Thèse n° 8253

Sampling Geometry and Colour

Présentée le 29 octobre 2021

Faculté informatique et communications
Laboratoire de communications audiovisuelles
Programme doctoral en informatique et communications

pour l’obtention du grade de Docteur ès Sciences

par

Michalina Wanda PACHOLSKA

Acceptée sur proposition du jury

Prof. M. C. Gastpar, président du jury
Prof. M. Vetterli, Dr A. J. Scholefield, directeurs de thèse
Prof. S. Mallat, rapporteur
Prof. A. Hansen, rapporteur
Prof. M. Unser, rapporteur

École
polytechnique
fédérale
de Lausanne

2021

Acknowledgements

Creating a thesis is a collaborative endeavour. First, behind every PhD thesis stands a thesis director. I am grateful to my thesis directors, Martin Vetterli and Adam Scholefield for accepting me as their student and for their unwavering support. I have greatly benefited from the collaborative environment they created; many people contributed to creation of my thesis and to my personal growth. Here, I describe my journey as a PhD student, through the lens of the help I obtained on the way.

The goal of a graduate student is to be accepted by the research community as a peer. I would like to thank my thesis jury members Anders Hansen, Stephane Mallat, Michaël Unser, and Michael Gastpar for taking the time to read my thesis, for the interesting discussions, and for accepting me in the research community. I really appreciate the time and thought Anders put into his research suggestions for me.[1]

Before graduation, a PhD student has to write their thesis. I think everyone in LCAV worked on this manuscript at some point: Adrian Jarret and Karen Adam translated the abstract to French. Adam, Karen, Frederike Dümbgen, and Eric Bezzam read different parts of it and provided valuable feedback. I am especially grateful to Adam for finding all the missing articles despite his busy schedule, to Karen for always being ready to procrastinate her own research to help others, and to Frederike for finding time to read my chapters while writing her own. Comments and corrections provided by Holly Cogliati were invaluable. I have learned a lot from her and gained confidence in my ability to write.

Even a flawlessly written thesis is worth only as much as the research it describes; and collaborating is the best way to do research. Adam was my co-supervisor, but he also worked closely with me on all my projects[2]. I thank him for always keeping his calm when I panicked, for getting me unstuck numerous times, and for never pushing his opinion over people junior to him. Adam suggested working on the trajectory recovery that started the domino of interesting questions that I answer in this thesis.

I was introduced to Adam by Benjamin Bejar; they both advised my master's thesis. Together with Golnoosh Elhami, they worked with me on my first PhD project. I appreciate the patience that all three kept when showing me the ropes. This project would not be complete without the help of Damian Orlef; his observations were essential for me to conclude the proofs.

I was still a novice when I joined the Lippmann project. This was a great opportunity to learn PhD-ing from Gilles Baechler, physics from Arnaud Latty, and photography from Filipe Alves. Thanks to Martin, the project evolved into a collaboration first with Yves Belluared, and later with Julien Gateau and Ruben Ricca, all from GALATEA laboratory

---

[1] Now I have research ideas for years to come.

[2] I will not be thanking him in every paragraph.

at EPFL. I thank them for all the discussions and their patience in explaining to me the optical properties of glass. Moreover, I would like to thank all members of GALATEA laboratory for being very welcoming and for entrusting me with access to their expensive lasers.

During my PhD years, I collaborated most closely Frederike, and not only because we shared an office. We worked on the trajectory project suggested by Adam. We brought different skills and different work styles to the table, yet we shared more or less every task. It was fun and an instructive experience: I learned good git practices and that it is never to late too rewrite half of a paper.

The continuation of this project led to my unusual collaboration with Karen: I helped her with maths once, and in exchange she read a flood of my confusing poofs. I truly appreciate that Karen accepted this clearly unfair deal.

For the students who worked on the semester projects with me, Yacine Hedef, Alix Jeannerot and Shuo Wen, I thank them for choosing me for their mentor and for their hard work. Special thanks go to Shuo for working with me for two semesters and to Krzystof (Kris) Lis for co-advising Shuo's project.

Finally, I would like to thank the whole Origami team at DeepMind for being welcoming and helpful, despite getting an intern at a very... dynamic time. It was a fun ride and an honour to be part of Origami, even if for a short while. I especially thank John Jumper for guiding me through the complex protein landscape and Augustin Žídek for his engineering and emotional support.

To create valuable results, a student needs guidance and research directions from mentors. The first among my mentors is, of course, Martin. I am grateful for all his advice; he always considered my interest first before his. I appreciate his patience for my negative results, even those contradicting his ideas. Martin always believed in me, and always took my complaints seriously[3]. I also thank Adam for all his time. Throughout my PhD time, Adam was my go-to person with any silly research or academia related problems.

Not only supervisors can be mentors. I would like to thank Benjamin, Mihailo Kolundzija and Miranda Kreković for introducing me to LCAV. Miranda gave me a lot of advice and support during the first year of my studies. I am grateful for having had the opportunity to work on an computer graphics project with Wenzel Jakob, before I joined LCAV. During this time, I learned a lot and made great friends. Furthermore, I would like to thank Victor Kuncak and Bruno Correia for conversations about research and for their advice about finishing PhD thesis.

Finally, my parents were also my mentors, and they supported me through the enitre PhD experience. My father was always there to help with academia (or cooking) related problems, and my mother always had my newest publication hanging on the fridge door, with a cake recipe ready.[4]

---

[3] Sometimes even more than I did.

[4] It is a common misconception that coffee is necessary to finish a PhD thesis. Coffee is not necessary; cakes are.

Not all work that goes into a PhD thesis is research oriented. LCAV would not exist without Nicoletta who, among other things, fought tirelessly to ensure everyone has an opportunity to talk with Martin. Adrien Hoffet helped us immensely with the the Lippmann video and found a portable video-conference set for everyone to use during the pandemic. Paolo made sure we did not talk to much about research, during LCAV coffee breaks.

If there is one thing that is necessary for the survival of a PhD student, it is peer support. I received it from many people in and outside of LCAV. Dalia El Badawy and Hanjie Pan shared lunches with me and would go to the other end of the campus in search for better food. I would also like to thank Sepand Kashani for being the cookie guardian, and for always[5] willing to help. It was also very helpful to exchange tips and tricks with Frederike.

Among peers, the most valuable for me were the "complaining lunches" with Tizian Zeltner. Our conversations got me through some tough times. I appreciate the time Tizian spent listening to my grumbling; he was always understanding and never judgmental. I also very much enjoyed our conversations about literature and visual media.

Peer support also comes in the form of common projects. I am grateful to Tizian and Kris for attending Creative Wednesdays throughout the pandemic; this made it much easier to distiguish the days of the week. I would also like to thank PolyProg for giving me an opportunity to design T-shirts, and for all the fun and free food during the PolyProg events. I especially appreciate the effort Tatiana Porté made to keep us connected during the pandemic. I also appreciate the walks with her and Solal Pirelli.

Of course, the emotional support from Jakub Sygnowski was invaluable for me. I would like to thank him for hiding in Switzerland with me, for *actually* reading my proofs, for organizing my internship, and all the fun projects I would not have had the courage to do without him.

Last but not least, I would like to thank my friends from Wrocław and Warsaw, and my sister Ewa Dudziak and brother Jan Pacholski and their families. I thank all of them for not giving up on me, even though I did not have much time for them, and for being there for me whenever I visited. I would like to especially thank Aleksandra Piotrowska for introducing me to the world of oatmeal, Barbara Latacz for the hikes and discussions about being a woman in science, and Michał Grochowski for inefficiently sabotaging my thesis writing.

---

[5] Always, except when he was held captive by Swiss government; I will leave this story for him to tell.

Abstract

In this thesis, we take a signal-processing approach to two research areas outside of the core of the signal processing research: geometry reconstruction and light propagation through non-uniform media.

In the first area, we consider new sampling schemes inspired by problems of surface recovery and continuous localisation. We analyse sampling at unknown locations and show signal recovery uniqueness under constraints. We also study non-synchronised sampling of vector-valued signals or fields, that arise in the problem of localisation of a smoothly moving device. We provide sufficient conditions for device localisation. To this end, we formulate the problem of rank-one sensing of full-rank matrices and develop sufficient conditions for matrix recovery when some measurements are missing.

In the second area, we study Lippmann photography, a century-old colour photography technique. We model recording and viewing of a Lippmann photograph as analysis and synthesis operators and study the invertibility of their composition. We extend our analysis to a potential method of creating digital photographs, namely internal modification of glass with femto-second lasers.

Keywords — sampling, sampling at unknown locations, SLAMpling, matrix recovery, matrix completion, rank-one sensing, product of frames, polynomials, bandlimited signals, linearization, linear system of equations, localization, positioning, range sensing, optimization, trajectory estimation, Lippmann, photography, interference, spectroscopy, silica, femto-second lasers.

Résumé

Dans cette thèse, nous adoptons le point de vue du traitement du signal pour aborder deux domaines de recherche qui lui sont tangentiels: la reconstruction de la géométrie et la propagation de la lumière à travers des milieux non-uniformes.

D'un côté, nous considérons de nouveaux schémas d'échantillonnage, inspirés par les problèmes de la récupération des surfaces et de la localisation continue. Nous analysons l'échantillonnage de signaux à des positions inconnues et montrons l'unicité de la reconstruction de ces signaux sous certaines conditions. Nous étudions aussi l'échantillonnage non-synchronisé de signaux ou de champs à valeurs vectorielles, situation qui se rencontre, en l'occurrence, lors de la localisation d'un véhicule autonome se déplaçant selon une trajectoire fluide. Dans ce contexte, nous mettons en avant des conditions suffisantes pour la localisation des véhicules. Pour cela, nous formulons le problème d'échantillonnage de rang-un sur des matrices de rang complet, et développons des conditions suffisantes pour la reconstruction de ces matrices lorsque l'échantillonnage est incomplet.

