same solution though they appear at different initial conditions. The doublets might be an artifact of the numerical method, since improving initial conditions causes the doublet to either disappear or become even more identical. Unfortunately the solving procedure simultaneously crashes, so difficult to say if these findings are significant.

In order to solve a wider range of parameter sets it is necessary to improve my current solving procedure for the density dependent dispersion. There are several paths one may try. One could try to scale the problem or make it dimensionless to decrease the effect of round-up errors in the calculations. This is not a straight forward task in the case of the self-contained model, where the heat capacity of the dispersion term is described by an equation that has yet to be analytically solved.

The most effective way to improve things is likely to implement an alternative optimization scheme for the initial value, and currently I have two ideas: The first idea was given to me recently by Andrew Jackson, who proposed to iterate from both tails until first maximum is reached from either side. The task is then to build an algorithm that optimizes the amplitude of the exponentially decreasing ends until the solutions meet at the maximum. Assuming exponentially localized solutions only have a single maximum, this procedure will converge towards solitons, and one avoids the issue of optimizing the stiff, divergent ends. Another idea was proposed to me by Thomas Heimburg, which was to use Newton-Raphson's method and improve on an approximated soliton or solution from the case of constant dispersion coefficient. This method relies on the terms of the differential equation being differentiable, so in case of the thermodynamical model of the heat capacity it would be necessary to approximate the heat capacity equations, unless an analytical solution has been found.

The use of pseudo spectral methods such as the fast Fourier transform to speed up calculations is very common in the field of numerical analysis, and has been applied to the soliton model in case of non-constant dispersion terms before [17]. I did not actively implement any such methods, but they might already have been included in an approximate sense in the ODE solvers, in order to know this one would have to study the source code. One could also look into the possibility of transforming boundary conditions into initial values [71, chap. 9] or integrate the initial conditions away like what is done in the constant dispersion coefficient cases. Again it is necessary to find an analytical expression for the heat capacity equations in case of the self-contained model.

Before going further into depth with the current version of the dispersion coefficient in the self-contained model, it might be of interest to ask wether any improvements can be done to Mosgaards dispersion coefficient. If the membrane area changes $\approx 25\%$ during phase transition in DPPC membranes, and the nerve membrane goes through 85% of the phase transition during a soliton - does the small amplitude assumption that the dispersion coefficient relies on still hold? It has also been pointed out by Engelbrecht et al. [16] that the dispersion relation leads to an unbounded group velocity at high frequencies. They propose the use of a mixed dispersion term to fix this issue, $\frac{\partial^4 \Delta\rho^A}{\partial x^2 \partial t^2}$, and continues their work using this version in later papers [17]. Unfortunately they then loose the self-contained, thermodynamic nature of the soliton theory, which is a big part of its appeal.

4 The influence of lithium ions on the phase behavior of lipid membranes and the excitability of nerves

The soliton theory has many applications with regards to drugs. In the framework of the soliton theory any component that influences the thermodynamics of the nerve membrane will have an impact on the propagation of the nerve pulse. The typical example is the role of freezing point depression in anesthesia [33, 30, 31]. Another well-known medicament worth consideration is the element lithium.

Lithium salts have been tested on a wide range of human conditions, from gout and mental illness [8] to viruses [59, 66], cluster headaches [62] and cancer [41], with varying levels of success. Most notable is its preventative effect on bipolar disorder, being the most well-established and effective treatment currently available [48]. Since lithium therapy was implemented in the 1960's, there have been numerous studies of its beneficial impact on the brain, such as the entrainment of the circadian rythm, regeneration and preservation of grey and white matter, remyelination of nerves and prevention of bipolar-related dementia [48]. Other studies have found a mitigating effect on mitochondria dysfunction in the brain related to bipolar disorder [48, 54, 63, 41]. Using a promising new neural model for bipolar disorder created by stem cell differentation, researchers have found that hippocampal neurons formed from bipolar patients were hyperexitable compared to the general population, and lithium reversed this trait only in patients responding to lithium treatment, indicating differences in underlying pathophysiology [54, 72, 67]. It would be tempting to conclude lithium responders suffer from some dysfunctional microbiological pathway mitigated by lithium, but one would still need an explanation for the myriad of other non-differential effects lithium has on the body, such as kidney function, cancer prevention [41] and antiviral effects [59, 66]. It is generally difficult to tell if lithium-induced changes are specific to bipolar patients, especially on the cellular level, and to this day there is no known molecular mechanism of action for lithium [48]. I will attempt to make the case that a general, thermodynamic effect is behind lithium's restorative effect on the human psyche, connecting the impact of lithium on membrane phase behaviour with nerve signaling using the soliton theory.

4.1 Lipid phase behaviour in presence of lithium ions

Lithium's remarkable effect on structure and phase transitions of lipid bilayers has been documented several times [24, 25, 9, 64, 46, 14], but has yet to be fully clarified. In the early 80s Hauser and Shipley [24] [25] compared melting temperatures of anionic DMPS bilayers at 5.8-7.0 pH and various concentrations of sodium, potassium and lithium. They discovered the addition of $\approx$ 1M lithium increased the melting temperature of the lipid membrane 40°C – 50°C. In comparison $>$ 1M concentrations of sodium or potassium only changed the melting temperature a few degrees. At 0 M – 0.7 M concentrations, lithium ions would partially crystallize and dehydrate the DMPS bilayers, inducing lateral phase separation of lipid-ion complexes and ion-free lipids, which also wasn't observed in sodium and potassium, but similar results have been reported for calcium

[25, 40]. Though it is generally known that water content in macromolecules such as fatty acids and proteins is strongly affected by metal ions [75], the dehydrating effect of lithium is quite unexpected, since water tend to associate strongly with small metal cations [5]. This behaviour mildly resembles that of protons [9] (pH titration) or ions of larger ionic radius and valence such as calcium ions [40]. Usually the primary effects of monovalent cations are attributed to non-specific screening [14], but lithium seems to have an affinity for the lipid membrane unlike other monovalent cations [41, 46], associating specifically to lipids head groups, locally altering conformation and hydration and thereby stabilizing the solid lipid phase [9, 14, 24, 25]. Anionic lipids, such as phosphatidylserines, show the strongest reactions. Phosphatidylserines are coincidentally (or not) one of the primary lipids of excitable membranes [24, 46], and their strong reaction to lithium causes Hauser and Shipley [24, 25] to propose this as a potential mode of action in pharmacological treatment.

Empirical and theoretical studies indicate that structure changes related to lipid phase transition caused by lithium binding correlate with decreased hydration of the lipid [10, 41]. This is generally seen as a consequence of lithium binding to the lipid head-groups of the membrane [25, 9, 64, 14]. In their molecular dynamics simulation of DPPS vesicles in the presence of lithium from 1997, López Cascales and Garcia de la Torre [46] find that lithium interacts strongly with the water at the interface of the membrane, dispels parts of the hydration water, breaks hydrogen bonds, slows down the diffusion of water into the membrane and effectively screens the membrane's electrical potential. Some of these effects have been reproduced experimentally [41], but not all. Among the monovalent cations, lithium's ability to dehydrate and alter the membrane structure of lipids is only exceeded by protons [41, 5, 11, 9]. Ranking ions according to their ability to change lipid interaction with water is referred to as the 'lyotropic series', a lipid analogy to the Hofmeister series which ranks ions according to their ability to precipitate proteins in solution [5]. The series emerge in other contexts, and indicates a possible connection to water interaction. As an example, the conductances of monovalent ions in pure lipid membranes have been shown to follow the lyotropic series close to melting temperature, lithium having a remarkably larger conductance than the other ion species [30] [29, chap. 17.3.1]. Coincidentally, lithium has been documented to alter the permeability of the membrane [24] and other transport functions of cell membranes [24]. This is not suprising in the view of the thermodynamics presented in this thesis, and further studies into the central mechanisms of lipid phase behaviour in presence of lithium ions could possibly help the understanding of these phenomena.

4.2 Nerve excitation in presence of lithium ions

How a certain cation influences a nerve can vary greatly depending on whether it's inside or outside of the nerve membrane, and on the concentrations of surrounding ions. Experiments by Tasaki and associates [78, 79] show that in general, monovalent ions in the external medium are not essential for the existence of an action potential, but have a great impact on its amplitude and shape [79, fig. 3]. In comparison a lack or surplus of divalent cations in the external medium, Ca^{2+} specifically, can abolish the action potential completely. On the other hand the intracellular concentrations of monovalent cations can have a great importance for the nerve signal. In their paper from 1966 Tasaki et al. [78], a nerve that has been rendered inexitable after being soaked in Ca^{2+}-ions regains its excitability by intracellular perfusion of Cs^{+} ions. This did not work for intracellular perfusion of Rb^{+}, Na^{+} or K^{+}, so not only is the balance between mono- and divalent cations important, ion species matter too.

Howarth et al. [38] conducted a number of experiments with nerves in a buffer where sodium had been replaced with lithium (154 mM, pH 7.2). They were able to produce a full action potential under the same conditions as with a sodium buffer, concluding that lithium ions are interchangeable with sodium ions in nerves similarly to what has been found for muscles [43]. There were differences between the two ions, however. Lithium delayed the action potential and initial heat about 20% and slowed down the repolarization of the membrane [38, fig. 16l]. Yet the most obvious differences were seen in the heat generation [38, fig. 9]. The heat production of the nerve is biphasic with an initial heat production coinciding with the membrane depolarization [75], followed by heat reuptake/negative heat and a afterheat/recovery heat production [1]. Lithium effectively eliminated the recovery heat, causing the total positive heat to go down ≈ 20%. Contrarily the negative heat is increased by ≈ 20%. As a comparison, removing potassium from the original buffer led to a 10% increase in positive heat and a doubling in the negative heat causing a temporary cooling of the nerve, but it did not abolish the recovery heat. This could be an indicator of lithium's ability to stabilize anionic lipids more towards the solid phase mentioned in the previous section 4.1, though one has to be careful making direct comparisons between pure lipid membranes and complex biological membranes on limited information. Further investigation (through of literature or by experiment) of the response in nerves subjected to immediate or long-term lithium exposure phrased in general, thermodynamic variables could potentially reveal some very interesting perspectives.

4.3 DSC and pressure pertubation experiments of DPPC membranes in presence of sodium, potassium and lithium

I investigated the thermodynamic properties of DPPC bilayers through pressure pertubation of membranes in suspension. This is done in a differential scanning calorimeter where temperature and, in less degree, pressure, can be precisely monitored. Pressure pertubation calorimetry, abbreviated PPC, exploits the well-known effect that increasing the pressure on a material increases the melting temperature and vice versa, to obtain information about relaxation times and thermal volume expansion coefficients of the system. Small decreases or increases in pressure at constant temperature causes lipids to move towards or further from transition. Close to transition a small move in either direction causes a manifold change in heat capacity. The lipids respond with a measurable uptake or release of heat, which the calorimeter must compensate for in order to stabilize the temperature while the system relaxes towards the new equilibrium. Previously I showed how the relaxation time of the system is proportional to the excess heat capacity, see section 2.2.1, equation 14:

$$\tau \approx \frac{T^2}{L}\Delta c_p \quad \text{(at phase transition)} \tag{32}$$

.

By comparing PPC measurements with differential scanning calorimetry measurements for a given sample, I can obtain values for the phenomenological coupling constant L. Likewise thermal volume expansion coefficients can be derived from the heat and pressure changes in the PPC experiments using the following relation between heat changes and pressure changes from Heerklotz and Seelig [27]:

$$\alpha_V = -\frac{\Delta Q_{\mathrm{rev}}}{T\langle V\rangle \Delta p} \tag{33}$$

These can then can be compared to DSC scans to obtain the empirical proportionality constant seen in section 2.2.1, equation 4:

$$\alpha_V = \frac{\gamma_V}{\langle V\rangle}\Delta c_p$$

4.3.1 The method

I prepared bulk solutions of 100 mM NaCl, KCl and LiCl in MilliQ water to be used in the lipid dispersions of LUV DPPC. I predicted the approximate weight of salt needed for 60 ml of each saline solution in the following way:

$$\text{needed weight} \quad = \quad \text{molar weight} \quad \text{x} \quad \text{molarity} \quad \text{x} \quad \text{volume} \tag{34}$$

Since it is hard to be precise when measuring such small quantities of dry material, I corrected the volume MilliQ needed using the same relation (34):

$$\frac{\text{measured weight}}{\text{molar weight} \quad \text{x} \quad \text{molarity}} = \text{corrected volume} \tag{35}$$

For the control sample plain MilliQ water was used. List of materials can be found in appendix C.

Following the same procedure I predicted the weight of frozen, dry DPPC lipid needed for a 1 ml sample of 10 mM DPPC. For each of the four samples, the lipid was weighed in the same glass vial later used for the dispersion. For each vial (labeled) the weight was noted, and the corrected volume of saline or control was calculated. Saline (or control) was added, the sample sealed with PARAFILM®, before it was repeatedly heated up and vortexed with a MS1 Minishaker from IKA®, until no residue was stuck on the surface of the vial. The sample was heated by holding it under a hot faucet or placing it in a container filled with hot tap water. The water destroyed the labels, so they had to be replaced. The samples where refrigerated until sonication.

Sonication disrupts the multilammelar lipid vesicles (MLVs) formed immediately after hydration, and the lipids change into another stable conformation of large unilamellar vesicles, LUVs. The particles in the suspension changes size from much larger than wavelength of visible light $\gg$ 700 nm, to about 20 nm (transparent after sonication). The particles then quickly fuses to about 100 - 140 nm vesicles, which gives the samples a slightly opalescent hue. Color and transparency is thus a good indicator for adequate sonication. I discovered the Branson Sonifier® cell disruptor B15 was faulty during the process. A program shifted between two modes of high and slightly lower power. During the high power mode, the sample did not get warmer or clearer, opposed to the low power mode. During the MilliQ sample, the machinery started making a loud, tooting sound. It took several attempts to sonicate the samples, all along I had to be careful of evaporation, excessive bubbling and contamination. A few white specks were visible in NaCl, LiCl and MilliQ series that I could not get rid off. The samples where subsequently refrigerated and ready for analyzation.

The time course of the sample preparation was as follows (year 2020): Samples prepared on 26th of November. Sonication 27th of November. NaCl, KCl and MilliQ samples sonicated again 30th of November since they looked milky and had visible specs when removed from the fridge (not the case for LiCl). After LiCl had been through DSC and PPC and subsequently refrigerated for a while, the samples looked similar (MilliQ more milky than the rest).

Samples and reference were degassed for about 15 minutes before they were added to the cells. This was mostly for the sake of the lipid samples that had been through vortexing and sonication, though they had all been refrigerated for at least 48 hours before analyzation, which also decreases the gas content of the dispersion considerably. The same water reference was used for all samples analyzed. When changing the lipid samples, capillaries were cleaned by pumping 70 % ethanol through the sample capillary followed by demineralized water and then removing residual water with a syringe.

The heat capacity measurements were performed using a VP-DSC MicroCalorimeter from MicroCal ™ Incorporated with tantalum cells. Since it is a differential scanning calorimeter, it has two cells, one for the sample and one for reference (MilliQ). The contribution of salts on their own is approximately $\approx \frac{5R}{2} \cdot 100\text{mM} \cdot 1\text{ml} = 0.0021 \frac{\text{J}}{\text{K}}$, so their contribution to the heat capacity of the solvent of the dispersion is considered negligible. Any difference between reference and sample cell is predominantly caused by ion interaction with the lipid dispersion and not the ions themselves. For each sample I would run 3 DSC scans at a scan rate of 5 Kelvin per hour: first for increasing, then decreasing and then again increasing temperatures. These scans are not always alike, see figure 20 and appendix E.6. Hysteresis can occur between up- and down-scans since the calorimeter isn't able to control cooling as well as heating, such that the system does not get to equlibrate properly at each temperature. Differences in between up-scans can be caused by changes in hydration of the lipids. Initially the samples might not be fully hydrated. Melting the lipids exposes the chains more to ions and water, which can cause changes in structure and transition temperatures that remain after transition back to gel state [9, 14]. The scans were run in high feedback mode, where the caloriemeter uses past heat inputs to calculate the next, which allows for higher resolution of the curve.

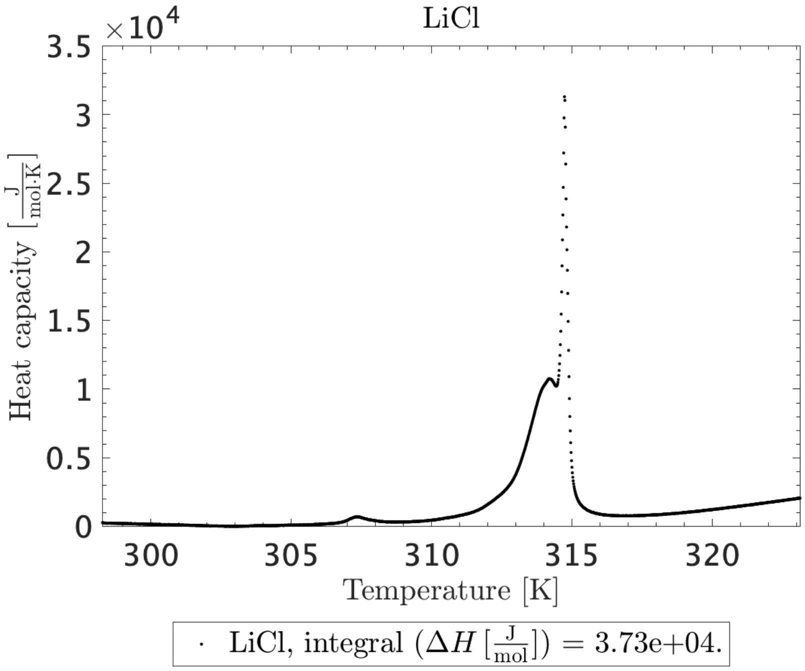


Figure 19: Example of raw DSC curve, LiCl sample.

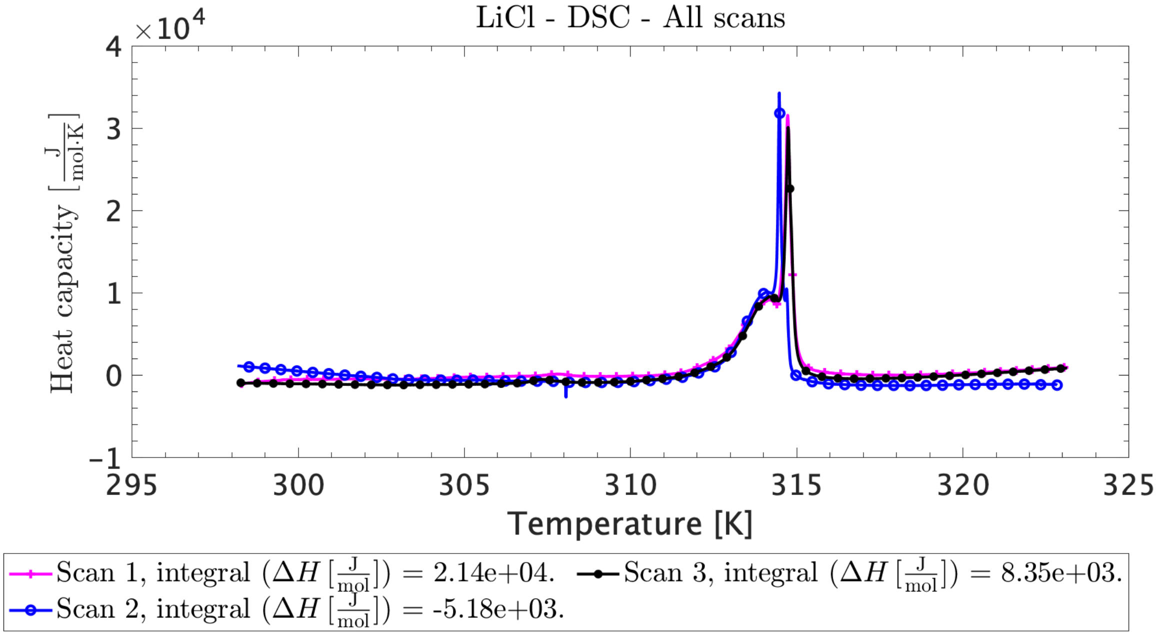


Figure 20: The two DSC up-scans (scan 1 and 3) and down-scan (scan 2) of the LiCl sample.

I ran the pressure pertubation calorimetry scan right after the DSC scan without removing the sample in question from the caloriemeter. The same manufacturer that has produced the calorimeter has also developed the extension to the calorimeter I used for pressure pertubation calorimetry. The device is connected by plastic tube to an air supply from Air Liquide S.A. at around 3 bars pressure. The device controls the pressure jump and release. It requires a different software setup to run, which allows for costumizable programs like the DSC scans. Since it is of most interest to investigate pressure pertubation close to melting temperature, I obtained approximate main- and pretransition temperatures from the previously obtained DSC scan of the same sample and focused the pressure pertubation runs in the determined range. I ran into problems when trying to initiate the PPC program for all but the last sample. The interface showed wildly fluctuating temperatures. I found it necessary to restart the computer and caloriemeter after changing from DSC to PPC mode, wait for a few hours and then return.

The time course of the DSC and PPC scans was as follows (year 2020): 30th of November to December 2nd, LiCl. 2nd of December to 4th of December, MilliQ. 4th of December to December 8th, KCl. 8th of December to 10th of December, NaCl.

It should be mentioned that this was not the only series of samples I ran, though it was the most succesful. I discarded a whole sample set after discovering how the lids (Chromacol 12-SCS 12 mm screw caps, solid top, lot# 42000979 from Thermo Scientific) of the vials would leave small pieces of plastic residue in the sample after unscrewing. This is a problem I have encountered in previous work, though I do not recall the manufacturer from before (bachelor's thesis). Before working with the LUV's I had conducted a series of experiments with MLV's. These show very high cooperativity and peaks magnitudes larger than that of LUVs (see also Heimburg [28]). When changing to LUV samples, I found a small contamination of residual MLVs at 314.6 K. After throrough cleaning some of the residue remained, which can be detected in the DSC and PPC runs of pure MilliQ water. I later use this measurement to remove any offset between the cells in my sample data. I suspect the contamination to have dissappeared over time with the changing content and cleaning of the sample cell, so I include the peak in the fit of the reference in order for the fit to converge, but omit the peak when subtracting the reference from the data. I did not run another water sample to verify the contamination had disappeared, since the integrated entalpy of this peak is about 2 orders of magnitude less than the sample entalpies at the same temperature, see the table in appendix D.

4.3.2 Data analyzation

DSC

I fitted temperature versus time with a linear fit and got scanrates of $5.01 - 5.04$ Kelvin per hour, which are in good agreement with the setting of 5 K/hr. I then converted all data to the SI units. The heat capacity per mole lipid is obtained by dividing the differential power by the scanrate and the number of lipids in the sample, in moles:

$$\Delta c_p = \frac{\text{differential power}}{\text{scan rate} \quad \text{x} \quad N_{mol}} \tag{36}$$

As mentioned in the previous section, there is visible differences between the three scans, though both up-scans look similar. I chose to use the 2nd up-scan for data analyzation, since up-scanning is least prone to hysteresis and the lipids should all be properly hydrated after having gone through the transition once. I fitted the reference with a 3d order polynomial describing the slow changing baseline and a Gaussian for the small contamination peak (see figure 21). By removing the contamination from the reference data, I was able to obtain a rough RMS of the noise on the data, see figure 22. There were still artifacts left from the contamination peak and the odd ends of the baseline had not been properly fitted, so I estimated the noise to be less than 9 $\frac{\text{J}}{\text{mol} \cdot \text{K}}$. I then subtracted the baseline polynomial from the four samples with no visible effect on the curves (compare fig. 19 to fig. 23).

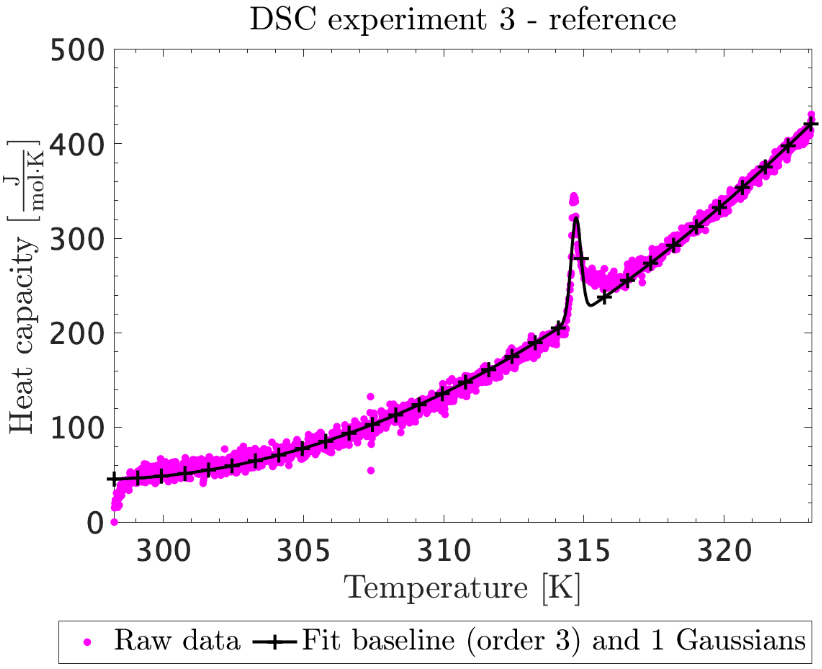


Figure 21: Fit of the slightly contaminated pure MilliQ sample using a polynomial of 3rd order and a Gaussian peak.

