Advanced data analysis for traction force microscopy and data-driven discovery of physical equations

Inaugural-Dissertation

zur
Erlangung des Doktorgrades
der Mathematisch-Naturwissenschaftlichen Fakultät
der Universität zu Köln
vorgelegt von

Yunfei Huang
aus Jingde, China

ii

"All things are numbers."

Pythagoras of Samos

Abstract

The plummeting cost of collecting and storing data and the increasingly available computational power in the last decade have led to the emergence of new data analysis approaches in various scientific fields. Frequently, the new statistical methodology is employed for analyzing data involving incomplete or unknown information. In this thesis, new statistical approaches are developed for improving the accuracy of traction force microscopy (TFM) and data-driven discovery of physical equations.

TFM is a versatile method for the reconstruction of a spatial image of the traction forces exerted by cells on elastic gel substrates. The traction force field is calculated from a linear mechanical model connecting the measured substrate displacements with the sought-for cell-generated stresses in real or Fourier space, which is an inverse and ill-posed problem. This inverse problem is commonly solved making use of regularization methods. Here, we systematically test the performance of new regularization methods and Bayesian inference for quantifying the parameter uncertainty in TFM. We compare two classical schemes, L1- and L2-regularization with three previously untested schemes, namely Elastic Net regularization, Proximal Gradient Lasso, and Proximal Gradient Elastic Net. We find that Elastic Net regularization, which combines L1 and L2 regularization, outperforms all other methods with regard to accuracy of traction reconstruction. Next, we develop two methods, Bayesian L2 regularization and Advanced Bayesian L2 regularization, for automatic, optimal L2 regularization. We further combine the Bayesian L2 regularization with the computational speed of Fast Fourier Transform algorithms to develop a fully automated method for noise reduction and robust, standardized traction-force reconstruction that we call Bayesian Fourier transform traction cytometry (BFTTC). This method is made freely available as a software package with graphical user-interface for intuitive usage. Using synthetic data and experimental data, we show that these Bayesian methods enable robust reconstruction of traction without requiring a difficult selection of regularization parameters specifically for each data set.

Next, we employ our methodology developed for the solution of inverse problems for automated, data-driven discovery of ordinary differential equations (ODEs), partial differential equations (PDEs), and stochastic differential equations (SDEs). To find the equations governing a measured time-dependent process, we construct dictionaries of non-linear candidate equations. These candidate equations are evaluated using the measured data. With this approach, one can construct a likelihood function for the candidate equations. Optimization yields a linear, inverse problem which is to be solved under a sparsity constraint. We combine Bayesian compressive sensing using Laplace priors with automated thresholding to develop a new approach, namely automatic threshold sparse Bayesian learning (ATSBL). ATSBL is a robust method to identify ODEs, PDEs, and SDEs involving Gaussian noise, which is also referred to as type I noise. We extensively test the method with synthetic datasets describing physical processes. For SDEs, we combine data-driven inference using ATSBL with a novel entropy-based heuristic for discarding data points with high uncertainty. Finally, we develop an automatic iterative sampling optimization technique akin to Umbrella sampling. Therewith, we demonstrate that data-driven inference of SDEs can be substantially improved through feedback during the inference process if the stochastic process under investigation can be manipulated either experimentally or in simulations.

Kurzzusammenfassung

In vielen Bereichen der Naturwissenschaften haben die in den letzten Jahrzehnten stark sinkenden Kosten für Rechenleistung, sowie für die Speicherung großer Datenmengen, zur Herausbildung neuer Methoden der Datenanalyse geführt. Eine häufige Anwendung solcher Methoden besteht aus der Analyse und Interpretation von Datensätzen die unvollständige Informationen enthalten. In dieser Doktorarbeit werden statistische Methoden entwickelt um einerseits die Genauigkeit der Zugkraftmikroskopie (ZKM) zu verbessern und andererseits auch eine datenbasierte Identifizierung von Gleichungen zur Beschreibung physikalischer Systeme zu verbessern.

Die ZKM ist eine vielseitig anwendbare Methode um die räumliche Verteilung von Zugkräften zu rekonstruieren, die von Zellen auf elastischen Gelsubstraten ausgeübt werden. Das Zugkraftfeld wird mit Hilfe eines linearen mechanischen Models berechnet, welches gemessene Substratverschiebung mit der gesuchten, zellgenerierten Spannung im Real- oder Fourierraum verbindet. Dies ist in der Regel ein schlecht konditioniertes, inverses mathematisches Problem. Für gewöhnlich wird das Problem mit Hilfe von Regularisierungsmethoden gelöst. In dieser Arbeit werden zunächst unterschiedliche Regularisierungsmethoden systematisch miteinander verglichen. Wir vergleichen zwei klassische Schemata, die L1- und L2 Regularisierung, mit drei bislang ungetesteten Schemata und zwar mit dem Elastic Net, dem Proximal-Gradient Lasso und dem Proximal-Gradient Elastic Net. Es wird festgestellt, dass die Elastic Net Regularisierung, welche L1 und L2 Regularisierung kombiniert, alle anderen Methoden im Bezug auf die Genauigkeit der Zugkraftrekonstruktion übertrifft. Als Nächstes entwickeln wir zwei Methoden für eine automatisierte, optimale Regularisierung, die wir Bayessche L2 Regularisierungen nennen. Mit Hilfe synthetischer und experimenteller Daten zeigen wir, dass die Bayessche Methode eine robuste Rekonstruierung der Zugkraft ermöglicht ohne eine Wahl der Regularisierungsparameter speziell für jeden Datensatz zu erfordern. Als weitere Verbesserung der Bayesschen L2 Regularisierung führen wir eine schnelle Berechnung der nötigen Größen im Fourier Raum ein. Das Ergebnis ist eine vollständig automatisierte Methode zur Rauschreduktion welche robuste sowie standardisierte Zugkraftrekonstruktion erlaubt. Diese Methode wird als Softwarepaket mit einer grafischen Benutzeroberfläche frei zugänglich gemacht.

Algorithmen zur Lösung inverser Probleme, wie sie für die ZKM wichtig sind, finden auch in vielen anderen Bereichen Anwendung. In den letzten Jahren insbesondere für das so genannte maschinelle Lernen. Basierend auf zuvor für ZKM erprobten Regularisationsmethoden werden in dieser Arbeit verschiedene Ansätze für die datenbasierte Inferenz gewöhnlicher Differentialgleichungen (GDGLs), partieller Differentialgleichungen (PDGLs), und stochastischer Differentialgleichungen (SDGLs) erprobt. Um die bestimmende Gleichung für einen gemessenen, zeitabhängigen Prozess zu finden, werden Bibliotheken verschiedener Gleichungskandidaten angelegt. Die Optimierung der Wahrscheinlichkeitsfunktion für die verschiedenen Gleichungskandidaten verlangt die Lösung ein linearen, inversen Systems. Hierfür wird eine Bayessche L1 Regularisierung mit einer iterativen Elimination überflüssiger Ergebniskomponenten kombiniert. Die entwickelte Methode mit dem Akronym ATSBL ist für die Inferenz von GDGLs, PDGLs und SDGLs verwendbar. Ein umfangreicher Test mit synthetischen Datensätzen zur Beschreibung physikalischer Prozesse ist erfolgt. Für SDGLs kombinieren wir ATSBL mit einer neuen entropiebasierten Heuristik um Datenpunkte mit großer Messungenauigkeit zu eliminieren. Zu guter Letzt wird eine iterative Stichprobenoptimierung ähnlich dem Umbrella Sampling entwickelt, um sie mit ATSBL zu kombinieren. Anhand von Beispielen wird gezeigt, wie durch gezielte Störung eines gemessenen stochastischen Prozesses die dem Prozess zugrundeliegenden Differentialgleichungen genauer bestimmt werden können.

Acknowledgements

First of all, I would like to express my gratitude toward Prof. Dr. Gerhard Gompper and Prof. Dr. Benedikt Sabass for providing me with this PhD position at the Theoretical Soft Matter and Biophysics institute (ICS-2/IAS-2) at Forschungszentrum Jülich three years ago and supervising me in the last more than three years for my PhD thesis. Thank you for introducing me to the fantastic world of advanced data analysis for the biophysical problems and something I had never thought possible when I was in the previous studies. Thank you for teaching me both intuition and rigor in the exploring and studying a new field. Thank you for sometimes subtly pushing me back towards thinking about the larger questions and the theoretical reasons when we solved a minor numerical or mathematical issue. Thank you for teaching me, both consciously and unconsciously, how good theoretical biophysics is done. Thank you for carefully teaching and instructing me how to conceive and write a fantastic scientific article. Thank you also for providing me many opportunities to learn broadly and to participate the biophysical conferences and schools. It is your excellent ideas and encouragement that have enable me to do this work presented in this thesis. I will keep fond memories of these three years in Jülich learning from you and working with you and hope that our paths will cross again many times in near the future.

I also would like to express my gratitude to Prof. Dr. Dirk Witthaut for accepting to review my doctoral thesis and to Prof. Dr. Markus Braden for accepting to act as the chair for my doctoral thesis defense committee.

I want to thank Prof. Dr. Rudolf Merkel, Dr. Christoph Schell, Prof. Dr. Tobias B. Huber, and Nils Hersch for the good collaboration on the project of traction force microscopy and providing the experimental test data. I am also grateful to our collaborators Prof. Dr. S. V. Plotnikov (University of Toronto) and C. Schell (AlbertLudwigs-University Freiburg) in the study of a Bayesian traction force microscopy method with automated denoising in a user-friendly software package for providing the experimental test data.

I thank all my colleagues at the Forschungszentrum Jülich not only for your direct helps in the physics, mathematics, and computer techniques, but also for your enthusiasm for discussing wide-ranging topics in science and beyond to shape and develop my own thinking more like a scientist. It is the precious moments we shared during and outside of work. I then want to thank all members of our group Ahmet Nihat Şimşek, Andrea Bräutigam, and Kai Zhou for discussing the project and together through the good times and the difficult times.

A special thank you goes to my supervisors, my wife, Carlos Plúa, Andrea Bräutigam, Chaojie Mo, Kai Zhou, and Ahmet Nihat Şimşek for correcting this manuscript and showing great patience with my English. I also would like to thank Christian Philipps and Joscha Mecke for helping the translation of this German abstract.

Finally, I want to express my deep gratitude towards all members of my family and all close friends who even if sometimes from afar have always been with me for their too many encouragement and support during these years.

<div align="right">

Yunfei Huang
Forschungszentrum Jülich
August 2020

</div>

x

Contents

List of Figures

List of Tables

List of Abbreviations

ECM	Extracellular Matrix
TFM	Traction Force Microscopy
STED	Stimulated Emission Depletion
BEM	Boundary Element Method
DTMA	Deviation of Traction Magnitude at Adhesions
DTMB	Deviation of Traction Magnitude in the Background
SNR	Signal to Noise Ratio
DMA	Deviation of the traction Maximum at Adhesions
L2	L2 Regularization
L1	L1 Regularization
EN	Elastic Net Regularization
PGL	Proximal Gradient Lasso
PGEN	Proximal Gradient Elastic Net
BL2	Bayesian L2 Regularization
ABL2	Advanced Bayesian L2 Regularization
BCS	Bayesian Compression Sensing
BCSL	Bayesian Compression Sensing using Laplace prior
BL	Bayesian Lasso
BEN	Bayesian Elastic Net
IRLS	Iterative Reweighed Least Squares
FTTC	Fourier Transform Traction Cytometry
BFTTC	Bayesian Fourier Transform Traction Cytometry
MLE	Maximum Likelihood Estimation
MAP	Maximum a Posteriori
GCV	Generalized Cross Validation
MCMC	Markov chain Monte Carlo
ODE	Ordinary Differential Equations
PDE	Partial Differential Equations
SDE	Stochastic Differential Equations
SVD	Singular Value Decomposition
RVM	Relevance Vector Machine
ATSBL	Automatic Threshold Sparse Bayesian Learning
UP	Universal Prediction
AISO	Automatic Iterative Sampling Optimization

List of Symbols

I	Self-information or Information content
X_i	State
H	Total uncertainty or Entropy
$\mathbf{g}$	Observed vector
N	Size of $\mathbf{g}$
$\mathbf{\Phi}$	Observed matrix
$\mathbf{w}$	Unknown vector
M	Size of $\mathbf{w}$
$\mathbf{s}$	Noise vector
$\hat{\mathbf{w}}$	Reconstructed vector
$\mathbf{I}$	Unit matrix
λ_1	L1 regularization parameter
λ_2	L2 regularization parameter
f_{GCV}	Generalized cross-validation (GCV) function
f_{QOC}	Quasi-optimality criterion function
$\mathscr{W}$	Unitary wavelet transform
$\mathscr{W}^*$	Inverse wavelet transform
$\mathcal{G}$	Observable random variable for $\mathbf{g}$
$\mathcal{W}$	Non-observable random variable for $\mathbf{w}$
$\mathcal{S}$	Random variable for $\mathbf{s}$
$p(\mathbf{w})$	Prior
Z_{w}	Normalization of prior
$p(\mathbf{g}|\mathbf{w})$	Likelihood
Z_{g}	Normalization of likelihood
$1/\alpha$	Variance of prior
$1/\beta$	Variance of likelihood
$p(\mathbf{w}|\mathbf{g}, \alpha, \beta)$	Posterior
Z_{K}	Normalization of posterior
$p(\mathbf{g}|\alpha, \beta)$	Evidence or Marginal likelihood
$\boldsymbol{\mu}$	Mean of posterior
$\boldsymbol{\Sigma}$	Covariance of posterior
$\mathbf{w}_{\mathrm{MP}}$	Maximum posterior
$\mathbf{A}$	Hessian
$\mathbf{C}$	Covariance of evidence
$\mathcal{L}$	Logarithm of evidence
$U_i(\mathbf{x})$	Continuous displacement field
$F_j(\mathbf{x}')$	Continuous force field
$\mathbf{x} = (x_1, x_2)$	Two-dimensional vector

Ω	Whole surface of the substrate
E	Young's modulus
ν	Poisson's ratio
G_{ij}	Green's function in real space
δ_{ij}	Kronecker delta function
z	Vertical position of the substrate
u_i	Discrete displacement field
f_j	Discrete traction force field
h	Shape function
$\mathbf{M}$	Coefficient matrix between u_i and f_j
$\mathbf{k} = (k_1, k_2)$	Wave vector
$\tilde{G}_{ij}$	Green's function in Fourier space
$\mathbf{t}^{\mathrm{real}}$	Real traction
$\mathbf{t}^{\mathrm{recon}}$	Reconstructed traction
$\hat{\alpha}, \hat{\beta}$	Optimal parameter
$\hat{\lambda}$	Optimal regularization parameter
t	Time
$\mathbf{z}, u$	Measurement data
$\dot{\mathbf{z}}$	Time derivative of measurement data
$u_x, u_{xx}, \cdots$	Space derivative of measurement data
$\boldsymbol{\Theta}$	Augmented library
$\mathbf{z}^{P_2}$	Higher polynomials
g	Drift function
h	Diffusion function
$\hat{L}(\mathbf{x}, t)$	Fokker-Plank operator
$D^{(n)}$	Kramers-Moyal (KM) coefficients
$U(\mathbf{x}, t)$	Potential energy function
Q	Number of bins
Δt	Time step
$\mathcal{L}_{EN}$	Elastic net regularization
$\mathcal{L}_{MSE}$	Mean squared error
$\mathcal{L}_{Reg}$	Regression based cost function
u_{recon}	Reconstructed dataset
T	Probability threshold

To my family…

Chapter 1

Introduction

1.1 Historical development of data analysis methods

More than 60 years ago, John Tukey [1] envisioned a future of science that is focused on learning from data, or "*data analysis*", which is a process of using statistical practices to describe, represent, organize, evaluate, and interpret data. Nowadays, there exists a large number of automated techniques for generating, collecting, and storing trillions of measurement data points every day in the whole world. For this reason, data analysis has become increasingly important in various scientific and engineering fields.

To introduce the concept of statistical learning, we start by considering the classical regression problem which aims at estimating the relationship between a given data set consisting of a N-dimensional vector of input data $\mathbf{g}$ (often called observed data) with given parameters in a $N \times M$ matrix $\mathbf{\Phi}$ and a M-dimensional vector of dependent data $\mathbf{w}$ (often called the output- or unobserved data). The most common form of regression analysis is *linear regression* [2, 3], in which the model is written in matrix notation as

$$\mathbf{g} = f(\mathbf{w}) = \mathbf{\Phi}\mathbf{w} + \mathbf{s}, \tag{1.1}$$

where $\mathbf{s}$ is an observed or unobserved error variable, often called noise. In more general cases where the relation between $\mathbf{w}$ and $\mathbf{g}$ may be a non-linear function $f(\mathbf{w})$ mapping the inputs to the outputs can also be defined [4–6]. In mathematical physics, the formulation of the *forward problem* for a physical field involves [7]: (1) The domain in which the field is studied. (2) The equation function that describes the field. (3) The initial conditions. (4) The conditions on the boundary of the domain. The inverse problem consists of finding functions used for the forward problem, for example, the unknown $f(\mathbf{w})$ in Eq. (1.1). Given $\mathbf{g}$ and $\mathbf{\Phi}$, the calculation of $\mathbf{w}$ is referred to as solving an *inverse problem*. This inverse problem is historically a long-standing issue and various approaches have been invented and developed to deal with this problem in past hundreds of years. The *least-squares method* [8] was a common early approach for solving an inverse problem like the one associated with Eq. (1.1).

1.1.1 Classic approaches for solving inverse problems

Least-squares method   Over 200 years ago, the method of least squares was first published by Adrien-Marie Legendre in 1805 [9, 10]. It is usually also credited to Carl Friedrich Gauss [11] because in 1809 Carl Friedrich Gauss [12] published the method of calculating

the orbits of celestial bodies by using the least-squares method and he claimed to have been in possession of the method since 1795. The simple least-squares fit minimizes the sum of the squares of the difference between the observed data g_i and the fitting function evaluated at w_i: $\sum_{i=1}^{N}(g_i - f(w_j))^2$. If each standard deviation σ_i for the observed data g_i is given, the least-squares method can also be written as the minimization of $\sum_{i=1}^{N}((g_i - f(w_j))/\sigma_i)^2$, which is called a χ^2 fit. This χ^2 fit was first described by the German statistician Friedrich Robert Helmert in 1876 [13, 14] and was independently rediscovered by the English mathematician Karl Pearson in 1900 [15]. The main disadvantages of this simple least-squares method are the requirement of small deviations between data and model, which allows one to assume a Gaussian distribution and causes sensitivity to outliers.

In 1809 [12], Gauss extended the least-squares method through a probabilistic perspective. He combined the Lambert-Bernoulli idea [16, 17] with Laplace's analytical formulation of inverse probability [18, 19]. Requiring that the mode of what is nowadays called the posterior probability distribution equals the arithmetic mean, Gauss derived the normal distribution [12, 20]. From the least-squares method to modern methods for solving complex inverse problems, normally distributed errors appear in many statistical approaches.

Maximum likelihood estimation   Before 1912, *maximum likelihood estimation* (MLE) occurred in rudimentary forms, but not under this name. Some of the estimates called "most probable" would today have been called "most likely" [20]. In 20th century statistics, the making of maximum likelihood was one of the most important developments [21, 22]. In 1912, the British statistician and geneticist R. A. Fisher [23] started producing one of the earliest contribution to modern statistics by using a simple maximum likelihood method ("absolute criterion") to estimate unknown parameters. R. A. Fisher introduced the term "likelihood" in 1921 [24] and the name "maximum likelihood estimate" finally appeared in the article "On the mathematical foundations of theoretical statistics" in 1922 [25]. Let $\mathbf{w}$ be an unobserved data set and $\mathbf{g}$ be a set of observed data. The conditional probability of $\mathbf{g}$, given $\mathbf{w}$, $p(\mathbf{g}|\mathbf{w})$ is called *likelihood*. MLE is to maximize the likelihood function $L(\mathbf{w})$ with respect to the unobserved data $\mathbf{w}$. For example, consider the maximum likelihood estimation applied for the inverse problem in Eq. (1.1) [26]. The errors $\mathbf{s}$ are assumed to be values of a random variable $\mathcal{S}$ which follows a Gaussian distribution $\mathcal{S} \sim \mathcal{N}(0, 1/\beta)$, where the variance is $1/\beta$. Thus, the likelihood is written as $p(\mathbf{g}|\mathbf{w}) = (2\pi/\beta)^{-N/2} \exp\left(-\beta(\mathbf{g} - \mathbf{\Phi w})^2/2\right)$. To obtain $\mathbf{w}$, one maximizes the log-likelihood function which is denoted as $l(\mathbf{w}) = -N/2\log(2\pi) + N/2\log\beta - \beta/2\left((\mathbf{g} - \mathbf{\Phi w})^{\mathbf{T}}(\mathbf{g} - \mathbf{\Phi w})\right)$. Here, the maximum likelihood estimation is equivalent to the east-squares method for minimizing $\sum_{i=1}^{N}(g_i - \Phi_{ij}w_j)^2$.

Between 1912 and 1922, R. A. Fisher had produced three justifications and three names for the technique of MLE [20, 22]. Although R. A. Fisher is certainly the father of maximum likelihood analysis, his attempts to prove the procedure remained largely fruitless [27]. The solid theoretical basis for the maximum likelihood estimation procedure was laid by Samuel S. Wilks in 1938 [27], now also called Wilks' theorem. This theorem states that the error in the logarithm of likelihood values for estimates from multiple independent observations is asymptotically χ^2-distributed, which enables convenient determination of a confidence region around any estimate of the parameters.

Bayesian methodology   The foundations of Bayesian probability theory were posthumously published in Thomas Bayes' article "An Essay Towards Solving a Problem in the Doctrine of Chances " in 1764 [28]. Later, the french mathematician, Pierre Simon Laplace independently

rediscovered the Bayes' modern mathematical form in the article "Mémoire sur la probabilité des causes par les événements" in 1774 [18] and the later article "Théorie analytique des probabilités" in 1812 [29]. Like in MLE, $\mathbf{w}$ is an unobserved data set and $\mathbf{g}$ is an observed data set. The conditional probability $p(\mathbf{w}|\mathbf{g})$ expresses the probability of finding the sought-for quantities $\mathbf{w}$, given the observed quantities $\mathbf{g}$. The *posterior* probability distribution $p(\mathbf{w}|\mathbf{g})$ is typically not known. Frequently, however, the reverse conditional probability distribution $p(\mathbf{g}|\mathbf{w})$ is known either based on generic assumptions regarding the experimental system or because one is dealing with numerical simulations. The posterior probability and the likelihood can be related to each other by making use of the unconditional probabilities $p(\mathbf{g})$ and $p(\mathbf{w})$. The standard modern-days Bayes' rule was first given by Laplace as [30]

$$p(\mathbf{w}|\mathbf{g}) = \frac{p(\mathbf{w}, \mathbf{g})}{p(\mathbf{g})} = \frac{p(\mathbf{g}|\mathbf{w})p(\mathbf{w})}{p(\mathbf{g})} \propto p(\mathbf{g}|\mathbf{w})p(\mathbf{w}). \tag{1.2}$$

Here, $p(\mathbf{w}, \mathbf{g})$ is the joint probability distribution. Since $p(\mathbf{w})$ can reflect a prior information about the sought-for quantity $\mathbf{w}$, it is called "*prior*". In contrast, $p(\mathbf{g})$ is the unconditional probability distribution for observing the values $\mathbf{g}$ and is therefore called "*evidence*" or "*marginal likelihood*". In Eq. (1.2), the posterior distribution is proportional to likelihood multiplied by prior. The *maximum a posterior* (MAP) estimate of $\mathbf{w}$ is defined as that value that maximizes the likelihood multiplied by a prior. Thus, MLE is a special case of the MAP estimation, where a uniform prior $p(\mathbf{w}) = $ const. is assumed in Eq. (1.2). However, the Bayesian methodology provides much broader flexibility since one can choose priors $p(\mathbf{w})$ together with the likelihood function.

1.1.2 Challenges and modern approaches for solving inverse and ill-posed problems

We have introduced the classic approaches to solve the inverse problem in Eq. (1.1). One common challenge is that the inverse problem is often a so-called *ill-posed problem*. The study of inverse and ill-posed problems began in the early 20th century [7]. In 1902, J. Hadamard proposed the concept of well-posedness of problems for differential equations [31] and he termed a problem well-posed if there exists a unique, robust solution to this problem. He also gave an example of an ill-posed problem, namely, the Cauchy problem for the Laplace equation, where the solution does not depend continuously on the data and any small change in the data causes large changes to the solution [32–34]. The challenge of solving ill-posed problems occurs in almost all fields of science, particular examples are image reconstruction [35, 36], traction force reconstruction [37, 38], machine learning [39, 40], seismic exploration [41], tomography [42], astronomy [43], and air and water quality control [44].

One common important property of ill-posed linear problems is the strong sensitivity of the solutions to small perturbations in the equation parameters. In linear regression Eq. (1.1), the well-posedness of the solution depends on the matrix $\mathbf{\Phi}$ and it can be analyzed by using a perturbation approach [7, 45]. Introducing the perturbations $\delta\mathbf{g}$ and $\delta\mathbf{w}$, Eq. (1.1) becomes $\mathbf{g} + \delta\mathbf{g} = \mathbf{\Phi}(\mathbf{w} + \delta\mathbf{w})$. We can also write $\delta\mathbf{g} = \mathbf{\Phi}\delta\mathbf{w}$, which implies $\delta\mathbf{w} = \mathbf{\Phi}^{-1}\delta\mathbf{g}$ and $\|\delta\mathbf{w}\|_2 \leq \|\mathbf{\Phi}^{-1}\|_2 \|\delta\mathbf{g}\|_2$. The unperturbed Eq. (1.1) yields $\|\mathbf{g}\|_2 \leq \|\mathbf{\Phi}\|_2 \|\mathbf{w}\|_2$. Thus, the estimate for the relative error of the solution becomes $\|\delta\mathbf{w}\|_2/\|\mathbf{w}\|_2 \leq \|\mathbf{\Phi}\|_2 \|\mathbf{\Phi}^{-1}\|_2 \|\delta\mathbf{g}\|_2/\|\mathbf{g}\|_2$, which shows

that the error is determined by the constant

$$\kappa(\mathbf{\Phi}) = \|\mathbf{\Phi}\|_2 \|\mathbf{\Phi^{-1}}\|_2, \tag{1.3}$$

where $\kappa(\mathbf{\Phi})$ is called the *condition number* of a system. A system with a *large* condition number is said to be ill-posed because small variations in the input $\delta \mathbf{g}$ may lead to relatively large variations in the solution. Variations (errors) of input data always exist in measurements or simulations. Solving ill-posed problems involving such input data is challenging with classical approaches, such as those from least-squares methodology. However, inverse and ill-posed problems can often be solved by imposing an additional regularization constraint when performing a least-squares optimization. A general form of this regularization procecdure when extending the linear regression Eq. (1.1) is written as

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}}\big[(\mathbf{g} - \mathbf{\Phi}\mathbf{w})^2 + \lambda H(\mathbf{w})\big], \tag{1.4}$$

where $\hat{\mathbf{w}}$ is the solution of inverse and ill-posed problem and $\lambda H(\mathbf{w})$ represents the regularization constraint with an unknown parameter $\lambda > 0$. Various regularization approaches have been developed in the past 80 years.

1.1.2.1   Regularization approaches

In 1943 [46], A. N. Tikhonov pointed out the practical importance of ill-posed problems and the possibility of finding stable solutions to them. The nowadays standard approach to solve ill-posed problems is called ridge regression and was published by David L. Phillips [47] in 1962 and A. N. Tikhonov in 1977 [32]. The ridge regression or Tikhonov regularization relies on the constraint $\lambda H = \lambda_2 \|\mathbf{w}\|_2^2$ involving the 2-norm and therefore it also called L2 regularization. The solutions of L2 regularization are typically smooth and non-sparse and the computation is efficient because analytical expressions can be derived for the L2 regularization.

In 1986, geophysicists observed that the constraint $\lambda H = \lambda_1 \|\mathbf{w}\|_1$ can be successfully applied to compute a sparse reflection function indicating changes between subsurface layers [48]. In 1996, Robert Tibshirani [49] greatly popularized the use of L1-norm and related greedy methods in statistics, called the Lasso, or Lasso regression, or L1 regularization. The main property of L1 regularization is the sparsity of solutions, which means that the solution contains only few non-zero components.

In 1994, the method of L0-norm constraint $\lambda H = \lambda_0 \|\mathbf{w}\|_0$ was suggested by D. P. Foster and E. I. George [50] and today this regularization method is called L0 regularization or best subset selection [51]. The solutions from L0 regularization are for certain sparse models more accurate than the solutions obtained from L1 regularization [52]. However, computational challenges related to L0 regularization result from the discontinuity and nonconvexity of the L0-penalty function. This issue is usually dealt with by replacing the L0-penalty function with a continuous or convex approximation function [53, 54].

In 2005, Zou and Hastie [55] introduced the elastic net regularization which combines the L2- with the L1-norm constraint $\lambda H = \lambda_2 \|\mathbf{w}\|_2^2 + \lambda_1 \|\mathbf{w}\|_1$. This approach balances smoothness and sparsity of the solutions from ridge- and Lasso regression, respectively. Recently, the elastic net regularization is widely employed in various scientific applications such as

studying anticancer drug sensitivity [56, 57], gene expression [58], structure and function of ocean microbiome [59], and face recognition [60].

Here, we have briefly introduced five regularization approaches to deal with inverse and ill-posed problems. When one uses these approaches, the challenge is to identify the optimal regularization parameter λ, which is a priori unknown. A number of methods exist for selecting the regularization parameters, such as the L-curve [61], generalized cross-validation [62] or quasi-optimality criterion [63]. However, these methods mostly rely on heuristics and manual selection of the regularization parameter is usually necessary. Those problems can be overcome by using Bayesian analysis.

1.1.2.2 Bayesian regularization approaches

The heuristic regularization approaches discussed above can be related to the concept of maximizing the posterior probability in a Bayesian framework. To illustrate this connection, we assume, as for the maximum likelihood estimator of the linear regression Eq. (1.1), a noise $\mathbf{s}$ that obeys a Gaussian distribution $s_i \sim \mathcal{N}(0, 1/\beta)$. Here, the prior is also assumed to be a Gaussian $w_i \sim \mathcal{N}(0, 1/\alpha)$, where the variance is $1/\alpha$. According to Bayes' rule in Eq. (1.2), the posterior distribution is written as $p(\mathbf{w}|\mathbf{g}) = (2\pi/\beta)^{-N/2}(2\pi/\alpha)^{-M/2} \exp\big(-\beta(\mathbf{g} - \mathbf{\Phi}\mathbf{w})^2/2\big) \exp\big(-\alpha\mathbf{w}^2/2\big) \propto \exp\big(-\beta(\mathbf{g} - \mathbf{\Phi}\mathbf{w})^2/2 - \alpha\mathbf{w}^2/2\big)$. To obtain $\mathbf{w}$, we maximize the posterior $\hat{\mathbf{w}} = \underset{\mathbf{w}}{\mathrm{argmax}}[-\beta(\mathbf{g} - \mathbf{\Phi}\mathbf{w})^2/2 - \alpha\mathbf{w}^2/2]$, where the form is the same as the L2 regularization $\hat{\mathbf{w}} = \underset{\mathbf{w}}{\mathrm{argmin}}[(\mathbf{g} - \mathbf{\Phi}\mathbf{w})^2 + \lambda_2\mathbf{w}^2]$ with $\lambda_2 = \alpha/\beta$. The constraint λH in each of the regularization approaches that were mentioned above can be related to a particular prior function, for example, here, L2 regularization yields a Gaussian prior. It is not difficult to find that the L1-, L0-, and elastic net regularization respectively, are equivalent to a MAP estimation employing a Laplace prior $p(\mathbf{w}) \propto \exp(-\lambda_1\|\mathbf{w}\|_1)$ [64], L0-regularized prior $p(\mathbf{w}) \propto \exp(-\lambda_0\|\mathbf{w}\|_0)$ [65], and Elastic net prior $p(\mathbf{w}) \propto \exp(-\lambda_1\|\mathbf{w}\|_1 - \lambda_1\|\mathbf{w}\|_2^2)$ [66].

While these results connect the regularization approaches to the Bayesian MAP estimates, the challenge of selection of regularization parameter remains when using only the MAP approach. However, this challenge can be overcome in the Bayesian framework.

Bayesian regularization   *Bayesian regularization* [67, 68] involves two levels of inference: (1) Choose models and fit data (model fitting). (2) Rank the alternative models (model comparison). In the first level of inference, the particular model is assumed and fitted to the data, which can be done with a MAP estimation procedure. In the second level of inference, the evidence for the model parameters is calculated, for example, by using the marginal likelihood which is given in the Bayes' rule of Eq. (1.2). This process embodies the colloquial "*Occam's razor*", which ensures that overly complex models will not be preferred over simpler models unless the data supports them. The details of Bayesian regularization employing a Gaussian prior will be laid out in Chapter 2.

Sparse Bayesian learning   In 2001, M. E. Tipping [69] demonstrated a probabilistic Bayesian learning framework where the solution is assumed to consist of only few basis functions, which is therefore called relevance vector machine (RVM). This work is an extension of Bayesian regularization techniques because it also utilizes the maximum evidence principle estimate the parameters or hyperparameters of likelihood and prior. In 2003, M. E. Tipping and A. C. Faul [70] described a highly accelerated algorithm which exploited the

evidence function [67] to enable maximization via a principled and efficient sequential addition and deletion of candidate basis functions. The details of this approach will be laid out in Chapter 2. *Sparse Bayesian learning approaches* were successfully applied to reduce the dimensionality of sensor data. This technique is now called Bayesian compressive sensing [71, 72]. These Bayesian sparse learning approaches require Markov chain Monte Carlo (MCMC) methods [73–77] or a more efficient variational Bayesian (VB) analysis [67, 78].

All the above approaches to solve inverse problems with an unknown vector $\mathbf{w}$, rely on the provision of a matrix $\mathbf{\Phi}$ and a (measurement) data vector $\mathbf{g}$. However, in some cases the matrix $\mathbf{\Phi}$ may be unknown. For example, in classification tasks one may have a set of training data for the relationship between the classifier $\hat{\mathbf{g}}$ and the feature vector $\mathbf{g}$, but the system matrix $\mathbf{X}$ and the noise $\mathbf{b}$ are unknown in $\hat{\mathbf{g}} = k(\mathbf{g}) = \mathbf{X}\mathbf{g} + \mathbf{b}$. In such cases, one can use neural networks and methods from deep learning to obtain the unknown system parameters.
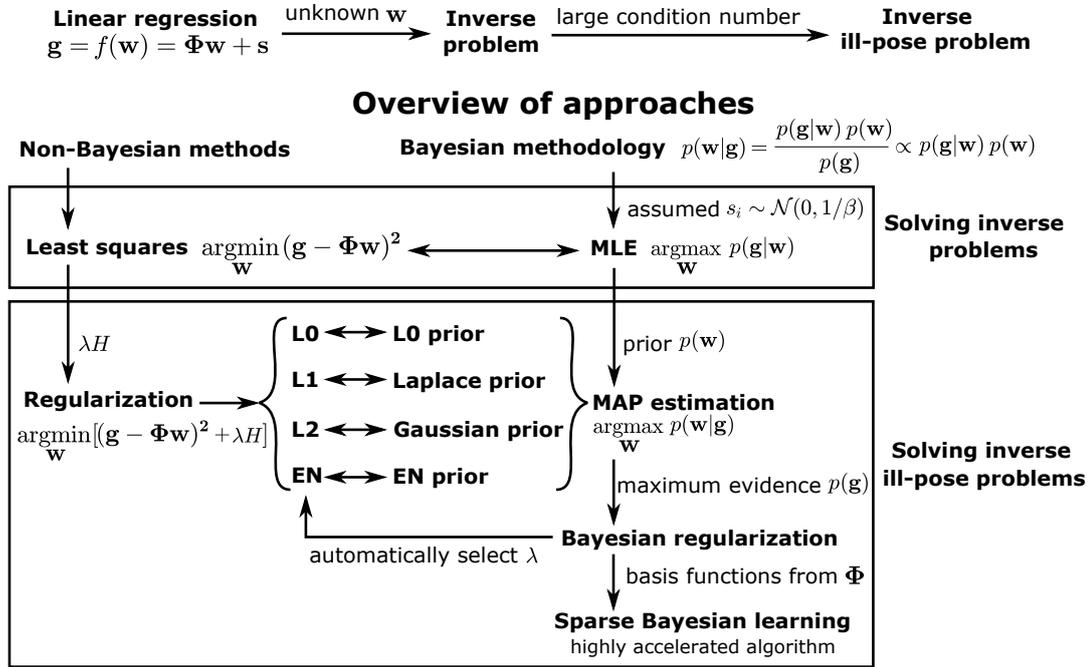
Figure 1.1: Schematic overview of Non-Bayesian and Bayesian methods for solving linear inverse problems.

1.1.3   Deep learning for analysis of noisy data

Deep learning can be traced back to 1943 [79], when W. S. McCulloch and W. H. Pitts created a computer model based on the neural network of the human brain. Recently, thousands of deep learning methods have been developed that involve, for example, backpropagation algorithm, convolutional neural networks, recurrent neural networks, recursive neural networks, autoencoders, and graph neural networks. Here, we utilize the example of noise reduction to explain the central idea of supervised neural networks [80, 81]. Given an input vector $\mathbf{t}$, e.g., a space or time vector, and a target noisy data $\mathbf{g}$, we train a neural network with the given relation between $\mathbf{t}$ and $\mathbf{g}$. A successfully trained network will, for any given $\mathbf{t}$, produce an output $\hat{\mathbf{g}}$ that is close to the target vector $\mathbf{g}$. To demonstrate this concept, a two-layer neural network [80] is employed. The hidden layer is written as $\zeta^{\mathbf{X}_1, \mathbf{b}_1}(\mathbf{t}) = \zeta(\mathbf{X}_1\mathbf{t} + \mathbf{b}_1)$, where ζ is an activation function, e.g., $\zeta(\cdot) = \tanh(\cdot)$. $\mathbf{X}_1$ and $\mathbf{b}_1$ are respectively called

weights and biases. The output layer is written as $\hat{\mathbf{g}} = \boldsymbol{\zeta}^{\mathbf{X_1}, \mathbf{b_1}}(\mathbf{t})\mathbf{X_2} + \mathbf{b_2}$. These unknown weights and biases $\mathbf{X_1}$, $\mathbf{b_1}$, $\mathbf{X_2}$ and $\mathbf{b_2}$ of each layer are learned by the minimization of a cost function, which is the sum of the deviation between the output and the target. Using such a neural network, the noise in the inferred output $\hat{\mathbf{g}}$ typically can be significantly reduced [82].

In Figure 1.1, we present an overview of methods for solving inverse problems. The Bayesian methodology has the advantage of allowing one to prescribe clearly defined assumptions in the form of priors which then automatically produce robust solutions. However, many approaches for solving inverse problems in physics and biophysics still exclusively rely on non-Bayesian methods. In this thesis, we systematically test the classic approaches and develop new methods to solve inverse ill-posed problems in two applications, namely *traction force microscopy* and *data-driven discovery of physical equations*. To prepare the stage, we will in the following sections introduce background information on these two applications.

1.2   Background information on measurement of cellular forces

The mechanical behaviour of cells and tissues plays a crucial role in a variety of biological, biophysical and biochemical processes, including cell migration [83, 84], tissue morphogenesis [85, 86], wound healing [87, 88], cell differentiation [89, 90], and gene expression [91, 92]. Many of the relevant mechanobiological processes occur on a subcellular lengthscale, for example at micrometer-sized cellular adhesions and at filopodia. However, mechanobiology also plays a critical role on the level of multiple cells and on lengthscales of millimeters, for example in embryonic development [93, 94]. The mechanical behaviours of cells are not only controlled by biochemical reactions inside of cells, but also depend on the mechanical properties of *extracellular matrix* (ECM). To understand the interplay of extracellular and intracellular mechanics and biological regulation, a reliable and accurate method for the measurement of cellular forces is required. Over the past 50 years, various approaches have been developed for this purpose. In Ref. [95], different kinds of methods were systematically summarized. Various approaches and tools for measuring the forces generated on ECM were also reviewed in Ref. [96].

In this section, some background information on the mechanobiology of cells is introduced. First, we introduce the mechanical structures: the cytoskeleton, transmembrane, and extracellular structures. Then, we introduce different mechanical models for the structures. Finally, we summarize several approaches and tools to measure the mechanical forces.

1.2.1   Mechanical structures of cell

1.2.1.1   Cytoskeleton

The *cytoskeleton* (CSK) is a polymeric fiber-based scaffold for structural integrity inside cells. These scaffolds not only serve as a traffic system for intracellular transport [97], for example, helping to ship vesicles or organelles, but also are important mechanical structures for cells migration. Usually, the CSK consists of at least three distinct filamentous elements: microtubules, actin-CSK, and intermediate filaments, illustrated in Fig. 1.2 (b). Here, we briefly introduce each of these structures and their mechanical properties.

- *Microtubules* have a length up to roughly $50\,\mu$m and are composed of α- and β-tubulin. The outer diameter of a microtubule is about $25\,$nm and the inner diameter is about
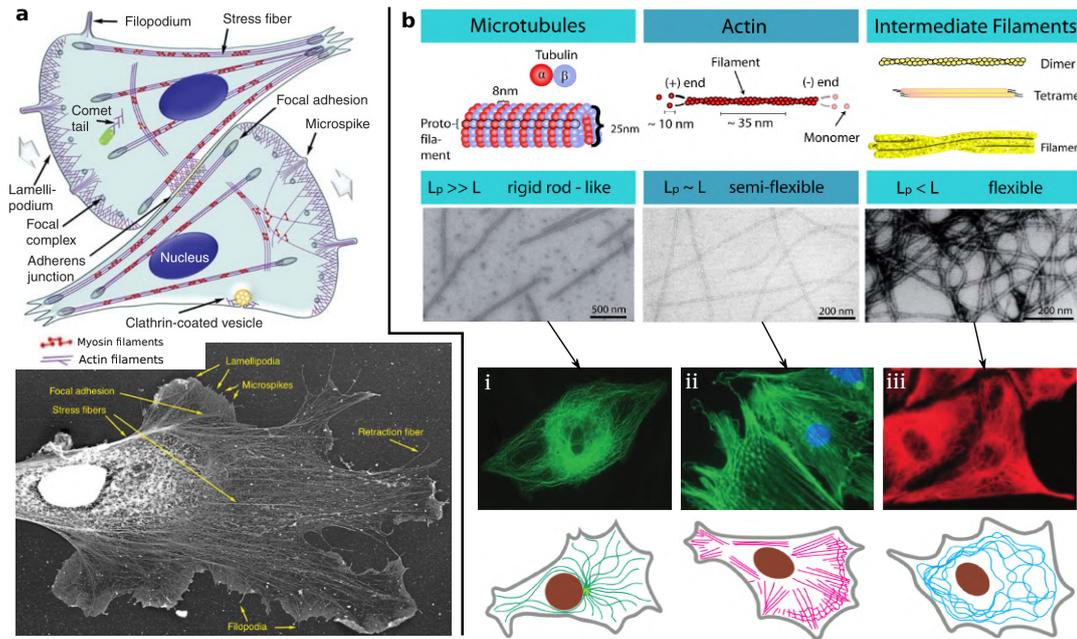
Figure 1.2: Structures of the cytoskeleton in cells. (a-Top) Illustration of the components of the actin cytoskeleton in representative fibroblast-like cells. The direction of migration is denoted by the wide gray arrows. (a-Bottom) Electron micrograph of the cytoskeleton of a *Xenopus laevis* fibroblast. (b-Top) The cytoskeleton (CSK) has the three types of biopolymers: microtubules, actin and intermediate filaments. The three types of CSK pertain to different stiffness regimes because of their differing filament architectures.
(b-Middle) Fluorescence micrographs for each type of CSKs in real cells.
(b-Bottom) Schematic graphs show that each type of CSKs is how to distribute inside cells. (Figure (a) adapted from T. Svitkina 2018 [98]. Figure (b) adapted from F. Huber, 2011 [99] with source material from D. E. Ingber, 2003 [100] and J. R. D. Soiné, 2014 [101].)

13 nm. They are very rigid polymer tubes that usually emerge as individual fibers, which are typically associated with organelle positioning and intracellular transport.

- *Actin-CSKs* are composed of monomeric G-actin and the linear polymeric F-actin with a diameter from 4 nm to 7 nm [102]. Actin filaments are semi-flexible polymers appearing as cross-linked networks within cells, illustrated in Fig. 1.2 (b-ii) [100]. The actin CSK is a dynamical structure that appears in different forms, including compact stress fibers and finely crosslinked nets.

- *Intermediate filaments* have a diameter of roughly 50 nm, which is between the size of actin filaments and microtubules, illustrated in Fig. 1.2 (b-iii) [100]. Intermediate filaments are composed of a family proteins including desmin, keratins, and lamins.

The CSK is the essential structure creating motility-driving forces. These forces are generated by polymerization and the interaction of the F-actin network with myosin motors. The former, the polymerization of actin filaments at the leading edge of cells, can move the edge forward [103] and the force generation in this process can be explained by two models, ratchet models [104, 105] and autocatalytic models [106, 107]. The latter, the myosin motors move on actin filaments through a usual three-step process of binding, power-stroke, and unbinding [108], which generates contractile forces. The force generated from each of the motors is extremely tiny. For example, to lift a 5 kg weight, about 10^{13} myosin motors

are required [108]. These two motility-driving mechanisms not only result in motion and deformation of the cytoskeleton and the cell, but also produce a mechanical load on the extracellular structures.

1.2.1.2 Transmembrane and extracellular structures

The mechanical connection between the intracellular CSK and the extracellular world is maintained by dedicated molecular structures. Figure 1.3 (a) shows a schematic diagram of different types of cell-cell adhesions, which include gap junctions, tight junctions, adherens junctions, and desmosomes [109]. These cell-cell adhesions not only can transfer mechanical forces from one cell to another cell, but also provide an aisle for different molecules between two cells, for example, ions and electrical impulses. Here, we briefly summarize the structures and their properties.

- *Gap junction* are intercellular connections with a variety of transport functions and, for instance, allow the passage of small molecules or electrical signals.

- *Tight junctions*, also called occluding junctions, seal the space between neighboring cells and can control the passage of ions and small moclecules.

- *Adherens junctions* provide the strong mechanical attachments between cells. Adherens junctions can contain nectin-afadin or E-cadherin-a-catenin-vinculin bonds, illustrated in Fig. 1.3 (a) and they are linked to the actin-CSK inside the cell.

- *Desmosomes* are localized patches, which can hold two adjacent cells closely together [110]. Desmosomes are attached to intermediate filaments of keratin in the cytoplasm, illustrated in Fig. 1.3 (a).

Cell-matrix connections transmit stresses to the ECM and are important in many physiological processes such as cell migration, proliferation, and differentiation. Cell-matrix connections can be classified into three different groups, as illustrated in Fig. 1.3 (b).

- *Nascent adhesions* (NAs) locate at the edge of a cell protrusion by nucleating three to six transmembrane proteins called integrins. NAs are dynamically coupled to the polymerizing branched actin network.

- *Focal adhesions* (FAs) mature from a small number of nascent adhesions and these stable FAs are important for regulating cell adhesion and motility. FAs typically can be highly dynamic assemblies and thus the cell can change its shape and persistently migrate through ECM [111, 112].

- *Fibrillar adhesions* (FBs) are developed from the maturing focal adhesions by growing their size and changing their protein composition [113, 114]. Fibrillar adhesions are typically large adhesions in protrusions and the cell body and represent the endpoint in terms of adhesion maturation [115].

Various types of ECM can serve as substrate for cell adhesion [119]. In *vivo*, the native ECM of one cell can be other cells, tissues or organs, illustrated in Fig. 1.3 (c-i) [116]. Recently, to study properties and behaviors of cells, many artificial ECMs have been designed and fabricated, for example, beds of microneedles to isolate mechanical force [117] and polyacrylamide hydrogels to study the traction force [118], illustrated in Fig. 1.3 (c-ii, c-iii), respectively.
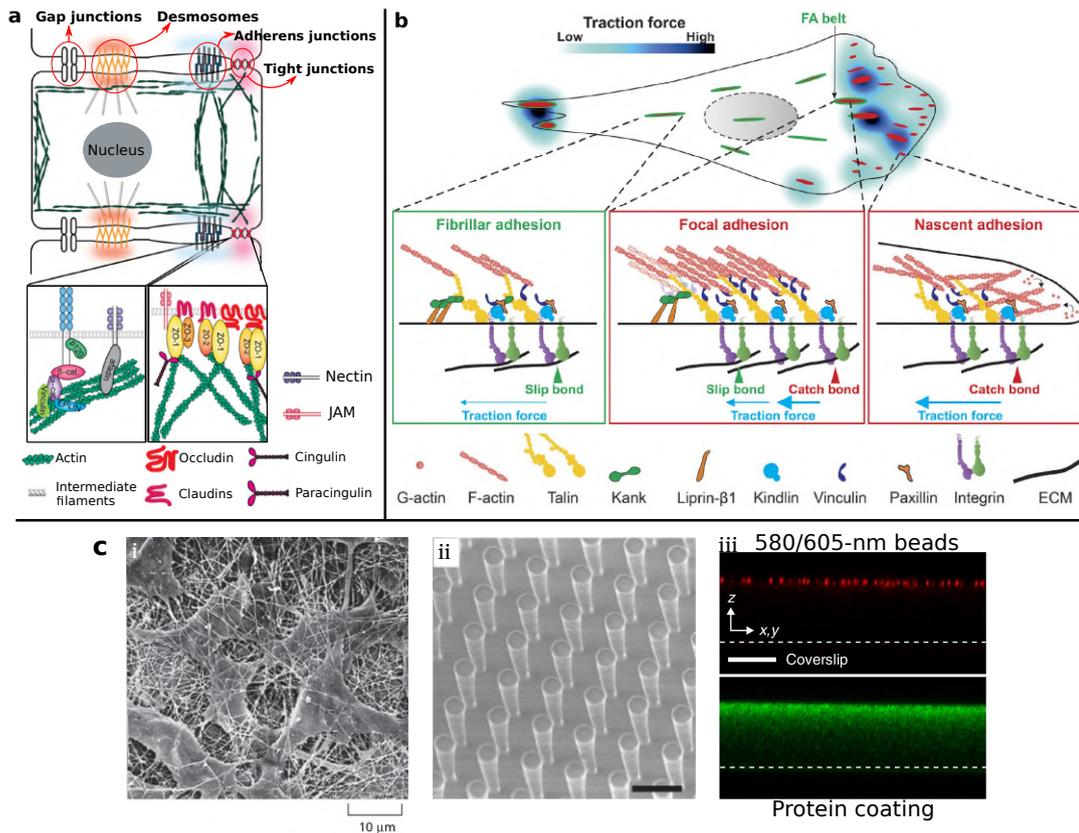
Figure 1.3: Extracellular structures. (a) Four types of cell-cell conjunctions, gap junctions, tight junctions, adherens junctions, and desmosomes. (b) Three different types of cell-matrix conjunctions are classified into nascent adhesions, focal adhesions, and fibrillar adhesions according to the different levels of traction forces. Each of cell-cell and cell-ECM conjunctions is connected to the cytoskeleton inside of cells and thus the conjunction can transfer forces from cytoskeleton to ECM. (c) Extracellular native and artificial matrix. (c-i) Scanning electron micrograph demonstrates a native ECM, a tissue from the cornea of a rat. (c-ii) Artificial beds of microneedles to isolate mechanical force and collagen fibers matrix to learn cell migration. Space bar: $10\,\mu m$. (c-iii) Artificial substrates to be used for traction force microscopy. The beads and protein are put on the top of substrates. Space bar: $30\,\mu m$. (Figure (a) adapted from S. Sluysmans et al., 2017 [114]. Figure (b) adapted from Z. Sun et al., 2016 [109]. Figure (c-i) taken from B. Alberts et al., 2002 [116]. Figure (c-ii) taken from J. L. Tan, et al., 2003 [117]. Figure (c-iii) adapted from H. Colin-York et al., 2017 [118].)

A general principle for measuring the cellular forces is that a displacement field generated by the forces in the extracellular environment is first measured by using a microscopy technique and then the forces are calculated from the displacement field by using a given mechanical model. In the next section, we briefly introduce the imaging methods and mechanical models used in this context.

1.2.2 Mechanical models and imaging techniques

The CSK and extracellular structures typically are solid materials. These solid materials can be deformed by the motility-driving forces on CSK or the external forces applied on them. The relationship between forces and deformation is called a material constitutive equation or a mechanical model. For solid elastic materials, this relationship can be defined by relating

the stress σ, the force per unit area, and strain ϵ, the fractional change in the length of a material.
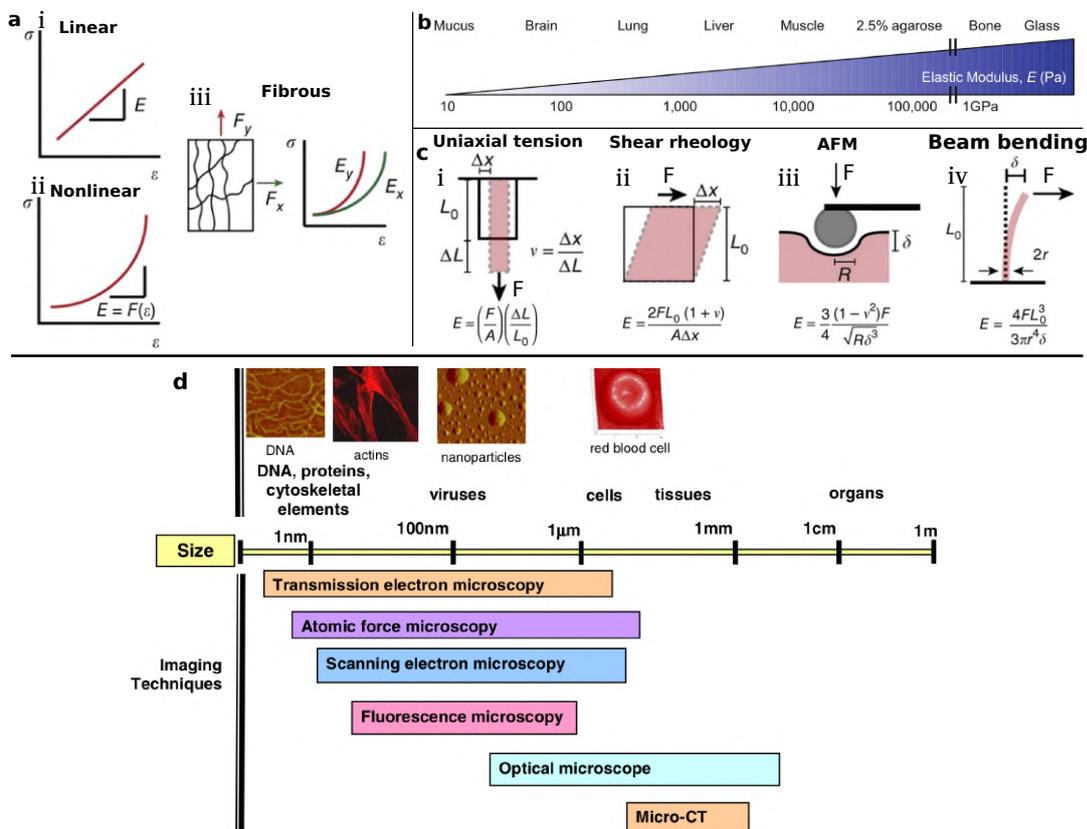
Figure 1.4: Mechanical models and imaging techniques. (a) Mechanical models relating the force and displacement. (a-i) Simple relationship between stress and strain in linear materials with a constant Young's modulus. (a-ii) The non-linear model can be also employed in measurement of cellular forces. (a-iii) The complex ECM can be described by an anisotropic and nonlinear model. (b) Young's modulus (E) represents the stiffness of solid materials, with units of Pa. Different types of cells have different Young's moduli. (c) Under different types of loading, the equations of Young's modulus can be written in different forms, for example, the forms of uniaxial tension, shear rheology, loading from atomic force microscopy (AFM), and beam bending shown in (c-i to c-iv). (d) Imaging techniques for conducting biomechanical tests in different length scales. Imaging techniques, transmission electron microscopy, atomic force microscopy, scanning electron microscopy, fluorescence microscopy, optical microscope, and micro-CT are employed from the biomolecules to organs. (Figure (a, c) adapted from J. M. Barnes et al., 2017 [96]. Figure (b) taken from W. J. Polacheck et al., 2016 [120]. Figure (d) taken from C. T. Lim et al., 2006 [121].)

The simplest mechanical model is linear elasticity. Here, stress is linearly related to the strain, illustrated in Fig. 1.4 (a-i), and the constant coefficient between the stress and strain is called Young's modulus E. The different types of materials have different Young's moduli, illustrated in Fig. 1.4 (b). Typical values are, for example, a Young's modulus of $E \approx 10\,\mathrm{Pa}$ for mucus and $E \approx 1\,\mathrm{GPa}$ for bone. For an elastic material, the equation allowing one to determine the Young's modulus depend on the types of loading, for example, uniaxial tension, shear loading, loading from atomic force microscopy (AFM), and beam bending illustrated in Fig. 1.4 (c-i to c-iv), respectively. The mechanical model can also be a non-linear relationship, illustrated in Fig. 1.4 (a-ii). The usual non-linear models include plasticity

and viscoplasticity. To model the complex structure of the ECM, one can also employ an anisotropic model, in which the anisotropy depends on the direction, illustrated in Fig. 1.4 (a-iii).

We have introduced mechanical models for the CSK and extracellular structures. To calculate the forces that cells exert on these structures, we also need to measure the material displacements. Due to the fact that these structures are typically microscopically small, the displacement needs to be obtained by using a microscope or related imaging techniques. Here, we briefly introduce several techniques to take images at different length scales, illustrated in Fig. 1.4 (d) [121] and the techniques include transmission electron microscopy (TEM), atomic force microscopy (AFM), scanning electron microscope (SEM), optical microscopy (includes fluorescence microscopy), and computer tomography. For example, to visualize the biomolecules of the cytoskeleton with sizes in the nm range AFM, TEM, and fluorescence microscopy are required. Cells with sizes in the μm range can be visualized with AFM, TEM, SEM, optical microscopy, and fluorescence microscopy. Fluorescence microscopy utilizes the characteristic emissions of excited fluorophores, for example, fluorescent proteins [122] and this microscopy technique has been widely employed to study cellar forces, for instance with traction force microscopy [118]. In the following section, we will introduce several special approaches to measure such cell-generated forces.

1.2.3 Measurement methods of cell-generated forces

Over the past 50 years, various methods have been developed to measure the cell-generated forces. Here, we briefly introduce methods for measuring forces on a single cell and methods for measuring forces at cell-ECM connections.

1.2.3.1 Methods for measuring the mechanical behavior of a single cell

To study the mechanical response of a single cell, the cell is often subjected to an external loading. Under this load, the deformation of the cell can be measured and forces can be calculated by using a constitutive law. According to the different types mechanical devices, the methods [123, 124] can be divided into the following categories:

- *Atomic force microscopy* (AFM) can be used to locally probe the mechanical response of cells. A schematic view of AFM is shown in Fig. 1.5 (a) and the minute displacement of the cantilever can be measured by using a high-resolution scanning probe microscopy. AFM not only can measure the mechanical response of the cell, but also can provide a three-dimensional surface profile [127].

- *Micropipette aspiration* is a technique to deform one cell by using a pipette in a solution. The cell is partially aspirated into a glass pipette, illustrated in Fig. 1.5 (b), and the forces on the cell can be calculated by using a model related to the suction pressure [128].

- *Optical tweezers* utilize a highly focused laser beam acting on colloidal particles to provide a force on cells, illustrated in Fig. 1.5 (c). These instruments have been widely employed to measure the forces on molecules and cells [129, 130].

- The *optical stretcher* is a contact-free measurement tool. A dual-beam optical trap is employed to deform the cells, which are in a flow channel, illustrated in Fig. 1.5 (d).
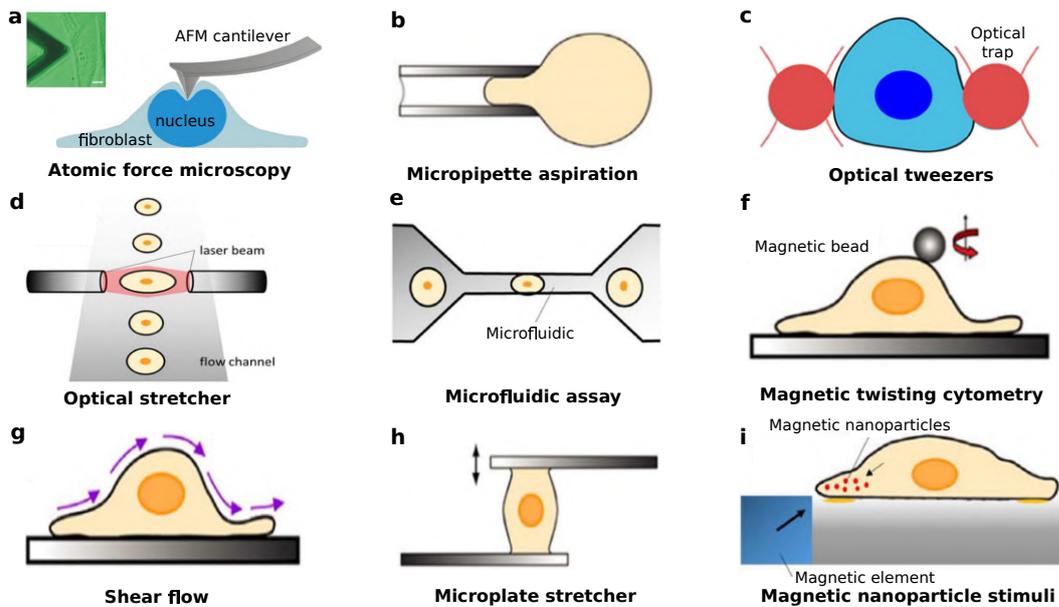
Figure 1.5: Schematic different methods for measuring forces on a single cell. (a) Atomic force microscopy (AFM). (b) Micropipette aspiration. (c) Optical tweezers. (d) Optical stretcher. (e) Microfluidic assay. (f) Magnetic twisting cytometry. (g) Shear flow. (h) Microplate stretcher. (i) Magnetic nanoparticle-based stimuli. (Figure adapted from M. Unal et al., 2012 [95] with source material from S. Suresh, 2007 [123]; G Bao et al., 2003 [124]; H. Milting et al., 2014 [125] and F. D. Modugno et al., 2019 [126].)

- The *microfluidic assay* is a technique to analyze the mechanics of a cell by using fluid pressure. When fluid is flown one tube into a constriction, the pressure is increased and the cells can be deformed by the fluid.