D'un autre côté, nous étudions la photographie interférentielle de Lippmann, une technique centenaire de photographie en couleur. Nous interprétons l'enregistrement et le visionnement d'une photographie Lippmann comme des opérations abstraites de synthèse et d'analyse, et étudions l'inversibilité de leur composition. Nous étendons notre analyse à une nouvelle méthode qui pourrait permettre de créer des photographies digitales, grâce à la modification interne du verre avec des lasers femtoseconde.

Mots-Clés — échantillonnage, échantillonnage de signaux à des positions inconnues, SLAMpling, reconstruction de matrice, complétion de matrice, échantillonnage de rang-un, produit de repères, polynômes, signaux à bande limitée, linéarisation, système d'équations linéaires, localisation, échantillonnage de distance, optimisation, estimation de trajectoire, Lippmann, photographie, interférence, spectroscopie, silice, laser femtoseconde.

How to read this thesis?

This thesis consists of two parts, and therefore we take the liberty to write this short note to help the reader navigate them.

The first part, titled Sampling & Geometry core of this thesis. The second part, titled Lippmann Photography describes our contribution to a larger interdisciplinary project. Parts I and II are independent and can be read in any order. The references to Part I that appear in Part II are there to give a broader context and the concepts they refer to are not required to understand the Lippmann photography.

This thesis is on the longer side of theses, not only because it contains two parts, but also because we took the opportunity to share information that is usually not included in publications. Therefore, this thesis includes many examples, side notes and explanations of the creative process that led to the results presented.

While we believe that fellow early career researchers can find this additional material useful, we are aware that a long thesis can be tedious to read. Therefore, we structured the thesis to allow the reader to cherry-pick which parts are interesting for them. For the readers who are not interested in the proofs, the proofs are conveniently marked. The reader who is not interested in our journey might only glance at chapters titled “Intuitions and Challenges”. Finally, the boxes titled “Want to know more?” are there for the most curious readers.

List of Symbols

i	the imaginary unit
s	scalars (lowercase letters)
C	scalar constants (uppercase letters)
$\boldsymbol{v}$	vectors (bold lowercase letters)
$\boldsymbol{M}$	matrices and linear operators (bold uppercase letters)
$\mathcal{S}$	sets (calligraphy letters) and spaces
$m_{k,j}$	the element in k-th row and j-th column of $\boldsymbol{M}$
$\boldsymbol{m}_j$	the j-th column of $\boldsymbol{M}$
$[N]$	set of integers up to N: $[N] = \mathbb{N} \cap [1, N]$
$\odot$	point-wise (Hadamard) product
$\otimes$	tensor product
$\otimes_K$	Kronecker product
$\langle \cdot, \cdot \rangle$	An inner product between
$\boldsymbol{M} \succ \boldsymbol{N}$	$\boldsymbol{M} - \boldsymbol{N}$ is positive definite
$\boldsymbol{M} \succcurlyeq \boldsymbol{N}$	$\boldsymbol{M} - \boldsymbol{N}$ is positive semidefinite
$\boldsymbol{M}^\top$, $\boldsymbol{v}^\top$	transpose of matrix $\boldsymbol{M}$ or vector $\boldsymbol{v}$
$\overline{\boldsymbol{M}}$, $\overline{\boldsymbol{v}}$, $\overline{s}$	complex conjugate of the *entries* of matrix $\boldsymbol{M}$ or vector $\boldsymbol{v}$ or scalar $\overline{s}$
$\boldsymbol{M}^\dagger$, $\boldsymbol{v}^\dagger$	Hermitian transpose of matrix $\boldsymbol{M}$ or vector $\boldsymbol{v}$, $\boldsymbol{M}^\dagger = \overline{\boldsymbol{M}}^\top$)
$\tilde{\boldsymbol{M}}$, $\tilde{\boldsymbol{v}}$, $\tilde{s}$	An estimator of the symbol without $\tilde{}$
$\hat{s}(\omega), \hat{s}(\boldsymbol{\xi})$	A Fourier transform of, respectively, signal $s(t)$ and $s(\boldsymbol{x})$
$\mathbf{1}_N$	a length-N vector of ones
$\boldsymbol{I}_N$ or $\boldsymbol{I}$	an $N \times N$ identity matrix; size not given if clear from context
$\mathbb{1}_{\mathcal{S}}$	indicator function of a set $\mathcal{S}$
almost all $x \in \mathcal{X}$	all $x \in \mathcal{X}$ except of a measure-zero subset;

Contents

Acknowledgements iii

Abstract vi

How to read this thesis? viii

List of Symbols ix

I. Sampling & Geometry 1

1. Introduction 2
- 1.1. Sampling and Interpolation . . . 3
- 1.2. Geometry Reconstruction . . . 9
- 1.3. Matrix Recovery . . . 15
- 1.4. Geometry from the Sampling Perspective . . . 16
- 1.5. Mathematical Tools . . . 18

2. Sampling at Unknown Locations 23
- 2.1. Theory . . . 24
  - 2.1.1. Intuitions and Challenges . . . 25
  - 2.1.2. Theorems and Proofs . . . 31
- 2.2. Application: Surface Retrieval . . . 39
  - 2.2.1. Methods . . . 40
  - 2.2.2. Experiments . . . 45
- 2.3. Future Work . . . 50

3. Rank One Sensing 54
- 3.1. Theory . . . 55
  - 3.1.1. Assumptions . . . 57
  - 3.1.2. Intuitions and Challenges . . . 58
  - 3.1.3. Theorems . . . 68
  - 3.1.4. Proofs . . . 73
- 3.2. Applications . . . 88
  - 3.2.1. Erasure Channel . . . 88
  - 3.2.2. Event-Based Sampling . . . 90
  - 3.2.3. Signals With Latent Variables . . . 91
- 3.3. Future Work . . . 92

4. Continuous Localisation **95**
- 4.1. Theory . . . 96
  - 4.1.1. Intuitions and Challenges . . . 100
  - 4.1.2. Theorems and Proofs . . . 107
- 4.2. Applications . . . 111
  - 4.2.1. Localisation with Anchors . . . 111
  - 4.2.2. Trajectory Self-Localisation . . . 124
  - 4.2.3. General Case . . . 126
- 4.3. Future Work . . . 130

II. Lippmann Photography **134**

1. Past **137**
- 1.1. Colour Photography . . . 137
- 1.2. Lippmann's Method . . . 140
- 1.3. After Lippmann . . . 142

2. Present **146**
- 2.1. Mathematical Model . . . 146
- 2.2. Information Contained in the Lippmann Plate . . . 156
- 2.3. Model Verification . . . 159
- 2.4. Conclusions . . . 171

3. Future **173**
- 3.1. Data Storage . . . 175
- 3.2. Femto-Lippmann . . . 177
- 3.3. Mathematical Model . . . 178
- 3.4. Preliminary Results . . . 182
  - 3.4.1. Pulse by Pulse . . . 182
  - 3.4.2. Interference . . . 184
  - 3.4.3. Increasing the Capacity . . . 190
- 3.5. Conclusions . . . 191

Curriculum Vitae **216**

Part I.

Sampling & Geometry

Chapter 1.

Introduction

Where are you?

Are you at work? In your home town? By the lake? Whatever your answer, it is likely to be your *location* with respect to the landmarks around you, rather than your *position*, for example 46.521 826 9 N 6.632 702 5 S. Observing landmarks and your relation to them is the natural way for humans to navigate and create maps — mental or otherwise. We do this instinctively, but it can take us only so far. To navigate over large distances with sparse landmarks, we need help from technology. Precise positioning and safe navigation over the seas required the creation of good maps, a global coordinate system and precise spring clocks. Today, navigation seems trivial with inexpensive GPS devices [1]. The most common positioning technique — lateration — is also very well understood in theory [2]. The remaining challenge for machines is the localisation by using landmarks in unmapped environments — the tasks that are easy for humans.

With the rise of probabilistic methods, and later with machine learning, positioning and localisation became successful in increasingly challenging conditions. For example, in environments where maps still need to be drawn or when a device moves while collecting sparse or noisy measurements [3].

Our intuition is that these methods work well, because the world has structure and objects do not move arbitrarily. Structure can be incorporated as a prior (in statistical methods) or learned (in machine learning). However, these localisation methods are not well understood from a theoretical point of view. In which scenarios is localisation even possible? What are the intrinsic limits?

To answer these questions, we believe that we need to consider the problem from the sampling perspective. The maps and positions on a computer are, by necessity, discrete with a certain frame rate and resolution. The world they model is continuous. Therefore, we need sampling and interpolation to move from the analog to the digital world, and vice versa.

In this thesis, we extend the understanding of localisation, positioning, and mapping. To this end, we study a number of cases of map making, positioning or sensor localisation with moving devices. Although these cases are simple enough to enable us to find guarantees, they capture some of the typical difficulties in their respective areas.

We begin this chapter with introductions to the research areas of interest. We discuss sampling and interpolation in Section 1.1, focusing on deterministic signals. In Section 1.2, we discuss topics of localisation, mapping and distance geometry. We give a high-level overview of traditional localisation and mapping methods that inspired us, and we dive deeper into geometric approaches to positioning.

In Section 1.4, we give an overview of our work presented in Part I of this thesis. We discuss the gaps in our understanding of localisation and mapping and how a sampling approach could help to bridge this gap. We describe the specific problems that we address in this thesis.

We also briefly discuss matrix recovery in Section 1.3. The in-depth introduction of the area is outside the scope of this work, but as we refer to matrix recovery in Chapter 3, we think that a short reference is needed.

Finally, in Section 1.5, we present assorted mathematical tools that we use throughout this thesis, and we introduce the notation. The reader familiar with the topics introduced there can safely skip this section.