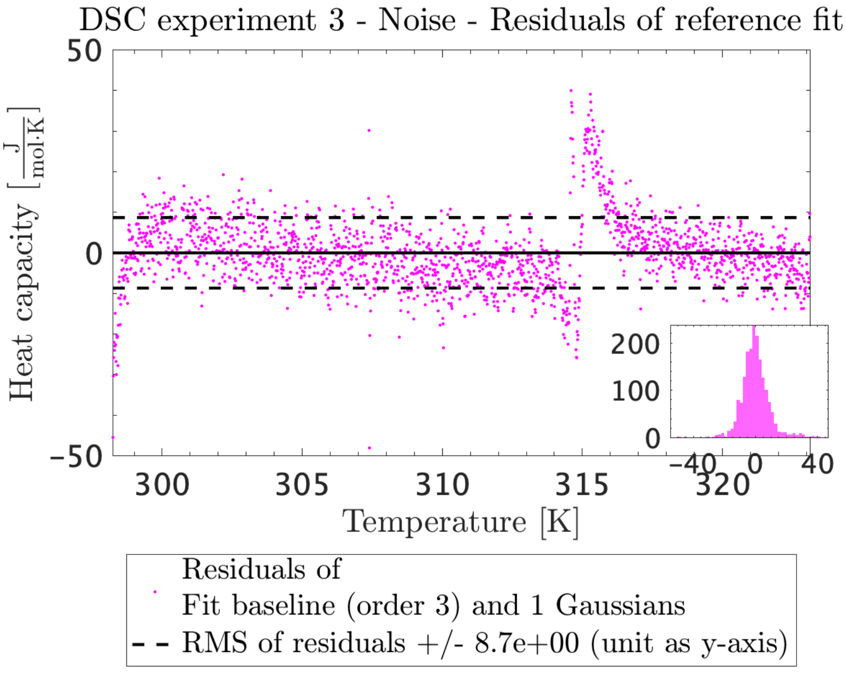


Figure 22: Size of noise: Residuals of the pure MilliQ water run when the data fit from figure 21 has been subtracted.

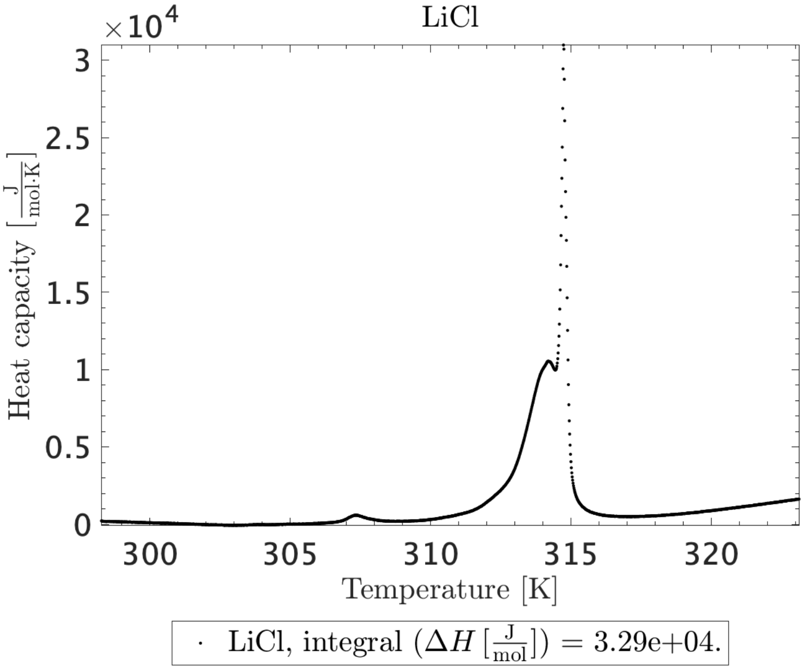


Figure 23: Example of data set where the offset fit has been subtracted. Hardly any difference can be seen.

Next step was to correct the baseline in the four samples. At first I omitted the melting region from the baseline fit and fitted the ends only, but then there were no points to guide the fit in the absent melting region. I decided to try to separate the melting behaviour from the baseline in the melting region and include the resulting points in the fit. Since the baseline is minute and slow changing compared to the extreme peaks of the lipid melting process in the four samples, it is difficult to fit peaks and baseline simultaneously as I did with the reference. I decided to fit the peaks and subtract them from the dataset before fitting the baseline using a 5 order polynomial. The data have all been normalized to be per mole of lipids. If we assume that all lipids pass through all of the transitions present in the melting profile, each of the peaks must contain a mole of lipids, which can be fitted using van't Hoff's law (see equation 23):

$$\Delta c_{\mathrm{p,lipid}} = \frac{n(\Delta H_0)^2}{RT^2}\frac{K(T)}{(K(T)+1)^2}, \qquad K(T) = \exp\left(-n\frac{\Delta H_0 - T\Delta S}{RT}\right)$$

Here n is the cooperativity of the peak and ΔH_0 is the melting entalpy per mole. An example of the full fit for the LiCl sample can be seen in fig. 24, and the individual fit components in fig. 25.

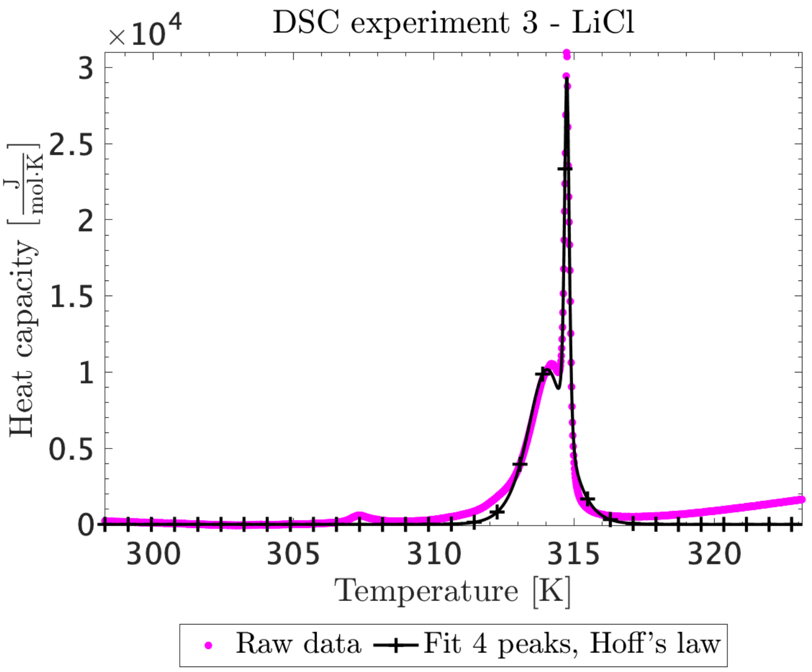


Figure 24: Full Van't Hoff fit of the DSC scan of the LiCl sample.

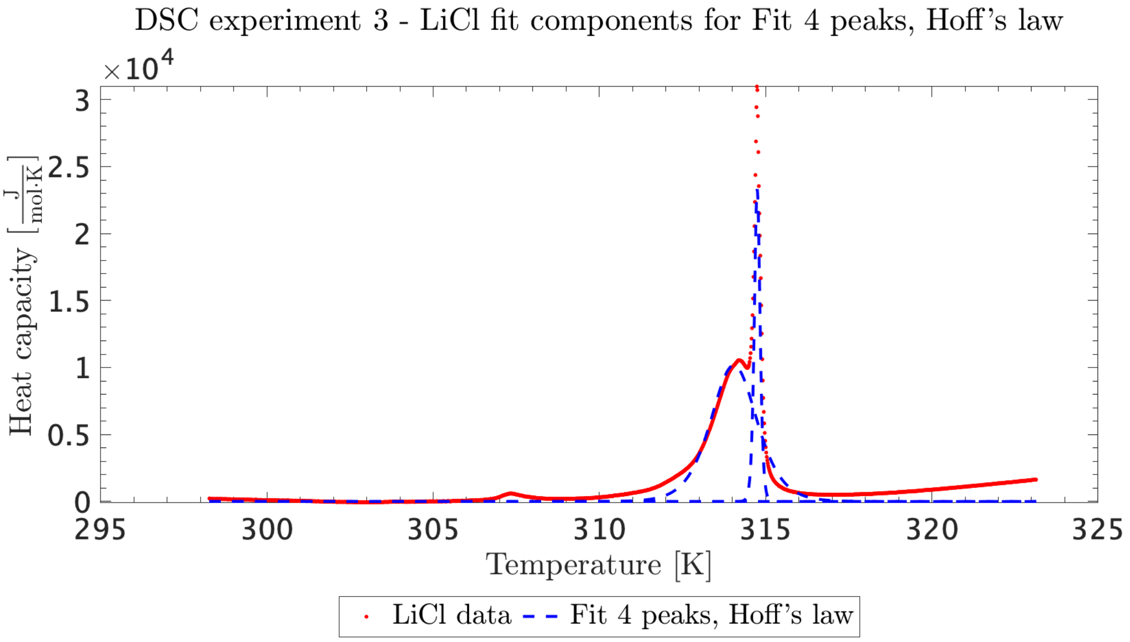


Figure 25: Fit components of the LiCl DSC data fit in figure 24. Since the convergence of MATLAB®'s fitting procedure fitnlm wasn't smooth, these plots helped me asses precise initial parameter guesses to ease the optimization.

Unfortunately the peaks weren't fitted well enough to be properly subtracted. At the tails of each peak, the discrepancies between the fit and the data were so large that they created artificial outliers in the data, see fig. 27. In hindsight it might have been better to manually remove peak data in order to avoid this problem. Instead I developed a weighing procedure to correct for this issue. It depended on two weight types, I named 'value' and 'position', in the following manner:

$$w = \sqrt{w_{\text{value}}^2 + w_{\text{position}}^2}$$

The value-weight is a penalty on data points with large heat capacities. The influence of a point decreases exponentially with increasing heat capacity value:

$$w_{\text{value}} = w_{\text{value,max}} \cdot \exp\left(a_{\text{value}} \cdot \text{abs}(\Delta c_p)\right)$$

Where the amplitude a_{value} is given by:

$$a_{\text{value}} = \frac{\log\left(\dfrac{w_0}{w_{\text{value,max}}}\right)}{\Delta c_{p,0}}, \quad \text{where} \quad \Delta c_{p,0} = 0.01 \cdot \Delta c_{p,\max} \tag{37}$$

$$\text{and} \quad w_0 = 0.01,\ w_{\text{value,max}} = 1 \tag{38}$$

The rate at which the exponential decays is adjusted by $\Delta c_{p,0}$.

The position-weight causes the end points (< 299 K and > 318K) to have the maximum influence of 1, since these points almost purely consist of the baseline. Since I see no reason to distinguish between the two ends, the weight is the same on both sides. From either end the weight then decreases exponentially towards a decided center point of lowest weight at 315 Kelvin, around main peak/peaks.

For either end,

$$w_{\text{position}} = \begin{cases} w_{\text{value,end}} \cdot \exp\left(a_{\text{position}} \cdot (T_{\text{center}} - T_{\text{end}})\right) \end{cases}$$

The amplitude for each end is calculated from:

$$a_{\text{position}} = \frac{\log\left(\frac{w_{\text{center}}}{w_{\text{end}}}\right)}{T_{\text{center}} - T_{\text{end}}}, \quad \text{where} \quad T_{\text{end}} = 299\ \text{K} \quad \text{if} \quad T \in [299\ \text{K}, 315\ \text{K}]$$

$$\text{or} \quad T_{\text{end}} = 318\ \text{K} \quad \text{if} \quad T \in [315\ \text{K}, 318\ \text{K}]$$

$$\text{and} \quad w_{\text{center}} = 0.001,\ w_{\text{end}} = 1$$

Fig. 26 shows the individual weights and the final product. The same weight parameters where used to fit all samples.

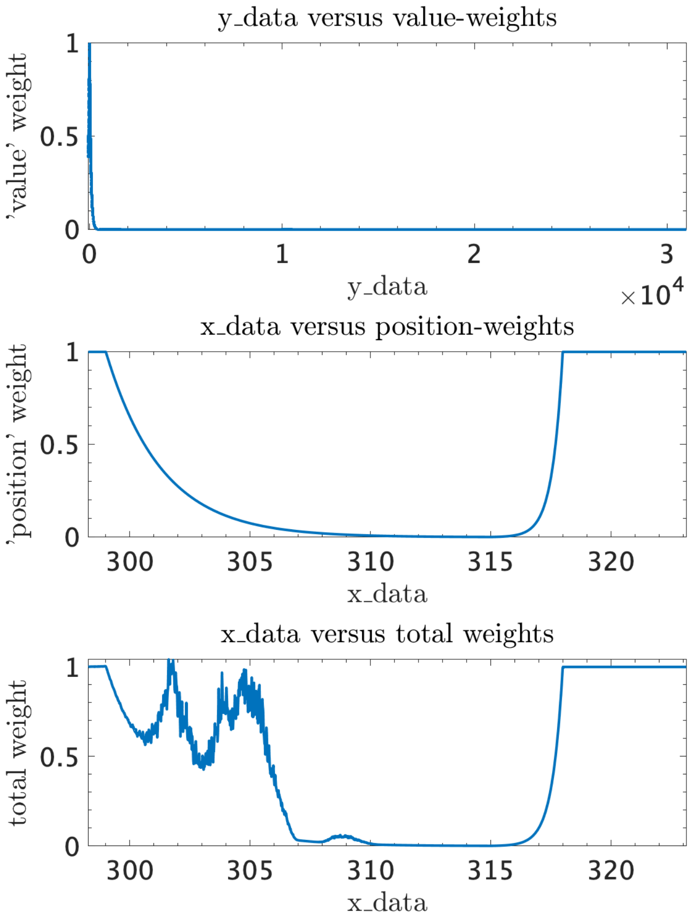


Figure 26: Weights used for fitting the baseline in figure 27.

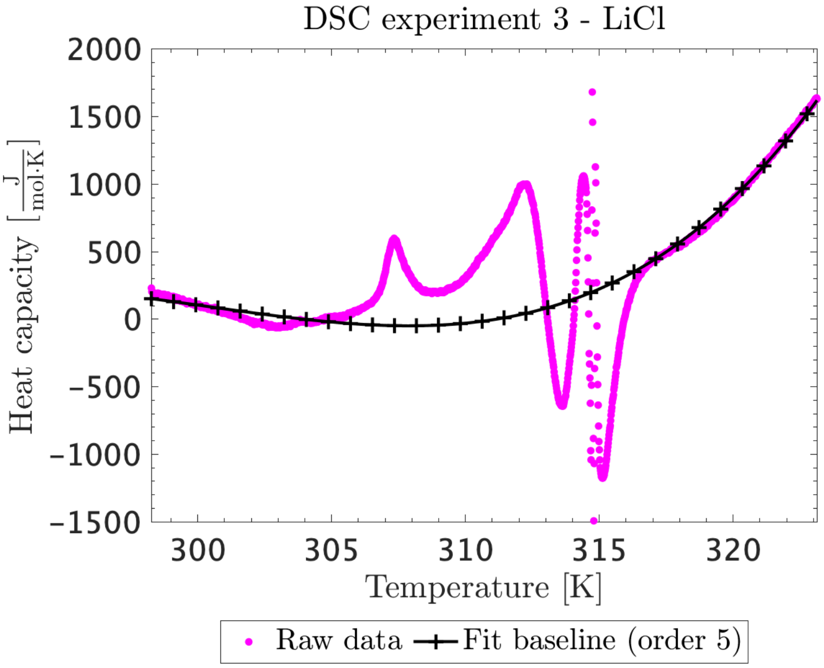


Figure 27: Fitted DSC baseline after subtracting the fit peaks seen in figure 25, LiCl sample. The negative heat capacity values are artifacts created by subtraction of inadequately fitted peaks.

Finally the fitted baseline was subtracted from the reference-corrected sample data, and the final plots can be seen in figure 28. The melting entalpies for the four samples were obtained by numerically integrating the heat capacity curves. The numbers vary within $28.1\frac{\text{kJ}}{\text{mol}} - 37.6\frac{\text{kJ}}{\text{mol}}$, which is on the lower side of what is reported for DPPC membranes in literature: $35 - 39.2\frac{\text{kJ}}{\text{mol}}$ (see appendix D). Usually one would expect elevated melting entalpies due to sample sedimentation in the cell from the compartment above [14]. Consistently lower entalpies are likely to be a fault in the sample preparation and handling, like adding too much solvent or that the lipids weren't properly in suspension when part of the sample is moved to the cell. I measured the melting temperatures, maximum entalpies and FWHM of the various peaks in the four plots manually using guides in GIMP as rulers. The values are listed in table 1 in appendix B. Each sample has its own characteristic profile, though all 4 samples shared 3 transitions at ≈ 314.6 K, ≈ 314 K, and ≈ 308 K. The transitions at ≈ 314.6 K, ≈ 314 K resembles an overlap between the sonicated LUV and MLV data from literature [28]. MLV's are highly cooperative membranes with very narrow transitions (widths less than ≈ 0.1K) that appear at slightly higher melting temperature than the main transition of the LUV [28]. The maximum heat capacity for a mole of MLV membranes in main transition is roughly 50 times larger than for sonicated LUVs. In comparison, the MLV peak is only 2.5 times larger than the LUV main transition in my own samples, so the transition at ≈ 314.6 K is likely caused by a very small residue of multilamellar vesicles left over from inadequate sonication.

The LUV main transitions for the ion samples all have distinguishable peaks as opposed to the pure water sample. The main transition peak of the sodium sample is especially apparent and has a notably lower melting temperature than the rest of the ion samples, indicating melting point depression. Unfortunately the pure water sample is clouded by the MLV contamination, so the control melting point hasn't been determined precisely enough for me to compare lithium and potassium. But it is likely that the way the ions set the main transition apart from the MLV contamination is by shifting the melting point of the main transition downwards.

All samples show a pre-transition around 308 K. The pre-transition causes a visible structure change in the gel bilayer from flat to periodic ripples [68, 22]. This transition has been documented to show similar proportionality between entalpy and volume changes as does the main transition [22], so it is assumed to be an initial step in the phase transition of lipid bilayers. The lithium sample has an especially well-defined pre-transition. Pure water, potassium and sodium samples all seem to have a separate transition going on in between pre-transition and main transition that obscures the individual peaks. I used the term 'sub-transition' for this area in the table 1 in appendix B (The term has been used before but not necessarily about the same phenomena [14]). Though it is inside of the melting region, the origin of the sub-transition is not clear. Not all features of the melting region are necessarily entalpy changes directly brought by chain melting. The reduced elastic constants and increased membrane area during transition increases the interaction and effect of ionic solvent on the lipid bilayer, which can bring about changes in membrane morphology [68].

Overall lithium, sodium and potassium all have visible effects on the melting behaviour. Lithium does not have as strong an effect on zwitterionic DPPC bilayers as in anionic membranes such as DPPS [24, 25]. Yet the heat capacity curve of the lithium sample stick out from the other ion samples by its simplicity, the melting region consists only of a main transition and pre-transition (apart from the MLV contamination). The sub-transition recorded in the pure water sample is gone. Schneider et al. [68] achieved a similar effect by increasing the chain length and thereby coherence of the lipids in an anionic DMPG bilayer. For short lipid chains the melting region consisted of a pre-transition, a main transition and a broad range of heat capacity features. The latter two were gradually gathered into a single peak as chain lengths increased, and this peak shifted towards higher temperatures. Longer chains increases membrane coherence, and increased membrane coherence is a hypothesized consequence of lithium's dehydrating abilities in anionic lipid bilayers [46]. It is possible that what is seen in the zwitterionic DPPC sample containing lithium is a remnant of the stabilizing, dehydrating effect on the lipid gel phase that has been documented in anionic lipids. The effect is not strong enough to move the phase transition out of experimental range, but enough to reduce auxiliary entalpy contributions from thermally induced structural changes during melting.

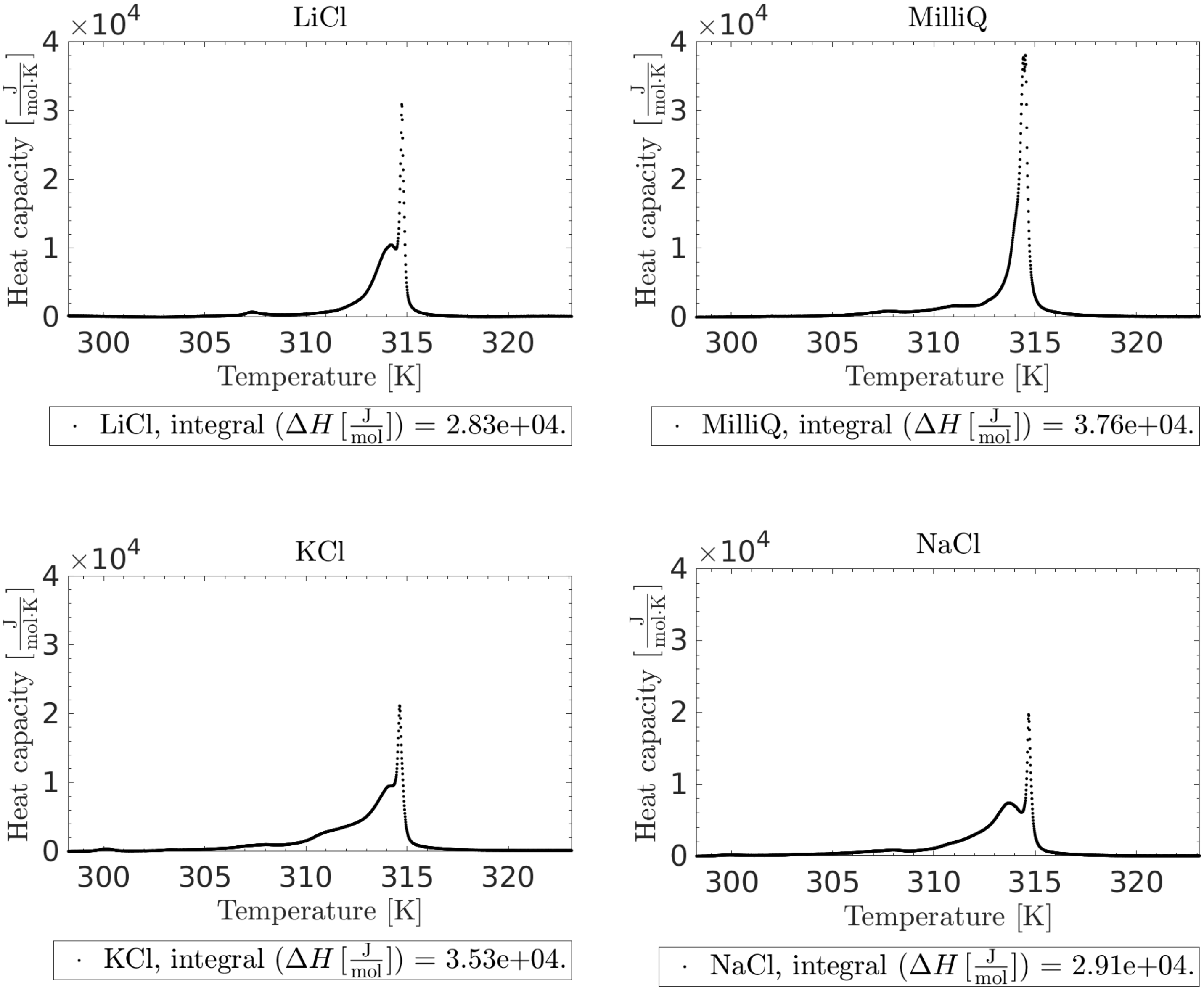


Figure 28: The final heat capacity profiles.

PPC

All data has been converted to SI units. I calculated the thermal volume expansion coefficient using the aforementioned relation between heat changes and pressure changes:

$$\alpha_V = -\frac{\Delta Q_{\mathrm{rev}}}{TV\Delta p} \tag{39}$$

I assume the average volume V of the lipids to be constant, since the gel and lipid specific volumes are comparable in size, $0.947 \cdot 10^{-3} \frac{\mathrm{m}^3}{\mathrm{kg}}$, and $0.999 \cdot 10^{-3} \frac{\mathrm{m}^3}{\mathrm{kg}}$ [28]. I thus don't have to estimate the lipid-gel fraction at every temperature. I use the volume of the gel phase without much reasoning. In the nerve the phase changes from fluid to solid going ≈ 85 % through the transition, so generally lipids might primarily be in the fluid state. The temperature used was the mean temperature after pressure pertubation.

An example of a full PPC run for the KCl sample is given in figure 29. The temperature does seem to respond to the pressure pertubation, dropping ≈ 0.02 K, which is ten-fold of the variation mentioned in the paper by Grabitz et al. [22] (though a similar drop in temperature can be spotted in their data, fig. 3). Cooling of the sample is a sign that the calorimeter cannot compensate for the heat changes and the system state might not be well defined during pertubation [22]. Yet the size of the drop, ≈ 0.02 K, seems to be very consistent over all runs, regardless of the the system's proximity to phase transition and thereby the size of the heat uptake. The new temperature is stable throughout the equilibration, varying ≈ 0.005K, thus the relaxation times are not likely to be affected by this. But the formula used for the expansion coefficients rely on the pressure pertubation being an isothermal process, and underestimation of the heat decreases the sizes of the coefficients. Conversely an underestimation of the temperature will increase the size of the coefficients. Whether this issue is related to the calibration malfunction mentioned in the previous section is uncertain.