- *Magnetic twisting cytometry* uses ferromagnetic microbeads to apply a twisting shear stress on cell surface receptors [131].

- Application of a *shear flow* to a cell that is adhered to a substrate can produce a deformation of the cell. The mechanical properties of the cell are studied by using a mechanical model.

- The *microplate stretcher* is a technique to stretch a cell by using a rigid glass micro-plate at the bottom and a flexible plate at the top [132] because the cell can adhere to the bottom and top microplates.

- *Magnetic nanoparticle-based stimuli* is a technique that relies on the displacement of magnetic nanoparticles deposited on the cell membrane or injected into the cell by using a magnetic field.

1.2.3.2 Methods for measuring forces of cell-ECM connections

Many approaches for quantifying cellular mechanics do not rely on external force applications. Such techniques include those for measuring the forces on cell-ECM connections, beds of microneedles, DNA hairpin force sensors, and traction force microscopy [96]. We mainly introduce the method based on beds of microneedles and DNA hairpin force sensors in this section. Traction force microscopy will be introduced in the next section.
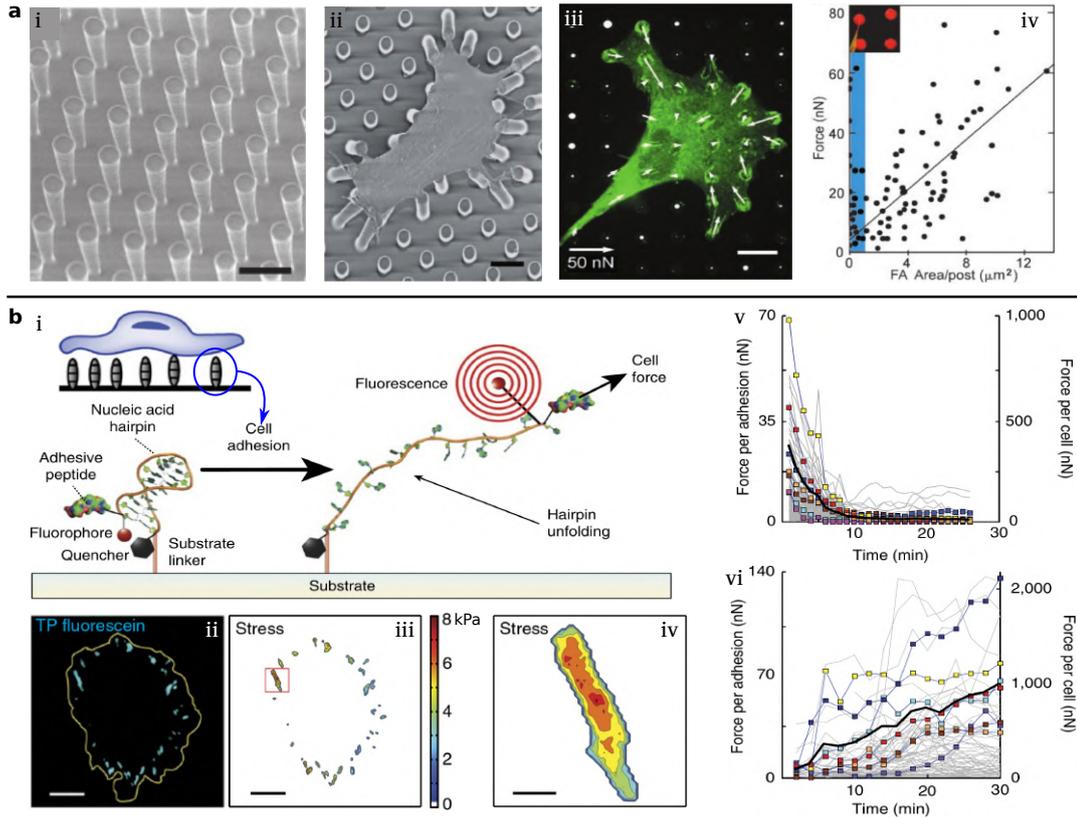
Figure 1.6: Two methods for measuring forces of cell-ECM connections. (a) Beds of microneedles. (a-i) A sample of artificial beds of microneedles. (a-ii) Scanning electron micrograph of a muscle cell attached to pillars. (a-iii) The force map was spatially correlated to immunofluorescence localization of the focal adhesion protein vinculin. (a-iv) Plot of the force generated on each of post as a function of total area of focal adhesion staining per post. Space bar: $10\,\mu$m. (b) DNA hairpin force sensors. (b-i) Schematic depiction of the tension probes. A DNA hairpin is functionalized with a fluorophore-quencher pair, covalently conjugated by the $3'$ end of the hairpin to a solid substrate and conjugated by its $5'$ end, via a PEG spacer, to the integrin-binding peptide RGD. (b-ii) Fluorescence signals of a mouse embryonic fibroblastsrate (MEF) cell on a TP-coated substrate. (b-iii) The traction stress maps calculated from the fluorescence level. (b-ii, iii) Space bar: $20\,\mu$m. (b-iv) The traction stress map of an individual adhesion site. Space bar: $3\,\mu$m. Mean stress per adhesion site as a function of time for individual cells treated with an inhibitor of contractions (Y-27632) (b-v) and lysophosphatidic acid (LPA) (b-vi), respectively. (Figure (a) adapted from J. L. Tan et al., 2003 [117]. Figure (b) adapted from B. L. Blakely et al., 2014 [133] with source material from W. J. Polacheck et al., 2016 [96].)

Beds of microneedles are an artificial substrate, illustrated in Fig. 1.6 (a-i) and were designed to measure the cell-generated forces on each of the needles [117]. A description of substrate fabrication can be found in Ref. [117]. To estimate the cell-generated forces one employs a simple model for beam bending, illustrated in Fig. 1.4 (c-iv). A horizontal force F is applied on the top of each microneedle. The horizontal force can be easily calculated to be $F = 3EId/L^3$, where E, I are respectively the Young's modulus and the moment of inertia. d and L are the horizontal displacement on the top of needle and the length of needle, respectively.

The mechanical and geometrical properties of the beds of microneedles can be adapted to the

experimental demand. For example, the beds of microneedles for measuring forces generated by a muscle sell are designed as a substrate containing posts (needles) of $3\,\mu$m diameter, $11\,\mu$m height, and $6\,\mu$m spacing, illustrated in Fig. 1.6 (a-ii) [117]. When a cell displaces the needles, the horizontal displacement on each needle d can be measured by using a microscope, for example, in fluorescence imaging mode. Then, the cell generated force on each needle are calculated from the displacement by using the beam bending model, as shown in Fig. 1.6 (a-iii). In Ref. [117], using this approach, the authors found that the force exerted on each one of the needles is proportional to the total area of the focal adhesion staining at this needle, illustrated in Fig. 1.6 (a-iv). The weaknesses of this approach are that it requires highly contractile cells and is unable to determine forces on lengthscales smaller than the needles [96].

The *DNA hairpin method* [133] utilizes molecular tension probes (TPs) to measure the traction forces of adherent cells. These TPs consist of DNA hairpins conjugated to fluorophore-quencher pairs and when the DNA hairpin unfolds under force, the shifted emission of the fluorophore can be measured by using fluorescence microscopy, illustrated in Fig. 1.6 (b-i), or by using Förster resonance energy transfer (FRET). In detail, the force at which the DNA hairpins unfold depends on the length and the composition of the DNA sequence.

Figure 1.6 (b-i) shows how the hairpin is connected to a cell and substrates. The $5'$ end of each hairpin is conjugated to the integrin adhesion receptor's GGRGDS peptide and the $3'$ end of the hairpin is functionalized with a free thiol linker, enabling chemical conjugation to the substrates. Using this DNA hairpin approach [134], the fluorescence signal field for a mouse embryonic fibroblast (MEF) is shown in Fig. 1.6 (b-ii) and the traction forces fields are displayed in Figs. 1.6 (b-iii, iv). Further, the authors found that the cell-generated forces are rapidly diminished, illustrated in Fig. 1.6 (b-v) when cells are treated with Y-27632, an inhibitor of contraction. The forces are increased as shown in Fig. 1.6 (b-vi) when the cells are treated with lysophosphatidic acid (LPA), a strong stimulant of contraction [135]. Although the DNA hairpin technique is a new approach to measure the forces exerted on the ECM, it also has several limitations, for example, bleaching of the fluorescent molecules, optical sensitivity, a limitation to two-dimensional imaging planes, long sample-prep time and, most importantly, the fact that an analog measurement of forces at each sensor is impossible. [96].

1.2.3.3 Traction force microscopy

Traction force microscopy (TFM) is a classical method to measure the cell-generated forces on ECM [37, 96, 136]. This perturbation-free method yields a spatial image of substrate stress exerted by cells on relatively soft elastic gel substrates, illustrated in Fig. 1.8 (a). The elastic gel is an artificial ECM, for example, a polyacrylamide hydrogel. This method has its origins in pioneering work by Harris et al. [137], who employed flexible silicone substrates to investigate the mechanical forces that cells generate. Nowadays, TFM has become a method that is routinely used in laboratories studying cell biology and soft matter physics around the world [96, 138–142].

In Fig. 1.7 (a), a 2D traction force field is displayed that was measured by using TFM for the case of a single Madin–Darby canine kidney (MDCK) epithelial cell on collagen-coated polyacrylamide gels [86]. The spots of traction force are distributed as a ring along the internal boundary of the cell. In Fig. 1.7 (b), we show the use of the TFM approach to measure the 2D forces field generated by groups of migrating bacteria and this work was to answer the
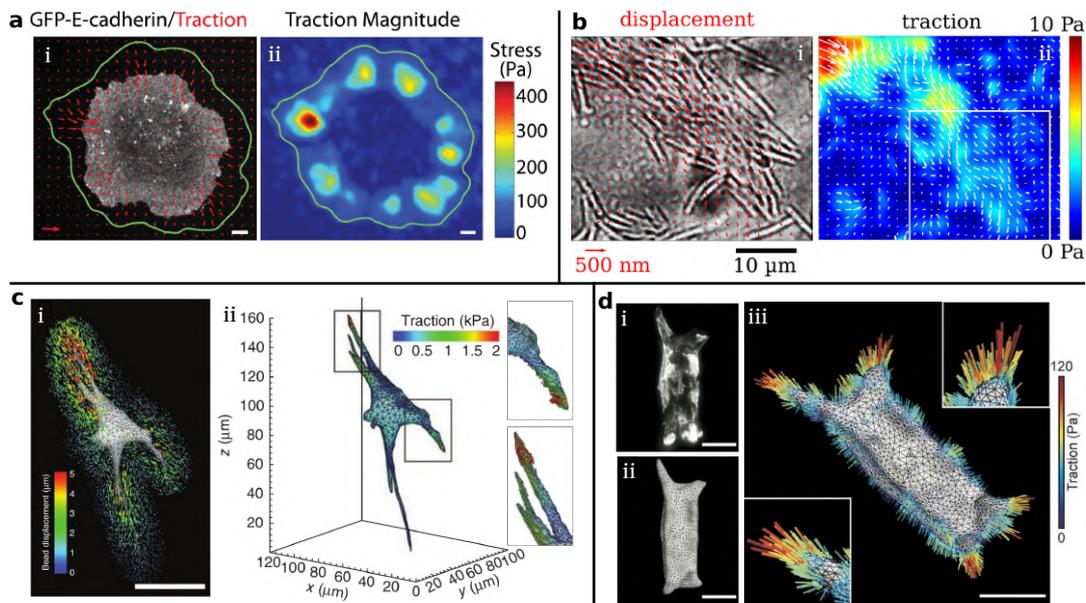
Figure 1.7: Several results of traction forces obtained by using TFM. (a-i) MDCK cell expressing GFP-E-cadherin with traction stress vectors superimposed on 2D substrates. The green line indicates the region used for calculation of total traction forces. (a-ii) Heat-scale plot of traction stress magnitudes of the cell shown in (a-i). Space bar: $5\,\mu$m. (b-i) Groups of cells move with a finger-forming spreading pattern on 2D substrates. Red quivers are gel displacement field. Only every fourth measurement is displayed, for clearer visibility. (b-ii) Calculated traction from the displacement field. (c-i) Bead displacement trajectories colour coded by magnitude for an EGFP-expressing NIH 3T3 fibroblast in 3D substrates. Space bar: $50\,\mu$m. (c-ii) Contour plot of the tractions (magnitude) exerted by a cell. (d-iii) A map of estimated traction forces for epithelial cells (d-i and d-ii) in 3D substrates. Space bar: $50\,\mu$m. (Figure (a) taken from V. Maruthamuthu et al., 2011 [86]. Figure (b) taken from B. Sabass et al., 2017 [143]. Figure (c) taken from W. R. Legant et al., 2010 [144]. Figure (d) adapted from N. Gjorevski et al., 2015 [145].)

question of how bacteria generate and coordinate forces during collective migration [143]. These experiments were done by using 2D TFM methods and recently the 3D TFM approach is also increasingly being used by biophysicists. In Fig. 1.7 (c), we show a 3D traction forces field that is obtained by using a TFM technique for the case of EGFP-expressing NIH 3T3 fibroblast spreading in the 3D substrates [144]. This 3D technique also can be employed to calculate the traction forces generated by a tissue of epithelial cells, illustrated in Fig. 1.7 (d) [86]. In this thesis, our aim is to develop some mathematical techniques for accurate traction reconstruction with a focus on 2D problems. Thus, we first introduce the experimental details for 2D TFM.

Figure 1.8 (b) contains a general flowchart for high-resolution 2D TFM that mainly includes seven steps [136, 146]: (1) Preparation of TFM substrates, which are coverslip-bound PAA gels containing two colors of fluorescent beads with covalently linked ECM. (2) Transfection of cells on TFM substrates. (3) Setting up a perfusion chamber. (4) Acquiring time-lapse TFM movies. (5) Detachement of cell by trypsinization and acquiring images. (6) Calculation of the displacement field by tracking the bead. (7) Reconstruction of traction force.

Figure 1.8: Schematic diagram of the procedure for performing high-resolution traction force microscopy (TFM) on a compliant PAA substrate. (a) Schematic of a TFM experiment depicting elastic substrate deformed by an adherent cell. (b) Flowchart for high-resolution TFM: (1) Preparation of TFM substrates. (2) Transfection of cells on TFM substrates. (3) Setting up a perfusion chamber. (4) Acquiring time-lapse TFM movies. (5) Detachment of cell by trypsinization and acquiring images. (6) Calculation of the displacement field by tracking the bead. (7) Reconstruction of traction force. Several steps of experimental images are shown in the right side. (Top) Space bar: $30\,\mu$m. (Bottom) Space bar: $10\,\mu$m. (Figure (a) taken from S. V. Plotnikov et al., 2014 [136]. Figure (b) adapted from S. V. Plotnikov et al., 2014 [136] and H. Colin-York et al., 2016 [146].)

In step (1) and (5), two techniques can be employed to obtain a high precision of the displacement measurement. One is the use of two colours of fluorescent beads in the preparation of TFM substrates [37]. Another one is the use of stimulated emission depletion (STED) microscopy for acquiring images because this technique can bypass the diffraction limit of light microscopy to increase the resolution [147]. When a pair of images containing fluorescent markers in the deformed and undeformed gel has been acquired, one can employ a tracking technique to obtain a discrete displacement field $\mathbf{u}$ from the images. Popular image analysis algorithms for this purpose are single-particle tracking (SPT), particle tracking velocimetry (PTV), and particle image velocimetry (PIV).

In step (7), our goal is to calculate the traction forces from the displacement field $\mathbf{u}$. A variety of methods exist for this purpose, for example, finite element methods [148–152], boundary element methods [37, 153, 154], and methods operating in Fourier space [37, 140, 152, 155, 156]. These approaches mostly are based on a mechanical model relating a discrete displacement $\mathbf{u}$ and the traction force field $\mathbf{f}$ [37]

$$\mathbf{u} = \mathbf{M}\mathbf{f} + \mathbf{s}, \tag{1.5}$$

where $\mathbf{M}$ is a coefficient matrix, whose form depends on the chosen method and which we will discuss the detail in Chapter 3 and Chapter 4. $\mathbf{s}$ represents measurement noise. Given a measured displacement field $\mathbf{u}$ and a coefficient matrix $\mathbf{M}$, we need to calculate $\mathbf{f}$ in Eq. (1.5). This problem has exactly the same form as the linear regression Eq. (1.1). Because the condition number $\kappa(\mathbf{M})$ is usually large, the calculation of $\mathbf{f}$ becomes an inverse and ill-posed problem. This means that even the smallest noise $\mathbf{s}$ leads to very large errors if a direct inversion of Eq. (1.5) is attempted. Thus, the TFM methodology is limited by two common serious issues that introduce systematic errors. First, the resolution of the measured traction is usually not high enough to resolve processes at micrometer-sized cellular structures. Secondly, the most commonly used TFM algorithms require the user to choose a filter or a regularization parameter, which introduces a considerable degree of subjectivity regarding smoothness and magnitude of the resulting traction.

The initial observation motivating this thesis was that all present methods for TFM rely on non-Bayesian techniques. L2 regularization was applied in TFM very early and this approach not only was used in real space calculations [157], but also in Fourier-space calculations [37]. Later, L1 regularization was employed in TFM and in approaches [158, 159], the L1 regularization was numerically implemented by making use of an algorithm called iterative reweighted least squares. However, L1 regularization can be also done by using proximal gradient methods, which usually operate in wavelet space and employ adaptive or non-adaptive thresholding of high spatial frequencies [160, 161]. Proximal gradient methods are for instance used for reconstructing lost parts of an image [162–164], for analysis of MRI data [165], and for analysis of genomic data [166, 167]. The L1 regularization solved by proximal gradient methods has not yet been tested for TFM. Moreover, other regularization approaches can be used for TFM. In our overview of approaches, Fig. 1.1, other regularization methods are mentioned, for example, the elastic net regularization. Literally, the elastic net regularization behaves like a stretchable fishing net that retains "all the big fish" while removing the small background signal [55]. The elastic net regularization problem also can be solved by using the proximal gradient methods [168]. The challenge regarding all regularization methods is that they require the manual selection of one or more regularization parameters. However, this challenge can be overcome by using a Bayesian approach. In this thesis, the Bayesian regularization will be employed for TFM.

Methods for solving inverse problems also play an important role for topical research on machine learning. In particular, sparse regularization is used for automatic inference of equations that describe measured data. In the following section, we will therefore provide some background information on data-driven methods for automatic discovery of governing equations.

1.3 Background information on data-driven discovery of governing physical equations

Many if not most models employed in the natural sciences can be expressed through differential equations. These can be ordinary-, partial-, and stochastic differential equations. In physics, the equations can often be derived from first principles, for instance, conservation of mass, energy and momentum, and thermodynamic considerations. A classical example is the Navier-Stokes equation in fluid mechanics which expresses conservation of momentum.

However, in complex systems as they occur for example in biophysics, climate science, and neuroscience, first principles determining the system behavior are hardly known since the systems are typically not close to an equilibrium state, processes can be highly nonlinear, and the dynamics can occur on multiple scales. In these cases, one can resort to phenomenological, effective descriptions that may include some level of coarse graining and are based on experimental data. Recently, an increased computational power has made it possible to construct such models in an automated fashion, which is known as *data driven discovery of governing equations*.

During the last decades, various approaches have been developed to determine the underlying governing equations in a non-linear dynamical system from measurement data. Focusing on chaotic systems, early attempts were made to fit parameters in an appropriate basis to generate an analytical description of the system dynamics [169]. In "*symbolic regression*", one employs function libraries to automatically extract the terms in a governing equation that best represents the measured data according to some optimization criterion [170, 171]. Recently, the use of sparse regression techniques for symbolic regression has received considerable scientific attention [39, 40]. In sparse symbolic regression, the physical quantity u often can be taken to have the general mathematical form $\check{u} = N(u, x, t, \vartheta)$, where $\check{u}$ is the left-hand side of the governing equation, e.g., $\check{u} = u_t$, and $N(\cdot)$ is an unknown function on the right-hand side of the governing equation. x, t, and ϑ respectively represent the space, time, and a parameter. To identity the function $N(\cdot)$, a *library* $\boldsymbol{\Theta}(\mathbf{u})$ can be constructed, where $\mathbf{u}$ is a time-series data and the candidate mathematical expressions in the library can be linear, nonlinear, and partial derivative terms. Thus, the general mathematical form can be written as $\check{\mathbf{u}} = \boldsymbol{\Theta}(\mathbf{u})\boldsymbol{\xi}$, where $\boldsymbol{\xi}$ is a *sparse vector* because the governing equation is expected to have a simple form. This sparse vector can be identified by using a sparse regression technique and each zero element in the vector $\boldsymbol{\xi}$ corresponds to a term in the library that the term is excluded from the governing equation. We will mainly focus on this sparse symbolic regression approach and next, we will show how this approach has recently been employed to identify ordinary differential equations (ODEs), partial differential equations (PDEs), and stochastic differential equations (SDEs).

1.3.1 Inference of ordinary and partial differential equations

In Ref. [39], the authors show that the governing ODE, e.g., Lorenz equations, can be discovered from time-series data by using a sparse symbolic regression approach, illustrated in Fig. 1.9 (a). In a three-dimensional Lorenz system, the data is a time-series of positions $\mathbf{X} = (\mathbf{x}, \mathbf{y}, \mathbf{z})$. By introducing a function library $\boldsymbol{\Theta}(\mathbf{x})$ that is evaluated with the measurement data, one can write the the dynamical equation as

$$\dot{\mathbf{x}} = \boldsymbol{\Theta}(\mathbf{x})\boldsymbol{\xi}_x + \mathbf{s},$$

where $\mathbf{s}$ is a noise, e.g., measurement noise. $\boldsymbol{\xi}_x$ is a sparse vector containing the weights associated with each non-vanishing term in the library. In this equation, $\dot{\mathbf{x}}$ and $\boldsymbol{\Theta}(\mathbf{x})$ can be constructed from the time-series data , details will be given in Chapter 5. The aim is to obtain the sparse vector $\boldsymbol{\xi}_x$. This problem has exactly the same form as our regression Eq. 1.1. In symbolic regression, the condition number $\kappa(\boldsymbol{\Theta})$ is usually large and thus this the inference of $\boldsymbol{\xi}_x$ is an inverse and ill-posed problem.

In Ref. [39], the authors propose a method called sparse identification of nonlinear dynamics (SINDy), which is essentially based on a sequential thresholded least squares algorithm. At each iteration, $\boldsymbol{\xi}_x$ is first obtained from a least-squares optimization and then $\boldsymbol{\xi}_x$ is thresholded such that values smaller than a cutoff $\varkappa$ are set to zero. These steps are repeated until convergence conditions are satisfied. Using this algorithm, the Lorenz equations are identified as illustrated in Fig. 1.9 (a).
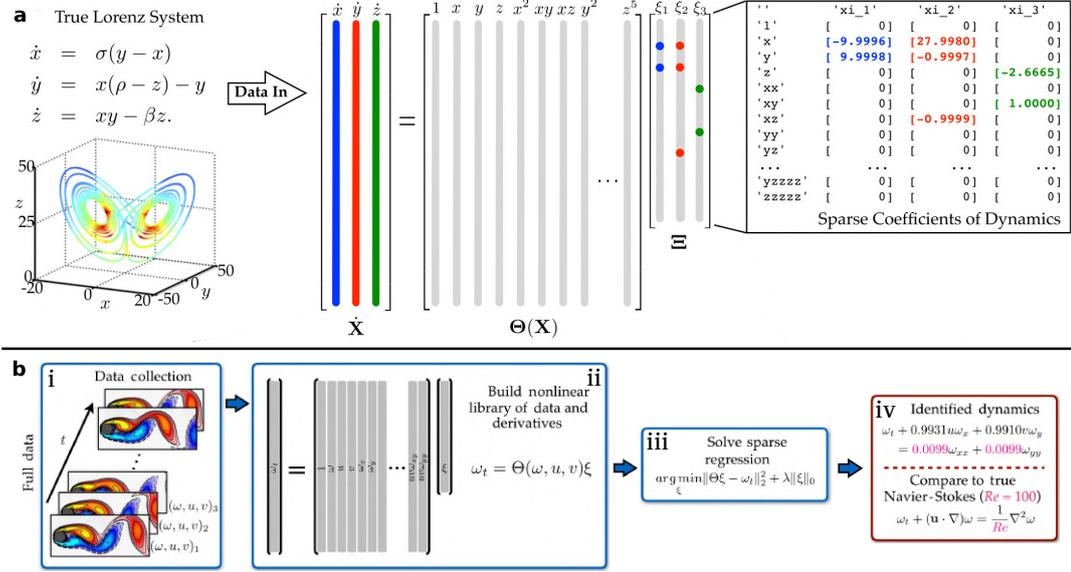
Figure 1.9: Identification of ordinary and partial differential equations by using sparse symbolic regression. (a) Schematic of the SINDy algorithm identifies the Lorenz equations. The artificial data, time-series positions $\mathbf{X} = (\mathbf{x}, \mathbf{y}, \mathbf{z})$, are calculated from the Lorenz system. Next, the library $\boldsymbol{\Theta}(\mathbf{X})$ and the $\dot{\mathbf{X}}$ are constructed from the given data. Using the library approach, the governing equations can be written $\dot{\mathbf{X}} = \boldsymbol{\Theta}(\mathbf{X})\boldsymbol{\Xi}$. The few terms of vector $\boldsymbol{\Xi}$ are calculated by sparse regression with a threshold. (b) Identification of Navier-Stokes equations. (b-i) The artificial data is collected as snapshots of a solution from the Naver-Stokes equations. (b-ii) The library $\boldsymbol{\Theta}$ and the $\boldsymbol{\omega}_t$ are constructed from the artificial data. The matrix library form is written as $\dot{\boldsymbol{\omega}} = \boldsymbol{\Theta}(\boldsymbol{\omega}, \mathbf{u})\boldsymbol{\xi}$. (b-iii) The sparse vector $\boldsymbol{\xi}$ is calculated by using the sparse regression. (b-iv) The identification of Naver-Stokes equations is obtained from the data. (Figure (a) adapted from S. L. Brunton et al., 2016 [39]. Figure (b) adapted from S. H. Rudy et al., 2017 [40].)

Partial differential equation, containing partial derivatives of a physical quantity with respect to the independent variables, e.g, time and space, have also been studied in the context of sparse regression. The identification of PDE, e.g., Navier-Stokes equations, is possible with given time-space-series data, e.g., the vorticity $\boldsymbol{\omega}$ and velocity $\mathbf{u}$, illustrated in Fig. 1.9 (b) [40]. The governing PDE's in the sparse symbolic regression approach are written as

$$\dot{\boldsymbol{\omega}} = \boldsymbol{\Theta}(\boldsymbol{\omega}, \mathbf{u})\boldsymbol{\xi} + \mathbf{s}, \tag{1.6}$$

where $\dot{\boldsymbol{\omega}}$ and $\boldsymbol{\Theta}(\boldsymbol{\omega}, \mathbf{u})$ can be constructed from the time-space-series data, see Chapter 5. Here, again, the calculation of $\boldsymbol{\xi}$ also becomes an inverse and ill-pose problem. To solve Eq. 1.6 for the identification of PDE, an algorithm called train Sequential Threshold Ridge regression (TrainSTRidge) was described in Ref. [40]. Sequential Threshold Ridge regression (STRidge) is a sequentially thresholded least squares (STLS) optimization procedure

for ridge regression. In STRidge, the vector $\boldsymbol{\xi}$ is first calculated by using ridge regression with a regularization parameter λ_2. Then, the $\boldsymbol{\xi}$ is set to zero for all elements that have a smaller absolute value than $\varkappa$, which is the same thresholding procedure as in SINDy. To automatically estimate the threshold $\varkappa$, the authors added a training step to the STRidge and therefore call their methods TrainSTRidge [40]. In this training step, the cost function is calculated from part of the data by using a L0 regularization and details of this algorithm will be described in Chapter 5. However, this TrainSTRidge algorithm still requires the user to choose a regularization parameter. This drawback has been eliminated in threshold sparse Bayesian regression which also was employed for identification of PDE [172]. This Bayesian regression approach is the same as the sparse Bayesian learning method shown in Fig 1.1. The advantage of this Bayesian approach is that it does not require the provision of a regularization parameter, but it still requires a manually selected threshold. In this thesis, we will develop a new approach to solve the inverse ill-pose problem in data-driven discovery of governing physical equations that combines the training approach with the Bayesian method.
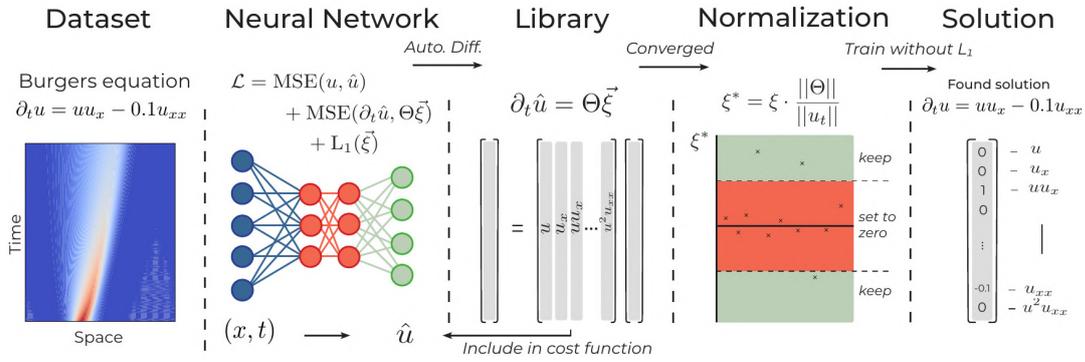
Figure 1.10: The neural network algorithm for the identification of partial differential equations from type II noisy data. The dataset is calculated from Burgers equation and then noise is added. The reconstructed data is obtained from a neural network. The Burgers equation is identified from the reconstructed data. (Figure adapted from G. J. Both et al., 2019 [82].)

The approaches for solving inverse problems that are schematically depicted in Fig 1.1 are mostly based on the assumption that the noise $\mathbf{s}$ obeys a Gaussian distribution in $\mathbf{g} = \boldsymbol{\Phi}\mathbf{w} + \mathbf{s}$. We call this type of noise *type I Gaussian noise*. However, in some experiments [40, 82], a Gaussian noise is directly added in the measurement data, for example, $\mathbf{u}$ and $\boldsymbol{\omega}$ in Eq. (1.6). We call this type of noise *type II Gaussian noise*. The vector $\dot{\boldsymbol{\omega}}$ and the library $\boldsymbol{\Theta}(\boldsymbol{\omega}, \mathbf{u})$ can be constructed like in the case of type I noise yielding $\dot{\boldsymbol{\omega}} = \boldsymbol{\Theta}(\boldsymbol{\omega}, \mathbf{u})\boldsymbol{\xi} + \hat{\mathbf{s}}$, where $\hat{\mathbf{s}}$ does not obey a Gaussian anymore. Therefore, it is a challenge to identify the governing equations from data with high levels of type II noise. To overcome this challenge, a neural network deep learning method can be employed to reduce the high level of noise in measurement data [82], as illustrated in Fig. 1.10. In this deep learning approach, the cost function is written as $\mathcal{L}_{\text{cost}} = \mathcal{L}_{\text{MSE}}(u, \hat{u}) + \mathcal{L}_{\text{Reg}}(\dot{\hat{u}}, \boldsymbol{\Theta}\boldsymbol{\xi}) + \mathcal{L}_{\text{L1}}(\boldsymbol{\xi})$, where $\mathcal{L}_{\text{MSE}}$ is a mean squared error, $\mathcal{L}_{\text{Reg}}$ presents a regression penalty, and $\mathcal{L}_{\text{L1}}$ is a L1 penalty. Using such a method, data-driven identification of the Burgers equation from 75% type II Gaussian noisy data has been demonstrated [82]. Besides ODE and PDE, stochastic differential equations are a further important framework for describing physical processes. In the next section we will introduce data-driven identification of stochastic systems.

1.3.2   Stochastic differential equations

Langevin-type equations describe the evolution of system variables under the influence of stochastic noise [173]. The Langevin equation, a SDE, is written as a sum of deterministic- and stochastic parts $dX(t) = g(X(t), t) + h(X(t), t)\Gamma(t)$, where $X(t)$ denotes a time dependent state. $\Gamma(t)$ is a time-dependent random variable satisfying $\langle\Gamma(t)\rangle = 0$ and $\langle\Gamma(t)\Gamma(t')\rangle = \delta(t - t')$.
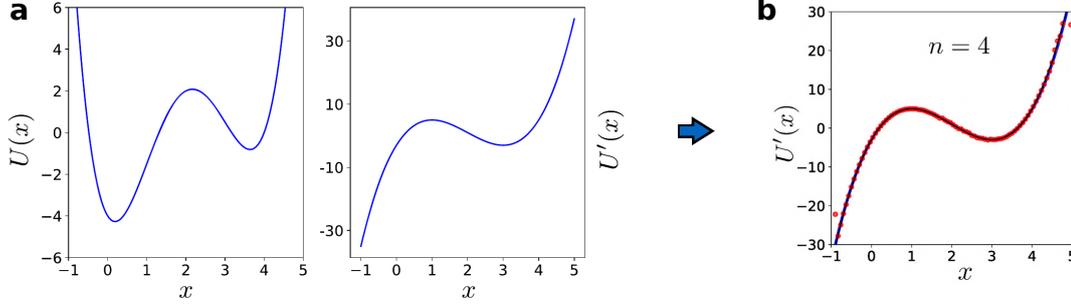
Figure 1.11: The identification of stochastic differential equations. (a) Potential energy profile $U(x)$ (Left), and its gradient $-\nabla U(x)$ called the drift force (Right). (b) Binned data from the trajectory (red points), the reconstructed result (blue line) and the real data (black line) for the drift force. (Figure adapted from L. Boninsegna et al., 2018 [174].)

The stochastic process described by the Langevin equation can also be described by a corresponding Fokker-Planck equation for the probability density function $f_X(x, t)$, $\partial f_X(x, t)/\partial t = \hat{L}(x, t) f_X(x, t)$, where $\hat{L}(x, t) = -\partial D^{(1)}/\partial x + \partial^2 D^{(2)}/\partial x^2$. Here, the Kramers-Moyal (KM) coefficients are $D^{(n)} = \lim_{\tau \to 0} 1/(n!\tau)\langle[X(t + \tau) - X(t)]^n\rangle_{X(t)=x}$. These coefficients are connected to the functions g and h in the Langevin equation provided above as, $g = D^{(1)}$ and $h = \sqrt{2D^{(2)}}$ [175]. To identify g and h from a trajectory data of the state X, we can use the formula for the KM coefficients and construct a general form for the drift and diffusion terms as

$$\mathbb{D} = \mathbf{\Theta W}_1,$$
$$\mathbb{F} = \mathbf{\Theta W}_2,$$

where $\mathbb{D}$, $\mathbb{F}$, and library $\mathbf{\Theta}$ can be reconstructed from the trajectory data, see Chapter 5. Our aim is to calculate the sparse vectors $\mathbf{W}_1$ and $\mathbf{W}_2$ for a given data set. Here, again, an inverse ill-posed problem needs to be solved. In Ref. [174], the authors modify the threshold sparse regression approach to enforce sparsity iteratively, which can remove the need of threshold parameter. Using this algorithm, they develop a Stepwise Sparse Regressor (SSR) for the standard least square regression and employ Cross Validation to select the number of iterations. The authors are able to correctly infer the drift forces from stochastic trajectories, illustrated in Fig. 1.11. However, since this algorithm is based on Cross Validation, a manual threshold must be provided. Thus, in this thesis, we will use our new approach to identity SDE in a completely parameter free approach.

1.4   Overview of the remaining chapters

The remainder of this thesis is structured in five chapters. In Chapter 2, various methods are described to solve inverse ill-posed problems. These methods mainly include regularization methods and Bayesian approaches. The regularization approaches studied are L2-, L1- and elastic net regularization, and the Bayesian methods include Bayesian regularization and sparse Bayesian learning.

In Chapter 3, we demonstrate the application of regularization methods and Bayesian regularization for TFM. First, we test various regularization methods which include the classical methods, L1- and L2 regularization, and previously untested methods from computer vision, namely Elastic Net (EN) regularization, Proximal Gradient Lasso (PGL), and Proximal Gradient Elastic Net (PGEN). Secondly, we develop two Bayesian methods into TFM that are Bayesian L2 regularization (BL2), where the magnitude of the noise in the displacement data must be measured separately, and Advanced Bayesian L2 regularization (ABL2) which requires no extra input.

Chapter 4 mainly contains the exposition of a new approach, Bayesian Fourier transform traction cytometry (BFTTC), which not only can calculate traction forces very efficiently by using Fourier transform traction cytometry, but also can automatically select the regularization parameter by using a Bayesian approach. This chapter also contains an explanation of a user-friendly software package which contains Fourier transform traction cytometry (FTTC) and BFTTC.

In Chapter 5, we develop a new approach to identify the governing equations from measurement data, called automatic threshold sparse Bayesian learning (ATSBL), which does not require the provision of any parameter. This approach is employed for the identification of ODE, PDE, and SDE. Further, we employ a deep learning approach to reduce the noise in measurement data and thus the governing equations can be discovered from data containing high levels of non-Gaussian noise.

In Chapter 6, we discuss the results obtained in this thesis and suggest future research directions regarding the two applications, TFM and the identification of governing equations from measurement data.

Chapter 2

Methods for solving ill-posed inverse problems

2.1   Inverse and ill-posed problems

We consider a system matrix $\{\Phi_{ij}\}$ and a vector of observation data $\{g_i\}$, where $i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, M$. Here, N, M are positive integers. A vector $\{w_j\}$ with unknown entries has a simple linear relationship with the observed data as [176]

$$g_i = \sum_{j=1}^{M} \Phi_{ij} w_j + s_i,$$

where $\{s_i\}$ represents an additive random noise drawn from a zero-mean Gaussian distribution. Like in Chapter 1, this relationship is given in matrix notation by

$$\mathbf{g} = \mathbf{\Phi}\mathbf{w} + \mathbf{s}, \tag{2.1}$$

In the previous Chapter 1, we mentioned that linear problems like Eq. (2.1) can be ill-posed. Here, we provide a generic definition for ill-posedness through the definition of *well-posedness* of a problem [177]. We posit that the solution $\mathbf{w}$ is to be estimated from a given data $\mathbf{g}$ with the operator $\mathbf{\Phi}$ in Eq. (2.1). Each element of $\mathbf{w}$ and $\mathbf{g}$ lies in their each metric space W and Q with metrics $\rho_W(w_1, w_2)$ for $w_1, w_2 \in W$ and $\rho_Q(g_1, g_2)$ for $g_1, g_2 \in Q$.

Figure 2.1 shows a relationship between the spaces (W and Q). The metrics $\rho_Q(g_1, g_2)$ and $\rho_W(w_1, w_2)$ frequently represent the Euclidean distance. The space W and Q are assumed to be Hilbert spaces. A well-posed problem of determining the solution w in the space W from the given data g in the space Q governed by the operator $\mathbf{\Phi}$ has to satisfy the following three conditions [45, 177], (1) existence: for any element $g \in Q$, there exists a solution w in the space W to the equation $\mathbf{g} = \mathbf{\Phi}\mathbf{w}$; (2) uniqueness: the solution w to the equation $\mathbf{g} = \mathbf{\Phi}\mathbf{w}$ is unique in the space W; (3) stability: for any neighborhood $\ddot{w} \subset W$ of the solution w to the equation $\mathbf{g} = \mathbf{\Phi}\mathbf{w}$, there is a neighborhood $\ddot{g} \subset Q$ of $\mathbf{g}$ such that for all $\hat{\ddot{g}} \in \ddot{g}$ the element $\mathbf{\Phi}^{-1}\hat{\ddot{g}} = \hat{\ddot{w}}$ belongs to the neighborhood $\ddot{w}$. The problem $\mathbf{g} = \mathbf{\Phi}\mathbf{w}$ is ill posed if at least one of the three conditions does not hold. The operator $\mathbf{\Phi}$ of a linear ill-posed problem is called an *ill-conditioned matrix*. In this chapter, we will introduce mathematical formulations of practical methods to solve linear, ill-posed problems.
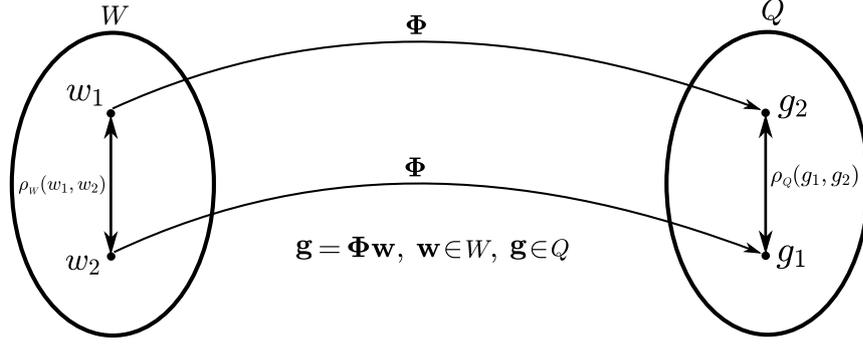
Figure 2.1: Well posedness is defined by the relationships between elements, spaces, the operator, and metrics. The solutions (w_1, w_2) in the space W can be obtained from the given data (g_1, g_2) in the space Q and the operator $\mathbf{\Phi}$. Typically, the metrics $\rho_W(w_1, w_2)$ and $\rho_Q(g_1, g_2)$ are the Euclidean distances and the spaces W and Q are Hilbert spaces. (Figure adapted from M. A. Abidi et al., 2016 [177].)

2.2 Regularization methods

A classical heuristic approach to calculate $\mathbf{w}$ for a linear, ill-posed inverse problem as given in Eq. (2.1) is regularization. Here, not only the residual ($\|\mathbf{g} - \mathbf{\Phi w}\|_2^2 = (\mathbf{g} - \mathbf{\Phi w})^{\mathbf{T}}(\mathbf{g} - \mathbf{\Phi w})$) is minimized in a least-squares sense, but also the magnitude of the solution is penalized through its p-norm denoted by $\|\mathbf{w}\|_p$. The trade-off between minimization of the residual and minimization of the solution norm is determined by fixed regularization parameters, λ_1 and λ_2, leading to a minimization problem of the type

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}}\left[\|\mathbf{\Phi w} - \mathbf{g}\|_2^2 + \lambda_1\|\mathbf{R}_1\mathbf{w}\|_1 + \lambda_2\|\mathbf{R}_2\mathbf{w}\|_2^2\right]. \tag{2.2}$$

The two norms are explicitly written as $\|\mathbf{x}\|_1 = \sum_k |x_k|$ and $\|\mathbf{x}\|_2^2 = \sum_k x_k^2$. $\mathbf{R}_1$ and $\mathbf{R}_2$ are functions that are to be defined, e.g., as the unit matrix $\mathbf{I}$. In this thesis, we will mainly focus on the following regularization methods: L2-, L1-, and elastic net regularization.

2.2.1 L2 regularization

L2 regularization, employing an $L2$-norm with $\lambda_2 > 0$ and $\lambda_1 = 0$ in Eq. (2.2) to penalize the magnitude of $\mathbf{w}$ through $\mathbf{R}_2 = \mathbf{I}$ is currently the most common technique used for the inverse ill-posed problem. Thus, the minimization problem becomes

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}}\left[\|\mathbf{\Phi w} - \mathbf{g}\|_2^2 + \lambda_2\|\mathbf{w}\|_2^2\right]. \tag{2.3}$$

L2 regularization is also known as ridge regression or Tikhonov regularization [178] and this method efficiently produces a continuous and smooth solution $\hat{\mathbf{w}}$. This approach is widely applied in the traction force microscopy [37, 146, 157].

Equation (2.3) can be easily solved by a variational approach and its analytical solution is given by

$$\hat{\mathbf{w}} = (\mathbf{\Phi^T\Phi} + \lambda_2\mathbf{I})^{-1}\mathbf{\Phi^T g}. \tag{2.4}$$

The inverse of the typically large matrix $(\mathbf{\Phi^T\Phi} + \lambda_2\mathbf{I})^{-1}$ in Eq. (2.4) is often difficult to calculate directly. This calculation can become easier by using the *singular value decomposition*

(SVD). The real $N \times M$ matrix $\mathbf{\Phi}$ can be decomposed uniquely as $\mathbf{\Phi} = \mathbf{UDV^T}$, where $\mathbf{U}$ is a $N \times M$ matrix and column orthogonal ($\mathbf{U^T U = I}$), $\mathbf{D} = \mathrm{diag}(d_1, \cdots, d_i, \cdots, d_M)$ is a $M \times M$ diagonal matrix, and $\mathbf{V}$ is a $M \times M$ orthogonal matrix ($\mathbf{V^T V = I}$). Thus, part of Eq. (2.4) can be rewritten as

$$
\begin{aligned}
(\mathbf{\Phi^T \Phi} + \lambda_2 \mathbf{I})^{-1} \mathbf{\Phi^T} &= (\mathbf{VD^T U^T UDV^T} + \lambda_2 \mathbf{VIV^T})^{-1} \mathbf{VD^T U^T} \\
&= \mathbf{V}(\mathbf{D^T D} + \lambda_2 \mathbf{I})^{-1} \mathbf{D^T U^T} \\
&= \mathbf{V}\mathrm{diag}(\frac{d_i}{d_i^2 + \lambda_2}) \mathbf{U^T}.
\end{aligned}
$$

Therefore, Eq. (2.4) becomes $\hat{\mathbf{w}} = \mathbf{V}\mathrm{diag}\big(d_i/(d_i^2 + \lambda_2)\big) \mathbf{U^T g}$, which does not require the calculation of a large inverse matrix. If $\mathbf{R}_2 \neq \mathbf{I}$, we can employ a generalized singular value decomposition (GSVD) for the matrix pair $\mathbf{\Phi}$ and $\mathbf{R}_2$, see Ref. [61].

2.2.2 L1 regularization

L1 regularization, also called Least absolute shrinkage and selection operator (Lasso) [49], is realized through setting $\lambda_2 = 0$, $\lambda_1 > 0$, and $\mathbf{R}_1 = \mathbf{I}$ in Eq. (2.2). Thus, Eq. (2.2) becomes in this case

$$
\hat{\mathbf{w}} = \underset{\mathbf{w}}{\mathrm{argmin}}\big[\|\mathbf{\Phi w} - \mathbf{g}\|_2^2 + \lambda_1 \|\mathbf{w}\|_1\big]. \tag{2.5}
$$

With L1 regularization, small values of the reconstructed signal are efficiently set to zero. L1 regularization is therefore frequently used in the field of compressive sensing (CS) [179, 180], where the underlying assumption is that the signal can be represented in a sparse form where all but a few components of the signal vanish. Recently, the technique has been used for TFM [158, 159, 181, 182], information transmission [183] and image analysis [71] and it is appropriate for traction fields containing few, sparsely located traction hotspots. The L1 regularization problem in Eq. (2.5) can be solved by using several approaches, for example, iterative reweighed least square, proximal gradient methods, and convex optimization solvers.

- Iterative reweighed least square

The *iterative reweighed least square* (IRLS) method is a general approach to solve L1 regularization. For IRLS [158, 184], we first compute the gradient of the function $f(\mathbf{w}) = \|\mathbf{\Phi w} - \mathbf{g}\|_2^2 + \lambda_1 \|\mathbf{w}\|_1$ with respect to $\mathbf{w}$ and obtain

$$
\nabla f(\mathbf{w}) = 2\mathbf{\Phi^T \Phi w} - 2\mathbf{\Phi^T g} + \lambda_1 \mathrm{sgn}(\mathbf{w}), \tag{2.6}
$$

where $\mathrm{sgn}(\mathbf{w}) = (w_1/|w_1|, \ldots, w_i/|w_i|, \ldots, w_M/|w_M|)^{\mathbf{T}}$. If w_i is zero, $f(\mathbf{w})$ is not differentiable because the absolute function at zero is not differentiable. To get a meaningful gradient, we employ a threshold ς

$$
w_i = \left\{ \begin{array}{ll} w_i, & w_i > \varsigma \\ \varsigma, & w_i \leq \varsigma \end{array} \right.,
$$

for example, $\varsigma = 1e - 7$. Then, Eq. (2.6) becomes

$$
\nabla f(\mathbf{w}) = 2\mathbf{\Phi^T \Phi w} - 2\mathbf{\Phi^T g} + \lambda_1 \mathbf{Kw},
$$

where $\mathbf{K}$ is defined as a diagonal matrix $\text{diag}(1/|w_1|, \cdots, 1/|w_i|, \cdots, 1/|w_M|)$. To find the value of $\mathbf{w}$ at the minimum of the function $f(\mathbf{w})$, we let $\nabla f(\mathbf{w}) = 0$ and obtain the following equation

$$(2\mathbf{\Phi^T\Phi} + \lambda_1\mathbf{K})\mathbf{w} = 2\mathbf{\Phi^T}\mathbf{g}. \tag{2.7}$$

Since $\mathbf{K}$ depends on $\mathbf{w}$, the nonlinear Eq. (2.7) is solved by an iteration method. In Ref. [158], the authors take the solution of L2 regularization as an initial solution $\mathbf{w}^0$ and then $\mathbf{K}$ is updated from $\mathbf{w}^0$. Thus, the new $\mathbf{w}^1$ can be calculated in Eq. (2.7), which includes an inverse matrix $(2\mathbf{\Phi^T\Phi} + \lambda_1\mathbf{K})^{-1}$. This inverse matrix can be easily solved by using the GSVD approach mentioned in above. The process is stopped at some iteration j, where the convergence conditions are satisfied, for example,

$$\frac{\|\mathbf{w}^j - \mathbf{w}^{j-1}\|_2}{1 + \|\mathbf{w}^{j-1}\|_2} < \epsilon, \tag{2.8}$$

where $\epsilon > 0$ is a small prescribed tolerance.

- Proximal gradient methods for L1 regularization (PGL)

Proximal gradient methods provide a way to robustly solve optimization problems involving locally non-differentiable, convex penalty functions. Proximal gradient methods are widely applied for image in-painting, which is the process of reconstructing lost or deteriorated parts of images [162, 165, 185–187]. We follow well-established approaches [168, 186]. The target function Eq. (2.5) to be considered reads

$$\hat{\mathbf{w}} = \operatorname*{argmin}_{\mathbf{w}}\Big[\frac{1}{2}\|\mathbf{\Phi}\mathscr{W}^*\mathscr{W}\mathbf{w} - \mathbf{g}\|_2^2 + \lambda_1\|\mathscr{W}\mathbf{w}\|_1\Big], \tag{2.9}$$

where $\mathscr{W}$ and $\mathscr{W}^*$ denote an unitary wavelet transform and an inverse wavelet transform, respectively. For wavelet transformation, we employ the lifting transform function provided with Ref. [187]. Here, the non-differentiable penalty function is denoted by $d(\mathbf{w}) = \lambda_1\|\mathscr{W}\mathbf{w}\|_1$. The optimization problem is solved through iterative soft thresholding where the regularization parameters control the threshold below which the wavelet-coefficients [188, 189] are to zero. The optimization procedure is iterative and based on gradient decent for the differentiable term $\|\mathbf{\Phi}\mathbf{w} - \mathbf{g}\|_2^2/2$. If we were to employ a gradient decent for this term only, the $\mathbf{w}$ at the iteration number $j+1$ would be given by $\mathbf{w}_{j+1} = \mathbf{k}_j$ with $\mathbf{k}_j = \mathbf{w}_j - \tau\mathbf{\Phi^T}(\mathbf{\Phi}\mathbf{w}_j - \mathbf{g})$ where τ is the step size. However, since the solution must also obey the regularization constraints, the iteration is modified as follows

$$\mathbf{w}_{j+1} = \operatorname*{argmin}_{\mathbf{w}}[\tau d(\mathbf{w}) + \frac{1}{2}\|\mathbf{k}_j - \mathbf{w}\|^2] =: \text{prox}_{\tau d(\mathbf{w})}(\mathbf{k}_j). \tag{2.10}$$

Hence, the proximal gradient scheme produces incremental changes that balance a gradient decent to minimize the solution residual with a minimization of the penalty. The right-hand side of Eq. (2.10) is a definition of the so-called proximity operator. For the L1 norm, $d(x) = \lambda_1|x|$, the proximity operator can be given in closed form as a threshold function to be applied to each element $\text{prox}_{\tau\lambda_1\|x\|_1}(x) = (|x| - \tau\lambda_1)_+\text{sgn}(x) =: S(x, \tau\lambda_1)$. Since the variable x are for our target function in Eq. (2.8) the wavelet coefficients, result must be transformed back to real space after every iteration. The result is, see e.g. [168, 185, 186],

$$\mathbf{w}_{j+1} = \mathscr{W}^*S\big[\mathscr{W}(\mathbf{w}_j - \tau\mathbf{\Phi^T}(\mathbf{\Phi}\mathbf{w}_j - \mathbf{g})); \tau\lambda_1\big].$$

In order to obtain a stable iteration, the step size needs to obey $\tau < 2/\|\mathbf{\Phi^T\Phi}\|_2$. Iterations are stopped when the change of the solution norm from time-step is smaller than a threshold, e.g., $10^{-8}\%$.

2.2.3 Elastic net regularization

The elastic net (EN) regularization [55] combines L1- and L2 regularization, $\lambda_1 > 0$ and $\lambda_2 > 0$ in Eq. (2.2). EN regularization is well established for a wide variety of applications, most notably the analysis of genetic data [59, 166, 190, 191] and images [192–194]. We can simply set $\mathbf{R}_1 = \mathbf{R}_2 = \mathbf{I}$ in Eq. (2.2) and then obtain the following equation

$$\hat{\mathbf{w}}_{\text{naive elastic net}} = \underset{\mathbf{w}}{\operatorname{argmin}}\big[\|\mathbf{\Phi w} - \mathbf{g}\|_2^2 + \lambda_1\|\mathbf{w}\|_1 + \lambda_2\|\mathbf{w}\|_2^2\big]. \tag{2.11}$$

It is called naive elastic net. Figure 2.2 (a) shows two-dimensional contours of constraint functions for L1, L2 and EN regression. The constraint functions are $(|w_1| + |w_2|) = t$, $\sqrt{w_1^2 + w_2^2} = t$ and $\sqrt{w_1^2 + w_2^2} + (|w_1| + |w_2|) = t$ for L1, L2 and EN regularization, respectively, where t is a small constant parameter, e.g., $t = 0.1$ [195]. The operational characteristics of these three methods in an orthogonal design are shown in Fig. 2.2 (b). The absolute solutions of Lasso regularization $\|w_i\| < s$ are set to zero, where s is a constant threshold. Thus, the Lasso regularization is called "soft thresholding". The naive EN regularization can be viewed as a combination of Lasso-type thresholding with a ridge-type direct shrinkage [55].
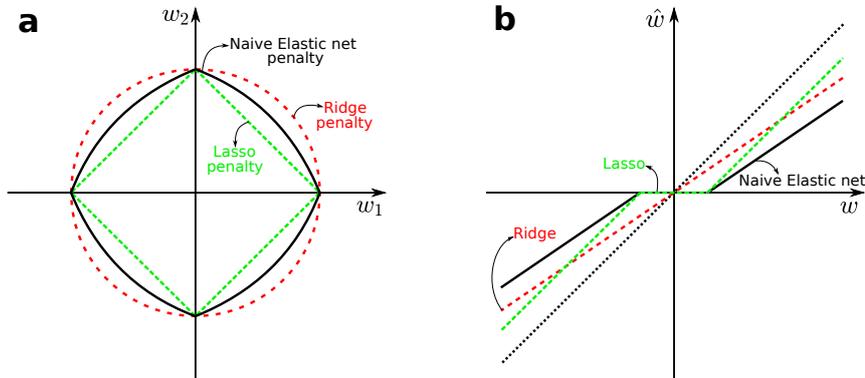
Figure 2.2: The 2D contour and exact solutions for different regularization methods. (a) Two-dimensional contours of constraint functions for L1, L2 and EN regression. (b) The operational characteristics of three regularization methods in an orthogonal design. Lasso regularization called "soft thresholding" means that the solution is set to zero for $\|w_i\| < s$, where s is a constant threshold. The naive EN regularization is viewed as a combination of Lasso-type thresholding with a ridge-regression type direct shrinkage. (Figure adapted from H. Zou et al., 2005 [55].)

The empirical evidence shows that the solution $\hat{\mathbf{w}}$ in Eq. (2.11) does not perform satisfactorily unless it is very close to either the result of ridge regression or the lasso, see Ref. [55]. Thus, it is called naive elastic net. In Ref. [55], an improved formula for the optimization problem in EN regularization is given as

$$\hat{\mathbf{w}}^* = \underset{\mathbf{w}^*}{\operatorname{argmin}}\big[\|\mathbf{\Phi}^*\mathbf{w}^* - \mathbf{g}^*\|_2^2 + \frac{\lambda_1}{\sqrt{1 + \lambda_2}}\|\mathbf{w}^*\|_1\big]. \tag{2.12}$$

Here, $\mathbf{\Phi}^*$ and $\mathbf{g}^*$ are called augmented variables and are written as

$$\mathbf{\Phi}^*_{(N+M)\times M} = (1+\lambda_2)^{-1/2} \begin{bmatrix} \mathbf{\Phi} \\ \sqrt{\lambda_2}\mathbf{I} \end{bmatrix}, \quad \mathbf{g}^*_{(N+M)\times 1} = \begin{bmatrix} \mathbf{g} \\ \mathbf{0} \end{bmatrix}.$$

We call $\hat{\mathbf{w}} = \hat{\mathbf{w}}_{\text{elastic net}} = \sqrt{1+\lambda_2}\mathbf{w}^*$. Thus, Eq. (2.12) yields

$$\begin{aligned}
\hat{\mathbf{w}} &= \underset{\mathbf{w}}{\operatorname{argmin}} \left[\left\| \mathbf{\Phi}^* \frac{\mathbf{w}}{\sqrt{1+\lambda_2}} - \mathbf{g}^* \right\|_2^2 + \frac{\lambda_1}{\sqrt{1+\lambda_2}} \left\| \frac{\mathbf{w}}{\sqrt{1+\lambda_2}} \right\|_1 \right] \\
&= \underset{\mathbf{w}}{\operatorname{argmin}} \left[\mathbf{w^T} \left(\frac{\mathbf{\Phi}^{*\mathbf{T}}\mathbf{\Phi}^*}{1+\lambda_2} \right)\mathbf{w} - 2\frac{\mathbf{g}^{*\mathbf{T}}\mathbf{\Phi}^*}{\sqrt{1+\lambda_2}} + \mathbf{g}^{*\mathbf{T}}\mathbf{g}^* + \frac{\lambda_1\|\mathbf{w}\|_1}{1+\lambda_2} \right].
\end{aligned}$$

Using the augmented quantities $\mathbf{g}^*$ and $\mathbf{\Phi}^*$, we calculate

$$\begin{cases} \mathbf{\Phi}^{*\mathbf{T}}\mathbf{\Phi}^* &= \frac{\mathbf{\Phi^T}\mathbf{\Phi}+\lambda_2\mathbf{I}}{1+\lambda_2} \\ \mathbf{g}^{*\mathbf{T}}\mathbf{\Phi}^* &= \frac{\mathbf{g^T}\mathbf{\Phi}}{\sqrt{1+\lambda_2}} \\ \mathbf{g}^{*\mathbf{T}}\mathbf{g}^* &= \mathbf{g^T}\mathbf{g} \end{cases}.$$

Finally, we obtain an alternative formulation that brings out the relationship between the original naive elastic net regularization problem as

$$\begin{aligned}
\hat{\mathbf{w}} &= \underset{\mathbf{w}}{\operatorname{argmin}} \left[\frac{1}{1+\lambda_2} \left\{ \mathbf{w^T} \left(\frac{\mathbf{\Phi^T}\mathbf{\Phi}+\lambda_2\mathbf{I}}{1+\lambda_2} \right)\mathbf{w} - 2\mathbf{g^T}\mathbf{\Phi}\mathbf{w} + \lambda_1\|\mathbf{w}\|_1 \right\} + \mathbf{g^T}\mathbf{g} \right] \\
&= \underset{\mathbf{w}}{\operatorname{argmin}} \left[\mathbf{w^T} \left(\frac{\mathbf{\Phi^T}\mathbf{\Phi}+\lambda_2\mathbf{I}}{1+\lambda_2} \right)\mathbf{w} - 2\mathbf{g^T}\mathbf{\Phi}\mathbf{w} + \lambda_1\|\mathbf{w}\|_1 \right].
\end{aligned} \tag{2.13}$$

Various approaches have been developed to solve the optimization problem given in Eq. (2.13). From Eq. (2.12), it is apparent that the elastic net problem is equivalent to L1 regularization for the augmented data $\mathbf{g}^*$ and $\mathbf{\Phi}^*$ when λ_2 is fixed. Therefore, solving the optimization problem for EN-regularization can be done efficiently with the same methods used for L1 regularization. We note, however, that this computation is rather slow when $M \gg N$.

Elastic net regularization can also be realized by using proximal gradient methods (PGEN). Like for PGL, we first consider a target function for elastic net regularization

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}} \left[\frac{1}{2}\|\mathbf{\Phi}\mathscr{W}^*\mathscr{W}\mathbf{w} - \mathbf{g}\|_2^2 + \lambda_1\|\mathscr{W}\mathbf{w}\|_1 + \frac{\lambda_2}{2}\|\mathscr{W}\mathbf{w}\|_2^2 \right]. \tag{2.14}$$

We define a function $d(\mathbf{w}) = \lambda_1\|\mathscr{W}\mathbf{w}\|_1 + (\lambda_2/2)\|\mathscr{W}\mathbf{w}\|_2^2$. Then, the iteration equation for PGEN reads

$$\mathbf{w}_{j+1} = \underset{\mathbf{w}}{\operatorname{argmin}}[\tau d(\mathbf{w}) + \frac{1}{2}\|\mathbf{k}_j - \mathbf{w}\|^2] =: \operatorname{prox}_{\tau d(\mathbf{w})}(\mathbf{k}_j).$$

This equation has the same form as Eq. (2.10) with the different definition of $d(\mathbf{w})$. Finally, using a wavelet transformation, the iteration becomes [168]

$$\mathbf{w}_{j+1} = \frac{1}{1+\tau\lambda_2}\mathscr{W}^* S\big[\mathscr{W}(\mathbf{w}_j - \tau\mathbf{\Phi^T}(\mathbf{\Phi}\mathbf{w}_j - \mathbf{g})); \tau\lambda_1\big].$$

Here, the step size needs to fulfill the condition $\tau < 2/\|\mathbf{\Phi^T}\mathbf{\Phi}\|_2$ for a stable iteration. The iterations are stopped when the convergence conditions are satisfied, for example, $\|\mathbf{w}_{j+1} - \mathbf{w}_j\|_2/\|\mathbf{w}_j\|_2\% < 10^{-8}\%$.

In this thesis, we also employ the robust convex optimization solver package CVX [196, 197] to solve the EN regularization problem given in Eq. (2.13). We provide a simple Matlab interface code to demonstrate the use of CVX in Appendix A.