1.1. Sampling and Interpolation

Sampling (the reduction of a continuous signal to a discrete one) and interpolation (expansion of a discrete signal to a continuous one) are at the core of signal processing. In digital signal processing, a continuous signal, e.g., a sound wave, is sampled, processed digitally, and interpolated into a new continuous signal, e.g., a new sound wave. In digital communication, a discrete signal is interpolated, sent as an electromagnetic wave, and then sampled at the receiver.

To successfully process a continuous signal digitally, we need to know if the continuous signal is fully described by its samples; and, to recover the original signal, to know how to interpolate the discrete signal. To efficiently send a discrete signal over the air, we need to know how many discrete samples can be encoded in a continuous wave. Hence, the core questions in the area of sampling and interpolations are, When is the recovery possible? What is the sufficient number of samples? What is the necessary number of samples?

Therefore, it is a not a surprise that the formulation of the first sampling theorem, i.e., the cardinal sampling theorem, is recognised as the beginning of signal processing by the Institute of Electrical and Electronics Engineers (IEEE) [4]. The sampling theorem can be split in two parts [5, 6]. The first states that a bandlimited signal is completely determined by its uniform samples, i.e., values at uniform points, provided it is sampled at the Nyquist rate. The second part describes that 2π-bandlimited signal $x(t)$ can be interpolated from its samples $x(n)$ as follows:

$$x(t) = \sum_{n=-\infty}^{\infty} x(n) \operatorname{sinc}(t-n). \tag{1.1}$$

We owe the modern formulation of the theorem, with the Nyquist rate, to Shannon [7], who knew the statement — but not the proof — from Whittaker [8]. Interestingly, this theorem first arose from studying polynomial interpolation. The word *cardinal* in the name of the theorem refers to cardinal functions: functions obtained by taking the limit of interpolating polynomials for infinite number of uniform samples. Functions that can be interpolated with the cardinal functions are exactly the bandlimited functions used in the theorem. Bandlimited functions are also the simplest of all signals that fit the uniform samples, because they "wiggle the least" between samples. The term *the Nyquist rate* used by Shannon originates from Nyquist's famous 1928 paper [9], where he studied digital communication. The Nyquist rate originally meant the amount of code pulses that can be sent through a channel with a given bandwidth.

We use the name cardinal sampling theorem rather than Whittaker-Shannon-Nyquist Theorem or a similar term, because there are more people who independently formulated or proved this result. In the times of Shannon, it was proved by Kotel'nikov [10, 11] in Russian literature, by Raabe in German, by Someya in Japanese, and by Weston (curiously, in English, the language of Shannon). Even before the work of Shannon, Ogura [12] provided the idea of the proof in terms of cardinal series. The first part of the theorem can be traced all the way back to Borel (1897). In our humble opinion, it would be fitting to call this theorem *the fundamental theorem of signal processing*, after the fundamental theorems of algebra and calculus.

As pivotal as the cardinal-sampling theorem has been, the assumption that the signal is bandlimited and that the sampling positions are uniform is restrictive. The real world signals are never exactly bandlimited [13], and the Shannon sampling formula (Equation (1.1)) is impractical due to the slow decay of the sinc function [14]. Furthermore, in many situations, the data is known only at non-uniform positions [15], for example, in communication theory, astronomy or medical imaging. Therefore, many extensions have emerged over time. We describe some of them in the remainder of this section.

The Earliest Extensions

Some of the first [16] extensions to the cardinal sampling theorem include using both the function values and derivatives, extensions to multiple dimensions (described in more detail in Multi-channel Sampling), sampling band-pass rather than bandlimited functions, and the very first results in non-uniform sampling [17] (Irregular Sampling). The earliest results in non-bandlimited sampling date as far back as 1908 to Vallée Poussin's [18] and later Marie Theis' [19] works on duration-limited signals.

Another early group of extensions is based on the observation that the sampling theorem is implicitly based on the Fourier transform. The signal is given by the inverse Fourier transform of its compact spectrum. The interpolation and the sampling positions are given an orthogonal basis of such signals. Replacing the Fourier transform with another integral transform, e.g., Hankel (Bessel) transform in optics leads to a new sampling theorem. The limitation of this approach is that it ties together the class of sampled functions, the sample positions, and interpolation formula. The results for many different integral transforms have been unified under the framework of reproducing kernel Hilbert spaces [20]. We return to Hilbert spaces in Sampling in Hilbert Spaces.

Wavelets

Continuous wavelet transforms and wavelet series are other examples of integral transforms that can be used to formulate sampling and interpolation results. However, wavelets are not only an interpolation method, but a research field in their own right, with many applications in data compression, denoising, differential equations with discontinuities [21], dictionary learning [22], and gravitational-wave detection [23, 24]. Here, we give only a brief overview.

Wavelets can be thought of as functions that are created from a prototype wavelet through shift and scaling. Wavelet transforms (discrete and continuous), as well as wavelet series, provide a linear method of decomposition of a signal into wavelets. They are a tool for time-frequency analysis, but where frequency is implicitly described by scale. This makes wavelets particularly well-suited for the analysis of multi-scale and non-stationary signals [21, 25, 26]. Due to its fractal structure, a discrete wavelet transform is fast; more precisely, it is linear in the input length [21, 26]

There are many different wavelet families, designed for different purposes. Many of them are frames and provide redundant representation of the signal. There are also wavelet bases, even orthogonal wavelet bases, but they are much more restricted in shape than redundant wavelets [21].

Wavelets have their origins in engineering, physics, and pure mathematics [27]. They owe their name to a geophysicist J. Morlet [28] who used different windows for different frequencies to analyse seismic data [26]. Since then, many people found connections between wavelets and their own fields. A theoretical physicist A. Grossmann, found the connection to coherent states formalism in quantum mechanics [29]. A harmonic analyst, Y. Meyer, found the connection to the analysis of singular-integral operators by Caldéron [28]. Mathematician S. Mallat noticed the connection with the multi-resolution analysis from computer vision [28] and used it to create the layered structure of wavelets [29]. Mallat and I. Daubechies are credited for developing transition between continuous and discrete signals [26].

There are many other scientists who have contributed to the field [26]: Stromberg discovered the same orthogonal wavelets as Mayer; Paul Levy, Littlewood and Paley extended the work by Haar; and electrical engineers, C.D. Esteban, C. Galland, M.Smith and T.Barnwell, worked on iteratively applied filters (filter banks) similar to the multiresolution analysis by Mallat [25, 26, 29].

Multi-channel Sampling

Multi-channel sampling, also called generalised sampling, is an early extension to the cardinal sampling theorem. This sampling scheme assumes that we pass the signal through m linear shift-invariant systems, and that we sample each result regularly at $1/m$th the Nyquist rate [30]. Then, we ask for the reconstruction of the continuous input signal. Sampling function values and first $m-1$ derivatives falls under this framework, as well as certain wavelet transforms and filter banks. Multichannel sampling can also be applied to multi-dimensional or multi-component signals [31, 32].

Irregular Sampling

There are situations that require irregular sampling, for example astronomy [33], where the measurements can be taken in only certain conditions, or in magnetic resonance imaging, where spiral sampling grids are more efficient than regular ones.

Irregular, also called non-uniform, sampling first appeared in the works of Yen [17] in the 50s. Yen considered a special case when some samples are moved from regular grid [17].

Since then, there have been three approaches to the problem. One is to use the same spaces as for uniform sampling, e.g., bandlimited functions [34] but also for wavelet and spline-like spaces. The main challenge of this approach, especially if the samples are concentrated in one place, is stable reconstruction [14]. The main result is the Beurling-Landau theorem [35, 36] about the minimum sample density needed to recover bandlimited functions. There are many fast iterative algorithms available for this approach [15, 37, 38].

The second approach is to consider special interpolation spaces suitable for non-uniform sampling, e.g., non-uniform splines [14], in particular B-splines. There, one of the core results is that, in the limit of the order of spline going to infinity, the spline interpolation converges to a bandlimited function [39, 40]. For more good references about these two approaches, we refer the reader to the work by Aldroubi and Gröchenig [15].

The third approach is based on using the Hilbert space formalism, described in the next section.

Sampling in Hilbert Spaces

Up to this point, we have talked about sampling functions from different classes: bandlimted functions, band-pass functions, and wavelets. All of them, and many others, e.g., uniform splines, can be described in the framework of shift invariant spaces: spaces of functions generated by regular shifts of a given function or set of functions.[1]

The framework of shift invariant spaces introduces a geometric view on sampling, based on the Hilbert space formalism [14, 41]. The core observation is that sampling and interpolation is a projection between Hilbert spaces [14]. For signals that belong to the interpolation space, this means exact recovery and provides a unified proof for a number of different sampling theorems: For example, the reproducing kernel Hilbert spaces framework corresponds to the orthogonal projection.

For signals that are not in the interpolation space, the projection perspective provides methods for recovering a signal as close as possible to the original. This leads to an area of error estimation, in particular entirely a Fourier-based error estimation [42, 43]. As we cannot recover the signal exactly, in order to judge the quality of the setup, we can resample the interpolated signal and check whether the sample values change. If they do not change, we say that the sampling and interpolation setup is *consistent* [44].

Although sampling in Hilbert spaces originated from shift-invariant spaces, it is much more general than this. It has been extended to non-uniform sampling or, more precisely,

---

[1] These regular shifts make shift invariant spaces inherently linked with regular sampling.

to sampling with arbitrary bases [45, 46]. New notions of optimality and more stable algorithms were proposed [47].

Implicit Sampling

Another area where the sampling is not uniform is implicit sampling: when the sampling positions depend on the input signal, e.g., in neuromorphic computing[2]. Implicit sampling is, in general, a non-linear operation, hence a linear operation is not sufficient to recover the signal [48]. Therefore, the interpolation methods discussed thus far do not necessary apply to it.