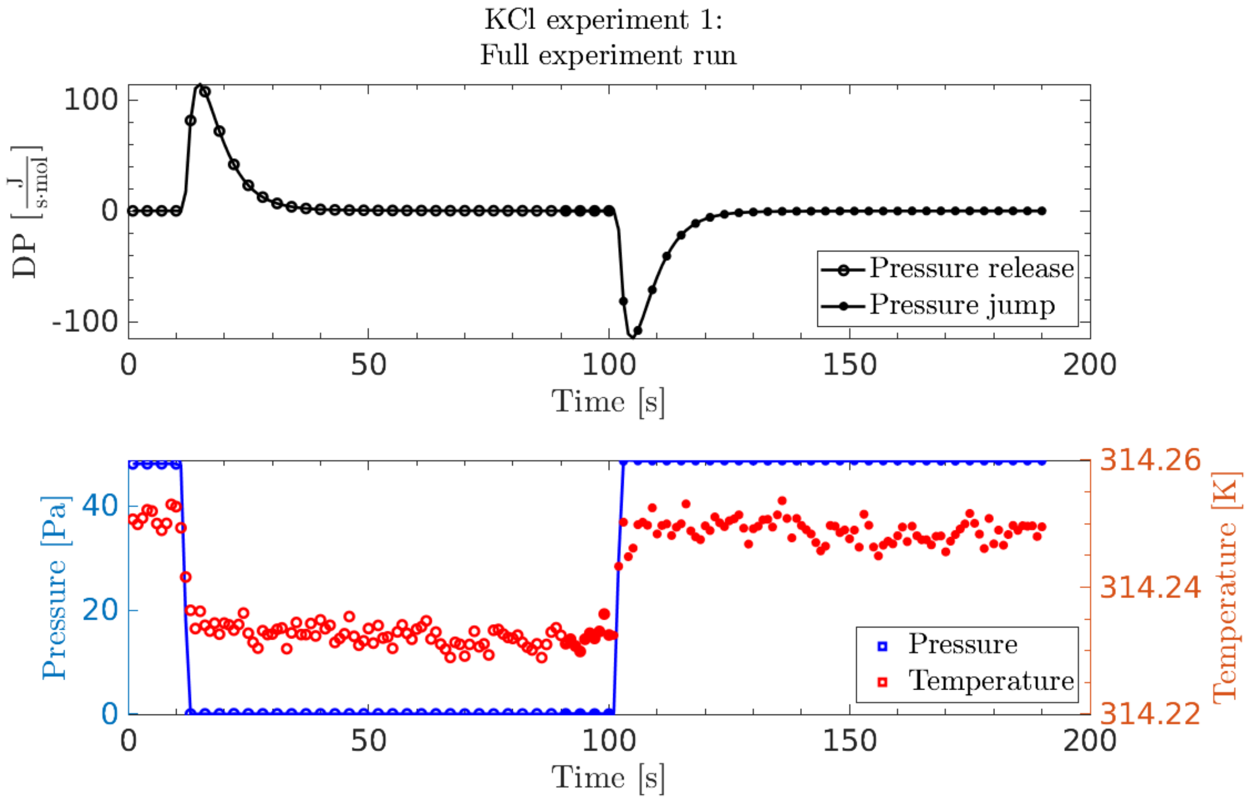

KCl experiment 1:
Full experiment run

Figure 29: Scan 42. The points marked 'pressure release' covers the pre-equilibration of the sample, the sudden drop in pressure and subsequent equilibration to the new state. The last points of the equilibration from the pressure release is included as the pre-equilibration in the 'pressure jump' data, that additionally covers the the sudden increase in pressure and following relaxation.

As in the case of the DSC down-scans of the previous section, the calorimeter is less able to compensate for a release of heat to the surroundings during pressure jump than heat uptake during pressure release [22], so I focus my analyzation on pressure releases. The lipid relaxation is not immediately available in the differential power data, instead the differential power signal is a convolution of the lipid heat transfer to the water reservoir (lipid relaxation), and the instrument response. I follow the deconvolution practices of Grabitz et al. [22], where it is assumed that the instrument response primarily consists of the heat transfer from the water to the cell wall of the calorimeter, and the heat transfer from cell wall to detector. The analytical representation of the convoluted signal, P_{exp}, is then:

$$P_{\text{exp}} = \int_{\tau=\text{pertbation time}}^{t} P_{\text{lipid}}(\tau) R_{\text{inst}}(t-\tau) d\tau + P_{\text{H}_2\text{O}}(t) \tag{40}$$

Where $P_{\text{lipid}}(\tau)$ is the instant lipid heat release at the time of pressure pertubation, decaying exponentially:

$$P_{\text{lipid}}(t) = \begin{cases} 0 \quad t < \text{pertubation time} \\ P^0_{\text{lipid}} \exp(-k_{\text{lipid}} t) \quad t \geq \text{pertubation time} \end{cases} \tag{41}$$

The relaxation time is the inverse of the rate of lipid relaxation, $\tau = (k_{\text{lipid}})^{-1}$. $R_{\text{inst}}(t)$ is the 'normalized instrument response function' modulating the lipid heat release:

$$R_{\text{inst}}(t) = \begin{cases} 0 \quad t < \text{pertubation time} \\ \frac{k_1 \cdot k_2}{-k_1 + k_2} \left(\exp(-k_1 t) - \exp(-k_2 t)\right) \quad \text{for} \quad t \geq \text{pertubation time} \end{cases} \tag{42}$$

Where k_1 is the rate of heat transfer from the water to the cell wall of the calorimeter, and the rate of heat transfer from cell wall to detector is k_2. Lastly, $P_{\text{H}_2\text{O}}(t)$ is the background water response to the pressure pertubation. Coefficients k_1 and k_2 are obtained by fitting a PPC run at a low temperature where the system is still outside of the lipid phase transition during pressure pertubation, so the lipid relaxation time is small and the lipid contribution to the heat uptake or release negligible.

$$P_{\text{H}_2\text{O}}(t) = \begin{cases} 0 \quad t < \text{pertubation time} \\ P^0_{\text{H}_2\text{O}} \left(\exp(-k_1 t) - \exp(-k_2 t)\right) \quad t \geq \text{pertubation time} \end{cases} \tag{43}$$

In figure 30 I've plotted the individual fit components and final signal for a PPC run at approximately 40.1° C.

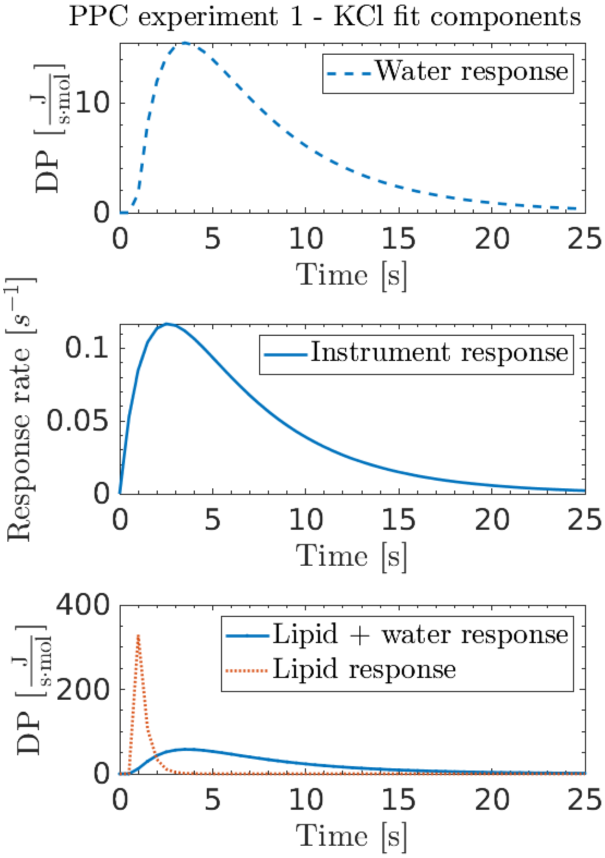


Figure 30: KCl sample, scan 32 deconvolution components $T = 40.1°$ C.

In figure 31 I have picked out a few scans to illustrate the variation in heat uptake and relaxation amplitudes through the temperature range. Not all fits converged as nicely as the ones presented. Currently only one relaxation process is allowed at a given temperature in the fit, but the heat capacity data from the DSC scans in fig. 28 indicate that the transition contains several overlapping processes.

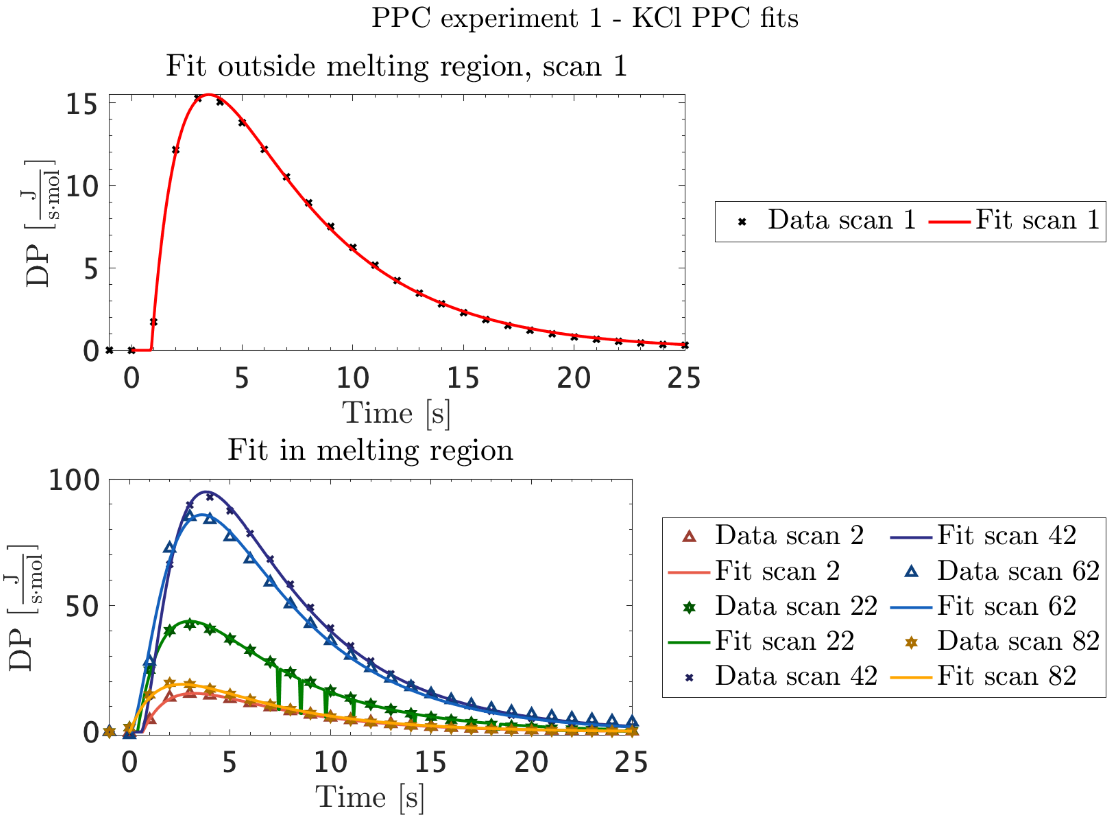


Figure 31: Examples of pressure releases Equilibration temperatures, scan 1: 25.3° C, scan 2: 26.0° C, scan 22: 39.0° C, scan 42: 41.1° C, scan 62: 41.6° C, scan 82: 42.3° C.

Proportionality

The LiCl sample show some kind of contamination resulting in a linear increment in the PPC data. I mitigate this by fitting the data with a linear fit and correct for the slope, but not the intersection, from the points. I fit points outside of the melting regime, since melting peaks would disrupt fit convergence and I learned from my previous mistakes not to subtract inadequately fitted peaks and create artifacts in the data. This correction does not have much influence on the proportionalities γ_V and L between heat capacity and thermal expansion coefficients and heat capacity and relaxation times respectively. Secondly I adjust the offset in all samples, a task that takes some preparation: Most of the temperature ranges of both DSC and PPC measurements are outside of the melting regime, where the data does not match up well due to the precision of the calorimeter and resolution of the deconvolution procedures. I therefore define a 'range of proportionality' for each constant of each sample, and the offset is calculated as the mean of the two points at the boundaries of this interval. Correcting for the offset has a large impact on γ_V and L for the lithium sample. I scale the curves of thermal volume expansion coefficients and relaxation times to match the heat capacity profile within the proportionality range defined for each the given data- and sample type. The final plots of the measured thermal expansion coefficients and relaxation times and their proportionalities can be seen in fig. 32 and fig. 33. The proportionality constants γ_V and L are calculated within the range of proportionality. Since the temperature ranges don't align, I bin the data every 0.1 K and plot γ_V and L as a function of temperature to see if they are truly constant. The plots are included in in appendix E.7 and appendix E.8. I readjust the range of proportionality for each sample based on these plots and rerun the data analyzation. From the look of it I am not convinced that γ_V and L are actually constant during the main transition. γ_V consistently go through one or two local minima reflecting the peaks of the main transition and the MLV contamination in the thermal expansion coefficient data. The variations surrounding the minima are much larger than the minima though. The L-data is more noisy than the γ_V-data (reflecting the variation in the relaxation times), but still it displays a single maximum corresponding to the main transition and MLV peak in the relaxation data. There are effectively no other quantities involved in the proportionality between Δc_p and thermal expansion coefficients or relaxation time than temperature, since the average volume included in the expression for the thermal expansion coefficient cancels out in the proportionality with Δc_p, see section 4.3. Furthermore the issue is likely caused by a shared dependency, since both constants display similar features. If this is an artifact, it could be linked to the experimental issue involving temperature measurements. For each sample I calculate the mean value and standard deviation of γ_V and L across the proportionality range, which are the numbers included in figure 32 and figure 33.

The thermal expansion coefficients follows the major features of the heat capacity curves in all four samples. Pure water, sodium and potassium samples show good proportionality throughout the whole of the melting region, from pre-transition past the MLV contamination. The lithium sample is less adherent, here the thermal expansion coefficients undershoots the heat capacity curve at temperatures lower than the main transition. I previously noted the consistent temperature change of ≈ 0.02 K during pressure release, see figure 29. Apparently the underlying error is consistent enough that any underestimation of the heat uptake or temperature doesn't disturb the proportionality visibly. All 4 proportionality constants, γ_V, within range of each other and similar to previously measured [28, 69]. Of the relaxation data, the lithium sample shows the best adherence with respect to the heat capacity of all the samples. It is also the only sample without a visible sub-transition in the heat capacity profile. It could be that the sub-transition relaxes at a different rate than the main transition and MLV contamination, such that the lower temperature melting region cannot be fitted with a single exponential relaxation. The phenomenological constants L are in the order of $\sim 10^8 \frac{\mathrm{J \cdot K}}{\mathrm{s \cdot mol}}$ to $\sim 10^9 \frac{\mathrm{J \cdot K}}{\mathrm{s \cdot mol}}$ which also aligns with previous studies [70] and [22, errata].

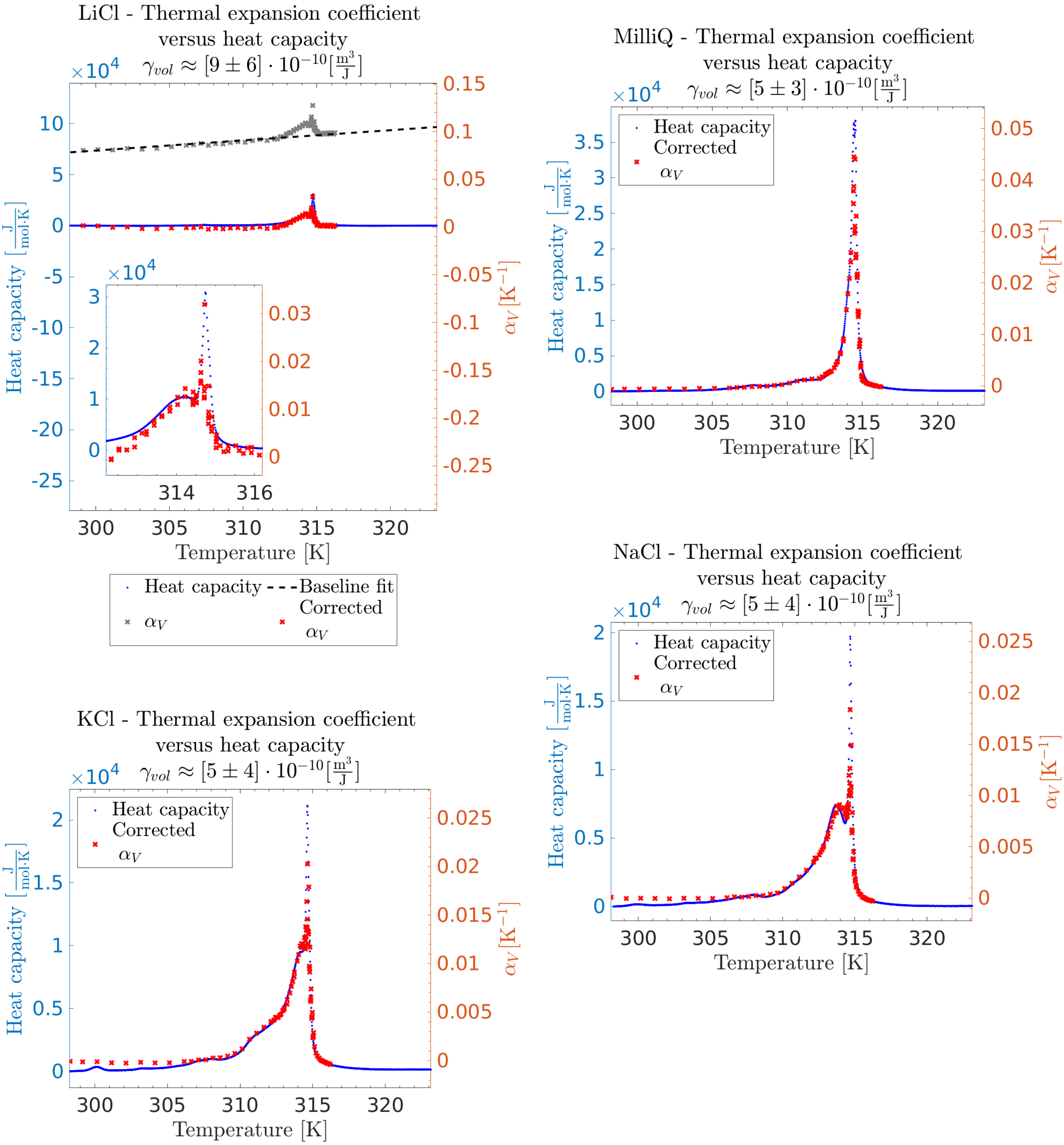


Figure 32: Thermal volume expansion coefficients of lithium, sodium, potassium and control samples measured with PPC. The raw lithium data is plotted along with the fit of the increment (grey points, black dotted line).

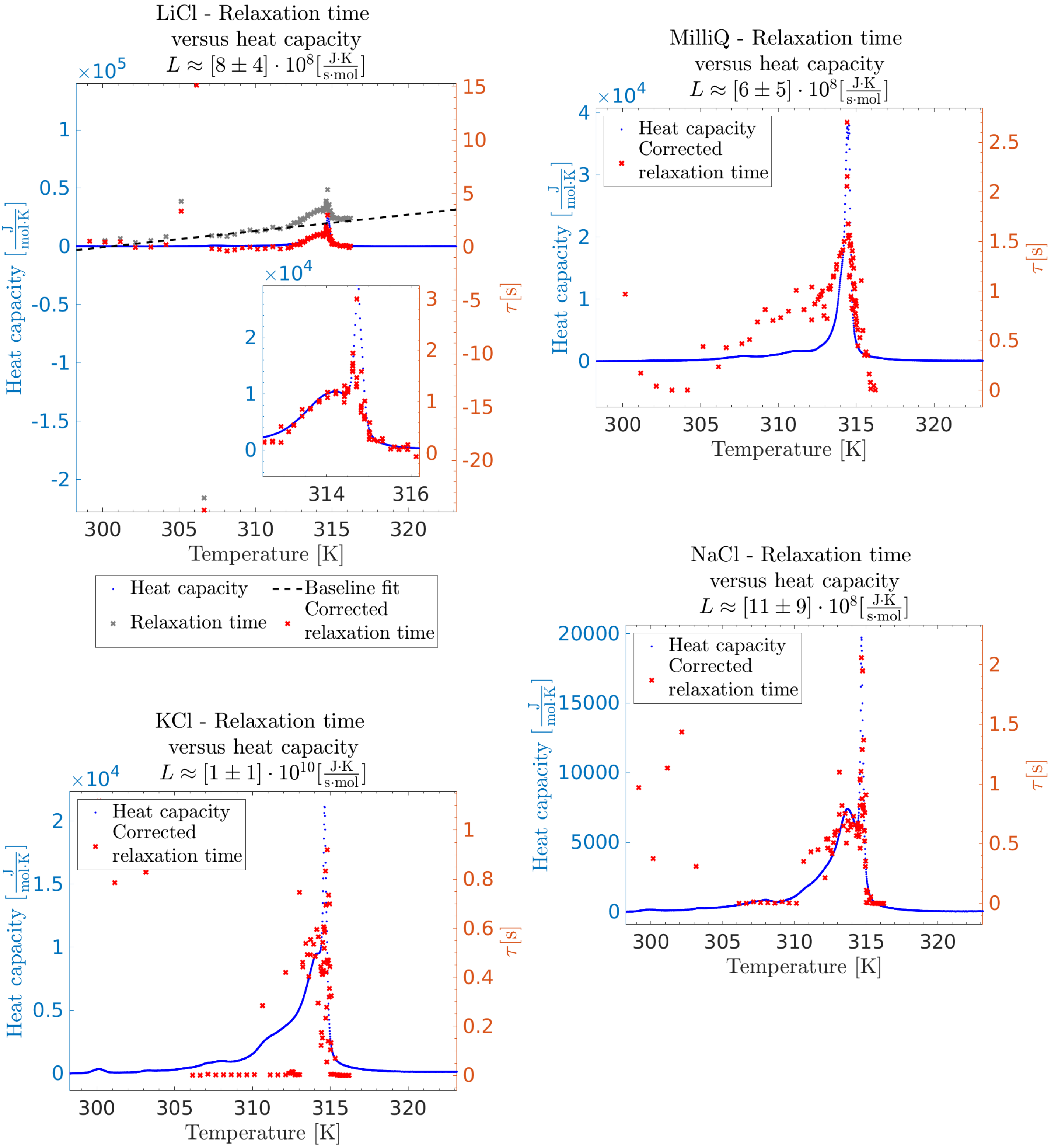


Figure 33: Relaxation times of lithium, sodium, potassium and control samples measured with PPC. The raw lithium data is plotted along with the fit of the increment (grey points, black dotted line).

4.4 Discussion

All four samples have their own distinguishable features, which is most evident for the heat capacity profiles and thermal volume expansion coefficients. During melting transition, changes in membrane area and morphology [68] increases the exposure to the solvent, so it makes sense that the differences in membrane interaction between ion species are especially evident during transition.

The lack of auxiliary melting features in the heat capacity of the lithium sample is similar what is seen in anionic DMPG bilayers when the chain length of the lipid is increased, or when sodium ions (0-500 mM, 7.5 pH) are added [68]. As chain length or sodium concentration is increased, the broad and complex melting profile of pure DMPG is stepwise reduced to a localized pre- and main transition. It would be interesting to reproduce results for DMPG using lithium instead of sodium, and conduct a similar series of experiments in zwitterionic lipids using LUV's made from DLPC, DMPC, DPPC and DSPC (chain length and melting temperature increases to the right). This would hopefully lead to a more thorough understanding of the effect of lithium on anionic and zwitterionic lipid membranes, especially if compared with identical experiments involving sodium and possibly calcium.

In section 4.2 I mention a study showing how lithium reduces and almost extinguishes the recovery heat in nerves. The recovery heat has been attributed to metabolism in literature [32]. It is possible that lithium toxicity abolishes the recovery heat by killing off metabolism in ways unrelated to the disorder of the lipid membranes. Yet lithium's extraordinary stabilizing effect on the gel-phase of anionic lipid bilayers, and it's potential ability to remove additional transition peaks of zwitterionic lipid bilayers makes me wonder whether the recovery heat is related to thermodynamic membrane fluctuations, such as curvature changes of the membrane, and these effects therefore disappear with additional lithium-induced membrane coherence. Yet I am comparing a single experiment to a single study so results need to be reproduced and further investigated before a decent hypothesis can be presented.

One should be careful with extrapolating the prophylaxis of lithium in bipolar disorder and other ailments from its immediate effect on lipid membranes or nerve excitation. López-Corcuera et al. [47] conducted an experiment on long-term lithium treatment in rats, showing how neural plasma membranes becomes disordered after chronic lithium treatment at therapeutic levels, partly due to changes in membrane saturation, which increases the membrane fluidity. If these findings are reproducible, lithium might be provoking a metabolic response reversing its own effects. Changes in lipid metabolism during long-term lithium treatment such as lipid peroxidation, [63, 48] and inositol breakdown inhibition [55, 12, 47, 48, 41] has been discussed extensively in literature.

The idea of changes in membrane fluidity as a metabolic response to lithium exposure is interesting in relation to co-administration of other drugs such as ouabain and anesthetics. Studies have shown how pre-administration of lithium decreases the toxicity of ouabain in rat hippocampal neurons and living rats [15, 45], and ouabain has been proposed as a model for bipolar disorder in rats for this reason. If lithium is administered shortly before ouabain in HeLa cell cultures, the cells' ability to bind ouabain is decreased. Conversely long-term lithium administration causes the cells to bind more ouabain [7]. On a slightly different, but possibly related note, the authors on the previously mentioned study on nerve heat [38] found ouabain to also abolish the recovery heat. One might suspect ouabain and lithium interact with the nerve membrane in a similar manner, so long-term administration of one of the drugs leads to increased tolerance of both. The opposite mechanism might be behind the drug interaction between lithium and anesthetics. Anesthesia stabilizes the fluid state through melting point depression [34, 30], whereas the immediate effect of lithium is to stabilize the solid state, and one would therefore expect an increased dose of anesthesia is needed in patients in chronic lithium treatment. Yet lithium reduces the requirements of anesthetics [62]. This makes sense if lithium induces a metabolic response increasing the membrane fluidity.