2.2.4 Choice of regularization parameters

When we employ regularization methods, the choice of a proper regularization parameter is central. Currently, there are three standard methods available for identification of proper regularization parameters, namely the L-curve criterion, the generalized cross-validation, and the quasi-optimality criterion. For the L-curve criterion, a log-log plot of residual norms $\log \|\mathbf{\Phi}\mathbf{w}_\lambda - \mathbf{g}\|_2$ and solution norms $\log \|\mathbf{w}_\lambda\|_p$ [198] is employed to select the proper regularization parameters. In double-logarithmic scale, the resulting curve often has a characteristic L-shaped appearance with a distinct corner separating the vertical and the horizontal parts of the curve, see an example for L2 regularization in Fig. 2.3 (a). The optimal regularization parameters λ are selected at the corner with maximum curvature [199]. At this point, a balance is reached between strong variation of the residual norm and strong variation of the solution norm. The L-curve criterion is a popular graphical approach to select the proper regularization parameters. It was early on employed for L2 regularization [198] and later it was used for L1- [158] and EN regularization [200].
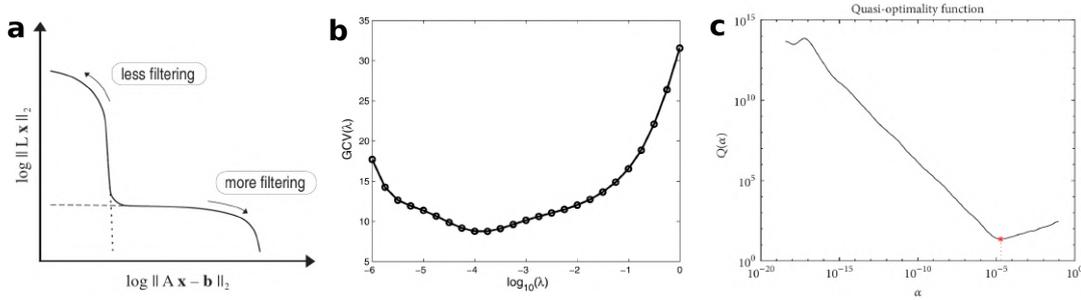
Figure 2.3: Three different approaches to select the L2 regularization parameter. (a) The L-curve criterion suggests to select the optimal parameter at the corner of a double-logarithmic plot of the solution resiudal norm vs the penalty norm. (b) For generalized cross validation, one selects the optimal regularization parameter at the minimum of a target function. (c) For the quasi-optimality criterion, the optimal parameter is selected at the minimum of the QOC function. (Figure (a) taken from P. C. Hansen, 2007 [62]. Figure (b) taken from J. O. Ramsay et al., 2009 [201]. Figure (c) taken from J. Dong et al., 2018 [63].)

The generalized cross-validation (GCV) criterion is also a popular method for choosing the regularization parameters. In case of the GCV for L2 regularization, the idea is simply to choose the optimal parameter such that the residual norm is equal to or smaller than a priori fixed upper bound δ_e with residual norm $\|e\|_2 = \|\mathbf{g} - \mathbf{g}^{\text{exact}}\|_2$ where is $\mathbf{g}^{\text{exact}} = \mathbf{\Phi}\hat{\mathbf{w}}$, i.e.

$$\|\mathbf{\Phi}\mathbf{w}_{\lambda_2} - \mathbf{g}\|_2 = \delta_e, \text{ where } \|e\|_2 \leq \delta_e.$$

Where $\mathbf{w}_{\lambda_2}$ is a result of L2 regularization. Using the *compensated discrepancy principle*, the above equation can be rewritten as $\|\mathbf{\Phi}\mathbf{w}_{\lambda_2} - \mathbf{g}\|_2 = \sigma_0(\text{trace}(\mathbf{I}_N - \mathbf{\Phi}\mathbf{\Phi}^{-1}))^{1/2}$, where σ_0 is a parameter [199]. With GCV, one seeks to minimize the predictive mean-square error δ_e. Since $\mathbf{g}^{\text{exact}}$ is unknown, the GCV approach works instead with the minimization of a GCV function, which is written as $f_{\text{GCV}} = \|\mathbf{\Phi}\mathbf{w}_{\lambda_2} - \mathbf{g}\|_2^2 / (\text{trace}(\mathbf{I}_N - \mathbf{\Phi}\mathbf{\Phi}^{-1}))^2$. Figure 2.3 (b) shows

an example where the GCV criterion is employed to select L2 regularization parameter [201]. The GCV approach is also used for other regularization methods, for example, L1- [202] and EN regularization [55].

Another approach for finding proper regularization parameters is the quasi-optimality criterion, which is derived from an error-estimate based on a *heuristic parameter choice rule* [203]. The goal of this approach is to minimize the estimated error between the regularized solution $\mathbf{w}_{\lambda_2}$ and $\hat{\mathbf{w}}$. In Ref. [204], the authors suggest an error estimate for L2 regularization is given by $\|\hat{\mathbf{w}} - \mathbf{w}_{\lambda_2}\|_2 \approx (\mathbf{g^T}(\mathbf{\Phi\Phi^T} + \lambda_2\mathbf{I})^{-4}\mathbf{\Phi\Phi^T}\mathbf{g})^{1/2}$. The minimization of the estimate leads to the problem of minimizing the function $f_{\mathrm{QOC}} = \|\lambda_2 d\mathbf{w}_{\lambda_2}/d(\lambda_2)\|_2$, where $d\mathbf{w}_{\lambda_2}/d(\lambda_2) = -(\mathbf{\Phi^T\Phi} + \lambda_2\mathbf{I})^{-1}\mathbf{w}_{\lambda_2}$ [199]. The optimal L2 regularization parameter is selected by minimization of f_{QOC}, see an example in Fig. 2.3 (c).

Although these methods can be employed to help with the selection of the regularization parameters, they are of limited use for real data. For example, for the L-curve criterion, multiple inflection points can often appear and the points are hard to localize precisely on the employed logarithmic scales. Moreover, the L-curve criterion does not behave consistently in the asymptotic limit of large system sizes or when the data is strongly corrupted by noise [205, 206]. The disadvantage of GCV criterion is that the optimal regularization parameter is typically not identified precisely because the GCV-function can have a very flat minimum [207] or have several minimum points [208]. The quasi-optimality criterion also has the drawback that the function f_{QOC} often has several minimum points [208]. Consequently, in practical applications involving the inversion of ill-conditioned matrices, such as Traction Force Microscopy, regularization parameters are often chosen by visual inspection of the resulting solution. This procedure lacks objectivity and significantly biases any conclusions drawn from later analysis of the solution. Note that this problem is not specific to regularization, but the issue of distinguishing between noise and "real" signal appears generally with any type of method if the data is processed in any way to reduce noise. Since the identification of optimal regularization parameters is usually a complex task, we will next introduce a Bayesian framework that is instrumental for this purpose.

2.3 Bayesian methods

In Bayesian methodology, all quantities in the formula $\mathbf{g} = \mathbf{\Phi w} + \mathbf{s}$ are viewed as random variables. The inverse and ill-posed problem in Eq. (2.1) can be connected to Bayes' theorem. Bayes' rule is written as

$$p(\mathbf{w}|\mathbf{g}) = \frac{p(\mathbf{g}|\mathbf{w})p(\mathbf{w})}{p(\mathbf{g})} \propto p(\mathbf{g}|\mathbf{w})p(\mathbf{w}), \qquad (2.15)$$

where $p(\mathbf{w}|\mathbf{g})$, $p(\mathbf{g}|\mathbf{w})$, $p(\mathbf{w})$, and $p(\mathbf{g})$ respectively represent the posterior, likelihood, prior, and evidence. The three subtasks of the Bayesian approach to solve inverse ill-posed problems are: (1) Based on all the prior information of the unknown solution $\mathbf{w}$, find a prior distribution $p(\mathbf{w})$ that can judiciously reflect this prior information. (2) Find the likelihood distribution $p(\mathbf{g}|\mathbf{w})$ which describes the interrelation between the unknown $\mathbf{w}$ and observation $\mathbf{g}$, for example, given the distribution of the noise. (3) Develop approaches to explore the posterior distribution function [209].

2.3.1 Bayesian regularization

Following above three subtasks, we assume that the unobserved data $\mathbf{w}$ is a sample drawn from a prior distribution which we denote by $p(\mathbf{w}|\alpha)$ with an undetermined parameter α. The function $p(\mathbf{w}|\alpha)$ describes any knowledge about the distribution of $\mathbf{w}$. For reasons that will become clear below, we will assume that the prior distribution for the $M \times 1$ vector $\mathbf{w}$ is a Gaussian [68]

$$p(\mathbf{w}|\alpha) = \frac{\exp[-\alpha E_{\mathrm{w}}(\mathbf{w})]}{Z_{\mathrm{w}}}, \tag{2.16}$$

where $Z_{\mathrm{w}} = (2\pi/\alpha)^{M/2}$ and $E_{\mathrm{w}} = \mathbf{w}^{\mathbf{T}}\mathbf{w}/2$. The acquisition noise $\mathbf{s}$ presents a source of randomness. Typically, $\mathbf{s}$ is assumed to be drawn from a zero-mean Gaussian with unknown variance $1/\beta$ [68, 78, 210, 211]. In the language of Bayesian statistics Eq. (2.15), the probability distribution $p(\mathbf{g}|\mathbf{w}, \beta)$ is called the likelihood function and determines the probability to measure a particular vector $\mathbf{g}$ given a vector $\mathbf{w}$. Since the noise is a Gaussian, the likelihood function is written as

$$p(\mathbf{g}|\mathbf{w}, \beta) = \frac{\exp[-\beta E_{\mathrm{g}}(\mathbf{g}|\mathbf{w})]}{Z_{\mathrm{g}}}, \tag{2.17}$$

where $E_{\mathrm{g}}(\mathbf{g}|\mathbf{w}) = (\mathbf{\Phi}\mathbf{w} - \mathbf{g})^{\mathbf{T}}(\mathbf{\Phi}\mathbf{w} - \mathbf{g})/2$ and $Z_{\mathrm{g}} = (2\pi/\beta)^{N/2}$. N is, as above, the size of the vector for the measurement data $\mathbf{g}$. The likelihood function $p(\mathbf{g}|\mathbf{w}, \beta)$ describes a situation that is exactly the reverse of the experimental situation, where we are looking for the probability of having $\mathbf{w}$ given measurement $\mathbf{g}$. This situation is described by the posterior distribution $p(\mathbf{w}|\mathbf{g})$ and can be related to the likelihood via Bayes' rule

$$P(\mathbf{w}|\mathbf{g}, \alpha, \beta) = \frac{p(\mathbf{g}|\mathbf{w}, \beta)p(\mathbf{w}|\alpha)}{p(\mathbf{g}|\alpha, \beta)} = \frac{\exp[-\beta E_{\mathrm{g}}(\mathbf{g}|\mathbf{w})]}{Z_{\mathrm{g}}} \frac{p(\mathbf{w}|\alpha)}{p(\mathbf{g}|\alpha, \beta)}. \tag{2.18}$$

Here, the Gaussian prior and Gaussian likelihood have already been made use of in Bayes' rule. Based on these assumptions, we prove that the posterior is also a Gaussian distribution and further we show the exact formula of the Gaussian posterior.

Given a general random variable $\mathcal{W}$ obeyed $\mathcal{W} \sim \mathcal{N}(w_0, \Gamma_{\mathrm{pr}})$ with mean w_0 and covariance Γ_{pr}, and the random noise $\mathcal{S}$ yielded $\mathcal{S} \sim \mathcal{N}(s_0, \Gamma_{\mathrm{noise}})$ with its mean s_0 and covariance Γ_{noise}, the posterior related to the likelihood and prior is also a Gaussian distribution.

According to the given information, we can write the following equations for the prior and likelihood

$$p(\mathbf{w}) = \left(\frac{1}{2\pi|\Gamma_{\mathrm{pr}}|}\right)^{M/2} \exp\left(-\frac{1}{2}(\mathbf{w} - w_0)^{\mathbf{T}}\Gamma_{\mathrm{pr}}^{-1}(\mathbf{w} - w_0)\right)$$

$$\propto \exp\left(-\frac{1}{2}(\mathbf{w} - w_0)^{\mathbf{T}}\Gamma_{\mathrm{pr}}^{-1}(\mathbf{w} - w_0)\right),$$

$$p(\mathbf{s}) = \left(\frac{1}{2\pi|\Gamma_{\mathrm{noise}}|}\right)^{M/2} \exp\left(-\frac{1}{2}(\mathbf{s} - s_0)^{\mathbf{T}}\Gamma_{\mathrm{noise}}^{-1}(\mathbf{s} - s_0)\right)$$

$$\propto \exp\left(-\frac{1}{2}(\mathbf{s} - s_0)^{\mathbf{T}}\Gamma_{\mathrm{noise}}^{-1}(\mathbf{s} - s_0)\right).$$

According to the Bayes' rule in Eq. (2.18), the posterior distribution of $\mathbf{w}$ conditioned on $\mathbf{g}$ is given by

$$p(\mathbf{w}|\mathbf{g}) \propto p(\mathbf{w})p(\mathbf{\Phi}\mathbf{w} - \mathbf{g})$$

$$\propto \exp\left(-\frac{1}{2}(\mathbf{w} - w_0)^{\mathbf{T}}\Gamma_{\mathrm{pr}}^{-1}(\mathbf{w} - w_0) - \frac{1}{2}(\mathbf{\Phi}\mathbf{w} - \mathbf{g} - s_0)^{\mathbf{T}}\Gamma_{\mathrm{noise}}^{-1}(\mathbf{\Phi}\mathbf{w} - \mathbf{g} - s_0)\right).$$

Since $\mathcal{W}$ and $\mathcal{S}$ are Gaussian distributions, their expectation values are well-defined and given by

$$E\left\{ \begin{bmatrix} \mathcal{W} \\ \mathcal{G} \end{bmatrix} \right\} = \begin{bmatrix} w_0 \\ g_0 \end{bmatrix},$$

where $\mathcal{G}$ is a random variable for $\mathbf{g}$ with its mean $g_0 = \mathbf{\Phi} w_0 + s_0$. We have two covariances for the random variables $\mathcal{W}$ and $\mathcal{G}$

$$E\{(\mathcal{W} - w_0)(\mathcal{W} - w_0)^{\mathbf{T}}\} = \Gamma_{\mathrm{pr}},$$

$$E\{(\mathcal{G} - g_0)(\mathcal{G} - g_0)^{\mathbf{T}}\} = E\{(\mathbf{\Phi}(\mathcal{W} - w_0) + (\mathcal{S} - s_0))(\mathbf{\Phi}(\mathcal{W} - w_0) + (\mathcal{S} - s_0))^{\mathbf{T}}\}$$
$$= \mathbf{\Phi}\Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}} + \Gamma_{\mathrm{noise}}.$$

Furthermore, we have a covariance for both random variables $\mathcal{W}$ and $\mathcal{G}$ as

$$E\{(\mathcal{W} - w_0)(\mathcal{G} - g_0)^{\mathbf{T}}\} = E\{(\mathcal{W} - w_0)(\mathbf{\Phi}(\mathcal{W} - w_0) + (\mathcal{S} - s_0))^{\mathbf{T}}\} = \Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}}.$$

Therefore, we obtain

$$\mathrm{cov}\begin{bmatrix} \mathcal{W} \\ \mathcal{G} \end{bmatrix} = E\left\{ \begin{bmatrix} \mathcal{W} - w_0 \\ \mathcal{G} - g_0 \end{bmatrix} \begin{bmatrix} \mathcal{W} - w_0 \\ \mathcal{G} - g_0 \end{bmatrix}^{\mathbf{T}} \right\} = \begin{bmatrix} \Gamma_{\mathrm{pr}} & \Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}} \\ \mathbf{\Phi}\Gamma_{\mathrm{pr}} & \mathbf{\Phi}\Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}} + \Gamma_{\mathrm{noise}} \end{bmatrix}.$$

Hence, the joint probability density function for the random variables $\mathcal{W}$ and $\mathcal{G}$ is of the form

$$p(\mathbf{w}, \mathbf{g}) \propto \left(-\frac{1}{2} \begin{bmatrix} \mathbf{w} - w_0 \\ \mathbf{g} - g_0 \end{bmatrix}^{\mathbf{T}} \begin{bmatrix} \Gamma_{\mathrm{pr}} & \Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}} \\ \mathbf{\Phi}\Gamma_{\mathrm{pr}} & \mathbf{\Phi}\Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}} + \Gamma_{\mathrm{noise}} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{w} - w_0 \\ \mathbf{g} - g_0 \end{bmatrix} \right)$$
$$= \left(-\frac{1}{2} \begin{bmatrix} \mathbf{w} - w_0 \\ \mathbf{g} - g_0 \end{bmatrix}^{\mathbf{T}} \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{w} - w_0 \\ \mathbf{g} - g_0 \end{bmatrix} \right).$$

In Ref. [209], the authors prove that the conditioned probability $p(\mathbf{w}|\mathbf{g})$ can be obtained from the above joint probability $p(\mathbf{w}, \mathbf{g})$. The conditioned probability is written as

$$p(\mathbf{w}|\mathbf{g}) \propto \exp\left(-\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu})^{\mathbf{T}}\mathbf{\Sigma}^{-1}(\mathbf{w} - \boldsymbol{\mu}) \right), \tag{2.19}$$

where

$$\boldsymbol{\mu} = w_0 + \Gamma_{12}\Gamma_{22}^{-1}(\mathbf{g} - g_0) = w_0 + \Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}}(\mathbf{\Phi}\Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}} + \Gamma_{\mathrm{noise}})^{-1}(\mathbf{g} - \mathbf{\Phi}w_0 - s_0),$$

$$\mathbf{\Sigma} = \Gamma_{11} - \Gamma_{12}\Gamma_{22}^{-1}\Gamma_{21} = \Gamma_{\mathrm{pr}} - \Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}}(\mathbf{\Phi}\Gamma_{\mathrm{pr}}\mathbf{\Phi}^{\mathbf{T}} + \Gamma_{\mathrm{noise}})^{-1}\mathbf{\Phi}\Gamma_{\mathrm{pr}}.$$

Equation (2.19) links the Gaussian posterior to the given Gaussian prior and likelihood. The mean and covariance of the Gaussian posterior are $\boldsymbol{\mu}$ and $\mathbf{\Sigma}$. According to the Woodbury identity, we can simplify the mean and covariance as

$$\boldsymbol{\mu} = (\Gamma_{\mathrm{pr}}^{-1} + \mathbf{\Phi}^{\mathbf{T}}\Gamma_{\mathrm{noise}}^{-1}\mathbf{\Phi})^{-1}(\mathbf{\Phi}^{\mathbf{T}}\Gamma_{\mathrm{noise}}^{-1}(\mathbf{g} - s_0) + \Gamma_{\mathrm{pr}}^{-1}w_0),$$

$$\mathbf{\Sigma} = (\Gamma_{\mathrm{pr}}^{-1} + \mathbf{\Phi}^{\mathbf{T}}\Gamma_{\mathrm{noise}}^{-1}\mathbf{\Phi})^{-1}.$$

In Refs. [68, 69], a special case is that the distribution of noise is assumed to be a Gaussian with zero mean $s_0 = 0$ and its variance $\sigma^2 = 1/\beta$ and the distribution of prior is assumed

a Gaussian with zero mean $w_0 = 0$ and its variances $(1/\alpha_1, \ldots, 1/\alpha_M)$. Therefore, we can further simplify the mean and covariance

$$\boldsymbol{\Sigma} = (\beta\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi} + \mathbf{A})^{-1},$$

$$\begin{aligned}
\boldsymbol{\mu} &= (\mathbf{A} + \boldsymbol{\Phi}^{\mathbf{T}}\beta\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^{\mathbf{T}}\beta\mathbf{g} \\
&= \beta(\beta\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi} + \mathbf{A})^{-1}\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{g} \\
&= \beta\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{g},
\end{aligned}$$

where $\mathbf{A} = \mathrm{diag}(\alpha_1, \ldots, \alpha_M)$. This Gaussian posterior $p(\mathbf{w}|\mathbf{g}, \boldsymbol{\alpha}, \beta) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ has been employed, for example, in Refs. [67, 69, 70].

In Eq. (2.18), the marginal likelihood $p(\mathbf{g}|\alpha, \beta)$ is the overall probability of finding the observed $\mathbf{g}$ when the known $\mathbf{w}$ distributions are integrated out. Thus, $p(\mathbf{g}|\alpha, \beta)$ is also called evidence for the model with $\{\alpha, \beta, \mathbf{g}\}$.

Assuming that α and β are known constants, one can maximize the posterior probability $P(\mathbf{w}|\mathbf{g}, \alpha, \beta)$ with respect to $\mathbf{w}$. The resulting solution then satisfies $\mathbf{w}_{\mathrm{MP}} = \mathrm{argmin}_{\mathbf{w}}\left[\beta\|\boldsymbol{\Phi}\mathbf{w} - \mathbf{g}\|_2^2 + \alpha\|\mathbf{w}\|_2^2\right]$, which is exactly the formula employed for L2 regularization, Eq. (2.3), if the parameters α and β are related to the L2 regularization parameter as $\lambda_2 = \alpha/\beta$ [136]. Thus, our choice of a Gaussian prior is justified if we intend to perform an L2 regularization. Other popular choices for replacing Eq. (2.16) as prior are the Laplace distribution $p(\mathbf{w}|\theta) = (\theta/2)\exp(-\theta/2\|\mathbf{w}\|_1)$ [72], and a product of a Gaussian and a Laplace distribution [77]. Using these priors, we would have found the formulas corresponding to L1 regularization and EN regularization, respectively. Thus, regularization is equivalent to maximizing the posterior probability of a measurement assuming fixed, known parameters of the prior distributions.

However, α and β can also be treated as variables whose values can be determined by maximizing their probability $p(\alpha, \beta|\mathbf{g}) = p(\mathbf{g}|\alpha, \beta)p(\alpha, \beta)/p(\mathbf{g}) \sim p(\mathbf{g}|\alpha, \beta)$, where we assume a uniform prior $p(\alpha, \beta)$ and we can omit the marginal probability $p(\mathbf{g})$ since it plays no role for the optimization. To calculate the evidence $p(\mathbf{g}|\alpha, \beta)$, we need to integrate out $\mathbf{w}$ in the posterior given in Eq. (2.18). Due to the Gaussian probabilities, this integration can be done analytically. For convenience, we expand the integrand around the most probable value $\mathbf{w}_{\mathrm{MP}}$. On defining $K(\mathbf{w}) \equiv \alpha E_{\mathrm{w}}(\mathbf{w}) + \beta E_{\mathrm{g}}(\mathbf{w})$ and its Hessian, $\mathbf{A} \equiv \nabla\nabla K(\mathbf{w}) = \alpha\mathbf{I} + \beta\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi}$, we expand as $\mathbf{K}(\mathbf{w}) \approx \mathbf{K}(\mathbf{w}_{\mathrm{MP}}) + (\mathbf{w} - \mathbf{w}_{\mathrm{MP}})^{\mathbf{T}}\mathbf{A}(\mathbf{w} - \mathbf{w}_{\mathrm{MP}})/2$. We have proved the Gaussian posterior and can write the posterior

$$P(\mathbf{w}|\mathbf{g}, \alpha, \beta) = \frac{\exp[-K(\mathbf{f})]}{Z_{\mathrm{K}}},$$

where $Z_K = \int \exp[-K(\mathbf{w})] d^M\mathbf{w}$ is the normalization. We employ multivariate Gaussian integrals and the evidence can be calculated as follows

$$
\begin{aligned}
p(\mathbf{g}|\alpha,\beta) &= \frac{Z_K}{Z_g Z_w} \\
&= \frac{1}{Z_g Z_w} \int_{\mathbf{w}} \exp[-K(\mathbf{w})] d^M\mathbf{w} \\
&= \frac{1}{Z_g Z_w} \int_{\mathbf{w}} \exp[-K(\mathbf{w}_{\mathrm{MP}}) - \frac{1}{2}(\mathbf{w}-\mathbf{w}_{\mathrm{MP}})^{\mathbf{T}}\mathbf{A}(\mathbf{w}-\mathbf{w}_{\mathrm{MP}})] d^M\mathbf{w} \\
&= \frac{(2\pi)^{M/2}(\det\mathbf{A})^{-1/2}}{Z_g Z_w} \exp[-\mathbf{K}(\mathbf{w}_{\mathrm{MP}})].
\end{aligned}
\tag{2.20}
$$

The logarithm of the evidence is

$$
\begin{aligned}
\log p(\mathbf{g}|\alpha,\beta) = &-\alpha E_w(\mathbf{w}_{\mathrm{MP}}) - \beta E_g(\mathbf{w}_{\mathrm{MP}}) - \frac{1}{2}\log(\det\mathbf{A}) \\
&+ \frac{M}{2}\log\alpha + \frac{N}{2}\log\beta - \frac{N}{2}\log(2\pi).
\end{aligned}
\tag{2.21}
$$

The right hand side of this equation is a typical example for the target functions employed in Bayesian analysis, for example in the context of data fitting [68]. Here, numerical calculation of $\log(\det\mathbf{A})$ requires some care. We employ here a Cholesky decomposition of the positive matrix $\mathbf{A} = \mathbf{L}\mathbf{L}^T$ yielding $\log(\det\mathbf{A}) = \log(\det(\mathbf{L}\mathbf{L}^{\mathbf{T}})) = 2\log\Pi_i L_{ii} = 2\Sigma_i\log(L_{ii})$.

The logarithmic evidence, Eq. (2.21), assumes a maximum at those parameters $\hat{\alpha}$ and $\hat{\beta}$ that are most likely associated with the measurement data $\mathbf{g}$. The implicit equations resulting from maximizing Eq. (2.21) read [68] $2\alpha E_w^{\mathrm{MP}} = 2n - \hat{\alpha}\mathrm{Tr}\mathbf{A}^{-1}$ and $2\hat{\beta}E_g = 2m - 2n + \hat{\alpha}\mathrm{Tr}\mathbf{A}^{-1}$. Since $\mathbf{w}_{\mathrm{MP}}$ and $\mathbf{A}$ depend on α and β, these equations need to be solved numerically. Once $\hat{\alpha}$ and $\hat{\beta}$ are determined, the optimal L2 regularization parameter follows as $\hat{\lambda}_2 = \hat{\alpha}/\hat{\beta}$.

We employ two approaches for determining the numerical values of α and β. In the first approach, called Bayesian L2 regularization (BL2), we estimate the inverse noise variance β directly from the data calculating the variance of the measured data $\mathbf{g}$, for instance in TFM, the displacements in spatial regions are very far away from any cell. Thus, in BL2 only α is determined through maximization of Eq. (2.21). In the second approach, termed advanced Bayesian L2 regularization (ABL2), we solve directly for α and β, which requires an increased computational effort. For both approaches, it is imperative to standardize the data to adjust its spread in different dimensions. For the observed vector $\mathbf{g}$ of length N, we first subtract the mean $\tilde{\mathbf{g}} = \mathbf{g} - \mathbf{1}_N\bar{g}$ with $\bar{g} = \sum_{i=1}^{N} g_i/N$. Next, we calculate the mean and standard deviation for all columns of the matrix $\mathbf{\Phi}$ as $\bar{\Phi}_j = 1/N\sum_{i=1}^{N}\Phi_{ij}$ and $\eta_j = (\sum_{i=1}^{N}(\Phi_{ij} - \bar{\Phi}_j)^2/(N-1))^{1/2}$. Thus, we can define a problem matrix where each column is normalized by its spread $\tilde{\Phi}_{ij} = (\Phi_{ij} - \bar{\Phi}_j)/\eta_j$. The standardized problem therefore reads $\tilde{g}_i = \tilde{\Phi}_{ij}\tilde{w}_j$, which yields $w_i = \tilde{w}_i/\eta_i$.

2.3.2 Sparse Bayesian learning

Kernel-based approaches have been widely employed for machine learning in recent years, for example, a classical kernel method, the "support vector machine" (SVM) [212–214], which is a sparse linearly-parameterised model. Tipping combined this SVM model with Bayesian regularization [68] and developed a sparse Bayesian learning algorithm to realize the relevance vector machine (RVM) [69]. For sake of completeness of this exposition, we

will in this section derive the mathematical formulation of the RVM. This sparse Bayesian learning is based on the same likelihood function as Bayesian regularization

$$p(\mathbf{g}|\mathbf{w},\beta) = \prod_{i=1}^{N} \mathcal{N}(\mathbf{\Phi}\mathbf{w},\beta^{-1}) = \left(\frac{2\pi}{\beta}\right)^{-\frac{N}{2}} \exp\left\{-\frac{\beta}{2}\|\mathbf{g} - \mathbf{\Phi}\mathbf{w}\|_2^2\right\}, \qquad (2.22)$$

where $\mathbf{\Phi}$ is a $N \times M$ matrix which can be written as a column form $\mathbf{\Phi} = [\phi_1, \ldots, \phi_i, \ldots, \phi_M]$. This form of the likelihood function is motivated by the assumption that the errors $\mathbf{s}$ obey a Gaussian distribution with variance $\sigma^2 = 1/\beta$. The likelihood function is complemented by an assumed Gaussian prior over the parameters which takes the form

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{i=1}^{M} \left(\frac{2\pi}{\alpha_i}\right)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\alpha_i \mathbf{w}^2\right\} = (2\pi)^{-\frac{M}{2}} \prod_{i=1}^{M} \alpha_i^{\frac{1}{2}} \exp\left\{-\frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{A}\mathbf{w}\right\}. \qquad (2.23)$$

Here, the hyper-parameters are $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_M]^{\mathbf{T}}$. Each parameter α_i individually controls the strength of the prior over its associated weight and these weights are responsible for the ultimate sparsity of the model. Based on these assumptions, the Gaussian posterior in Equation (2.19), can be written with Bayes' rule

$$P(\mathbf{w}|\mathbf{g},\boldsymbol{\alpha},\beta) = \frac{p(\mathbf{g}|\mathbf{w},\beta)p(\mathbf{w}|\boldsymbol{\alpha})}{p(\mathbf{g}|\boldsymbol{\alpha},\beta)}.$$

$p(\mathbf{g}|\boldsymbol{\alpha},\beta) = \int p(\mathbf{g}|\mathbf{w},\beta)p(\mathbf{w}|\boldsymbol{\alpha})d\mathbf{w}$ is called evidence, or marginal likelihood. Next, we will re-derive the following central formula

$$\begin{aligned} p(\mathbf{g}|\boldsymbol{\alpha},\beta) &= \int p(\mathbf{g}|\mathbf{w},\beta)p(\mathbf{w}|\boldsymbol{\alpha})d\mathbf{w} \\ &= (2\pi)^{-\frac{N}{2}}\left|\frac{1}{\beta}\mathbf{I} + \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathbf{T}}\right|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\mathbf{g}^{\mathbf{T}}\left(\frac{1}{\beta}\mathbf{I} + \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathbf{T}}\right)^{-1}\mathbf{g}\right\} \qquad (2.24) \\ &= (2\pi)^{-\frac{N}{2}}|\mathbf{C}|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\mathbf{g}^{\mathbf{T}}\mathbf{C}^{-1}\mathbf{g}\right\}, \end{aligned}$$

where $\mathbf{C} = 1/\beta\mathbf{I} + \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathbf{T}}$. This formula means that given a random variable $\mathcal{W}$ with $\mathcal{W} \sim \mathcal{N}(0,\boldsymbol{\alpha}^{-1})$ and a random noise $\mathcal{S}$ with $\mathcal{S} \sim \mathcal{N}(0,\beta^{-1})$, the evidence resulting from margnizalizing out $\mathcal{W}$ with given prior and likelihood, is also a Gaussian.

We will employ the following three equalities for the calculation

$$\begin{cases} \boldsymbol{\Sigma}^{-1} = \beta\mathbf{\Phi}^{\mathbf{T}}\mathbf{\Phi} + \mathbf{A} \\ \boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma} = \boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1} = \mathbf{I} \\ \boldsymbol{\mu}^{\mathbf{T}} = \beta\mathbf{g}^{\mathbf{T}}\mathbf{\Phi}\boldsymbol{\Sigma}^{\mathbf{T}} = \beta\mathbf{g}^{\mathbf{T}}\mathbf{\Phi}\boldsymbol{\Sigma} \end{cases},$$

where the last equation can be obtained because the covariance $\boldsymbol{\Sigma}$ is a symmetric matrix. According to Eq. (2.18), the integral evidence can be calculated as

$$
\begin{aligned}
p(\mathbf{g}|\boldsymbol{\alpha},\beta) &= \int p(\mathbf{g}|\mathbf{w},\beta)p(\mathbf{w}|\boldsymbol{\alpha})d\mathbf{w} \\
&= \int \Big(\frac{2\pi}{\beta}\Big)^{-\frac{N}{2}} \exp\Big\{ -\frac{\beta}{2}\|\mathbf{g}-\boldsymbol{\Phi}\mathbf{w}\|_2^2\Big\}(2\pi)^{-\frac{M}{2}}\prod_{i=1}^{M}\alpha_i^{\frac{1}{2}}\exp\Big\{ -\frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{A}\mathbf{w}\Big\}d\mathbf{w} \\
&= \Big(\frac{\beta}{2\pi}\Big)^{\frac{N}{2}}\Big(\frac{1}{2\pi}\Big)^{\frac{M}{2}}\prod_{i=1}^{M}\alpha_i^{\frac{1}{2}}\int \exp\Big\{ -\Big[\frac{\beta}{2}\|\mathbf{g}-\boldsymbol{\Phi}\mathbf{w}\|_2^2+\frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{A}\mathbf{w}\Big]\Big\}d\mathbf{w} \\
&= \Big(\frac{\beta}{2\pi}\Big)^{\frac{N}{2}}\Big(\frac{1}{2\pi}\Big)^{\frac{M}{2}}\prod_{i=1}^{M}\alpha_i^{\frac{1}{2}}\int \exp\{-Y(\mathbf{w})\}d\mathbf{w},
\end{aligned}
\tag{2.25}
$$

where we define a new function $Y(\mathbf{w}) = \beta\|\mathbf{g}-\boldsymbol{\Phi}\mathbf{w}\|_2^2/2 + \mathbf{w}^{\mathbf{T}}\mathbf{A}\mathbf{w}/2$. In order to obtain a complete integral with respect to $\mathbf{w}$, we need to separate $\mathbf{w}$ and $\mathbf{g}$ in Eq. (2.25) and therefore $Y(\mathbf{w})$ can be simplified as

$$
\begin{aligned}
Y(\mathbf{w}) &= \frac{\beta}{2}\|\mathbf{g}-\boldsymbol{\Phi}\mathbf{w}\|_2^2 + \frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{A}\mathbf{w} \\
&= \frac{1}{2}(\beta\mathbf{g}^{\mathbf{T}}\mathbf{g} - 2\beta\mathbf{g}^{\mathbf{T}}\boldsymbol{\Phi}\mathbf{w} + \beta\mathbf{w}^{\mathbf{T}}\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi}\mathbf{w} + \mathbf{w}^{\mathbf{T}}\mathbf{A}\mathbf{w}) \\
&= \frac{1}{2}(\beta\mathbf{g}^{\mathbf{T}}\mathbf{g} - 2\beta\mathbf{g}^{\mathbf{T}}\boldsymbol{\Phi}\mathbf{w} + \mathbf{w}^{\mathbf{T}}(\beta\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi}+\mathbf{A})\mathbf{w}) \\
&= \frac{1}{2}(\beta\mathbf{g}^{\mathbf{T}}\mathbf{g} - 2\beta\mathbf{g}^{\mathbf{T}}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}\mathbf{w} + \mathbf{w}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{w}) \\
&= \frac{1}{2}(\beta\mathbf{g}^{\mathbf{T}}\mathbf{g} - 2\boldsymbol{\mu}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{w} + \mathbf{w}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{w} + \boldsymbol{\mu}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \boldsymbol{\mu}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}) \\
&= \frac{1}{2}(\beta\mathbf{g}^{\mathbf{T}}\mathbf{g} - \boldsymbol{\mu}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}) + \frac{1}{2}(\boldsymbol{\mu}-\mathbf{w})^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\mathbf{w}) \\
&= R(\mathbf{g}) + \frac{1}{2}(\boldsymbol{\mu}-\mathbf{w})^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\mathbf{w}),
\end{aligned}
$$

where $R(\mathbf{g}) = (\beta\mathbf{g}^{\mathbf{T}}\mathbf{g} - \boldsymbol{\mu}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu})/2$. According to the formula for multivariate Gaussian integrals $\int \exp\{-Y(\mathbf{w})\}d\mathbf{w} = \exp\{-R(\mathbf{g})\}(2\pi)^{M/2}|\boldsymbol{\Sigma}|^{1/2}$, Eq. (2.25) can be simplified as

$$
p(\mathbf{g}|\boldsymbol{\alpha},\beta) = \Big(\frac{\beta}{2\pi}\Big)^{\frac{N}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}\prod_{i=1}^{M}\alpha_i^{\frac{1}{2}}\exp\{-R(\mathbf{g})\}.
$$

This result can be simplified by making use of the following two expressions

$$
\begin{aligned}
\Big(\frac{\beta}{2\pi}\Big)^{\frac{N}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}\prod_{i=1}^{M}\alpha_i^{\frac{1}{2}} &= (2\pi)^{-\frac{N}{2}}\beta^{\frac{N}{2}}\big|(\beta\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi}+\mathbf{A})^{-1}\big|^{\frac{1}{2}}\prod_{i=1}^{M}\alpha_i^{\frac{1}{2}} \\
&= (2\pi)^{-\frac{N}{2}}\beta^{\frac{N}{2}}\prod_{i=1}^{M}\alpha_i^{-\frac{1}{2}}\big|\big[\beta\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}}+\mathbf{I}\big]^{-1}\big|^{\frac{1}{2}}\prod_{i=1}^{M}\alpha_i^{\frac{1}{2}} \\
&= (2\pi)^{-\frac{N}{2}}\Big|\frac{1}{\beta}\mathbf{I}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}}\Big|^{-\frac{1}{2}},
\end{aligned}
$$

$$-R(\mathbf{g}) = -\frac{1}{2}(\beta \mathbf{g}^{\mathbf{T}}\mathbf{g} - \boldsymbol{\mu}^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu})$$

$$= -\frac{1}{2}(\beta \mathbf{g}^{\mathbf{T}}\mathbf{g} - \beta \mathbf{g}^{\mathbf{T}}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}\beta\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{g})$$

$$= -\frac{1}{2}\beta \mathbf{g}^{\mathbf{T}}(\mathbf{I} - \beta\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathbf{T}})\mathbf{g}$$

$$= -\frac{1}{2}\beta \mathbf{g}^{\mathbf{T}}(\mathbf{I} - \beta\boldsymbol{\Phi}(\beta\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi} + \mathbf{A})^{-1}\boldsymbol{\Phi}^{\mathbf{T}})\mathbf{g}$$

$$= -\frac{1}{2}\beta \mathbf{g}^{\mathbf{T}}(\mathbf{I} - \beta\boldsymbol{\Phi}\mathbf{A}^{-1}(\mathbf{I} - \boldsymbol{\Phi}^{\mathbf{T}}\beta\boldsymbol{\Phi}\mathbf{A}^{-1})^{-1}\boldsymbol{\Phi}^{\mathbf{T}})\mathbf{g}$$

$$= -\frac{1}{2}\beta \mathbf{g}^{\mathbf{T}}(\mathbf{I} + \beta\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}})^{-1}\mathbf{g}$$

$$= -\frac{1}{2} \mathbf{g}^{\mathbf{T}}\left(\frac{1}{\beta}\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}}\right)^{-1}\mathbf{g}.$$

Overall, Eq. (2.25) can be written as

$$p(\mathbf{g}|\boldsymbol{\alpha},\beta) = (2\pi)^{-\frac{N}{2}}\left|\frac{1}{\beta}\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}}\right|^{-\frac{1}{2}}\exp\left\{-\frac{1}{2}\mathbf{g}^{\mathbf{T}}\left(\frac{1}{\beta}\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}}\right)^{-1}\mathbf{g}\right\}.$$

This equation shows that the evidence (or marginal likelihood) is a Gaussian distribution, where the mean is zero and the covariance is $(1/\beta\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}})^{-1}$

The logarithm of the evidence $\mathcal{L}(\boldsymbol{\alpha})$ is written as

$$\mathcal{L}(\boldsymbol{\alpha}) = \log p(\mathbf{g}|\boldsymbol{\alpha},\beta) = -\frac{1}{2}[N\log(2\pi) + \log|\mathbf{C}| + \mathbf{g}^{\mathbf{T}}\mathbf{C}^{-1}\mathbf{g}]. \tag{2.26}$$

In Bayesian regularization, we have employed a similar form of the logarithmic evidence to calculate the optimal regularization parameter, see Eq. Eq. (2.21). Here, for sparse Bayesian learning [69, 70], the key property is that the inferred result vectors are exceedingly sparse. The majority of entries in the solution vector are automatically set to zero which is controlled by the marginal likelihood maximization related to the set of vectors $(\phi_1, \ldots, \phi_M)$. Next, we will present details of the mathematical procedure that allows one to obtain the sparse $\mathbf{w}$. The covariance of the evidence $\mathbf{C}$ in Eq. (2.26) can be decomposed as follows

$$\mathbf{C} = \sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathbf{T}}$$

$$= \sigma^2\mathbf{I} + \sum_{m\neq i}\alpha_m^{-1}\phi_m\phi_m^{\mathbf{T}} + \alpha_i^{-1}\phi_i\phi_i^{\mathbf{T}}$$

$$= \mathbf{C}_{-i} + \alpha_i^{-1}\phi_i\phi_i^{\mathbf{T}},$$

where $\mathbf{C}_{-i}$ does not contain the i vector from in $(\phi_1, \ldots, \phi_i, \ldots, \phi_M)$. Using properties of determinants and inverse matrices, we can exactly obtain

$$|\mathbf{C}| = |\mathbf{C}_{-i}| \ |1 + \alpha_i^{-1}\phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\phi_i|,$$

$$\mathbf{C}^{-1} = \mathbf{C}_{-i}^{-1} - \frac{\mathbf{C}_{-i}^{-1}\phi_i\phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}}{\alpha_i + \phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\phi_i}.$$

Therefore, the logarithm of the evidence in Eq. (2.26) can be written

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{\alpha}) &= -\frac{1}{2}[N\log(2\pi) + \log|\mathbf{C}| + \mathbf{g}^{\mathbf{T}}\mathbf{C}^{-1}\mathbf{g}] \\
&= -\frac{1}{2}\Bigg[N\log(2\pi) + \log|\mathbf{C}_{-i}| + \mathbf{g}^{\mathbf{T}}\mathbf{C}_{-1}^{-1}\mathbf{g} \\
&\qquad - \log\alpha_i + \log(\alpha_i + \phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\phi_i) - \frac{(\phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\mathbf{g})^2}{\alpha_i + \phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\phi_i} \Bigg] \\
&= \mathcal{L}(\boldsymbol{\alpha}_{-i}) + -\frac{1}{2}\Bigg[\log\alpha_i - \log(\alpha_i + s_i) + \frac{q_i^2}{\alpha_i + s_i} \Bigg] \\
&= \mathcal{L}(\boldsymbol{\alpha}_{-i}) + \ell(\alpha_i),
\end{aligned}
\tag{2.27}
$$

where we define $s_i = \phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\phi_i$ and $q_i = \phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\mathbf{g}$. In Ref. [70], s_i and q_i are respectively called the "sparsity factor" and "quality factor", which depend on the basis vector ϕ_i. In order to find each optimal α_i at the maximal marginal likelihood, we directly do the partial derivative of Eq. (2.27) respect with α_i. Since $\mathcal{L}(\boldsymbol{\alpha}_{-i})$ does not depend on α_i, we can easily calculate derivatives as

$$
\begin{aligned}
\frac{\partial\mathcal{L}(\boldsymbol{\alpha})}{\partial\alpha_i} = \frac{\partial\ell(\alpha_i)}{\partial\alpha_i} &= \frac{1}{2}\Bigg[\frac{1}{\alpha_i} - \frac{1}{\alpha_i + \phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\phi_i} - \frac{(\phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\mathbf{g})^2}{(\alpha_i + \phi_i^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\phi_i)^2} \Bigg] \\
&= \frac{1}{2}\Bigg[\frac{1}{\alpha_i} - \frac{1}{\alpha_i + s_i} - \frac{(q_i)^2}{(\alpha_i + q_i)^2} \Bigg] \\
&= \frac{\alpha_i^{-1}s_i^2 - (q_i^2 - s_i)}{2(\alpha_i + s_i)^2}.
\end{aligned}
$$

In order to obtain the optimal α_i at the maximal marginal likelihood, we calculate $\partial\mathcal{L}(\boldsymbol{\alpha})/\partial\alpha_i = 0$ and find

$$
\begin{cases}
\alpha_i = \frac{s_i^2}{q_i^2 - s_i}, & \text{if } q_i^2 > s_i, \\
\alpha_i = \infty, & \text{if } q_i^2 \le s_i.
\end{cases}
$$

s_i and q_i has the term $\mathbf{C}_{-i}^{-1}$, which excludes the current utilized function ϕ_i. In fact, these values can be simply updated with the following equations

$$
\begin{cases}
S_j = \phi_j^{\mathbf{T}}\mathbf{C}^{-1}\phi_j, \\
Q_j = \phi_j^{\mathbf{T}}\mathbf{C}^{-1}\mathbf{g},
\end{cases}
$$

and they are related to s_i and q_i as

$$
\begin{cases}
s_j = \frac{\alpha_j S_j}{\alpha_j - S_j}, \\
q_j = \frac{\alpha_j Q_j}{\alpha_j - S_j}.
\end{cases}
$$

When $\alpha_j = \infty$, $s_i = S_j$ and $q_i = Q_j$. Using linear algebra, we find $\mathbf{C}^{-1} = \mathbf{B} - \mathbf{B}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{B}$, where $\mathbf{B} = \sigma^2\mathbf{I}$. Thus, S_j and Q_j are written as

$$
\begin{cases}
S_j = \phi_j^{\mathbf{T}}\mathbf{B}\phi_j - \phi_j^{\mathbf{T}}\mathbf{B}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{B}\phi_j, \\
Q_j = \phi_j^{\mathbf{T}}\mathbf{B}\mathbf{g} - \phi_j^{\mathbf{T}}\mathbf{B}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{B}\mathbf{g}.
\end{cases}
$$

At this point, we have collected all the required formulas to update the s_i, q_i, μ, and $\boldsymbol{\Sigma}$ for the sparse Bayesian learning method. In Ref. [70], a fast RVM algorithm is developed to update

these quantities in a very efficient way. The details of this fast algorithm are as follows.

(1) Adding a new function ϕ_i,

$$2\Delta\mathcal{L} = \frac{Q_i^2 - S_i}{S_i} + \log\frac{S_i}{Q_i^2},$$

$$\tilde{\boldsymbol{\Sigma}} = \begin{bmatrix} \boldsymbol{\Sigma} + \beta^2\boldsymbol{\Sigma}_{ii}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\mathbf{T}\phi_i\phi_i^\mathbf{T}\boldsymbol{\Phi}\boldsymbol{\Sigma} & -\beta^2\boldsymbol{\Sigma}_{ii}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\mathbf{T}\phi_i \\ -\beta^2\boldsymbol{\Sigma}_{ii}(\boldsymbol{\Sigma}\boldsymbol{\Phi}^\mathbf{T}\phi_i)^\mathbf{T} & \boldsymbol{\Sigma}_{ii} \end{bmatrix},$$

$$\tilde{\mu} = \begin{bmatrix} \mu - \mu_i\beta\boldsymbol{\Sigma}\boldsymbol{\Phi}^\mathbf{T}\phi_i \\ \mu_i \end{bmatrix},$$

$$\tilde{S}_j = S_j - \boldsymbol{\Sigma}_{ii}(\beta\phi_j^\mathbf{T}\mathbf{e}_i)^2,$$

$$\tilde{Q}_j = Q_j - \mu_i(\beta\phi_j^\mathbf{T}\mathbf{e}_i),$$

where $\boldsymbol{\Sigma}_{ii} = (\alpha_i + S_i)^{-1}$, $\mu_i = \Sigma_{ii}Q_i$. We define $\mathbf{e}_i = \phi_i - \beta\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\mathbf{T}\phi_i$.

(2) Re-estimating a function ϕ_i,

$$2\Delta\mathcal{L} = \frac{Q_i^2}{S_i + [\tilde{\alpha}_i^{-1} - \alpha_i^{-1}]^{-1}} - \log 1 + S_i[\tilde{\alpha}_i^{-1} - \alpha_i^{-1}],$$

$$\tilde{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma} - \kappa_h\boldsymbol{\Sigma}_h\boldsymbol{\Sigma}_h^\mathbf{T},$$

$$\tilde{\mu} = \mu - \kappa_h\mu_h\boldsymbol{\Sigma}_h,$$

$$\tilde{S}_j = S_j + \kappa_h(\beta\boldsymbol{\Sigma}_h^\mathbf{T}\boldsymbol{\Phi}^\mathbf{T}\phi_j)^2,$$

$$\tilde{Q}_j = Q_j + \kappa_h\mu_h(\beta\boldsymbol{\Sigma}_h^\mathbf{T}\boldsymbol{\Phi}^\mathbf{T}\phi_j),$$

where $\kappa_h = (\boldsymbol{\Sigma}_{hh} + (\tilde{\alpha}_i - \alpha_i)^{-1})^{-1}$ and $\boldsymbol{\Sigma}_h$ is the h-th column of $\boldsymbol{\Sigma}$.

(3) Deleting a function ϕ_i,

$$2\Delta\mathcal{L} = \frac{Q_i^2}{S_i - \alpha_i} - \log\left(1 - \frac{S_i}{\alpha_i}\right),$$

$$\tilde{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma} - \frac{1}{\boldsymbol{\Sigma}_{hh}}\boldsymbol{\Sigma}_h\boldsymbol{\Sigma}^\mathbf{T}_h,$$

$$\tilde{\mu} = \mu - \frac{\mu_h}{\boldsymbol{\Sigma}_{hh}}\boldsymbol{\Sigma}_h,$$

$$\tilde{S}_j = S_j - \frac{1}{\boldsymbol{\Sigma}_{hh}}(\beta\boldsymbol{\Sigma}_h^\mathbf{T}\boldsymbol{\Phi}^\mathbf{T}\phi_j)^2,$$

$$\tilde{Q}_j = Q_j + \frac{\mu_h}{\boldsymbol{\Sigma}_{hh}}(\beta\boldsymbol{\Sigma}_h^\mathbf{T}\boldsymbol{\Phi}^\mathbf{T}\phi_j).$$

Using this relevance vector machine and an iterative approach, the sparse solution $\hat{\mathbf{w}}$, being the mean of the posterior $\boldsymbol{\mu}$, can be efficiently calculated. This method is what we call the sparse Bayesian learning approach [69, 70].

The sparse Bayesian learning problem also can be solved with Markov chain Monte Carlo (MCMC). In Ref. [75], the authors employ the MCMC for Bayesian L1-, L2-, and EN regularization in hierarchical models. A graphical representation of these three Bayesian models is provided in Fig. 2.4. The MCMC approach requires more computation time than variational methods.

The sparse learning method is also employed in the field of compressive sensing, which aims at efficiently acquiring and reconstructing signals by finding solutions to underdetermined, linear systems based on the assumption that the solutions are sparse. Here, the
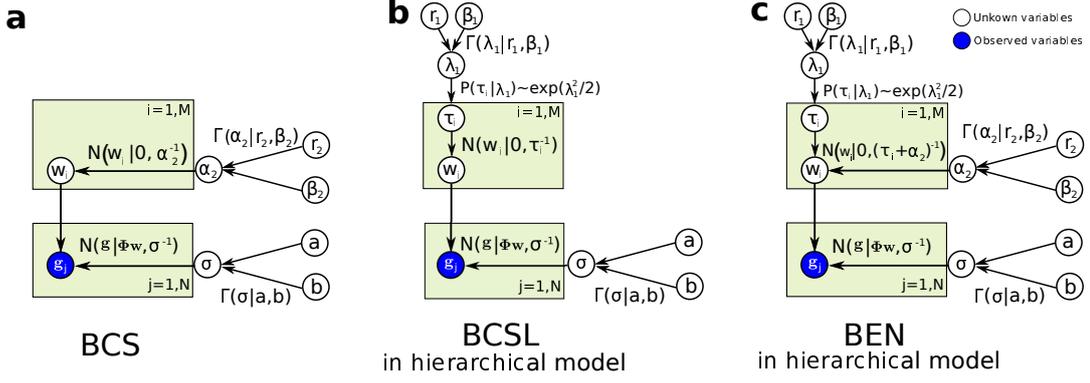
Figure 2.4: Graphical model of the sparse Bayesian approaches. (a) Bayesian compressive sensing (BCS) using the Gaussian prior. (b) Bayesian compressive sensing using the Laplace priors (BCSL) in hierarchical model. (c) Bayesian elastic net (BEN) using the elastic net prior in hierarchical model.

method is called Bayesian compressive sensing (BCS) [71]. The prior used for these approaches is a Gaussian distribution. Based on BCS, Bayesian compressive sensing using Laplace priors (BCSL) method has been proposed in Ref. [215]. Here, we briefly summarize the differences between BCS and BCSL. The Laplace prior with a parameter λ is written as $p(\mathbf{w}|\lambda) = \lambda/2 \exp(-\lambda/2\|\mathbf{w}\|_1)$. Since this prior is not conjugate to the likelihood in Eq. (2.22), an additional hyper-prior $\gamma_i\, p(\gamma_i|\lambda) = \Gamma(\gamma_i|1, \lambda/2)$ is employed, see Ref. [215]. Therefore, the Laplace prior can be rewritten as

$$p(\mathbf{w}|\lambda) = \int p(\mathbf{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}|\lambda)d\boldsymbol{\gamma} = \frac{\lambda^{N/2}}{2^N} \exp\left(-\sqrt{\lambda}\sum_{i=1} |w_i|\right),$$

where $p(\lambda|\nu) = \Gamma(\lambda|\nu/2, \nu/2)$ with a parameter ν. The posterior is again a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L)$, where the mean and covariance are respectively $\boldsymbol{\mu}_L = \beta\boldsymbol{\Sigma}_L\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{g}$ and $\boldsymbol{\Sigma} = (\beta\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi} + \boldsymbol{\Lambda})^{-1}$ with $\Lambda = \mathrm{diag}(1/\gamma_i)$. The evidence can be calculated by using a similar approach as above

$$p(\mathbf{g}|\boldsymbol{\gamma}, \beta, \lambda) = (2\pi)^{-\frac{N}{2}}|\mathbf{C}_L|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\mathbf{g}^{\mathbf{T}}\mathbf{C}_L^{-1}\mathbf{g}\right\}p(\boldsymbol{\gamma}|\lambda)p(\lambda)p(\beta),$$

where $\mathbf{C}_L = 1/\beta\mathbf{I} + \boldsymbol{\Phi}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}^{\mathbf{T}}$ and $p(\beta) = \Gamma(\beta|a, b)$. $p(\beta)$ is taken a non-informative prior with $a = b = 0$ [69]. Here, we find that the evidence is not a Gaussian distribution anymore. The logarithm of the evidence is written as

$$\mathcal{L}_L = -\frac{1}{2}\log\mathbf{C}_L - \frac{1}{2}\mathbf{g}^{\mathbf{T}}\mathbf{C}_L^{-1}\mathbf{g} + N\log\frac{\lambda}{2} - \frac{\lambda}{2}\sum_i \gamma_i + \frac{\nu}{2}\log\frac{\nu}{2}$$
$$- \log\Gamma(\frac{\nu}{2}) + (\frac{\nu}{2} - 1)\log\lambda - \frac{\nu}{2}\lambda + (a^\beta - 1)\log\beta - b^\beta\beta. \tag{2.28}$$

To update the hyper-parameters in the above equations, we take the derivative of the logarithmic evidence with respect to each hyper-parameter and set it to zero. Thus, we obtain

$$\lambda = \frac{N - 1 + \nu/2}{\sum_i \gamma_i/2 + \nu/2}, \tag{2.29a}$$

$$\log(\nu/2) + 1 - \psi(\nu/2) + \log(\lambda) - \lambda = 0. \tag{2.29b}$$

The logarithmic evidence is also decomposed in two parts of which only one depends on the vector ϕ_i.

$$\mathcal{L}_L(\boldsymbol{\gamma}) = -\frac{1}{2}\left[\log|\mathbf{C}_{-i}| + \mathbf{g}^{\mathbf{T}}\mathbf{C}_{-i}^{-1}\mathbf{g} + \frac{\nu}{2}\sum_{j\neq i}\gamma_j\right] + \frac{1}{2}\left[\log\frac{1}{1+\gamma_i s_i} + \frac{q_i^2\gamma_i}{1+\gamma_i s_i} - \lambda\gamma_i\right]$$

$$= \mathcal{L}(\boldsymbol{\gamma}_{-i}) + \ell(\gamma_i),$$

where $s_i = \phi_i^T\mathbf{C}_{-i}^{-1}\phi_i$ and $q_i = \phi_i^T\mathbf{C}_{-i}^{-1}\mathbf{g}$. In the fast algorithm, we only consider the second term, which depends on γ_i. Maximization of the evidence yields γ_i

$$\begin{cases} \gamma_i = \frac{-s_i(s_i+2\lambda)+s_i\sqrt{(s_i+2\lambda)^2-4\lambda(s_i-q_i^2+\lambda)}}{2\lambda s_i^2}, & \text{if } q_i^2 - s_i > \lambda, \\ \gamma_i = 0, & \text{otherwise.} \end{cases}$$

Here, $s_i = S_i/(1-\gamma_i S_i)$ and $q_i = Q_i/(1-\gamma_i S_i)$, where S_i and Q_i are calculated from the same formulas as in the BCS approach. The updates of the S_i, Q_i, μ and $\boldsymbol{\Sigma}$ are used as in BCS where the function ϕ_i can be efficiently added, deleted, and re-estimated to maximize the evidence function.

2.4 Summary

In this chapter, we discussed two complementary approaches for solving inverse problems in length, namely regularization and the Bayesian approach.

The former includes the L2-, L1- and Elastic Net regularization. These regularization methods require one to manually select one or two regularization parameters. A heuristic for choosing these regularization parameters is the L-curve criterion. The latter approaches mainly consist of Bayesian regularization and sparse Bayesian learning. In Bayesian regularization, the optimal regularization parameter can be automatically selected by maximizing an evidence function. Then, the reconstructed data is calculated by using the L2 regularization with the optimal parameter. Sparse Bayesian learning is based on priors that force most of the entries of the solution to zero. Here too, the optimal sparsity is determined by maximizing an evidence function. Sparse Bayesian learning algorithms employing a Laplace prior have been developed for compressive sensing. Algorithms based on iterative approximation of the evidence function are computationally efficient.

In the following, all these approaches will be tested for application in traction force microscopy, see Chapter 3 and 4. The sparse Bayesian learning methods will also be employed for another application, namely the identification of governing equations from measurement data, see Chapter 5.

Chapter 3

Traction force microscopy

3.1 Mechanical model

The basic assumption that is being made for traction force microscopy (TFM) is that cell-generated forces are exerted on the free surface of a substrate, as illustrated in Fig. 3.2 (c). Thus, these forces can be calculated from the displacements of the substrate by using a mechanical model. In the mechanical model, the substrate is usually assumed to be a homogeneous, isotropic, and linear elastic semi-infinite medium. It is required that the forces vanish at infinity. In such a case, the mechanical equilibrium model relating a continuous displacement field $U_i(\mathbf{x})$ to the traction force field $F_j(\mathbf{x}')$ on a two-dimensional ($\mathbf{x} = (x_1, x_2)$) surface of the gel is expressed as the integral equation [216]

$$U_i(\mathbf{x}) = \int_\Omega G_{ij}(\mathbf{x} - \mathbf{x}')F_j(\mathbf{x}')d\mathbf{x}', \tag{3.1}$$

where Ω denotes the whole surface of the substrate. The integrand contains the Green's function

$$G_{ij}(\mathbf{x}) = \frac{(1+\nu)}{\pi E}\left[\frac{1-\nu}{r}\delta_{ij} + \frac{\nu x_i x_j}{r^3}\right] = \frac{(1+\nu)}{\pi E}\begin{pmatrix}(1-\nu)r^2 + \nu x_1^2 & \nu x_1 x_2 \\ \nu x_1 x_2 & (1-\nu)r^2 + \nu x_2^2\end{pmatrix}, \tag{3.2}$$

where E and ν represent Young's modulus and Poisson's ratio, respectively. We also write $r^2 = |\mathbf{x}|^2$ and δ_{ij} is the Kronecker delta function. The model in Eq. (3.1) can also be extended to include the case that the displacements are not measured at the surface of the gel but below the surface in a plane with constant coordinate z. The corresponding Green's function is given as [136]

$$G_{ij}(x_1, x_2, z) = \frac{1+\nu}{2\pi E}\left[\frac{2(1-\nu)r + z}{r(r+z)}\delta_{ij} + \frac{(2r(\nu r + z) + z^2)x_i x_j}{r^3(r+z)^2}\right], \tag{3.3}$$

where z is a vertical distance from the surface of gel, as shown in Fig. 3.2 (c). In both cases, the continuous forces and displacements in Eq. (3.1) can be calculated with discretized fields. A very popular and fast approach to solve Eq. (3.1) is called *Fourier-transform traction cytometry* (FTTC) [37, 155, 217]. FTTC is a global discrete method, see Chapter 4. However, Eq. (3.1) can also be solved by using a local discrete approach, for example, the *boundary element method* (BEM) [37, 153, 154] and *finite element method* (FEM) [148–152]. In this chapter, we mainly focus on the BEM.