Implicit sampling, also called signal-dependent sampling or event-based sampling, is growing in popularity [49], in particular in the field of vision [50], because the event-based cameras recently became commercially available [50, 51]. Event-based sampling is more energy efficient that classic sampling, and has potential for better resolutions and dynamic range [50, 52]. The basic principles of implicit sampling are to encode information in sample positions, rather than sample values, and to produce the samples only when something interesting, i.e., an event, occurs.

The idea was introduced by Bond and Cahn [53], in the form of recovery of signals from their zero crossings: locations of real and complex zeros [54]. Later, Longan characterized band-pass signals that can be represented (up to scaling) by their zero crossings [54]. The idea was extended to sine crossings [55]: intersections with a known sinusoid and level crossings [56].

Recently, a renewed interest in neuromorphic computing [49] created new applications for level-crossing sampling. This inspired researchers to study time-encoding machines (TEM): idealised event-based sensors [57, 58]. A TEM produces samples when the integral of the signal hits a threshold, thus imitating the integrate-and-fire neuron. In addition to preserving crucial properties of neuromorphic computing, it is amenable to study. For example, bandlimited functions [57] and functions from shift-invariant spaces [58] can be recovered from samples created by a TEM, provided that certain conditions, similar to sampling rate, are met.

Another family of implicit sampling reacts to change in the signal, rather than its actual value. This includes delta modulation [59], sigma-delta modulation [60], and send-on-delta sampling [61], where pulses are generated when the signal amplitude changes by more than a given difference. It should be noted that there is a close relation between send-on-delta sampling of a signal and level-crossing sampling of the integral of this signal.

Non-subspace Sampling

Most of the sampling and interpolation frameworks we have discussed so far concern signals from a known linear space. There are situations, however, when the signals belong to unions of subspaces [62–64] that are not themselves subspaces.

The most widely known example is compressed (or compressive) sensing [65–68], where discrete signals in $\mathbb{R}^N$ are assumed to be k-sparse, i.e., to have only $k \ll N$ non-zero

---

[2] As the name suggests, neuromorphic computing is inspired by the workings of neurons.

values. Adding two k-sparse signals results in a signal that has up to $2k$ non-zero values and might not be k-sparse. Thus, k-sparse signals are not a linear subspace.

Another example is continuous signals with a finite rate of innovation (FRI): finite and bounded number of degrees of freedom per unit of time [69, 70]. Examples of signals with a finite rate of innovation include streams of Diracs or non-uniform splines. Here again, adding two signals can double the rate of innovation.

Probably the oldest example of non-subspace sampling is sampling signals with unknown spectral support, i.e., signals that occupy a known fraction of the spectrum, but at an unknown location [35, 71].

In principle, we can always extend the sum of subspaces to a larger subspace. However, in the case of compressed sensing, this would require $O(N)$ samples, when only $2k$ numbers[3] are needed to specify a k-sparse signal. In the other two cases of non-subspace sampling, the smallest linear space that contains the union of subspaces is infinitely dimensional. Therefore, more efficient methods are needed.

Each of the aforementioned sampling schemes has their own methods and challenges. In compressed sensing, the restricted-isometry property (RIP) [72] is the classic tool; but it was quickly replaced with more general incoherence properties [73, 74]. These properties were used to show that a sparse signal can be recovered from random measurements, and that the number of samples needed is of the same order of magnitude as the number the significant coefficients of the signal. Later, methods that do not require either RIP or incoherence were developed [75, 76]. Compressed sensing was also extended to low-rank matrices, see Section 1.3.

In the sampling of FRI signals, the primary tool traditionally is Prony's method [77], also known as the annihilating-filter method [78]. The method originated from the spectral estimation of signals with a finite number of spectral components [79] but was then generalised [69]. In recent works, Prony's method was replaced with more robust optimisation methods [80, 81].

For signals with unknown spectral support, it was shown that the number of samples required for recovery is much smaller than needed for subspace sampling [71]. Finally, Lu [62] proposed a generalised theoretical framework for sampling signals from a union of subspaces. This frameworks provides a geometrical explanation of why $2k$ samples are often needed to sample a union of k-dimensional subspaces. Both signals with unknown spectral support and FIR signals can be interpreted as infinite-dimensional compressed sensing [82].

Sampling at Unknown Locations

Compared to other frameworks, sampling at unknown locations is a relatively unexplored topic. Yet, sometimes sample locations are not known: for example, if the stream of uniform samples is sent over a deletion channel, or when the samples are taken with multiple sensors whose ordering is unknown.

Interestingly, the capacity of a deletion channel is still an open problem [83]. Nevertheless, under certain conditions, both the discrete bandlimited signal and the signal

---

[3] That is, the non-zero values and their indexes.

positions can be recovered [84]. Recently there have been a number of results on sampling with unknown permutation of sensors; these results are often called unlabelled sensing [85–91].

Another situation where sample positions could be unknown is when they are taken with a jittered clock. In this case, samples are approximately uniform and an alternating least-squares algorithm can be shown to converge to a local minimum [92]. If the alternating algorithm is too costly, we can treat the samples as if they were uniform. If the distribution of sample positions satisfies certain weak conditions, the reconstruction error will be asymptotically inversely proportional to the number of samples [93–96].

Sampling of Composites of Functions

Unlabelled sensing can be also interpreted as sampling a composite of two unknown functions — the permutation and the signal — at known locations. Time warped signals [97–101] also fall into this category. For example, it can be shown that for particular warpings of bandlimited signals, unique recovery is possible [102]. The sampling of a composite of functions can be also interpreted as non-subspace sampling, where the warping defines to which linear space the signal belongs.

There is an interesting connection between sampling composites of functions and single-index models [103] that are commonly used in econometrics. However, with single-index models, only the linear part of the warping is recovered and only up to a scaling — only the direction of the warping.

We will see more on the composite of functions in Chapter 2.

1.2. Geometry Reconstruction

A mammoth tusk, with carvings representing a mountain, a river, valleys and routes to the south of present-day Brno, in the Czech Republic, might be the oldest preserved map, dated to 25,000 BC [104, 105].

Since then, our maps have become much more sophisticated. They are precise, interactive, and can display 3D topography. We can navigate to a given location. This progress would not be possible without the advances in physics and time keeping that led to a GPS system [1], as well as advances in computer vision that enabled 3D reconstruction from images [106].

The modern maps make autonomous vehicles, such as self-driving cars, possible. Although the use of modern maps makes our life easier, it impairs our spacial knowledge [107].

Map making and navigation are not the only areas where the estimation of geometry is needed. For example, geometry is crucial in understanding human health. First, sensor networks monitor, for example, pollution and temperature. To make sense of the produced data, the geometry of the sensor network has to be measured or calculated [108, 109]. On a smaller scale, understanding the human brain requires, for instance, understanding the brain structure, and this has to be reconstructed from different kinds of data [110]. On an even smaller scale, the function of a protein is defined by its shape. Therefore, to understand the biological processes inside cells, we first have to understand the structures of the proteins involved; these structures can be estimated from measurements [111] or predicted from the DNA sequence [112, 113].

In this section, we give a general overview of some of the problems that can be jointly characterized as geometry reconstruction. No proper literature review that talks about map-making and positioning can neglect the famous robotics problem of simultaneous localisation and mapping (SLAM). Therefore, we discuss it first. We then proceed to areas that are more immediately related to our work, such as structure from motion, and range-based localisation. Finally, we discuss the existing work on trajectory recovery.

Simultaneous Localisation and Mapping

SLAM solves the problem of localising a moving device in an unknown or rapidly changing environment [114, 115]. For SLAM to be practical, it has to be real-time and robust.

In the absence of GPS, the simplest way to obtain the real-time position of a robot is to equip it with a module that can track motion. This can be inertial measurement unit for odometry or a camera for visual-odometry. The problem with odometry is that it is not robust. The errors in the position estimated accumulate over time, thus resulting in a drift. Therefore, SLAM systems employ various *loop-closure* methods to make sure the same place visited twice is recognised as such in the robot's internal map. To solve the loop-closure problem, some information about the environment has to be incorporated. This information can be obtained from a camera (visual slam [116]), lidar, radar, or even microphones (acoustic slam [117–119]). Often different measurements are complementary and can be combined with sensor fusion [3]. Many of the methods that we will discuss later in this section could be the source of environment information for SLAM.

There are also different ways to integrate the environment information. The classic methods of estimation of robot and landmarks positions are Bayesian, e.g., extended Kalman filters (EKF-SLAM) Rao-Backwellized particle filters (FastSLAM) [120, 121]. In these methods, at each step of a robot's movement, the estimation of the robot's position is updated based the detected landmarks; and landmark information is updated based on the estimation of the robot's position. Classic methods are well-studied and suitable when multiple sensor fusion is needed. However, for a visual slam with a single camera, key-frame methods perform better by separating position estimations and landmark estimations, [116].

It is possible for a robot to build a consistent map of the unknown environment due to a high degree of correlation between landmark estimates, which grows with each measurement. Over time, even if a robot's position is uncertain, the relative map of landmarks converges [121, 122].

Finally, there are methods that can incorporate information from multiple devices via *collaborative SLAM*. Such approaches can improve not only environment mapping, but also collision avoidance and route planning. They usually use a centralized architecture [123].

Structure from Images

In this subsection, we discuss methods called jointly *structure from X*, or *shape from X*, where the *X* can be, e.g., motion, shading or texture. Their purpose is to reconstruct the 3D model of an object, or of the environment, by using one or multiple photographs. In many cases, the position of the camera is also reconstructed.