5 Conclusion

I have provided solitary solutions to the self-contained soliton model in case of a pure DPPC bilayer. The widths of the solutions are approximately 1 meter and maximum density changes relative to equilibrium is of $\approx 8\%$. I also reproduced the solutions of the soliton in case of the constant dispersion term, and found a maximum relative density of $\approx 20\%$ in agreement with previous studies. The maximum relative density in case of constant dispersion is independent of the value of the dispersion constant, so I can conclude that in general the amplitudes are quite larger in case of constant dispersion than the self-contained model, despite using the same parameters. Furthermore I solved a not quite self-contained soliton model, where the heat capacity term in the dispersion term of the self-contained soliton model had been approximated from experimental data of DPPC. With this model I obtained widths of approximately 0.5 m, and maximum relative density changes of $\approx 17\%$. The differences between the self-contained model and the pseudo-self-contained model is due to the first having a slightly more cooperative transition than the other. Both cases are wide in comparison to experimental data from nerves. Pure DPPC bilayers have a much higher cooperativity than heterogeneous biological membranes, that consists of various lipid species [60, 69] and proteins [30] contributing to the disorder of the membrane, which broadens the phase transition and shifts the melting point towards lower temperatures. It would be of great interest to solve the self-contained model in case of a more realistic membrane, unfortunately I wasn't able to change the cooperativeness of my model by much due to instability, let alone run the parameters for an e. coli membrane. In order to do be successful it will be necessary to improve my method, possibly by following the suggestions I've mentioned in the discussion.

I have presented literature suggesting a potential connection between the curative effect of lithium in bipolar disorder and the thermodynamics of lithium interaction with the lipid bilayer of the nerve membrane within the framework of the soliton theory. Most of the studies investigating membrane properties in presence of lithium were quite old, so it would be best to reproduce their findings using modern day equipment. I investigated the effect of lithium, sodium, potassium on the heat capacities, thermal volume expansion coefficients and relaxation times of zwitterionic DPPC LUVs suspended in water. For three ion species and control I found the empirical proportionality between heat capacity thermal volume expansion coefficients to be within the range of literature with values of $5 \cdot 10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}} - 9 \cdot 10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}}$ and standard deviation of $3 \cdot 10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}} - 6 \cdot 10^{-10}\,\frac{\mathrm{m}^3}{\mathrm{J}}$. From the relationship between relaxation times and heat capacities I determined the Onsager's phenomenological constant for each sample, they were all in the magnitude of $10^{8}\,\frac{\mathrm{J \cdot K}}{\mathrm{s \cdot mol}} - 10^{10}\,\frac{\mathrm{J \cdot K}}{\mathrm{s \cdot mol}}$, which is in agreement with literature. The sample containing lithium had a notably different heat capacity profile. Lithium removed auxiliary transition features from the melting profile, so that pre- and main transition appeared much clearer, an effect similar to increasing the chain length of the lipid. There was no remarkable shift in melting temperature as reported for anionic lipid membranes in literature, but part of lithium's ability to induce order is likely present in zwitterionic lipid membranes. Lithium's ability to induce order and stabilize the gel state of the lipid membrane will likely move the nerve membrane closer to transition at physiological temperatures, which will directly affect the propagation of solitons. Yet one cannot tell the full story without accounting for the metabolic response to long term treatment of lithium. Overall the documentation on lithium's interaction with lipids and living membranes is still too scarce to build a thorough hypothesis for the thermodynamic prophylaxis of lithium, but it is a promising area for further research.

Thermodynamically coupled pulses are not necessarily restricted to nerve cells. Action potentials exists in other excitable tissues in the body, such as muscles. Keynes and Swan [43] have documented increased permeability during the action potential spike in frog muscle tissue, similarly to the nerve membrane. E. coli membranes have been documented to undergo melting transitions, accompanied by an increase in transport rates [82], and there has been observed cell deformation during action potential in plant cells [20]. Nerve pulses might in that sense only be a highly optimized version of a basic cellular mechanism common to most living organisms. The soliton theory thus has many exciting prospects extending further than nerve signaling if one only accepts its physical premises and resists the temptation to merge it with currently accepted nerve models as what is done by Engelbrecht et al. [18] and Tamm et al. [73].

References

1. B. Abbott, A. V. Hill, and J. Howarth. The positive and negative heat production associated with a nerve impulse. *Proceedings of the Royal Society of London. Series B-Biological Sciences*, 148(931):149–187, 1958.

2. F. Anthony, R. Biltonen, and E. Freire. Modification of a vibrating-tube density meter for precise temperature scanning. *Analytical Biochemistry*, 116(1):161–167, 1981. ISSN 0003-2697. doi: 10.1016/0003-2697(81)90339-0. URL `https://www.sciencedirect.com/science/article/pii/0003269781903390`.

3. V. F. Antonov, V. V. Petrov, A. A. Molnar, D. A. Predvoditelev, and A. S. Ivanov. The appearance of single-ion channels in unmodified lipid bilayer membranes at the phase transition temperature. *Nature*, 283 (5747):585–586, feb 1980. doi: 10.1038/283585a0. URL `https://doi.org/10.1038%2F283585a0`.

4. R. Appali, B. Lautrup, T. Heimburg, and U. van Rienen. Soliton collision in biomembranes and nerves- a stability study. In *Scientific Computing in Electrical Engineering SCEE 2010*, pages 205–212. Springer Berlin Heidelberg, oct 2011. doi: 10.1007/978-3-642-22453-9_22. URL `https://doi.org/10.1007%2F978-3-642-22453-9_22`.

5. H. Binder and O. Zschörnig. The effect of metal cations on the phase behavior and hydration characteristics of phospholipid membranes. *Chemistry and Physics of Lipids*, 115(1):39–61, 2002. ISSN 0009-3084. doi: 10.1016/S0009-3084(02)00005-1. URL `https://www.sciencedirect.com/science/article/pii/S0009308402000051`.

6. A. Blicher. *Electrical Aspects of Lipid Membranes.* PhD thesis, Niels Bohr Institutet, 2011.

7. L. Boardman, S. Hume, J. Lamb, and J. Polson. Effect of growth in lithium on ouabain binding, Na-K-ATPase and Na and K transport in hela cells. *The Journal of Physiology*, 244, 1975.

8. J. F. Cade. Lithium salts in the treatment of psychotic excitement (reprint from 1949). *Bull World Health Organ*, 78(4):518–20, 2000.

9. G. Cevc, J. M. Seddon, and D. Marsh. Thermodynamic and structural properties of phosphatidylserine bilayer membranes in the presence of lithium ions and protons. *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 814(1):141–150, 1985. ISSN 0005-2736. doi: 10.1016/0005-2736(85)90429-8. URL `https://www.sciencedirect.com/science/article/pii/0005273685904298`.

10. G. Cevc, J. M. Seddon, and D. Marsh. The mechanism of regulation of membrane phase behaviour, structure and interactions by lipid headgroups and electrolyte solution. *Faraday Discussions of the Chemical Society*, 81:179, 1986. doi: 10.1039/dc9868100179. URL `https://doi.org/10.1039%2Fdc9868100179`.

11. B. Cunningham, J. Shimotake, W. Tamura-Lis, T. Mastran, W.-M. Kwok, J. Kauffman, and L. Lis. The influence of ion species on phosphatidylcholine bilayer structure and packing. *Chemistry and Physics of Lipids*, 39(1):135–143, 1986. ISSN 0009-3084. doi: 10.1016/0009-3084(86)90107-6. URL `https://www.sciencedirect.com/science/article/pii/0009308486901076`.

12. A. H. Drummond. Lithium and inositol lipid-linked signalling mechanisms. *Trends in Pharmacological Sciences*, 8(4):129–133, 1987. ISSN 0165-6147. doi: 10.1016/0165-6147(87)90181-7. URL `https://www.sciencedirect.com/science/article/pii/0165614787901817`.

13. H. Ebel, P. Grabitz, and T. Heimburg. Enthalpy and volume changes in lipid membranes. I. The proportionality of heat and volume changes in the lipid melting transition and its implication for the elastic constants. *The Journal of Physical Chemistry B*, (30):7353–7360. doi: 10.1021/jp010515s.

14. K. K. Eklund, I. S. Salonen, and P. K. Kinnunen. Monovalent cation dependent phase behaviour of dipalmitoylphosphatidylglycerol. *Chemistry and Physics of Lipids*, 50(1):71–78, 1989. ISSN 0009-3084. doi: 10.1016/0009-3084(89)90028-5. URL `https://www.sciencedirect.com/science/article/pii/0009308489900285`.

15. R. El-Mallakh, A. Schurr, R. Payne, and R. Li. Ouabain induction of cycling of multiple spike responses in hippocampal slices is delayed by lithium. *Journal of psychiatric research*, 34:115–20, 03 2000. doi: 10.1016/S0022-3956(99)00045-X.

16. J. Engelbrecht, K. Tamm, and T. Peets. On mathematical modelling of solitary pulses in cylindrical biomembranes. *Biomechanics and Modeling in Mechanobiology*, 14(1):159–167, may 2014. doi: 10.1007/s10237-014-0596-2. URL `https://doi.org/10.1007%2Fs10237-014-0596-2`.

17. J. Engelbrecht, K. Tamm, and T. Peets. On solutions of a Boussinesq-type equation with displacement-dependent nonlinearities: the case of biomembranes. *Philosophical Magazine*, 97(12):967–987, jan 2017. doi: 10.1080/14786435.2017.1283070. URL `https://doi.org/10.1080%2F14786435.2017.1283070`.

18. J. Engelbrecht, K. Tamm, and T. Peets. Primary and secondary components of nerve signals. *arXiv: Biological Physics*, 2018.

19. Eric H. Chudler. The hows, whats and whos of neuroscience. `https://faculty.washington.edu/chudler/what.html`, 2021-06-24 .

20. C. Fillafer, M. Mussel, J. Muchowski, and M. F. Schneider. Cell surface deformation during an action potential. *Biophysical Journal*, 114(2):410–418, jan 2018. doi: 10.1016/j.bpj.2017.11.3776. URL `https://doi.org/10.1016%2Fj.bpj.2017.11.3776`.

21. A. Gonzalez-Perez, R. Budvytyte, L. D. Mosgaard, S. Nissen, and T. Heimburg. Penetration of action potentials during collision in the median and lateral giant axons of invertebrates. *Phys. Rev. X*, 4:031047, Sep 2014. doi: 10.1103/PhysRevX.4.031047. URL `https://link.aps.org/doi/10.1103/PhysRevX.4.031047`.

22. P. Grabitz, V. P. Ivanova, and T. Heimburg. Relaxation kinetics of lipid membranes and its relation to the heat capacity. *Biophysical Journal*, 82(1):299–309, 2002. ISSN 0006-3495. doi: 10.1016/S0006-3495(02)75395-2. URL `https://www.sciencedirect.com/science/article/pii/S0006349502753952`.

23. S. Halstenberg, T. Heimburg, T. Hianik, U. Kaatze, and R. Krivanek. Cholesterol-induced variations in the volume and enthalpy fluctuations of lipid bilayers. *Biophysical Journal*, 75(1):264–271, 1998. ISSN 0006-3495. doi: 10.1016/S0006-3495(98)77513-7. URL `https://www.sciencedirect.com/science/article/pii/S0006349598775137`.

24. H. Hauser and G. Shipley. Crystallization of phosphatidylserine bilayers induced by lithium. *Journal of Biological Chemistry*, 256(22):11377–11380, nov 1981. doi: 10.1016/s0021-9258(19)68404-8. URL `https://doi.org/10.1016%2Fs0021-9258%2819%2968404-8`.

25. H. Hauser and G. G. Shipley. Interactions of monovalent cations with phosphatidylserine bilayer membranes. *Biochemistry*, 22(9):2171–2178, apr 1983. doi: 10.1021/bi00278a018. URL `https://doi.org/10.1021%2Fbi00278a018`.

26. M. T. Heath. *Scientific Computing - An Introductory Survey.* McGraw-Hill, 2nd, international edition, 2002.

27. H. Heerklotz and J. Seelig. Application of pressure perturbation calorimetry to lipid bilayers. *Biophysical Journal*, 82(3):1445–1452, 2002. ISSN 0006-3495. doi: 10.1016/S0006-3495(02)75498-2. URL `https://www.sciencedirect.com/science/article/pii/S0006349502754982`.

28. T. Heimburg. Mechanical aspects of membrane thermodynamics. Estimation of the mechanical properties of lipid membranes close to the chain melting transition from calorimetry. *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 1415(1):147–162, 1998. ISSN 0005-2736. doi: 10.1016/S0005-2736(98)00189-8. URL `https://www.sciencedirect.com/science/article/pii/S0005273698001898`.

29. T. Heimburg. *Thermal Biophysics of Membranes.* Wiley-VCH, 2007, ISBN: 978-3-527-40471-1.

30. T. Heimburg. Lipid ion channels. *Biophysical Chemistry*, 150(1-3):2–22, aug 2010. doi: 10.1016/j.bpc.2010.02.018. URL `https://doi.org/10.1016%2Fj.bpc.2010.02.018`.

31. T. Heimburg. The physics of nerves. *arXiv: Biological Physics*, 2010.

32. T. Heimburg. The important consequences of the reversible heat production in nerves and the adiabaticity of the action potential. *Progress in Biophysics and Molecular Biology*, 2020. ISSN 0079-6107. doi: 10.1016/j.pbiomolbio.2020.07.007. URL `https://www.sciencedirect.com/science/article/pii/S0079610720300729`.

33. T. Heimburg and A. D. Jackson. On soliton propagation in biomembranes and nerves. *Proceedings of the National Academy of Sciences*, 102(28):9790–9795, 2005. ISSN 0027-8424. doi: 10.1073/pnas.0503823102. URL `https://www.pnas.org/content/102/28/9790`.

34. T. Heimburg and A. D. Jackson. On the action potential as a propagating density pulse and the role of anesthetics. *Biophysical Reviews and Letters*, 02(01):57–78, jan 2007. doi: 10.1142/s179304800700043x. URL `https://doi.org/10.1142%2Fs179304800700043x`.

35. A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4):500–544, aug 1952. doi: 10.1113/jphysiol.1952.sp004764. URL `https://doi.org/10.1113%2Fjphysiol.1952.sp004764`.

36. S. E. Horvath and G. Daum. Lipids of mitochondria. *Progress in Lipid Research*, 52(4):590–614, 2013. ISSN 0163-7827. doi: 10.1016/j.plipres.2013.07.002. URL `https://www.sciencedirect.com/science/article/pii/S0163782713000519`.

37. Hosted by Thomas R. Heimburg. Dr. Konrad Kaufmann: Publications and manuscripts. `https://www.nbi.ku.dk/membranes/Kaufmann/publications.html`, 2021-06-24 .

38. J. V. Howarth, R. D. Keynes, and J. M. Ritchie. The origin of the initial heat associated with a single impulse in mammalian non-myelinated nerve fibres. *The Journal of Physiology*, 194(3):745–793, 1968. doi: 10.1113/jphysiol.1968.sp008434. URL `https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.1968.sp008434`.

39. K. Iwasa and I. Tasaki. Mechanical changes in squid giant axons associated with production of action potentials. *Biochemical and Biophysical Research Communications*, 95(3):1328–1331, Aug 1980. ISSN 0006-291X. doi: 10.1016/0006-291x(80)91619-8. URL `http://dx.doi.org/10.1016/0006-291X(80)91619-8`.

40. K. Jacobson and D. Papahadjopoulos. Phase transitions and phase separations in phospholipid membranes induced by changes in temperature, pH, and concentration of bivalent cations. *Biochemistry*, 14(1):152–161, jan 1975. doi: 10.1021/bi00672a026. URL `https://doi.org/10.1021%2Fbi00672a026`.

41. E. Jakobsson, O. Argüello-Miranda, S.-W. Chiu, Z. Fazal, J. Kruczek, S. Nunez-Corrales, S. Pandit, and L. Pritchet. Towards a unified understanding of lithium action in basic biology and its significance for applied biology. *The Journal of Membrane Biology*, 250(6):587–604, nov 2017. doi: 10.1007/s00232-017-9998-2. URL `https://doi.org/10.1007%2Fs00232-017-9998-2`.

42. F. H. Johnson and E. A. Flagler. Hydrostatic pressure reversal of narcosis in tadpoles. *Science*, 112(2899): 91–92, jul 1950. doi: 10.1126/science.112.2899.91-a. URL `https://doi.org/10.1126%2Fscience.112.2899.91-a`.

43. R. D. Keynes and R. C. Swan. The permeability of frog muscle fibres to lithium ions. *The Journal of Physiology*, 147(3):626–638, oct 1959. doi: 10.1113/jphysiol.1959.sp006265. URL `https://doi.org/10.1113%2Fjphysiol.1959.sp006265`.

44. B. Lautrup, R. Appali, A. D. Jackson, and T. Heimburg. The stability of solitons in biomembranes and nerves. *The European Physical Journal E*, 34(6), jun 2011. doi: 10.1140/epje/i2011-11057-0. URL `https://doi.org/10.1140%2Fepje%2Fi2011-11057-0`.

45. R. Li, R. El-Mallakh, L. Harrison, D. Changaris, and R. S. Levy. Lithium prevents ouabain-induced behavioral changes. Toward an animal model for manic depression. *Molecular and chemical neuropathology*, 31 1:65–72, 1997.

46. J. López Cascales and J. Garcia de la Torre. Effect of lithium and sodium ions on a charged membrane of dipalmitoylphosphatidylserine: A study by molecular dynamics simulation. *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 1330(2):145–156, 1997. ISSN 0005-2736. doi: 10.1016/S0005-2736(97)00156-9. URL `https://www.sciencedirect.com/science/article/pii/S0005273697001569`.

47. B. López-Corcuera, C. Giménez, and C. Aragón. Change of synaptic membrane lipid composition and fluidity by chronic administration of lithium. *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 939(3): 467–475, apr 1988. doi: 10.1016/0005-2736(88)90093-4. URL `https://doi.org/10.1016%2F0005-2736%2888%2990093-4`.

48. G. S. Malhi and T. Outhred. Therapeutic mechanisms of lithium in bipolar disorder: Recent advances and current understanding. *CNS Drugs*, 30(10):931–949, sep 2016. doi: 10.1007/s40263-016-0380-1. URL `https://doi.org/10.1007%2Fs40263-016-0380-1`.

49. MATLAB documentation. Solve stiff ODEs — MathWorks help center. `https://se.mathworks.com/help/matlab/math/solve-stiff-odes.html`, 2021-03-08.

50. MATLAB documentation. Choose an ODE solver — MathWorks help center. `https://www.mathworks.com/help/matlab/math/choose-an-ode-solver.html`, 2021-05-05.

51. MATLAB documentation. ODE15s — MathWorks help center. `https://www.mathworks.com/help/matlab/ref/ode15s.html`, 2021-05-21.

52. MATLAB documentation. ODE23tb — MathWorks help center. `https://www.mathworks.com/help/matlab/ref/ode23tb.html`, 2021-05-21.

53. MATLAB documentation. ODE45 — MathWorks help center. `https://www.mathworks.com/help/matlab/ref/ode45.html`, 2021-05-21.

54. J. Mertens, Q.-W. Wang, Y. Kim, D. X. Yu, S. Pham, B. Yang, Y. Zheng, K. E. Diffenderfer, J. Zhang, S. Soltani, T. Eames, S. T. Schafer, L. Boyer, M. C. Marchetto, J. I. Nurnberger, J. R. Calabrese, K. J. Oedegaard, M. J. McCarthy, P. P. Zandi, M. Alda, C. M. Nievergelt, S. Mi, K. J. Brennand, J. R. Kelsoe, F. H. Gage, and J. Yao. Differential responses to lithium in hyperexcitable neurons from patients with bipolar disorder. *Nature*, 527(7576):95–99, oct 2015. doi: 10.1038/nature15526. URL `https://doi.org/10.1038%2Fnature15526`.

55. B. Michell. A link between lithium, lipids and receptors? *Trends in Biochemical Sciences*, 7:387–388, 1982.

56. L. D. Mosgaard. *The Electrical And Dynamical Properties of Membranes*. PhD thesis, University of Copenhagen, PhD School of The Faculty of Science, 2014.

57. L. D. Mosgaard, A. D. Jackson, and T. Heimburg. Fluctuations of systems in finite heat reservoirs with applications to phase transitions in lipid membranes. *The Journal of Chemical Physics*, 139(12):125101, sep 2013. doi: 10.1063/1.4821837. URL `https://doi.org/10.1063%2F1.4821837`.

58. L. D. Mosgaard, S. B. Madsen, A. D. Jackson, and T. Heimburg. The timescale of solitons in nerves (working title). *Draft paper, contact Thomas Heimburg at Niels Bohr Institutet*, 2015.

59. A. Murru, M. Manchia, T. Hajek, R. E. Nielsen, J. K. Rybakowski, G. Sani, T. G. Schulze, L. Tondo, and M. Bauer. Lithium's antiviral effects: a potential drug for CoViD-19 disease? *International Journal of Bipolar Disorders*, 8(1), may 2020. doi: 10.1186/s40345-020-00191-4. URL `https://doi.org/10.1186%2Fs40345-020-00191-4`.

60. T. Mužić, F. Tounsi, S. B. Madsen, D. Pollakowski, M. Konrad, and T. Heimburg. Melting transitions in biomembranes. *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 1861(11):183026, nov 2019. doi: 10.1016/j.bbamem.2019.07.014. URL `https://doi.org/10.1016%2Fj.bbamem.2019.07.014`.

61. L. C. Panagiotis. The free energy of the nerve pulse and the effect of anesthetics. Master's thesis, University of Copenhagen, The Niels Bohr Institute, 2019.

62. T. Peck, A. Wong, and E. Norman. Anaesthetic implications of psychoactive drugs. *Continuing Education in Anaesthesia, Critical Care & Pain*, 10:177–181, 2010.

63. C. Pereira, V. Chavarria, J. Vian, M. M. Ashton, M. Berk, W. Marx, and O. M. Dean. Mitochondrial agents for bipolar disorder. *International Journal of Neuropsychopharmacology*, 21(6):550–569, mar 2018. doi: 10.1093/ijnp/pyy018. URL `https://doi.org/10.1093%2Fijnp%2Fpyy018`.

64. M. Rappolt, G. Pabst, H. Amenitsch, and P. Laggner. Salt-induced phase separation in the liquid crystalline phase of phosphatidylcholines. *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, 183-185: 171–181, 2001. ISSN 0927-7757. doi: 10.1016/S0927-7757(01)00568-4. URL `https://www.sciencedirect.com/science/article/pii/S0927775701005684`.

65. J. M. Ritchie and R. D. Keynes. The production and absorption of heat associated with electrical activity in nerve and electric organ. *Quarterly Reviews of Biophysics*, 18(4):451–476, Nov 1985. ISSN 1469-8994. doi: 10.1017/s0033583500005382. URL `http://dx.doi.org/10.1017/s0033583500005382`.

66. J. K. Rybakowski. Antiviral and immunomodulatory effect of lithium. *Pharmacopsychiatry*, 33(5):159–164, Sep 2000. ISSN 0176-3679 (Print); 0176-3679 (Linking).

67. R. Santos, S. B. Linker, S. Stern, A. P. D. Mendes, M. N. Shokhirev, G. Erikson, L. Randolph-Moore, V. Racha, Y. Kim, J. R. Kelsoe, A. G. Bang, M. Alda, M. C. Marchetto, and F. H. Gage. Deficient LEF1 expression is associated with lithium resistance and hyperexcitability in neurons derived from bipolar disorder patients. *Molecular Psychiatry*, jan 2021. doi: 10.1038/s41380-020-00981-3. URL `https://doi.org/10.1038%2Fs41380-020-00981-3`.

68. M. F. Schneider, D. Marsh, W. Jahn, B. Kloesgen, and T. Heimburg. Network formation of lipid membranes: triggering structural transitions by chain melting. *Proceedings of the National Academy of Sciences of the United States of America*, 96 25:14312–7, 1999.

69. W. Schrader, H. Ebel, P. Grabitz, E. Hanke, T. Heimburg, M. Hoeckel, M. Kahle, F. Wente, and U. Kaatze. Compressibility of lipid mixtures studied by calorimetry and ultrasonic velocity measurements. *Journal of Physical Chemistry B*, 106:6581–6586, 2002.

70. H. M. Seeger, M. L. Gudmundsson, and T. Heimburg. How anesthetics, neurotransmitters, and antibiotics influence the relaxation processes in lipid membranes. *The Journal of Physical Chemistry B*, 111(49): 13858–13866, nov 2007. doi: 10.1021/jp075346b. URL `https://doi.org/10.1021%2Fjp075346b`.

71. R. Seshadri and T. Y. Na. Transformation of a boundary value problem to an initial value problem. In *Group Invariance in Engineering Boundary Value Problems*, pages 157–168. Springer New York, 1985. doi: 10.1007/978-1-4612-5102-6_9. URL `https://doi.org/10.1007%2F978-1-4612-5102-6_9`.

72. S. Stern, R. Santos, M. C. Marchetto, A. P. D. Mendes, G. A. Rouleau, S. Biesmans, Q.-W. Wang, J. Yao, P. Charnay, A. G. Bang, M. Alda, and F. H. Gage. Neurons derived from patients with bipolar disorder divide into intrinsically different sub-populations of neurons, predicting the patients' responsiveness to lithium. *Molecular Psychiatry*, 23(6):1453–1465, feb 2017. doi: 10.1038/mp.2016.260. URL `https://doi.org/10.1038%2Fmp.2016.260`.

73. K. Tamm, J. Engelbrecht, and T. Peets. Temperature changes accompanying signal propagation in axons. *Journal of Non-Equilibrium Thermodynamics*, 44(3):277–284, apr 2019. doi: 10.1515/jnet-2019-0012. URL `https://doi.org/10.1515%2Fjnet-2019-0012`.

74. I. Tasaki. Collision of two nerve impulses in the nerve fibre. *Biochimica et Biophysica Acta*, 3:494–497, 1949. ISSN 0006-3002. doi: 10.1016/0006-3002(49)90121-3. URL `https://www.sciencedirect.com/science/article/pii/0006300249901213`.