For the BEM, the traction forces F_j are approximated by a shape function on discrete nodes. To discretize the traction field, we introduce a rectangular, regular mesh with meshsize w. The position of the mesh nodes is denoted by $\mathbf{y}_k$. The distance between any point in the traction field at $\mathbf{x}'$ and the mesh nodes at $\mathbf{y}_l$ is abbreviated for simplicity as $(d_1, d_2) = \mathbf{x}' - \mathbf{y}_l$. The traction at $\mathbf{x}'$ is assumed to be a linear combination of the traction values at the four surrounding nodes. Hence, we introduce a pyramidal shape function, located at every node $\mathbf{y}_l$, which is equal to 1 on the local node and drops linearly to 0 in all adjacent nodes. The shape function scales the traction magnitude at $\mathbf{x}'$ depending on the distances (d_1, d_2) as

$$
\begin{aligned}
h(d_1, d_2) =\,& \theta(d_1)\theta(d_2)(1 - d_1/w)(1 - d_2/l)\theta(w - d_1)\theta(w - d_2) \\
& + \theta(-d_1)\theta(d_2)(1 + d_1/l)(1 - d_2/l)\theta(w + d_1)\theta(w - d_2) \\
& + \theta(d_1)\theta(-d_2)(1 - d_1/l)(1 + d_2/l)\theta(w - d_1)\theta(w + d_2) \\
& + \theta(-d_1)\theta(-d_2)(1 + d_1/l)(1 + d_2/l)\theta(w + d_1)\theta(w + d_2),
\end{aligned}
$$

where $\theta(x)$ is written as

$$
\theta(x) := \left\{ \begin{array}{l} 1, x \geq 0 \\ 0, x < 0 \end{array} \right. .
$$

With the help of the shape function, the continuous traction field is linearly approximated as $F_j(\mathbf{x}') = \sum_l h(\mathbf{x}' - \mathbf{y}_l) f_{j,l}$. Thus, Eq. (3.1) becomes

$$
U_i(\mathbf{x}) = \sum_{j,i} \int_\Omega G_{ij}(\mathbf{x} - \mathbf{x}') h(\mathbf{x}' - \mathbf{y}_l) d^2\mathbf{x}' f_{j,l} = \sum_{j,i} \mathbf{M}'_{j,l}(\mathbf{x}') f_{j,l}. \tag{3.4}
$$

The coefficient matrix $\mathbf{M}$ is constructed from the $\mathbf{M}'_{j,l}(\mathbf{x}')$ by inserting discrete measurement positions for $\mathbf{x}$. This approach is also suitable for the case of three-dimensional displacement fields or the alternative Green' function in Eq. (3.3). We can write Eq. (3.4) in a matrix-notation as $\mathbf{u} = \mathbf{M}\mathbf{f}$. In the two-dimensional case, we write the discrete displacement field $\mathbf{u}$ as a $2m \times 1$ vector, where m is the number of discretization nodes. The discrete traction field $\mathbf{f}$ is a $2n \times 1$ vector, where n is the number of nodes at which traction is prescribed. Due to the fact that measurement noise always exists in the displacement field $\mathbf{u}$ for TFM, we extend the equation as

$$
\mathbf{u} = \mathbf{M}\mathbf{f} + \mathbf{s}, \tag{3.5}
$$

where $\mathbf{s}$ is the linear acquisition noise. Given $\mathbf{u}$ and $\mathbf{M}$, Eq. (3.5) allows one to calculate f. This becomes an inverse ill-posed problem because the condition number of the matrix $\mathbf{M}$ is usually large.

In Fourier space, Eq. (3.1) also can be calculated by using the convolution and shift theorem and a Fourier-transformed shape function. We employ a Fourier transformation with wave vector $\mathbf{k} = (k_1, k_2)$. The shape function $h(d_1, d_2)$ is transformed into Fourier space and becomes

$$
\tilde{h}(k_1, k_2) = \text{sinc}^2(k_1)\text{sinc}^2(k_2).
$$

The Green's function reads in Fourier space

$$
\tilde{G}_{ij}(k_1, k_2) = \frac{2(1 + \nu)}{Ek^3} \begin{bmatrix} (1 - \nu)k^2 + \nu k_2^2 & -\nu k_1 k_2 \\ -\nu k_1 k_2 & (1 - \nu)k^2 + \nu k_1^2 \end{bmatrix}, \tag{3.6}
$$

where $k^2 = k_1^2 + k_2^2$. The Green's function depending on the vertical coordinate z becomes in Fourier space [136]

$$\tilde{G}_{ij}(k_1, k_2, z) = \frac{2(1+\nu)e^{-z\sqrt{k_1^2+k_2^2}}}{E(k_1^2 + k_2^2)^{3/2}} \left[(k_1^2 + k_2^2)\delta_{ij} - k_i k_j \left(\nu + \frac{z\sqrt{k_1^2 + k_2^2}}{2} \right) \right]. \qquad (3.7)$$

Using the convolution and shift theorem, we rewrite Eq. (3.4) to obtain a discrete equation in Fourier space as

$$U_i(\mathbf{x}) = \sum_{j=1}^{n} \mathrm{Ft}^{-1}(\tilde{G}_{ij}(\mathbf{k})\tilde{h}(\mathbf{k})e^{-i\mathbf{k}\mathbf{y}_l})f_{j,l}. \qquad (3.8)$$

This equation is an alternative expression for Eq. (3.4) but it provides a fast way to calculate the coefficient matrix $\mathbf{M}$ via Fast Fourier Transform. Note however that the sharp corners in the pyramidal shape functions can result in artifacts due to the finite cutoff on high spatial wave vectors.
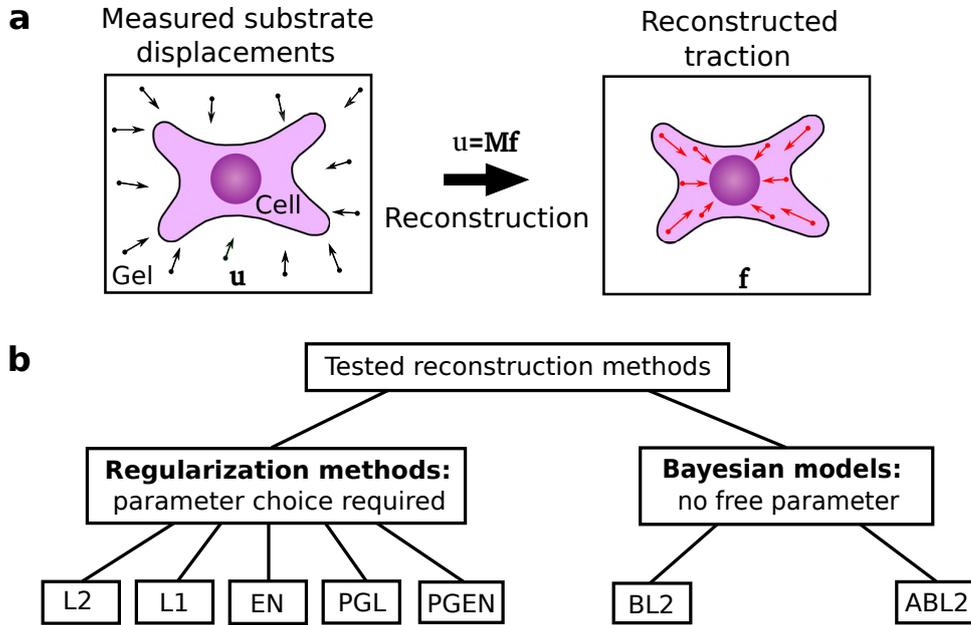
Figure 3.1: Schematic representation of a typical traction force microscopy (TFM) setup and different reconstruction methods for TFM. (a) Cells are plated on a planar gel substrate containing fiducial markers. Tracking the markers allows to infer the deformations $\mathbf{u}$ in the surrounding of the cell. These deformations are linearly related to the cellular traction forces $\mathbf{f}$. The problem of calculating traction $\mathbf{f}$ from displacement $\mathbf{u}$ is associated with inverting an ill-conditioned matrix $\mathbf{M}$. This problem can be solved with different reconstruction methods. (b) In this work, we test five regularization methods for traction reconstruction: L2 regularization (L2), L1 regularization (L1), EN regularization (EN), Proximal Gradient Lasso (PGL) and Proximal Gradient Elastic Net (PGEN). Furthermore, we develop two Bayesian approaches that do not have any free parameters, namely Bayesian L2 regularization (BL2) and Advanced Bayesian L2 regularization (ABL2).

In this chapter, we systematically compare various approaches for solving Eq. (3.5), the inverse ill-posed problem in TFM. We compare the performance of seven approaches, illustrated as a schematic diagram in Fig. 3.1. First, we test various regularization methods. Among these are the classical TFM methods, L1- and L2 regularization and previously untested methods from computer vision, namely *Elastic Net* (EN) regularization, *Proximal*

Gradient Lasso (PGL), and *Proximal Gradient Elastic Net* (PGEN). We find that the new EN regularization scheme has a substantially improved accuracy as compared to previous approaches but requires considerable extra computational cost. Secondly, we seek to establish Bayesian models that can automatically perform an optimal regularization of the data. Initial tests indicate that different freely available Bayesian hierarchical models are of little use for TFM since a large number of hidden variables, even when used with sparsity priors, does not enforce sufficient data faithfulness. Instead, we find that the simplest-possible Bayesian models with global priors yield robust results that can be interpreted as optimal L2 regularization. We study two variants of this algorithm: *Bayesian L2 regularization* (BL2), where the magnitude of the noise in the displacement data must be measured separately, and *Advanced Bayesian L2 regularization* (ABL2) which requires no extra input. We test the Bayesian methods using artificial data and real experimental data. Our results suggest that BL2 is not only very robust but also superior to classical L2 regularization when measurement noise is large. Most importantly, BL2 automatically determines the degree of regularization, which removes subjectiveness from the result. This advance is particularly relevant for in-detail comparison of cells in different conditions, where the varying signal-to-noise ratio previously made an unambiguous comparison challenging.

3.2 Generation of artificial test data and experimental procedures

3.2.1 Artificial test data

To quantitatively compare the performance of different reconstruction methods, we require artificial data with exactly known traction force and displacements. To generate synthetic test data that is completely known, we employ explicit formulas for the displacement field around a single circular traction patch. We calculate displacements on the surface of the substrate and in a plane below the surface, given a circular traction patch on the substrate surface, as illustrated in Fig. 3.2 (c). Test images containing multiple patches can be assembled by adding the displacement vectors resulting from the individual patches. The analytical solution is calculated for a patch with radius R located at the origin. We employ polar coordinates, $\mathbf{r} = (r\cos\theta, r\sin\theta)$, with r and θ being the radial and angular coordinate. The traction patch is described as

$$f(r) = \begin{cases} f_0, & |r| < R, \\ 0, & |r| \geq R. \end{cases} \tag{3.9}$$

The traction vector is given by $f_x = f(r)\cos\gamma$ and $f_y = f(r)\sin\gamma$, where γ is the angle between the x axis and the direction of the traction. Fourier transformation yields

$$\tilde{f}(\rho) = \int_0^{2\pi} \int_0^\infty f_0 e^{i\rho r \cos(\phi-\theta)} r \mathrm{d}r \mathrm{d}\theta = 2\pi R f_0 J_1(\rho R)/\rho,$$

where $J_1(\rho R)$ is a Bessel function, ρ is a radial wave vector, and ϕ is an angle. The traction vector in Fourier space is thus $\tilde{f}_x(\rho) = \tilde{f}(\rho)\cos\gamma$ and $\tilde{f}_y(\rho) = \tilde{f}(\rho)\sin\gamma$. Given such force patch, next, we calculate an analytical 2D displacement on the surface of the gel and the displacement in the gel.

3.2.1.1 A analytical calculation of displacements around a circular traction patch

In the two-dimensional case ($z = 0$), the Fourier space Green's function in polar coordinates reads

$$\tilde{G}_{ij}(\rho, \phi) = \frac{2(1+\nu)}{E\rho} \begin{bmatrix} (1-\nu) + \nu \sin^2 \phi & -\nu \sin \phi \cos \phi \\ -\nu \sin \phi \cos \phi & (1-\nu) + \nu \cos^2 \phi \end{bmatrix}.$$

According to the convolution theorem from Eq. (3.1), the displacement field in Fourier space becomes

$$\begin{bmatrix} \tilde{u}_x \\ \tilde{u}_y \end{bmatrix} = \frac{2(1+\nu)}{E\rho} \begin{bmatrix} (1-\nu) + \nu \sin^2 \phi & -\nu \sin \phi \cos \phi \\ -\nu \sin \phi \cos \phi & (1-\nu) + \nu \cos^2 \phi \end{bmatrix} \begin{bmatrix} \tilde{f}_x \\ \tilde{f}_y \end{bmatrix}.$$

Thus, the displacements in real space can be calculated through inverse Fourier transformation

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \left(\frac{1}{2\pi} \right)^2 \int_0^{2\pi} \int_0^\infty \begin{bmatrix} \tilde{u}_x \\ \tilde{u}_y \end{bmatrix} e^{-i\rho r \cos(\phi - \theta)} \rho \, d\rho \, d\phi$$

The result can be simplified as

$$u_x(r, \theta) = \frac{R(1+\nu)}{\pi E} \left[\left((1-\nu)N_1 + \nu N_2 \right) f_0 \cos \gamma - \nu N_3 f_0 \sin \gamma \right],$$

$$u_y(r, \theta) = \frac{R(1+\nu)}{\pi E} \left[-\nu N_3 f_0 \cos \gamma + \left((1-\nu)N_1 + \nu N_4 \right) f_0 \sin \gamma \right].$$

For the functions N_1 to N_4, we have for the inner region where $r < R$ and $\xi_1 = r^2/R^2$

$$N_1 = 4E_0(\xi_1),$$

$$N_2 = \frac{4\cos(2\theta)\left((r^2 + R^2)E_0(\xi_1) + (r^2 - R^2)K_0(\xi) \right)}{3r^2} + 4\sin^2 \theta E_0(\xi_1),$$

$$N_3 = \frac{2\sin(2\theta)\left((r^2 - 2R^2)E_0(\xi_1) + 2(R^2 - r^2)K_0(\xi_1) \right)}{3r^2},$$

$$N_4 = 4\cos^2 \theta E_0(\xi_1) - \frac{4\cos(2\theta)\left((r^2 + R^2)E_0(\xi_1) + (r^2 - R^2)K_0(\xi_1) \right)}{3r^2}.$$

Here, E_0 is complete elliptic integral of the first kind and K_0 is complete integral of the second kind. For the outer region where $r > R$ and $\xi_2 = R^2/r^2$, we have

$$N_1 = \frac{4\left(r^2 E_0(\xi_2) + (R^2 - r^2)K_0(\xi_2) \right)}{rR},$$

$$N_2 = \frac{\left(6r^2 - 2(r^2 - 2R^2)\cos(2\theta) \right)E_0(\xi_2) + 2(r^2 - R^2)(\cos(2\theta) - 3)K_0(\xi_2)}{3rR},$$

$$N_3 = \frac{2\sin(2\theta)\left((r^2 - 2R^2)E_0(\xi_2) + (R^2 - r^2)K_0(\xi_2) \right)}{3rR},$$

$$N_4 = \frac{\left(6r^2 + 2(r^2 - 2R^2)\cos(2\theta) \right)E_0(\xi_2) - 2(r^2 - R^2)(\cos(2\theta) + 3)K_0(\xi_2)}{3rR}.$$

In this thesis, we call this completely analytical solution of displacements around a circular traction patch, 2D theory.
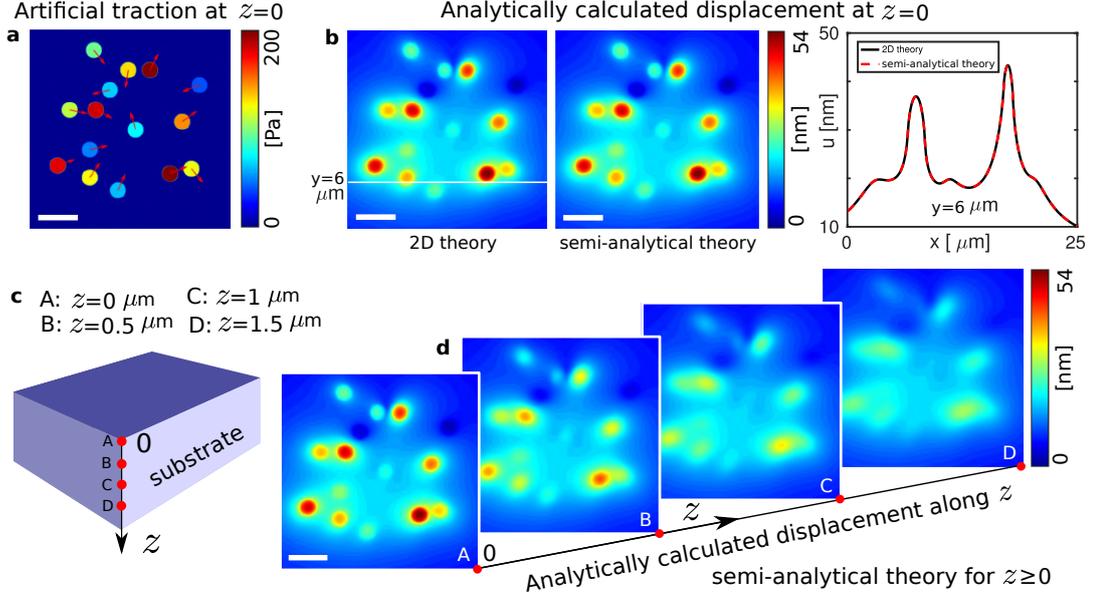
Figure 3.2: Analytically calculated displacement field around 15 circular traction patches. (a) 15 circular traction patches are randomly distributed on a $25\,\mu\mathrm{m} \times 25\,\mu\mathrm{m}$ region. The maximal traction force is $200\,\mathrm{Pa}$. The traction forces are applied on the free surface of substrate ($z = 0$), shown in (c). (b) The displacements on the surface of substrate $z = 0$ are calculated by using 2D theory (Left) and semi-analytical theory. (Middle). A local comparison from 2D theory and semi-analytical theory is obtained at $y = 6\,\mu\mathrm{m}$ (Right). (d) The displacement fields are obtained by using semi-analytical theory along z at point A, B, C, and D, for z equal to $0\,\mu\mathrm{m}$, $0.5\,\mu\mathrm{m}$, $1\,\mu\mathrm{m}$, and $1.5\,\mu\mathrm{m}$, respectively. Space bar: $5\,\mu\mathrm{m}$.

3.2.1.2   Displacements resulting from a circular traction patch for $z \geq 0$

The same traction forces shown in Eq. (3.9) are applied on the surface of substrate and we aim to calculate the displacements in a plane below the surface. The displacement in the substrate can be calculated by using a similar approach like in the 2D case. In this approach, Fourier space Green's function in polar coordinates can be written as

$$\tilde{G}_{ij}(\rho, \phi, z) = \frac{2(1+\nu)e^{-z\rho}}{E\rho} \begin{bmatrix} 1 - (\nu + z\rho/2)\cos^2\phi & -(\nu + z\rho/2)\sin\phi\cos\phi \\ -(\nu + z\rho/2)\sin\phi\cos\phi & 1 - (\nu + z\rho/2)\sin^2\phi \end{bmatrix}.$$

The displacement field in Fourier space as a function of z is calculated by making use of the convolution theorem

$$\begin{bmatrix} \tilde{u}_x \\ \tilde{u}_y \end{bmatrix} = \frac{2(1+\nu)e^{-z\rho}}{E\rho} \begin{bmatrix} 1 - (\nu + z\rho/2)\cos^2\phi & -(\nu + z\rho/2)\sin\phi\cos\phi \\ -(\nu + z\rho/2)\sin\phi\cos\phi & 1 - (\nu + z\rho/2)\sin^2\phi \end{bmatrix} \begin{bmatrix} \tilde{f}_x \\ \tilde{f}_y \end{bmatrix}.$$

Therefore, the real space displacements can be obtained by the inverse Fourier transformation

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi} \int_0^\infty \begin{bmatrix} \tilde{u}_\rho(\rho, \phi, z) \\ \tilde{u}_\phi(\rho, \phi, z) \end{bmatrix} e^{-i\rho r \cos(\phi - \theta)} \rho\, \mathrm{d}\rho\, \mathrm{d}\phi.$$

We simplify the results as

$$u_x(r, \theta, z) = \frac{(1+\nu)R}{\pi E}\left[(t_1 - \nu t_2 - \frac{z}{2}t_3)f_0 \cos\gamma - (\nu t_4 + \frac{z}{2}t_5)f_0 \sin\gamma\right],$$

$$u_y(r, \theta, z) = \frac{(1+\nu)R}{\pi E}\left[-(\nu t_4 + \frac{z}{2}t_5)f_0 \cos\gamma + (t_1 - \nu t_6 - \frac{z}{2}t_7)f_0 \sin\gamma\right],$$

where we define the functions t_1 to t_7

$$t_1 = 2\pi I(1, 0; -1),$$

$$t_2 = 2\pi \cos^2\theta I(1, 0; -1) - \frac{\cos(2\theta)}{r}I(1, 1; -2),$$

$$t_3 = 2\pi \cos^2\theta I(1, 0; 0) - \frac{\cos(2\theta)}{r}I(1, 1; -1),$$

$$t_4 = -\pi \sin(2\theta)I(1, 2; -1),$$

$$t_5 = -\pi \sin(2\theta)I(1, 2; 0),$$

$$t_6 = 2\pi \sin^2\theta I(1, 0; -1) + \frac{\cos(2\theta)}{r}I(1, 1; -2),$$

$$t_7 = 2\pi \sin^2\theta I(1, 0; 0) + \frac{\cos(2\theta)}{r}I(1, 1; -1).$$

The integrals of Lipschitz-Hankel types are shown as the follows

$$I(1, 0; -1) = \int_0^\infty J_1(2\pi R\rho)J_0(2\pi r\rho)\rho^{-1}e^{-z\rho}\mathrm{d}\rho,$$

$$I(1, 1; -2) = \int_0^\infty J_1(2\pi R\rho)J_1(2\pi r\rho)\rho^{-2}e^{-z\rho}\mathrm{d}\rho,$$

$$I(1, 0; 0) = \int_0^\infty J_1(2\pi R\rho)J_0(2\pi r\rho)e^{-z\rho}\mathrm{d}\rho,$$

$$I(1, 1; -1) = \int_0^\infty J_1(2\pi R\rho)J_1(2\pi r\rho)\rho^{-1}e^{-z\rho}\mathrm{d}\rho,$$

$$I(1, 2; -1) = \int_0^\infty J_1(2\pi R\rho)J_2(2\pi r\rho)\rho^{-1}e^{-z\rho}\mathrm{d}\rho,$$

$$I(1, 2; 0) = \int_0^\infty J_1(2\pi R\rho)J_2(2\pi r\rho)e^{-z\rho}\mathrm{d}\rho,$$

where J_i is Bessel function with the integer order $i = 0, 1, 2$. We take two approaches to solve these integrals of Lipschitz-Hankel types. One is numerical integration, called semi-analytical theory. Another one is a completely analytical solution based on integral tables, see in Appendix B.

Figure 3.2 shows the solution of analytical displacement field calculated from 15 circular traction patches. The mechanical properties of the substrate are characterized by its Young's modulus of $E = 10\,\mathrm{kPa}$ and a Poisson ratio of $\nu = 0.3$. The 15 circular traction patches with random directions are shown in Fig. 3.2 (a), in which the maximal traction is $200\,\mathrm{Pa}$. Figure 3.2 (b) shows the analytical displacement fields obtained by using 2D theory and semi-analytical theory on the surface of substrate at $z = 0$. We find that these two displacement fields are identical and a local comparison obtained by using 2D theory and semi-analytical theory at $y = 6\,\mu\mathrm{m}$ is shown in Fig. 3.2 (b-Right). We further plot the displacement at different positions, $z = 0\,\mu\mathrm{m}$, $0.5\,\mu\mathrm{m}$, $1\,\mu\mathrm{m}$, and $1.5\,\mu\mathrm{m}$ in Fig. 3.2 (d). The maximal magnitude of displacement in each layer decreases with growing vertical distance from the surface z.

The solutions for a displacement field at $z \geq 0$ obtained by evaluating tabulated integrals are shown in Appendix B. Note that the analytical solutions and the semi-analytical solutions for theory deviate from each other due to significant errors resulting from numerical evaluation of tabulated integral expressions.

In this thesis, we employ the 2D analytical solutions to generate the artificial test data. The process of generating this data is shown in Fig. 3.4 (a, i–iv) and involves prescribing traction force magnitude and direction in distributed circular areas, analytical calculation of the resulting displacements [37], sampling displacements at discrete positions and addition of noise. Throughout the chapter, artificial test data is generated for gel substrates with a Young's modulus of $E = 10 \, \text{kPa}$ and a Poisson's ratio of $\nu = 0.3$. The size of the image plane is arbitrary, but fixed to $25 \, \mu m \times 25 \, \mu m$ and involves 9 or 15 circular traction spots. For these fixed geometries we vary the traction magnitude, density of displacements, and the noise level.

3.2.2 Evaluation metrics for assessing the quality of traction reconstruction

To evaluate the quality of the reconstructed traction, we introduce four different error measures comparing reconstructed traction and known original traction. For this purpose, traction at every grid node is written as a two-dimensional vector $\mathbf{t} = \{t_x, t_y\}$. Real traction and reconstructed traction are discriminated by superscripts as $\mathbf{t}^{\text{real}}$ and $\mathbf{t}^{\text{recon}}$. The error measures are calculated by discriminating traction inside and outside of N_i circular traction patches in a test sample.

- The *Deviation of Traction Magnitude at Adhesions* (DTMA) [37] is defined as

$$\text{DTMA} = \frac{1}{N_i} \sum_i \frac{\text{mean}_j \left(\|\mathbf{t}_{j,i}^{\text{recon}}\|_2 - \|\mathbf{t}_{j,i}^{\text{real}}\|_2 \right)}{\text{mean}_j \left(\|\mathbf{t}_{j,i}^{\text{real}}\|_2 \right)}, \tag{3.14}$$

  where N_i is the number of circular traction patches and the index i runs over all patches. The index j runs over all traction vectors in one patch. A DTMA of 0 represents a perfect average traction recovery and a negative or positive value implies underestimation or overestimation, respectively.

- The *Deviation of Traction Magnitude in the Background* (DTMB) is the normalized difference between the reconstructed and real traction magnitude outside the circular patches

$$\text{DTMB} = \frac{\text{mean}_k \left(\|\mathbf{t}_k^{\text{recon}}\|_2 - \|\mathbf{t}_k^{\text{real}}\|_2 \right)}{\frac{1}{N_i} \sum_i \text{mean}_j \left(\|\mathbf{t}_{j,i}^{\text{real}}\|_2 \right)}, \tag{3.15}$$

  where the index k runs over all traction vectors outside the patches. A DTMB with a magnitude much smaller than unity implies low background noise in the reconstructed traction.

- The *Signal to Noise Ratio* (SNR) for TFM

$$\text{SNR} = \frac{\frac{1}{N_i} \sum_i \text{mean}_j (\|\mathbf{t}_{j,i}^{\text{recon}}\|_2)}{\text{std}_k (\mathbf{t}_k^{\text{recon}})}. \tag{3.16}$$

measures the detectability of a real signal within a noisy background [218]. As before, the index k runs over all traction vectors outside the patches while j is the index of each traction vector in the patch i. The value of the SNR runs from 0 to infinity where a SNR that is much larger than unity indicates a good separation between traction and noise.

- The *Deviation of the traction Maximum at Adhesions* (DMA) measures how peak-values of the traction over- or underestimate the true value. The quantity is defined as

$$\text{DMA} = \frac{1}{N_A} \sum_i \frac{\left[\max_j(\|\mathbf{t}_{j,i}^{\text{recon}}\|_2) - \max_j(\|\mathbf{t}_{j,i}^{\text{real}}\|_2) \right]}{\max_j(\|\mathbf{t}_{j,i}^{\text{real}}\|_2)}, \tag{3.17}$$

where the maxima of traction magnitude are calculated for each traction patch separately through index j. This error measure is particularly important since traction maxima are easy to extract from real experimental data. A DMA of 0 means that the local traction maxima in the reconstruction and in the original data are equal. Positive or negative values of the DMA indicate that the maximum of traction is overestimated or underestimated.

3.2.3 Experimental procedures

The experimental data used in this thesis was generated by collaborators. For completeness, we here briefly summarize the employed procedures. Primary murine podocytes were isolated and maintained by following previously published protocols [219]. In brief, mGFP positive podocytes were isolated from mTom/mGFP*Nphs2Cre reporter mice and subsequent FACS based purification resulted in a primary podocyte culture of highest purity [220]. Mice were housed in a SPF facility with free access to chow and water, according to the NIH guide for the care and use of laboratory animals as well as the German law for the welfare of animals (kept at 12 hour day/night cycle). All animal experiments were approved by local authorities (Regierungspraesidium Freiburg, Germany - G11/51). TFM substrates were prepared according to previously established protocols74. For all experiments, gels were prepared from 12% acryamide and 0.15% Bis-acrylamide and contained fluorescent beads. Elastic properties of this gel are characterized by a Young's modulus of $49\,\text{kPa}$ and we assumed a Poisson's modulus of 0.47. The gel surface was covered with the crosslinker Sulfo-SANPAH and Fibronectin solution was added. Crosslinking was enhanced by exposure of the gels to UV light for 5minutes. After letting the reaction proceed for 12hours, the gels were thoroughly washed with water and PBS. Subsequently, primary podocytes were seeded and cultivated for 12–16hours. Then, coverslips were placed in flow chambers and images of beads and cells were recorded on an inverted confocal microscope with a 63x objective. After recording images of a cell and of the beads below it, the cell was completely removed from the substrate by using a micromanipulator (Eppendorf). Subsequently, images of the fluorescent beads in the relaxed gel were recorded, yielding an unstressed reference state. Substrate deformations between the stressed state with cells and the unstressed reference state were quantified using correlation-based tracking of the beads.

Embryonic rat heart muscle cells were obtained from pregnant rats (Wistar, Charles River, Sulzfeld) at 18–19 days of gestation, details are published elsewhere [221]. Before decapitation, the pregnant rat was anesthetized with CO2. Afterwards the rat embryos were
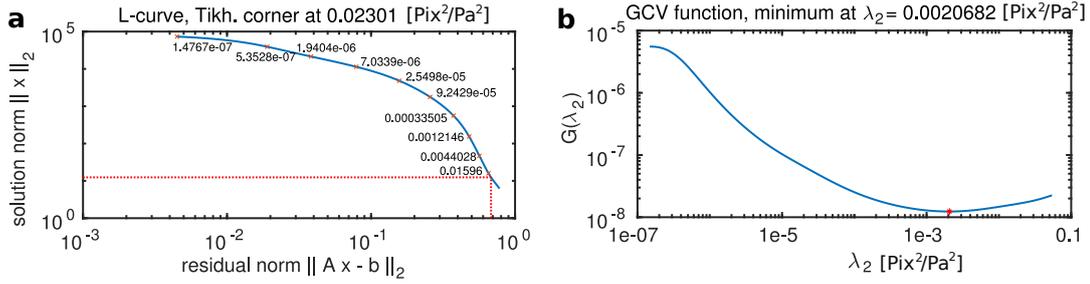
Figure 3.3: Classical methods for selecting the regularization parameter λ_2 with the L-curve and GCV for strong noise $\sigma_{\mathbf{n}}/\sigma_{\bar{\mathbf{u}}} \simeq 0.85$. The L2 regularization parameters suggested by the L-curve criterion and the GCV differ considerably, about by a factor of ten. Data is artificial and consists of 15 circulars traction spots.

taken and decapitated immediately (Animal testing license number 84-02.04.2015.A173, Landesamt fuer Natur, Umwelt und Verbraucherschutz Nordrhein-Westfalen, Duesseldorf, Germany). Cover slides were coated with approximately $70\,\mu$m thick silicone elastomer layer produced from a commercial two-component formulation (Sylgard 184, Dow Corning; mixing ratio 50:1 base to crosslinker by weight; cured overnight at 60 °C). These substrates contained fluorescent beads in their uppermost layer (FluoSpheres Crimson carboxylate-modified beads; Invitrogen) and were coated with fibronectin before cell seeding. Details on sample preparation and cell culture are published elsewhere [221]. Calibration of stiffness [222] yielded a Young's modulus of $15\,$kPa and a Poisson's ratio of 0.5. Live cell microscopy on spontaneously beating cardiac myocytes was performed and positions of fluorescent beads were determined by cross-correlation [221, 223].

3.3 Results

3.3.1 Manual selection of optimal regularization parameters is challenging

The optimal regularization parameters $\lambda_{1/2}$ in Eq. (2.3) are usually unknown. Classical methods for their choice are the L-curve criterion [205, 224] or the generalized cross validation (GCV) for L2 regularization [62, 225]. However, these two methods hardly ever produce the same parameter values and results can differ substantially in the presence of noise, see Fig. 3.3. The regularization parameters suggested by L-curve criterion and the GCV are $0.023\,\mathrm{Pix}^2/\mathrm{Pa}^2$ and $0.0021\,\mathrm{Pix}^2/\mathrm{Pa}^2$.

To illustrate the strong effect of regularization on traction reconstruction, we focus on artificial test data where the underlying traction pattern is known. Figure 3.4 (a) illustrates the generation of artificial traction fields consisting of circular patches each exerting $100\,\mathrm{Pa}$. Fig. 3.4 (c) demonstrates how variation of the regularization parameters affects the error of traction reconstruction with different methods. Note that the errors exhibit minima for intermediate values of the regularization parameters. For the methods shown in panels i, iv, v of Fig. 3.4 (c) (L2 regularization, PGL, PGEN), minima occur in the positive error of the background traction DTMB. In contrast, L1 regularization shown in panel ii of Fig. 3.4 (c) exhibits a maximum in the DTMA, indicating that the average traction magnitude is estimated reasonably accurately here.
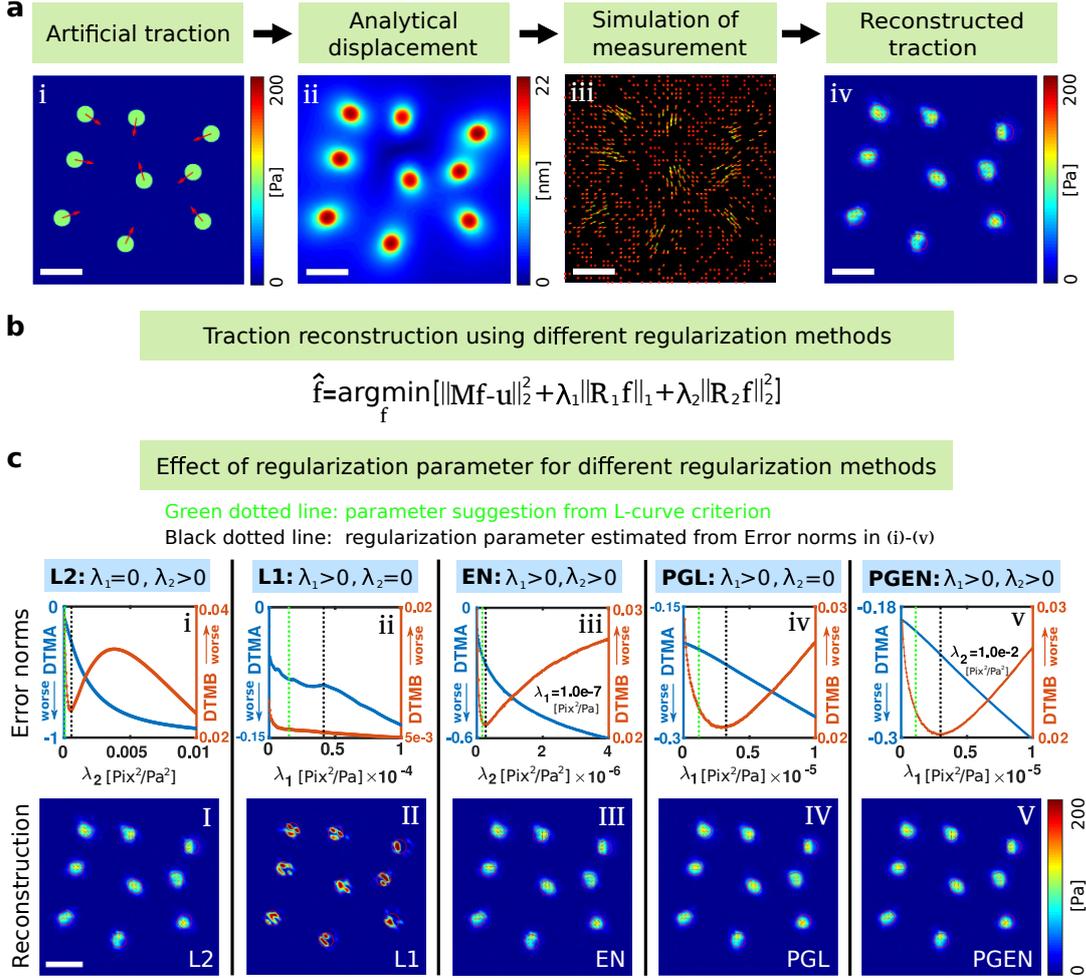
Figure 3.4: Systematic tests illustrate substantial ambiguity in the choice of regularization parameters. (a) Schematic of the employed procedure to test the reconstruction methods. (a-i) Artificial traction pattern consisting of circular spots that uniformly exert a traction of 100 Pa. (a-ii) Analytical calculation of the gel displacements. (a-iii) The displacement field is sampled at random positions representing measurements of motion of fiducial markers. (a-iv) Reconstruction of the traction. (b) Central formula summarizing different regularization approaches. (c) Dependence of various error measures on the regularization parameters. (b-i)-(b-v) Error measures defined in Eqns. ((3.14)-(3.17)) exhibit various extrema and turning points, making the definition of an optimal parameter challenging. Note that the minima of the errors do not correspond to values of regularization parameters suggested by the L-curve criterion (Green dotted lines vs. black dotted lines). DTMA: Deviation of traction magnitude at adhesion, DTMB: deviation of traction magnitude in background. (b-I)-(b-V) Traction fields calculated with regularization parameters that correspond to the error minima at the black dotted lines. Space bar: $5\,\mu$m.

The occurrence of clear minima in the error measures suggests that the corresponding regularization parameter values produce a faithful traction reconstruction. Indeed, employing the values corresponding to the error minima yields traction fields that visually compare well with the original data, see Fig. 3.4 (a and d). Note that for L1 regularization, the reconstruction clearly overestimates the maximum traction locally. As shown in the Appendix B Fig. B.2, the overestimation of the maximum quantified by the DMA can only be reduced through ~ 10 fold reduction of λ_1, which however leads to strong background traction and
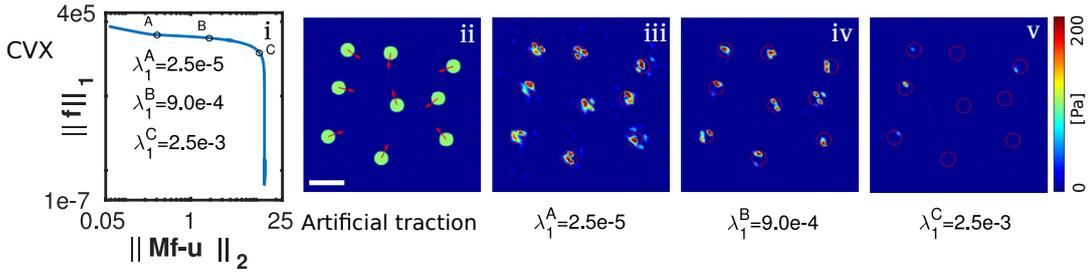
Figure 3.5: Results of L1 regularization using CVX for different λ_1. Reconstruction using the L1_CVX method for 9 circular spots. (i) The L-curve exhibits a turning point indicating a transition from a data-dominated to a regularization-dominated regime. (ii) The employed artificial data contains 2% noise. (iii)-(v) Traction fields obtained by L1_CVX at λ_1^A, λ_1^B and λ_1^C.

suppression of real traction, see also Fig. 3.5. While the minima of the error measures in Fig. 3.4 (c) allow to determine a "best" regularization for test data, the resulting regularization parameter values deviate from those suggested by the L-curve criterion, see green lines in Fig. 3.4 (c). Moreover, the L-curves for these samples are complex and exhibit multiple turning points, illustrating the difficulty in choosing the right regularization parameter in experiments, see the Appendix B Fig. B.5.

3.3.2 The elastic net outperforms other regularization methods for traction reconstruction

To facilitate quantitative comparison of different reconstruction methods, we employ artificial data consisting of 15 circular traction spots with traction magnitude between 0 Pa and 250 Pa, see Fig. 3.6 (a). Gaussian noise with a standard deviation given in percent of the maximal absolute value of the of true displacements is added. The spots have a diameter of 2 μm and the mesh constant for traction reconstruction is 0.5 μm.

Results from different regularization approaches are shown in Fig. 3.6 (a). The figure illustrates that L2 regularization can yield realistic estimates for the absolute magnitude of traction on the spots but produces a strong traction background which may render identification of traction sites difficult. The opposite deficiencies occur for results from L1 regularization. Here, the background is nicely suppressed, which can allow excellent resolution of very small traction spots. However, the peak tractions are significantly overestimated, which can not be mitigated by increasing the regularization parameter, see Fig. 3.7. Note that the quality of L1 regularization can be improved by using an Iterative Reweighted Least Squares algorithm and the solution from the L2 regularization as an initial guess, see Fig. 3.7. The best results are obtained with the EN regularization which combines the advantages of L1- and L2-regularization. Here, we obtain a clean background combined with acceptable accuracy in the absolute traction magnitude on the circular patches. The results from the proximal gradient methods PGL and PGEN qualitatively have a smooth appearance with a level of background traction that is between those of L2 and L1. Fig. 3.6 (b) quantifies the described differences between the regularization methods through the error in traction magnitude on the traction spots (DTMA), the error in traction magnitude in the background (DTMB), signal to noise ratio (SNR), and error in maximum traction on the spots (DMA). The Appendix B Fig. B.8 contains additional plots of these quantities. We find that the reconstruction quality of traction and background improves with increasing number of displacement
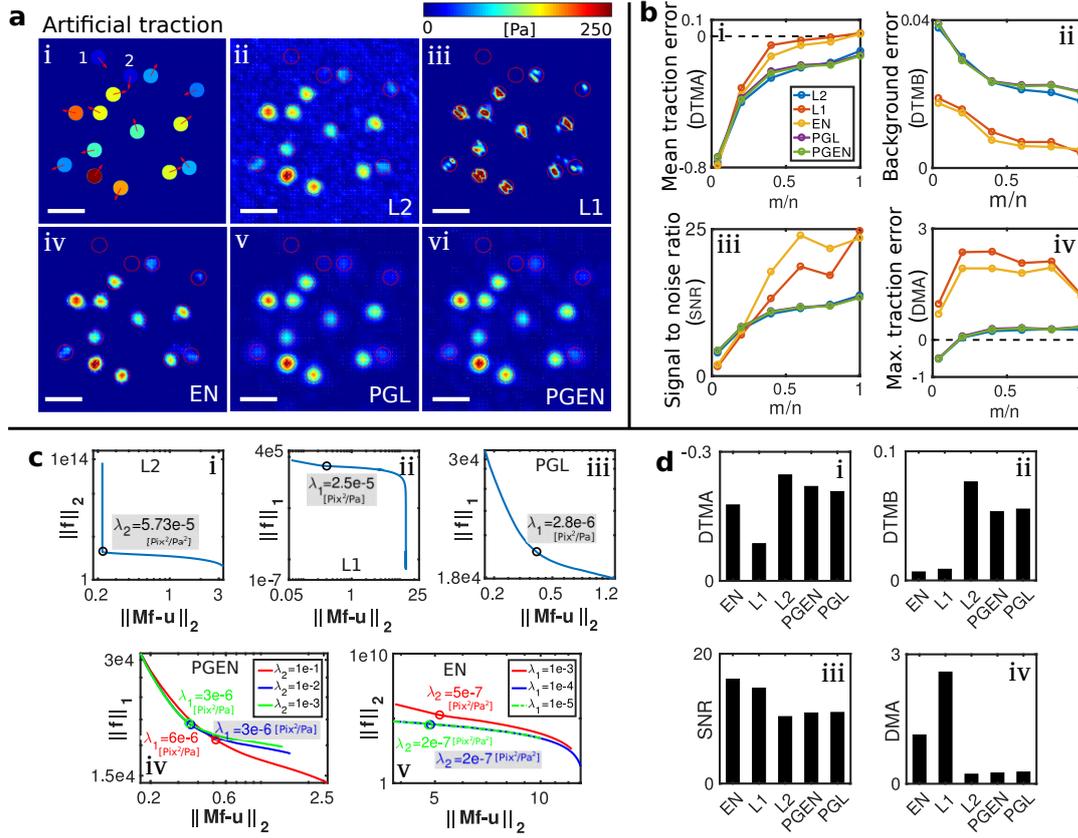
Figure 3.6: The elastic net (EN) outperforms other reconstruction methods in the presence of noise and when applied to undersampled data. (a) Artificial test data with uniform traction spots and 4% noise in the displacements. Traction maps in (ii–vi) result from usage of different regularization methods. Space bar: $5\,\mu m$; displacements are sampled on average every $0.5\,\mu$m. (b) Comparison of errors resulting from undersampled data. Undersampling is realized by reducing the number of displacement vectors m. (c) L-curves with chosen regularization parameters (gray boxes) for a data set containing 2% noise and $m/n = 0.4$. (d) Comparison of errors for the regularization parameters shown in (c). EN regularization shows a favorable tradeoff between error and background signal.

measurements m. Furthermore, EN regularization outperforms other regularization methods with regard to the reconstruction accuracy of undersampled data ($m/n < 1$). However, the advantage of EN regularization comes at a significantly increased computation time and memory requirement as shown in Table 3.1.

Table 3.1: Overview of the runtime RAM requirement for each method.

Reconstruction method	Regularization					Bayesian models	
Name	L2	L1	EN	PGL	PGEN	BL2	ABL2
Simulation time	8 s	75 s	0.8 h	126 s	127 s	0.1h	0.5h
RAM requirement	350 MB	1.98 GB	3.87 GB	101 MB	107 MB	400 MB	400 MB

The benchmark tests were conducted with a data set consisting of 1000 displacement measurements and a traction field consisting of 2500 entries.
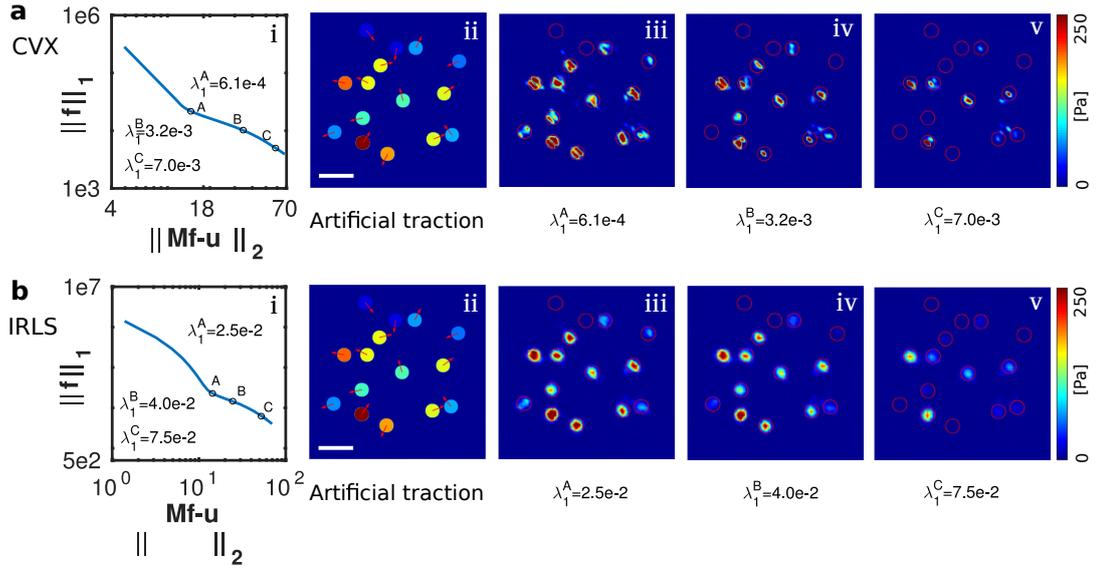
Figure 3.7: Results of L1 regularization using CVX and IRLS for different λ_1. (a) and (b) Reconstruction using the L1_CVX and L1_IRLS method for 15 circular spots, respectively. (b-i) and (c-i) The L-curve and three selection of points A, B and C for CVX and IRLS. (a-ii) and (b-ii) The input artificial test traction. (a-iii)-(a-v) Reconstruction using L1_CVX at different λ_1. (b-iii)-(b-v) Reconstruction using L1_IRLS at different λ_1. Clearly, the regularization parameter should be chosen well-above the turning point to avoid partial suppression of traction patterns.

3.3.3 Bayesian variants of the L2 regularization allow parameter-free traction reconstruction

We next consider the performance of the two Bayesian methods, BL2 and ABL2, that allow automatic choice of the optimal L2 regularization parameter, as schematically shown in Fig. 3.8 (a). Both methods select the optimal regularization parameter by maximizing the logarithmic evidence, Eq. (2.20). As illustrated in Fig. 3.8 (a), the regularization parameter is here deduced from the parameters β and α, characterizing the distributions of measurement noise and traction respectively. We first employ the same test data as used for Fig. 3.6, containing 5% Gaussian noise in the displacements with $\beta = 400 \, \mathrm{Pix}^{-2}$.

With BL2, the log evidence exhibits a clear maximum in a one-dimensional space as seen in Fig. 3.8 (c). Figure 3.8 (d) shows the reconstructed traction employing the optimal parameter $\hat{\lambda}_2 = 76.75 \, \mathrm{Pa}^2/\mathrm{Pix}^2$. Visual comparison of the color-coded traction magnitude in Fig. 3.8 (d,b) clearly shows that the reconstructed traction has the correct range.

For ABL2, the evidence is a function of β and α as seen in Fig. 3.8 (e). Numerical localization of the maximum yields $\hat{\alpha} = 3.06e4 \, \mathrm{Pa}^{-2}$ and $\hat{\beta} = 394 \, \mathrm{Pix}^{-2}$, which is very close to the known input value of $\beta = 400 \, \mathrm{Pix}^{-2}$. The optimal regularization parameter in this case is thus $\hat{\lambda}_2 = \hat{\alpha}/\hat{\beta} = 77.66 \, \mathrm{Pa}^2/\mathrm{Pix}^2$, which agrees well with the estimate from BL2 (76.75 $\mathrm{Pa}^2/\mathrm{Pix}^2$). The resulting traction map is shown in Fig. 3.8 (f) and is very similar to the traction map resulting from BL2 in Fig. 3.8 (d). Thus, BL2 and ABL2 yield consistent parameter estimates that produce traction reconstruction of good accuracy. See the Appendix B Fig. B.8 for a comparison of the Bayesian methods with non-Bayesian approaches.

As with other regularization approaches, the quality of reconstruction strongly depends on
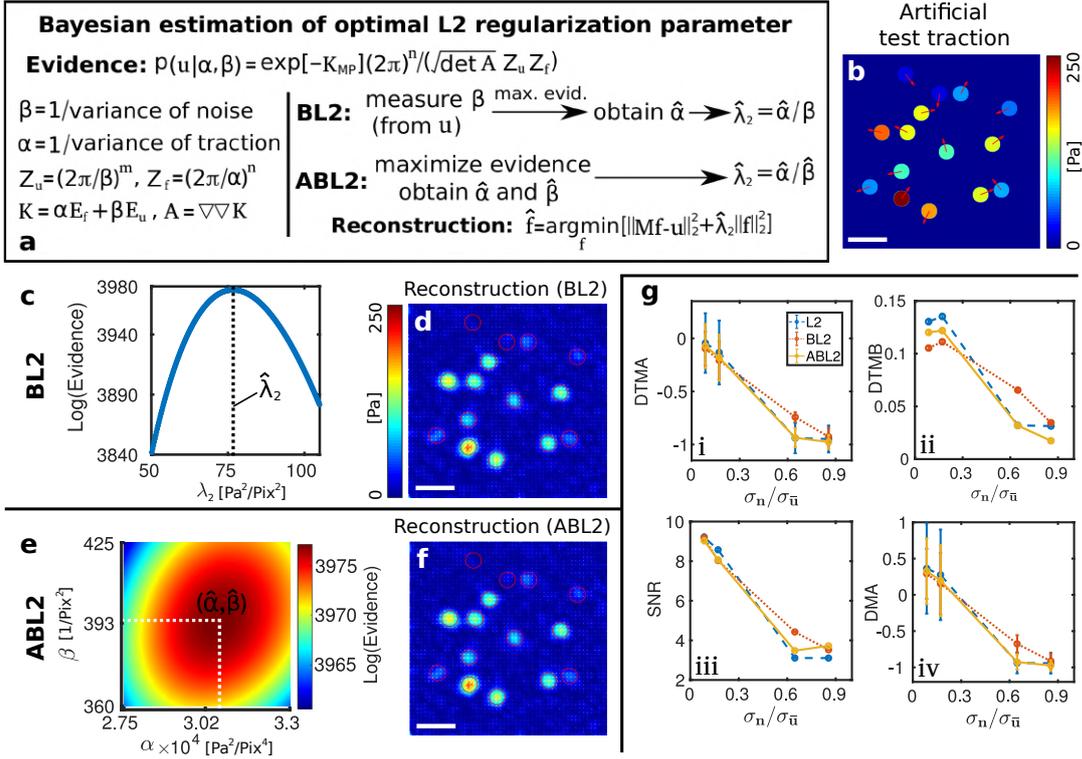
Figure 3.8: Bayesian L2 regularization (BL2) and Advanced Bayesian L2 regularization (ABL2) are robust methods for automatic, optimal regularization. (a) Schematic diagram of the procedure employed to infer $\hat{\lambda}_2$ in BL2 and ABL2. BL2 requires the variance of the displacement measurements $1/\beta$, which can be obtained by analyzing displacement noise far away from any cell. ABL2 estimates this noise strength directly from the data. (b) Artificial test data. For the shown results, 5% Gaussian noise is added to the displacements. Space bar: $5\,\mu$m. (c) For BL2, the optimal regularization parameter is located at the maximum of a one-dimensional plot of the evidence Eq. (4.9). (d) Reconstruction of traction force using BL2. (e) For ABL2, the optimal regularization parameter is located at the maximum of a two-dimensional plot of the data evidence vs. α and β. (f) Reconstruction of traction force using ABL2. (g-i)-(g-iv) Comparison of reconstruction accuracy for L2, BL2 and ABL2. Different levels of traction forces were applied to change signal-to noise ratio. Here, σ_n is the standard deviation of the noise and $\sigma_{\bar{u}}$ is the standard deviation of the noise-free traction field. Note that BL2 outperforms the other methods for high noise levels.

the present noise. When the magnitude of the noise is comparable to the magnitude of the displacements caused by the traction ($\sigma_{\mathbf{n}}/\sigma_{\bar{\mathbf{u}}} \approx 1$), little information can be recovered. For instance, the circular spots with weak traction labelled 1 and 2 in Fig. 3.6 (a) are almost impossible to detect in the presence of 5 % noise, but can be reconstructed in the noise-free case, see the Appendix B Fig. B.6. To quantitatively assess the fidelity of reconstruction with small traction forces, we employ a constant 5 % but scale the tractions to mean values of (12 Pa, 16 Pa, 60 Pa and 120 Pa). The resulting relative strength of noise and displacements is quantified through the ratio of standard deviations $\sigma_{\mathbf{n}}/\sigma_{\bar{\mathbf{u}}}$, which is plotted against our reconstruction quality measures in Fig. 3.8 (g). For comparison, results from manual selection of the regularization parameter using the L-curve criterion are also given. The reconstruction qualities of BL2, ABL2 and L2 are similar when $\sigma_{\mathbf{n}}/\sigma_{\bar{\mathbf{u}}} \ll 1$ (high traction). However, BL2 and ABL2 have an improved signal to ratio SNR compared with the L-Curve approach when $\sigma_{\mathbf{n}}/\sigma_{\bar{\mathbf{u}}}$

approaches unity, see (iii). This is due to the difficulties with the L-curve criterion at high noise. The logarithmic evidence function exhibits in all cases a clear maximum that enables robust and reliable choice of optimal parameters with BL2 and ABL2. In general, the results from BL2 are however more reliable since the optimization involves here only one parameter. Overall, the tests with artificial data show that these Bayesian methods containing few additional parameters to be determined from the data can resolve the ambiguity associated with manual choice of the regularization parameters over a wide range of signal strengths $\sigma_\mathbf{n}/\sigma_\mathbf{\bar{u}} < 1$.
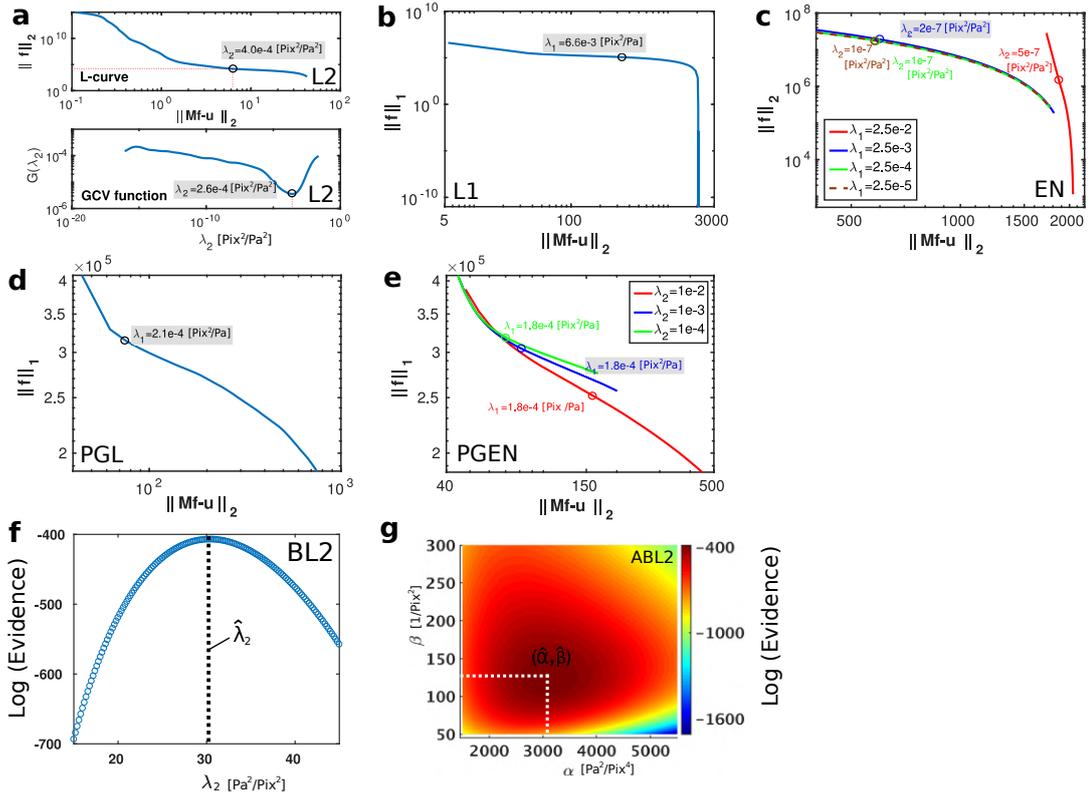
Figure 3.9: The determination of regularization parameters from different methods for the experimental data. (a) L-curve and GCV function for the classical L2 regularization. (b)-(c) L-curves for L1- and EN-regularization. (d)-(e) L-curves for PGL and PGEN. (f) The logarithmic evidence curve calculated in BL2 as function of the regularization parameter. The variance of the measurement noise is estimated from the data to be about $0.01\,\mathrm{Pix}^2$. (g) Map of the logarithmic evidence for ABL2. The maximum of this function is located at $1/\beta = 0.008\,\mathrm{Pix}^2$, which is close to the noise variance estimated from the data

BL2 and ABL2 are based on the simplest structure of a Bayesian model with only one, global prior distribution. One may hypothesize that more complex hierarchies of priors yield an improved traction estimate. For instance, it is possible to prescribe a position-dependent prior for sparse traction patterns through hierarchical Bayesian networks. Such methods require more advanced techniques for sampling of the probability distributions and optimization, such as variational techniques or Markov-chain Monte Carlo methods. We have tested three such algorithms that were originally developed for purposes other than TFM [70, 72, 75, 226]. Results are shown in the information of the Appendix B Fig. B.7 and Fig. B.8. However, the tested algorithms all produce highly overestimated, localized traction patterns that sensitively depend on noise. Such errors are likely due to the many free parameters of the

models that do not favor a faithful data reconstruction in spite of the sparsity constraints. Thus, our tests suggests that these hierarchical network models are not suited for the inverse problem associated with TFM.
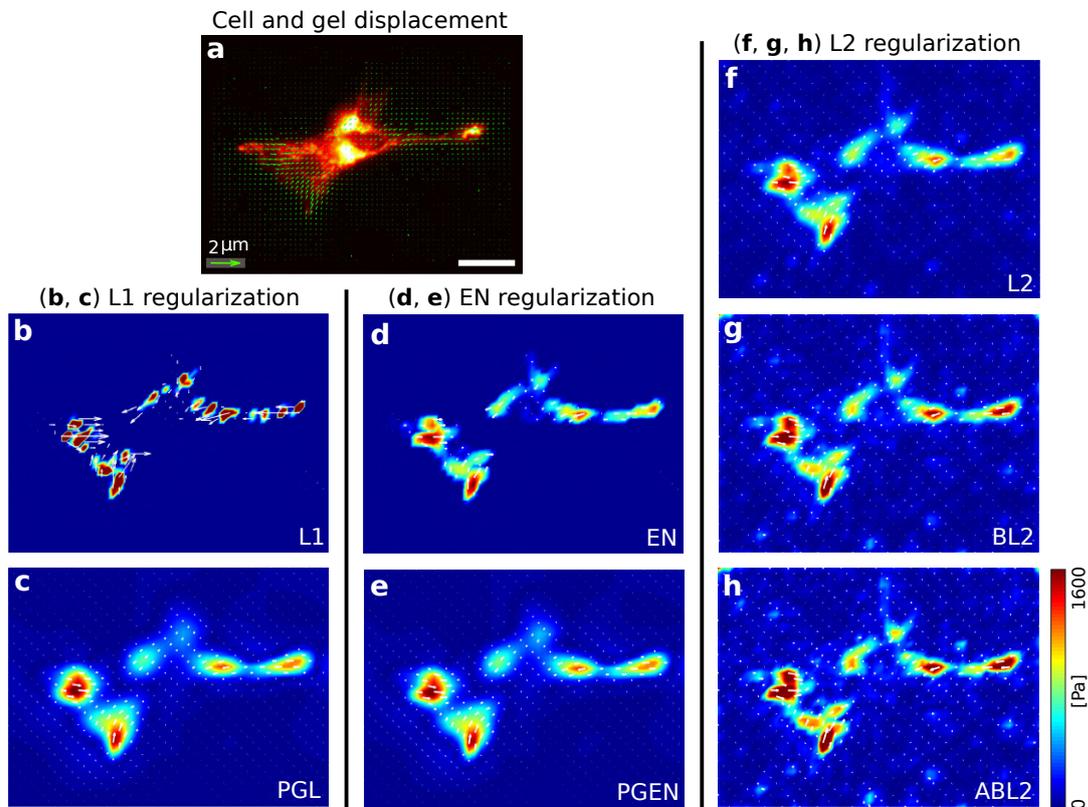
Figure 3.10: Test of all reconstruction methods using experimental data. (a) Image of an adherent podocyte with substrate displacements shown as green vectors. (b)-(h) Reconstructed traction forces using L2, L1, EN, BL2, PGL, PGEN and ABL2, respectively. Reconstruction with L2-type regularization exhibits a comparatively high background noise. L1-regulation shows very high, localized traction. Based on tests with artificial data, we expect that these peaks overestimate the traction. The EN method combines the advantages of L1 and L2 regularization, namely a clean background and localized traction of reasonable magnitude. PGL and PGEN have smooth traction forces at adhesion and background. (g-h) The Bayesian methods BL2 and ABL2 yield very similar results as the standard L2 regularization without requiring a search for the optimal regularization parameters. For better visibility, only every fourth traction vector is shown. Space bar: $25\,\mu\text{m}$.