In **structure from motion** (SfM), the input data consists of multiple *unordered* images of the same scene, taken from different viewpoints [124]. The output is a 3D model of the scene and an estimate of the pose of the camera for each of the input images. This was traditionally done by extracting key feature points from the scene, matching them between views and verifying the geometry [124–126]. This iterative matching and verification in a greedy way led to the high complexity of SfM algorithms [127]. Over time, deep-learning methods overtook the field [128], as they can learn good priors about how the real-world objects look. Recently, differentiable rendering [129] was used to estimate not only the shapes but also the textures, thus producing a scene ready to render [130].

The purpose of **shape from texture** (SfT) is to recover the shape of the object from the perceived distortions of its texture [131, 132]. The main advantage of SfT is that it can be estimated using a single photograph.

The basic principle of SfT is that the texture viewed at an angle will be compressed. If the camera projection is not perfectly orthographic, additional deformations are introduced, which become stronger farther from the centre of the image. The first step in SfT is to measure this distortion and to infer the surface normals (direction perpendicular to the surface). Next, the surface coordinates are calculated. In order to estimate the surface distortion, some assumptions on the texture have to be made, e.g., homogeneity or isotropy. A specific texture model can also be assumed, e.g., 2D wavelets called warplets [132].

Another assumption that can be made on a texture is its spectral content [133, 134]. This special case of SfT is often called **shape from bandwidth** (SfB) [102, 135].

Shape from a single or multiple image can be also recovered based on a light and shade pattern; this is called **shape from shading** (SfS) [136]. It is a classic problem in computer vision, accredited to Hom [137]. As both the shape and the light source have to be recovered, the problem is ill posed and additional constrains, such as smoothness or integrability, are needed.

Our brain can estimate the shape of an object in a still photo, not only by shades and textures but also via a visual system's knowledge of objects [136]. Therefore, it is not a surprise that deep learning accurately reconstructs the shape from a single image. Recent works focus on finding the best 3D geometry representation, rather than improving the already excellent accuracy. Some of the proposed representations include point clouds [138], surfaces [139], meshes [106] and occupancy networks [140] that classify if the given point is inside or outside the object.

Range-Only Localisation

There are situations when we cannot, or do not want, to use vision-based methods. This could when the environment does not exhibit enough reliable features, for example under water [141, 142], at high altitudes [143], and in large exhibition-style rooms [144]. We might not use cameras if we want to preserve privacy of people in these spaces. In such situations, it is more feasible to use distance measurements, for example, time of flight or received signal strength of Wi-Fi or Bluetooth.

In some of these scenarios, we control the positions of the measurement devices, for example, if we can equip an exhibition room with Bluetooth sensors. Then, we can simply use lateration methods that we discuss below. In other situations, for example, as for the calibration of ad-hoc microphone arrays, we do not know positions of any devices. We discuss this situation later in this section (Localisation from a Graph of Distances).

Lateration The lateration problem is to recover a position of a point from distance measurements to a number of known points, called anchors. It is known that at least 4 distance measurements[4] are needed to localise a point. This is the principle behind GPS positioning [1].

Lateration is a classic problem and there are different methods for solving it, many of them are in closed form or are non-iterative [2]. The techniques differ in how they handle the noise. For example, in squared range least squares (SRLS)[5] [145] it is assumed that *square* measurements are measured, thus the noise is added to squared distances. In certain situations, this model behaves differently than range least squares (RLS) [146], where it is assumed that noise is added directly to the distances. Although both problems are non-convex, RLS is more difficult to solve and non-linear least-squares solvers, such as Levenberg-Marquardt algorithm [147], have to be used.

Localisation from a Graph of Distances There are many situations where we know some of the distances between points, but we do not have any reference locations. This could be in the localisation of wireless sensor networks [109, 148][6], or in the position calibration of ad-hoc microphone arrays [149]. In localisation with sound waves, distances to nearby walls are measured, but the wall's positions are not known [150, 151]. Another situation when no anchors are available is the molecular confirmation problem, where the points describe atoms, and the distances between them are measured by nuclear magnetic resonance (NMR) [111] or crystallography [152]. A related problem appears in the visualisation of multi-dimensional data sets, if data points are related by a distance that should be approximated in the visualisation [153].

Partial distances can be represented by a graph, where nodes are the points we want to localise (sensors, atoms). If the distance between two points is measured, then there is an edge between the corresponding nodes in the graph; the edge is weighted by the distance.

---

[4] If we are interested in 2D position only, 3 distance measurements are sufficient.

[5] SRLS also stands for *simple robust least squares*.

[6] Wireless sensor network localisation is sometimes called self-localisation [108].

Given a graph, we want to know if its embedding in D dimensions is unique (up to rotations and translations). This question is relevant for sensor-network design and, in the noiseless case, can be addressed by rigidity theory that provides efficient algorithms to check uniqueness of the embedding [148, 154]. A similar problem is when we have all the distances, but some of them are corrupted; we want to know if it is still possible to recover the correct embedding [155].

In the *distance geometry problem* [156–158], we want to know if our measurements can be embedded in D dimensions, without any distortions. We do not find this approach useful in real world, see Box 1.1.

In data visualisation, we want to know how to best approximate the multi-dimensional distances on a two dimensional plot. We know all the distances, but they are noisy. In this case, we can use methods called multi-dimensional scaling [159, 160], and multi-dimensional unfolding [161]. The latter is suitable, in particular, when the data similarity forms a bipartite graph.

In the localisation problem, we rarely have all the distances. Various optimisation methods can be used to estimate the positions from noisy and incomplete distances. Here, due to its straightforward formulation, we use the Euclidean distance matrix (EDM) [162–164], as an example. EDM theory can be extended to situations when permutations of distances are no longer known [165].

The name Euclidean distance matrix is not entirely accurate, as EDMs are matrices of *squared* Euclidean distances; this gives them better algebraic properties [163, 164]. They are useful descriptions of point sets, because there is one-to-one mapping between a EDM and the Gram matrix, i.e., a matrix of inner products between vectors of point coordinates. From a Gram matrix, the positions of the points can be easily recovered, up to global rotations and translations, via matrix factorisation.

More formally, consider a set of N points in D dimensions $\boldsymbol{X} \in \mathbb{R}^{D \times N}$. The entries of the EDM, assuming all are available, are squared distances:

$$d_{jk} = \|\boldsymbol{x}_k - \boldsymbol{x}_j\|^2 = \boldsymbol{x}_k^\top \boldsymbol{x}_k - 2\boldsymbol{x}_k^\top \boldsymbol{x}_j + \boldsymbol{x}_j^\top \boldsymbol{x}_j$$

therefore, we can express the EDM $\boldsymbol{D}$ in the matrix form:

$$\boldsymbol{D} = \mathbf{1}_N \operatorname{diag}(\boldsymbol{G})^\top - 2\boldsymbol{G} + \operatorname{diag}(\boldsymbol{G})\mathbf{1}_N^\top,$$

where $\boldsymbol{G} = \boldsymbol{X}^\top \boldsymbol{X}$ is the Gram matrix of the N points. Moreover, Gram matrix can be calculated from the EDM by multiplication via two matrices, see [164].

As we do not have all distances available, we cannot calculate $\boldsymbol{G}$ directly. We can, however, formulate it as an optimisation problem:

$$\begin{aligned} \underset{\boldsymbol{G}}{\arg\min} \quad & ||\widetilde{\boldsymbol{D}} - \boldsymbol{W} \odot \Big(\mathbf{1}_N \operatorname{diag}(\boldsymbol{G})^\top - 2\boldsymbol{G} + \operatorname{diag}(\boldsymbol{G})\mathbf{1}_N^\top, \Big) ||^2 \\ \text{s.t.} \quad & \operatorname{rank}(\boldsymbol{G}) = D \\ & \boldsymbol{G} \succcurlyeq \mathbf{0}, \end{aligned}$$

where we know that $\boldsymbol{G}$ has to be rank D in order to describe points in D dimensions.

This problem can be relaxed and solved with various methods of low-rank matrix recovery (including semidefinite programming [166], see Section 1.3: Semi-definite Programming), smart initialization, expectation-minimisation, or locally rigid embeddings [154].

Parametric-Trajectory Recovery

All the methods discussed, so far, model device positions as single points; if the device is moving, it creates a sequence of its positions. But, as mentioned in the introduction, in the real world, the movement is continuous. To model this more accurately, a number of methods were proposed, predominantly using splines. A comprehensive review of this field is given in [167]. Li *et al.* [168] solve the classic SLAM problem, replacing the position update of the usual state-space equations with a continuous, parametric trajectory. Li et al. consider polynomial-basis functions and update over time the trajectory coefficients. They use a standard iterative solver that, in general, converges to a local minimum. Other methods [167, 169, 170] solve the same trajectory estimation problem, thus parameterizing the trajectory with B-spline basis functions. As B-splines have local support, they automatically offer more flexibility in fitting complex trajectories, without recursively updating the coefficients. However, these papers lack optimality guarantees because Gauss-Newton solvers are used.

After our results were published, parametric trajectory models were used in the localisation of diffuse sources [171] and in target tracking [172].

Trajectory models can also be integrated in the traditional EDM framework with so-called **kinetic EDMs** [173], where all points are considered to move on trajectories.

Want to know more? 1.1: Distance Geometry

Some authors claim that distance geometry has applications in positioning, in sensor-network localisation and protein-structure estimation [156]. We find these claims a little far-fetched.

First, as the embedding has to be exact, there is no space for modelling noise, hence distance geometry cannot handle real data. Second, the distance geometry problem can, in principle, be solved with a non-constructive proof, i.e., without providing any positions. Even a constructive proof might not give uniqueness guarantees. Finally, the graph of distances that were created by sensors in a D dimensional space can always be embedded in D dimensions, (up to the error due to measurement noise). Hence, in any real-world localisation scenario, the distance geometry problem is trivial.