75. I. Tasaki. A macromolecular approach to excitation phenomena: mechanical and thermal changes in nerve during excitation. *Physiological chemistry and physics and medical NMR*, 20(4):251—268, 1988. ISSN 0748-6642. URL `http://europepmc.org/abstract/MED/3076013`.

76. I. Tasaki and P. Byrne. Volume expansion of nonmyelinated nerve fibers during impulse conduction. *Biophysical Journal*, 57(3):633–635, 1990. ISSN 0006-3495. doi: 10.1016/S0006-3495(90)82580-7. URL `https://www.sciencedirect.com/science/article/pii/S0006349590825807`.

77. I. Tasaki and P. M. Byrne. Large mechanical changes in the bullfrog olfactory bulb evoked by afferent fiber stimulation. *Brain Research*, 475(1):173–176, 1988. ISSN 0006-8993. doi: 10.1016/0006-8993(88)90214-4. URL `https://www.sciencedirect.com/science/article/pii/0006899388902144`.

78. I. Tasaki, A. Watanabe, and I. Singer. Excitability of squid giant axons in the absence of univalent cations in the external medium. *PNAS; Proceedings of the National Academy of Sciences*, 56(4):1116–1122, 1966.

79. I. Tasaki, A. Watanabe, and L. Lerman. Role of divalent cations in excitation of squid giant axons. *American Journal of Physiology-Legacy Content*, 213(6):1465–1474, dec 1967. doi: 10.1152/ajplegacy.1967.213.6.1465. URL `https://doi.org/10.1152%2Fajplegacy.1967.213.6.1465`.

80. I. Tasaki, K. Kusano, and P. Byrne. Rapid mechanical and thermal changes in the garfish olfactory nerve associated with a propagated impulse. *Biophysical Journal*, 55(6):1033–1040, jun 1989. doi: 10.1016/s0006-3495(89)82902-9. URL `https://doi.org/10.1016%2Fs0006-3495%2889%2982902-9`.

81. S. Terakawa. Potential-dependent variations of the intracellular pressure in the intracellularly perfused squid giant axon. *The Journal of Physiology*, 369(1):229–248, dec 1985. doi: 10.1113/jphysiol.1985.sp015898. URL `https://doi.org/10.1113%2Fjphysiol.1985.sp015898`.

82. H. Trauble and H. Eibl. Electrostatic effects on lipid phase transitions: Membrane structure and ionic environment. *Proceedings of the National Academy of Sciences*, 71(1):214–219, jan 1974. doi: 10.1073/pnas.71.1.214. URL `https://doi.org/10.1073%2Fpnas.71.1.214`.

83. E. Villagran-Vargas, A. Ludu, R. Hustert, P. Gumrich, A. D. Jackson, and T. Heimburg. Periodic solutions and refractory periods in the soliton theory for nerves and the locust femoral nerve. *Biophysical Chemistry*, 153(2-3):159–167, jan 2011. doi: 10.1016/j.bpc.2010.11.001. URL `https://doi.org/10.1016%2Fj.bpc.2010.11.001`.

84. Wikipedia contributors. Speed of sound — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Speed_of_sound`, 2021-06-26.

A dditional derivations

A.1 Density dependent dispersion coefficient

Mosgaard [56] finds an expression for the density dependent dispersion coefficient in case of small amplitude sinusoidal pertubations.The following is a walk-through of the calculations.

For low amplitude sound (small $\Delta\rho^A$), the Heimburg-Jackson model from section 2.2.1, equation 1 is approximately linear, meaning the sound velocity is independent of density and I have the equation:

$$\begin{aligned}
\frac{\partial^2 \Delta\rho^A}{\partial t^2} &= \frac{\partial}{\partial x}\left(c^2(\rho^A)\frac{\partial \Delta\rho^A}{\partial x}\right) - h\frac{\partial^4 \Delta\rho^A}{\partial x^4} \\
&= \frac{\partial}{\partial x}c^2\frac{\partial \Delta\rho^A}{\partial x} + c^2\frac{\partial^2 \Delta\rho^A}{\partial x^2} - h\frac{\partial^4 \Delta\rho^A}{\partial x^4} \\
&= c_0\frac{\partial^2 \Delta\rho^A}{\partial x^2} - h\frac{\partial^4 \Delta\rho^A}{\partial x^4} \qquad (44)
\end{aligned}$$

I can then derive a dispersion relation for the soliton model for of low amplitude periodic solutions of the form $\Delta\rho^A = \Delta\rho_0^A e^{i\omega\left(t-\frac{x}{c}\right)}$. I insert the solution into equation 44 and get:

$$(i\omega)^2 = c_0^2 \cdot \left(\frac{-i\omega}{c_0}\right)^2 - h\left(\frac{-i\omega}{c_0}\right)^4 \rightarrow -\omega^2 = c_0^2 \cdot \left(\frac{-\omega^2}{c_0^2}\right) - h\left(\frac{\omega^4}{c_0^4}\right) \rightarrow \frac{\omega^2}{k^2} = c_0^2 + h\frac{\omega^2}{c_0^2} \qquad (45)$$

Thus the phase velocity or propagation speed $c(\omega) = \frac{\omega}{k}$ is given by the dispersion relation:

$$c^2(\omega) \approx c_0^2 + h\left(\frac{\omega^2}{c_0^2}\right) \qquad (46)$$

Mosgaard [56, chap. 5.1] then derives an expression for the propagation speed in the melting regime as a function of thermodynamic variables. The frequency dependent speed of sound is expressed as a complex quantity, $\hat{c}$, which is a function of a real density ρ^A and a complex compressibility, κ_S^A:

$$\hat{c}(\omega) = \frac{1}{\sqrt{\kappa_S^A(\omega)\,\rho^A}} \qquad (47)$$

The source of the dispersion is the nonlinear interaction between the frequency of sound and the relaxation time of the membrane. This is expressed through a frequency dependent, complex excess heat capacity:

$$\Delta\hat{c}_p(\omega) = \left(\frac{1 - i\omega\tau}{1 + (\omega\tau)^2}\right)\Delta c_p$$

Where τ is the characteristic time of the exponential relaxation of the membrane and ω is the frequency of pertubation/sound. The transfer function describing the available part of the excess heat capacity is given by a Debye-term due to the sinusoidal nature of the low amplitude sound.

In section 2.2.1 equation 8 I've shown how the isentropic compressibilty in the limit of large reservoirs is directly dependent on the heat capacity:

$$\kappa_S^V = \kappa_{T,0}^V + \frac{T\gamma_V^2}{V}\Delta c_p$$

Combining the previous two equations, I get:

$$\kappa_S^A = \kappa_T^{A,0} + b\left(\frac{1 - i\omega\tau}{1 + (\omega\tau)^2}\right), \quad \boldsymbol{b = \frac{\gamma_A^2 T}{A}\Delta c_p} \tag{48}$$

The effective speed of sound in equation 46, c, is related to the modulus and real part of the complex speed $\hat{c}$ in the following manner:

$$c = \left(\frac{\mathrm{Re}(\hat{c})}{|\hat{c}|^2}\right)^{-1} \tag{49}$$

Mosgaard [56] ends up at the following expression for the effective speed of sound:

$$c^2(\omega) = (\rho^A)^{-1}\frac{2}{\mathrm{Re}(\kappa_S^A) + |\kappa_S^A|} \tag{50}$$

I'll briefly show that equation 49: and 50 are equal (thanks to Andrew Jackson).

Equation 49 I express the speed of light in it's Cartesian form $\hat{c} = r + is$. The effective speed of sound in eq. 49 is then given by:

$$c^2 = \left(\frac{\mathrm{Re}(r + is)}{|r + is|^2}\right)^{-2} = \left(\frac{r^2 + s^2}{r}\right)^2$$

Equation 50: Inserting the cartesian form into $\kappa_S^A = \left(c^2\rho^A\right)^{-1}$, and expanding the fraction, I get:

$$\kappa_S^A = \frac{1}{(r + is)^2\,\rho^A} = \frac{(r - is)^2}{(r^2 + s^2)^2\,\rho^A}$$

If I evaluate the speed of sound in eq. 50 using the above expression for the compressibility, I get the same expression as with eq. 49:

$$\begin{aligned} c^2 =& (\rho^A)^{-1}\frac{2}{\mathrm{Re}\left(\frac{(r-is)^2}{(r^2+s^2)^2\rho^A}\right) + \left|\frac{(r-is)^2}{(r^2+s^2)^2\rho^A}\right|} \\ =& \frac{2}{\frac{r^2-s^2}{(r^2+s^2)^2} + \frac{\sqrt{(r^2+s^2)^2}}{(r^2+s^2)^2}} \\ =& \frac{2\left(r^2+s^2\right)^2}{2r^2} \\ =& \left(\frac{r^2+s^2}{r}\right)^2 \end{aligned}$$

The full expression for the effective speed of sound (suqared) is then found by inserting the complex compressibility (equation 48) into equation 50.

Next step is to expand the full expression for the speed of sound squared to 2nd order around $\omega = 0$. By comparing the Taylor-expansion to the dispersion relation in equation 46, which is also a polynomial of second order, one can derive an expression for h as a function of adiabatic compressibility and density.

I determine the differentials of the functions $\mathrm{Re}(\kappa_S^A)$ and $|\kappa_S^A|$ needed for the expansion:

$$\begin{aligned} \mathrm{Re}(\kappa_S^A) &= \kappa_T^{A,0} + \frac{\gamma_A^2 T}{A}\left(\frac{\Delta c_p}{1+(\omega\tau)^2}\right) \\ \rightarrow \frac{\partial \mathrm{Re}(\kappa_S^A)}{\partial\omega} &= \frac{b(-2\omega\tau^2)}{\left(1+(\omega\tau)^2\right)^2}, \quad \boldsymbol{b = \frac{\gamma_A^2 T}{A}\Delta c_p} \end{aligned}$$

$$\begin{aligned} |\kappa_S^A| &= \sqrt{\left(\mathrm{Re}(\kappa_S^A)\right)^2 + b^2\left(\frac{\omega\tau}{1+(\omega\tau)^2}\right)^2} \\ \rightarrow \frac{\partial|\kappa_S^A|}{\partial\omega} &= \frac{2\left(\mathrm{Re}(\kappa_S^A)\frac{\partial\mathrm{Re}(\kappa_S^A)}{\partial\omega} + b^2\frac{\omega\tau}{1+(\omega\tau)^2}\left[\frac{\tau}{1+(\omega\tau)^2} + \frac{(-2\omega^2\tau^3)}{(1+(\omega\tau)^2)^2}\right]\right)}{2\left(\left(\mathrm{Re}(\kappa_S^A)\right)^2 + b^2\left(\frac{\omega\tau}{1+(\omega\tau)^2}\right)^2\right)^{1/2}} \end{aligned}$$

Thus I have the first derivative of the squared velocity:

$$\begin{aligned} \frac{\partial c^2}{\partial\omega} =& (\rho^A)^{-1}\frac{-2}{\left(\mathrm{Re}(\kappa_S^A) + |\kappa_S^A|\right)^2} \\ &\cdot\left[\frac{b(-2\omega\tau^2)}{\left(1+(\omega\tau)^2\right)^2} + \frac{2\left(\mathrm{Re}(\kappa_S^A)\frac{b(-2\omega\tau^2)}{(1+(\omega\tau)^2)^2} + b^2\frac{\omega\tau}{1+(\omega\tau)^2}\left[\frac{\tau}{1+(\omega\tau)^2} + \frac{(-2\omega^2\tau^3)}{(1+(\omega\tau)^2)^2}\right]\right)}{2\left(\left(\mathrm{Re}(\kappa_S^A)\right)^2 + b^2\left(\frac{\omega\tau}{1+(\omega\tau)^2}\right)^2\right)^{1/2}}\right] \end{aligned}$$

For the second derivative things get slightly tedious. Differentiating once again using the chainrule $(f\cdot g)' = f'\cdot g + f\cdot g'$, where:

$$\begin{aligned} f &= (\rho^A)^{-1}\frac{-2}{\left(\mathrm{Re}(\kappa_S^A) + |\kappa_S^A|\right)^2} \\ g &= \left[\frac{b(-2\omega\tau^2)}{\left(1+(\omega\tau)^2\right)^2} + \frac{2\left(\mathrm{Re}(\kappa_S^A)\frac{b(-2\omega\tau^2)}{(1+(\omega\tau)^2)^2} + b^2\frac{\omega\tau}{1+(\omega\tau)^2}\left[\frac{\tau}{1+(\omega\tau)^2} + \frac{(-2\omega^2\tau^3)}{(1+(\omega\tau)^2)^2}\right]\right)}{2\left(\left(\mathrm{Re}(\kappa_S^A)\right)^2 + b^2\left(\frac{\omega\tau}{1+(\omega\tau)^2}\right)^2\right)^{1/2}}\right] \end{aligned}$$

one realizes that g evaluated at $\omega = 0$ is zero, meaning the second term of $(f\cdot g)'$ cancels, and I am left with:

$$\begin{aligned} \left.\frac{\partial^2 c^2}{\partial\omega^2}\right|_0 =& (\rho^A)^{-1}\frac{-2}{\left(\mathrm{Re}(\kappa_S^A) + |\kappa_S^A|\right)^2}\left[\left(\frac{b(-2\tau^2)}{\left(1+(\omega\tau)^2\right)^2} + 0\right)\right. \\ &\left. + \frac{\left(0 + \mathrm{Re}(\kappa_S^A)\frac{b(-2\tau^2)}{(1+(\omega\tau)^2)^2} + 0\right) + \left(b^2\frac{\tau}{1+(\omega\tau)^2}\left[\frac{\tau}{1+(\omega\tau)^2} + \frac{(-2\omega^2\tau^3)}{(1+(\omega\tau)^2)^2}\right] + 0\right)}{\left(\left(\mathrm{Re}(\kappa_S^A)\right)^2 + b^2\left(\frac{\omega\tau}{1+(\omega\tau)^2}\right)^2\right)^{1/2}}\right] \end{aligned}$$

The zeros are to mark derivatives that dissappear when evaluated at $\omega = 0$.

$$
\begin{aligned}
&= (\rho^A)^{-1} \frac{-2\tau^2}{\left(\frac{2}{\rho^A c^2(\omega=0)}\right)^2} \left[-2b + \frac{(\kappa_T^{A,0} + b) \cdot (-2b) + b^2}{\kappa_T^{A,0} + b} \right] \\
&= \frac{\tau^2 c^4(0)}{2} \left[\frac{4\kappa_T^{A,0} + 3b}{\frac{1}{\rho^A b}(\kappa_T^{A,0} + b)} \right]
\end{aligned}
$$

To simplify things Mosgaard [56] rephrases his expression using the following velocity components:

$$
\begin{aligned}
(c_1(\rho))^2 &\equiv \frac{1}{\rho^A \kappa_T^{A,0}} \quad \text{High frequency limit} \\
(c_2(\rho))^2 &\equiv \frac{1}{\rho^A b} \quad \text{Melting component} \\
(c_0(0,\rho))^2 &\equiv \frac{1}{(c_1(\rho))^{-2} + (c_2(\rho))^{-2}} \quad \text{Low frequency limit}
\end{aligned}
$$

By multiplying nominator and denominator with $\frac{1}{\rho^A \kappa_T^{A,0} b}$, he can fully substitute all mentions of density and compressibility:

$$
\begin{aligned}
&= \frac{\tau^2 c_0^4}{2} \left[\frac{4\frac{1}{\rho^A b} + 3\frac{1}{\rho^A \kappa_T^{A,0}}}{\frac{1}{\rho^A b}\left(\frac{1}{\rho^A b} + \frac{1}{\rho^A \kappa_T^{A,0}}\right)} \right] \\
&= \frac{\tau^2 c_0^4}{2} \left[\frac{4c_2^2 + 3c_1^2}{c_2^2(c_2^2 + c_1^2)} \right]
\end{aligned}
$$

The second derivative is set equal to the 2nd order coefficient in the dispersion relation, equation 46:

$$
\begin{aligned}
\frac{h}{c_0^2} &= \frac{1}{2!} \frac{\partial^2 c^2}{\partial \omega^2} \bigg|_0 = \frac{1}{2!} \frac{\tau^2 c_0^4}{2} \left[\frac{4c_2^2 + 3c_1^2}{c_2^2(c_2^2 + c_1^2)} \right] \\
\leftrightarrow h &= \frac{\tau^2 c_0^6}{4} \left[\frac{4c_2^2 + 3c_1^2}{c_2^2(c_2^2 + c_1^2)} \right]
\end{aligned}
$$

Using the relationship $\tau = \frac{T^2}{L} \Delta c_p$, one finds the approximate relationship between h and $\Delta\rho$, where $\Delta\rho^A = \rho^A - \rho_0$ is the change in density from the unpertubed membrane state:

$$
h = \Delta c_p^2 \left(\frac{T^2 c_0^3}{2L} \right)^2 \left[\frac{4c_2^2 + 3c_1^2}{c_2^2(c_2^2 + c_1^2)} \right] \tag{51}
$$

This is the density dependent dispersion coefficient used in the self-contained soliton model. Checked in Maple using Physics package ('Parameters') and evaluating assuming all constants are positive and real.

A.2 Invariance under translation

I verify the differential equation is invariant under translation, $z \mapsto u = z + a \;\forall\; a \in \Re$. Since there is no explicit dependency on z, one only needs to show that the derivatives are independent too:

$$\text{1st)}\quad \frac{\partial}{\partial z} = \frac{\partial}{\partial u}\frac{\partial u}{\partial z} = \frac{\partial}{\partial u} \qquad \text{2nd)}\quad \frac{\partial^2}{\partial z^2} = \frac{\partial}{\partial u}\left(\frac{\partial}{\partial u}\right) = \frac{\partial^2}{\partial u^2} \qquad \text{4th)}\quad \frac{\partial^4}{\partial z^4} = \frac{\partial^2}{\partial u^2}\left(\frac{\partial^2}{\partial u^2}\right) = \frac{\partial^4}{\partial u^4}$$

In figure 34 I have illustrated the invariance by starting the ODE-solver for 3 different initial z-values for a fixed range of initial values. The plots are indistinguishable from each other.

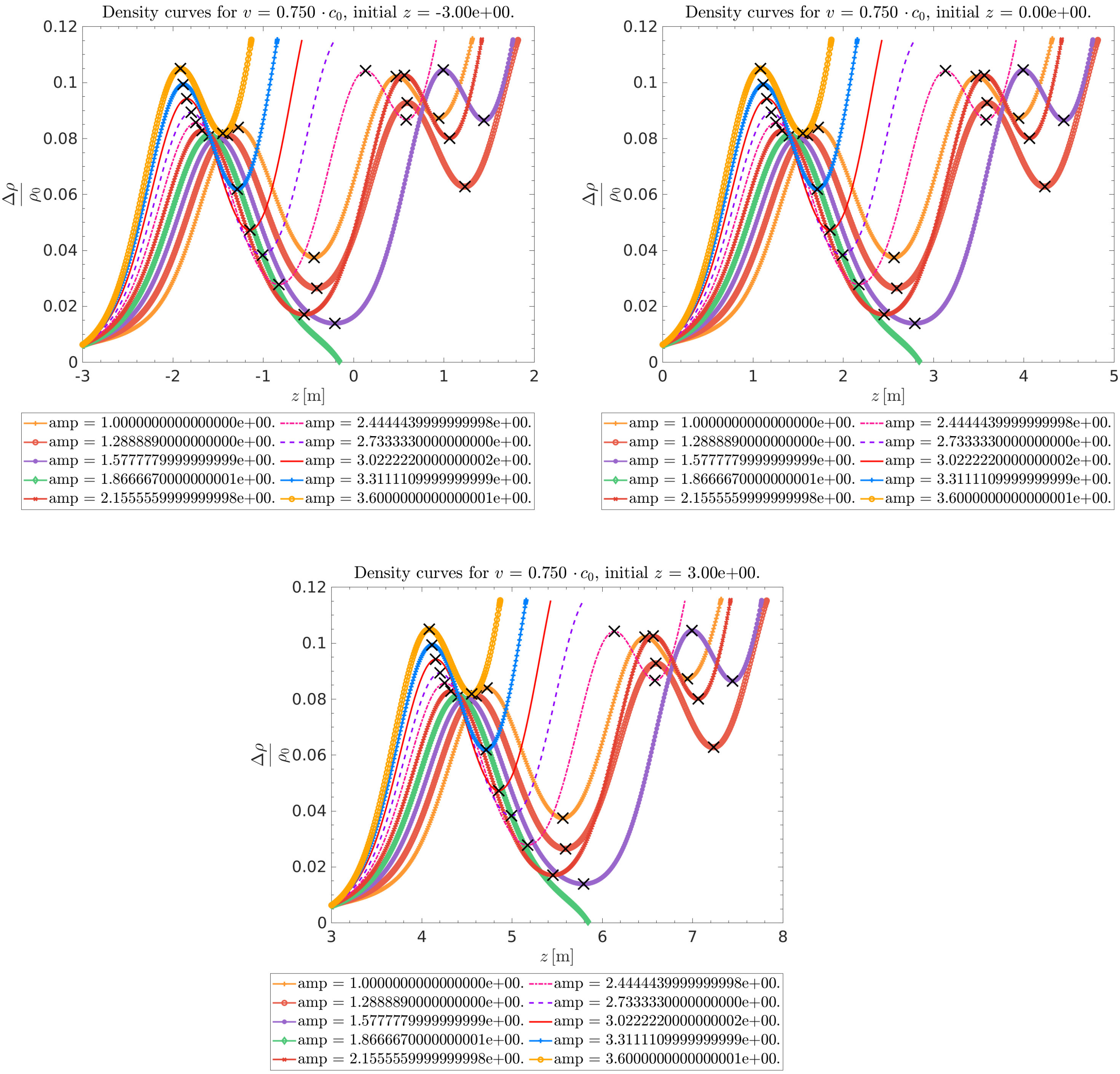


Figure 34: ODE15s, same range of initial values for 3 different initial z.

B Table of DSC values

Sample type	**Melting temperature, [K]**	**Maximum heat capacity, $\left[\frac{\text{J}}{\text{mol}\cdot\text{K}}\right]$**	**FWHM, [K]**
		MLV residual peak	
MilliQ	314.5	$3.8 \cdot 10^4$	0.4
LiCl	314.7	$3.1 \cdot 10^4$	0.2
NaCl	314.7	$2.0 \cdot 10^4$	0.2
KCl	314.6	$2.1 \cdot 10^4$	0.3
Reference	314.6	$1.3 \cdot 10^2$	0.3
		Main transition, LUV	
MilliQ	314 ±0.25	$1.6 \cdot 10^4$	-
LiCl	314.2	$1.1 \cdot 10^4$	1.6
NaCl	313.7	$7.4 \cdot 10^3$	2.3
KCl	314.1	$9.5 \cdot 10^3$	2.0
		Pre-transition, LUV	
MilliQ	307.8	$9.4 \cdot 10^2$	-
LiCl	307.4	$8.2 \cdot 10^2$	0.9
NaCl	308.0	$9.0 \cdot 10^2$	-
KCl	308 ±0.25	$9.9 \cdot 10^2$	-
		Sub-transition, LUV	
MilliQ	311 ±0.25	$1.7 \cdot 10^3$	-
KCl	311.3	$[3.0 \pm 0.5] \cdot 10^3$	-
		Additional bump, LUV	
KCl	293.0	$4.1 \cdot 10^2$	1.1

Table 1: The values were obtained from the heat capacity curves in figure 28. The numbers have been rounded to the approximated accuracy.

C List of materials

DPPC lipids - 1,2-Dipalmitoyl-*sn*-Glycero-3-Phosphocholine

Molar weight 734.05 fracgmol

Avanti Polar Lipids, Inc, lot #: 160PC-250 (to be stored at -20° C).

Lithium chloride - LiCl

Molar weight 42.39 $\frac{\text{g}}{\text{mol}}$

Merck, article number 5679, lot # 2330343

Sodium chloride - NaCl

Molar weight 58.44 $\frac{\text{g}}{\text{mol}}$

AnalaR NORMAPUR, lot # 16E114102

Potassium chloride - KCl

Molar weight 75.55 $\frac{\text{g}}{\text{mol}}$

Fluka Analytical, lot # BCBF3016V

MilliQ water - Obtained from a Direct-Q® 3 UV

Demineralized water - From tap in the Niels Bohr institute

D List of table values

Constant dispersion coefficient, dispersion constant $h = 50\frac{\text{m}^4}{\text{s}^2}$.

Thermodynamic heat capacity parameters (DPPC)

Avogadro's number: $N_A = 6.02214076 \cdot 10^{23}\ \text{mol}^{-1}$ (Google)

Melting entalpy: $\Delta H_0 = 38.1 \cdot 10^3\ \frac{\text{J}}{\text{mol}}$ (I'm not sure of the origins of this number, but it's not far from values found in literature. Examples from literature are: $\Delta H_0 = 39.2 \cdot 10^3\ \frac{\text{J}}{\text{mol}}$[28], $\Delta H_0 = 35 \cdot 10^3\ \frac{\text{J}}{\text{mol}}$ [32] and $36.4 \cdot 10^3\ \frac{\text{J}}{\text{mol}}$ [6]. The first number is for MLVs and includes a pretransition of $5.26 \cdot 10^3\ \frac{\text{J}}{\text{mol}}$. The vesicle structure and whether pretransition is included is not known for the latter two.)

Melting temperature: $T_m = 41.3 + 273.15$ K (Ebel et al. [13])

Intrinsic thermal expansion coefficient, fluid phase: $X_{\text{fluid}} = 0.0042\ \text{K}^{-1}$ (Heimburg [28])

Intrinsic thermal expansion coefficient, gel phase: $X_{\text{gel}} = 0.0026\ \text{K}^{-1}$ (Heimburg [28])

Area per lipid, fluid phase: $\langle A_{\text{fluid}}(T_1)\rangle = 62.9 \cdot 10^{-20}\ \text{m}^2$, $T_1 = 50 + 273.15$ K (Heimburg [28])

Area per lipid, gel phase: $\langle A_{\text{gel}}(T_2)\rangle = 47.4 \cdot 10^{-20}\ \text{m}^2$, $T_2 = 25 + 273.15$ K (Heimburg [28])

Boltzmann's constant: $k = 1.38064852 \cdot 10^{-23}\ \frac{\text{J}}{\text{K}}$ (Google).