3.3.4 Test of methods with experimental data

To compare the performance of the different methods for real cells, we employ primary mouse podocytes studied with a standard TFM setup. The deformation field resulting from cellular traction is shown in Fig. 3.10 (a). Using this displacement data, we find that the variance of the noise is $\sim 0.01\,\text{pix}^2 = 103.4\,\text{nm}^2$ in regions that are far away from the cell. The maximum displacement is $0.52\,\mu\text{m}$. Fig. 3.10 (b)-(h) show reconstruction results using all methods discussed above. As for artificial data, we find here that the EN regularization results in a very clear background. The traction magnitudes and shapes of the traction spots are similar to those resulting from regularization with the L2 method. For L1 regularization,

traction localizes in sparse regions and has a significantly higher value than for other methods. Proximal gradient methods produce smooth traction profiles as expected from the use of the soft wavelet thresholding. The magnitude of traction measured with PGL and PGEN is close to the results of EN and L2.

Next, we considered the performance of the Bayesian methods. The logarithmic evidence, Eq. (2.20), calculated with BL2 and ABL2 reveals pronounced maxima, allowing to robustly choose the optimal parameters for the experimental data. See Fig. 3.9 (f) and (g). The resulting values for $\hat{\lambda}_2$ are $30.4\,\mathrm{Pa}^2/\mathrm{Pix}^2$ and $24.4\,\mathrm{Pa}^2/\mathrm{Pix}^2$ for BL2 and ABL2, respectively, and thus agree reasonably well with each other. For BL2, only a single maximum of the evidence was found in all our tests. For ABL2, we found that further extrema may occur at the boundary of the region of scanned parameters. Figures 3.10 (e), (h) show the traction fields calculated with BL2 and ABL2. These traction fields are visually very similar to the one obtained with standard L2 regularization. However, the L-curve criterion provides a much more uncertain estimate of a regularization parameter due to the difficulty of localizing it on a logarithmic scale, see Fig. 3.9 (a-e). Note that the regularization parameters obtained from the L-curve criterion can not be directly compared to the parameters resulting from the Bayesian methods due to standardization employed for the latter. Overall, the suggested Bayesian models can eliminate ambiguity in TFM by automatically providing a consistent parameter choice.
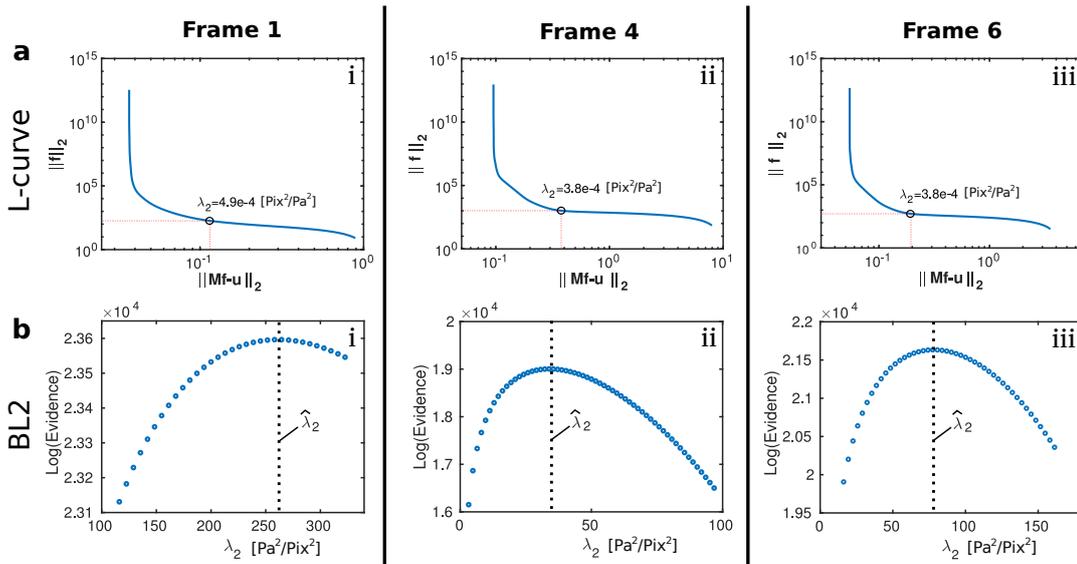
Figure 3.11: L-curve selection regularization parameter for the data for analysis of traction time sequences. (a) L-curve and GCV function for the classical L2 regularization. (b)-(c) L-curves for L1- and EN-regularization. (d)-(e) L-curves for PGL and PGEN.

3.3.5 Bayesian regularization enables consistent analysis of traction time sequences

TFM is frequently employed to study dynamic aspects of cell mechanics. Examples include cell migration, cell division, or cytoskeletal reorganization in response to extracellular stimuli. Such processes are usually accompanied by a change in the traction distribution. As a result, the optimal regularization parameter varies among different images in a time sequence

of microscopy data. Additionally, the regularization parameter can also change if the degree of noise varies over time, which can be caused for example by stage drift or photo bleaching. In such cases, it is very challenging to perform a consistent, frame-by frame analysis to determine the degree of regularization with conventional methods. Thus, one fixed parameter is commonly employed for the whole time sequence whereby precision and accuracy of traction reconstruction are sacrificed.
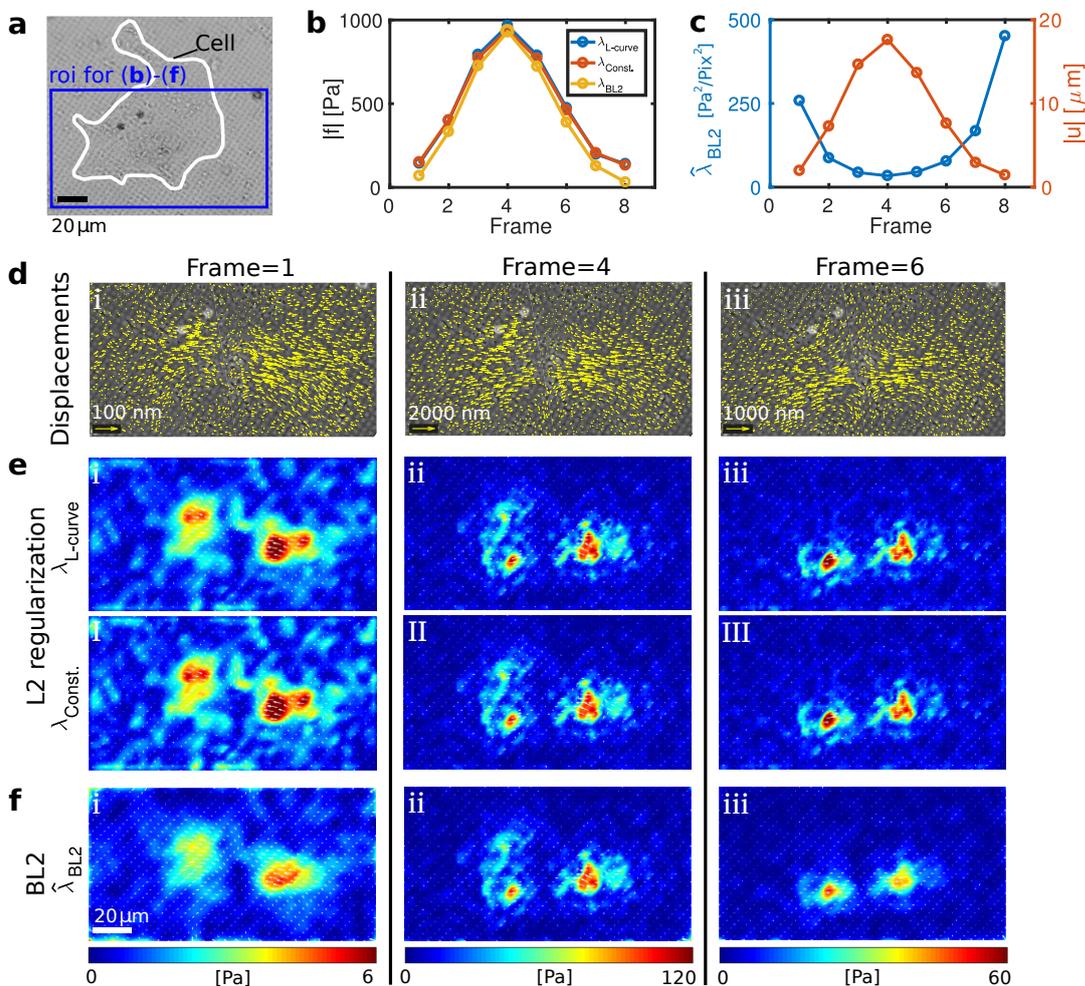
Figure 3.12: Bayesian L2 regularization robustly adapts to different traction levels allowing quantitative analysis of time series. (a) Image of a spontaneously beating heart muscle cell on an elastic, micropatterned substrate. (b) Overall norm of traction magnitude in successive image frames. The maximum corresponds to one contraction of the heart muscle cell. Traction is calculated with BL2 or, for comparison, via L2 regularization where λ_2 is either selected manually for every frame using the L-curve criterion or held constant throughout the image sequence. (c) Optimal regularization parameter suggested by BL2 and the norm displacement field correlate. (d-i)-(d-iii) Cell images with displacement field at frames 1, 4, and 6. (e-i)-(e-iii) Snapshots of the traction fields resulting from L2 regularization with a manually chosen parameter $\lambda_{\text{L-curve}}$ and a constant parameter $\lambda_{\text{L-const.}}$ in an intermediate range. (f-i)-(f-iii) Snapshots of the traction fields resulting from BL2. Note the different scaling of displacement and tractions for the different frames. Frame 1 (I) illustrates that BL2 yields a smaller traction magnitude than L2 in the presence of large noise, where the L-curve criterion is hard to employ. As a result, BL2 allows to differentiate real traction from noise outside of the cell. For better visibility, only every fourth traction vector is shown.

To test whether our Bayesian methods can be useful in this situation, we employ TFM data with a spontaneously beating cardiac myocyte, see Fig. 3.12. Due to the large size of the cell, we focus on a region of interest shown in Fig. 3.12 (a). The analyzed time sequence corresponds to one cell contraction. Snapshots from frames 1, 4, and 6 are shown for illustration. Figure 3.12 (b) shows the overall norm of reconstructed traction where λ_2 is either chosen according to the L-curve criterion, held at an intermediate, constant value, or automatically determined in BL2. The overall traction magnitudes are similar in frames 2–6 where traction is high. Differences occur, however, in the low-traction regime, where BL2 systematically yields lower values of traction. We expect that the results from BL2 are more trustworthy in this regime since the L-curve criterion yields highly ambiguous values for the regularization parameters, see Fig. 3.11. Figure 3.12 (c) shows the overall norm of the gel displacement and the optimal regularization parameter estimated with BL2. The noise variance is small, $\sim 0.00003\,\mathrm{pix}^2$, in regions that are far away from the cell (pixel size $0.1\,\mu\mathrm{m}$). As expected, λ_{BL2} is inversely correlated with the displacement magnitude.

Figure 3.12 (e) shows snapshots of the resulting traction fields that illustrate again that BL2 produces slightly different results for low traction, most apparent in Fig. 3.12 (e) i, I and Fig. 3.12 (f), i. Note that the traction field resulting from classical L2 regularization in Fig. 3.12 (e), i, I shows a noise background outside of the cell that is almost comparable to the real cellular traction. In contrast, BL2 suppresses this background at the price of an apparently reduced spatial resolution as seen in Fig. 3.12 (f), i. However, this provides an objective distinction between real signal and noise, which is what is to be expected from a faithful data reconstruction.

3.4   Summary and discussion for Chapter 3

Many, if not all, TFM methods critically rely on some form of noise reduction. Usually, traction is calculated from substrate displacement through the solution of a linear problem involving elastic Green's functions. Here, the effects of noise are not a technical issue relating to the data precision, but connected directly to the structure of the linear problem where even the slightest numerical noise can be amplified to an extent that the true solution is entirely lost. The most immediate approach is to deal with noise is to filter the displacement field prior to traction reconstruction. Filtering becomes possible if the solution is calculated in Fourier space because the convolution theorem simplifies the matrix inversion [155]. However, filtering the input data generally reduces the spatial resolution and optimal resolution can only be gained if the filter is adapted for each sample. In certain cases, moreover, data filtering is not sufficient to guarantee stability of the solution, for example, if the three-dimensional position of displacements is included.

A popular alternative strategy for enforcing well-behaved solutions is regularization. With regularization, Fourier-space inversion becomes more robust [37, 152]. However, regularization is also used for real-space approaches and has been used in conjunction with finite element methods or boundary element methods. Solving the linear problem in real space is generally more demanding, but has the advantage that the spatial sparsity of traction patterns is conserved. For TFM, two regularization methods have to date been used, namely L2 regularization [37, 136, 218] and L1 regularization [158, 159, 182]. These methods each have

one regularization parameter that is chosen manually based on heuristics, which introduces a considerable degree of subjectivity in the resulting traction.

In this work, we systematically compare the classical L1- and L2 regularization to three other methods that have, to our knowledge, not yet been employed for TFM. These three regularization methods are the Elastic Net (EN), Proximal Gradient Lasso (PGL) and Proximal Gradient Elastic Net (PGEN). Our tests with artificial data clearly demonstrate that EN regularization outperforms other regularization methods with regard to the reconstruction quality. Here, accurate traction reconstruction is due to a simultaneous suppression of background noise and penalization of large traction magnitude. In contrast, the proximal gradient methods PGL and PGEN are effective at producing a smoothed traction field, due to the local removal of high-frequency spatial variations. These results obtained with artificial data agree qualitatively with results from tests with experimental data. Here too, L1 and L2 regularization yield overestimated or underestimated traction on adhesion sites. EN again yields a clear background without producing excessively sharp traction peaks at adhesions, see Fig. 3.10. PGL and PGEN yield smooth traction fields and rounded adhesion site contours. While our work presents a comprehensive overview of regularization variants in TFM, it does not cover all variants and solution procedures. For example, an Iterative Reweighted Least Squares algorithm for L1 regularization outperforms in our experience the L1 regularization method studied here, see Fig. 3.7. Also, it has been suggested to use an L1 norm for both the residual and regularization term [159], and various other iterative regularization procedures can be tested for TFM in the future.

Next, we ask if Bayesian methods can eliminate the necessity of a manual choice of regularization parameters. Here, the corresponding parameter values are inferred by maximizing their evidence given a fixed class of chosen probability distributions. Using the simplest approach, our prior assumption on the traction forces is that they are drawn from one global Gaussian distribution with an unknown variance $1/\alpha$. The posterior distribution determining the probability of a particular traction field given a measured displacement field is then determined by the parameter $1/\alpha$ and a further parameter $1/\beta$, quantifying the variance of the measurement noise. For fixed values of α and β, maximization of the posterior distribution corresponds exactly to L2 regularization with λ_2 given by α/β. However, the values of α and β can also be determined through maximizing their probability conditioned on a given measurement and the chosen probability distributions. Here, this is equivalent to maximizing the evidence for $\mathbf{u}$, given any two parameter values. We refer to the simultaneous determination of both parameters from the evidence as advanced Bayesian L2 regularization (ABL2). In an even simpler approach, we estimate the noise strength directly from the displacement data, leaving only one parameter α to be determined by maximization of the evidence; which we call Bayesian L2 regularization (BL2). These methods represent an automatic optimization of the L2 regularization. Thus the resulting traction field has all the qualitative features of L2 regularization, including the suppression of exceedingly high traction values and a visible background traction. For all our tests, we found that BL2 was a robust method yielding reasonable estimates for traction and regularization parameters. Due to the difficulty in choosing the correct regularization parameter manually, BL2 has substantial advantages over the classical L2 procedure, in particular if the traction is so small that the resulting displacements are comparable to the noise $\sigma_{\mathbf{n}}/\sigma_{\bar{\mathbf{u}}} \approx 1$.

We mention that we have also tested more elaborate hierarchical Bayesian network algorithms that were originally designed for other purposes than for use with TFM. These include a variational approach termed "Bayesian compressive sensing using Laplace priors" (BCSL) [72], and Markov chain Monte Carlo methods, for instance the "Bayesian Lasso" [75]. In our experience, however, none of these methods could compete with the much simpler Bayesian L2 regularization when applied to TFM problems, see the Appendix B Fig. B.6 to Fig. B.9.

The advantage of employing Bayesian traction reconstruction is most apparent when cells in different conditions are to be compared. To perform a correct comparison of situations with different traction, different substrate rigidities, etc., it is technically necessary to adapt the regularization parameter for each case. However, the difficulty of finding the corresponding parameters usually makes this impossible, which introduces significant quantitative errors. Bayesian methods present a possible solution to this problem. We have shown here that BL2 produces a regularization that adapts smoothly and robustly to changes of cellular traction. Thus, we expect that this method can be of wide use for quantitative studies of cell physiology.

Chapter 4

Traction force calculations in Fourier space

We have introduced the BEM approach to solve the TFM problem in Chapter 3. The BEM is based on the idea of partitioning the whole surface of the substrate Ω into numbered elements of small, non-overlapping subdomains. The forces on each discrete point can be approximated by using a local shape function, e.g., a pyramidal shape function. The advantage of the BEM is that the formulas can be easily understood and implemented into a numerical algorithm. However, it requires substantial computation (time) to integrate each element. To overcome this weakness, a fast method called Fourier-transform traction cytometry (FTTC) was introduced [37, 153, 155]. In FFTC, the integral Eq. (3.1) can be directly solved by using the convolution theorem, see next section. Using FFTC, we also can obtain a similar linear equation in Fourier space like Eq. (3.5) and thus, the calculation of traction forces also becomes an inverse problem. In Ref. [37], the authors develop an approach which utilizes L2 regularization to solve the inverse ill-posed problem in Fourier space. However, these approaches require one to manually select the regularization parameter.

Virtually all standard methods for the calculation of traction forces require the implicit or explicit choice of a parameter that suppresses noise and leaves as much of the true signal conserved as possible. Within a Bayesian framework, this parameter choice can be rationalized by relating filter- or regularization parameters to prior distributions that represent prior knowledge about the data. Maximizing the likelihood of the prior distributions yields the corresponding optimal trade-off between noise suppression and faithful data reconstruction. Bayesian regularization has been used for example in astrophysics [78, 210] and mechanical structure monitoring [227]. For inference of internal stress in a cell monolayer, an interative maximum a posteriori estimation has been employed [228]. Bayesian L2 regularization was first employed for real-space TFM methods in Chapter 3 since this variant allows comparison of a broad variety of approaches. For practical applications, however, calculations in Fourier domain have significant advantages in terms of robustness and speed. In this chapter, we present the corresponding method that we term *Bayesian Fourier transform traction cytometry* (BFTTC). We compare BFTTC with other methods such as classical L2 regularization, Bayesian L2 regularization in real-space (BL2), and regularized Fourier transform traction cytometry (FTTC). We find that BTTC is a computationally fast method that provides robust traction calculations without requiring manual adjustment of the noise-suppression level. We also provide a Matlab software package for BFTTC that is freely available for download.

This software package is intended to provide a simple and robust solution for data analysis in the hands of experimentalists. A graphical user interface allows intuitive use of the program and little theoretical background knowledge is required.

4.1   Methods and software

4.1.1   Fourier-transform traction cytometry (FTTC)

In Chapter 3, the integral equation Eq. (3.1) was solved through numerical discretization in real space. Alternatively, Eq. (3.1) can be solved in Fourier space. We employ a spatial wave vector $\mathbf{k} = (k_1, k_2)$ with absolute value $k = |\mathbf{k}|$. In standard FTTC, the integral Eq. (3.1) is written as

$$\tilde{u}_{i\mathbf{k}} = \left\{ \sum_j \tilde{G}_{ij} \tilde{f}_j \right\}_{\mathbf{k}}. \tag{4.1}$$

where the tilde denotes the Fourier-transformed quantity. The Fourier-transformed Green function $\tilde{G}_{ij\mathbf{k}}$ is written as

$$\tilde{G}_{ij\mathbf{k}} = \frac{2(1+\nu)}{E} \left(\frac{\delta_{ij}}{k} - \frac{\nu k_i k_j}{k^3} \right), \tag{4.2}$$

where E and ν represent the Young modulus and Poisson ratio, respectively. δ_{ij} is the Kronecker delta function. This equation is also given in Eq. (3.6).

Using a matrix formulation analogous to the real-space expression, we have $\tilde{\mathbf{u}} = \tilde{\mathbf{M}} \tilde{\mathbf{f}}$ with $\tilde{\mathbf{M}}$ having a tri-diagonal structure. For conceptual clarity, in the following we will write the measurement noise in the recorded displacement explicitly as $\mathbf{s}$ in the real-space domain and as $\tilde{\mathbf{s}}$ in Fourier space. This noise can be estimated in the experiment by quantifying the variance of the measured displacements in absence of traction. The discretetized equations then read

$$\begin{cases} \mathbf{u} = \mathbf{M}\mathbf{f} + \mathbf{s} & \text{in real space,} \\ \tilde{\mathbf{u}} = \tilde{\mathbf{M}}\tilde{\mathbf{f}} + \tilde{\mathbf{s}} & \text{in Fourier space.} \end{cases} \tag{4.3}$$

The continuous traction and displacement fields are discretized by rectangular meshes where m and n are the number of discretization nodes for tractions and displacements respectively. In these discretized equations in Eq. (4.3), the size of the displacement vector $\mathbf{u}$ is $2m \times 1$ and size of the traction vector $\mathbf{f}$ is $2n \times 1$ where the two vector components of the planar fields are concatenated. For the Fourier space methods, the displacement and traction fields are discretized with the same grid and we then have $m = n$.

For traction force microscopy, either of these equations is employed to determine the tractions $\mathbf{f}$. The removal of noise is critical in most TFM methods. In real-space calculations, the condition number of $\mathbf{M}$, defined as the ratio of the largest singular value to the smallest, is almost always much larger than unity, typically above 10^5. $\mathbf{M}$ is therefore ill-conditioned which implies that small noise produces drastic changes in the calculated traction forces. For FTTC, spatially varying random noise occurs mainly at high spatial wave numbers. Hence, noise suppression can be achieved by suppressing high frequency data. In Chapter 3, we systematically tested a variety of traction reconstruction approaches based on Eq. (4.3). The standard approach for solving the equation in real space is L2 regularization [37, 141, 153, 229], which invokes a penalty on the traction magnitude to robustly suppress the effects
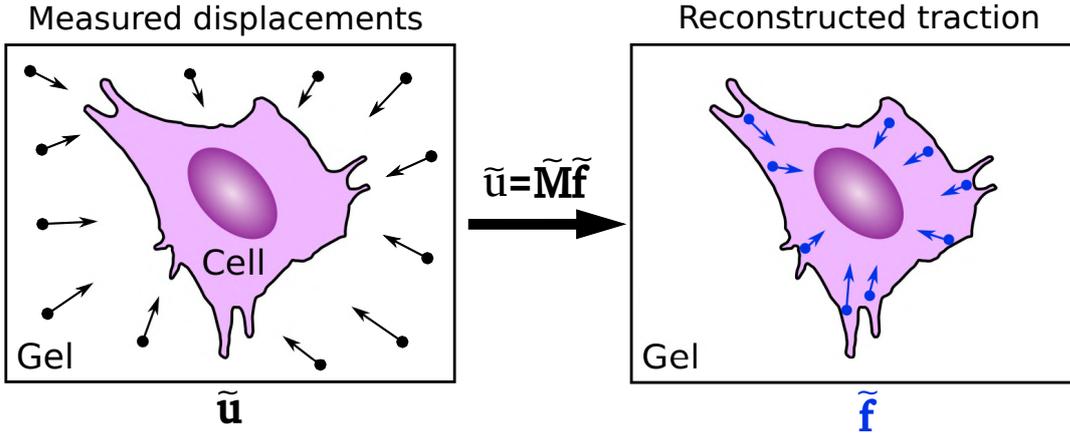
Figure 4.1: Schematic of traction force microscopy (TFM) to measure cellular traction on flat elastic substrates. Adherent cells deform the substrate and the displacement field **u** is obtained by tracking markers within the gel. The traction force field **f** generated by the cell is calculated by inverting a linear equation system.

of noise. With Fourier space methods, a low-pass filter is frequently employed to suppress noise in the displacement field before direct inversion of Eq. (4.3) [155]. Alternatively, Fourier-space traction reconstruction can also be combined with L2 regularization, which conveys additional robustness [37, 136, 218, 230].

4.1.2 L2 regularization for Fourier-transform traction cytometry

The classical approach to solve Eq. (4.3) is L2 regularization, which is also called Tikhonov regularization or ridge regression. L2 regularization is a robust procedure that suppresses noise and produces a smoothed traction field [200]. Here, the residual $\|\mathbf{u} - \mathbf{Mf}\|_2^2 = (\mathbf{Mf} - \mathbf{u})^{\mathrm{T}}(\mathbf{Mf} - \mathbf{u})$ is minimized together with the solution norm $\lambda_2\|\mathbf{f}\|_2^2 = \lambda_2\mathbf{f}^{\mathrm{T}}\mathbf{f}$. The factor λ_2 is called regularization parameter. The reconstructed traction $\hat{\mathbf{f}}$ satisfies

$$\hat{\mathbf{f}} = \underset{\mathbf{f}}{\mathrm{argmin}}\big[\|\mathbf{Mf} - \mathbf{u}\|_2^2 + \lambda_2\|\mathbf{f}\|_2^2\big]. \tag{4.4}$$

This approach can be employed for real-space TFM and in Fourier space, where the square norms can be calculated conveniently with Parseval's theorem. The proper choice of the regularization parameter λ_2 is critical in case accurate traction calculations are required. A popular heuristic for choosing the regularization parameter is based on a double-logarithmic plot of the solution residual *vs.* the traction norm for varying λ_2. Often, the plotted line resembles an "L" shape and the regularization parameter is chosen to lie in the corner of this curve, thus providing a trade-off between faithful reconstruction and smoothing [62]. However, this "L-curve criterion" is often of little use, in particular when the corner is either absent or can not be localized precisely on the double-logarithmic scale. Moreover, it has also been shown that the L-curve criterion can fail systematically [206]. Thus, the L-curve criterion is often complemented with other methods for finding the regularization parameter, such as cross-validation [200]. In any case, a manual parameter variation is mandatory to check the validity of the solution.

The Fourier traction forces in Eq. (4.3) can also be solved by using a similar L2 regularization method in real space [37, 136]. The integral Eq. (3.1) with the L2 regularization scheme is

written to be a corresponding variational equation in real space [37]:

$$\iint \left[\sum_{l,j} G_{li}(\mathbf{x}, \mathbf{x}'') G_{lj}(\mathbf{x}'', \mathbf{x}') f_j(\mathbf{x}') - \sum_j G_{ji}(\mathbf{x}, \mathbf{x}'') u_j(\mathbf{x}'') \right] d\mathbf{x}'' d\mathbf{x}'$$

$$+ \lambda^2 \int \sum_j H_{ji}(\mathbf{x}, \mathbf{x}') f_j(\mathbf{x}') d\mathbf{x}' = 0,$$

where H_{ji} is the square of R_2 introduced in Chapter 2. This equation can now be transformed into Fourier space and solved there:

$$\tilde{f}_{i\mathbf{k}} = \left\{ \sum_{l,j} \left[\sum_m \tilde{G}_{ml} \tilde{G}_{mi} + \lambda^2 \tilde{H}_{il} \right]^{-1} \tilde{G}_{jl} \tilde{u}_j \right\}. \tag{4.5}$$

For the regularization kernel $\tilde{H}_{il}$, we choose the identity matrix (0th order regularization). The whole expression on the right-hand side of Eq. (4.5) can, like in the former method L2 regularization in real space, be calculated at once, making the regularized method only marginally slower. The final step is, like above, the inverse Fourier transformation. Thus, this is a complete L2 regularization method for Fourier-transform traction cytometry.

4.1.3 Bayesian Fourier transform traction cytometry

Bayesian methods can be used to regularize data in a systematic and automated way. Our approach is based on an established iterative inference procedure [68]. In the first step, a model is fitted to the data. In the second step, the evidence for the chosen model is calculated. Traction computations with Bayesian L2 regularization (BL2) were first introduced as a real-space approach in Ref. [200]. Here we describe the adaptation of this method to Fourier-space traction calculation. It is assumed that the noise **s** in Eq. (4.3) has a Gaussian distribution with vanishing mean and a variance of $1/\beta$. Therefore, given a traction vector **f**, the likelihood of measuring a particular $2m \times 1$ displacement vector **u** is

$$p(\mathbf{u}|\mathbf{f}, \beta) = \frac{\exp[-\beta E_{\mathrm{u}}]}{Z_{\mathrm{u}}} = \frac{\exp[-\beta(\mathbf{Mf} - \mathbf{u})^{\mathbf{T}}(\mathbf{Mf} - \mathbf{u})/2]}{Z_{\mathrm{u}}}, \tag{4.6}$$

where $Z_{\mathrm{u}} = (2\pi/\beta)^m$. As a prior distribution for the $2n \times 1$ vector of traction forces **f** we choose a Gaussian distribution with variance $1/\alpha$ as

$$p(\mathbf{f}|\alpha) = \frac{\exp[-\alpha E_{\mathrm{f}}]}{Z_{\mathrm{f}}} = \frac{\exp[-\alpha \mathbf{f}^{\mathbf{T}} \mathbf{f}/2]}{Z_{\mathrm{f}}}, \tag{4.7}$$

where $Z_{\mathrm{f}} = (2\pi/\alpha)^n$. According to Bayes' rule, the posterior distribution of **f** is given by

$$p(\mathbf{f}|\mathbf{u}, \alpha, \beta) = \frac{p(\mathbf{u}|\mathbf{f}, \beta) p(\mathbf{f}|\alpha)}{p(\mathbf{u}|\alpha, \beta)} = \frac{\exp[-K(\mathbf{f})]}{p(\mathbf{u}|\alpha, \beta) Z_{\mathrm{u}} Z_{\mathrm{f}}}, \tag{4.8}$$

where $K(\mathbf{f}) = \beta E_{\mathrm{u}} + \alpha E_{\mathrm{f}}$ and $p(\mathbf{u}|\alpha, \beta) = \int d^{2n}\mathbf{f} \exp[-K(\mathbf{f})]/(Z_{\mathrm{u}} Z_{\mathrm{f}})$. To find the traction vector with the highest posterior probability, we maximize $p(\mathbf{f}|\mathbf{u}, \alpha, \beta)$ with respect to **f**. The calculation yields $\mathbf{f}_{\mathrm{MP}} = \underset{\mathbf{f}}{\mathrm{argmin}} \left[\beta \|\mathbf{Mf} - \mathbf{u}\|_2^2/2 + \alpha \|\mathbf{f}\|_2^2/2 \right]$, which is equivalent to our formula for L2 regularization in real space, when $\lambda_2 = \alpha/\beta$ [136].

Next, the values of the hyperparameters α and β have to be determined. In principle, both values can be found by maximizing the evidence $p(\alpha, \beta|\mathbf{u})$ that depends on the measured

displacements $\mathbf{u}$. However, the noise variance $1/\beta$ can also be estimated directly from the measurement uncertainty. Thereby, the maximization of $p(\alpha, \beta|\mathbf{u})$ can be reduced to a robust one-dimensional search for the optimal value of α. Bayes' law yields $p(\alpha, \beta|\mathbf{u}) = p(\mathbf{u}|\alpha, \beta)p(\alpha, \beta)/p(\mathbf{u})$. We next assume a uniform prior $p(\alpha, \beta) \simeq \text{const.}$ and note that evidence $p(\mathbf{u})$ does not play a role for the optimization. Thus, we only need to maximize $p(\mathbf{u}|\alpha, \beta) \sim \int \mathrm{d}^{2n}\mathbf{f} \exp[-K(\mathbf{f})]$ with respect to α. The integral can be analytically calculated by completing the square. On defining $\mathbf{A} = \alpha\mathbf{I} + \beta\mathbf{M}^{\mathbf{T}}\mathbf{M}$ one finds

$$p(\mathbf{u}|\alpha, \beta) = \frac{\int \mathrm{d}^{2n}\mathbf{f} \exp[-K(\mathbf{f})]}{Z_{\mathrm{u}}Z_{\mathrm{f}}} = \frac{(2\pi)^n (\det \mathbf{A})^{-1/2}}{Z_{\mathrm{u}}Z_{\mathrm{f}}} \exp[-\mathbf{K}(\mathbf{f}_{\mathrm{MP}})]. \tag{4.9}$$

Since $\mathbf{f}_{\mathrm{MP}}$ and $\mathbf{A}$ both depend on α, the maximization of Eq. (4.9) with respect to α needs to be done iteratively. This iteration can be sped up by performing the calculations in Fourier space. For notational clarity, we will write Fourier-space variables and derived quantities with a tilde. The Fourier-transformation of $\mathbf{f}_{\mathrm{MP}}$ yields $\tilde{\mathbf{f}}_{\mathrm{MP}} = (\tilde{\mathbf{M}}^{\dagger}\tilde{\mathbf{M}} + \alpha/\beta\mathbf{I})^{-1}\tilde{\mathbf{M}}^{\dagger}\tilde{\mathbf{u}}$ [37], where the complex transpose is indicated by a $\dagger$. Parseval's theorem allows convenient expression of Eq. (4.9) through Fourier-space variables. We have $\tilde{E}_{\mathrm{u}} = (\tilde{\mathbf{M}}\tilde{\mathbf{f}} - \tilde{\mathbf{u}})^{\dagger}(\tilde{\mathbf{M}}\tilde{\mathbf{f}} - \tilde{\mathbf{u}})/(2m)$, $\tilde{E}_{\mathrm{f}} = \mathbf{f}^{\dagger}\mathbf{f}/(2n)$, and $\tilde{\mathbf{A}} = \alpha\mathbf{I}/n + \beta\tilde{M}^{\dagger}\tilde{M}/m$. Using these expressions, the logarithm of the evidence, cf. Eq. (4.9), can be written as

$$\log p(\tilde{\mathbf{u}}|\alpha, \beta) = -\beta\tilde{E}_{\mathrm{u}}(\tilde{\mathbf{f}}_{\mathrm{MP}}) - \alpha\tilde{E}_{\mathrm{f}}(\tilde{\mathbf{f}}_{\mathrm{MP}}) - \frac{1}{2}\log(\det \tilde{\mathbf{A}}) + n\log\alpha + m\log\beta - m\log(2\pi). \tag{4.10}$$

This expression is evaluated numerically. The calculation of $\log(\det \tilde{\mathbf{A}})$ is done by a Cholesky decomposition of the positive matrix $\tilde{\mathbf{A}} = \mathbf{L}\mathbf{L}^T$ as the form $\log(\det(\mathbf{L}\mathbf{L}^{\mathbf{T}})) = 2\log\Pi_i L_{ii} = 2\Sigma_i \log(L_{ii})$ [200]. To determine the value of $\alpha = \hat{\alpha}$ that maximizes $\log p(\tilde{\mathbf{u}}|\alpha, \beta)$ we employ a golden-section search. Finally, the L2 regularization parameter follows as $\hat{\lambda}_2 = \hat{\alpha}/\beta$.

The calculation of the parameter value $\hat{\lambda}_2$ requires a well-defined maximum of the logarithmic evidence as a function of α. To assess whether this maximum exists, we investigate the condition $\frac{\mathrm{d}}{\mathrm{d}\alpha}\log p(\tilde{\mathbf{u}}|\alpha, \beta) = 0$. For evaluation of the derivatives of $\tilde{E}_{\mathrm{u}}(\tilde{\mathbf{f}}_{\mathrm{MP}})$ and $\tilde{E}_{\mathrm{f}}(\tilde{\mathbf{f}}_{\mathrm{MP}})$ we use that $n = m$ and that $\tilde{\mathbf{M}}$ commutes with $(\tilde{\mathbf{M}}^{\dagger}\tilde{\mathbf{M}} + \alpha/\beta\mathbf{I})^{-1}$ since the Fourier-transformed Green's function is a real, symmetric matrix. A straight forward calculation yields $\frac{\mathrm{d}}{\mathrm{d}\alpha}\tilde{E}_{\mathrm{u}}(\tilde{\mathbf{f}}_{\mathrm{MP}}) = -\lambda\frac{\mathrm{d}}{\mathrm{d}\alpha}\tilde{E}_{\mathrm{f}}(\tilde{\mathbf{f}}_{\mathrm{MP}})$. Therefore, the condition determining the maximum becomes $0 = \frac{\mathrm{d}}{\mathrm{d}\alpha}\log p(\tilde{\mathbf{u}}|\alpha, \beta) = -\tilde{E}_{\mathrm{f}}(\tilde{\mathbf{f}}_{\mathrm{MP}}) - \frac{1}{2n}\mathrm{Tr}[\tilde{\mathbf{A}}^{-1}] + \frac{n}{\alpha}$. We next perform a symbolic eigenvalue decomposition of $\tilde{\mathbf{M}}$ and denote the eigenvalues by $\{m_i\}$, the matrix of eigenvectors by $\mathbf{V}^T$, and define $\hat{u}_i = V_{ij}\tilde{u}_j$. The condition determining the maximum of the logarithmic evidence then reads

$$\frac{1}{2n}\sum_{i=1}^{2n} \frac{\beta\hat{u}_i^{\dagger}\hat{u}_i m_i^2}{(m_i^2 + \lambda_2)^2} = \frac{1}{2}\sum_{i=1}^{2n} \frac{m_i^2}{\lambda_2(m_i^2 + \lambda_2)}. \tag{4.11}$$

Solutions exist if the functions of λ_2 on the left hand side and on the right hand side of Eq. (4.11) cross each other. Both functions decrease monotonously with λ_2. However, for $\lambda_2 \to 0$ the left hand side remains finite while the right hand side diverges. Thus, Eq. (4.11) has a real solution if the left hand side becomes bigger than the right hand side for any $\lambda_2 \geq 0$. In the limit of $\lambda_2 \to \infty$, the condition for the occurrence of a maximum becomes $\frac{1}{n}\sum_{i=1}^{2n} \beta\hat{u}_i^{\dagger}\hat{u}_i m_i^2/\left(\sum_{j=1}^{2n} m_j^2\right) \geq 1$. For the TFM data, we find that the values of $\hat{u}_i^{\dagger}\hat{u}_i$ roughly

decrease with decreasing squared eigenvalues m_i^2 since the displacement magnitudes typically decrease with higher Fourier modes, as do the entries of $\tilde{\mathbf{M}}$. Assuming that the approximate ordering of m_i^2 and $\hat{u}_i^\dagger \hat{u}_i$ holds strictly, we can invoke Chebyshev's sum inequality to obtain $\frac{1}{n} \sum_{i=1}^{2n} \beta \hat{u}_i^\dagger \hat{u}_i \, m_i^2 / \left(\sum_{j=1}^{2n} m_j^2 \right) \geq \frac{1}{n^2} \sum_{i=1}^{n} \beta \hat{u}_i^\dagger \hat{u}_i$. Since for all reasonable TFM datasets the mean squared displacement is larger than the noise variance, we expect that $\frac{1}{n^2} \sum_{i=1}^{n} \beta \hat{u}_i^\dagger \hat{u}_i = \frac{2\beta}{n} \sum_{i=1}^{n} u_i^2 > 1$. Therefore, the condition for the occurrence of a maximum in $\log p(\tilde{\mathbf{u}}|\alpha, \beta)$ should be fulfilled for some $\lambda_2 > 0$. The resulting maximum is unique. In summary, a semi-quantitative argument supports the existence of a maximum of the logarithmic evidence $\log p(\tilde{\mathbf{u}}|\alpha, \beta)$ when appropriate TFM data is used. In our tests, a unique maximum was found for all datasets.

4.1.4 Software for traction force calculation

We provide a Matlab software package containing the presented Fourier-space methods for calculating traction forces. Note that the program requires the input of substrate displacements. Usually, these are quantified by measuring the lateral displacements of fluorescent marker beads in a stressed substrate with respect to the marker positions recorded in a stress-free state. The standard computational image analysis method for this task is called particle image velocimetry (PIV) and various well-established software packages are available [231–233]. Once the displacement data has been extracted, our program can be used to calculate the traction forces with standard L2 regularization or with Bayesian L2 regularization in Fourier space. The software is split into a routine for loading data and two routines for TFM. The routine "get input data" allows the user to select folders containing the data for the measured displacements, the noise, and for images. The required data structure in the file with the displacement data is illustrated in Fig. 4.2 (a). Parameters of the experimental setup, including the Young modulus and the Poisson ratio, also need to be provided. Next, the user can choose between "Regularization" and "Bayesian regularization", as shown in Fig. 4.2 (b) and (c). Selecting "Regularization" allows the choice of a regularization parameter, which is then held fixed for the whole sequence of images that are analyzed in the data set. For "Bayesian regularization", an optimal regularization parameter is selected automatically from the data set and the noise variance. The standard deviation of the noise can either be provided as an input or can be determined by manually selecting an image region that is far away from the cell, as illustrated in Fig. 4.2 (c). Once selected, the region used for determining the noise remains the same throughout the whole data set of multiple images. After pressing "Analyze sequence" the results are calculated and saved in automatically named files, see Fig. 4.2 (c). Since the regularization parameter λ_2 depends in our framework on the noise and the traction magnitudes, it should be adapted if the signal-to-noise level changes significantly. However, note that a change of the parameter within one image sequence is not always necessary, which reduces the computational effort and may be advantageous for data postprocessing.

The TFM software is a collection of MATLAB functions that are called via intuitive menus in a graphical user interface. Fig. 4.3 illustrates the purpose of the TFM software, namely to calculate spatial maps of the cellular traction forces from measured substrate displacements. The calculations can be done either with Regularized Fourier transform traction cytometry [37] or with Bayesian Fourier transform traction cytometry [200]. The difference between the two methods is that data smoothing is either done manually or automatically:
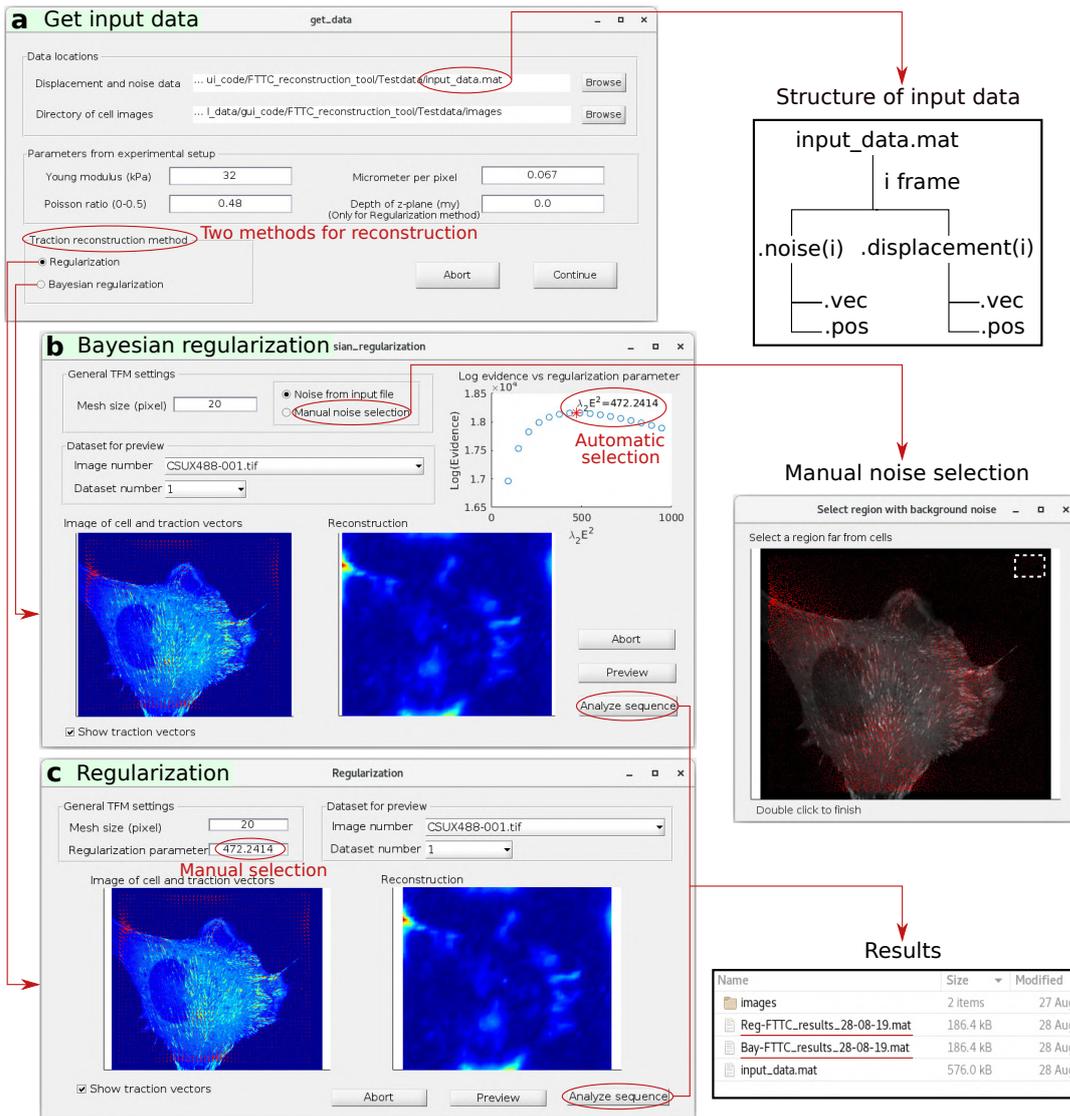
Figure 4.2: Graphical user interface of the provided software for regularized FTTC and BFTTC. (a) The "get data" interface allows users to input data locations and parameters of experimental setup. The data structure of the input files can handle a whole video sequence or individual traction recordings. (b) If the "Regularization" option is chosen, a regularization parameter in units of Pix^2 must be provided by the user. (c) If the option "Bayesian regularization" is chosen, the regularization parameter is automatically determined from the measured displacement data and its noise variance. A sample with displacement noise can either be provided with the input file or it can be determined from a manually selected region that is far away from the cell. A "Preview" button offers the possibility to visually inspect the solution before one presses "Analyze sequence" to calculate and save the results.

- Regularization $\rightarrow$ manual selection of a regularization parameter.

- Bayesian Regularization $\rightarrow$ automatic selection of a regularization parameter.

To calculate the traction forces, the following experimental data must be provided: (1) A list of two-dimensional substrate displacements [pixel]. The displacements can be measured, e.g., by tracking the motion of fluorescent marker beads in the substrate. (2) Optionally a sequence cell images in .tif or .jpg format corresponding to the time points at which the

displacements were measured.
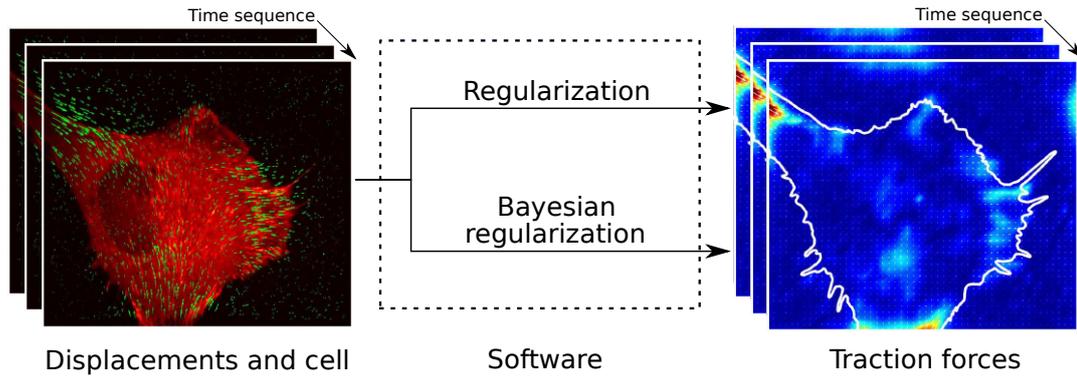
Figure 4.3: The software provides two methods to calculate cellular traction
forces from experimental data. The software provides the regularization (FTTC)
and Bayesian regularization (BFTTC) two approaches in Fourier space. The
software can analysis the time sequence of displacements and calculated the
time sequence traction force field.

This software runs in a MATLAB environment. Therefore, you need two requirements: a recent copy of MATLAB (https://ch.mathworks.com/), as well as the Image Processing Toolbox (https://ch.mathworks.com/products/image.html). The package has been developed with MATLAB R2017b and should work for versions R2010b and above. It can be run on Windows, Linux, and Macintosh OS X operating systems. The TFM software is freely available for download at GitHub: https://github.com/CellMicroMechanics.

4.2 Generation of synthetic test data and reconstruction quality measures

4.2.1 Generation of synthetic test data

To confirm that the Bayesian approach yields a correct estimate for the regularization parameter we employ synthetic data sets with known properties. In our first test series, we generate random traction fields by drawing individual traction vectors from Gaussian distributions with fixed variances, as illustrated in Fig. 4.4 (a-i) and (a-ii). The traction field is produced on a 50×50 grid with a Young modulus of $E = 10\,\mathrm{kPa}$ and a Poisson ratio of $\nu = 0.3$. For example, we employ a Gaussian traction distribution with a variance of $10^4\,\mathrm{Pa}^2$ and therefore $\alpha = 10^{-4}\,\mathrm{Pa}^{-2}$. After calculation of the displacements from the traction, Gaussian noise with a variance of $10^{-4}\,\mathrm{Pix}^2$ is added, thus $\beta = 10^4\,\mathrm{Pix}^{-2}$.

In the second test series, we construct synthetic data to study the reconstruction quality for localized traction patterns. As in previous work [37, 200], we assume that the traction is localized in circular spots, each having a constant traction magnitude. For every individual spot, the step-like traction profile can be integrated analytically to produce a displacement field. Due to the linearity of the problem, displacements from different spots can be added to produce the final result. Explicit formulas for the displacement field are provided in the supplementary of Ref. [200]. For generation of this data, we fix the Young modulus $E = 10\,\mathrm{kPa}$ and the Poisson ratio $\nu = 0.3$. The traction patterns consist of $10 - 20$ circular traction spots, as illustrated in Fig. 4.5 (a). The diameter of the spots is $2\,\mu\mathrm{m}$ and the mesh

size of the reconstructed traction is $0.5\,\mu\text{m}$. The traction force magnitude in the spots is randomly chosen in the range $[0 - 700]\,\text{Pa}$ and the sum of the x- and y components of the traction forces vanishes. To simulate the measurement uncertainty, Gaussian noise is added after calculation of the displacement field. The noise variance in the different samples is between 2% and 8% of the maximum absolute displacement value.

4.2.2 Reconstruction quality measures

For the synthetic test data with circular spots the traction force is exactly known. Therefore, we can qualitatively calculate the reconstruction errors. Here, we use four different error measures introduced and defined in Chapter 3. To provide simple definitions of the error measures, we rewrite the $2m \times 1$ traction vector $\mathbf{f}$ as a $m \times 2$ traction vector with the values $\mathbf{t} = \{t_x, t_y\}$ at every grid node. Real traction and reconstructed traction are denoted by $\mathbf{t}^{\text{true}}$ and $\mathbf{t}^{\text{recon}}$, respectively. Here, we only provide a simple explanation of each quality measure and the detail definitions can be found in Eq. (3.14 - 3.17) in Chapter 3.

- The Deviation of Traction Magnitude at Adhesions (DTMA) lies between -1 and 1 where 0 indicates a perfect average traction recovery and a negative or positive value implies underestimation or overestimation, respectively.

- The Deviation of Traction Magnitude in the Background (DTMB) is the normalized difference between the reconstructed and real traction magnitude outside the circular patches. The DTMB lies in the range $[0, 1]$ and a value close to 0 indicates low background noise in the reconstructed traction.

- The Signal to Noise Ratio (SNR) measures the detectability of a real signal within a noisy background. Its value ranges from 0 to infinity where a SNR that is much larger than unity indicates a good separation between traction and noise.

- The Deviation of the traction Maximum at Adhesions (DMA) measures how peak values of the traction over- or underestimate the true value. A DMA of 0 indicates that the local traction maxima in the reconstruction and in the original data are equal. Positive or negative values of the DMA imply that the maximum of traction is overestimated or underestimated.

4.3 Results

4.3.1 Validation of the method with synthetic data

To check whether the proposed method actually finds the correct regularization parameter, synthetic data sets with exactly known underlying distributions are required. Therefore, we create random traction patters with traction vectors at each grid point drawn from a Gaussian distribution. Exemplary data is shown in Fig. 4.4 (a-i). The calculated displacement field is then corrupted with a controlled level of noise, see Fig. 4.4 (a-ii). For the reconstruction, we search for the hyperparameter α that maximizes the log-evidence function, Eq. (4.10). As illustrated in Fig. 4.4 (a-iii), $\log p(\mathbf{u}|\alpha, \beta)$ has a unique, clear maximum. The regularization parameter determined from the optimization compares favorably with the true optimal parameter resulting from the distributions used for simulating the data, here

$\hat{\lambda}_2 = 9.3 \times 10^{-9} \, \text{Pix}^2/\text{Pa}^2 \simeq \alpha/\beta = 10^{-8} \, \text{Pix}^2/\text{Pa}^2$. Visual comparison of the traction patterns in Figs 4.4 (a-i,a-iv), as well as a comparison of the traction distributions in Fig. 4.4 (b), confirm that the Bayesian traction reconstruction yields correct results.
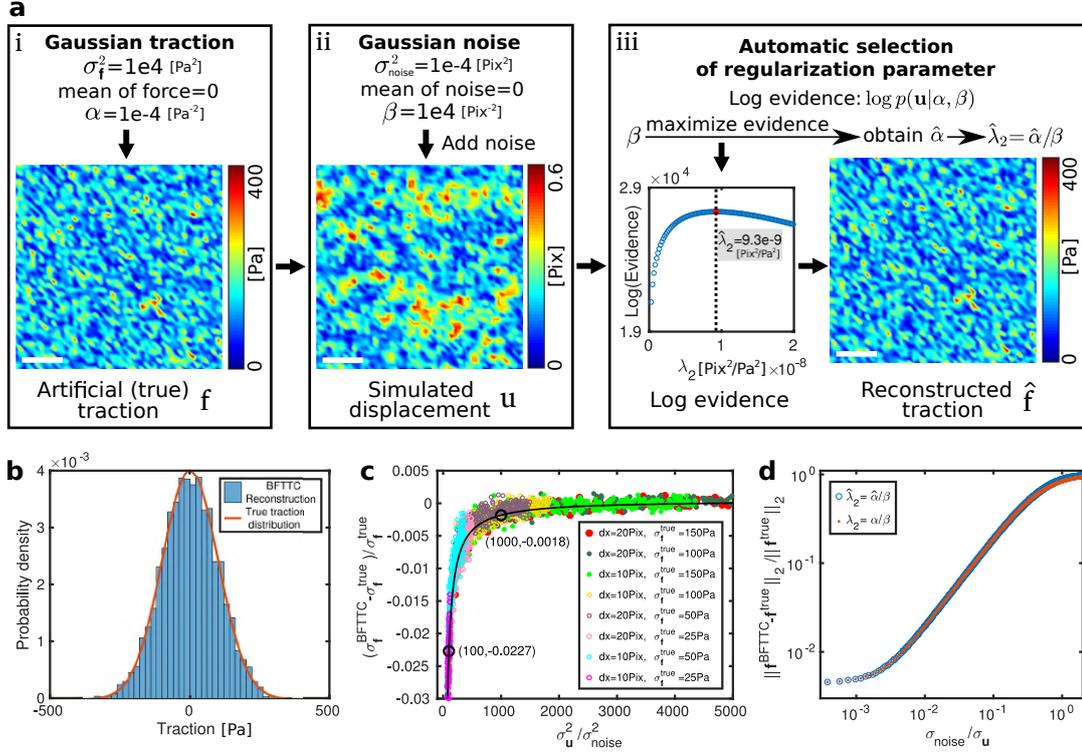
Figure 4.4: Validation of the Bayesian method for regularization parameter choice. (a,i) Traction force vectors, discretized on a quadratic mesh, are randomly chosen from a Gaussian distribution with fixed variance $\sigma_{\mathbf{f}}^2$. Space bar: $100 \, \text{Pix} = 10$ grid spacings on a 50×50 mesh. (a-ii) Using the prescribed traction as input, a displacement field is calculated and Gaussian noise with a variance σ_{noise}^2 is added. (a-iii) The regularization parameter is determined by localizing the maximum in the log evidence curve and traction forces are subsequently calculated. (b) Histogram of the tractions for the sample shown in (a). In the limit of weak noise, the histogram of the reconstructed traction matches the true traction distribution. (c) Relative difference between the standard deviation of the measured traction distribution $\sigma_{\mathbf{f}}^{\text{BFTTC}}$ and the width of true traction distribution $\sigma_{\mathbf{f}}^{\text{true}}$. The grid mesh sizes are denoted by dx. $\sigma_{\mathbf{u}}^2$ is the variance of the synthetic displacement data prior to corruption with noise. Increasing the noise level produces a measured (posterior) traction distribution that no longer agrees with the true traction distribution. (d) Mean error of the reconstructed traction as a function of the relative measurement uncertainty $\sigma_{\text{noise}}/\sigma_{\text{u}}$. The Bayesian estimate for the regularization parameter $\hat{\lambda}_2$ and the optimal regularization parameter α/β produce comparable errors for all noise levels.

Note that the measured (posterior) traction distribution does not agree with the original traction distribution when the noise magnitude is large. This fact corresponds to a deviation of the posterior Eq. (4.8) from the prior probability distribution. In Fig. 4.4 (c), we illustrate the difference between the measured traction distribution and the original traction distribution for the synthetic data. The relative difference of the traction standard deviations is plotted against the variance of the noise-free displacement field divided by the noise variance, $\sigma_{\text{u}}^2/\sigma_{\text{noise}}^2$. The relative difference of the standard deviation of the measured posterior and the

original traction distribution scales with the relative noise variance. Figure 4.4 (d) illustrates how the measurement uncertainty affects the mean traction error. For the experimentally relevant regime of measurement uncertainties, $0.01 \gtrsim \sigma_{\text{noise}}/\sigma_{\text{u}} \gtrsim 0.1$, the relative mean traction error is almost proportional to the relative measurement uncertainty $\sigma_{\text{noise}}/\sigma_{\text{u}}$. For
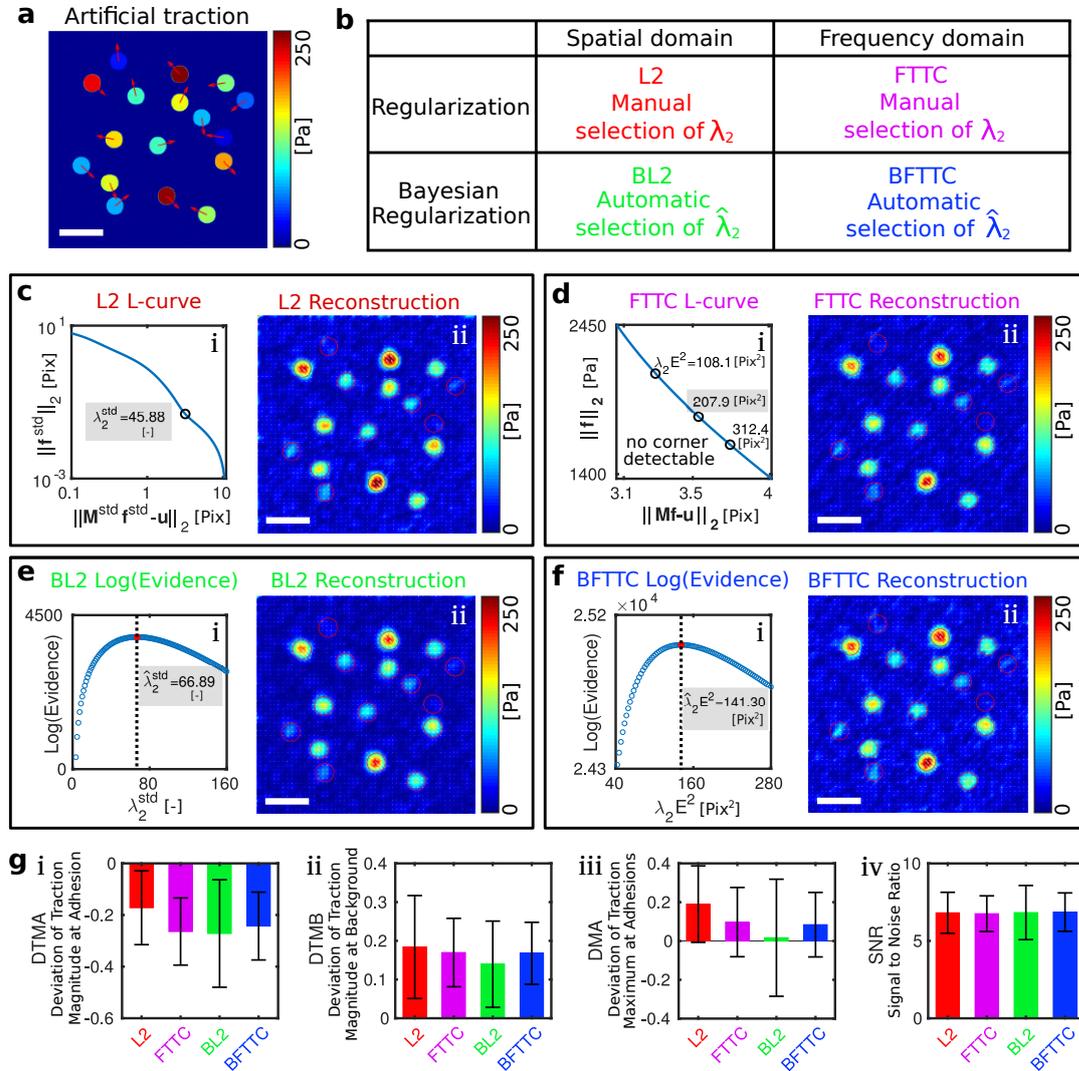
Figure 4.5: Reconstruction quality of BFTTC compared to other regularization methods. (a) Synthetic traction force pattern that is used for testing the reconstruction. Space bar: $5\,\mu$m. (b) Tabulated overview of the compared traction reconstruction methods. (c) Classical traction reconstruction in real space with L2 regularization. The L-curve shows a slight "corner", which is used to determine the value of the regularization parameter. Note that calculations in real space are done with standardized data [200], which renders the regularization parameter dimensionless. (d) Bayesian L2 regularization (BL2) in real space determines the regularization parameter value automatically. The automatically determined regularization parameter is close to the one predicted in (c) from the L-curve. (e) Classical, regularized Fourier transform traction cytometry (FTTC). The L-curve does not show a "corner", which makes it difficult to determine an appropriate regularization parameter. (f) Bayesian Fourier transform traction cytometry (BFTTC) determines an optimal regularization parameter automatically. (g) Comparison of the reconstruction quality measures in 8 synthetic data sets; error bars are the standard deviations of the samples. The reconstruction accuracy of all four methods is found to be similar.

very low measurement noise, the mean traction error is dominated by numerical inaccuracy and aliasing effects. Note that the Bayesian estimate for the regularization parameters $\hat{\lambda}_2$ produces errors that are close to the optimal errors resulting from regularization with the known parameters α/β for synthetic data.