Dynamical-distance geometry [174, 175] suffers from the same problems, but with points moving over time. To our knowledge, there is no trajectory modelling in dynamical-distance geometry, and the time is discrete.

1.3. Matrix Recovery

We have mentioned low-rank matrix recovery twice now: in the context of compressed sensing (Non-subspace Sampling), and in the context of range-only localisation (Localisation from a Graph of Distances). In addition to localisation, low-rank matrix recovery has applications in recommender systems and collaborative filtering [176], image processing [177], and computer vision, e.g. structure from motion [178, 179].

In this section, we give a quick overview of matrix recovery from linear measurements. A general review is outside the scope of this thesis.

Formally, in the low-rank matrix recovery problem, we are given measurements

$$y_n = \langle \boldsymbol{A}_n, \boldsymbol{X} \rangle,$$

where $\boldsymbol{X}$ is an unknown low-rank matrix and $\boldsymbol{A}_n$ are known sensing matrices. A special case when $\boldsymbol{A}_n$ are all zeros except one entry is called *matrix completion* because, in this case, $\boldsymbol{A}_n$ determine which entries of $\boldsymbol{X}$ are measured. Matrix completion is usually formulated using binary $\boldsymbol{W}$ that indicate which entries are available:

$$\boldsymbol{Y} = \boldsymbol{W} \odot \boldsymbol{X}.$$

Here, we use the name *matrix sensing* when we refer to matrix recovery that is not matrix completion. For a more detailed overview of matrix sensing, we refer the reader to [180].

As mentioned in Section 1.1, low-rank matrix recovery can be interpreted as a special case of compressed sensing where, instead of assuming that the unknown we want to recover is sparse in the standard basis, we assume that that its spectral (in matrix sense) representation is sparse.

As searching for a sparse solution is difficult, nuclear-norm minimisation is used instead. This can be done with proximal algorithms [180] with semidefinite programming [181] (described at the end of this section) or with iterative algorithms [180, 182].

There are many works that study when the solution to the relaxed problem is unique and is the solution to the original problem. The most widely known results are probabilistic, i.e., show that the solution is unique with high probability. In matrix completion, to guarantee the uniqueness, certain conditions are assumed on the measured matrix; they can be based on compressed sensing [181, 183–185] or on the incoherence [186]. In matrix sensing, certain assumptions are made on matrices $\boldsymbol{A}_n$. One such assumption is the restricted isometry property (RIP) [187] that, in the matrix case, has connections with the Johnson–Lindenstrauss lemma [188]. There are also incoherence based results in matrix sensing [189, 190].

Although probabilistic results are used widely as incentive for the use of low-rank matrix completion in applications, they are not always realistic. There are situations when we have a fixed set measurements that we want to know if the unique recovery from this set of measurements is possible.

This was the basis for matrix completion from deterministic masks [191]. The results were obtained using coherence properties of the recovered matrix [192, 193], by using rigidity theory [194], geometry [195], and algebraic geometry [196]. Recently, a question about which matrices can be completed for a specific mask was studied [197]. In the case of matrix sensing, deterministic matrices that satisfy RIP were constructed [198].

Semi-definite Programming

Using semi-definite programming (SDP) is not the only method for recovering a matrix, nor the most general one. It is, however, relevant in this thesis.

SDP is an area of convex optimisation[7] concerned with symmetric matrices. The objective is linear, and there can be two kinds of constraints: affine and positive semi-definite. We state below one of the many formulations of SDP:

$$\begin{aligned} \operatorname*{arg\,min}_{\boldsymbol{X}} \quad & \langle \boldsymbol{C}, \boldsymbol{X} \rangle \\ \text{s.t.} \quad & \langle \boldsymbol{A}_n, \boldsymbol{X} \rangle = b_n \qquad \text{for } n \in [N] \\ & \boldsymbol{X} \succcurlyeq \boldsymbol{0}. \end{aligned}$$

The intersection of the cone of positive semi-definite matrices with an affine space is a convex set, hence the whole problem is convex.

As a convex problem, SDP can be solved efficiently using, among other methods, interior-point methods. It has also been proven that, in the worst-case scenario, it can be solved in polynomial time [200]. Another advantage of SDP is that it is a generalisation of linear and quadratic programming [200].

Semi-definite programming is often used in problems that are not originally convex but that can be relaxed to a SDP. It is called *semi-definite relaxation* [166, 201]. It was first used by Lovász [202] and Goemans and Williamson [203]. For many non-convex problems, it returns a correct solution or at least a good initialization for further optimisation. Its accuracy and optimality conditions were studied for a number of non-convex problems [204].

1.4. Geometry from the Sampling Perspective

We have seen that among the methods used in various geometry problems, many model the world in a discrete manner. Yet, there is no geometry representation that is suited to all tasks; for example, machine-learning methods use many different representations. Both classic and learned geometry reconstruction methods rarely provide recovery guarantees; if they do, the guarantees are often asymptotic. We believe that guarantees for geometry problems can be obtained by modelling the world in a continuous way. To do this, we propose to take a sampling perspective and tap into the signal-processing expertise.

We demonstrate this approach with two examples. The first problem bridges shape from texture with sampling at unknown locations (Chapter 2). We propose a toy problem of surface recovery by using bandlimited or polynomial texture, and we show how it can

---

[7] The word “programming” refers to planning; the name *linear programming*, which created this convention, has been used before the word “programming” was used for coding, as Dantzig states in the foreword to his book about linear programming [199].

be formulated as a special case of sampling at unknown locations, where the sample positions are constrained. The second problem we study connects range-only trajectory recovery and matrix sensing by using rank-one measurements (Chapters 3 and 4). We devote two chapters to it. In Chapter 3, we consider rank-one sensing as an abstract problem and, in Chapter 4, we look at its applications in distance geometry. We make this separation, as we feel that rank-one sensing — which fits in between matrix sensing and matrix completion — is also relevant in other applications, e.g., in event-based sampling [205] or when measuring signals with latent variables.

In these three problems, we consider polynomial and bandlimited models to represent the continuous objects, e.g., textures and trajectories. We prove sufficient and/or necessary conditions for a unique recovery of the geometry. We propose algorithms for recovering the geometry and show how they perform in the presence of noise, on simulated or real-world data.

By analysing these problems, we expand the understanding of the principles of geometry reconstruction and other fields. We provide building blocks upon which more application oriented methods can be built, and we provide bounds on the theoretically possible. Moreover, we hope our results can point the community to unexplored, but interesting research areas, as there are more applications that would benefit from a more abstract linear-algebraic point of view.

In our work, we focus on providing guarantees, hence we do not compare with state-of-the-art in terms of speed or robustness of our algorithms. Furthermore, except for Chapter 2, we consider only subspace sampling; non-subspace sampling could be incorporated into a spline trajectory model or low-rank assumption in matrix sensing. Finally, many of our results in Chapters 3 and 4 are up to a measure zero set; we show that the results cannot be extended to all points.

Thesis Structure

As mentioned in How to read this thesis?, this thesis consists of two parts.

Part I is organised as follows. In the remainder of this chapter, we provide a reference of mathematical concepts used in this thesis. In Chapter 2, we study sampling at unknown locations, with an application to surface recovery, and in Chapter 4, we study localisation of moving devices. The latter is based on our work in matrix sensing with rank-one measurements; we describe this in Chapter 3.

As mentioned in How to read this thesis?, at the beginning of this thesis, we separate our main results from the more educational discussions designed for a curious reader. The latter occupy mostly sections entitled “Intuitions and Challenges” and boxes entitled “Want to know more?”.

Finally, Part II is devoted to the totally different topic of Lippmann photography, to which we nevertheless apply signal-processing methods.

1.5. Mathematical Tools

In this section, we define some generally known concepts and state some facts that will be useful in this thesis, but we do not want to clutter further chapters with them. In particular, we define the bases of polynomials and bandlimited functions that we use in Chapters 2 to 4.

Function Spaces

Throughout this thesis, we focus on polynomials and on bandlimited functions on a real line or on an interval. Below, we define vector spaces of polynomials and bandlimited functions and specify the bases that we use. For consistency, we define both families to be K dimensional; this means that we use K-bandlimited functions and polynomials of degree at most $K-1$. We use $\mathcal{F}$ to refer to both these families, as we often use them interchangeably. We use $\mathcal{F}^J$ if we want to refer to families defined as below, but with K replaced by J.

Polynomials We define the family $\mathcal{F}_p$ of polynomials of real variable of degree at most $K-1$ as follows:

$$\mathcal{F}_p = \left\{ \sum_{k=0}^{K-1} a_k t^k : a_k \in \mathbb{R} \right\}, \tag{1.2}$$

where a_k are coefficients in the basis $1, \dots, t^{K-1}$. This definition can be expanded to complex polynomials by replacing $t \in \mathbb{R}$ with $z \in \mathbb{C}$ and similarly extend coefficients to complex values $a_k \in \mathbb{C}$.

We can represent a polynomial $p \in \mathcal{F}_p$ as a product between its coefficients and a vector of basis functions:

$$p(t) = \langle \boldsymbol{a}, \boldsymbol{f}(t) \rangle,$$

where $\boldsymbol{a} = \begin{bmatrix} a_0, \cdots, a_{K-1} \end{bmatrix}^\top$ and $\boldsymbol{f}(t) = \begin{bmatrix} 1 & t & \cdots t^{K-1} \end{bmatrix}^\top$. We often refer to $\boldsymbol{f}(t)$ as the basis vector (of functions).