Gas constant: $R = 8.31446261815324\ \frac{\text{J}}{\text{K}\cdot\text{mol}}$ (Wikipedia)

Cooperativity: $n = 80$ (own choice);

Proportionality constant: $\gamma_A = 0.893\ \frac{\text{m}^2}{\text{J}}$; (Heimburg [28])

Molar weight: $M = 734.039 \cdot 10^{-3}\ \frac{\text{kg}}{\text{mol}}$ (Google)

Additional parameters (DPPC)

Temperature: $T = 45 + 273$ K (Mosgaard et al. [58])

Reference density, Taylor expansion of the low freqency velocity: $\rho_0 = 4.035 \cdot 10^{-6}\ \frac{\text{kg}}{\text{m}^2}$ Also used as reference frequency in simulation, (Mosgaard et al. [58])

Small amplitude velocity of soliton: $c(\rho_0) = 176.6\ \frac{\text{m}}{\text{s}}$ (Mosgaard et al. [58])

Molar area of DPPC, fluid phase: $A = 0.5 \cdot 62.9 \cdot 10^{-20} \cdot 6.022 \cdot 10^{23}\ \text{m}^2$ (Heimburg [28])

Taylor-expansion coefficient of low frequency velocity: $p = -16.6 \cdot \frac{c(\rho_0)^2}{\rho_0}$ (Mosgaard et al. [58])

Taylor-expansion coefficient of low frequency velocity: $q = 79.5 \cdot \frac{c(\rho_0)^2}{\rho_0^2}$ (Mosgaard et al. [58])

Reference density, Taylor expansion of the high freqency velocity: $\rho_0' = 4.107 \cdot 10^{-6}\ \frac{\text{kg}}{\text{m}^2}$ (Mosgaard et al. [58])

Taylor-expansion coefficient of high frequency velocity: $k = 1.22$ (Mosgaard et al. [58])

Taylor-expansion coefficient of high frequency velocity: $f = 2.3 \cdot \frac{c(\rho_0)^2}{\rho_0'}$ (Mosgaard et al. [58])

Taylor-expansion coefficient of high frequency velocity: $g = 20.9 \cdot \frac{c(\rho_0)^2}{(\rho_0')^2}$ (Mosgaard et al. [58])

Onsager's phenomenological constant: $L = 0.5 \cdot 10^{12}\ \frac{\text{J}\cdot\text{K}}{\text{s}\cdot\text{mol}}$ (Mosgaard et al. [58])

Gaussian heat capacity parameters

Loosely fitted from DPPC data in figure 3 of Mosgaard et al. [58]

Scaling: Maximum heat capacity $= (1472 - 140.5)/1662 \cdot 20 \cdot 10^3\ \frac{\text{J}}{\text{mol}\cdot\text{K}}$

Full width at half max: FWHM $= 1613/1754 \cdot 0.6 \cdot 10^{-6}\ \frac{\text{kg}}{\text{m}^2}$, use FWHM to derive the standard deviation, $\sigma = \text{FWHM}(2\sqrt{\log(2)\cdot 2})^{-1}$

Mean: $\mu = 0.4 \cdot 10^{-6}\ \frac{\text{kg}}{\text{m}^2}$

Thermodynamic heat capacity parameters, E. Coli membrane

Overwrites some of the DPPC default values. Van't Hoff fit of figure 3 in Mužić et al. [60].

Cooperativity: $n = 5.5$;

Melting entalpy $\Delta H_0 = 35 \cdot 10^3\ \frac{\text{J}}{\text{mol}}$;

Melting temperature: $T_m = 20.65 + 273.15$ K

E Additional plots

E.1 Numerical analysis: Differences in ODE solvers

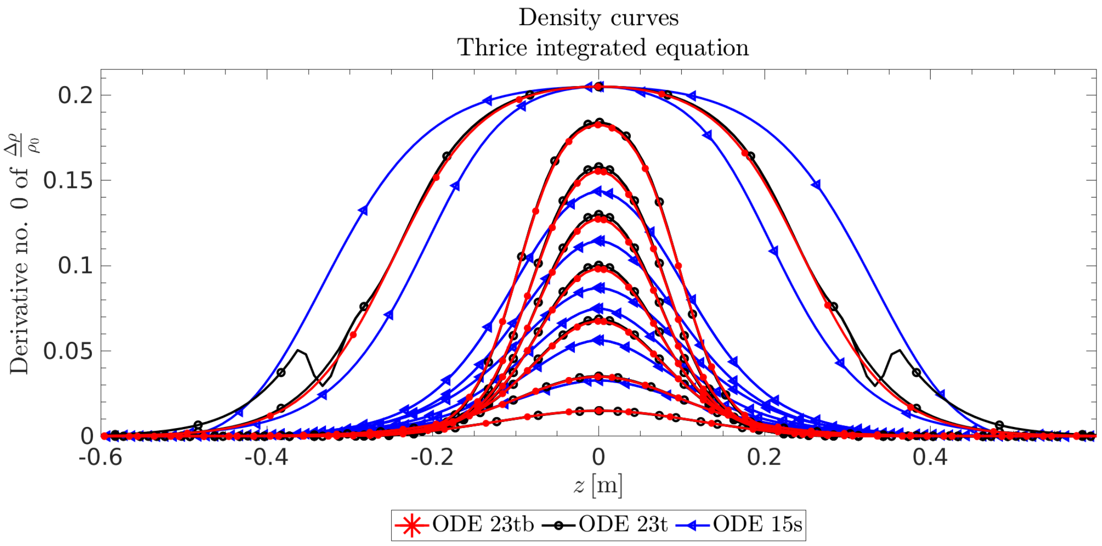


Figure 35: Solitary solutions for velocities in the range (large amplitudes to small): $\{0.65 \cdot c_0, 0.70 \cdot c_0, 0.75 \cdot c_0, 0.80 \cdot c_0, 0.85 \cdot c_0, 0.90 \cdot c_0, 0.95 \cdot c_0, 1.00 \cdot c_0\}$ obtained by three different solvers. Both boundaries of the bisection interval for each solution are plotted, differ at $0.65 \cdot c_0$ in the case of ODE15s and ODE 23t.

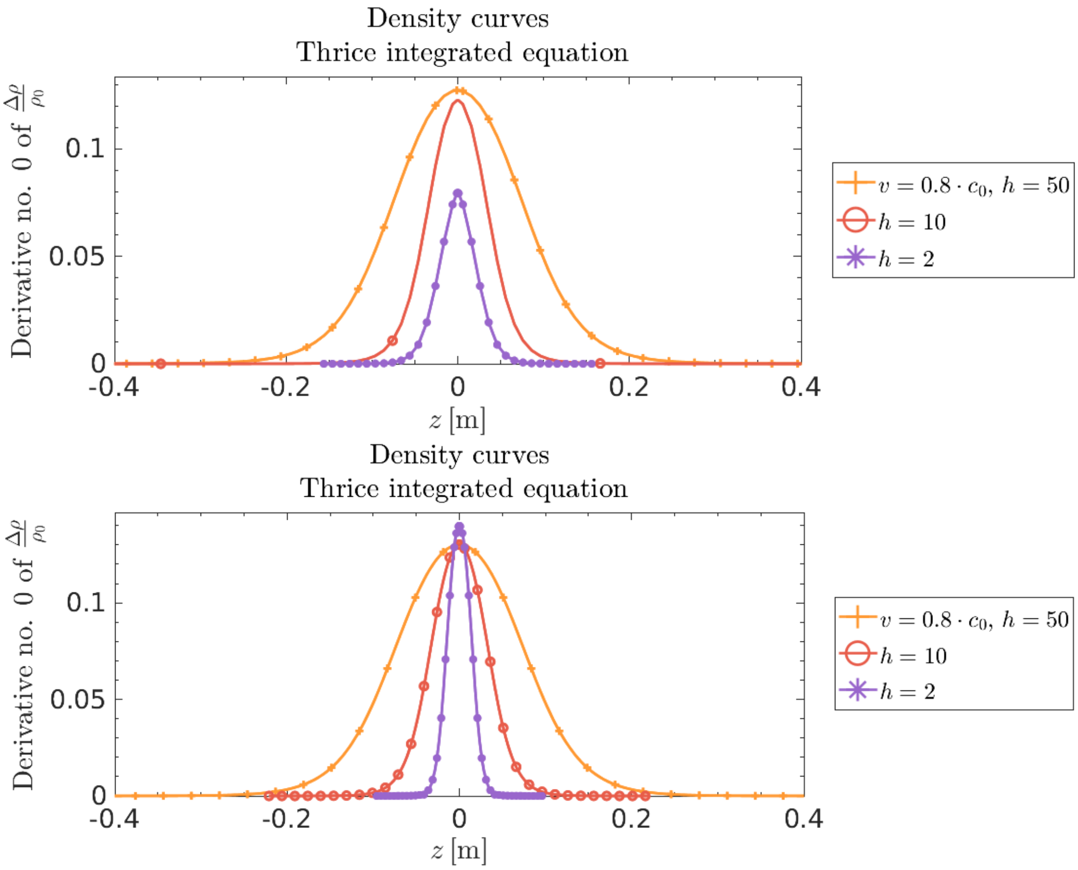


Figure 36: Varying the constant dispersion coefficient h in the thrice integrated case. **ABOVE:** ODE23tb. **BELOW:** ODE45.

E.2 Numerical analysis: Full velocity plots

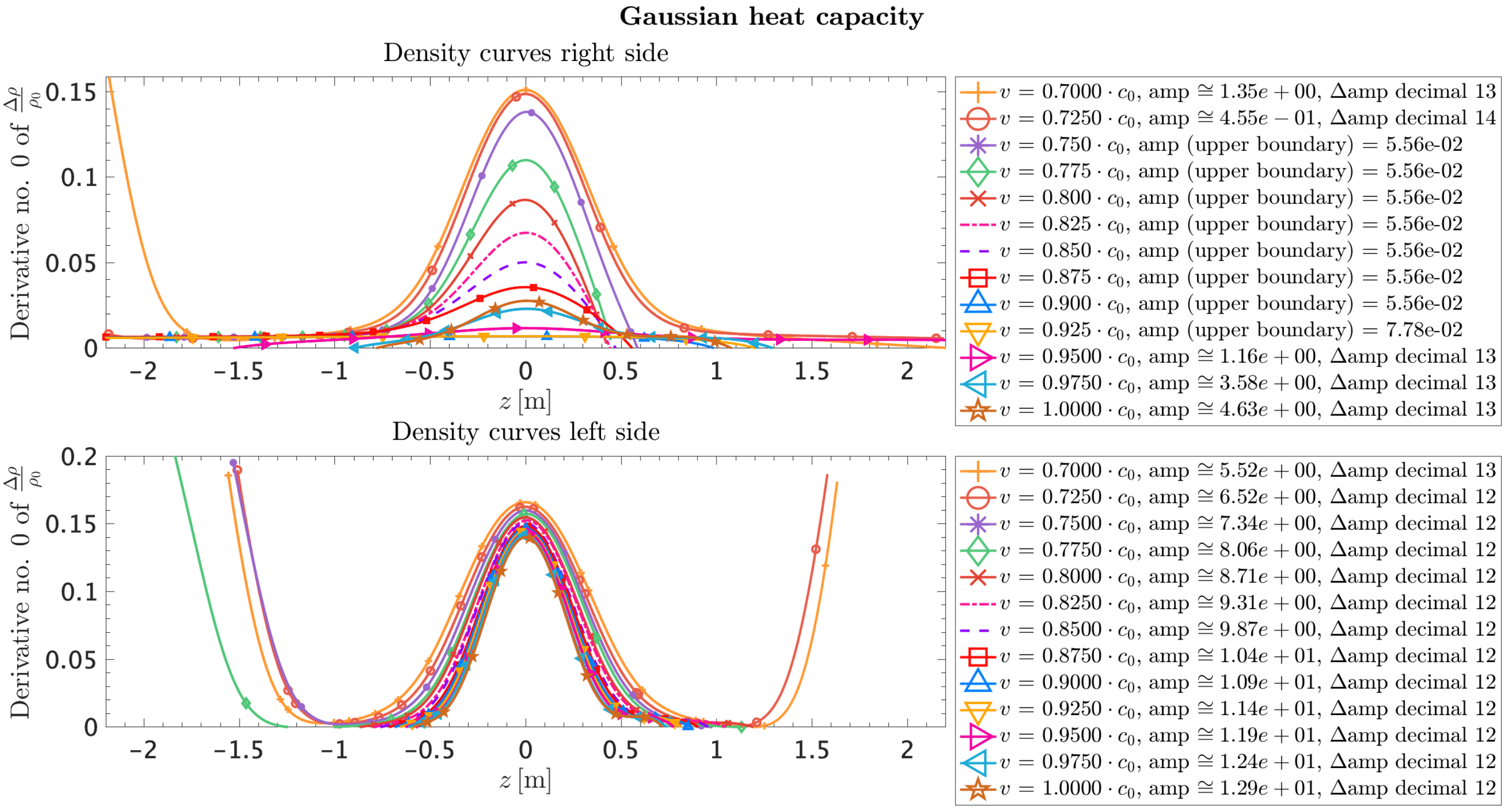


Figure 37: ODE 15s

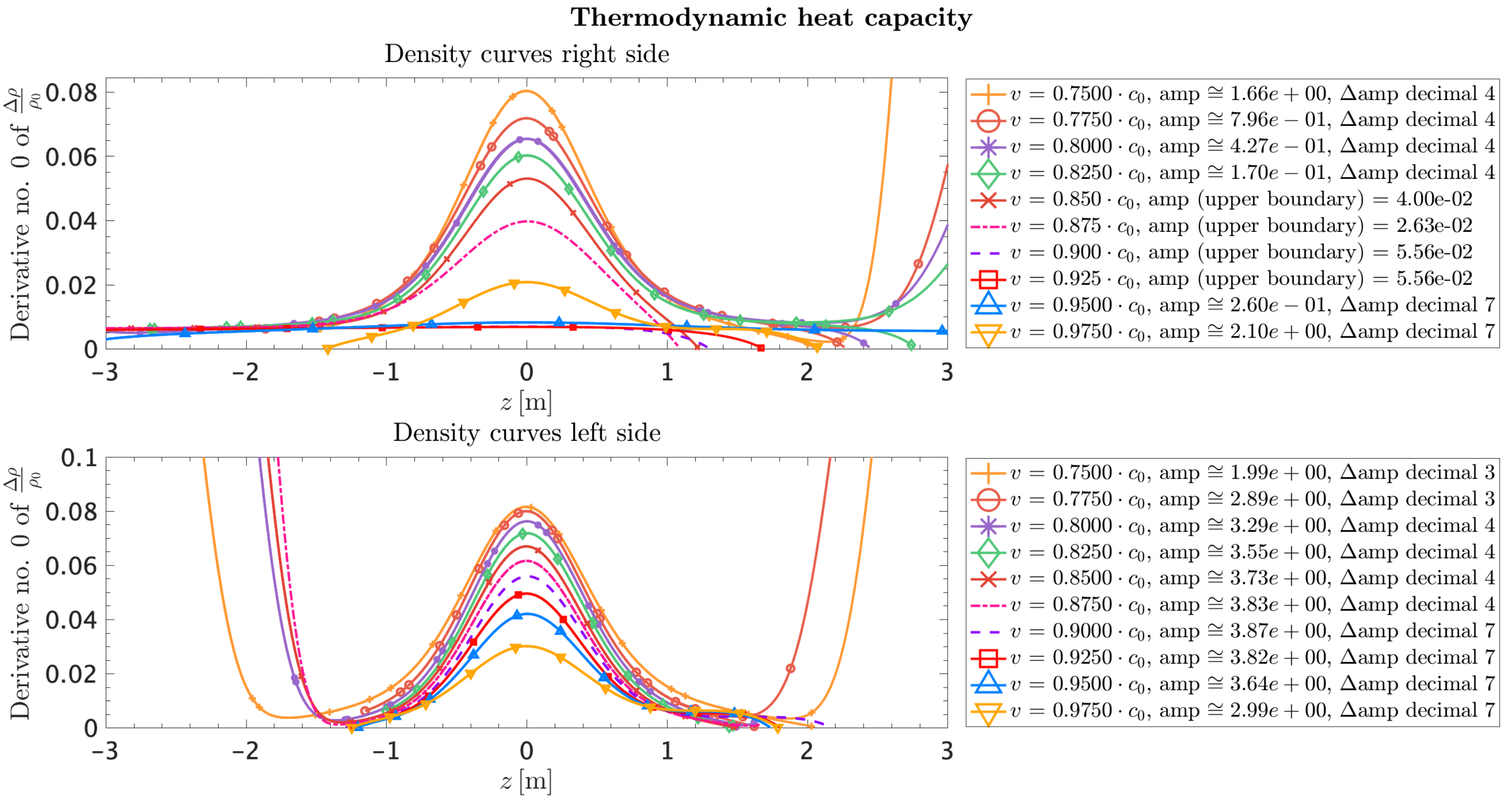


Figure 38: ODE 15s, cooperativity of 80

E.3 Numerical analysis: E. coli run

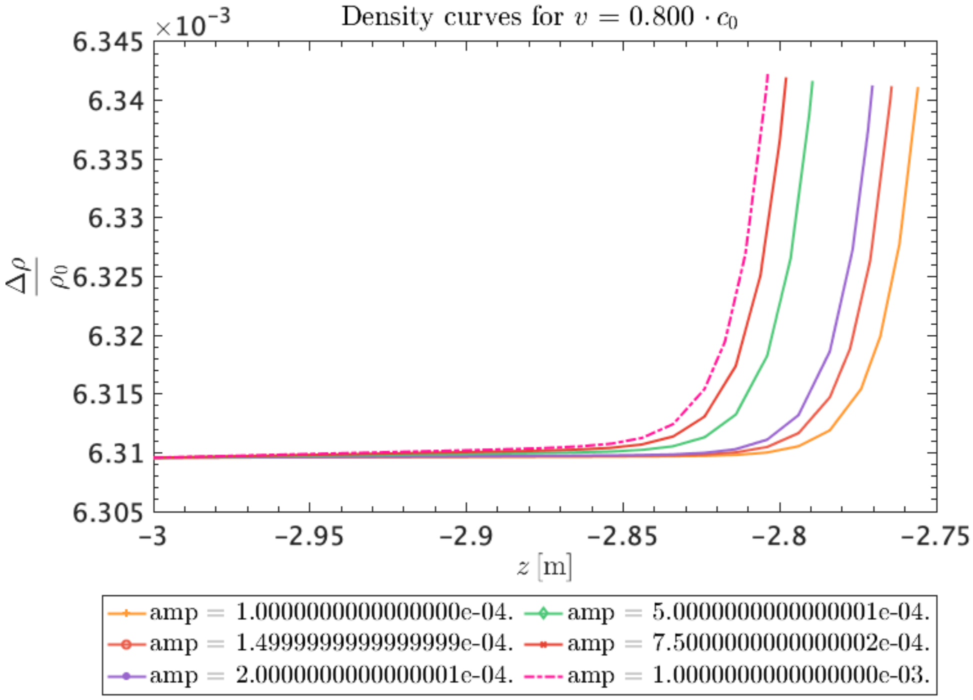


Figure 39: ODE15s run of e. coli parameters included in appendix D.

E.4 Numerical analysis: Improving initial conditions, Gaussian heat capacity

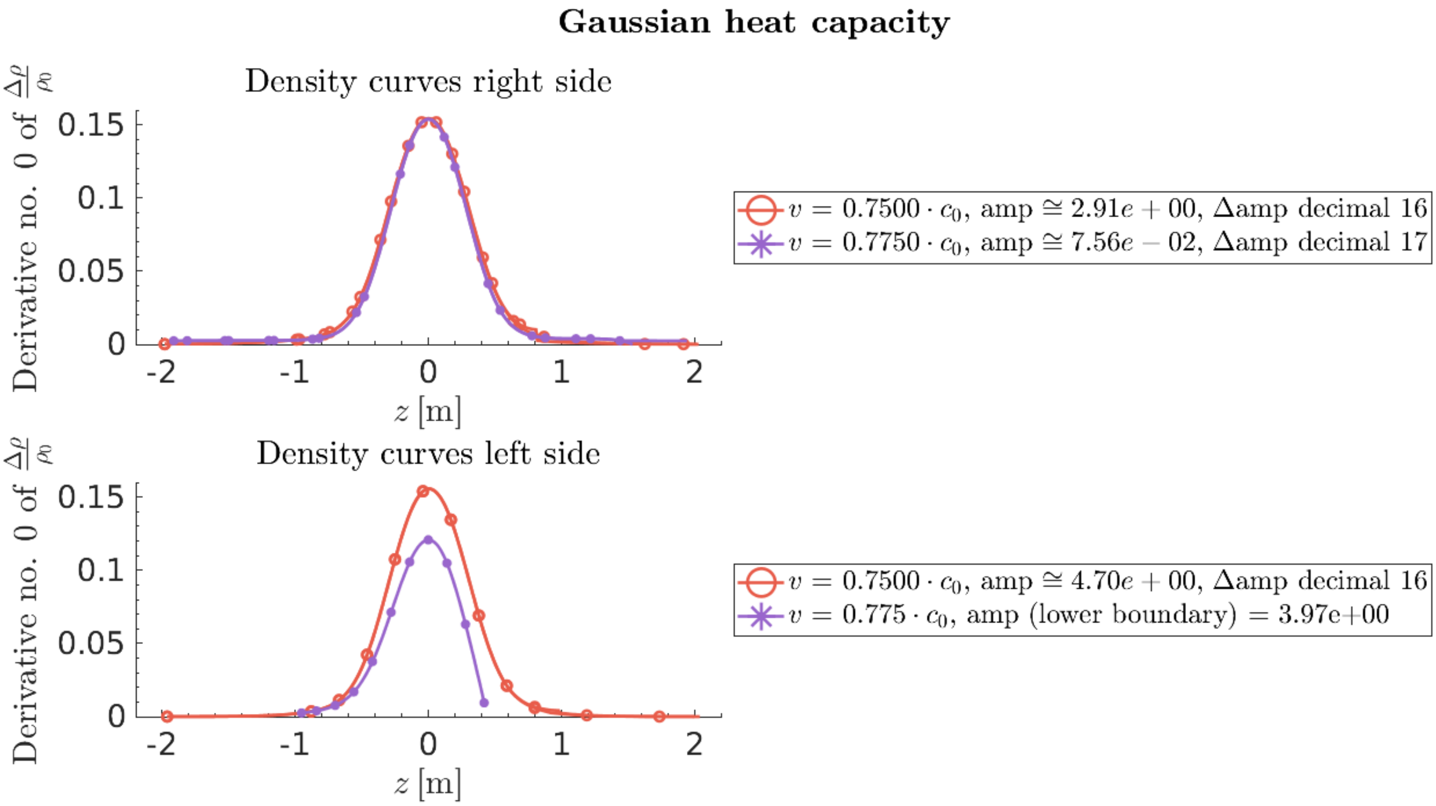


Figure 40: ODE15s, Gaussian heat capacity, decreasing initial density from $y_0 = \Delta\rho = 10^{-2.2} \cdot \rho_0$ to $10^{-2.6} \cdot \rho_0$

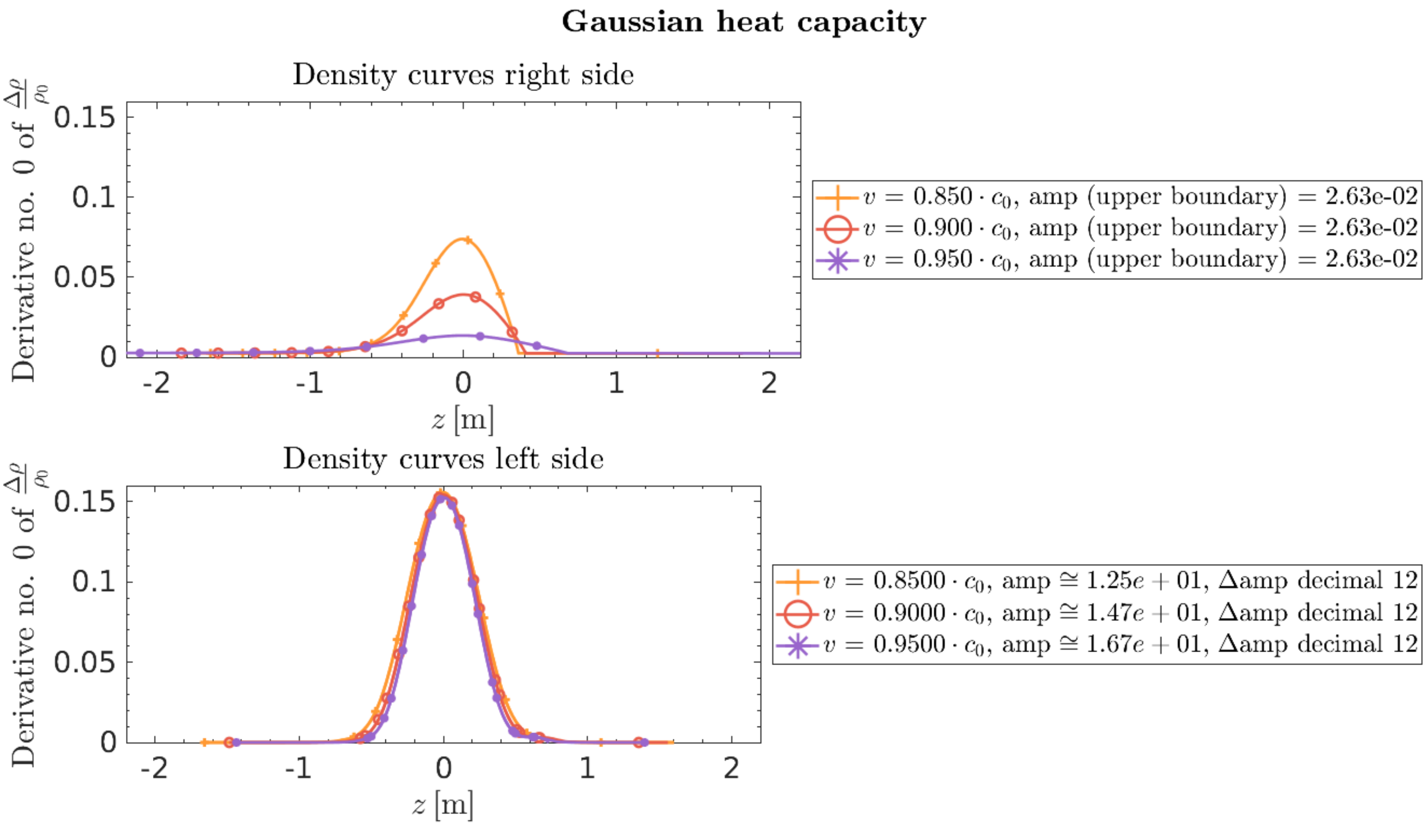


Figure 41: ODE23tb, Gaussian heat capacity, decreasing initial density from $y_0 = \Delta\rho = 10^{-2.2} \cdot \rho_0$ to $10^{-2.6} \cdot \rho_0$