4.3.2 Quality assessment of traction reconstruction with BFTTC

To quantify the reconstruction quality for localized traction patterns, we construct synthetic data consisting of circular spots of constant traction as shown in Fig. 4.6 (a). We employ two classical methods where the regularization parameter value is selected by the L-curve criterion, namely a real space calculation with L2 regularization and regularized Fourier transform traction cytometry (FTTC). The results are compared with the corresponding parameter-free approaches, namely Bayesian L2 regularization (BL2) in the real-space domain and Bayesian Fourier transform traction cytometry (BFTTC), see Fig. 4.5 (b). For the real-space TFM results shown exemplarily in Fig. 4.5 (c), the L-curve can have a visible corner. Note that the calculations in real space are done with standardized data [200], which renders the regularization parameter dimensionless. For the Bayesian real-space approach, illustrated in Fig. 4.5 (e), the logarithmic evidence always exhibits a clear maximum in our experience. The resulting optimal regularization parameter is usually close to the value from the L-curve criterion. However, in the Fourier-space approach, illustrated with the example in Fig. 4.5 (d), the L-curve often does not show a clear corner and it becomes challenging to select an appropriate regularization parameter. This weakness of the Fourier-space approach is overcome with BFTTC. As illustrated in Fig. 4.5 (f), the logarithmic evidence calculated in BFTTC has a pronounced maximum, which provides a clear criterion for the automated choice of the optimal regularization parameter.

Table 4.1: Computation time for different methods

Reconstruction method	L2	BL2	FTTC	BFTTC
Building of $\mathbf{M}$ or $\tilde{\mathbf{M}}$	23.3 h		0.07 s	
Traction reconstruction	67.4 s	338.8 s	0.06 s	3.1 s

The employed data set consists of a rectangular grid with 2500 displacement and traction vectors. Benchmark tests were done on a desktop computer equipped with 16 GB RAM and an Intel I5-7500 CPU (3.40 GHz).

To generate statistics on the performance of the different methods, we next record the traction reconstruction quality in 8 separate tests with different traction magnitudes and patterns. The resulting error norms show that all four methods offer similar traction reconstruction accuracies, see Figs. 4.5 (e) (i-iv). The most noticeable reconstruction errors are an underestimation of mean traction (negative DTMA) and a pronounced traction background (positive DTMB) [200]. The similarity in reconstruction accuracy is expected because all methods are based on L2 regularization and also make use of the same spatial grid for discretization. However, the numerical effort required for the four methods is very different. Table 4.1 summarizes the computation time required for building the coefficient matrices $\mathbf{M}$ or $\tilde{\mathbf{M}}$ and for reconstructing the traction forces. While $\tilde{\mathbf{M}}$ is rapidly built in Fourier space, the assembly of a large coefficient matrix $\mathbf{M}$ in real space requires can require many hours. Inferring the optimal regularization parameter requires additional computation time. Overall, real-space methods are not prohibitively slow but quite impractical for every-day use

by experimental scientists. BFTTC, however, requires acceptable computation times ranging from seconds to a few minutes.
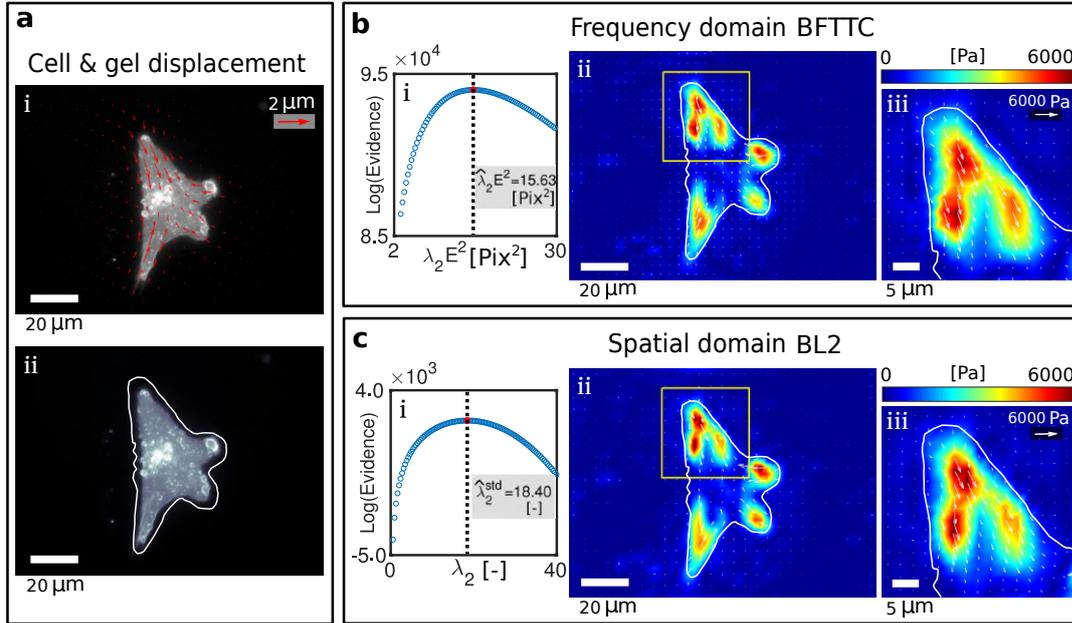
Figure 4.6: Test of Bayesian Fourier transform traction cytometry (BFTTC) using experimental data. (a,i-ii) Image of an adherent cell and the measured gel displacements. Only every 7-th displacement is shown for better visibility. The cell edge is outlined in white. (b) Results from traction calculation with BFTTC. (b,i) A plot of the logarithmic evidence reveals a clear maximum, which serves to determine the regularization parameter. (b,i-ii) Calculated traction forces. (c) Results from traction calculation with the real-space method BL2 for comparison with BFTTC. While the two methods produce similar results, traction fields calculated with BFTTC are slightly smoother than the fields calculated with the real-space method due to the different discretizations.

4.3.3 Application of BFTTC to experimental data

To provide an application example for BFTTC, we quantified the traction forces generated by a NIH 3T3 fibroblast on a polyacrylamide gel substrate. The experiments were done precisely as described in Ref. [136]. The gel substrate had a Young's modulus of $E = 32\,\mathrm{kPa}$ and a Poisson's ratio of $\nu = 0.48$. Figure 4.6 (a) shows the cell outline and an image of the fluorescent nanobeads that are embedded in the substrate. After recording images of the cell and the nanobeads, the cell was removed from the substrate to provide a stress-free reference for tracking the motion of the nanobeads. For visualization of the force-generating structures, the cell-substrate adhesions were labeled with GFP-paxillin. Focal substrate adhesions can be seen in Fig. 4.6 (b) as bright red spots. We estimate the variance of the noise in the displacement data by quantifying the displacement variance in a small region that is very far away from the cell and contains no systematic displacement. Plotting the logarithmic evidence as a function of α yields a curve with a clearly defined maximum, see Fig. 4.6 (c), which results in an unambiguous selection of the regularization parameter. Regularization with the thus chosen parameter produces a traction map with defined foci that co-localize with the GFP-labeled sites of focal adhesion, see Fig. 4.6 (d).

4.4   Summary and discussion for Chapter 4

Traction force microscopy is a popular technique for studying minute forces generated by biological cells, as well as wetting or frictional forces, on soft substrates. The technique is based on the measurement of substrate displacements below the specimen, which allows calculation of the traction forces. Usually, this calculation is done by solving an inverse linear problem involving elastic Green's functions. The procedure requires methods for noise suppression. Dealing with noise appropriately is an essential issue since the linear system can be ill-conditioned, which means that the noise can become amplified to an extent that the true solution is entirely degraded. A simple way to remove the effects of noise is to filter the displacement field prior to traction reconstruction. This strategy usually works if the linear problem is solved in Fourier space because the resulting linear system is sparse. An alternative strategy for dealing with noise is regularization, most popular is L2 regularization. With L2 regularization, spatial high-frequency variations in the data are suppressed, which leads to a robust solution of the inverse problem of calculating the traction. Regularization is more versatile than data filtering since it can deal with higher levels of noise, works both in real-space and Fourier-space approaches, and ensures robust reconstruction if non-standard Green's functions are employed, for example to take into account three-dimensional substrate topography and tractions. Regardless of the method, suppression of noise always reduces the spatial resolution. Optimal resolution of the fine details of the traction field can only be gained if the level of noise suppression is adapted for each sample. For L2 regularization, this adaptation is done by changing the regularization parameter, which is usually a manual process based on heuristics, which introduces a considerable degree of subjectivity in the resulting traction.

Here, we have introduced a Bayesian method for automatic inference of the L2 regularization parameter for traction reconstruction in Fourier space. Using synthetic data of different type, we demonstrate that Bayesian Fourier transform traction cytometry (BFTTC) is a fast and reliable method. Our tests show that BFTTC can handle large measurement noise. However, the noise- and displacement variances ideally satisfy $\sigma_{\mathrm{noise}}^2/\sigma_{\mathbf{u}}^2 \lesssim 0.01$ for accurate traction reconstruction. While the quality of traction reconstruction with BFTTC is comparable to other methods based on L2 regularization, the choice of the regularization parameter is now automated. Heuristics like the L-curve criterion, which is particularly ambiguous in Fourier space, are no longer required. The additional computation time required for determining the optimal regularization parameter in BFTTC is only a few seconds to minutes for large data sets. In our experience, the logarithmic evidence always exhibits a maximum that is sufficiently pronounced to yield a regularization parameter estimate. However, it is important to keep in mind that the algorithm is based on the assumption of a Gaussian prior distribution that is symmetric around the origin. Thus, the use of BFTTC is not recommended if the traction forces in the field of view do not balance each other. Moreover, if complex, non-Gaussian traction distributions, e.g., multi-modal distributions, are expected, it may be preferable to resort to Bayesian methods with prior distributions taylored to the specific problem in order to maximize the reconstruction quality.

To provide users from biology, physics, and materials sciences with an easy-to-use tool to analyze their TFM data, we implemented BFTTC as well as regularized FTTC as a Matlab package. The package comes with a user-friendly graphical interface, requires minimal knowledge of the algorithmic details, and is freely available [234].

Chapter 5

Data-driven, automated discovery of differential equations for physical processes

In this thesis, we mainly develop approaches to solve the inverse- or ill-posed inverse problem for two applications. In Chapters 3 and 4, we have focused on Traction Force Microscopy as an application of solution strategies for ill-posed, inverse problems. In this Chapter, we focus on another application for a problem with a similar mathematical structure, namely data-driven discovery of equations from measurement data.

We consider ordinary-, partial-, and stochastic differential equations describing physical processes. Classically, these equations can often be derived from first principles, for instance, conservation of mass, energy and momentum, and thermodynamic considerations. However, in complex systems as they occur for example in biophysics, climate science, and neuroscience, first principles determining the system behavior are hardly known since the systems are typically not close to equilibrium, processes can be highly nonlinear, and the dynamics can occur on multiple scales. In these cases, one can resort to phenomenological, effective descriptions that may include some level of coarse graining and are based on experimental data. Recently, increased computational power has made it possible to construct such models in an automated fashion, which is known as data driven discovery of governing equations.

5.1 Model setup and solution strategies

5.1.1 Inference of ordinary and partial differential equations from data

We assume a given set of measurement data that is recorded as a time-series, for example, a time-series of positions. In a data-driven approach to modeling such systems, we rely on the measurement data to automatically infer the a priori unknown equations that govern the observed process. In this thesis, inference is based on libraries of possible equations, which can include linear, nonlinear, and partial derivative terms. The simplest examples are ordinary differential equations (ODEs). To discover ODEs, we assume that we have a n-dimensional measurement data $\mathbf{Z} = [\mathbf{z}_1, \ \mathbf{z}_2, \ \ldots, \ \mathbf{z}_n] \in \mathbb{R}^n$ which is recorded at different

times $t_1, \ldots, t_m$. Thus, this data can be written in a matrix form

$$
\mathbf{Z} = [\mathbf{z}_1,\ \mathbf{z}_2,\ \ldots,\ \mathbf{z}_n] = \underset{\text{time}}{\Big\downarrow} \overset{\text{state}}{\longrightarrow}
\begin{bmatrix}
z_1(t_1) & z_2(t_1) & \cdots & z_n(t_1) \\
z_1(t_2) & z_2(t_2) & \cdots & z_n(t_2) \\
\vdots & \vdots & \ddots & \vdots \\
z_1(t_m) & z_2(t_m) & \cdots & z_n(t_m)
\end{bmatrix}.
$$

Using the finite difference approximation, the time derivative of the measurement data is written as

$$
\dot{\mathbf{Z}} = [\dot{\mathbf{z}}_1,\ \dot{\mathbf{z}}_2,\ \ldots,\ \dot{\mathbf{z}}_n] = \underset{\text{time}}{\Big\downarrow} \overset{\text{state}}{\longrightarrow}
\begin{bmatrix}
\dot{z}_1(t_1) & \dot{z}_2(t_1) & \cdots & \dot{z}_n(t_1) \\
\dot{z}_1(t_2) & \dot{z}_2(t_2) & \cdots & \dot{z}_n(t_2) \\
\vdots & \vdots & \ddots & \vdots \\
\dot{z}_1(t_m) & \dot{z}_2(t_m) & \cdots & \dot{z}_n(t_m)
\end{bmatrix},
$$

An ODE for each state at different times, e.g., $\mathbf{z}_1$, may be written as

$$
\dot{\mathbf{z}}_1 = f(\mathbf{z}_1,\ \mathbf{z}_1^2,\ \mathbf{z}_1\mathbf{z}_2,\ \cdots,\ \sin(\mathbf{z}_1),\ \cos(\mathbf{z}_1),\ \sin(\mathbf{z}_1\mathbf{z}_2),\ \cdots, c).
$$

where c represents a constant parameter. In the above equation, the left-hand side $\dot{\mathbf{z}}_1$ is known and the right-hand side $f(\cdot)$ can be written as a library matrix $\mathbf{\Theta}(\mathbf{z})$ multiplied with a sparse vector $\mathbf{w}_1$. Thus, we obtain

$$
\dot{\mathbf{z}}_1 = \mathbf{\Theta}(\mathbf{z})\mathbf{w}_1, \tag{5.1}
$$

where the library matrix $\mathbf{\Theta}(\mathbf{z})$ is constructed as assumed terms from the measurement data, e.g., included $\mathbf{z}_1$ and $\sin(\mathbf{z}_2)$. Note that the library does not include the differential term for ODEs. For example, the library matrix $\mathbf{\Theta}(\mathbf{z})$ may consist of constant, polynomial and trigonometric terms

$$
\mathbf{\Theta}(\mathbf{z}) =
\begin{bmatrix}
| & | & | & | & & | & | & | & | & \\
1 & \mathbf{z} & \mathbf{z}^{P_2} & \mathbf{z}^{P_3} & \cdots & \sin(\mathbf{z}) & \cos(\mathbf{z}) & \sin(2\mathbf{z}) & \cos(2\mathbf{z}) & \cdots \\
| & | & | & | & & | & | & | & | &
\end{bmatrix},
$$

where $\mathbf{z}^{P_2}$ and $\mathbf{z}^{P_3}$ are the higher-order polynomials. For instance, $\mathbf{z}^{P_2}$ is written in the following form:

$$
\mathbf{z}^{P_2} =
\begin{bmatrix}
z_1^2(t_1) & z_1(t_1)z_2(t_1) & \cdots & z_2^2(t_1) & z_2(t_1)z_3(t_1) & \cdots & z_n^2(t_1) \\
z_1^2(t_2) & z_1(t_2)z_2(t_2) & \cdots & z_2^2(t_2) & z_2(t_2)z_3(t_2) & \cdots & z_n^2(t_2) \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
z_1^2(t_m) & z_1(t_m)z_2(t_m) & \cdots & z_2^2(t_m) & z_2(t_m)z_3(t_m) & \cdots & z_n^2(t_m)
\end{bmatrix}.
$$

In Eq. (5.1), $\dot{\mathbf{z}}_1$ and $\mathbf{\Theta}(\mathbf{z})$ can be calculated from the measurement data. Given $\dot{\mathbf{z}}_1$ and $\mathbf{\Theta}(\mathbf{z})$, our aim is to calculate the sparse vector $\mathbf{w}_1$, which is non-zero for those components that correspond to libary terms that appear in the inferred ODE. This problem also becomes an inverse problem that can be often solved by using sparse regression learning [39, 40]. The approach used for the first variable $\mathbf{z}_1$ can be used analogously to infer governing equations

for the variables $\mathbf{z}_2, \ldots, \mathbf{z}_n$.

If the library includes partial derivatives of a variable, the library can be used for inference of PDE's. The measurement data for inference of PDEs needs to allow for the estimation of at least two derivatives, for example, with respect to time and space. Usually, measurements consist therefore of space-time series recordings of system variables. For example, the n-dimensional state $\check{\mathbf{Z}}$ is written as

$$
\check{\mathbf{Z}} = \begin{array}{c} \text{time} \end{array} \left[\begin{array}{cccc} z_1(x_1,t_1) & z_2(x_1,t_1) & \cdots & z_n(x_1,t_1) \\ z_1(x_1,t_2) & z_2(x_1,t_2) & \cdots & z_n(x_1,t_2) \\ \vdots & \vdots & \ddots & \vdots \\ z_1(x_1,t_m) & z_2(x_1,t_m) & \cdots & z_n(x_1,t_m) \end{array} \right]
$$

where the data in the x-space ranges from x_1 to x_r. The derivative with respect to x is also calculated by using a finite difference approximation. For example, the first derivative $(\check{\mathbf{Z}})_x$ is written as

$$
(\check{\mathbf{z}})_x = \begin{array}{c} \text{time} \end{array} \left[\begin{array}{cccc} (z_1)_x(x_1,t_1) & (z_2)_x(x_1,t_1) & \cdots & (z_n)_x(x_1,t_1) \\ (z_1)_x(x_1,t_2) & (z_2)_x(x_1,t_2) & \cdots & (z_n)_x(x_1,t_2) \\ \vdots & \vdots & \ddots & \vdots \\ (z_1)_x(x_1,t_m) & (z_2)_x(x_1,t_m) & \cdots & (z_n)_x(x_1,t_m) \end{array} \right]
$$

Using the same approach, the higher derivatives $(\check{\mathbf{Z}})_{xx}$, $(\check{\mathbf{Z}})_{xxx}$, ... can also be constructed from the measurement data. Inclusion of these derivative terms into the library $\boldsymbol{\Theta}^P(\check{\mathbf{Z}})$ yields a matrix of the form

$$
\boldsymbol{\Theta}^P(\check{\mathbf{z}}) = \left[\begin{array}{ccccccccccc} | & | & | & & | & | & & | & | & & | \\ \mathbf{1} & \check{\mathbf{z}} & \check{\mathbf{z}}^{P_2} & \cdots & (\check{\mathbf{z}})_x & (\check{\mathbf{z}})_{xx} & \cdots & \check{\mathbf{z}}(\check{\mathbf{z}})_x & \cdots & \check{\mathbf{z}}^{P_3}(\check{\mathbf{z}})_{xxx} \\ | & | & | & & | & | & & | & | & & | \end{array} \right],
$$

where the highest derivatives of the library is in this example given by the cubed nonlinearities $\check{\mathbf{z}}^{P_3}$ multiplied $(\check{\mathbf{z}})_{xxx}$. With this library, inference of the governing equations again becomes a linear optimization problem. For example, the PDE governing $\check{\mathbf{z}}_1$ can be written as

$$
\dot{\check{\mathbf{z}}}_1 = \boldsymbol{\Theta}^P(\check{\mathbf{z}})\mathbf{w}_1, \tag{5.2}
$$

which has the same form as in Eq. (5.1). As above, the resulting equation systems are large and possibly involve a matrix with a large condition number. Next, we will discuss how one can introduce stochasticity into the inferred equations for the observed variables.

5.1.2 Inference of stochastic differential equations from data

Langevin equations are a popular class of stochastic differential equations (SDEs) that are used to describe the time evolution of state variables, $\mathbf{X}(t)$ [173, 235]. In general, Langevin equations take the form

$$\frac{dX_i(t)}{dt} = \underbrace{g_i(\mathbf{X}(t), t)}_{\text{deterministic part}} + \underbrace{h_{ij}(\mathbf{X}(t), t)dW_j(t)}_{\text{stochastic part}}, \tag{5.3}$$

where we employ the sum convention and $X_i(t)$ denotes the state of the system at time t. The function g_i represents deterministic forces that determine the evolution of X_i. The function matrix h_{ij} represents the strength of stochastic perturbations to the trajectory of X_i. Frequently, the stochastic forcing is assumed to result from a Wiener process with a driving noise source $\Gamma_j(t)$ and $dW_j = \Gamma_j(t)dt$. The noise is then assumed to be δ-correlated and to have a Gaussian distribution with vanishing mean

$$\langle \Gamma_j(t) \rangle = 0,$$
$$\langle \Gamma_i(t)\Gamma_j(t') \rangle = \delta_{ij}\delta(t - t').$$

In this thesis, the states of the trajectory $\mathbf{X}$ are calculated by making use of Ito's interpretation of stochastic integrals [173]. The Fokker-Planck equation of the evolution of a probability density function $f_{X_i}(\mathbf{x}, t)$ for state X_i, which corresponds to Eq. (5.3) can be written as

$$\frac{\partial f_{X_i}(\mathbf{x}, t)}{\partial t} = \hat{L}(\mathbf{x}, t)f_{X_i}(\mathbf{x}, t),$$

where $\hat{L}(x_m, t)$ is called the Fokker-Planck operator and m represents the number of states. The operator is written as [175]

$$\hat{L}(\mathbf{x}, t) = -\frac{\partial}{\partial x_i}D_i^{(1)}(\mathbf{x}, t) + \frac{\partial^2}{\partial x_i \partial x_j}D_{ij}^{(2)}(\mathbf{x}, t),$$

where the Kramers-Moyal (KM) coefficients read as

$$D_i^{(1)}(\mathbf{x}, t) = \lim_{\tau \to 0}\frac{1}{\tau}\langle [X_i(t + \tau) - X_i(t)]\rangle_{\mathbf{X}(t)=\mathbf{x}}, \tag{5.4}$$

$$D_{ij}^{(2)}(\mathbf{x}, t) = \lim_{\tau \to 0}\frac{1}{2\tau}\langle [X_i(t + \tau) - X_i(t)][X_j(t + \tau) - X_j(t)]\rangle_{\mathbf{X}(t)=\mathbf{x}}. \tag{5.5}$$

Here, $D_i^{(1)}(\mathbf{x}, t)$ and $D_{ij}^{(2)}(\mathbf{x}, t)$ are also respectively referred to as the *drift coefficient* and *diffusion coefficient*. τ is a small time step. In the Langevin equation, the functions g_i and h_{ij} are related to these KM coefficients as

$$g_i(\mathbf{x}, t) = -\nabla_{x_i}U(\mathbf{x}, t) = D_i^{(1)}(\mathbf{x}, t),$$
$$h_{ij}(\mathbf{x}, t) = \sqrt{2D_{ij}^{(2)}(\mathbf{x}, t)},$$

where $U(\mathbf{x}, t)$ is a potential energy function, for example, a double-well-, three-well-, or four-well potential.

In the following, we focus on the case that the drift- and diffusion coefficients do not depend explicitly on time. To estimate the drift- and diffusion coefficients for one state variable X

from the measurement data $\{\mathbf{X}_l\}_{l=1,\ldots,N+1}$ with the help of Eq. (5.5), we need to construct two new sequences $\mathbf{D}$ and $\mathbf{F}$ [174]

$$\{\mathbf{D}_l\}_{l=1,\ldots,N} = \left\{ \frac{\mathbf{X}_{l+1} - \mathbf{X}_l}{s} \right\}_{l=1,\ldots,N+1},$$

$$\{\mathbf{F}_l\}_{l=1,\ldots,N} = \left\{ \frac{(\mathbf{X}_{l+1} - \mathbf{X}_l)^2}{2s} \right\}_{l=1,\ldots,N+1},$$

where s is the time step which we assume to be sufficiently small. To reduce the effects of minor observation errors in these trajectories, we utilize data binning to group more or less continuous values into a number of Q bins

$$\{\mathbf{X}_l\}_{l=1,\ldots,N} \mapsto \{x_i\}_{i=1,\ldots,Q} = \mathbf{x},$$

$$\{\mathbf{D}_l\}_{l=1,\ldots,N} \mapsto \{d_i\}_{i=1,\ldots,Q} = \mathbb{D},$$

$$\{\mathbf{F}_l\}_{l=1,\ldots,N} \mapsto \{f_i\}_{i=1,\ldots,Q} = \mathbb{F}.$$

We use these binned data to identify the drift coefficient $D_i^{(1)}(x)$ and diffusion coefficient $D_{ij}^{(2)}(x)$. Note that these coefficients are only a function of x in this thesis. To achieve the identification, we construct a library $\mathbf{\Theta}_{Q \times K}$ from the binned data $\mathbf{x}$, where K is the number of terms in the library, for example, $\mathbf{\Theta}_{Q \times K} = [\mathbf{1}, \ \mathbf{x}, \ \mathbf{x}^2, \ \ldots, \ \sin(\mathbf{x})]$. The drift- and diffusion coefficients in Eq. (5.5) can be respectively written as two linear equations

$$\mathbb{D} = \mathbf{\Theta}\mathbf{W}_1, \tag{5.6a}$$

$$\mathbb{F} = \mathbf{\Theta}\mathbf{W}_2, \tag{5.6b}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are sparse weight vectors. These two equations are the data-driven model for SDEs. In these equations, given $\mathbb{D}$, $\mathbb{F}$ and $\mathbf{\Theta}$, $\mathbf{W}_1$ and $\mathbf{W}_2$ need to be calculated. Thus, the mathematical formulation becomes an inverse problem.

So far, we have not discussed an essential part of the data-driven identification of differential equations, namely, the solution strategies typically employed for the resulting linear problems of type $\mathbf{g} = \mathbf{\Phi}\mathbf{w} + \mathbf{s}$. Various approaches have been developed for solving this problem in the context of data-driven model inference, for example, sequential thresholded least squares algorithm [39], TrainSTRidge [40], Stepwise Sparse Regressor [174] and Thresholded sparse Bayesian regression [172]. All these approaches have in common that they require at least one threshold or regularization parameter that must be provided based on expert system knowledge of the user. In this thesis, we develop a new algorithm which combines sparse Bayesian learning [69, 70, 172] with automatic thresholding methods for selecting parameters [40]. We call our algorithm automatic threshold sparse Bayesian learning (ATSBL). This approach eliminates the need for manual fine-tuning of parameters to correctly infer governing differential equations from measured data.

5.1.3 Automatic threshold sparse Bayesian learning

Sparse Bayesian learning is a robust and reliable approach to obtain a sparse solution vector for inverse and ill-posed problems [69, 72]. In Ref. [70], an accelerated algorithm has been suggested which maximizes the evidence in linear systems by iterative addition and removal of candidate basis functions in the solution space. A similar algorithm has been

Function: ATSBL($\boldsymbol{\Theta}$, $\mathbf{g}$, d_{tol}, n_{iters})

%% Split the measurement data into training and test parts

$\boldsymbol{\Theta} \mapsto [\boldsymbol{\Theta}^{\text{train}}, \boldsymbol{\Theta}^{\text{test}}]$;   % e.g., 80% for training, 20% for test

$\mathbf{g} \mapsto [\mathbf{g}^{\text{train}}, \mathbf{g}^{\text{test}}]$;

%% Initial setting

$[\mathbf{U}, \mathbf{s}, \mathbf{V}] = svd(\boldsymbol{\Theta}^{\text{train}})$;   % SVD for approximation of the inverse of $\boldsymbol{\Theta}^{\text{train}}$

$t = 1e - 4$;   %% Small threshold for $\mathbf{s}$ matrix

for ($i = 1$; $i < size(\mathbf{s})$; $i = i + 1$ **) {**
> **if** $s_i > t$ **then**
>> $s_i = 1/s_i$;
>
> **else**
>> $s_i = 0$;
>
> **end**

}

$\boldsymbol{\Theta}^{\text{train}}_{inv} = \mathbf{VsU}'$;

$\mathbf{w}_{\text{best}} = \boldsymbol{\Theta}^{\text{train}}_{inv} \mathbf{g}^{\text{train}}$;

$\eta = 10^{-3}\kappa(\Theta)$;   % $\kappa(\Theta)$ is the condition number of the matrix

$e_{\text{best}} = \|\boldsymbol{\Theta}^{\text{test}}\mathbf{w}_{\text{best}} - \mathbf{g}^{\text{test}}\|_2^2 + \eta\|\mathbf{w}_{\text{best}}\|_0$;   % Assign initial error predictor

$tol = d_{\text{tol}}$;   % Assign initial threshold

%% Automatically estimate the sparse vector and the threshold

for ($i = 0$; $i < n_{iters}$; $i = i + 1$ **) {**
> $\sigma^2 = std(\mathbf{g}^{\text{train}})^2/1e2$;   % Initial noise variance
>
> $\mathbf{w} =$ FastLaplace($\boldsymbol{\Theta}^{\text{train}}, \mathbf{g}^{\text{train}}, \sigma^2$);   % Calculate the sparse vector from BCSL
>
> $e = \|\boldsymbol{\Theta}^{\text{test}}\mathbf{w} - \mathbf{g}^{\text{test}}\|_2^2 + \eta\|\mathbf{w}\|_0$;   % Calculate current error predictor
>
> %% Automatically obtain the threshold
>
> **if** $e \leq e_{best}$ **then**
>> %% Decrease the error
>>
>> $e_{\text{best}} = e$;
>>
>> $tol = tol + d_{\text{tol}}$;
>
> **else**
>> %% Tolerance too high
>>
>> $tol = \max([0, tol - 2d_{\text{tol}}])$;
>>
>> $d_{\text{tol}} = \frac{2d_{\text{tol}}}{n_{\text{iters}}-i}$;
>>
>> $tol = tol + d_{\text{tol}}$;
>
> **end**
>
> %% Use the threshold
>
> **for (** $i = 1$; $i < size(\mathbf{w})$; $i = i + 1$ **) {**
>> **if** $w_i < tol$ **then**
>>> $\boldsymbol{\Theta}^{\text{train}}(:, i) = 0$;
>>
>> **end**
>
> **}**
>
> %% One of convergence conditions
>
> **if** $\|\mathbf{w}_{best} - \mathbf{w}\|_2/\|\mathbf{w}_{best}\|_2 < 1e - 9$ **then**
>> return $\mathbf{w}_{\text{best}}$;
>
> **end**
>
> $\mathbf{w}_{\text{best}} = \mathbf{w}$;

}

return $\mathbf{w}_{\text{best}}$;

Algorithm 1: Pseudocode for automatic threshold sparse Bayesian learning (ATSBL)

employed for compressive sensing using Laplace priors [72], which is called Bayesian compressive sensing using Laplace priors (BCSL). Details of the BCSL algorithm can be found in Chapter 2.

Since a large library is usually constructed, the solution vector generated by the BCSL algorithm typically still contains quite a few non-vanishing but small entries. To improve the identification of governing equations, a thresholding procedure is employed for removing

those entries in the library that have very small weight [39, 172, 174]. Usually, the threshold needs to be manually provided by the user. However, in Ref. [40], a procedure was proposed to estimate the threshold automatically based on test data (20% of the data) in a training process. In our work, we combine BCSL with this training method. Details of the proposed algorithm which we call automatic threshold sparse Bayesian learning (ATBSL) are shown in Algorithm 1.

For ATSBL, we provide the input data, namely the vector $\mathbf{g}$ and the library matrix $\mathbf{\Theta}$, an initial increment for the threshold d_{tol} and the maximum number of iterations n_{iters}. We first split the input data $\mathbf{g}$ and $\mathbf{\Theta}$ respectively into two parts that are $\mathbf{\Theta}^{\text{train}}$, $\mathbf{\Theta}^{\text{test}}$, $\mathbf{g}^{\text{train}}$ and $\mathbf{g}^{\text{test}}$. For example, the training data is 80% and the test data is 20%. Then, we calculate some initial parameters, for example, an initial error predictor e_{best} and an initial threshold tol. Next, the core part of the training algorithm is a loop to automatically calculate the sparse vector $\mathbf{w}$ and the threshold tol. Here, we first employ the FastLaplace function from BCSL to obtain $\mathbf{w}$ from the training part of data, where the function is shown in Appendix C. Then, a new error predictor e can be calculated as [40]

$$e = \|\mathbf{\Theta}^{\text{test}}\mathbf{w} - \mathbf{g}^{\text{test}}\|_2^2 + \eta\|\mathbf{w}\|_0,$$

where $\eta = c\kappa(\mathbf{\Theta})$, where c is a constant parameter, e.g., $c = 1e-3$. κ is the condition number. If the estimated error is decreased, we increase the threshold tol, otherwise the threshold is reduced. Finally, the entries in the library matrix $\mathbf{\Theta}$ corresponding to entries in the solution $\mathbf{w}$ that are smaller than the threshold are set to zero. This training process will be stopped when the convergence conditions are satisfied. The final solution $\mathbf{w}_{\text{best}}$ is the sparse vector that determines the terms in the governing differential equations, ODEs, PDEs and SDEs.

5.2 Results

5.2.1 Identification of ordinary and partial differential equations

5.2.1.1 Identification of a chaotic Lorenz system

We employ a Lorenz system to demonstrate the identification of ODEs from data [236]. The Lorenz system is a paradigm for chaotic behavior and is written as

$$\dot{x} = a(y - x),$$
$$\dot{y} = x(b - z) - y,$$
$$\dot{z} = xy - cz,$$

where the standard parameters are $a = 10$, $b = 28$ and $c = 3/8$.

We solve the equations numerically to obtain trajectory data $\mathbf{X} = \mathbf{x}_i = (\mathbf{x}, \mathbf{y}, \mathbf{z})$, where the initial values are chosen as $[x_0, \ y_0, \ z_0] = [-8, \ 7, \ 27]$ and the time step is chosen as $\Delta t = 0.001$ [39]. We consider a short-time trajectory $0 \leq t \leq 20$ and a long-time trajectory, $0 \leq t \leq 250$, respectively illustrated in Fig. (5.1) (b-Top) and (b-Bottom). In these plots, we see that the chaotic system involves two attractors. To identify the Lorenz equations from the data, we utilize a library of ODEs for each of the components, x, y and z. The library $\mathbf{\Theta}(\mathbf{X})$ and $\dot{\mathbf{x}}_i$ are constructed from the simulated trajectory. The library is written as $\mathbf{\Theta}(\mathbf{X}) = [\mathbf{1} \ \mathbf{x} \ \mathbf{y} \ \mathbf{z} \ \dots \ \mathbf{z}^4]$, which includes 56 terms, which is the library size also employed

a

True ODE Lorenz system	Identified system $\iota = 1$	Identified system $\iota = 10$
$\dot{x} = -10x + 10y,$	$\dot{x} = -9.999x + 10.000y,$	$\dot{x} = -9.998x + 9.995y,$
$\dot{y} = 28x - y - xz,$	$\dot{y} = 28.000x - 0.998y - 1.000xz,$	$\dot{y} = 28.040x - 1.004y - 1.001xz,$
$\dot{z} = xy - 8/3z$	$\dot{z} = 1.000xy - 2.667z$	$\dot{z} = 1.000xy - 2.667z$

b

Full simulation Identified system, $\iota = 1$ Identified system, $\iota = 10$
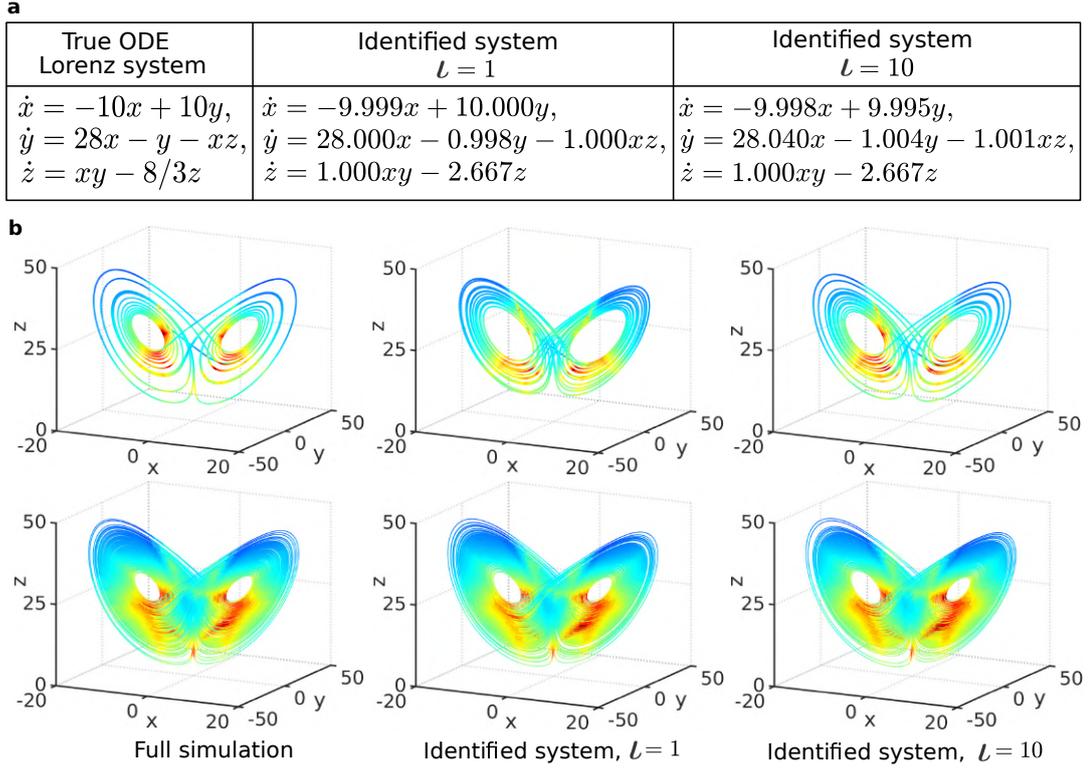
Figure 5.1: Exemplary results for data-driven identification of the ordinary differential equations for the Lorenz system. (a) The original Lorenz equations and the identified corresponding equations for different noise levels are reported in the table. (b) Numerically calculated trajectories for time from $t = 0$ to $t = 20$ (Top) and $t = 0$ to $t = 250$ (Bottom) with a time step $\Delta t = 0.001$. The trajectories for the original system are shown on the left. The trajectories calculated with the identified equations are shown in the middle and on the right for the noise level $\iota = 1$ and $\iota = 10$, respectively. The colour-map represents the local sensitivity.

in previously published approaches [39]. Three linear equations are used to identify the Lorenz system, for example, $\dot{\mathbf{x}} = \mathbf{\Theta}(\mathbf{X})\mathbf{w} + \mathbf{s}$, where $\mathbf{s}$ is the type I Gaussian noise. Here, we consider two noise levels where the Gaussian noises have standard deviations of $\iota = 1$ and $\iota = 10$.

The governing Lorenz equations are identified correctly with ATSBL, as illustrated in Fig. (5.1) (a). To quantitatively assess quality of the equation identification, we introduce a *deviation of identified coefficient* (DIC) and a *maximum deviation of identified coefficient* (MDIC) in this thesis,

$$\text{DIC} = \frac{1}{R} \sum_i \frac{\|p_i - p_i'\|_2}{\|p_i'\|_2} \times 100, \tag{5.7}$$

$$\text{MDIC} = \max_i \frac{\|p_i - p_i'\|_2}{\|p_i'\|_2} \times 100, \tag{5.8}$$

where p_i is a coefficient of one term in the identified equation and p_i' is the corresponding coefficient in the real equation. R represents the number of coefficients. The sums only run over non-zero coefficients. DIC and MDIC lie in the range $[0, \infty]$ where 0 indicates a perfectly identified equation. For the Lorenz equations, when $\iota = 1$, DIC is equal to 0.028% and MDIC is equal to 0.2%. When $\iota = 10$, DIC and MDIC are respectively equal to 0.1% and 0.5%.

	Correct PDE	$u_t + uu_x - 0.1u_{xx} = 0$
	Identified PDE (clean data)	$u_t + 1.0000uu_x - 0.1000u_{xx} = 0$
	Identified PDE (5% noise)	$u_t + 0.9997uu_x - 0.1001u_{xx} = 0$
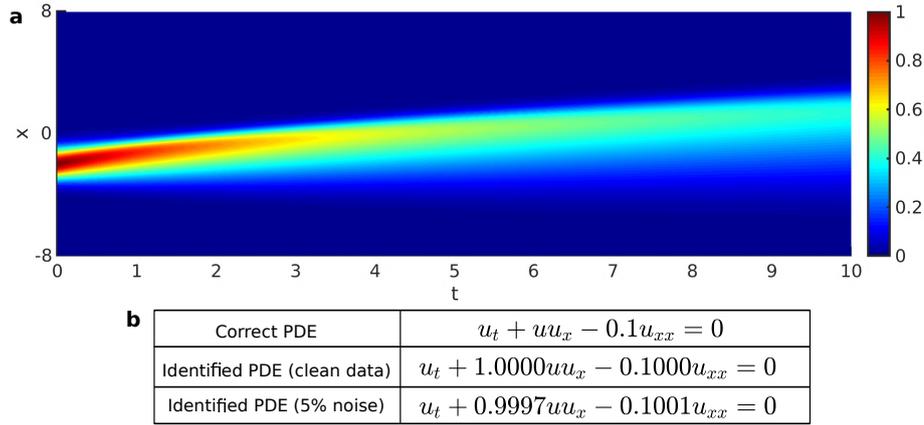
Figure 5.2: The identification of partial differential equations for Burgers' equation. (a) 101 time snapshots of a solution calculated from the Burgers' equation. The time step is $\Delta t = 0.1$. The initial condition is a Gaussian distribution on one-dimensional space x, discretized on a uniform grid with 256 elements. The Gaussian distribution propagates as a travelling wave. (b) The original and identified Burgers' equations.

5.2.1.2 Identification of one-dimensional partial differential equations

Burgers' equation   Burgers' equation is wildly employed in mathematics and physics to describe wave propagation. This fundamental PDE has the form of a Navier-Stokes equation for the velocity field with omitted pressure gradient term. Solutions to Burgers' equation do not exhibit turbulent behavior. The Burgers' equation is written as

$$u_t + uu_x - 0.1u_{xx} = 0.$$

We employ as data set for inference the velocity field $u(x, t)$ provided with Ref. [40], which consists of snapshots of a travelling wave, illustrated in Fig. (5.2) (a). The time step is $\Delta t = 0.1$ with 101 total steps. The spatial discretization is done on an uniform-mesh with 256 elements in one-dimensional space $-8 \leq x \leq 8$. The initial condition of the wave is a Gaussian distribution.

Using the time-space series data $u(x, t)$, we can easily calculate u_t, u_x, u_{xx}, ... by using a finite difference approximation. With these values, we construct the library matrix being $\boldsymbol{\Theta}$, which has 16 terms with the highest derivative term u_{xxx}. Again, we assume $\mathbf{u}_t = \boldsymbol{\Theta}\mathbf{w} + \mathbf{s}$, where $\mathbf{s}$ is a type I Gaussian noise. Using ATSBL, we identify the Burgers' equation, as illustrated in Fig. (5.2) (b). For clean data, DIC and MDIC are equal to zero, where the identified equation is exactly the same as the correct equation. For the data corrupted by 5% noise, DIC and MDIC are respectively equal to 0.043% and 0.1%.

The KdV equation   The Korteweg-de Vries (KdV) equation is a non-linear model for waves on the water surfaces. This equation was first introduced by Boussinesq [237]. The KdV had subsequently been applied in a wide range of physical systems, for example, shock waves, waves in a density-stratified ocean, and acoustic waves on a crystal lattice. The KdV equation is given by

$$u_t + 6uu_x + u_{xxx} = 0.$$

The solution of this non-linear PDE also shows a travelling wave behaviour, illustrated in Fig. (5.3) (a). The initial condition of this solution is two solitons in the space domain $-30 \leq$
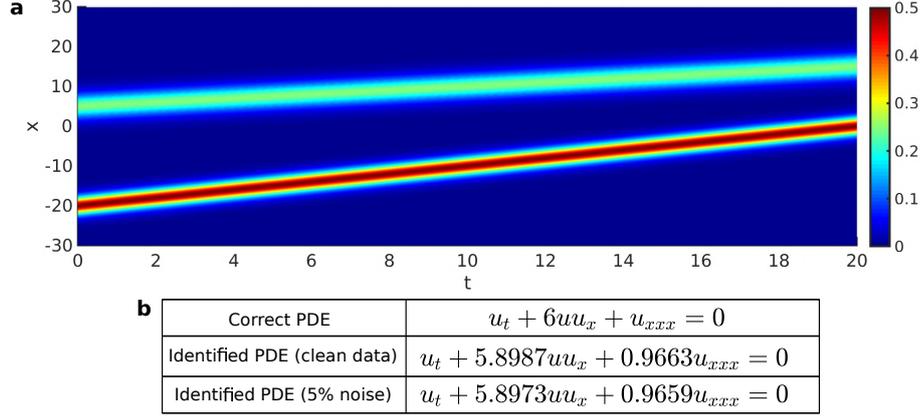
Figure 5.3: Testing the identification of partial differential equations with the KdV equation. (a) A solution of the KdV equation $u(x, t)$ is plotted on a one-dimensional space x from -30 to 30 and a time t from 0 to 20. The solution has 102912 total points with 201 time steps and 512 spatial grids. The initial condition consists of two solitons and the wave propagates with increasing time. (b) The table shows the correct KdV equation and two identified equations from data without noise and with 5% type I Gaussian noise.

$x \leq 30$. The total data covers 512 spatial grid points and 201 time steps with a time step of $\Delta t = 0.1$.

Using the data, we first to calculate $\mathbf{u}_t$, $\mathbf{u}_x$, $\mathbf{u}_{xx}$, $\mathbf{u}_{xxx}$ and $\mathbf{u}_{xxxx}$. Then, we construct the library $\boldsymbol{\Theta}$ which includes 12 terms and write a linear equation as $\mathbf{u}_t = \boldsymbol{\Theta}\mathbf{w} + \mathbf{s}$, where $\mathbf{s}$ is a type I Gaussian noise. Using ATSBL, we identify the KdV equation from the original data set and for the addition of 5% type I Gaussian noise, illustrated in Fig. (5.3) (b). We find that the identified equations contain the correct terms compared to real equation. For the original, noise free data, DIC and MDIC are equal to 1.686% and 3.4%, respectively. For 5% type I Gaussian noise, DIC and MDIC are respectively equal to 1.707% and 4.0%.

5.2.1.3   Identification of two-dimensional partial differential equations

Navier-Stokes equations   Here, the Navier-Stokes equation is employed to describe a two-dimensional fluid that passes a cylinder [40]. In this system, the vorticity equation deduced from the Navier-Stokes equations is

$$\omega_t + (\mathbf{v} \cdot \nabla)\omega = \frac{1}{\text{Re}}\nabla^2\omega,$$

where ω is the vorticity and $\mathbf{v} = (u, v)$ is the flow velocity. Re is the Reynolds number, which is equal to 100. Thus, the vorticity equation in a two-dimensional space can be rewritten as

$$\omega_t = 0.01\omega_{xx} + 0.01\omega_{yy} - u\omega_x - v\omega_y.$$

The system of the fluid past a cylinder can be numerically solved by using an Immersed Boundary Projection Method (IBPM) [239, 240]. Here, we employ previously published datasets containing numerical results for the fluid flow. Exemplary results including the vorticity ω, velocity u and velocity v are respectively shown in Fig. (5.4) (a-i to a-iii) [238]. These data is given on a grid of size 199×449 with 501 time steps.
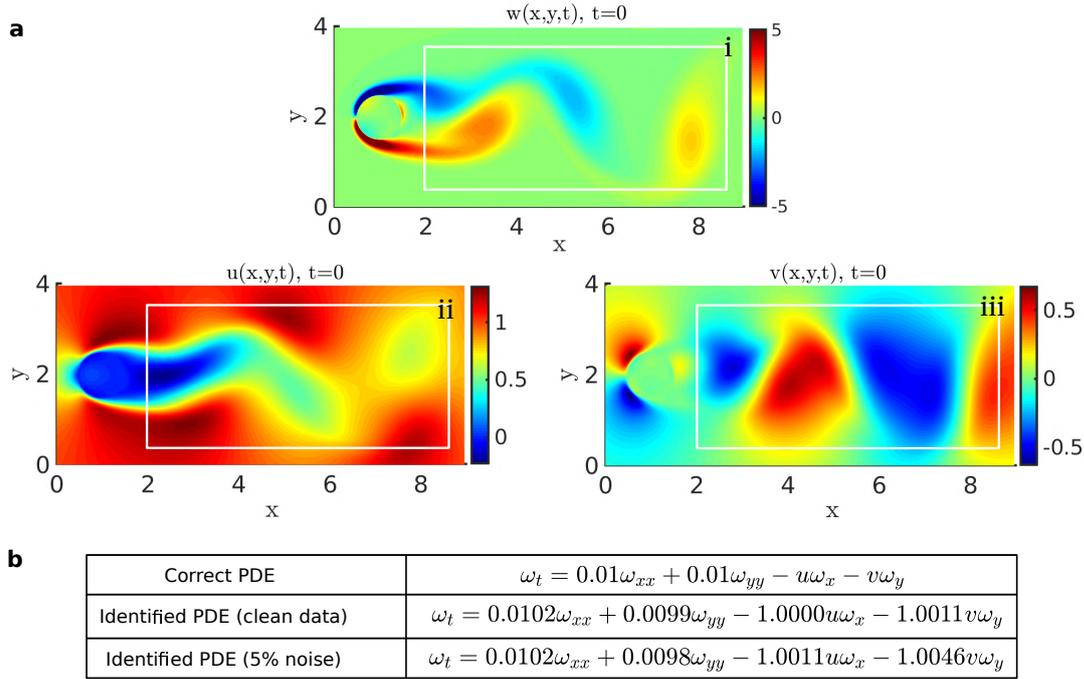
Figure 5.4: The identification of partial differential equations for Navier-Stokes equations. (a-i to a-iii) The snapshot of vorticity ω, velocity u and v at $t = 0$ for numerically calculated test data [238]. Only data inside the white box is used for identification in order to avoid boundary effects. (b) The table shows the correct vorticity equation and the identified vorticity equation from the data without noise and with 5% type I Gaussian noise.

Since the original data $199 \times 449 \times 501$ is too large, we only employ the data in the white box shown in Fig. (5.4) (a), where x is from $(2 - 8.5)$ and y is from $(0.25 - 3.75)$. We also increase the spacial grid size by a factor of 5. Thus, the size of the data for ω, u and v becomes $35 \times 66 \times 501$. Using the data-driven model for discovery of PDEs, we first construct the vector ω_t and the library matrix Θ, which has 15 terms. Then, we obtain a linear equation $\omega_t = \Theta \mathbf{w} + \mathbf{s}$, where $\mathbf{s}$ is type I Gaussian noise. The results from ATSBL for the vorticity equation are shown in Fig. (5.4) (b). We find that these identified equations have the correct terms compared to the real equation. For the noise-free data, DIC and MDIC are equal to 0.624% and 2%, respectively. For the 5% type I Gaussian noise data, DIC and MDIC are respectively equal to 0.914% and 2%.

Reaction diffusion equation   Reaction-diffusion equations have attracted interest as a prototype model for pattern formation in biochemical systems, which involve constituents locally transformed into each other by chemical reactions and transported in space by diffusion. The reaction-diffusion equation can describe a wide range of phenomena, for example, the formation of traveling waves, wave-like phenomena, and self-organized patterns. One of the common reaction diffusion systems is the Turing reaction-diffusion model which is usually used to describe biological pattern formation [241–243]. Here, we consider a two-variable Turing model in a two-dimensional, planar, periodic region with the following

a

u(x,y,t)

v(x,y,t)

Correct PDE	$u_t = -u_t v^2 + 0.1u_{xx} + 0.1u_{xx}v^2 + 0.1u_{yy} + 0.1u_{yy}v^2 + u^3 v$ $v_t = 0.1v_{xx} + 0.1v_{yy} + u^2 v + uv - 0.5v^3$
Identified PDE (noise-free data)	$u_t = -0.9969u_t v^2 + 0.1002u_{xx} + 0.1004u_{xx}v^2 + 0.1002u_{yy} + 0.1003u_{yy}v^2 + 1.0020u^3 v$ $v_t = 0.1003v_{xx} + 0.1003v_{yy} + 1.0033u^2 v + 1.0041uv - 0.5023v^3$
Identified PDE (5% noise)	$u_t = -1.0304u_t v^2 + 0.1025u_{xx} + 0.0982u_{xx}v^2 + 0.1025u_{yy} + 0.09846u_{yy}v^2 + 1.0071u^3 v$ $v_t = 0.1022v_{xx} + 0.1020v_{yy} + 1.0382u^2 v + 1.0063uv - 0.5106v^3$

Figure 5.5: The identification of partial differential equations for reaction-diffusion equations. (a) Snapshots of the initial conditions for u and v and the numerically calculated patterns after a time t=0.4. The results are obtained by using a time step $\Delta t = 0.02$ and the mesh grid is 256×256 with periodic boundary conditions. (b) The table shows the correct reaction diffusion equations, the identified equations from noise-free data and from data with 5% type I Gaussian noise.

equations

$$u_t = D_u \nabla^2 u + f_1(u, v), \tag{5.9a}$$

$$v_t = D_v \nabla^2 v + f_2(u, v), \tag{5.9b}$$

$$f_1 = \frac{u^3 v}{1 + v^2}, \quad f_2 = u^2 v + uv - 0.5v^3, \tag{5.9c}$$

Where two states u and v can the thought of as morphogen concentrations. The diffusion coefficients D_u and D_v are equal to 0.1. The functions f_1 and f_2 model chemical transformations. The Eq. (5.9) can be rewritten as

$$u_t = -u_t v^2 + 0.1u_{xx} + 0.1u_{xx}v^2 + 0.1u_{yy} + 0.1u_{yy}v^2 + u^3 v,$$

$$v_t = 0.1v_{xx} + 0.1v_{yy} + u^2 v + uv - 0.5v^3.$$

We solve the equations (5.9) numerically by using a spectral transform method. Equations (5.9) can be written after a Fourier transformation of the spatial coordinates as

$$\tilde{u}_t = 0.1(k_x^2 + k_y^2)\tilde{u} + \widetilde{f_1(u, v)},$$

$$\tilde{v}_t = 0.1(k_x^2 + k_y^2)\tilde{v} + \widetilde{f_2(u, v)},$$

where the spatial wave vector is $\mathbf{k} = (k_x, k_y)$. These equations can be solved numerically by using a Runge-Kutta ODE solver. Figure (5.5) (a) shows exemplary results. The two solutions u and v are calculated on grids of size 256×256 in a domain 14×14 with periodic

boundary conditions. The total simulation time is $t = 2$ with a time step $\Delta t = 0.02$.

As for the previous data-driven model inference procedures, we first construct $\dot{\mathbf{u}}$, $\dot{\mathbf{v}}$, $\mathbf{u}_x$, $\mathbf{u}_{xx}$, $\mathbf{u}_{xxx}$, $\mathbf{v}_x$, $\mathbf{v}_{xx}$, and $\mathbf{v}_{xxx}$ from the numerical data on u and v. Then, we build the library matrix $\boldsymbol{\Theta}$, which includes 35 terms and write two linear equations, $\mathbf{u}_t = \boldsymbol{\Theta}\mathbf{w} + \mathbf{s}$ and $\mathbf{v}_t = \boldsymbol{\Theta}\mathbf{w} + \mathbf{s}$, where $\mathbf{s}$ is the type I Gaussian noise. Finally, using ATSBL, we identify the reaction-diffusion equations from the noise-free data and 5% type I Gaussian noise data, as illustrated in Fig. (5.5) (b). We find the correct terms in both cases. For the noise-free data, DIC and MDIC are equal to 0.31% and 0.46%, respectively. For the 5% type I Gaussian noise data, DIC and MDIC are equal to 2.08% and 3.40%, respectively.

We have so far demonstrated a method for identification of governing ODEs and PDEs from data in the presence of type I Gaussian noise where the problem can be written in a form $\mathbf{g} = \boldsymbol{\Phi}\mathbf{w} + \mathbf{s}$. Next, we will employ a neural network deep learning approach to reduce type II Gaussian noise which usually present in measurement data.

5.2.1.4 Neural network deep learning improves the identification of PDEs from type II Gaussian noise data

One issue related to library-based data-driven inference of governing equations is that numerical derivatives are required. However, derivatives amplify noise in the data. Neural networks are widely employed in many fields, for example, image analysis [244–246], autonomous cars [247–249], and diagnosis [250–252]. Deep neural networks can be used for so-called automatic differentiation where the derivatives are estimated together with a cost function to directly suppress the measurement noise. In Ref. [82], the authors describe a neural network deep learning approach to reduce noise in the data for the identification of governing PDEs. Essentially, the proposed method is a neural-network-based filter. In this method, an L1 penalty is employed for the cost function in deep learning. We modify the cost function by using the Elastic Net penalty because the Elastic Net regularization combines both L2- and L1 regularization and is expected to outperform both of the two regulation methods invidually, as demonstrated in Chapter 3. The cost function is written as

$$\mathcal{L} = \mathcal{L}_{MSE} + \mathcal{L}_{Reg} + \mathcal{L}_{EN},$$

where $\mathcal{L}_{EN}$ is the Elastic Net penalty for the vector w_i, $\mathcal{L}_{MSE}$ is the mean squared error between the output u_{recon} and the target data u_{noise} and $\mathcal{L}_{Reg}$ is the regression based cost function. These functions are respectively written as

$$\mathcal{L}_{EN} = \lambda_1 \|w_i\|_1 + \lambda_2 \|w_i\|_i^2,$$

$$\mathcal{L}_{MSE} = \frac{1}{N} \sum_{i=1}^{N} |u_{\text{noise}} - u_{\text{recon}}|^2,$$

$$\mathcal{L}_{Reg} = \frac{1}{N} \sum_{i=1}^{N} |\Phi_{ij} w_j - \dot{u}_i^{\text{recon}}|^2.$$

Using this approach, we use the time t and space x as an input data and the noisy data u_{noise} as target data. The reconstruction u_{recon} is an output data. The output data is produced by the neural network deep learning framework TensorFlow by minimizing the given cost
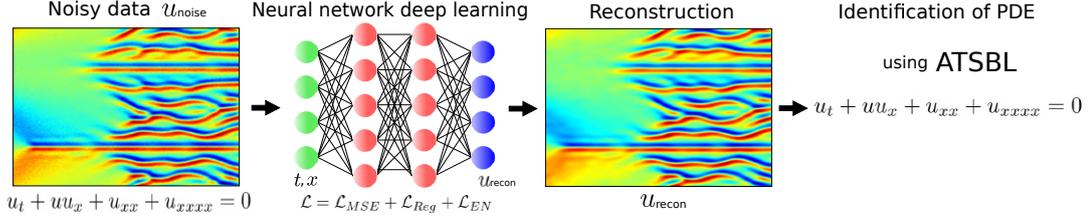
Figure 5.6: The neural network deep learning reduces the type II Gaussian noise data and the governing equations is identified by using ATSBL. The solution of the KS equation is corrupted with 20% type II Gaussian noise and denoted by u_{noise}. Time t and space x dimensions of the data. The reconstructed data u_{recon} is predicted by using the neural network where the cost function includes the MSE-loss, a regression loss and an Elastic Net penalty. The identified KS equation is obtained by using ATSBL.

function. The noise in the reconstructed data u_{recon} is significantly reduced. Then, we employ ATSBL to identify the governing equation from the reconstructed data u_{recon}.

Correct PDE	$u_t + uu_x + u_{xx} + u_{xxxx} = 0$
Identified PDE (clean data)	$u_t + 0.9998uu_x + 0.9998u_{xx} + 0.9998u_{xxxx} = 0$
Identified PDE (10% noise)	$u_t + 0.712uu_x + 0.663u_{xx} + 0.667u_{xxxx} = 0$
Identified PDE (20% noise)	$u_t + 0.583uu_x + 0.496u_{xx} + 0.506u_{xxxx} = 0$
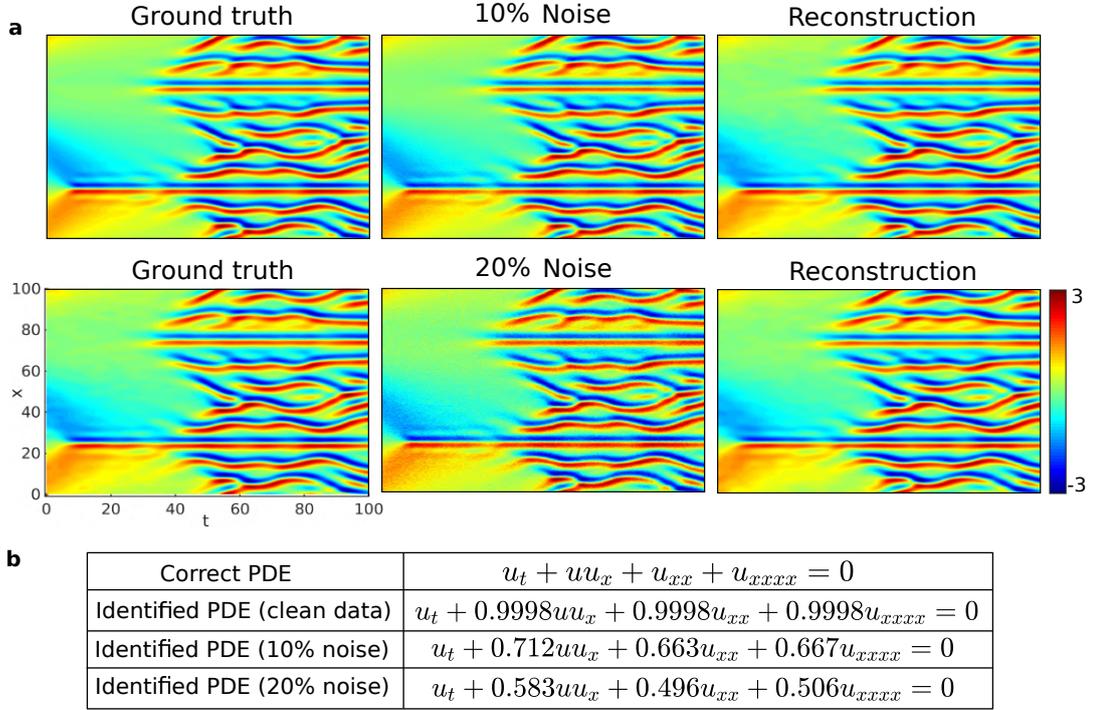
Figure 5.7: The neural network deep learning method improves the identification of the Kuramoto-Sivashinsky (KS) equation. (a-Left) The snapshots of the KS equation with 1024 spatial points and 251 time-steps. The time step is $\Delta t = 0.4$. (a-Middle) 10% and 20% type II Gaussian noise is added to the data. (a-Right) The reconstructed data obtained by using the neural network. (b) The table shows the correct KS equation and identified equations.