Bandlimited Functions We define the family $\mathcal{F}_b$ of *real-valued* K-bandlimited functions on the interval $[0, T]$ as follows:

$$\mathcal{F}_b = \left\{ \sum_{k=0}^{(K-1)/2)} a_k \cos(tk) + \sum_{k=1}^{(K-1)/2} b_k \sin\left(\frac{2\pi}{T} tk\right) : a_k, b_k \in \mathbb{R} \right\}. \tag{1.3}$$

Similarly as in the polynomial case, we define the basis vector $\boldsymbol{f}$:

$$\boldsymbol{f}(t) = \begin{bmatrix} 1 & \sin\left(\frac{2\pi}{T} t\right) & \cos\left(\frac{2\pi}{T} t\right) & \cdots & \sin\left(\frac{K\pi}{T} t\right) & \cos\left(\frac{K\pi}{T} t\right) \end{bmatrix}^\top \tag{1.4}$$

We define the family $\mathcal{F}_c$ of *complex-valued* K-bandlimited functions on the interval $[0, T]$ as follows:

$$\mathcal{F}_c = \left\{ \sum_{k=-(K-1)/2}^{(K-1)/2} a_k e^{i \frac{2\pi}{T} kt} : a_k \in \mathbb{C} \right\}. \tag{1.5}$$

We also define basis vector. For consistency with real-valued bandlimited functions, we order elements according to their frequency:

$$\boldsymbol{f}(t) = \Big[1 \quad e^{i\frac{2\pi}{T}t} \quad e^{-i\frac{2\pi}{T}t} \quad \cdots \quad e^{i\frac{K\pi}{T}t} \quad e^{-i\frac{K\pi}{T}t}\Big]^\top \tag{1.6}$$

Of course, these definitions require K to be odd; and we implicitly assume this in the rest of this thesis.

Sinc Function Here we define the sinc function as

$$\operatorname{sinc}(t) = \frac{\sin(t)}{t}.$$

Shifted sinc functions can be used as a basis of the space of bandlimited functions, see Equation (1.1).

Matrices

Here, we define and state a number of facts about matrices. Many of them are widely known, but we want to provide the reader with reference. Unless stated otherwise, the facts in this section below are from the book of James Gentle [206].

Trace Tricks An inner product of matrices can be defined using the trace:

$$\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \operatorname{tr}\Big(\boldsymbol{A}^\top \boldsymbol{B}\Big),$$

$\boldsymbol{A}, \boldsymbol{B}$ in $\mathbb{R}^{N\times M}$. Furthermore, $\operatorname{tr}\big(\boldsymbol{A}^\top \boldsymbol{B}\big) = \operatorname{tr}\big(\boldsymbol{B}^\top \boldsymbol{A}\big)$. Therefore, for $\boldsymbol{v} \in \mathbb{R}^N$ and $\boldsymbol{u} \in \mathbb{R}^M$, we have

$$\begin{aligned}
\boldsymbol{v}^\top \boldsymbol{A}\boldsymbol{u} &= \operatorname{tr}\Big(\boldsymbol{v}^\top \boldsymbol{A}\boldsymbol{u}\Big) && \text{(because it is a scalar)} \\
&= \operatorname{tr}\Big(\boldsymbol{u}(\boldsymbol{v}^\top \boldsymbol{A})\Big) && \text{(from the trace property)} \\
&= \operatorname{tr}\Big((\boldsymbol{u}\boldsymbol{v}^\top)\boldsymbol{A}\Big) && \text{(associative property of matrix multiplication)} \\
&= \langle \boldsymbol{v}\boldsymbol{u}^\top, \boldsymbol{A} \rangle && \text{(interpretation as inner product).}
\end{aligned}$$

In summary $\boldsymbol{v}^\top \boldsymbol{A}\boldsymbol{u} = \langle \boldsymbol{v}\boldsymbol{u}^\top, \boldsymbol{A} \rangle$. The product $\boldsymbol{v}\boldsymbol{u}^\top$ can also be interpreted as a tensor product $\boldsymbol{v} \otimes \boldsymbol{u}$ (described later in this section). This way, we get $\boldsymbol{v}^\top \boldsymbol{A}\boldsymbol{u} = \langle \boldsymbol{v} \otimes \boldsymbol{u}, \boldsymbol{A} \rangle$.

Permanent If $\boldsymbol{A}$ is a square $N \times N$ matrix then the permanent of $\boldsymbol{A}$ is defined as determinant, but without the sign. More formally:

$$\operatorname{perm}(\boldsymbol{A}) = \sum_{\boldsymbol{\sigma} \in \mathcal{P}_N} \prod_{n=1}^{N} a_{n,\sigma_n},$$

where $\mathcal{P}_N$ is the set of all permutations of $[N]$.

The permanent is an invariant for matrices, used in combinatorial settings, where the problem is independent of the permutation of rows or columns [207].

Spark of a Matrix The *spark* of a matrix $\boldsymbol{A}$ is the smallest number of columns of $\boldsymbol{A}$ that are linearly-dependent [208].

Matrix Factorisation Matrix factorisation is its own field, but we mention only a few facts that will be needed in this thesis; all are based on [206].

First, any symmetric matrix $\boldsymbol{A}$ is diagonalisable, in other words, it can be written in the from $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^\top$ where $\boldsymbol{U}$ is a unitary matrix and $\boldsymbol{D}$ is diagonal matrix.

If $\boldsymbol{A}$ is positive definite, then all its eigenvalues are positive, and it can be decomposed using the Cholesky decomposition $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^\top$. Matrix $\boldsymbol{L}$ is unique up to unitary transformation, which describes rotations and translations [209]. Matrix $\boldsymbol{L}$ is related to $\boldsymbol{U}$ above by $\boldsymbol{L} = \boldsymbol{U}\sqrt{\boldsymbol{D}}$. If $\boldsymbol{A}$ is positive definite, then $\boldsymbol{L}$ is no longer a square matrix.

Finally, any matrix $\boldsymbol{A}$ can be decomposed with the singular value decomposition (SVD): $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$, where $\boldsymbol{S}$ is diagonal and $\boldsymbol{U}$ and $\boldsymbol{V}$ are possibly different unitary matrices. If all singular values of $\boldsymbol{A}$, i.e., all diagonal entries of $\boldsymbol{S}$ are different, then the SVD is unique up to the order of entries in $\boldsymbol{S}$. Usually they are sorted from the largest to the smallest, or from the largest in absolute value to the smallest in absolute value.

Polynomial-Matrix Factorisation We restate here the results from the work of Jacob van der Woude [210] because we find his writing clear. His work is based directly on the work of Lasha Ephremidze [211], whose work is in turn based on the older work in polynomial-matrix factorisation [212, 213].

Van der Woude [210] formulates complex and real-valued spectral factorisation theorems as follows:

Theorem 1.1: Theorem 1 in [210]

Given $\boldsymbol{A}(t) \in \mathbb{C}^{N\times N}[t]$ such that $\boldsymbol{A}(t) \succcurlyeq 0$, there exists a $\boldsymbol{B}(t) \in \mathbb{C}^{N\times N}[t]$ such that $\boldsymbol{A}(t) = \boldsymbol{B}^*(t)\boldsymbol{B}(t)$.

Corollary 1.2: Corollary 2 in [210]

Given $\boldsymbol{A}(t) \in \mathbb{R}^{N\times N}[t]$ such that $\boldsymbol{A}(t) \succcurlyeq 0$, there exists a $\boldsymbol{B}(t) \in \mathbb{R}^{2N\times N}[t]$ such that $\boldsymbol{A}(t) = \boldsymbol{B}^\top(t)\boldsymbol{B}(t)$.

Theorem 1.1 also applies to complex bandlimited functions as polynomials in $e^{i\frac{2\pi}{T}t}$. As Theorem 1.1 considers only positive powers of t, we have to multiply the polynomials by the factor $e^{i\frac{2K\pi}{T}t}$. As the absolute value of this factor is 1, it does not change the result. We can use similar reasoning to extend Corollary 1.2 to real-valued bandlimited functions.

Tensors and Tensor Product Tensors appear in different disciplines, e.g., in linear algebra, data science, and physic. Each discipline introduces its own perspective and sometimes its own formalism. Here we give a minimal introduction, needed for Chapter 3.

We define the tensor outer product[8], tensor spaces and tensors by describing their properties. This approach might seem a bit abstract, but in our opinion it is the simplest way.

Consider linear spaces $\mathcal{V}$ and $\mathcal{W}$, vectors $\boldsymbol{v}, \boldsymbol{v} \in \mathcal{V}$, and vector $\boldsymbol{w} \in \mathcal{W}$. The tensor product $\otimes$ an operation that linear in each of the elements, i.e.:

$$\begin{aligned}(\boldsymbol{v} + \boldsymbol{u}) \otimes \boldsymbol{w} &= \boldsymbol{v} \otimes \boldsymbol{w} + \boldsymbol{u} \otimes \boldsymbol{w} \\ (\alpha \boldsymbol{v}) \otimes \boldsymbol{w} &= \alpha(\boldsymbol{v} \otimes \boldsymbol{w}),\end{aligned} \tag{1.7}$$

and the same for the second element $\boldsymbol{w}$. Of course, there are many such bi-linear operations, for example the inner product. We choose the tensor product to be the most general bi-linear operation; we do not assume any of its properties, apart from what is implied by linearity by bi-linearity[9].

Now, we can define the tensor product of spaces $\mathcal{V}$ and $\mathcal{W}$ as a space of tensor products of elements of $\mathcal{V}$ and $\mathcal{W}$:

$$\mathcal{V} \otimes \mathcal{W} = \{\boldsymbol{v} \otimes \boldsymbol{w} : \boldsymbol{v} \in \mathcal{V}, \boldsymbol{w} \in \mathcal{W}\}.$$

It can be shown that the product space with induced operations is also a linear space.