E.5 Numerical analysis: Fitting ends

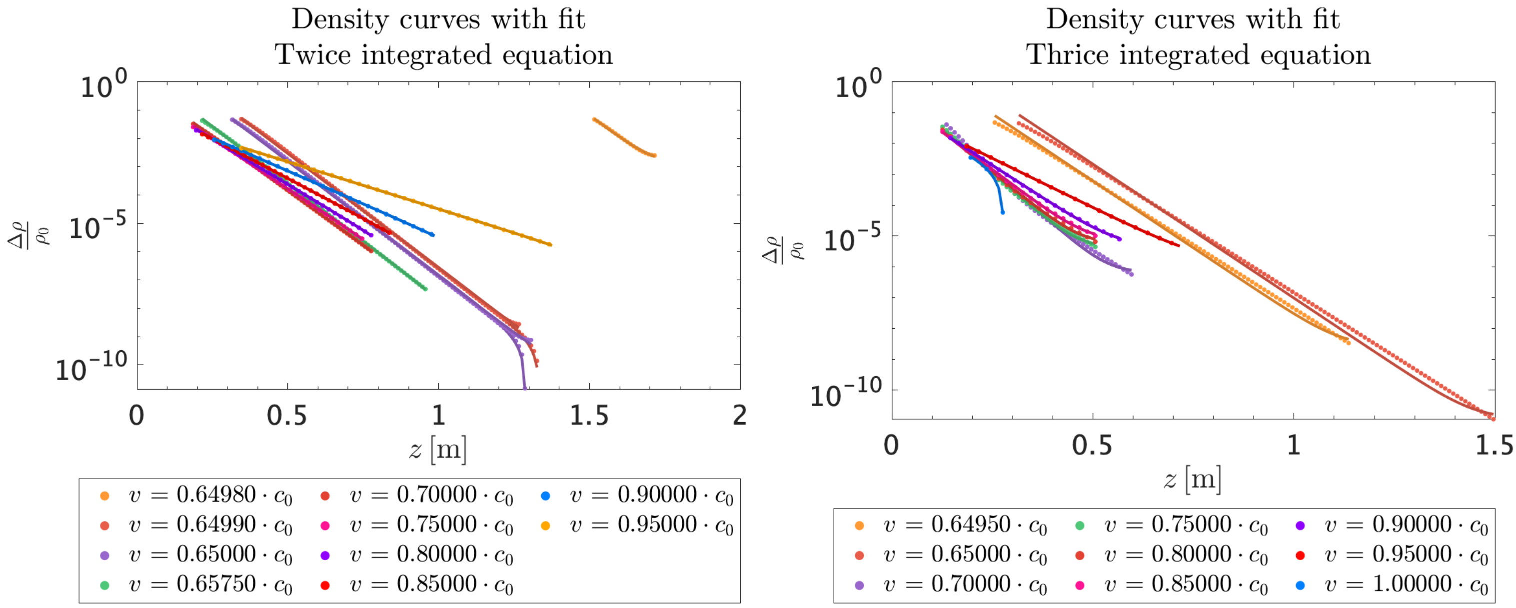


Figure 42: Fit of ends for twice and thrice integrated soliton model with a constant dispersion coefficient, ODE23tb.

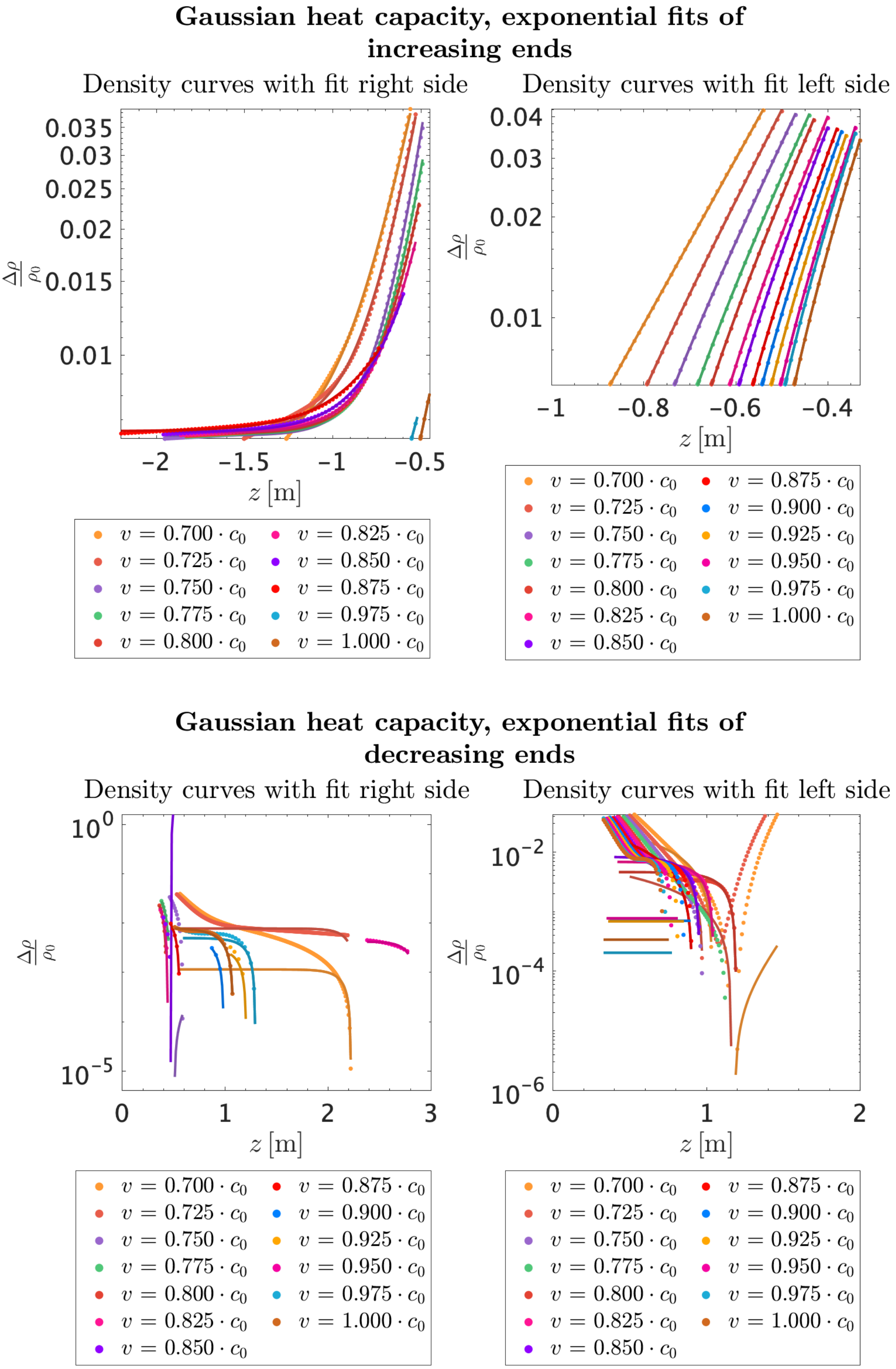
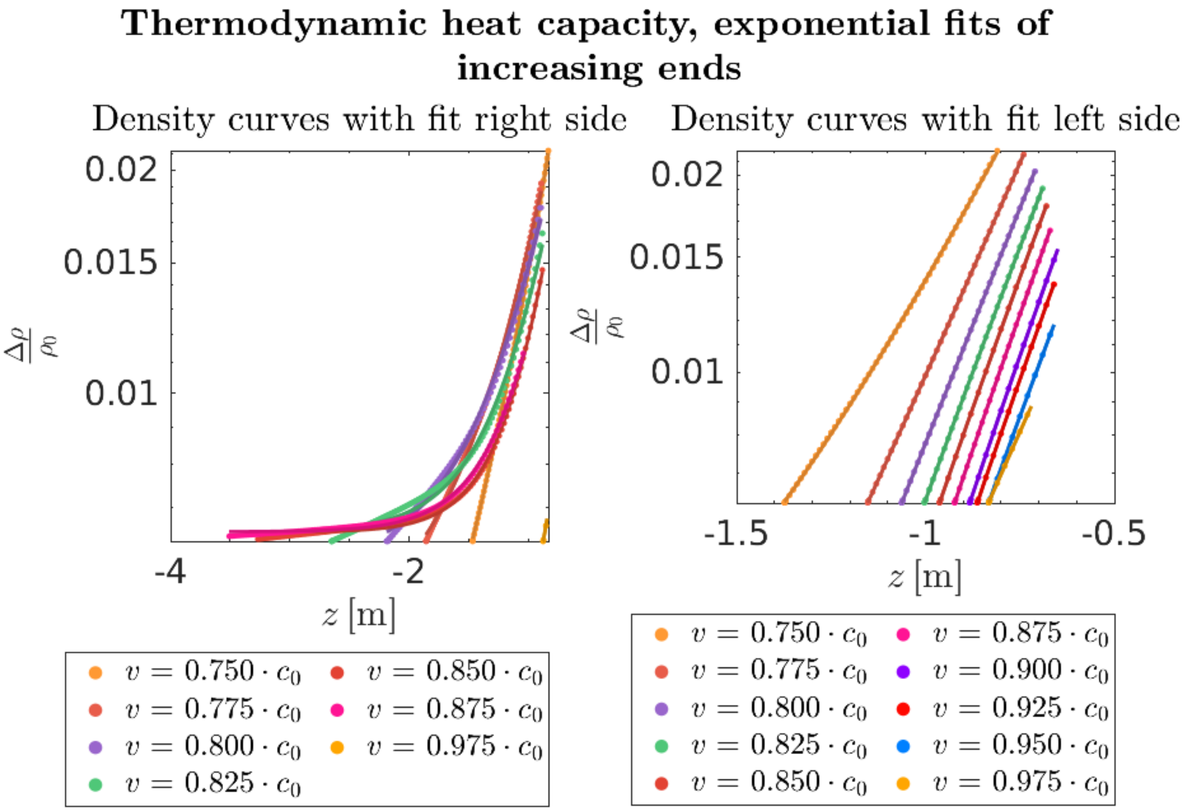


Figure 43: Fit of ends of both right and left side solutions for the soliton model with a density-dependent dispersion coefficient containing a Gaussian approximation of the heat capacity density dependence, ODE15s.



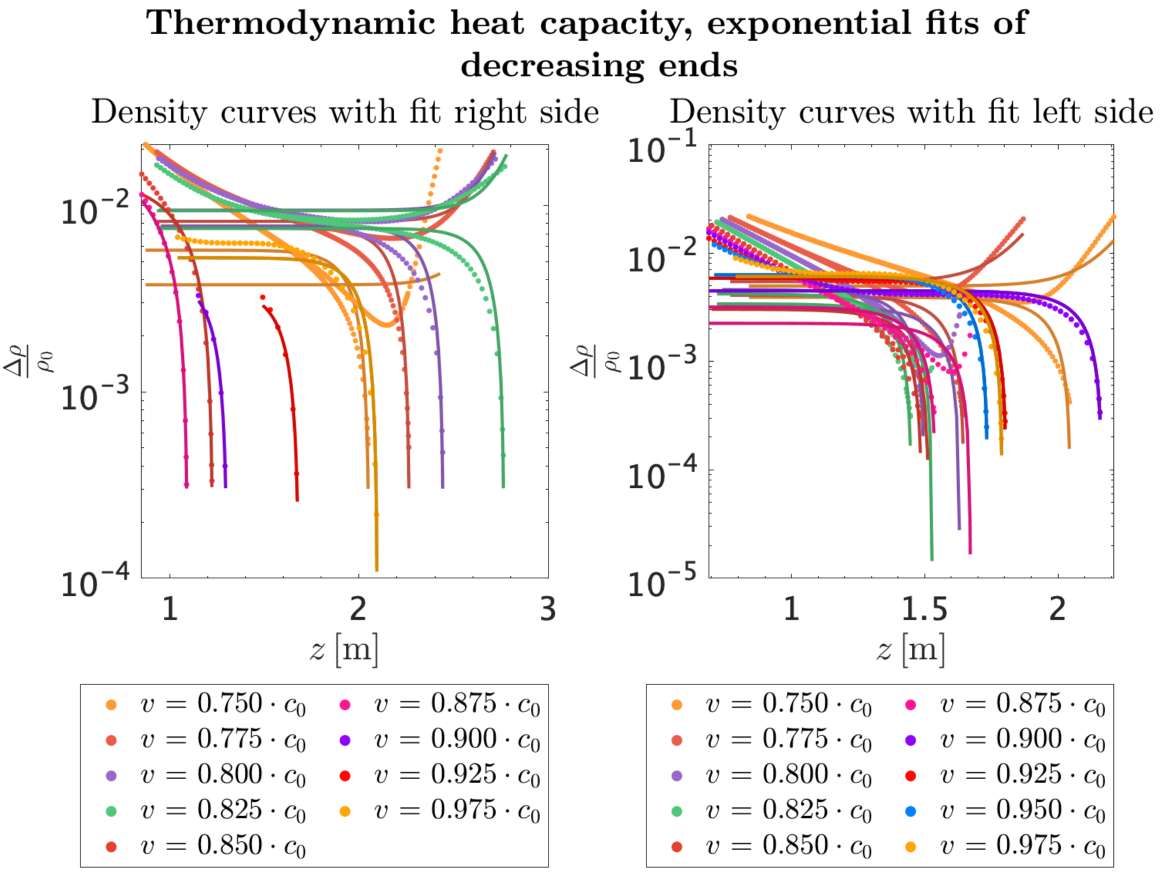


Figure 44: Fit of ends of both right and left side solutions for the self-contained soliton model´, ODE15s

E.6 Experiment: Up- and down scans

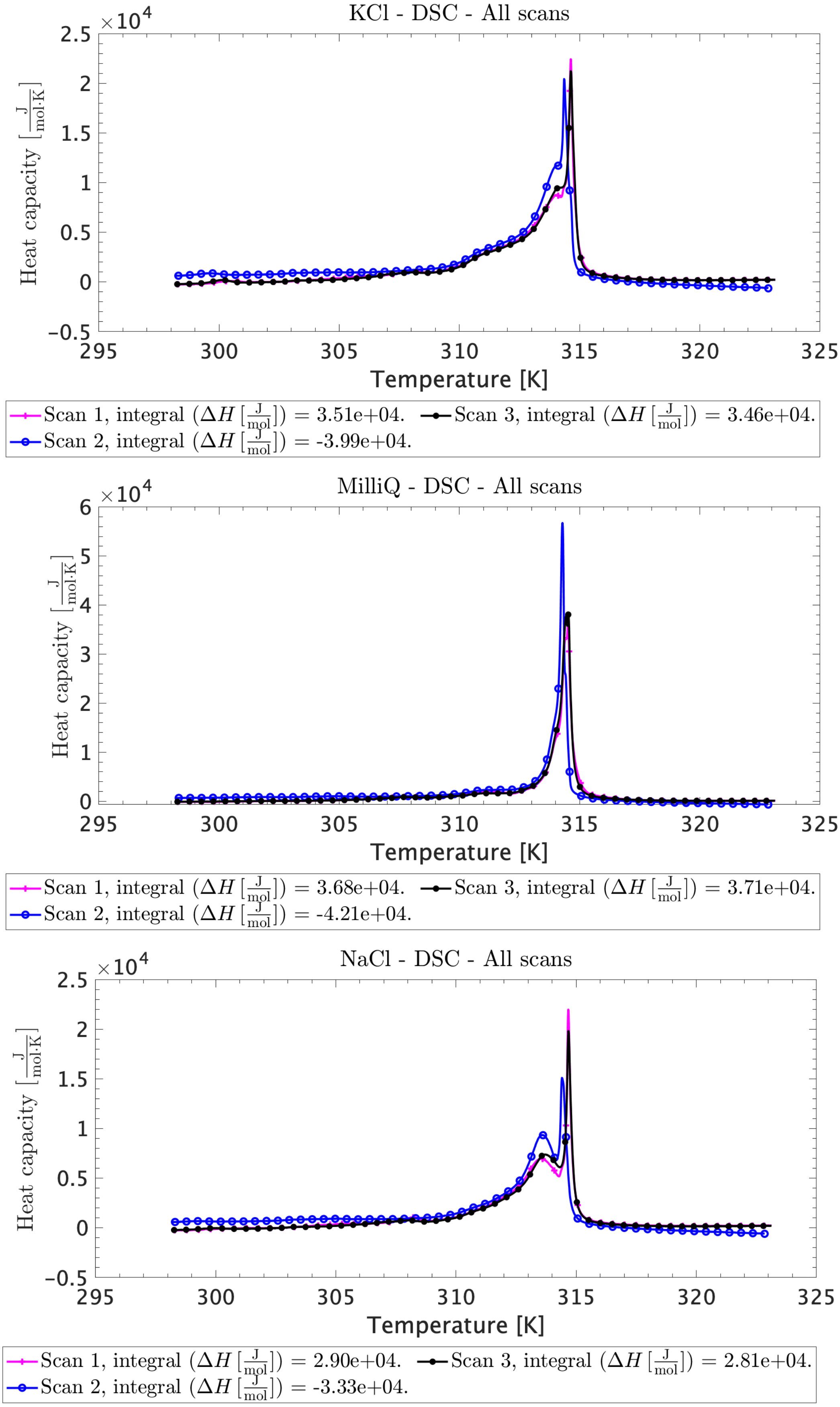
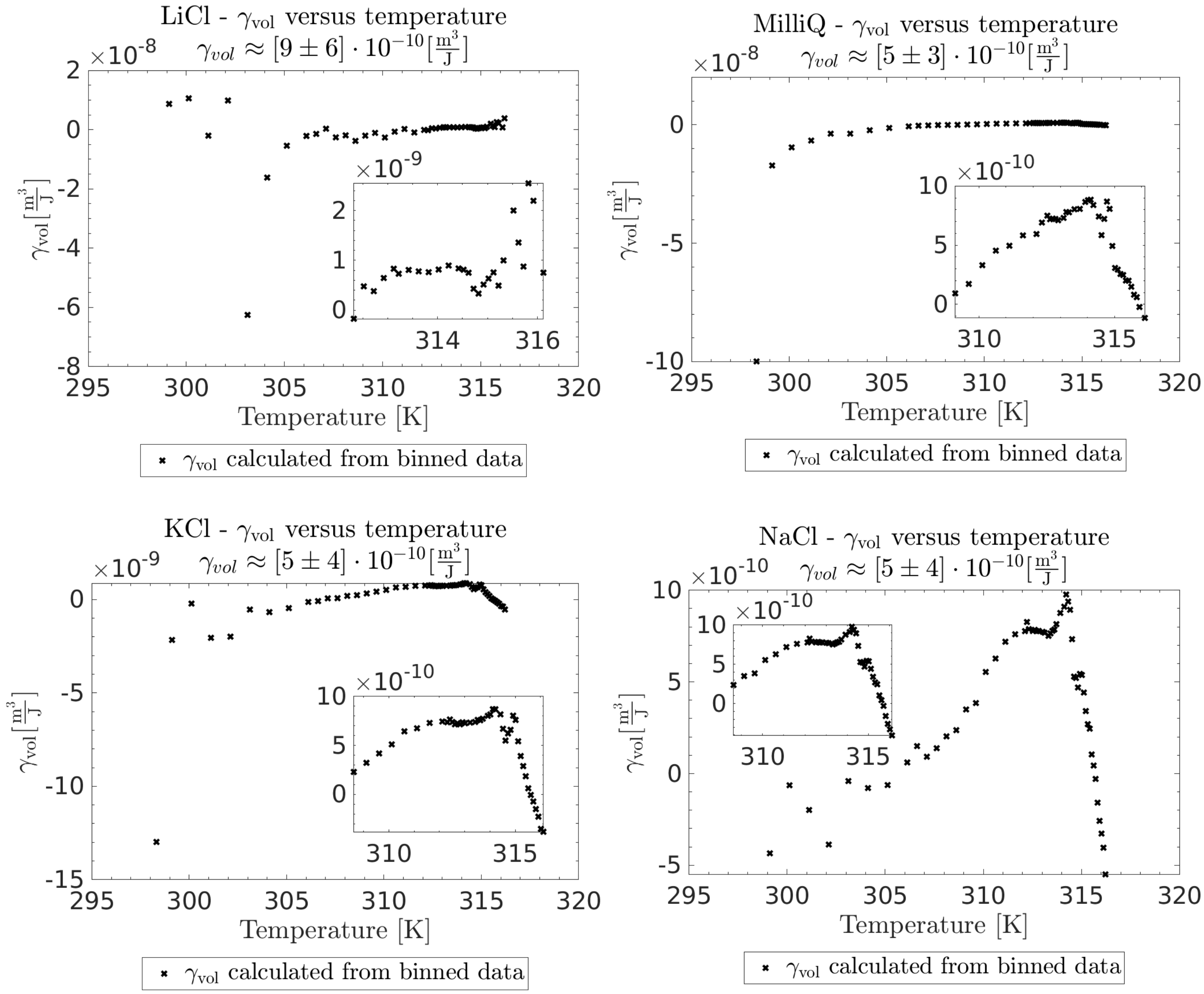


Figure 45: Up- and down-scans of . Similar plot for the LiCl-sample found in section 4.3.1, figure 20.

E.7 Experiment: Binned γ_{vol} data plots


LiCl - γ_{vol} versus temperature $\gamma_{vol} \approx [9 \pm 6] \cdot 10^{-10}[\frac{\mathrm{m}^3}{\mathrm{J}}]$

MilliQ - γ_{vol} versus temperature $\gamma_{vol} \approx [5 \pm 3] \cdot 10^{-10}[\frac{\mathrm{m}^3}{\mathrm{J}}]$

KCl - γ_{vol} versus temperature $\gamma_{vol} \approx [5 \pm 4] \cdot 10^{-10}[\frac{\mathrm{m}^3}{\mathrm{J}}]$

NaCl - γ_{vol} versus temperature $\gamma_{vol} \approx [5 \pm 4] \cdot 10^{-10}[\frac{\mathrm{m}^3}{\mathrm{J}}]$

Figure 46: γ_{vol} binned for every 0.1 K. The subplot zooms in on the range of proportionality.

E.8 Experiment: Binned L data plots

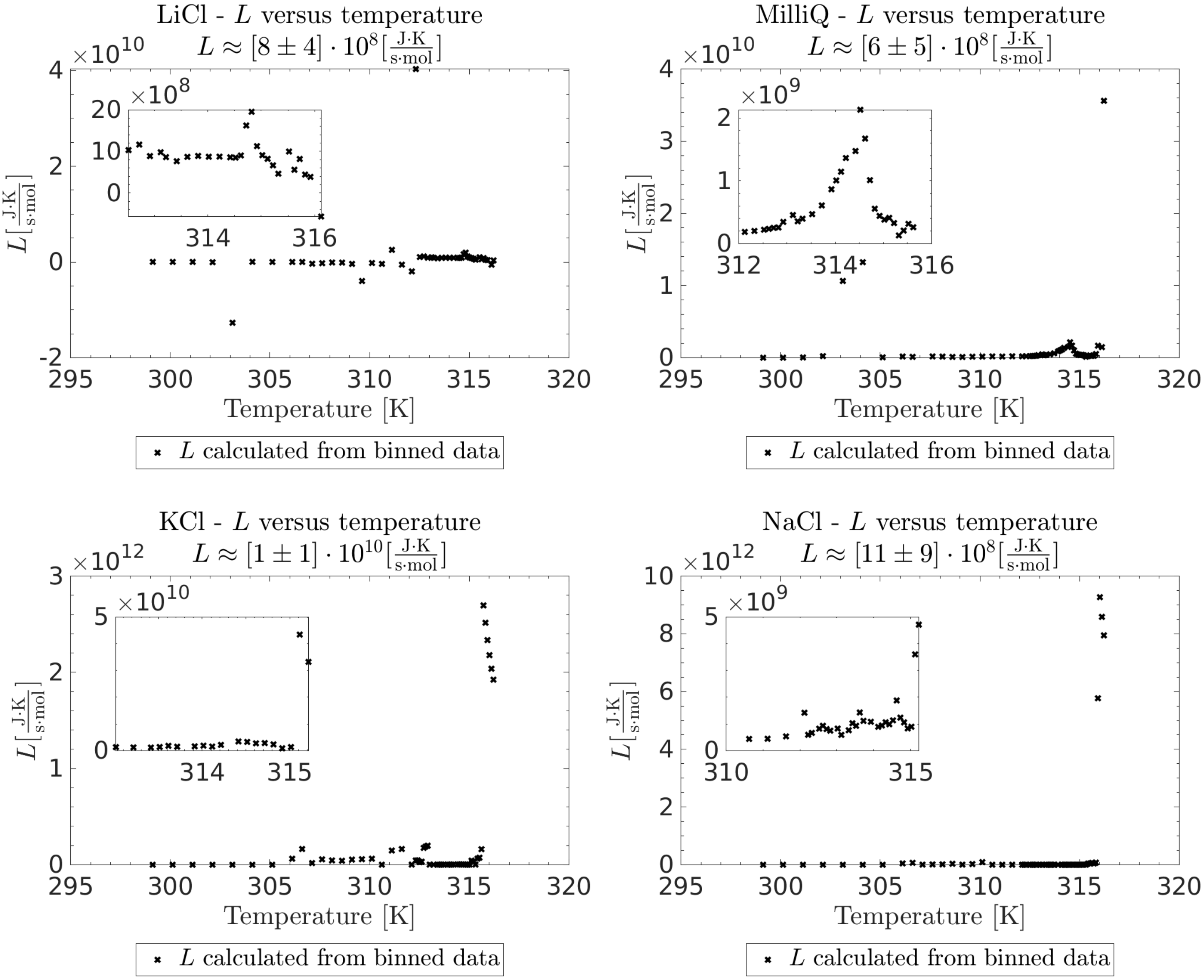


Figure 47: L binned for every 0.1 K. The subplot zooms in on the range of proportionality.