Next, we employ the Kuramoto-Sivashinsky (KS) equation as an example to demonstrate the power of the approach. The KS equation is a one-dimensional fourth-order PDE, which has been independently derived for several physical systems, for example, for the dissipative trapped ion model in plasmas, instabilities in laminar fronts, fluctuations in fluid and phase dynamics in reaction-diffusion systems. The KS equation is written as

$$u_t + uu_x + u_{xx} + u_{xxxx} = 0.$$

This equation is solved by using the spectral method [40]. The size of solution is 1024×251 data points in the format space$\times$time, illustrated in Fig. (5.7) (a-Left). The snapshots of $u(x, t)$ are calculated by using the dimensionless time step $\Delta t = 0.4$ and time ranges from 0 to 100. The one dimensional solution along the non-dimensional space coordinate x ranges from 0 to 100. We first add 10% and 20% type II Gaussian noise into the solution of snapshots $u(x, t)$, illustrated in Fig. (5.7) (a-Middle). Using the neural network, we apply a filter to obtain reconstructed data, illustrated in Fig. (5.7) (a-Right). Next, using the data-driven model for the identification of PDEs, we construct the vector $\dot{\mathbf{u}}$ and library matrix $\boldsymbol{\Theta}$ from the reconstructed data and build a linear equation $\mathbf{u}_t = \boldsymbol{\Theta}\mathbf{w} + \mathbf{s}$. Finally, we identify the KS equations by using ATSBL, as illustrated in Fig. (5.7) (b). We find that these identified equations have the correct terms. For 10% type II Gaussian noise, DIC and MDIC are equal to 23.95% and 33.7%, respectively. For 20% type II Gaussian noise, DIC and MDIC are respectively equal to 35.38% and 50.4%. Note that in the article that originaly proposed neural networks for identification of PDEs, Ref. [40], the identified KS equation is $u_t + 0.46uu_x + 0.48u_{xx} + 0.49u_{xxxx} = 0$ when only 1% type II Gaussian noise is added and its DIC and MDIC are equal 39.25% and 54.0%, respectively. Thus, we demonstrate that our modifications of the deep learning approach along with ATSBL significantly improve the inference of PDE's.

5.2.2 Identification of stochastic differential equations

According the data-driven model approach for inference of SDEs with Eq. (5.6), the drift- and diffusion coefficient can be identified from trajectories of stochastic processes that can be modeled by a Fokker-Planck equation. In this section, we will explain details of the inference procedure and provide examples.

5.2.2.1 Identification of stochastic differential equations from double-well potential systems

To generate exemplary stochastic trajectory data, we consider a one-dimensional double-well potential $U(x)$. The potential and the drift force $g(x)$ are written as

$$U(x) = \frac{1}{2}x^4 - 4x^3 + 9x^2 - 3x, \;\; g(x) = D^{(1)}(x) = -\frac{dU}{dx} = -2x^3 + 12x^2 - 18x + 3. \quad (5.10)$$

An inhomogeneous diffusion coefficient is written as $D^{(2)}(x) = x^2 + 1$. We employ the Euler–Maruyama method to numerically solve the one-dimensional Langevin equation

$$\frac{dX(t)}{dt} = g(X(t), t) + h(X(t), t)dW(t). \quad (5.11)$$

The initial state of the trajectory is located at $X = 0$ and the time step Δt is equal to $5e - 3$. We numerically obtain a trajectory $\mathbf{X}$ with $2e7$ states, with an example shown in Fig. (5.8-a). We find that most states are located in one of the two potential wells (yellow region).

According to the data-driven model for discovery of SDEs, we first construct datasets containing the discretized derivatives $\mathbf{D}$ and $\mathbf{F}$ from the trajectory $\mathbf{X}$. Then, using data binning, we construct three datasets $\{x_i\}$, $\mathbb{D}$ and $\mathbb{F}$ from these trajectories. Next, we utilize the binned data $\{x_i\}$ to construct a 11-term library $\boldsymbol{\Theta}_1 = [I, \; x, \; x^2, \; x^3, \; x^4, \; x^5, \ldots, \; x^{10}]$ and a 6-term library $\boldsymbol{\Theta}_2 = [I, \; x, \; x^2, \; x^3, \; x^4, \; x^5]$ respectively for the drift and diffusion coefficients. Finally,
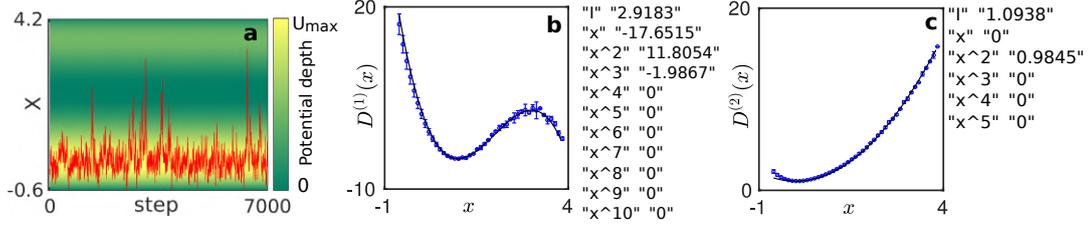
Figure 5.8: The identification of stochastic differential equations for a double-well potential with an inhomogeneous diffusion. (a) A part of a stochastic trajectory on a double well potential depth map. The trajectory is calculated by using a time step $\Delta t = 5e - 3$ and the initial state is $X = 0$. (b, c) The identification of the drift and diffusion coefficients from the trajectory data. The blue points are binned data and the black line is the true solution. The identified terms of the equation are shown in the lists of library terms for the drift and diffusion coefficients.

we build two linear equations for the drift- and diffusion coefficients as

$$\mathbb{D} = \mathbf{\Theta}_1 \mathbf{W}^1,$$
$$\mathbb{F} = \mathbf{\Theta}_2 \mathbf{W}^2.$$

Using ATSBL, we obtain the sparse solution vectors $\mathbf{W}^1$ and $\mathbf{W}^2$, which are given in Fig. (5.8-b and -c). With the help of these vectors, we can write the drift and diffusion coefficients as

$$\breve{D}^{(1)}(x) = -1.9867x^3 + 11.8054x^2 - 17.6515x + 2.9183,$$
$$\breve{D}^{(2)}(x) = 0.9845x^2 + 1.0938.$$

We find that the identified drift- and diffusion coefficients have the correct terms that also appear in the original equations in Eq. (5.10) and Eq. (5.11). DIC and MDIC for the drift coefficient $\breve{D}^{(1)}(x)$ are equal to 1.737% and 2.723%, respectively. DIC and MDIC for the diffusion coefficient $\breve{D}^{(2)}(x)$ are respectively equal to 5.465% and 9.380%.

5.2.2.2 A novel probability-threshold procedure improves the identification of stochastic differential equations

In our approach for identification of SDEs, the trajectories are dealt with by using data binning. Thus, we know the probability $p(x_i)$ of each binned state x_i. In information theory, *self-information* is defined as $I(X_i) := -\log_2[p(x_i)]$. One state has a large self-information if its occurrence is highly improbable, for example, consider two states A and B, with $p(A) = 0.9$, $I(A) = \log_2(0.9) = 0.152$bit and $p(B) = 0.1$, $I(B) = \log_2(0.1) = 3.322$bit. The *Shannon entropy* or *total uncertainty* is given by [253, 254]

$$H = -\sum_{i=1}^{\hat{Q}} p(x_i) \log_2 p(x_i), \tag{5.12}$$

where we only consider states x_i with a non-zero probability and $\hat{Q}$ is the number of states bins with non-vanishing probability and Q is the overall number of bins, thus, $\hat{Q} \leq Q$. To reduce the total uncertainty for the binned data, we introduce a *probability threshold T* to remove states with small probabilities and high uncertainty. To find a optimal probability

threshold, we plot the log-log curve of the total uncertainty vs different thresholds and we select the optimal probability threshold at the corner of the curve. While we were not able to justify this heuristic procedure theoretically, we found that it produces excellent results. As an example, we employ a one-dimensional double-well potential given in Eq. (5.10) with a homogeneous diffusion coefficient as

$$D^{(1)}(x) = -2x^3 + 12x^2 - 18x + 3 \tag{5.13}$$

$$D^{(2)}(x) = 1. \tag{5.14}$$

Using a Langevin equation containing this drift- and diffusion terms, we simulate a trajectory of $1e6$ steps with a time step of $\Delta t = 5e - 3$.
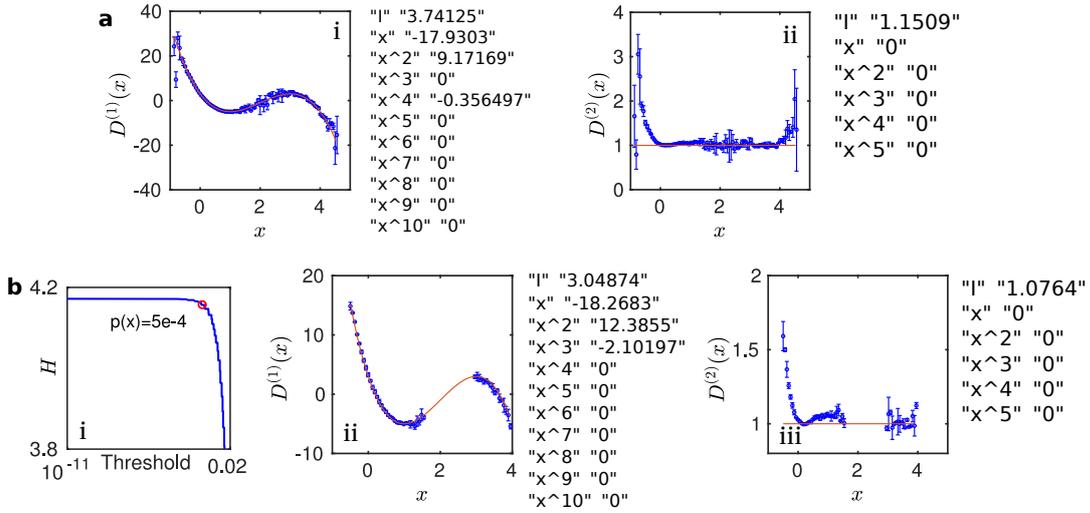
Figure 5.9: An improved identification of stochastic differential equations through using a probability threshold. (a) The binned data and identified drift- and diffusion coefficients without using the probability threshold. (b-i) A log-log curve of the total uncertainty vs different probability thresholds. The optimal probability threshold is selected at the corner of the curve equal to $5e - 4$. (b-ii and b-iii) The binned data and the identified terms in the drift and diffusion functions when using the probability threshold. Here, blue points are binned data and red curves are exact solutions.

Then, we produce the binned data $\{x_i\}$, $\mathbb{D}$ and $\mathbb{F}$ from the trajectories. In Fig. (5.9-a) we show results from the reconstruction without using a threshold on the probability. We find that the binned data exhibits large errors as quantified by the blue error bars on the distribution. Using ATSBL, we identify the drift- and diffusion coefficients as

$$\breve{D}^{(1)}(x) = -0.3565x^4 + 9.1717x^2 - 17.9303x + 3.7413,$$

$$\breve{D}^{(2)}(x) = 1.1509.$$

We find that $\breve{D}^{(1)}(x)$ contains a term x^4 which is not present in the original equation. The DIC of the diffusion coefficient is equal to 15.09%. Next, we employ the probability threshold. The log-log curve of the total uncertainty vs different threshold values is plotted in Fig. (5.9-b-i). We select the optimal probability threshold at the corner of the curve. Using this threshold, we plot the binned data for the $D^{(1)}(x)$ and $D^{(2)}(x)$, and results are shown in Fig. (5.9-b-ii and iii), respectively. As illustrated in the plots, we choose a threshold at a value where the entropy just starts to change significantly. Thereby, we efficiently remove the bins

with high uncertainty. Using ATSBL, the drift- and diffusion coefficients are identified from the thresholded data as

$$\breve{D}^{(1)}(x) = -2.102x^3 + 12.386x^2 - 18.268x + 3.049,$$
$$\breve{D}^{(2)}(x) = 1.076.$$

Here, $\breve{D}^{(1)}(x)$ and $\breve{D}^{(2)}(x)$ contain the correct terms. For the drift coefficient, $\breve{D}^{(1)}(x)$, DIC and MDIC are equal to 2.857% and 5.099%, respectively. For the diffusion coefficient, $\breve{D}^{(2)}(x)$, DIC is equal to 7.60%.

5.2.2.3 An automatic iterative sampling optimization improves the identification of stochastic differential equations

From above two tests for the identification of SDEs, we find that the density of measured trajectories is highest around the minima of the potential, see Fig. (5.8-a). At the minimum of the potential, the probability of the binned data is the highest and its uncertainty is the smallest. For example, we employ a double-well potential as given in Eq. (5.10) with a homogeneous diffusion $D^{(1)}(x) = 1$ to numerically simulate a trajectory. The two minima of the potential are respectively located at A and C, illustrated in Fig. (5.10-a-i). A saddle point B is located between A and C. Because most measurement points lie close to the two minima, A and C, the binned probability exhibits maxima there, as shown in Fig. (5.10-a-ii). The saddle point B is located between these two maximum probabilities.
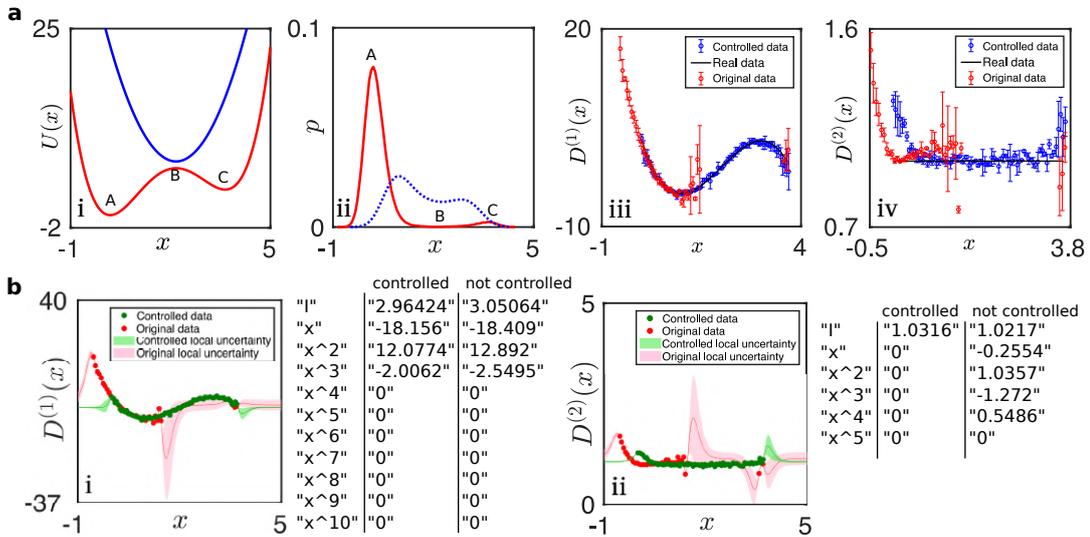
Figure 5.10: The identification of stochastic differential equations is improved by using sampling strategy. (a-i) The double-well potential (red line) is controlled by one-well potential (blue line) at the saddle point B. (a-ii) The probability for original and controlled binned data. (a-iii) The binned data for the drift coefficients are plotted, where controlled and original data are respectively shown as blue- and red points. (a-iv) The controlled and original binned data for the diffusion coefficients are plotted. (b-i and iii) The local uncertainty of drift- and diffusion coefficients are distributed at the range of binning data. The green plot is for the controlled data and the red plot is for the original data.

In oder to decrease uncertainty of the binned data, we employ an Umbrella Sampling technique where we add a control potential to reduce the saddle point and to increase original minimum potential [255, 256], for example, a one-well control potential $U_c(x) = 2(x - 2.1683)^2$, where 2.1683 is the position of the saddle point. The sampling potential is plotted as a blue curve, illustrated in Fig. (5.10-a-i). Using this sampling strategy, we obtain a modified trajectory and its binned data, see Fig. (5.10-a-iii and -iv). Note that the binned data with the "control potential" has been corrected for the known "control force" $4(x - 2.1683)$. We find that the probability in the data with a control potential is approximately uniformly distributed from A to C, as illustrated in Fig. (5.10-a-ii). In Fig. (5.10-a-iii and iv), the binned original data (red points) does not contain many measurements in the range from $1 = 1.5$ to $x = 3.5$ while the data with the control potential (blue points) is well-spread in the range $x = 0$ to $x = 3.7$. For these binned data, we can calculate the local uncertainty by using *universal prediction distribution for surrogate models* [257]. The local uncertainty is high where the density of binned data is small. Thus, the original data missed in $1.5 < x < 3.5$ leads to a high local uncertainty in this region and the local uncertainty for the controlled data is small from -0.4 to 3.7, illustrated in Fig. (5.10-b-i and b-ii).

Next, we identify $D^{(1)}(x)$ and $D^{(2)}(x)$ with real drift- and diffusion terms given by Eqs. (5.13) and (5.14) for both controlled- and non-controlled cases by using ATSBL. The identified equations in the case of an added control potential are given by

$$\breve{D}^{(1)}(x) = -2.0062x^3 + 12.077x^2 - 18.156x + 2.964,$$
$$\breve{D}^{(2)}(x) = 1.0316.$$

The identified equations for the orginal case without control potential are

$$\breve{D}^{(1)}(x) = -2.550x^3 + 12.892x^2 - 18.409x + 3.051,$$
$$\breve{D}^{(2)}(x) = 0.549x^4 - 1.272x^3 + 1.036x^2 - 0.255x + 1.022.$$

For the identified $\breve{D}^{(1)}(x)$ from the controlled data, DIC and MDIC are equal to 0.755% and 1.2%, respectively. For the identified $\breve{D}^{(2)}(x)$ from the controlled data, DIC is equal to 3.16%. For the identified $\breve{D}^{(1)}(x)$ from the original data, DIC and MDIC are equal to 9.726% and 27.50%, respectively. The identified $\breve{D}^{(2)}(x)$ here contains wrong terms. Thus, we find that the sampling strategy improves the identification of SDEs.

For use of the sampling strategy suggested above, we need to know the saddle points on the potential. These saddle points are of course not known initially. However, the positions of saddle points can be well approximated by using the probability distribution of the binned original data, and assuming that saddle points are located between the known position of maximum probability, see Fig. (5.10-a-ii). For this purpose, we develop an *automatic iterative sampling optimization* to improve the identification of SDEs. The method is illustrated in Algorithm 2. In this algorithm, we fist find the n maximum probability points $\{p_1, \ldots, p_n\}$ and the saddle points $\{pc_1, \ldots, pc_{n-1}\}$ are approximated as the center between two maximum probability points. Secondly, the control potential can be written as $U_c(x) = \text{coeff}(x - pc_1)^2 \ldots (x - pc_{n-1})^2$, where coeff is a coefficient. Then, we employ this control potential to generate data and the contribution of the control potential is removed after binning. Finally, we build libraries and identify the drift- and diffusion coefficients by using ATSBL. In the next subsections, we demonstrate the use of this algorithm to identify

Algorithm: Automatic iterative sampling optimization

%% Initial setting

coeff = 0.1; j = 0; control = 0; x = 0; dt = 0.005;

while 1 **do**

    j=j+1;

    %% Form of control

    if *j>=2* **then**

        if *n==2* **then**

            drift_c = coeff$*$2$*$(x(i-1)-pc_1);   % in case of estimated double-well, add quadratic potential

        end

        if *n==3* **then**

            %% in case of estimated tripe-well, add double-well potential

            drift_c = coeff$*$(2$*$(x(i-1)-pc_2)$*$(x(i-1)-pc_2)^2+2$*$(x(i-1)-pc_1)^2$*$(x(i-1)-pc_2);

        end

        . . .

        $[pc_1^o, \ldots, pc_{n-1}^o] = [pc_1, \ldots, pc_{n-1}]$;   % save data

    end

    %% Generate trajectory

    for ($i = 1 : step + 1$ **)** {

        drift= real_drift+drift_c; % real_drift is a given drift coefficient

        x(i)=x(i-1)- drift$*$dt+sqrt(2$*$diff$*$dt)$*$randn(1,1); % diff is a diffusion coefficient

        y=(x(i)-x(i-1))/dt;

        D=(x(i)-x(i-1))^2/(2$*$dt);

    }

    %% Binned data

    $[p, xc, bins]$ = histcount(x, number_bin);   % p is probability, xc is center of each bin

    drift_bin=mean(y(bins));

    diffu_bin=mean(D(bins));

    $pp = [p_1, \ldots, p_n]$;   %% Find all local maxima of the probability

    %% Find center of maximum

    $[pc_1, \ldots, pc_{n-1}] = pp(1 : end - 1) + 0.5 * (pp(2 : end) - pp(1 : end - 1))$;

    %% Convergence condition

    pmax=max(pp);

    b = $pp(pp \sim$=pmax);

    pmax2=max(b);

    if *pmax<pmax2* **then**

        break;

    else

        coeff = coeff+0.05;

    end

    %% Remove the control

    if *j>=2* **then**

        if *n==2* **then**

            mc = coeff$*$2$*$(xc-pc_1^o);

        end

        if *n==3* **then**

            mc = coeff$*$(2$*$(xc-pc_1^o)$*$(xc-pc_2^o)^2+2$*$(xc-pc_1^o)^2$*$(xc-pc_2^o);

        end

        . . .

    end

    drift=drift_bin+mc;

    %% Build libraries and ddentify the drift and diffusion coefficients by using ATSBL

    w = ATSBL(**Θ**, **g**, d_{tol}, n_{iters})

end

        Algorithm 2: Pseudocode for automatic iterative sampling optimization

SDEs from three- and four-well potential systems.
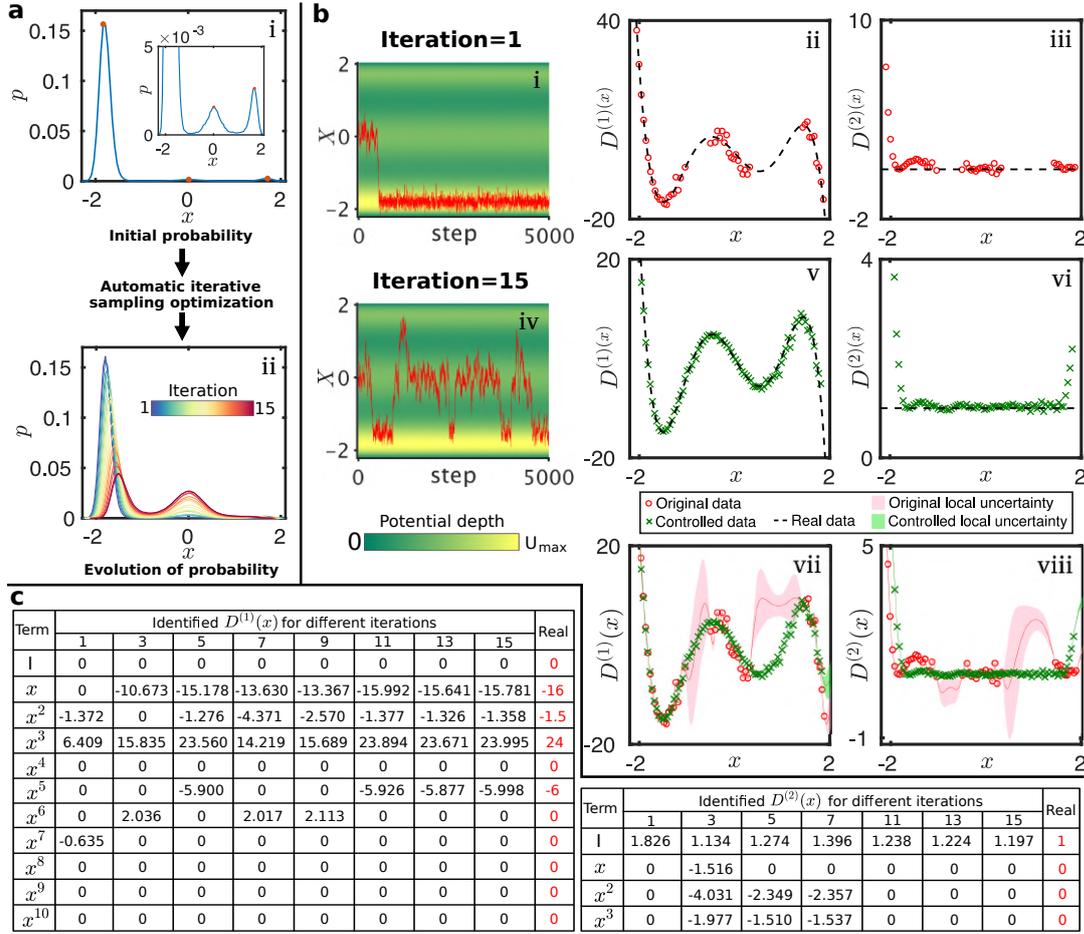
Figure 5.11: The identified SDEs from three-well potential by using an automatic iterative sampling optimization. (a) Evolution of probability distribution in automatically iterative control process. Three maximum probability points are clearly found in (a-i). (b-i) The original trajectory is plotted on the three potential depth map. (b-ii and iii) The plot of binned data for drift- and diffusion coefficients from the original trajectory. The binned data is lost at $1 = -0.8$ and $x = 1$. (b-iv) The trajectory is numerically obtained by using the control strategy on the three potential depth map. (b-v and vi) The binned data for drift- and diffusion coefficients from controlled trajectory are uniformly distributed from $x = -2$ to $x = 2$. (b-vii and viii) The local uncertainty of controlled (light green) and original data (light red) for the drift- and diffusion coefficients. (c) The identified the drift- and diffusion coefficients for different iterations, which are improved with increasing iterations.

Three-well potential   We consider a three-well potential with a homogeneous diffusion coefficient $D^{(2)}(x) = 1$ to generate a stochastic trajectory data. The potential function and drift force are

$$U(x) = x^6 - 6x^4 + 0.5x^3 + 8x^2, \quad g(x) = D^{(1)}(x) = -6x^5 + 24x^3 - 1.5x^2 - 16x. \tag{5.15}$$

In the fist iteration, we obtain a trajectory with $6e5$ states that are simulated by using the time step $\Delta t = 5e-3$ and the initial state $X = 0$. A trajectory on the potential map is illustrated in Fig. (5.11-b-i). This first iteration produces a trajectory corresponding to the

Panel (c) tables:

Term	Identified $D^{(1)}(x)$ for different iterations								Real
	1	3	5	7	9	11	13	15	
I	0	0	0	0	0	0	0	0	0
x	0	-10.673	-15.178	-13.630	-13.367	-15.992	-15.641	-15.781	-16
x^2	-1.372	0	-1.276	-4.371	-2.570	-1.377	-1.326	-1.358	-1.5
x^3	6.409	15.835	23.560	14.219	15.689	23.894	23.671	23.995	24
x^4	0	0	0	0	0	0	0	0	0
x^5	0	0	-5.900	0	0	-5.926	-5.877	-5.998	-6
x^6	0	2.036	0	2.017	2.113	0	0	0	0
x^7	-0.635	0	0	0	0	0	0	0	0
x^8	0	0	0	0	0	0	0	0	0
x^9	0	0	0	0	0	0	0	0	0
x^{10}	0	0	0	0	0	0	0	0	0

Term	Identified $D^{(2)}(x)$ for different iterations							Real
	1	3	5	7	11	13	15	
I	1.826	1.134	1.274	1.396	1.238	1.224	1.197	1
x	0	-1.516	0	0	0	0	0	0
x^2	0	-4.031	-2.349	-2.357	0	0	0	0
x^3	0	-1.977	-1.510	-1.537	0	0	0	0

original potential and we clearly find three probability maxima, illustrated in Fig. (5.11-a-i). Thus, we can approximate the saddle points and write the control potential for the next iteration. The original binned data for the drift- and diffusion coefficients are illustrated in Fig. (5.11-b-ii and iii), which are filtered by using a probability threshold equal to $4e-5$. We find that the data is lost at $x=-0.8$ and $x=1$. Therefore, the local uncertainty is high in these two regions for both $\breve{D}^{(1)}(x)$ and $\breve{D}^{(2)}(x)$, as illustrated in Fig. (5.11-b-vii and viii). Using ATSBL, we identify $D^{(1)}(x)$ and $D^{(2)}(x)$ from original data as

$$\breve{D}^{(1)}(x) = -0.635x^7 + 6.409x^3 - 1.372x^2,$$
$$\breve{D}^{(2)}(x) = 1.826.$$

We find that $\breve{D}^{(1)}(x)$ contains wrong terms compared to the real equation. As for the identified $\breve{D}^{(2)}(x)$, DIC is equal to 82.6%.

As iterations of the algorithm proceed, the coefficient of the control potential is increasing, the initial maximum probability is deceasing, and the probability of trajectories crossing the saddle points are increasing, as illustrated in Fig. (5.11-a-ii). Results for the intermediate iteration step 15 are shown together with the potential depth map in Fig. (5.11-b-iv). We find that the trajectory is concentrated around three potential minima. The binned controlled data for the drift- and diffusion coefficients are respectively shown in Fig. (5.11-b-v and vi). The local uncertainty of this controlled data is shown in Fig. (5.11-b-vii and viii). We find that the local uncertainty are is drastically reduced by the additional control potential in the region $-2 < x < 2$. Using ATSBL, we identify $D^{(1)}(x)$ and $D^{(2)}(x)$ from the binned controlled data at iteration$= 15$ as

$$\breve{D}^{(1)}(x) = -5.998x^5 + 23.995x^3 - 1.358x^2 - 15.781x,$$
$$\breve{D}^{(2)}(x) = 1.197.$$

These equations have the correct terms as compared to the real equations. For the identified $\breve{D}^{(1)}(x)$, DIC and MDIC are equal to 2.72% and 9.47%, respectively. For the identified $\breve{D}^{(2)}(x)$, DIC is equal to 19.7%. Further iterations of the algorithm lead to a continuous improvement of the estimates for the drift- and diffusion coefficients, see Fig. (5.11-c).

Four-well potential   We consider another one-dimensional example that is a four-well potential with a homogeneous diffusion. The four-well potential and its drift force are written as

$$U(x) = 0.055x^8 - 0.855x^6 + 4x^4 - 6x^2 + 1.5x,$$
$$g(x) = D^{(1)}(x) = -0.44x^7 + 5.13x^5 - 16x^3 + 12x - 1.5.$$

The homogeneous diffusion coefficient is given by $D^{(2)}(x) = 1.2$. Using automatic iterative sampling optimization, at iteration$= 1$, an original trajectory is obtained by using the time step $\Delta t = 5e-3$ and the initial state $X = 0$. States from an exemplary trajectory are shown in Fig. (5.12-b-i). Using the data-driven model for identification of SDEs, we obtain binned discretized datasets for $D^{(1)}(x)$ and $D^{(2)}(x)$, which are filtered by using a probability threshold equal to $5e-4$, illustrated in Fig. (5.12-b-ii and b-iii), respectively. We find that the binned data is lost in the regions $x \in [-0.3, 0.6]$ and $x \in [1.1, 2.3]$. Thus, the local uncertainty is very high in these regions, illustrated in Fig. (5.12-b-vii and b-viii). We employ ATSBL and

Table for panel c, left — Identified $D^{(1)}(x)$:

Term	Identified $D^{(1)}(x)$ for different iterations								Real
	1	2	4	6	8	10	13	15	
I	0.248	0	-5.006	-1.448	-2.075	-1.407	-1.409	-1.403	-1.5
x	-9.520	-5.204	-0.850	11.734	11.731	11.634	11.858	12.000	12
x^2	-10.559	0	16.988	0	0	0	0	0	0
x^3	0	5.629	11.514	-15.449	-18.862	-15.250	-15.695	-15.835	-16
x^4	0	0	0	0	0	0	0	0	0
x^5	0	-0.752	0	4.921	9.137	4.838	5.008	5.048	5.13
x^6	0.175	0	0.287	0	2.520	0	0	0	0
x^7	0	0	0	-0.419	0	-0.409	-0.426	-0.429	-0.44
x^8	0	0	0	0	0	0	0	0	0
x^9	0	0	0	0	0	0	0	0	0
x^{10}	0	0	0	0	0	0	0	0	0

Table for panel c, right — Identified $D^{(2)}(x)$:

Term	Identified $D^{(2)}(x)$ for different iterations							Real
	1	2	4	8	10	13	15	
I	2.248	2.095	1.963	1.820	1.574	1.467	1.520	1.2
x	-2.993	0	0	0	0	0	0	0
x^2	-6.949	-2.849	-2.464	-1.853	0	0	0	0
x^3	-2.477	-1.285	-1.137	-0.930	0	0	0	0

Figure 5.12: The identified SDEs from four-well potential by using an automatic iterative sampling optimization. (a) Probability distribution of binned data in the process of automatic iterative sampling optimization. The saddle points can be obtained in probability distribution. (b-i) The part of original trajectory data is from $t = 0$ to $t = 30000$ on the four-well potential depth map, which the most state are located at minimum potential. (b-ii and iii) The binned data for the drift- and diffusion coefficients from original trajectory. The data is lost at $x = 0.5$ and $x = 2$. (b-iv) The control trajectory on the potential map at iteration=15. (b-v and vi) The binned data for drift- and diffusion coefficients from controlled trajectory. (b-vii and viii) The local uncertainty for controlled- and original binned data from $x = -3$ to $x = 2.5$. (c) Identified drift- and diffusion coefficients in the iterations process. The identification of equations become better and better with increasing the iterations.

identify $D^{(1)}(x)$ and $D^{(2)}(x)$ at iteration 1 as

$$\check{D}^{(1)}(x) = 0.175x^6 - 10.559x^2 - 9.520x + 0.248,$$
$$\check{D}^{(2)}(x) = -2.477x^3 - 6.949x^2 - 2.993x + 2.248.$$

We find that the identified $\check{D}^{(1)}(x)$ and $\check{D}^{(2)}(x)$ from the original data have wrong terms. However, at iteration 1, we clearly find four maximum probability points, illustrated in Fig. (5.12-a-i) and we obtain three saddles points. The probability maxima are decreasing with increasing the iteration, see Fig. (5.12-a-ii). Results from an exemplary trajectory simulated in the presence of a control potential at iteration 15 are shown in Fig. (5.12-b-iv). At this iteration, the trajectories already cover the whole system quite evenly. The binned data for

$D^{(1)}(x)$ and $D^{(2)}(x)$ are illustrated in Fig. (5.12-b-v and vi). These binned data are uniformly distributed in $-2 < x < 2$. Thus, the local uncertainty of the controlled binned data for the drift- and diffusion coefficients is smaller than the local uncertainty in the original data in the region $-2 < x < 2$, see Fig. (5.12-b-vii and viii). Using ATSBL, we identify the $D^{(1)}(x)$ and $D^{(2)}(x)$ equations from the controlled binned data at iteration 15 as

$$\check{D}^{(1)}(x) = -0.429x^7 + 5.048x^5 - 15.835x^3 + 12.000x - 1.403,$$
$$\check{D}^{(2)}(x) = 1.520.$$

Both identified equations have the correct terms appearing in Eq. (5.2.2.3) and $D^{(2)}(x) = 1.2$. For the identified expression $\check{D}^{(1)}(x)$, the DIC and MDIC are equal to 2.319% and 6.47%, respectively. For the identified expression $\check{D}^{(2)}(x)$, the DIC is equal to 26.667%.

5.2.2.4 Identification of two-dimensional stochastic differential equations

The strategy for automatic iterative sampling optimization suggested above only works for identification of SDEs in one-dimensional problems. However, the Langevin equation in Eq. (5.3) can be used to describe a higher-dimensional system. Here, we present a test of the idea of using a similar sampling strategy for a two-dimensional case. We consider a two-dimensional system with two variables X and Y for the position of a particle along the x- and y direction. We consider a two-dimensional potential $U(x, y)$. The potential and the resulting two drift forces are written as

$$U(x, y) = x^4 - 3x^2 - 3xy + y^4, \tag{5.16}$$
$$g_x(x, y) = -\frac{\partial U}{\partial x} = D_x^{(1)}(x, y) = -4x^3 + 6x + 3y, \tag{5.17}$$
$$g_y(x, y) = -\frac{\partial U}{\partial y} = D_y^{(1)}(x, y) = -4y^3 + 3x. \tag{5.18}$$

The diffusion coefficient matrix is

$$\begin{pmatrix} D_{xx}^{(2)}(x, y) & D_{xy}^{(2)}(x, y) \\ D_{yx}^{(2)}(x, y) & D_{yy}^{(2)}(x, y) \end{pmatrix} = \begin{pmatrix} 0.3 & 0 \\ 0 & 0.3 \end{pmatrix}. \tag{5.19}$$

The two-dimensional potential is plotted in Fig. (5.13-a-i). We find that the potential map has two minima, $(-1.4, -1)$ and $(1.4, 1)$. The saddle point of this two-dimensional potential is at $(0, 0)$. The two drift coefficients $D_x^{(1)}(x, y)$ and $D_y^{(1)}(x, y)$ are plotted in Fig. (5.13-a-ii and iii), respectively.

Using above model, an original two-dimensional trajectory (total number of time steps $1e7$) is numerically calculated with an initial state at $(-1.4, -1)$ and a time step $\Delta t = 0.002$, illustrated in Fig. (5.13-b-Left). We find that the trajectory clusters around at the left local minimal potential point from $2 < x < -0.5$ and $-1.5 < y < 0$. Like in the one-dimensional case, the data is binned to generate discretized numerical data for $D_x^{(1)}(x, y)$, $D_y^{(1)}(x, y)$, $D_{xx}^{(2)}(x, y)$, $D_{xy}^{(2)}(x, y)$, $D_{yx}^{(2)}(x, y)$ and $D_{yy}^{(2)}(x, y)$ by using 80×80 bins. These binned data are filtered by using a probability threshold of $5e - 4$. The drift- and diffusion coefficients identified using

a

b original total step =5e7

	$D_x^{(1)}(x,y)$		$D_{xx}^{(2)}(x,y)$		$D_y^{(1)}(x,y)$		$D_{yy}^{(2)}(x,y)$		$D_{xy}^{(2)}(x,y)\,D_{yx}^{(2)}(x,y)$
"I"	"-4.1596"	"I"	"0.30743"	"I"	"0"	"I"	"0.3099"	"I"	"0"
"x"	"0"	"x"	"0"	"x"	"2.959"	"x"	"0"	"x"	"0"
"x^2"	"0"	"x^2"	"0"	"x^2"	"0"	"x^2"	"0"	"x^2"	"0"
"x^3"	"-0.90909"	"x^3"	"0"	"x^3"	"0"	"x^3"	"0"	"x^3"	"0"
"x^4"	"1.0958"	"x^4"	"0"	"x^4"	"0"	"x^4"	"0"	"x^4"	"0"
"y"	"2.9648"	"y"	"0"	"y"	"0"	"y"	"0"	"y"	"0"
"y^2"	"0"	"y^2"	"0"	"y^2"	"0"	"y^2"	"0"	"y^2"	"0"
"y^3"	"0"	"y^3"	"0"	"y^3"	"-3.9502"	"y^3"	"0"	"y^3"	"0"
"y^4"	"0"	"y^4"	"0"	"y^4"	"0"	"y^4"	"0"	"y^4"	"0"
"xy"	"0"	"xy"	"0"	"xy"	"0"	"xy"	"0"	"xy"	"0"
"xy^2"	"0"	"xy^2"	"0"	"xy^2"	"0"	"xy^2"	"0"	"xy^2"	"0"
"x^2y"	"0"	"x^2y"	"0"	"x^2y"	"0"	"x^2y"	"0"	"x^2y"	"0"
"x^2y^2"	"0"	"x^2y^2"	"0"	"x^2y^2"	"0"	"x^2y^2"	"0"	"x^2y^2"	"0"

c control total step =5e7

	$D_x^{(1)}(x,y)$		$D_{xx}^{(2)}(x,y)$		$D_y^{(1)}(x,y)$		$D_{yy}^{(2)}(x,y)$		$D_{xy}^{(2)}(x,y)\,D_{yx}^{(2)}(x,y)$
"I"	"0"	"I"	"0.30149"	"I"	"0"	"I"	"0.30232"	"I"	"0"
"x"	"6.0155"	"x"	"0"	"x"	"2.9772"	"x"	"0"	"x"	"0"
"x^2"	"0"	"x^2"	"0"	"x^2"	"0"	"x^2"	"0"	"x^2"	"0"
"x^3"	"-3.9779"	"x^3"	"0"	"x^3"	"0"	"x^3"	"0"	"x^3"	"0"
"x^4"	"0"	"x^4"	"0"	"x^4"	"0"	"x^4"	"0"	"x^4"	"0"
"y"	"2.9732"	"y"	"0"	"y"	"0"	"y"	"0"	"y"	"0"
"y^2"	"0"	"y^2"	"0"	"y^2"	"0"	"y^2"	"0"	"y^2"	"0"
"y^3"	"0"	"y^3"	"0"	"y^3"	"-3.9691"	"y^3"	"0"	"y^3"	"0"
"y^4"	"0"	"y^4"	"0"	"y^4"	"0"	"y^4"	"0"	"y^4"	"0"
"xy"	"0"	"xy"	"0"	"xy"	"0"	"xy"	"0"	"xy"	"0"
"xy^2"	"0"	"xy^2"	"0"	"xy^2"	"0"	"xy^2"	"0"	"xy^2"	"0"
"x^2y"	"0"	"x^2y"	"0"	"x^2y"	"0"	"x^2y"	"0"	"x^2y"	"0"
"x^2y^2"	"0"	"x^2y^2"	"0"	"x^2y^2"	"0"	"x^2y^2"	"0"	"x^2y^2"	"0"

Figure 5.13: The identification of two-dimensional stochastic differential equations. (a-i) A plot of two-dimensional potential $U(x,y)$ shows two local minimal potential. (a-ii-iii) The two drift equations $D_x^{(1)}(x,y)$ and $D_y^{(1)}(x,y)$ are plotted. (b-Left) An original stochastic trajectory is shown on the two-dimensional potential map, and all states are located at the region of left minimal potential. (b-Middle and Right) The drift- and diffusion coefficients equation are identified from the original trajectory data. The equations are shown in the Library by using the automatic threshold sparse Bayesian learning approach. (c-Left) The controlled trajectory data is plotted on the two-dimensional potential map, which the trajectory has larger region than original one. (c-Middle and Right) The identification of drift- and diffusion coefficients are calculated by the controlled trajectory in the library table.

ATSBL are shown in Fig. (5.13-b-Right). The identified equations are

$$\begin{pmatrix} \breve{D}_x^{(1)}(x,y) \\ \breve{D}_y^{(1)}(x,y) \end{pmatrix} = \begin{pmatrix} 1.096x^4 - 0.909x^3 + 2.965y - 4.160 \\ -3.950y^3 + 2.959x \end{pmatrix},$$
$$\begin{pmatrix} \breve{D}_{xx}^{(2)}(x,y) & \breve{D}_{xy}^{(2)}(x,y) \\ \breve{D}_{yx}^{(2)}(x,y) & \breve{D}_{yy}^{(2)}(x,y) \end{pmatrix} = \begin{pmatrix} 0.307 & 0 \\ 0 & 0.310 \end{pmatrix}.$$

Here, we find that the identified expression for $\breve{D}_x^{(1)}(x,y)$ has wrong terms compared with the real equation. For the identified expression $\breve{D}_y^{(1)}(x,y)$, the DIC and MDIC are equal to 1.308% and 1.367%, respectively. The DIC for expression $\breve{D}_{xx}^{(2)}(x,y)$ and $\breve{D}_{yy}^{(2)}(x,y)$ are respectively equal to 69.3% and 69.0%.

Next, we employ a two-dimensional one-well potential to control the double-well potential

at the saddle point $(0,0)$ in two-dimensional space. The added potential to control sampling is $U_c(x,y) = 3x^2 - 0.5y^2$. The shape of this control potential is the inverse of the saddle shape of the original potential. Using the same parameter set as in the original data, we simulate trajectories in the presence of the control potential, as illustrated in Fig. (5.13-c-Left). We find that this trajectory is spread over a much larger area than in the original data, where the region is in $-1.5 < x < 1.5$ and $-1.5 < y < 1.5$. Using the data-driven model for identification of SDEs, the binned data are first filtered by using a probability threshold $5e-4$. Then, using ATSBL, we identify the drift- and diffusion coefficients as

$$\begin{pmatrix} \breve{D}_x^{(1)}(x,y) \\ \breve{D}_y^{(1)}(x,y) \end{pmatrix} = \begin{pmatrix} -3.978x^3 + 6.016x + 2.973y \\ -3.969y^3 + 2.977x \end{pmatrix},$$

$$\begin{pmatrix} \breve{D}_{xx}^{(2)}(x,y) & \breve{D}_{xy}^{(2)}(x,y) \\ \breve{D}_{yx}^{(2)}(x,y) & \breve{D}_{yy}^{(2)}(x,y) \end{pmatrix} = \begin{pmatrix} 0.301 & 0 \\ 0 & 0.3023 \end{pmatrix}.$$

We find that these identified equations have the correct terms, see Eq. (5.18) and Eq. (5.19). For the identified expression for $\breve{D}_x^{(1)}(x,y)$, the DIC and MDIC are equal to 0.497% and 0.90%, respectively. For the identified expression for $\breve{D}_y^{(1)}(x,y)$, the DIC and MDIC are equal to 0.77% and 0.767%, respectively.

5.3 Discussion

Data-driven approaches for semi-automated inference of governing equations from measurement data are becoming increasingly popular among physicists. The data is usually recorded as a time-space series of states. The mathematical problem of identifying ordinary differential equations (ODEs), participial differential equations (PDEs) and stochastic differential equations (SDEs) can be recast as a linear equation system $\mathbf{g} = \mathbf{\Phi w} + \mathbf{s}$, where $\mathbf{w}$ are the coefficients of different terms in the equations and $\mathbf{g}$ and $\mathbf{\Phi}$ are constructed from the measurement data [39, 40, 174]. Thus, data-driven identification of governing equations essentially results in an inverse problem that is usually overdetermined and possibly ill-posed. Since the goal is to obtain a simple governing equation, sparse learning approaches with a threshold have previously been introduced to solve the inverse problem, for example, the sequential thresholded least squares algorithm [39], TrainSTRidge [40], Stepwise Sparse Regressor [174] and Threshold sparse Bayesian regression [172]. However, these approaches require the input of at least one parameter, e.g., a regularization parameter or a threshold.

Bayesian methodology can be employed to devise algorithms for generating sparse solutions to inverse problems that do not rely on explicit provision of tuning parameters. Sparse Bayesian learning can be implemented as an efficient algorithm by using the relevance vector machine [69, 70]. In the field of compressive sensing, an algorithm, called Bayesian compressive sensing using Laplace priors has gained substantial popularity [215]. Alternatively, threshold parameters can be automatically estimated by using a training approach [40]. Here, we combine the Bayesian compressive sensing using Laplace priors with the training method in an approach that we call automatic threshold sparse Bayesian learning (ATSBL), which is a parameter-free method.

ATSBL can be employed to identify ordinary-, partial- and stochastic differential equations. Noise in the measurement can drastically impair the inference of differential equations due

to the presence of high-derivative terms in the library. We test established filtering procedures employing a neural network implementation with TensorFlow and find that this approach can improve the robustness of data-driven inference of ODEs in the presence of noise.

In our data-driven approach for discovery of SDEs, the trajectory data is dealt with as binned data. To improve the quality of the probability distributions underlying the model-inference procedure, we filter out highly uncertain binned data by using a probability threshold. The optimal probability threshold is selected as the corner value on the log-log curve of the total uncertainty vs different probability thresholds. To our knowledge, this approach is new and we find that thresholding the probability according to our heuristic significantly improves the accuracy of the identified SDE. If one can manipulate the studied stochastic system, for example, in a simulation or in a well-defined experimental environment, one can improve the precision of the sampled distributions by adding additional potential forces as is done in "Umbrella Sampling" [255, 258]. To leverage this concept for improving data-driven discovery of SDE, we develop an automatic iterative sampling optimization which employs an automatically optimized control potential for one-dimensional problems. Combining ATSBL with iterative re-sampling in the presence of an adaptive potential significantly improves the accuracy of data-driven identification of SDE.

Chapter 6

Conclusion

The purpose of this thesis is to develop and test advanced data analysis methods for solving inverse problems in the context of traction force microscopy and data-driven inference of governing differential equations. The employed new methods allow us to significantly improve the precision and reliability of the results in both applications.

Ill-conditioned, inverse problems have the characteristic that a small perturbation in the measurement data leads to large variations in the solutions. The classical approach to solve this problem is regularization, where the target function is supplemented with penalty terms that render the solution less sensitive to noise. According to the different penalty terms, regularization methods can be classified into L2-, L1-, and elastic net regularization. These approaches require one to provide one or two regularization parameters. These regularization techniques can be motivated in a Bayesian framework where the penalty terms result from prior knowledge about the statistical nature of the solution. The Bayesian perspective, however, allows one to go much further in that it provides means to estimate the optimal regularization parameters and can produce solutions with desired characteristics, for example solutions with a sparse, hierarchical structure.

The first class of inverse problems that are studied occur in Traction Force Microscopy. Traction force microscopy is a versatile and perturbation-free method yielding a spatial image of the forces exerted by cells on elastic substrates. The traction forces are calculated by using a linear mechanical model between the forces and displacements on the surface of the substrates. We systematically compare the classical L2- and L1 regularization with three other regularization methods which have not yet been employed for TFM. These three regularization approaches are the Elastic Net (EN), Proximal Gradient Lasso (PGL) and Proximal Gradient Elastic Net (PGEN). We find that Elastic Net regularization combines the L1- with L2 regularization and outperforms all other regularization methods with regard to accuracy of traction reconstruction. Further, we develop two methods, Bayesian L2 regularization (BL2) and advanced Bayesian L2 regularization (ABL2), for automatic, optimal L2 regularization in real space. In Fourier space, we combine the robustness of BL2 with the computation speed of Fourier transform traction cytometry (FTTC) in a new algorithm called Bayesian Fourier-transform traction cytometry (BFTTC). This BFTTC method is made freely available as a software package with graphical user-interface for intuitive usage. Using artificial data and experimental data, we show that these Bayesian methods enable robust reconstruction of traction without requiring a difficult selection of regularization parameters specifically for each data set.

Perspectives for future improvement of TFM perhaps lie in a taylored use of prior distributions for different applications. We find that Elastic Net regularization, combining L1- with L2 regularizations, is better than the two methods with regard to accuracy of traction reconstruction. However, Bayesian Elastic Net algorithms tested in this thesis did not perform well. In the future, a proper Bayesian Elastic Net regularization algorithm could be developed for TFM. Such a dedicated algorithm could potentially adapt the weight of different priors automatically and generate results with superior precision and accuracy. Further improvements could result from three-dimensional traction analysis and the incorporation of additional data dimensions in the calculations of cellular tractions.

The second class of inverse problem we study in this thesis occurs in data-driven discovery of differential equations. The data-driven approach for the discovery of governing physical differential equations, ODEs, PDEs and SDEs, is also called symbolic regression and relies on extensive use of process data to evaluate libraries of mathematical expressions. We develop a new algorithm for solving the sparse, linear problem related to finding the optimal terms in the library and call the algorithm automatic threshold sparse Bayesian learning (ATSBL). ATSBL is a parameter free approach in which the regularization parameter is selected by using Bayesian sparse learning approach. Using ATSBL, we can identify ODEs and PDEs precisely in synthetic test data. For the identification of SDEs, we introduce a probability threshold to filter out highly uncertain binned data. Our heuristic for finding an optimal probability threshold is to choose the corner value on the log-log curve of the total uncertainty versus different probability thresholds. The identification of SDEs is improved by using this probability threshold. Furthermore, to improve data-driven discovery of SDEs, additional potential forces are employed to manipulate stochastic systems as is done in "Umbrella Sampling". For finding optimal perturbations of the stochastic process, we suggest an automatic iterative sampling optimization which employs an automatically optimized control potential.

Regarding automated identification of ordinary and partial differential equations from data, the main issue that remains to be tackled is the fact that most processes are only observed partially. In our work, we only identify the governing equations from the given complete measurement data. In reality, much information may be hidden in the data. For example, imagine a the Lorenz system, where only one coordinate x is given, while the coordinates y and z are hidden. For this case, the hidden data may be estimated by using Kalman filtering. Such filtering approaches could be combined with sparse inference schemes like ATSBL. Regarding the data-driven identification of SDEs, we find that the identification is improved by our probability thresholding procedure. In the future, one may build on this finding to devise optimal data rejection procedures based on information theory. Furthermore, one may devise optimal protocols for the iterative perturbation of accessible stochastic processes that facilitate data-driven learing of the underlying equations. The coming years will certainly see the development of a multitude of novel methods that allow researchers to generate models automatically from their data, which will speed up scientific progress in many fields.

Appendix A

Supplementary material for Chapter 2

A.1   Implementation of the regularization routines

For L2 regularization in real space, we employ a singular value decomposition and the routines "tikhonov" and "l_curve" provided by the Matlab package "Regularization Tools" [62]. The use of this package for the traction force microscopy has been described earlier [37, 157]. To perform L1- and EN regularization we minimize well-established formulas [55] using the convex optimization package CVX [196, 197]. The target functions for L1- and EN regularization are given by

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}}\big[\mathbf{w}^{\mathbf{T}}\mathbf{\Phi}^{\mathbf{T}}\mathbf{\Phi}\mathbf{w} - 2\mathbf{g}^{\mathbf{T}}\mathbf{\Phi}\mathbf{w} + \mathbf{w}^{\mathbf{T}}\mathbf{w} + \lambda_1\|\mathbf{w}\|_1\big], \tag{A.1}$$

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}}\bigg[\mathbf{w}^{\mathbf{T}}\Big(\frac{\mathbf{\Phi}^{\mathbf{T}}\mathbf{\Phi} + \lambda_2\mathbf{I}}{1 + \lambda_2}\Big)\mathbf{w} - 2\mathbf{g}^{\mathbf{T}}\mathbf{\Phi}\mathbf{w} + \lambda_1\|\mathbf{w}\|_1\bigg]. \tag{A.2}$$

Comparison of the two formulas shows that EN regularization is a stabilized version of the L1 regularization. Below, we provides a short Matlab code for L1- and EN regularization.

```matlab
function F = L1_EN_cvx(n,X,u,lambda1,lambda2)
% This code requires the Convex optimization package CVX.
% n is the lenght of the  vector "w"; X in the matix "Phi".
% u is given vector "g".
% lambda1 and lambda2 are regularization parameters;

a = size(X);
R= eye(a(2))

cvx_begin
cvx_solver sedumi;
variable f(n);

% This line is for L1 regularization. Comment out if not needed.
% minimize((X*f-u)'*(X*f-u)+lambda1*(norm(f,1)));
```

```
% This line is for EN regularization. Comment out if not needed.
minimize(f'*(X'*X+lambda2*R)/(1+lambda2)*f-2*u'*X*f+ ...
            lambda1*(norm(f,1)));

cvx_end;
end
```

A.2   Bayesian Lasso (BL) and Bayesian elastic net (BEN)

As an alternative to the variational Bayes approach, hierarchical models can be solved by Markov chain Monte Carlo methods. To test this approach in conjunction with complex models, we employ two models provided as Matlab packages together with Ref. [75]. The BL is based on the same network structure as the BCSL shown in Fig. 2.4 (b). Furthermore, we also tested a Bayesian version of the elastic net (BEN) where the network is shown in Fig. 2.4 (c). Both algorithms perform similarly to the BCSL for very low noise. Here, the assumption of sparsity helps to produce a clear background and allows to distinguish traction sites clearly. However, traction magnitude estimates are strongly exaggerated. As expected, the full solution of the models via Monte Carlo sampling makes the performance of BL and BEN slightly more robust than BCSL in the presence of noise. Overall, the performance of these methods in the context of TFM is unsatisfactory.

Appendix B

Supplementary material for Chapter 3

B.1 An analytical solution for displacements resulting from a circular traction patch for $z \geq 0$

We have provided the analytical solution for the displacement field in $z \geq 0$ resulting from local traction spots as functions of integrals of Lipschitz-Hankel I(1,0;-1), I(1,1;-2), I(1,0;0), I(1,1;-1), I(2,1;-1), and I(2,1;0). The Lipschitz-Hankel functions are related to Bessel functions via the integral $I(\mu, \gamma, \lambda) = \int_0^\infty J_\mu(at) J_\gamma(bt) t^\lambda e^{-ct} \mathrm{d}t$. The integrals I(1,0;-1),I(1,1;-2), I(1,0;0) and I(1,1;-1) are given in Refs. [259, 260]

$$I(1, 0; -1) = \frac{\sqrt{ab}}{ka} E_0(k) + \frac{(a^2 - b^2)k}{4a\sqrt{ab}} F_0(k) + \frac{c}{2a} \Lambda_0^*(\alpha, \beta) - \frac{c}{a},$$

$$I(1, 1; -2) = \frac{2(a^2 + b^2) - c^2}{6ka\sqrt{ab}} E_0(k) + \frac{c^2(2a^2 + 2b^2 + c^2) - 2(a^2 - b^2)^2}{24a^2b\sqrt{ab}} k F_0(k)$$
$$- \frac{(a^2 - b^2)c}{4a^2b} \Lambda_0^*(\alpha, \beta) - \frac{bc}{2a^2},$$

$$I(1, 0; 0) = -\frac{kc}{4a\sqrt{ab}} F_0(k) - \frac{1}{2a} \Lambda_0^*(\alpha, \beta) + \frac{1}{a},$$

$$I(1, 1; -1) = \frac{c}{2k\sqrt{ab}} E_0(k) - \frac{kc(2a^2 + 2b^2 + c^2)}{8ab\sqrt{ab}} F_0(k) + \frac{a^2 - b^2}{4ab} \Lambda_0^*(\alpha, \beta) + \frac{b}{2a},$$

with

$$\Lambda_0^*(\alpha, \beta) = \begin{cases} \Lambda_0(\alpha, \beta), & a > b \\ 1, & a = b \\ 2 - \Lambda_0(\alpha, \beta), & a < b \end{cases}.$$

Here, $\Lambda_0(\alpha, \beta)$ is the Heuman lambda function, $F_0(k)$ is a complete elliptic integral of the first kind and $E_0(k)$ is a complete elliptic integral of the second kind. k is defined follows

$$k^2 = \sin^2 \alpha = \frac{4ab}{(a+b)^2 + c^2},$$

$$\sin^2 \beta = \frac{c^2}{(a-b)^2 + c^2}.$$

$I(1, 2; -1)$ and $I(1, 2; 0)$ are given by a recurrence relation of the following form

$$I(1, 2; -1) = \frac{2}{b}I(1, 1; -2) - I(1, 0; -1),$$

$$I(1, 2; 0) = \frac{2}{b}I(1, 1; -1) - I(1, 0; 0).$$

Thus, we have a complete set of analytical expressions for the displacement field. We show an example with one traction force patch in Fig. B.1 (a). Two different displacement fields obtained by 2D theory and semi-analytical theory, where we evaluate the integrals numerically, are shown in Fig. B.1 (b). Evaluation of the analytical expressions without use of numerical integration was found to lead to significant numerical errors.
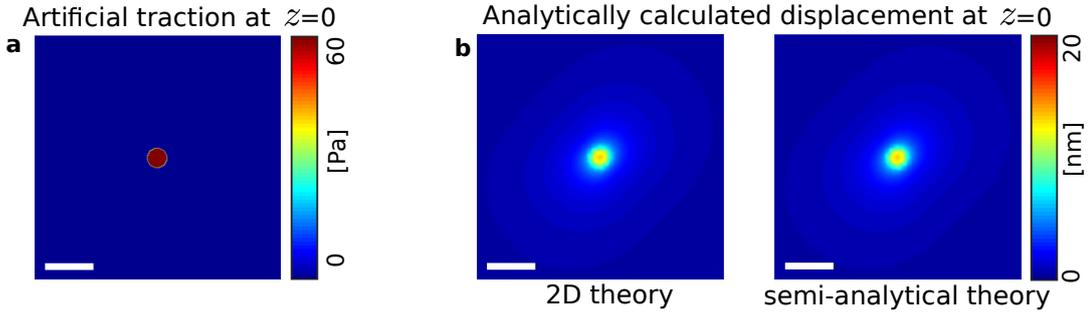
Figure B.1: Analytical displacement fields around one circular traction patch. (a) The one circular traction patch is applied on a surface of a substrate and the magnitude of the traction is $60\,\mathrm{Pa}$. (b) Two different displacement fields are calculated by 2D theory and semi-analytical theory. Space bar: $5\,\mu\mathrm{m}$.

B.2 Supplementary figures
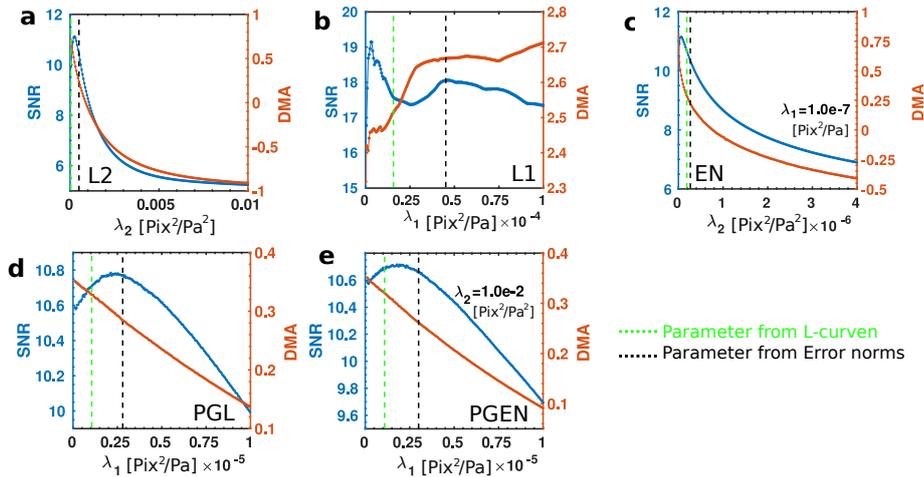
Figure B.2: Additional error quantification for the regularization examples shown in Fig. 2 of the main text. Figures (a)-(e) are the signal to noise ratios (SNR) and deviations of traction maxixima (DMA) for the same tests as shown in Fig. 2(c) of the main text.
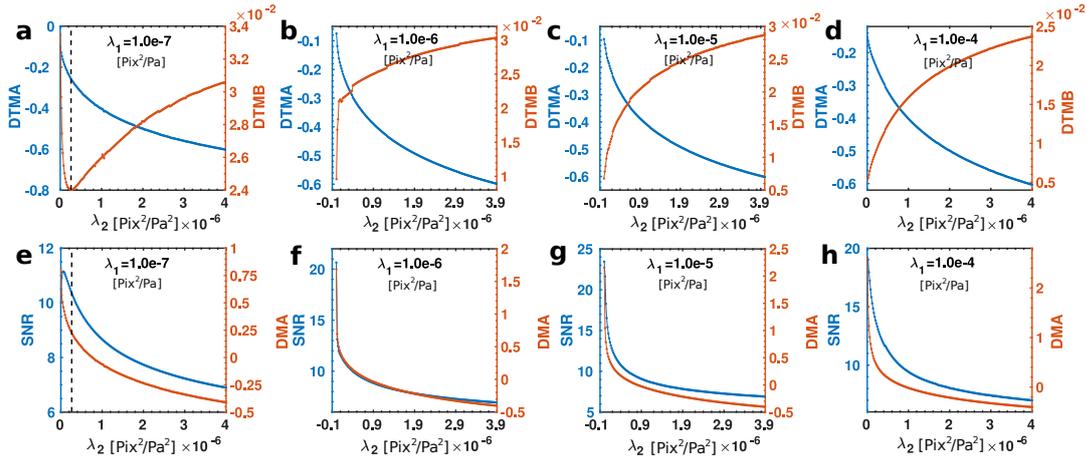
Figure B.3: Parameter-dependence of EN regularization error at fixed values of λ_1. (a)-(d) The DTMA and DTMB. (e)-(h) SNR and DTM.
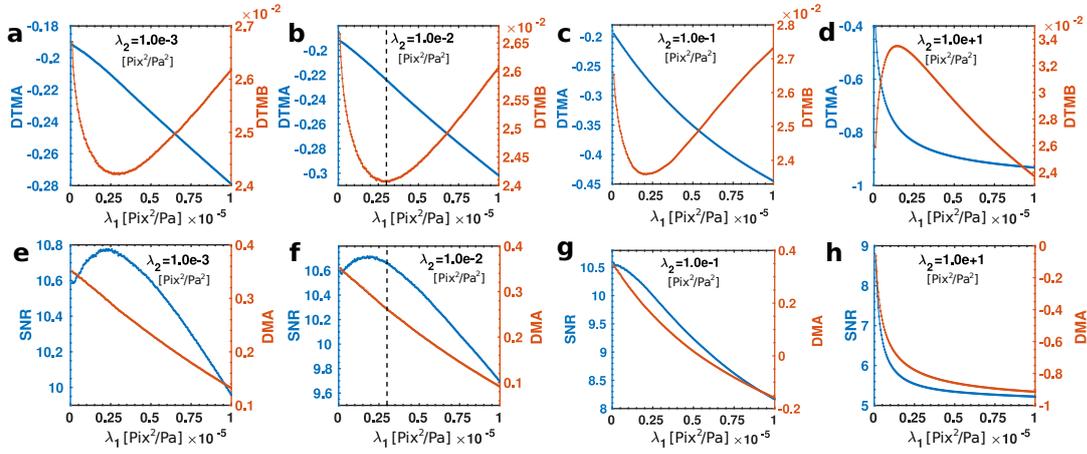
Figure B.4: Parameter-dependence of PGEN regularization error at fixed values of λ_2. (a)-(d) Error norms DTMA and DTMB. (e)-(h) SNR and DTM.
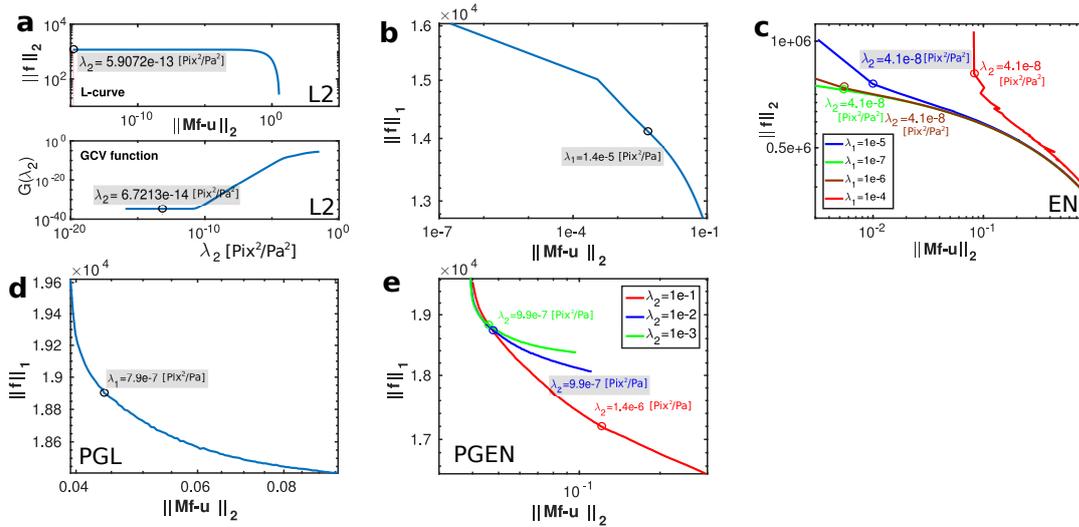
Figure B.5: L-curves for the regularization methods shown in Fig. 2 of the main text. (a) L-curve and GCV function for the classical L2 regularization. (b)-(c) L-curves for L1- and EN-regularization. (d)-(e) L-curves for PGL and PGEN.
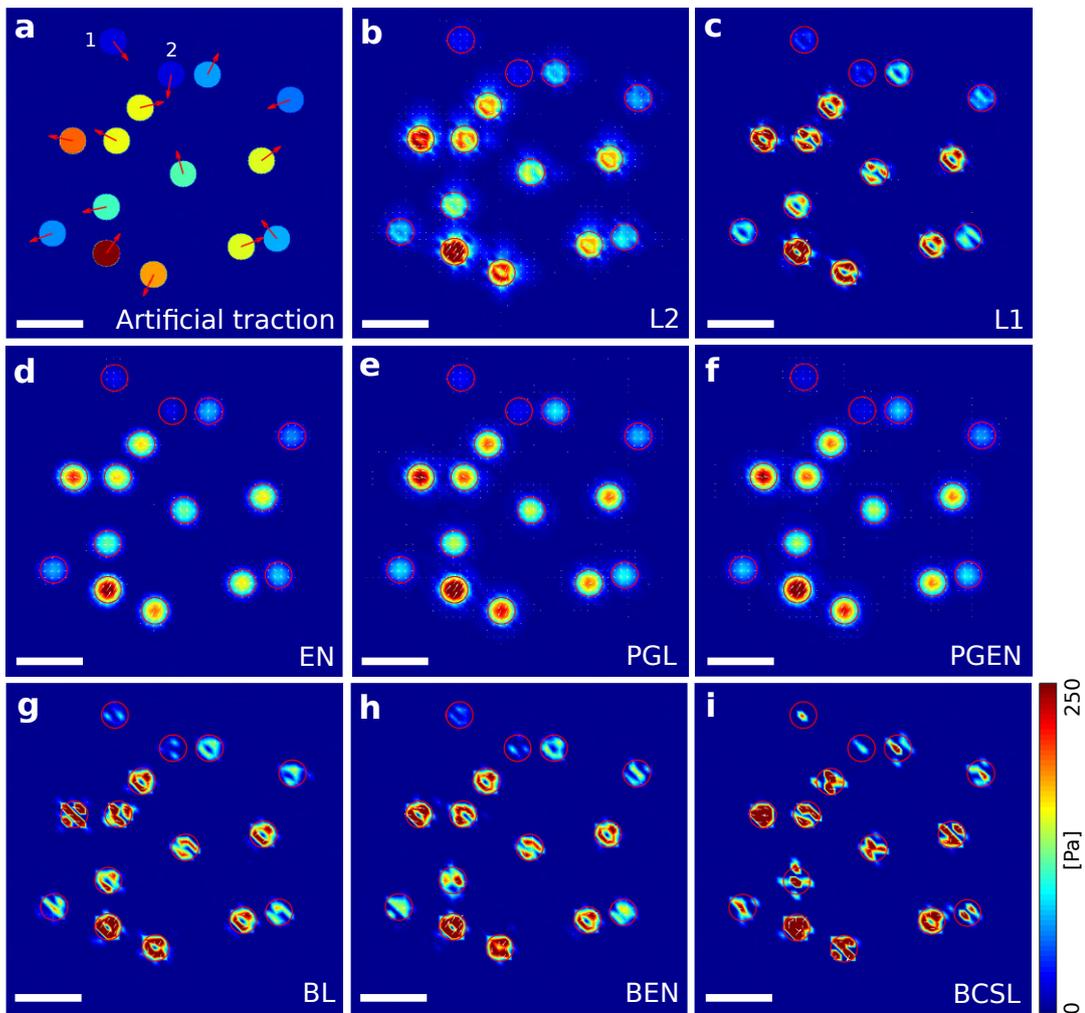
Figure B.6: Exemplary traction fields reconstructed from noise-free artificial data. (a) The artificial data consists of 15 circular traction patterns with random magnitude (from 0-250 Pa). (b)-(f) Reconstructed traction fields obtained with the regularization methods L2, L1, EN, PGL and PGEN, respectively. (**g**)-(**i**) Reconstructed traction fields obtained using the complex Bayesian hierarchical network algorithms BL, BEN and BCSL. Space bar 5 μm.
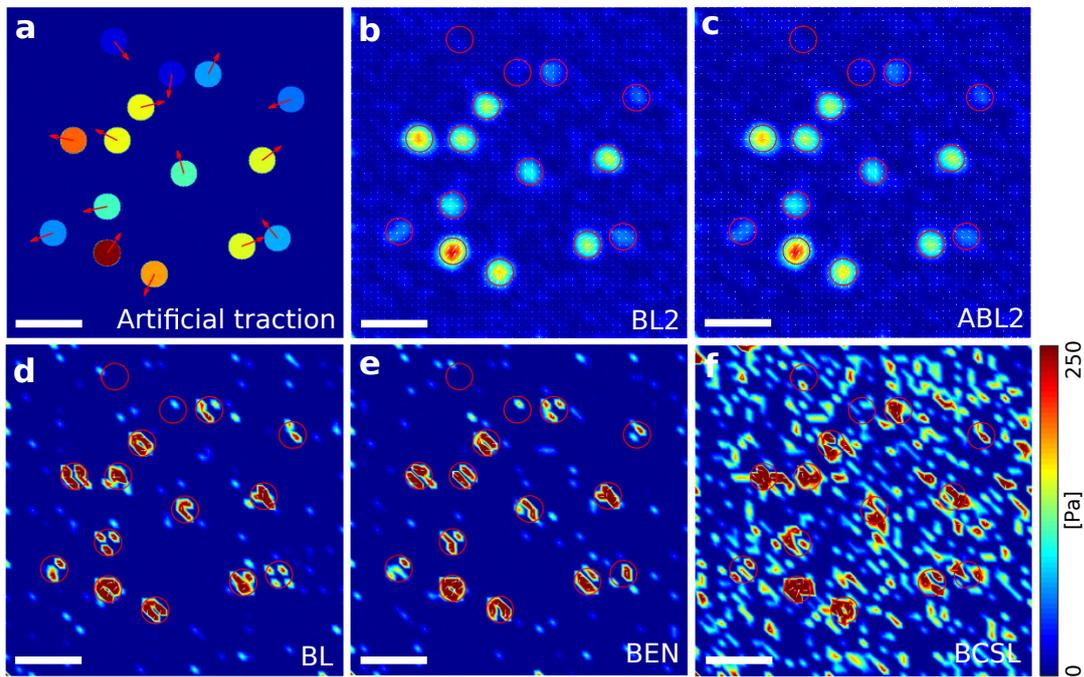
Figure B.7: Exemplary traction fields reconstructed with Bayesian methods from artificial data containing 5% noise. (a) Artificial data as in Fig. **??** (a), but 5% Gaussian noise is added to the displacements. (b)-(c) Reconstructed traction from BL2 and ABL2, respectively. (d)-(f) Reconstruction traction using the Bayesian hierarchical network algorithms BL, BEN and BCSL, respectively. Space bar 5 μm.
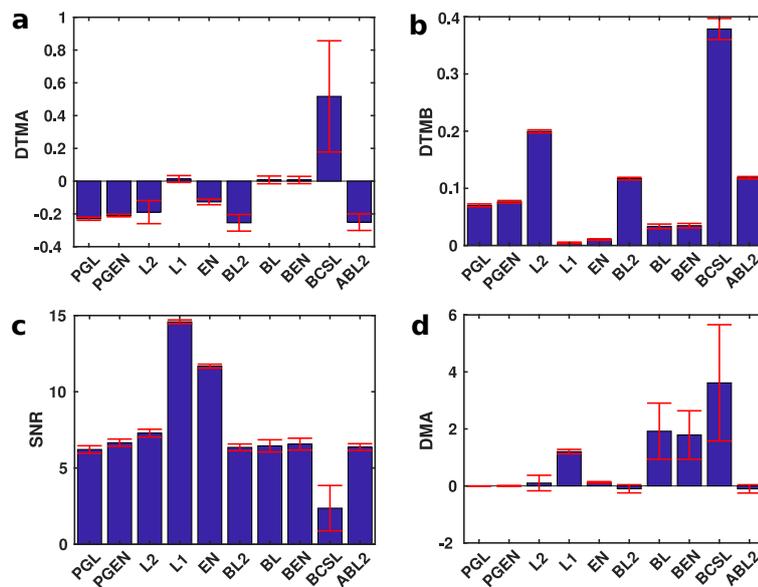
Figure B.8: Exemplary comparison of errors of ten different methods employed for reconstruction of artificial data with 5% noise. (a) DTMA with error bars, s.d. (standard deviation error bars). (b) DTMB with error bars, s.e.m.(standard error of the mean). (c) SNR with error bars, s.e.m. (d) DMA with error bars, s.d.
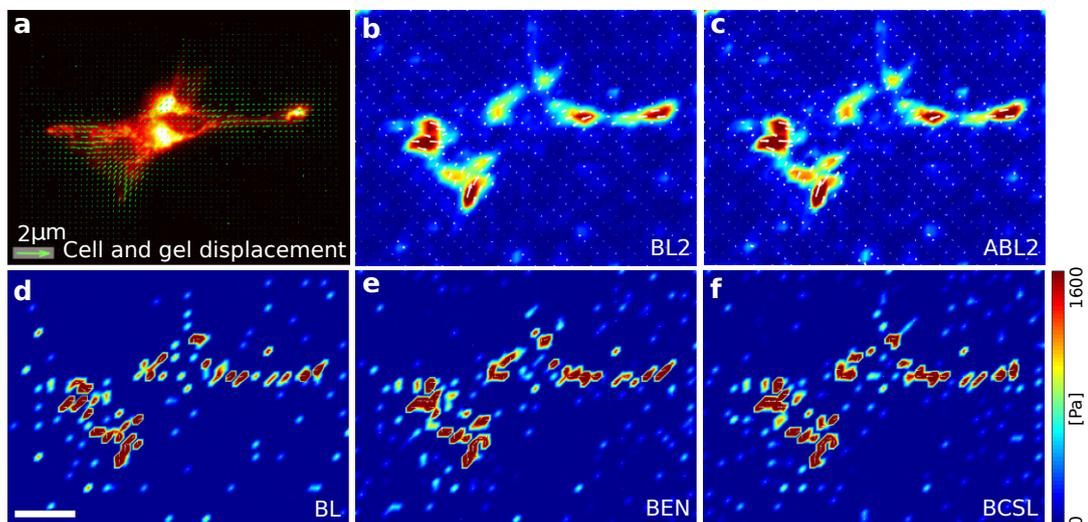
Figure B.9: Comparison of various Bayesian methods for TFM. (a) Image of cell and displacement field (Green vectors). (b)-(c) Reconstruction using BL2 and ABL2, respectively.(d)-(f) The hierarchical Bayesian network algorithms BL, BEN, and BCSL produce very sparse patterns with strongly overestimated traction. Space bar 25 μm.

Appendix C

Description of the FastLaplace algorithm

A sparse Bayesian learning approach called Bayesian compressive sensing using Laplace priors (BCSL) has been proposed in Ref. [215]. Here, we show the pseudocode for BCSL, called FastLaplace, which can also be found in Ref. [215].

Function: FastLaplace($\boldsymbol{\Theta}$, $\mathbf{g}$)

%% Initialize all

$\gamma_i = 0$, $\lambda = 0$;

while *convergence criterion not met* **do**

    Choose a γ_i (or equivalently choose a basis vector ϕ_i);

    if $q_i^2 - s_i > \lambda$ *and* $\gamma_i = 0$ **then**

       | add γ_i to the model;

    else if $q_i^2 - s_i > \lambda$ *and* $\gamma_i > 0$ **then**

       | Re-estimate γ_i;

    else if $q_i^2 - s_i < \lambda$ **then**

       | Prune i from the model (set $\gamma_i = 0$);

    end if

    Update $\boldsymbol{\Sigma}$ and μ;

    Update s_i and q_i;

    Update λ using Eq. (2.29a);

    Update ν using Eq. (2.29b);

end

return $\mathbf{w}$, $\boldsymbol{\Sigma}$, γ;

Algorithm 3: Pseudocode for FastLaplace for Bayesian compressive sensing using Laplace priors (BCSL)

Bibliography

[1]   John W Tukey. "The future of data analysis". In: *The annals of mathematical statistics* 33.1 (1962), pp. 1–67.

[2]   George AF Seber and Alan J Lee. *Linear regression analysis*. Vol. 329. John Wiley & Sons, 2012.

[3]   Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining. *Introduction to linear regression analysis*. Vol. 821. John Wiley & Sons, 2012.

[4]   David A Ratkowsky and David EA Giles. *Handbook of nonlinear regression models*. 04; QA278. 2, R3. M. Dekker New York, 1990.

[5]   Halbert White and Ian Domowitz. "Nonlinear regression with dependent observations". In: *Econometrica: Journal of the Econometric Society* (1984), pp. 143–161.

[6]   Harvey J Motulsky and Ronald E Brown. "Detecting outliers when fitting data with nonlinear regression–a new method based on robust nonlinear regression and the false discovery rate". In: *BMC bioinformatics* 7.1 (2006), p. 123.

[7]   Sergei Igorevich Kabanikhin. "Definitions and examples of inverse and ill-posed problems". In: *Journal of inverse and Ill-posed problems* 16.4 (2008), pp. 317–357.

[8]   Robin L Plackett. "Studies in the History of Probability and Statistics. XXIX: The discovery of the method of least squares". In: *Biometrika* 59.2 (1972), pp. 239–251.

[9]   Adrien Marie Legendre. *Nouvelles méthodes pour la détermination des orbites des comètes*. F. Didot, 1805.

[10]   Stephen M Stigler. "Gauss and the invention of least squares". In: *The Annals of Statistics* (1981), pp. 465–474.

[11]   Otto Bretscher. *Linear algebra with applications*. Prentice Hall Eaglewood Cliffs, NJ, 1997.

[12]   Carl Friedrich Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*. Vol. 7. Perthes et Besser, 1809.

[13]   Friedrich Robert Helmert. "Über die Wahrscheinlichkeit der Potenzsummen der Beobachtungsfehler". In: *Z. Math. u. Phys* 21 (1876), pp. 192–218.

[14]   Anders Hald. *A History of Mathematical Statistics from 1750 to 1930*. Vol. 2. 4. Wiley New York, 1998.

[15]   Karl Pearson. "X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 50.302 (1900), pp. 157–175.

[16]   Johann Heinrich Lambert. "Photometria, sive de mensura et gradibus luminis". In: *Colorum et Umbrae, Eberhard Klett, Augsberg, Germany* (1760).

[17]   Daniel Bernoulli. "Dijudicatio maxime probabilis plurium observationum discrepantium atque verisimillima inductio inde formanda". In: *Acta Acad. Sci. Petropolit* 1 (1769), pp. 3–33.

[18]   Pierre S De Laplace. "Mémoire sur la probabilité des causes par les événements". In: *Mém. de math. et phys. présentés à l'Acad. roy. des sci* 6 (1774), pp. 621–656.

[19]   Rodolfo Guzzi. *Data assimilation: mathematical concepts and instructive examples*. Springer.

[20]   Anders Hald et al. "On the history of maximum likelihood in relation to inverse probability and least squares". In: *Statistical Science* 14.2 (1999), pp. 214–222.

[21]   C Radhakrishna Rao et al. "RA Fisher: The founder of modern statistics". In: *Statistical Science* 7.1 (1992), pp. 34–48.

[22]   John Aldrich et al. "RA Fisher and the making of maximum likelihood 1912-1922". In: *Statistical science* 12.3 (1997), pp. 162–176.

[23]   Ronald A Fisher. "On an absolute criterion for fitting frequency curves". In: *Messenger of Mathematics* 41 (1912), pp. 155–156.

[24]   Ronald Aylmer Fisher et al. "014: On the" Probable Error" of a Coefficient of Correlation Deduced from a Small Sample." In: (1921).

[25]   Ronald A Fisher. "On the mathematical foundations of theoretical statistics". In: *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character* 222.594-604 (1922), pp. 309–368.

[26]   Eric Zivot. "Maximum likelihood estimation". In: *Lecture notes* (2001).

[27]   Samuel S Wilks. "The large-sample distribution of the likelihood ratio for testing composite hypotheses". In: *The annals of mathematical statistics* 9.1 (1938), pp. 60–62.

[28]   Thomas Bayes. "LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S". In: *Philosophical transactions of the Royal Society of London* 53 (1763), pp. 370–418.

[29]   Pierre Simon Laplace. "Théorie analytique des Probabilités, 2 vols". In: *Paris: Courcier Imprimeur* (1812).

[30]   Andrew Gelman et al. *Bayesian data analysis*. CRC press, 2013.

[31]   Jacques Hadamard. "Sur les problèmes aux dérivées partielles et leur signification physique". In: *Princeton university bulletin* (1902), pp. 49–52.

[32]   Andrey N Tikhonov and Vasiliy Y Arsenin. "Solutions of ill-posed problems". In: *New York* (1977), pp. 1–30.

[33]   Aleksandr Mihajlovič Denisov. *Elements of the theory of inverse problems*. Vol. 14. VSP, 1999.

[34]   Lamia Jaafar Belaid, Amel Ben Abda, and Nawal Al Malki. "The Cauchy problem for the Laplace equation and application to image inpainting". In: *ISRN Mathematical Analysis* 2011 (2011).

[35]   H Malcolm Hudson and Richard S Larkin. "Accelerated image reconstruction using ordered subsets of projection data". In: *IEEE transactions on medical imaging* 13.4 (1994), pp. 601–609.

[36]   Sung Cheol Park, Min Kyu Park, and Moon Gi Kang. "Super-resolution image reconstruction: a technical overview". In: *IEEE signal processing magazine* 20.3 (2003), pp. 21–36.

[37]   Benedikt Sabass et al. "High resolution traction force microscopy based on experimental and computational advances". In: *Biophysical journal* 94.1 (2008), pp. 207–220.

[38]   Yunfei Huang, Gerhard Gompper, and Benedikt Sabass. "A Bayesian traction force microscopy method with automated denoising in a user-friendly software package". In: *Computer Physics Communications* (2020), p. 107313.

[39] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. "Discovering governing equations from data by sparse identification of nonlinear dynamical systems". In: *Proceedings of the National Academy of Sciences* 113.15 (2016), pp. 3932–3937.

[40] Samuel H Rudy et al. "Data-driven discovery of partial differential equations". In: *Science Advances* 3.4 (2017), e1602614.

[41] Marielba Rojas and Danny C Sorensen. "A trust-region approach to the regularization of large-scale discrete forms of ill-posed problems". In: *SIAM Journal on Scientific Computing* 23.6 (2002), pp. 1842–1860.

[42] DS Anikonov and AE Kovtanyuk. *Transport equation and tomography*. Vol. 30. Vsp, 2002.

[43] Rafael Molina. "On the hierarchical Bayesian approach to image restoration: applications to astronomical images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16.11 (1994), pp. 1122–1128.

[44] Ilker T Telci and Mustafa M Aral. "Contaminant source location identification in river networks using water quality monitoring systems for exposure analysis". In: *Water Quality, Exposure and Health* 2.3-4 (2011), pp. 205–218.

[45] Sergey I Kabanikhin. *Inverse and ill-posed problems: theory and applications*. Vol. 55. Walter De Gruyter, 2011.

[46] Andrey Nikolayevich Tikhonov. "On the stability of inverse problems". In: *Dokl. Akad. Nauk SSSR*. Vol. 39. 1943, pp. 195–198.

[47] David L Phillips. "A technique for the numerical solution of certain integral equations of the first kind". In: *Journal of the ACM (JACM)* 9.1 (1962), pp. 84–97.

[48] Fadil Santosa and William W Symes. "Linear inversion of band-limited reflection seismograms". In: *SIAM Journal on Scientific and Statistical Computing* 7.4 (1986), pp. 1307–1330.

[49] Robert Tibshirani. "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 267–288.

[50] Dean P Foster and Edward I George. "The risk inflation criterion for multiple regression". In: *The Annals of Statistics* (1994), pp. 1947–1975.

[51] Ryan Tibshirani. "A closer look at sparse regression". In: *Lecture notes* (2016).

[52] Dongyu Lin, Dean P Foster, and Lyle H Ungar. "A risk ratio comparison of l0 and l1 penalized regressions". In: *University of Pennsylvania, techical report* (2010).

[53] Cun-Hui Zhang et al. "Nearly unbiased variable selection under minimax concave penalty". In: *The Annals of statistics* 38.2 (2010), pp. 894–942.

[54] Lee Dicker, Baosheng Huang, and Xihong Lin. "Variable selection and estimation with the seamless-L 0 penalty". In: *Statistica Sinica* (2013), pp. 929–962.

[55] Hui Zou and Trevor Hastie. "Regularization and variable selection via the elastic net". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2 (2005), pp. 301–320.

[56] Jordi Barretina et al. "The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity". In: *Nature* 483.7391 (2012), pp. 603–607.

[57] Mathew J Garnett et al. "Systematic identification of genomic markers of drug sensitivity in cancer cells". In: *Nature* 483.7391 (2012), pp. 570–575.

[58] Eric E Schadt et al. "Mapping the genetic architecture of gene expression in human liver". In: *PLoS biology* 6.5 (2008).

[59]   Shinichi Sunagawa et al. "Structure and function of the global ocean microbiome". In: *Science* 348.6237 (2015), p. 1261359.

[60]   Lishan Qiao, Songcan Chen, and Xiaoyang Tan. "Sparsity preserving projections with applications to face recognition". In: *Pattern Recognition* 43.1 (2010), pp. 331–341.

[61]   Per Christian Hansen. "Analysis of discrete ill-posed problems by means of the L-curve". In: *SIAM review* 34.4 (1992), pp. 561–580.

[62]   Per Christian Hansen. "Regularization tools version 4.0 for Matlab 7.3". In: *Numerical algorithms* 46.2 (2007), pp. 189–194.

[63]   Jinlong Dong et al. "Inverse Problem Solution and Regularization Parameter Selection for Current Distribution Reconstruction in Switching Arcs by Inverting Magnetic Fields". In: *Mathematical Problems in Engineering* 2018 (2018).

[64]   Trevor Park and George Casella. "The bayesian lasso". In: *Journal of the American Statistical Association* 103.482 (2008), pp. 681–686.

[65]   Jinshan Pan et al. "Blind image deblurring using dark channel prior". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 1628–1636.

[66]   Anhui Huang, Shizhong Xu, and Xiaodong Cai. "Empirical Bayesian elastic net for multiple quantitative trait locus mapping". In: *Heredity* 114.1 (2015), pp. 107–115.

[67]   David JC MacKay. "Bayesian methods for adaptive models". PhD thesis. California Institute of Technology, 1991.

[68]   David JC MacKay. "Bayesian interpolation". In: *Neural computation* 4.3 (1992), pp. 415–447.

[69]   Michael E Tipping. "Sparse Bayesian learning and the relevance vector machine". In: *Journal of machine learning research* 1.Jun (2001), pp. 211–244.

[70]   Michael E Tipping, Anita C Faul, et al. "Fast marginal likelihood maximisation for sparse Bayesian models." In: *AISTATS*. 2003.

[71]   Shihao Ji, Ya Xue, and Lawrence Carin. "Bayesian compressive sensing". In: *IEEE Transactions on Signal Processing* 56.6 (2008), pp. 2346–2356.

[72]   SD Babacan, R Molina, and AK Katsaggelos. "Bayesian Compressive Sensing Using Laplace Priors". In: *IEEE Transactions on Image Processing* 1.19 (2010), pp. 53–63.

[73]   Peter J Green. "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination". In: *Biometrika* 82.4 (1995), pp. 711–732.

[74]   Walter R Gilks, Sylvia Richardson, and David J Spiegelhalter. "Introducing markov chain monte carlo". In: *Markov chain Monte Carlo in practice* 1 (1996), p. 19.

[75]   Dimitris Korobilis. "Hierarchical shrinkage priors for dynamic regressions with many predictors". In: *International Journal of Forecasting* 29.1 (2013), pp. 43–59.

[76]   Dani Gamerman and Hedibert F Lopes. *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. CRC Press, 2006.

[77]   Qing Li, Nan Lin, et al. "The Bayesian elastic net". In: *Bayesian analysis* 5.1 (2010), pp. 151–170.

[78]   Sherry H Suyu et al. "A Bayesian analysis of regularized source inversions in gravitational lensing". In: *Monthly Notices of the Royal Astronomical Society* 371.2 (2006), pp. 983–998.

[79]   Warren S McCulloch and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5.4 (1943), pp. 115–133.

[80]   David J. C. MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[81]   Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.

[82]   Gert-Jan Both et al. "DeepMoD: Deep learning for Model Discovery in noisy data". In: *arXiv preprint arXiv:1904.09406* (2019).

[83]   Steven Munevar, Yu-li Wang, and Micah Dembo. "Traction force microscopy of migrating normal and H-ras transformed 3T3 fibroblasts". In: *Biophysical journal* 80.4 (2001), pp. 1744–1757.

[84]   Yan-Ting Shiu et al. "Rho mediates the shear-enhancement of endothelial cell migration and traction force generation". In: *Biophysical journal* 86.4 (2004), pp. 2558–2565.

[85]   Carl-Philipp Heisenberg and Yohanns Bellaïche. "Forces in tissue morphogenesis and patterning". In: *Cell* 153.5 (2013), pp. 948–962.

[86]   Venkat Maruthamuthu et al. "Cell-ECM traction force modulates endogenous tension at cell–cell contacts". In: *Proceedings of the National Academy of Sciences* 108.12 (2011), pp. 4708–4713.

[87]   Agustí Brugués et al. "Forces driving epithelial wound healing". In: *Nature physics* 10.9 (2014), p. 683.

[88]   Bin Li and James H-C Wang. "Fibroblasts and myofibroblasts in wound healing: force generation and measurement". In: *Journal of tissue viability* 20.4 (2011), pp. 108–120.

[89]   Jessica H Wen et al. "Interplay of matrix stiffness and protein tethering in stem cell differentiation". In: *Nature materials* 13.10 (2014), p. 979.

[90]   Jeffrey G Jacot et al. "Cardiac myocyte force development during differentiation and maturation". In: *Annals of the New York Academy of Sciences* 1188 (2010), p. 121.

[91]   Matthias Chiquet. "Regulation of extracellular matrix gene expression by mechanical stress". In: *Matrix biology* 18.5 (1999), pp. 417–426.

[92]   D Karamichos, RA Brown, and V Mudera. "Collagen stiffness regulates cellular contraction and matrix remodeling gene expression". In: *Journal of Biomedical Materials Research Part A: An Official Journal of The Society for Biomaterials, The Japanese Society for Biomaterials, and The Australian Society for Biomaterials and the Korean Society for Biomaterials* 83.3 (2007), pp. 887–894.

[93]   Matteo Rauzi et al. "Embryo-scale tissue mechanics during Drosophila gastrulation movements". In: *Nature communications* 6 (2015), p. 8677.

[94]   Janna K Mouw et al. "Tissue mechanics modulate microRNA-dependent PTEN expression to regulate malignant progression". In: *Nature medicine* 20.4 (2014), p. 360.

[95]   Mustafa Unal et al. "Micro and nano-scale technologies for cell mechanics". In: *Nanobiomedicine* 1.Godište 2014 (2014), pp. 1–5.

[96]   William J Polacheck and Christopher S Chen. "Measuring cell-generated forces: a guide to the available tools". In: *Nature methods* 13.5 (2016), p. 415.

[97]   Phillip A Harries, James E Schoelz, and Richard S Nelson. "Intracellular transport of viruses and their components: utilizing the cytoskeleton and membrane highways". In: *Molecular plant-microbe interactions* 23.11 (2010), pp. 1381–1393.

[98]   Tatyana Svitkina. "The actin cytoskeleton and actin-based motility". In: *Cold Spring Harbor Perspectives in Biology* 10.1 (2018), a018267.

[99]   Florian Huber. "Emergent structure formation of the actin cytoskeleton". PhD thesis. 2011.

[100] Donald E Ingber. "Tensegrity I. Cell structure and hierarchical systems biology". In: *Journal of cell science* 116.7 (2003), pp. 1157–1173.

[101]   Jérôme Roger Dennis Soiné. "Reconstruction and Simulation of Cellular Traction Forces". PhD thesis. 2014.

[102]   Gary Joseph Doherty and Harvey T McMahon. "Mediation, modulation, and consequences of membrane-cytoskeleton interactions." In: *Annual review of biophysics* 37 (2008), pp. 65–95.

[103]   Revathi Ananthakrishnan and Allen Ehrlicher. "The forces behind cell movement". In: *International journal of biological sciences* 3.5 (2007), p. 303.

[104]   Alexander Mogilner and George Oster. "Cell motility driven by actin polymerization". In: *Biophysical journal* 71.6 (1996), pp. 3030–3045.

[105]   Alex Mogilner and George Oster. "Force generation by actin polymerization II: the elastic ratchet and tethered filaments". In: *Biophysical journal* 84.3 (2003), pp. 1591–1605.

[106]   AE Carlsson. "Growth velocities of branched actin networks". In: *Biophysical journal* 84.5 (2003), pp. 2907–2918.

[107]   Anders E Carlsson. "Growth of branched actin networks against obstacles". In: *Biophysical journal* 81.4 (2001), pp. 1907–1923.

[108]   Manfred Schliwa and Guenther Woehlke. "Molecular motors". In: *Nature* 422 (2003), pp. 759–765.

[109]   Zhiqi Sun, Shengzhen S Guo, and Reinhard Fässler. "Integrin-mediated mechanotransduction". In: *J Cell Biol* 215.4 (2016), pp. 445–456.

[110]   David Garrod and Martyn Chidgey. "Desmosome structure, composition and function". In: *Biochimica et Biophysica Acta (BBA)-Biomembranes* 1778.3 (2008), pp. 572–587.

[111]   Haguy Wolfenson et al. "Actomyosin-generated tension controls the molecular kinetics of focal adhesions". In: *J Cell Sci* 124.9 (2011), pp. 1425–1432.

[112]   Karen A Beningo et al. "Nascent focal adhesions are responsible for the generation of strong propulsive forces in migrating fibroblasts". In: *The Journal of cell biology* 153.4 (2001), pp. 881–888.

[113]   Ben Short. "Stress fibers guide focal adhesions to maturity". In: *The Journal of Cell Biology* 196.3 (2012), p. 301.

[114]   Sophie Sluysmans et al. "The role of apical cell–cell junctions and associated cytoskeleton in mechanotransduction". In: *Biology of the Cell* 109.4 (2017), pp. 139–161.

[115]   Miguel Vicente-Manzanares and Alan Rick Horwitz. "Adhesion dynamics at a glance". In: *Journal of cell science* 124.23 (2011), pp. 3923–3927.

[116]   Bruce Alberts et al. "Molecular Biology of the Cell 4th edn (New York: Garland Science)". In: *Ann Bot* 91 (2002), p. 401.

[117]   John L Tan et al. "Cells lying on a bed of microneedles: an approach to isolate mechanical force". In: *Proceedings of the National Academy of Sciences* 100.4 (2003), pp. 1484–1489.

[118]   Huw Colin-York, Christian Eggeling, and Marco Fritzsche. "Dissection of mechanical force in living cells by super-resolved traction force microscopy". In: *Nature protocols* 12.4 (2017), p. 783.

[119]   Agustina Setiawati et al. "Future research directions in the design of versatile extracellular matrix in tissue engineering". In: *International neurourology journal* 22.Suppl 2 (2018), S66.

[120] J Matthew Barnes, Laralynne Przybyla, and Valerie M Weaver. "Tissue mechanics regulate brain development, homeostasis and disease". In: *J Cell Sci* 130.1 (2017), pp. 71–82.

[121] CT Lim et al. "Experimental techniques for single cell and single molecule biomechanics". In: *Materials Science and Engineering: C* 26.8 (2006), pp. 1278–1288.

[122] Armen R Kherlopian et al. "A review of imaging techniques for systems biology". In: *BMC systems biology* 2.1 (2008), p. 74.

[123] Subra Suresh. "Biomechanics and biophysics of cancer cells". In: *Acta Materialia* 55.12 (2007), pp. 3989–4014.

[124] Gang Bao and Subra Suresh. "Cell and molecular mechanics of biological materials". In: *Nature materials* 2.11 (2003), p. 715.

[125] Hendrik Milting et al. "The TMEM43 Newfoundland mutation p. S358L causing ARVC-5 was imported from Europe and increases the stiffness of the cell nucleus". In: *European heart journal* 36.14 (2014), pp. 872–881.

[126] Francesca Di Modugno et al. "3D models in the new era of immune oncology: focus on T cells, CAF and ECM". In: *Journal of Experimental & Clinical Cancer Research* 38.1 (2019), p. 117.

[127] Kunio Takeyasu. *Atomic force microscopy in nanobiology*. CRC Press, 2014.

[128] Robert M Hochmuth. "Micropipette aspiration of living cells". In: *Journal of biomechanics* 33.1 (2000), pp. 15–22.

[129] Ming Dao, Chwee Teck Lim, and Subra Suresh. "Mechanics of the human red blood cell deformed by optical tweezers". In: *Journal of the Mechanics and Physics of Solids* 51.11-12 (2003), pp. 2259–2280.

[130] Hu Zhang and Kuo-Kang Liu. "Optical tweezers for single cells". In: *Journal of The Royal Society Interface* 5.24 (2008), pp. 671–690.

[131] Ben Fabry et al. "Implications of heterogeneous bead behavior on cell mechanical properties measured with magnetic twisting cytometry". In: *Journal of magnetism and magnetic materials* 194.1-3 (1999), pp. 120–125.

[132] Martial Balland et al. "Power laws in microrheology experiments on living cells: Comparative analysis and modeling". In: *Physical Review E* 74.2 (2006), p. 021911.

[133] Brandon L Blakely et al. "A DNA-based molecular probe for optically reporting cellular traction forces". In: *Nature methods* 11.12 (2014), p. 1229.

[134] Erkki Ruoslahti, Michael D Pierschbacher, et al. "Arg-Gly-Asp: a versatile cell recognition signal". In: *Cell* 44.4 (1986), pp. 517–518.

[135] Kiyoko Yoshioka et al. "Small GTP-binding protein Rho stimulates the actomyosin system, leading to invasion of tumor cells". In: *Journal of Biological Chemistry* 273.9 (1998), pp. 5146–5154.

[136] Sergey V Plotnikov et al. "High-resolution traction force microscopy". In: *Methods in cell biology*. Vol. 123. Elsevier, 2014, pp. 367–394.

[137] Albert K Harris, Patricia Wild, and David Stopak. "Silicone rubber substrata: a new wrinkle in the study of cell locomotion". In: *Science* 208.4440 (1980), pp. 177–179.

[138] Partha Roy et al. "Microscope-based techniques to study cell adhesion and migration". In: *Nature cell biology* 4.4 (2002), E91.

[139] Janina R Lange and Ben Fabry. "Cell and tissue mechanics in cell migration". In: *Experimental cell research* 319.16 (2013), pp. 2418–2423.

[140]   Robert W Style et al. "Traction force microscopy in physics and biology". In: *Soft matter* 10.23 (2014), pp. 4047–4055.

[141]   Ulrich S Schwarz and Jérôme RD Soiné. "Traction force microscopy on soft elastic substrates: A guide to recent computational advances". In: *Biochimica et Biophysica Acta (BBA)-Molecular Cell Research* 1853.11 (2015), pp. 3095–3104.

[142]   Pere Roca-Cusachs, Vito Conte, and Xavier Trepat. "Quantifying forces in cell biology". In: *Nature cell biology* 19.7 (2017), pp. 742–751.

[143]   Benedikt Sabass et al. "Force generation by groups of migrating bacteria". In: *Proceedings of the National Academy of Sciences* 114.28 (2017), pp. 7266–7271.

[144]   Wesley R Legant et al. "Measurement of mechanical tractions exerted by cells in three-dimensional matrices". In: *Nature methods* 7.12 (2010), p. 969.

[145]   Nikolce Gjorevski et al. "Dynamic tensile forces drive collective cell migration through three-dimensional extracellular matrices". In: *Scientific reports* 5 (2015), p. 11458.

[146]   Huw Colin-York et al. "Super-resolved traction force microscopy (STFM)". In: *Nano letters* 16.4 (2016), pp. 2633–2638.

[147]   Stefan W Hell and Jan Wichmann. "Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy". In: *Optics letters* 19.11 (1994), pp. 780–782.

[148]   Zhaochun Yang et al. "Determining substrate displacement and cell traction fields—a new approach". In: *Journal of theoretical biology* 242.3 (2006), pp. 607–616.

[149]   Sung Sik Hur et al. "Live cells exert 3-dimensional traction forces on their substrata". In: *Cellular and molecular bioengineering* 2.3 (2009), pp. 425–436.

[150]   Xin Tang et al. "A novel cell traction force microscopy to study multi-cellular system". In: *PLoS computational biology* 10.6 (2014), e1003631.

[151]   Jérôme RD Soiné et al. "Measuring cellular traction forces on non-planar substrates". In: *Interface focus* 6.5 (2016), p. 20160024.

[152]   Ankur H Kulkarni et al. "Traction cytometry: regularization in the Fourier approach and comparisons with finite element method". In: *Soft matter* 14.23 (2018), pp. 4687–4695.

[153]   Micah Dembo et al. "Imaging the traction stresses exerted by locomoting cells with the elastic substratum method". In: *Biophysical journal* 70.4 (1996), pp. 2008–2022.

[154]   Micah Dembo and Yu-Li Wang. "Stresses at the cell-to-substrate interface during locomotion of fibroblasts". In: *Biophysical journal* 76.4 (1999), pp. 2307–2316.

[155]   James P Butler et al. "Traction fields, moments, and strain energy that cells exert on their surroundings". In: *American Journal of Physiology-Cell Physiology* 282.3 (2002), pp. C595–C605.

[156]   Christian Franck et al. "Three-dimensional traction force microscopy: a new tool for quantifying cell-matrix interactions". In: *PloS one* 6.3 (2011), e17833.

[157]   Ulrich S Schwarz et al. "Calculation of forces at focal adhesions from elastic substrate data: the effect of localized force and the need for regularization". In: *Biophysical journal* 83.3 (2002), pp. 1380–1394.

[158]   Sangyoon J Han et al. "Traction microscopy to identify force modulation in subresolution adhesions". In: *Nature methods* 12.7 (2015), p. 653.

[159]   Alejandro Suñé-Auñón et al. "Full L 1-regularized Traction Force Microscopy over whole cells". In: *BMC bioinformatics* 18.1 (2017), p. 365.

[160] Heinz H Bauschke et al. *Fixed-point algorithms for inverse problems in science and engineering*. Vol. 49. Springer Science & Business Media, 2011.

[161] Mark Schmidt, Nicolas L Roux, and Francis R Bach. "Convergence rates of inexact proximal-gradient methods for convex optimization". In: *Advances in neural information processing systems*. 2011, pp. 1458–1466.

[162] Sofia Mosci et al. "Solving structured sparsity regularization with proximal methods". In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2010, pp. 418–433.

[163] Gabriel Peyré, Sébastien Bougleux, and Laurent Cohen. "Non-local regularization of inverse problems". In: *European Conference on Computer Vision*. Springer. 2008, pp. 57–68.

[164] Jalal M Fadili and Gabriel Peyré. "Total variation projection with first order schemes". In: *IEEE Transactions on Image Processing* 20.3 (2010), pp. 657–669.

[165] Vincent Michel et al. "Total variation regularization for fMRI-based prediction of behavior". In: *IEEE transactions on medical imaging* 30.7 (2011), pp. 1328–1340.

[166] Gregory Hannum et al. "Genome-wide methylation profiles reveal quantitative views of human aging rates". In: *Molecular cell* 49.2 (2013), pp. 359–367.

[167] Artem Sokolov et al. "Pathway-based genomics prediction using generalized elastic net". In: *PLoS computational biology* 12.3 (2016), e1004790.

[168] Neal Parikh, Stephen Boyd, et al. "Proximal algorithms". In: *Foundations and Trends® in Optimization* 1.3 (2014), pp. 127–239.

[169] James P Crutchfield and Bruce S McNamara. "Equations of motion from a data series". In: *Complex systems* 1.417-452 (1987), p. 121.

[170] Michael Schmidt and Hod Lipson. "Distilling free-form natural laws from experimental data". In: *science* 324.5923 (2009), pp. 81–85.

[171] Josh Bongard and Hod Lipson. "Automated reverse engineering of nonlinear dynamical systems". In: *Proceedings of the National Academy of Sciences* 104.24 (2007), pp. 9943–9948.

[172] Sheng Zhang and Guang Lin. "Robust data-driven discovery of governing physical laws with error bars". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 474.2217 (2018), p. 20180305.

[173] Hannes Risken. "Fokker-planck equation". In: *The Fokker-Planck Equation*. Springer, 1996, pp. 63–95.

[174] Lorenzo Boninsegna, Feliks Nüske, and Cecilia Clementi. "Sparse learning of stochastic dynamical equations". In: *The Journal of chemical physics* 148.24 (2018), p. 241723.

[175] Christoph Honisch and Rudolf Friedrich. "Estimation of Kramers-Moyal coefficients at low sampling rates". In: *Physical Review E* 83.6 (2011), p. 066701.

[176] Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, 2011.

[177] Mongi A Abidi, Andrei V Gribok, and Joonki Paik. *Optimization Techniques in Computer Vision: Ill-posed problems and regularization*. Springer, 2016.

[178] Andrei Nikolaevich Tikhonov et al. *Numerical methods for the solution of ill-posed problems*. Vol. 328. Springer Science & Business Media, 2013.

[179] Emmanuel J Candès, Justin Romberg, and Terence Tao. "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information". In: *IEEE Transactions on information theory* 52.2 (2006), pp. 489–509.

[180] David L Donoho. "Compressed sensing". In: *IEEE Transactions on information theory* 52.4 (2006), pp. 1289–1306.

[181] Jonatan Bohr Brask et al. "Compressed sensing traction force microscopy". In: *Acta biomaterialia* 26 (2015), pp. 286–294.

[182] Alejandro Suñé-Auñón et al. "L1-regularized reconstruction for traction force microscopy". In: *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*. IEEE. 2016, pp. 140–144.

[183] Christian R Berger et al. "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing". In: *IEEE Transactions on Signal Processing* 58.3 (2009), pp. 1708–1721.

[184] John A Scales, Adam Gersztenkorn, and Sven Treitel. "Fast Ip solution of large, sparse, linear systems: Application to seismic travel time tomography". In: *Journal of Computational Physics* 75.2 (1988), pp. 314–333.

[185] Mário AT Figueiredo and Robert D Nowak. "An EM algorithm for wavelet-based image restoration". In: *IEEE Transactions on Image Processing* 12.8 (2003), pp. 906–916.

[186] Amir Beck and Marc Teboulle. "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems". In: *IEEE transactions on image processing* 18.11 (2009), pp. 2419–2434.

[187] Gabriel Peyré. "The numerical tours of signal processing". In: *Computing in Science & Engineering* 13.4 (2011), p. 94.

[188] David L Donoho and Jain M Johnstone. "Ideal spatial adaptation by wavelet shrinkage". In: *biometrika* 81.3 (1994), pp. 425–455.

[189] David L Donoho et al. "Wavelet shrinkage: asymptopia?" In: *Journal of the Royal Statistical Society: Series B (Methodological)* 57.2 (1995), pp. 301–337.

[190] Anupama Reddy et al. "Genetic and functional drivers of diffuse large B cell lymphoma". In: *Cell* 171.2 (2017), pp. 481–494.

[191] Max A Horlbeck et al. "Compact and highly active next-generation libraries for CRISPR-mediated gene repression and activation". In: *Elife* 5 (2016), e19760.

[192] Pin-Yu Chen et al. "Ead: elastic-net attacks to deep neural networks via adversarial examples". In: *Thirty-second AAAI conference on artificial intelligence*. 2018.

[193] Julien Mairal, Francis Bach, Jean Ponce, et al. "Sparse modeling for image and vision processing". In: *Foundations and Trends® in Computer Graphics and Vision* 8.2-3 (2014), pp. 85–283.

[194] Kun-Hsing Yu et al. "Predicting non-small cell lung cancer prognosis by fully automated microscopic pathology image features". In: *Nature communications* 7 (2016), p. 12474.

[195] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[196] Michael Grant and Stephen Boyd. *CVX: Matlab Software for Disciplined Convex Programming, version 2.1*. `http://cvxr.com/cvx`. Mar. 2014.

[197] Michael Grant and Stephen Boyd. "Graph implementations for nonsmooth convex programs". In: *Recent Advances in Learning and Control*. Ed. by V. Blondel, S. Boyd, and H. Kimura. Lecture Notes in Control and Information Sciences. `http://stanford.edu/~boyd/graph_dcp.html`. Springer-Verlag Limited, 2008, pp. 95–110.

[198] Per Christian Hansen, Toke Koldborg Jensen, and Giuseppe Rodriguez. "An adaptive pruning algorithm for the discrete L-curve criterion". In: *Journal of computational and applied mathematics* 198.2 (2007), pp. 483–492.

[199] Per Christian Hansen. *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion*. SIAM, 1998.

[200] Yunfei Huang et al. "Traction force microscopy with optimized regularization and automated Bayesian parameter selection for comparing cells". In: *Scientific reports* 9.1 (2019), p. 539.

[201] JO Ramsay, Giles Hooker, and Spencer Graves. "Smoothing: computing curves from noisy data". In: *Functional Data Analysis with R and MATLAB*. Springer, 2009, pp. 59–82.

[202] Wenjiang J Fu. "Penalized regressions: the bridge versus the lasso". In: *Journal of computational and graphical statistics* 7.3 (1998), pp. 397–416.

[203] Martin Hanke and Toomas Raus. "A general heuristic for choosing the regularization parameter in ill-posed problems". In: *SIAM Journal on Scientific Computing* 17.4 (1996), pp. 956–972.

[204] Aleksandr Sergeevich Leonov. "On the choice of regularization parameters by means of the quasi-optimality and ratio criteria". In: *Doklady Akademii Nauk*. Vol. 240. 1. Russian Academy of Sciences. 1978, pp. 18–20.

[205] PerChristian Hansen and P Johnston. *Computational inverse problems in electrocardiology*. 2001.

[206] Martin Hanke. "Limitations of the L-curve method in ill-posed problems". In: *BIT Numerical Mathematics* 36.2 (1996), pp. 287–301.

[207] Colin G Farquharson and Douglas W Oldenburg. "A comparison of automatic techniques for estimating the regularization parameter in non-linear inverse problems". In: *Geophysical Journal International* 156.3 (2004), pp. 411–425.

[208] Souleymane Samagassi et al. "Bayesian sparse regularization for multiple force identification and location in time domain". In: *Inverse Problems in Science and Engineering* 27.9 (2019), pp. 1221–1262.

[209] Jari Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*. Vol. 160. Springer Science & Business Media, 2006.

[210] Abhik Ghosh et al. "A Bayesian analysis of redshifted 21-cm H i signal and foregrounds: simulations for LOFAR". In: *Monthly Notices of the Royal Astronomical Society* 452.2 (2015), pp. 1587–1600.

[211] Rafael Molina, Aggelos K Katsaggelos, and Javier Mateos. "Bayesian and regularization methods for hyperparameter estimation in image restoration". In: *IEEE Transactions on Image Processing* 8.2 (1999), pp. 231–246.

[212] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. "A training algorithm for optimal margin classifiers". In: *Proceedings of the fifth annual workshop on Computational learning theory*. 1992, pp. 144–152.

[213] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: *Machine learning* 20.3 (1995), pp. 273–297.

[214] Vladimir Vapnik et al. "Statistical learning theory". In: (1998).

[215] S Derin Babacan, Rafael Molina, and Aggelos K Katsaggelos. "Bayesian compressive sensing using Laplace priors". In: *IEEE Transactions on Image Processing* 19.1 (2009), pp. 53–63.

[216] Lev Davidovich Landau et al. "Theory of elasticity". In: (1986).

[217] Iva Marija Tolic-Nørrelykke et al. "Spatial and temporal traction response in human airway smooth muscle cells". In: *American Journal of Physiology-Cell Physiology* 283.4 (2002), pp. C1254–C1266.

[218] Claude N Holenstein, Unai Silvan, and Jess G Snedeker. "High-resolution traction force microscopy on small focal adhesions-improved accuracy through optimal marker distribution and optical flow tracking". In: *Scientific reports* 7 (2017), p. 41633.

[219] Christoph Schell et al. "N-wasp is required for stabilization of podocyte foot processes". In: *Journal of the American Society of Nephrology* 24.5 (2013), pp. 713–721.

[220] Christoph Schell et al. "The FERM protein EPB41L5 regulates actomyosin contractility and focal adhesion formation to maintain the kidney filtration barrier". In: *Proceedings of the National Academy of Sciences* 114.23 (2017), E4621–E4630.

[221] Nils Hersch et al. "The constant beat: cardiomyocytes adapt their forces by equal contraction upon environmental stiffening". In: *Biology open* 2.3 (2013), pp. 351–361.

[222] Claudia M Cesa et al. "Micropatterned silicone elastomer substrates for high resolution analysis of cellular force patterns". In: *Review of scientific instruments* 78.3 (2007), p. 034301.

[223] Rudolf Merkel et al. "Cell force microscopy on elastic layers of finite thickness". In: *Biophysical journal* 93.9 (2007), pp. 3314–3323.

[224] David W Winters, Barry D Van Veen, and Susan C Hagness. "A sparsity regularization approach to the electromagnetic inverse scattering problem". In: *IEEE transactions on antennas and propagation* 58.1 (2009), pp. 145–154.

[225] Gene H Golub, Michael Heath, and Grace Wahba. "Generalized cross-validation as a method for choosing a good ridge parameter". In: *Technometrics* 21.2 (1979), pp. 215–223.

[226] James P Hobert and George Casella. "The effect of improper priors on Gibbs sampling in hierarchical linear mixed models". In: *Journal of the American Statistical Association* 91.436 (1996), pp. 1461–1473.

[227] Yong Huang et al. "Robust Bayesian compressive sensing for signals in structural health monitoring". In: *Computer-Aided Civil and Infrastructure Engineering* 29.3 (2014), pp. 160–179.

[228] Vincent Nier et al. "Inference of internal stress in a cell monolayer". In: *Biophysical journal* 110.7 (2016), pp. 1625–1635.

[229] Sebastian Houben, Norbert Kirchgeßner, and Rudolf Merkel. "Estimating force fields of living cells–comparison of several regularization schemes combined with automatic parameter choice". In: *Joint Pattern Recognition Symposium*. Springer. 2010, pp. 71–80.

[230] Jean-Louis Martiel et al. "Measurement of cell traction forces with ImageJ". In: *Methods in cell biology*. Vol. 125. Elsevier, 2015, pp. 269–287.

[231] *PIVlab Update 2.31*. https://pivlab.blogspot.com/. 2019.

[232] *Mpiv*. http://www.oceanwave.jp/softwares/mpiv/. 2003.

[233] *JPIV*. http://www.jpiv.vennemann-online.de/.

[234] *Easy-to-use TFM Software*. https://github.com/CellMicroMechanics. 2019.

[235] J Náprstek and R Král. "Multi-dimensional Fokker-Planck equation analysis using the modified finite element method". In: *Journal of Physics: Conference Series*. Vol. 744. 2016, p. 012177.

[236] Edward N Lorenz. "Deterministic nonperiodic flow". In: *Journal of the atmospheric sciences* 20.2 (1963), pp. 130–141.

[237] Olivier Darrigol. *Worlds of flow: A history of hydrodynamics from the Bernoullis to Prandtl*. Oxford University Press, 2005.

[238] Jared L Callaham, Kazuki Maeda, and Steven L Brunton. "Robust flow reconstruction from limited measurements via sparse representation". In: *Physical Review Fluids* 4.10 (2019), p. 103907.

[239] Kunihiko Taira and Tim Colonius. "The immersed boundary method: a projection approach". In: *Journal of Computational Physics* 225.2 (2007), pp. 2118–2137.

[240] Tim Colonius and Kunihiko Taira. "A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions". In: *Computer Methods in Applied Mechanics and Engineering* 197.25-28 (2008), pp. 2131–2146.

[241] Alan Mathison Turing. "The chemical basis of morphogenesis". In: *Bulletin of mathematical biology* 52.1-2 (1990), pp. 153–197.

[242] JD Murray. *Mathematical biology II: spatial models and biomedical applications*. Vol. 3. Springer-Verlag, 2001.

[243] Marcus R Garvie, Philip K Maini, and Catalin Trenchea. "An efficient and robust numerical algorithm for estimating parameters in Turing systems". In: *Journal of Computational Physics* 229.19 (2010), pp. 7058–7071.

[244] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. "V-net: Fully convolutional neural networks for volumetric medical image segmentation". In: *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE. 2016, pp. 565–571.

[245] Jiexiong Tang et al. "Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine". In: *IEEE Transactions on Geoscience and Remote Sensing* 53.3 (2014), pp. 1174–1185.

[246] Quanzeng You et al. "Robust image sentiment analysis using progressively trained and domain transferred deep networks". In: *Twenty-ninth AAAI conference on artificial intelligence*. 2015.

[247] Yuchi Tian et al. "Deeptest: Automated testing of deep-neural-network-driven autonomous cars". In: *Proceedings of the 40th international conference on software engineering*. ACM. 2018, pp. 303–314.

[248] Tianhao Zhang et al. "Learning deep control policies for autonomous aerial vehicles with mpc-guided policy search". In: *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE. 2016, pp. 528–535.

[249] Bichen Wu et al. "Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2017, pp. 129–137.

[250] Andre Esteva et al. "Dermatologist-level classification of skin cancer with deep neural networks". In: *Nature* 542.7639 (2017), p. 115.

[251] Geert Litjens et al. "A survey on deep learning in medical image analysis". In: *Medical image analysis* 42 (2017), pp. 60–88.

[252] Travers Ching et al. "Opportunities and obstacles for deep learning in biology and medicine". In: *Journal of The Royal Society Interface* 15.141 (2018), p. 20170387.

[253] Claude E Shannon. "A mathematical theory of communication". In: *Bell system technical journal* 27.3 (1948), pp. 379–423.

[254]   Derek W Robinson. "Entropy and uncertainty". In: *Entropy* 10.4 (2008), pp. 493–506.

[255]   Hugo Touchette and Seth Lloyd. "Information-theoretic approach to the study of control systems". In: *Physica A: Statistical Mechanics and its Applications* 331.1-2 (2004), pp. 140–172.

[256]   Hugo Touchette and Seth Lloyd. "Information-theoretic limits of control". In: *Physical review letters* 84.6 (2000), p. 1156.

[257]   Malek Ben Salem et al. "Universal prediction distribution for surrogate models". In: *SIAM/ASA Journal on Uncertainty Quantification* 5.1 (2017), pp. 1086–1109.

[258]   Glenn M Torrie and John P Valleau. "Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling". In: *Journal of Computational Physics* 23.2 (1977), pp. 187–199.

[259]   George Eason, Benjamin Noble, and IN Sneddon. "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions". In: *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 247.935 (1955), pp. 529–551.

[260]   DL De Jong. "Computer program BISAR, Layered systems under normal and tangential loads". In: *External Report, Koninkljke/Shell-Laboratorium* (1979).

Eigenhändigkeitserklärung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Dissertation selbstständig und ohne die Benutzung anderer als der angegebenen Hilfsmittel und Literatur angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichen und nicht veröffentlichten Werken dem Wortlaut oder dem Sinn nach entnommen wurden, sind als solche kenntlich gemacht. Ich versichere an Eides statt, dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie - abgesehen von unten angegebenen Teilpublikationen und eingebundenen Artikeln und Manuskripten - noch nicht veröffentlicht worden ist sowie, dass ich eine Veröffentlichung der Dissertation vor Abschluss der Promotion nicht ohne Genehmigung des Promotionsausschusses vornehmen werde. Die Bestimmungen dieser Ordnung sind mir bekannt. Darüber hinaus erkläre ich hiermit, dass ich die Ordnung zur Sicherung guter wissenschaftlicher Praxis und zum Umgang mit wissenschaftlichem Fehlverhalten der Universität zu Köln gelesen und sie bei der Durchführung der Dissertation zugrundeliegenden Arbeiten und der schriftlich verfassten Dissertation beachtet habe und verpflichte mich hiermit, die dort genannten Vorgaben bei allen wissenschaftlichen Tätigkeiten zu beachten und umzusetzen. Ich versichere, dass die eingereichte elektronische Fassung der eingereichten Druckfassung vollständig entspricht.

(Ort, Datum)	(Unterschrift)

Teilpublikation

• Yunfei Huang, Gerhard Gompper and Benedikt Sabass. *"A Bayesian traction force microscopy method with automated denoising in a user-friendly software package"*, Computer Physics Communications, 2020, 256, 107313.

• Yunfei Huang, Christoph Schell, Tobias B. Huber, Ahmet Nihat Şimşek, Nils Hersch, Rudolf Merkel, Gerhard Gompper and Benedikt Sabass. *"Traction force microscopy with optimized regularization and automated Bayesian parameter selection for comparing cells"*, Scientific Reports, 2019, 9, 1-16.

• Ahmet Nihat Simsek, Andrea Braeutigam, Matthias D. Koch, Joshua W. Shaevitz, Yunfei Huang, Gerhard Gompper and Benedikt Sabass. *"Substrate-rigidity dependent migration of an idealized twitching bacterium"*, Soft Matter, 2019,15, 6224–6236.