F Guide to the MATLAB directory

Directories are ordered alphabetically. I've used relative paths so things should run smoothly as long as the directory is kept intact. I've used the special character ':' in directory names while on Ubuntu which caused trouble when switching back to macOS. I've corrected ':' to the mutually compatible '-' wherever I could find it, but in case directory problems arise this might be the source.

If the word 'solution' is mentioned in a description, what is meant is the density curve produced by the ODE-solver, not a soliton, which is the end product after further optimization.

This manual includes all dependencies of a given script or function. Dependants can be found by searching through all the dependencies listed under each function for mention of the specific script or function in question using a pdf viewer search tool.

MATLAB® functions allows you to add new outputs to a function without having to backtrack every instance where the function has been called to correct assignments. One simply adds the output to the output list after 'function' has been declared. If a function has four outputs but you only assign three variables in your script, the fourth is simply ignored. If one needs to skip an output in the list, $\sim$ is used as a placeholder.

F.1 The main directory

Contents of the **MATLAB** folder is as follows:

1. Constant dispersion coefficient
    (a) Previous versions of code
    (b) Thrice integrated
    (c) Twice integrated
2. DSC and PPC analyzation
3. Density dependent dispersion coefficient
    (a) Gamma heat capacity
    (b) Gaussian heat capacity
    (c) Thermo heat capacity
4. Shared functions

Constant dispersion coefficient and **Density dependent dispersion coefficient** are the directories for solving the soliton model for constant and density dependent dispersion respectively.

DSC and PPC analyzation is the directory for analyzation of data from the heat capacity and pressure pertubation experiments.

Shared functions includes functions shared by the above three directories, such as recordings of experimental parameters and table values.

The directories **1c** and **3a** are intermediate steps of the work process and will not be discussed further.

F.2 Shared directory structures

The directories share similarities on different levels. The directories **1a**, **1b**, **2**, **3b**, **3c** all share a similar overall structure of their content:

Archived This folder is for manually storing content produced by the **master scripts**. For most parts divided into the subdirectories **Data** and **Plots**.

Data Data produced by the **master scripts** can be set to automatically be saved to this folder. Sometimes the data is divided into subfolders through directories generated by the master script itself. This data can afterwards be moved manually to the **Archived** folder.

Functions This folder contains the majority of scripts used by the **master scripts**. The remainder are located in **Shared functions** (4).

Master scripts This is not a subdirectory, but a collection of files in the top directory from which the code is run.

Plots Plots produced by the **master scripts** can be set to automatically be saved to this folder. These plots can afterwards be moved to the **Archived** folder.

Previous versions of code Some of the mentioned directories contain a folder with intermediate steps of the work process as mentioned before. This content will not be discussed further.

Since the approach to each of the four versions of the soliton model has been practically the same, the directories **1a**, **1b**, **3b** and **3c** shares most of their structure, content and naming of their scripts. The only major differences can be found in the scripts themselves such as the different linearization schemes, the nature of the bisection method used to optimize the initial conditions and the appearance of the 'valley' for the density dependent density curves, which creates a need for parallel treatment of two density curves instead of just one. In the following subsection the full structure for these 4 subdirectories is described, and the subsections afterwards takes on the full structure of the **DSC and PPC analyzation** directory and the **Shared functions** in that order.

F.3 Full structure of soliton directories

Of the directory list in the previous section **Functions** and the **master scripts** need a more thorough explanation.

F.3.1 Functions

calculate_equation_parameters This function is used by **dydt_eq** to calculate different components of the linearization.

Dependencies:

- heat_capacity.m (Gaussian and Thermo heat capacity only)
- Shared functions/load_table_values.m

dydt_eq Linearization of the differential equation. Along with the initial values this function describes the system to be solved. Called by the ODE-solver in **generate_data.m**.

Dependencies:

- calculate_equation_parameters.m
- heat_capacity.m (Gaussian and Thermo heat capacity only)
- Shared functions/load_table_values.m

finding_optimization_interval This function is called in the master script **single_velocity_optimization.m** and determines in what area among the initial values and with what method **optimizing_data.m** will optimize density curves for soliton behaviour.

No dependencies

generate_data.m Generates the density curve related to every initial value (amplitude) in a specified range for a given velocity. Automatically saves data files and log file of the file names and properties to **Data/Raw data**, can amass quite an amount of data. The automatic saving was implemented since the numerical solver in **heat_capacity.m** for **Thermo heat capacity** contains a numerical solver that crashes often. If the ODE solving breaks down for some initial value, the whole script is halted, and all of the previous work achieved by **generate_data.m** up until the crash is lost. One can circumvent this issue by wrapping the ODE-solver in a 'try-catch' command in **generate_data.m** and saving the workspace after each run of the ODE-solver regardless if the solving was successful or not.

Dependencies:

- ODE_options.m
- calculate_equation_parameters.m
- dydt_eq.m
- heat_capacity.m (Gaussian and Thermo heat capacity only)
- jacobian.m
- stop_events.m (Gaussian and Thermo heat capacity only)
- Shared functions/load_table_values.m

heat_capacity.m (Gaussian and Thermo heat capacity only) Provided with a density it calculates the heat capacity. Called by **dydt_eq**. The expression for the **Thermo heat capacity** isn't analytical and is approximated using the build-in numerical solver **vpasolve** which crashes quite a lot.

Dependencies:

- Shared functions/load_table_values.m

jacobian.m According to MATLAB documentation [49] providing the ODE-solver with a description of the jacobian of the linearization will increase accuracy and speed of the solver.

Dependencies:

- calculate_equation_parameters.m
- heat_capacity.m (Gaussian and Thermo heat capacity only)
- Shared functions/load_table_values.m

load_data.m Function for loading a set of data files generated by **generate_data.m**. Takes the directory and ID of the log file as input.

No dependencies

load_opt_structs_from_table.m Used in **plot_velocities.m** to load optimized data from many different directories specified manually in a table.

Dependencies:

- Shared functions/exponential.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

ODE_options.m A separate script for defining ODE options so one does not have to edit **generate_data.m**. Builds a settings menu containing predefined ODE-option configurations saved to a MATLAB® struct. These configurations are defined in the bottom and can then be called in the top of the document. If one wants to try a new settings configuration it can easily be added to the struct and one does not have to delete previous settings configurations. Instead just name the new configuration something unique and change the settings call in the top of the document.

Dependencies:

- calculate_equation_parameters.m
- jacobian.m
- heat_capacity.m (Gaussian and Thermo heat capacity only)
- jacobian.m
- stop_events.m (Gaussian and Thermo heat capacity only)
- Shared functions/load_table_values.m

optimizing_data.m Optimizes density curves by iteratively providing initial values of **generate_data.m** and analyzing the output in search of soliton behaviour. The starting point of the optimization is provided by **finding_optimization_interval**. Contains the bisection method individual to each of the four soliton directories. The output struct is automatically saved to **Data/Optimized data**. OBS: If you resume optimizing on some previously calculated data, make sure to set the same **ODE_options** settings!

Dependencies:

- ODE_options.m
- calculate_equation_parameters.m
- dydt_eq.m
- generate_data.m
- heat_capacity.m (Gaussian and Thermo heat capacity only)
- jacobian.m
- load_data.m
- stop_events.m (Gaussian and Thermo heat capacity only)
- Shared functions/load_table_values.m

plot_opt_struct.m Plots the density curves of the optimized data structure from **optimizing_data.m**.

Dependencies:

- ODE_options.m
- calculate_equation_parameters.m
- heat_capacity.m (Gaussian and Thermo heat capacity only)
- jacobian.m
- stop_events.m (Gaussian and Thermo heat capacity only)
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

plot_opt_structs_from_table.m Used in **plot_velocities.m** to plot the density curves from many different structs of optimized data. The data directories are specified manually in a table.

Dependencies:

- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

plot_raw_data.m Plots all density curves from a set of data files loaded with **load_data.m**.

Dependencies:

- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

stop_events.m (Gaussian and Thermo heat capacity only) This function is included in the ODE-settings by **ODE_options.m**. The ODE-solver runs it parallel with the algorithm. If the criteria stated in the **stop_events.m** are met at a given step of the algorithm, the algorithm is stopped.

No dependencies

F.3.2 Master scripts

heat_capacity_curve.m The directories **Gaussian heat capacity** and **Thermo heat capacity** contain this script for plotting their respective heat capacity curves as a function of density for various parameter setups. This script also includes velocity- and dispersion coefficients plots.

Dependencies:

- Functions/heat_capacity.m (Gaussian only)
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m
- Shared functions/plot_settings.m

plot_ode_solvers.m The **Integrated** directory contains a script for exploring the effect of different ODE-algorithms.

Dependencies:

- Functions/load_opt_structs_from_table.m
- Functions/plot_opt_structs_from_table.m
- Shared functions/exponential.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

plot_velocities.m This script plots and analyzes multiple sets of optimized data generated by **single_velocity_optimizatio** One has to manually add the directories of the optimized data. The data sets can be stored conveniently in the data subfolder of **Archived**.

Dependencies:

- Functions/load_opt_structs_from_table.m
- Functions/plot_opt_structs_from_table.m
- Shared functions/exponential.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

single_velocity_optimization.m The optimization script for generating solitons. This script can generate quite an amount of data. Both intermediary density curves and the final optimization MATLAB® struct are automatically saved to the **Data** folder in the subfolders **Raw data** and **Optimized data** respectively.

Dependencies:

- Functions/ODE_options.m
- Functions/calculate_equation_parameters.m
- Functions/dydt_eq.m

- Functions/finding_optimization_interval.m
- Functions/generate_data.m
- Functions/heat_capacity.m
- Functions/jacobian.m
- Functions/load_data.m
- Functions/optimizing_data.m
- Functions/plot_opt_struct.m
- Functions/plot_raw_data.m
- Functions/stop_events.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

testing_ODE_option.m This script can be used to test different settings of the ODE-solver through **Functions/ODE_options.m** This script generates and automatically saves density curves but does not optimize to obtain soliton behaviour, thus the output is not as immense as **single_velocity_optimization.m**.

Dependencies:

- Functions/ODE_options.m
- Functions/calculate_equation_parameters.m
- Functions/dydt_eq.m
- Functions/generate_data.m
- Functions/heat_capacity.m
- Functions/jacobian.m
- Functions/load_data.m
- Functions/stop_events.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

F.4 Full structure of DSC and PPC directory

Of the directory list in section F.2, only **Functions** and the **master scripts** need a more thorough explanation.

F.4.1 Functions

decreasing_exponential.m General function describing a decreasing exponential. Used by dependencies in **fit_data_PPC.m** for constructing relaxation time fits.

No dependencies

experimental_parameters.m Experimental parameters and quantities used for unit conversion in the DSC and PPC experiments can be logged here. It is possible to save multiple sets of parameters. The parameter index should correspond to the data index in **load_data.m**.

No dependencies

fit_data_DSC.m Fit peaks in DSC data with or without baseline. Initial values are specified in **initial_gues_DSC.m**. 2 types of peaks are specified (Gaussian, Van't Hoff), more can be added. The baseline is modeled by a nth order polynomial. This fit will not converge if the order of the peaks is much larger than the baseline (likely any other data set than contaminated references). Fit the major peaks using this function and proceed with **subtract_peaks_DSC.m** and **remove_baseline_DSC.m** instead.

Dependencies:

- Hoffs_law.m
- Hoffs_law_with_baseline.m
- fit_data_DSC.m
- gaussians.m
- gaussians_with_baseline.m
- pol.m
- Shared functions/load_table_values.m

fit_data_PPC.m Fit relaxation times for every sample type in an experiment. For each sample type one has to predefine what file is to be used as water reference and add the index in the struct 'water_indexes.list' in the master script **PPC_analyzation.m**. Each relaxation time is fitted with a numerical convolution, so the process is very slow and sensitive to the initial guess of parameters.

Dependencies:

- P_exp.m
- P_lipid.m
- P_water.m
- R_instr.m

gamma_function.m Not used.

No dependencies

gaussians.m General function describing a sum of gaussians. Used by **fit_data_DSC.m** for fitting peaks.

No dependencies

gaussians_with_baseline.m General function describing a sum of gaussians and a polynomial baseline. Used by **fit_data_DSC.m** for fitting data.

Dependencies:

- gaussians.m
- pol.m

Hoffs_law.m General function describing a sum of single melting events modeled by Hoff's law. Used by **fit_data_DSC.m** for fitting peaks.

Dependencies:

- Shared functions/load_table_values.m

Hoffs_law_with_baseline.m General function describing a sum of single melting events modeled by Hoff's law and a polynomial baseline. Used by **fit_data_DSC.m** for fitting data.

Dependencies:

- Hoffs_law.m
- pol.m
- Shared functions/load_table_values.m

initial_guess_DSC.m Initial guesses for parameters fitted in **fit_data_DSC.m**. The initial guess defines the fit. For instance one can add an extra peak to a specific fit by increasing 'n_peaks' with 1 and adding the extra set of parameters in 'initial_guess_array'.

Dependencies:

- Shared functions/load_table_values.m

initial_guess_PPC.m Initial guesses for relaxtion time fit parameters. **fit_data_PPC.m** will attempt to fit all run files for which an initial guess for the parameters has been made.

No dependencies:

linear.m General linear function used in **load_data.m** to fit the scan rate in the DSC experiment.

No dependencies

load_data.m Loads all data for both DSC and PPC experiments into three outputs: A DSC struct, a PPC struct and a data menu describing the sample names in each set of experiments.

Dependencies:

- experimental_parameters.m
- initial_guess_DSC.m
- initial_guess_PPC.m
- linear.m

P_exp.m General function describing the power output of the PPC experiment. Used for fitting data in **fit_data_PPC.m**. Contains a numerical convolution.

Dependencies:

- P_lipid.m
- P_water.m
- R_instr.m

P_lipid.m General function describing the lipid relaxation output of the PPC experiment. Part of **P_exp.m** used in **fit_data_PPC.m**.

No dependencies

plot_all_scans_DSC.m Plots the DSC scans for a selection of experiments (i.e. up- and downscans) of a certain data type as to compare for hysterisis and hydration effects.

Dependencies:

- Shared functions/plot_settings.m

plot_data_DSC.m Plots raw data of the DSC experiment.

Dependencies:

- Shared functions/plot_settings.m

plot_data_PPC.m Plots raw data of the PPC experiment.

Dependencies:

- Shared functions/plot_settings.m

plot_eq_release_and_jump.m Plot the full run from a single PPC file (equilibration, pressure relase and jump).

Dependencies:

- Shared functions/plot_settings.m

plot_fit_DSC.m Plot the fit obtained in **fit_data_DSC.m**.

Dependencies:

- Shared functions/plot_settings.m

plot_total_fit_PPC.m Plot the water reference fit and the relaxation time fits obtained in **fit_data_PPC.m** on top of data. One can choose only to plot a certain range of fits.

Dependencies:

- Shared functions/plot_settings.m

pol.m General function describing a polynomial of nth order. Used by **fit_data_DSC.m** for fitting data.

No dependencies

print_PPC_parameters_in_Command_Window.m This function is very practical for determining good initial guesses for parameters when fitting relaxation times. Prints the parameters of all fitted run files for a given sample in the command window.

No dependencies

P_water.m General function describing the water reference of the PPC experiment. Part of **P_exp.m** used in **fit_data_PPC.m**.

No dependencies

remove_baseline_DSC.m Subtracts a previously fitted baseline from a specified set of data. 'n_peaks' and 'pol_order' specifies what fit to take the baseline from.

Dependencies:

- pol.m

R_instr.m General function describing the normaliized instrument response function in the PPC experiment. Part of **P_exp.m** used in **fit_data_PPC.m**.

No dependencies

split_fit_DSC.m When one is fitting multiple peaks and maybe an additional baseline, this function can be used to plot each fit component on top of each other instead of summing them together.

Dependencies:

- Hoffs_law.m
- gaussians.m
- pol.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

split_fit_PPC.m This fit plots the total relaxation time fit and its components: Water reference fit, instrument response fit and lipid response fit in separate subplots.

Dependencies:

- P_lipid.m
- R_instr.m
- Shared functions/plot_settings.m

subtract_peaks_DSC.m This function is the preliminary stage to **remove_baseline_DSC.m** in case baseline and peaks cannot be fitted in one go. First it removes the peaks fitted with **fit_data_DSC.m**. Then it fits the baseline with a polynomial of the requested order. The fit is weighted such that data on the edges have exponentially more influence and high heat capacities have exponentially less influence, plots of these weight distributions can be activated by setting 'plot_weights' to 1 in **DSC_analyzation.m**.

Dependencies:

- Hoffs_law.m
- Hoffs_law_with_baseline.m
- fit_data_DSC.m
- gaussians.m
- gaussians_with_baseline.m
- plot_fit_DSC.m
- pol.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

subtract_reference_DSC.m Fits the reference with the requested fit, then determines the RMS of the residuals. This function also defines the weights used to in **fit_data_DSC** to be uniformly $\frac{1}{\mathrm{RMS}^2}$ (in the subfunction 'subtraction'), so currently the data fitting taking place is de facto not weighted, but the structure is prepared.

Dependencies:

- Hoffs_law.m
- Hoffs_law_with_baseline.m
- fit_data_DSC.m
- gaussians.m
- gaussians_with_baseline.m
- plot_fit_DSC.m
- pol.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

F.4.2 Master scripts

DSC_analyzation.m Analyzes DSC data. Converts the differential power data in the data file to heat capacity data, subtracts reference data, corrects the baseline and saves the final data to a MATLAB® file. Plots of the initial, intermediate and final steps can be activated. Currently only one experiment of 4 samples is being analyzed, but the code can be expanded to contain more than one experiment.

Dependencies:

- Functions/load_data.m
- Functions/Hoffs_law.m
- Functions/Hoffs_law_with_baseline.m

- Functions/fit_data_DSC.m
- Functions/gaussians.m
- Functions/gaussians_with_baseline.m
- Functions/plot_data_DSC.m
- Functions/plot_fit_DSC.m
- Functions/pol.m
- Functions/remove_baseline_DSC.m
- Functions/split_fit_DSC.m
- Functions/subtract_peaks_DSC.m
- Functions/subtract_reference_DSC.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

Plot_fit_parts.m Plots **Functions/P_exp.m, Functions/P_lipid.m, Functions/P_water.m** and **Functions/R_instr.m** for manually defined sets of parameters. One can then by eye compare these plots to raw data or fit plots of **PPC_analyzation.m** to obtain good initial guesses or improve the fitting process for the relaxation time.

Dependencies:

- Functions/P_exp.m
- Functions/P_lipid.m
- Functions/P_water.m
- Functions/R_instr.m
- Functions/decreasing_exponential.m
- Shared functions/plot_settings.m

plot_proportionality.m Plots the relaxation times and thermal volume expansion coefficients obtained with **PPC_analyzation.m** on top of the heat capacity curves from **DSC_analyzation.m** for all samples in an experiment. It also derives the phenomenological constant L and the proportionality constant γ_V.

Dependencies:

- Functions/experimental_parameters.m
- Shared functions/load_table_values.m
- Shared functions/plot_settings.m

PPC_analyzation.m Analyzes PPC data. Fits relaxation times for pressure release only at the current moment, but the structure permits for pressure jump fits to be added without difficulty. Data is saved to a MATLAB$^{®}$ file. Plots of the initial, intermediate and final steps can be activated. The fitting process involves a convolution and is very slow and sensitive to initial guess of parameters. Currently only one experiment of 4 samples is being analyzed, but the code can be expanded to contain more than one experiment.

Dependencies:

- Functions/load_data.m
- Functions/P_exp.m
- Functions/P_lipid.m
- Functions/P_water.m
- Functions/R_instr.m
- Functions/fit_data_PPC.m
- Functions/plot_data_PPC.m
- Functions/plot_total_fit_PPC.m
- Functions/split_fit_PPC.m
- Shared functions/plot_settings.m

F.5 The 'Shared Functions' directory

exponential.m General function describing an exponential. Mainly used by **load_opt_structs_from_table.m** in **plot_velocities.m** in the soliton directories to fit the ends of the solitons.

No dependencies

load_table_values.m Contains table values used in the thermodynamic model for heat capacity and the gaussian approximation and more. Build as a settings menu like the **Functions/ODE_options.m**-file in order to test the effect of various parameters. OBS: If you resume optimizing on some previously calculated data, make sure to set the same **load_table_values.m** settings!

No dependencies

plot_settings.m General plot settings such as font sizes and color cycles for iterative plotting.

No dependencies

F.6 Filenames and overwriting

Since I am running the ODE-solver repeatedly in each optimization, one can easily overwrite previously saved data from the same run if the files are not named systematically. This is prevented by wrapping the data in a subdirectory of date and time on the form:

'year_month_day/hour_min/sec/filename.filetype'

Example: '2021_1_12/19_53/22.38/v_0.800_amp_1.2553e-07.mat'.

It is done consistently for all data, also for the **DSC_PPC analysis** directory. Plots and data can be linked by these directories if the system is not disrupted afterwards. Initially I chose to nest the output instead of creating elaborate file names since different operating systems handle delimiters in file names differently, and a time stamp down to milliseconds with no delimiters can make a file name difficult to read. I have changed my mind, but haven't corrected this issue.

F.7 Bisection methods

Optimizing has 2 modes: If the optimization interval has been detected, bisection method (mode 1). Here a new solution is being generated in between the boundaries of the interval to be correctly classified as either upper or lower boundary of new, smaller interval.
Else keep searching for the other boundary by linearly changing the initial condition (mode 2).

Bisection method, twice integrated
Finding optimization interval:

Lower boundary: Oscillating solution, sum of maxima ≥ 2.

Upper boundary: Diverging solution, one maximum and $\Delta\rho_{\text{end}} \leq 0$, or one minima and $\Delta\rho_{\text{end}} \geq 0$.

Optimizing initial condition:

Mode 1: Bisection method

> New lower boundary: Oscillating solution, sum of maxima > 1.
>
> New upper boundary: Diverging solution, one maximum or one minima.

Mode 2: Search for second boundary

> Look for lower bound: Find oscillating solution, sum of maxima > 1.
>
> Look for upper bound: Find diverging solution, one maximum or one minima.

Bisection method, thrice integrated
Finding optimization interval:

Lower boundary: Any solution not fitting the criteria of upper boundary.

Upper boundary: Complex solutions, detected by > 1 minima or > 1 maxima.

Optimizing initial condition:

Mode 1: Bisection method

> New lower boundary: Any solution not fitting the criteria of upper boundary.
>
> New upper boundary: If > 1 minima or > 1 maxima.

Mode 2: Search for second boundary

> Look for lower bound: If precisely 1 maximum and no minima.
>
> Look for upper bound: If > 1 minima or > 1 maxima.

Optimization method, Gaussian or thermodynamic heat capacity
Finding optimization interval:

Find the valley: Valley solutions have one maximum and the end of the curve is lower than the initial value, $\Delta\rho_{\text{end,stop}} \leq \Delta\rho_{\text{end,start}}$.

Find a direction: The solution of largest amplitude in the valley constitutes the lowest boundary on the left side interval, whereas the solution of smallest amplitude is set to the highest boundary on the right side interval. I look for an indication of direction in the valley in the following way:

- Look for a valley bottom: At the bottom of the valley, the maxima of the solution, $\Delta\rho_{\max}$, reaches a minima out of all solutions in the valley.
- Detect sides: If the valley bottom isn't found, try looking for the sides instead.
  - Valley left side: $\Delta\rho_{\max}$ of the solutions are increasing with increasing amplitude.
  - Valley right side: $\Delta\rho_{\max}$ of the solutions are decreasing with increasing amplitude.

If valley bottom or side is found: Look for nearest oscillating solution to the valley side(s):

- Valley left side: Sum of minima and maxima > 1, amplitude of solution larger than that of highest valley amplitude. Upper boundary on the left side interval.
- Valley right side: Sum of minima and maxima > 1, amplitude of solution less than that of lowest valley amplitude. Lower boundary on the right side interval.

Optimizing initial condition:

Mode 1: Bisection method

- Right side:
  - New lower boundary: If sum of minima and maxima > 1.
  - New upper boundary: If sum of minima and maxima $= 1$.
- Left side:
  - New lower boundary: If sum of minima and maxima $= 1$.
  - New upper boundary: If sum of minima and maxima > 1.

Mode 2: Search for second boundary: I optimize from the inside of the valley and out, so regardless of direction I only need to act when an oscillating solution arises, meaning sum of minima and maxima > 1. In this case:

- Right side: Lower boundary is found.
- Left side: Upper boundary is found.