Equivalently, we can define the tensor product of spaces as the set of all pairs of elements form spaces $\mathcal{V}$ and $\mathcal{W}$, where the pairs that result in the same tensor product are equal. More formally, the product space is the quotient set of equivalence relation $\sim_\otimes$:

$$\mathcal{V} \otimes \mathcal{W} = \{(\boldsymbol{v}, \boldsymbol{w}) : \boldsymbol{v} \in \mathcal{V}, \boldsymbol{w} \in \mathcal{W}\}_{/\sim_\otimes},$$

where $(\boldsymbol{v}_1, \boldsymbol{w}_1) \sim_\otimes (\boldsymbol{v}_2, \boldsymbol{w}_2)$ if and only if $\boldsymbol{v}_1 \otimes \boldsymbol{w}_1 = \boldsymbol{v}_2 \otimes \boldsymbol{w}_2$, as given by Equation (1.7).

We can make this definition more tangible by seeing how the tensor product behaves on basis coefficients. In product space $\mathcal{V} \otimes \mathcal{W}$ we consider the basis that consists of pairs of elements of the bases of $\mathcal{V}$ and $\mathcal{W}$:

$$\begin{array}{cccc} (\boldsymbol{e}_1^{(v)}, \boldsymbol{e}_1^{(w)}) & (\boldsymbol{e}_1^{(v)}, \boldsymbol{e}_2^{(w)}) & (\boldsymbol{e}_1^{(v)}, \boldsymbol{e}_3^{(w)}) & \cdots \\ (\boldsymbol{e}_2^{(v)}, \boldsymbol{e}_1^{(w)}) & (\boldsymbol{e}_2^{(v)}, \boldsymbol{e}_2^{(w)}) & (\boldsymbol{e}_2^{(v)}, \boldsymbol{e}_3^{(w)}) & \cdots \\ (\boldsymbol{e}_3^{(v)}, \boldsymbol{e}_1^{(w)}) & (\boldsymbol{e}_3^{(v)}, \boldsymbol{e}_2^{(w)}) & (\boldsymbol{e}_3^{(v)}, \boldsymbol{e}_3^{(w)}) & \cdots, \\ \vdots & \vdots & \vdots & \ddots \end{array}$$

where $\boldsymbol{e}_j^{(v)}$ are elements of the basis of $\mathcal{V}$ and $\boldsymbol{e}_j^{(w)}$ are elements of the basis of $\mathcal{W}$. None of these pairs are equivalent under $\sim_\otimes$ because they are linearly independent. We purposefully write the basis on the grid to illustrate that coefficients of a tensor in this basis form a matrix. Moreover, they are the outer product of coefficients of vectors $\boldsymbol{v}$ and $\boldsymbol{w}$: $\boldsymbol{v}\boldsymbol{w}^\top$[10].

As the tensor product of linear spaces is a linear space, we can consider its product with another linear space, e.g., $\mathcal{U} \otimes \mathcal{V} \otimes \mathcal{W}$, without changing the definition.

---

[8] In case of vectors it is also called outer product.

[9] For example, bi-linearity implies that $(\alpha \boldsymbol{v}) \otimes \boldsymbol{w} = \boldsymbol{v} \otimes (\alpha \boldsymbol{w})$.

[10] Here we slightly abuse the notation by identifying a vector with its coefficient in the basis

All the objects that live in tensor products of spaces are called tensors. The order of the tensor describes how many components its product space has. Tensors in $\mathcal{V} \otimes \mathcal{W}$ are second order tensors, and as we have seen they can be identified with a matrix (see [206] for further discussion). Tensors in $\mathcal{U} \times \mathcal{V} \otimes \mathcal{W}$ are third order tensors, and can be identified with a cube of coefficients in the basis induced by bases in $\mathcal{U}, \mathcal{V}$ and $\mathcal{W}$. By convention, we say that first order tensors are vectors, and zeroth-order tensors are scalars.

Tensors can also be understood as mappings between vector spaces [214]; an order-3 tensor can be interpreted as a mapping from the space of vectors to the space of matrices in as an $N \times M$ matrix can be interpreted as a linear mapping from $\mathbb{R}^M$ to $\mathbb{R}^N$.

Furthermore, in the same way an $N \times M$ matrix can be interpreted as a bilinear operator, mapping pairs of vectors in $\mathbb{R}^N$ and $\mathbb{R}^M$ to a scalar, then an order-3 tensor is a multi-linear mapping, from triplets of vectors to a scalar.

Related to tensor product is the Kronecker product, which we denote $\otimes_K$ to avoid confusion. Consider two matrices $\boldsymbol{A} \in \mathbb{R}^{N \times M}$ and $\boldsymbol{B} \in \mathbb{R}^{J \times K}$ and vectors $\boldsymbol{v} \in \mathbb{R}^M$ and $\boldsymbol{w} \in \mathbb{R}^K$. The Kronecker product is defined to satisfy the following:

$$(\boldsymbol{A} \otimes_K \boldsymbol{B}) \operatorname{vec}(\boldsymbol{v} \otimes \boldsymbol{w}) = \operatorname{vec}((\boldsymbol{A} \otimes \boldsymbol{B})(\boldsymbol{v} \otimes \boldsymbol{w})) = \operatorname{vec}(\boldsymbol{A}\boldsymbol{v} \otimes \boldsymbol{B}\boldsymbol{w}).$$

Measure Zero

Measure theory is its own field. Here, we focus only on Lebesgue measure [215]. Lebesgue measure is a generalisation of length (volume) to multiple dimensions and complicated sets. Here, we need to know only what it means that a set has measure zero.

A Lebesgue measure μ of an interval is simply its length: $\mu((a, b)) = b - a$ for $a > b$. A Lebesgue measure of an N dimensional cube $\mathcal{C}$ is a product of measures of the dimensions of the cube, i.e.,

$$\mu(\mathcal{I}_1 \times \mathcal{I}_2 \times \cdots \times \mathcal{I}_N) = \mu(\mathcal{I}_1)\mu(\mathcal{I}_2) \cdots \mu(\mathcal{I}_N)$$

A subset $\mathcal{S}$ of $\mathbb{R}^N$ is measure zero, if for every $\epsilon > 0$ there exists (at most) countable sequence of open cubes $\mathcal{C}_1, \mathcal{C}_2, \ldots$ such that

$$\mathcal{S} \subseteq \bigcup_{n \in \mathbb{N}} \mathcal{C}_n \quad \text{and} \quad \sum_{n \in \mathbb{N}} \mu(\mathcal{C}_n) < \epsilon.$$

Chapter 2.

Sampling at Unknown Locations

Sampling at unknown locations is an inherently ill-posed problem. For every N sample values we measure, we have $2N$ unknowns: N sample values and N locations. Let us illustrate it on the example of structure from texture. We take a picture of a texture on some surface, and would like recover the texture and the shape of the textured surface. Without any priors, to recover the texture we need to know the depth of the pixels, and to recover the depth, we need to know the texture, see Figure 2.1.

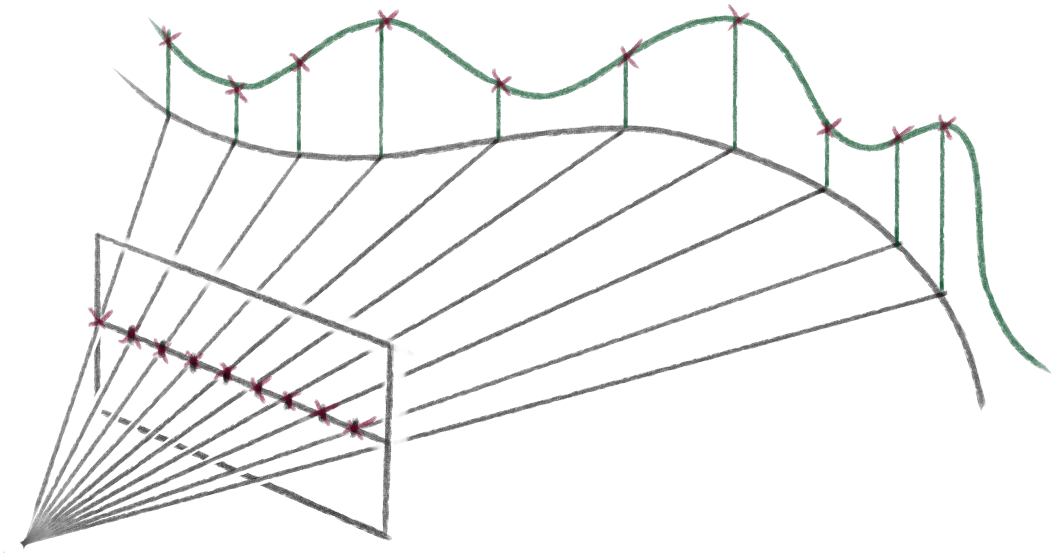


Figure 2.1.: Surface retrieval as a sampling at unknown locations. The samples are taken uniformly at the camera plane, but as the surface is not known, neither the 3D sample position nor the distance along the surface is known.

Contributions: The work in this chapter is joint work of the author of this thesis (MP) with Golnoosh Elhami (GE), Benjamín Béjar Haro (BH), Martin Vetterli (MV), Adam Scholefield (AS) [216–218] and Damian Orlef (DO). MP conducted research on the non-uniqueness and recovery guarantees for polynomial case and Lemma 2.4 for the bandlimited case. GE conducted research on bandlimited case. BH, MV and AS designed and advised research. DO provided necessary hints to prove Lemma 2.4. MP wrote this chapter, taking liberty to rearrange and rename some of the results.

The code used for simulations is available at `github.com/LCAV/surface-reconstruction`.

As discussed in Chapter 1, with correct priors, SfM is a successful tool for surface recovery. Therefore, given the correct constraints, it is possible to recover the measurement positions and the underlying function from samples at unknown locations.

To find the such constraints, we study the kinds of non-uniqueness that arise in sampling at unknown locations. We show for polynomial and bandlimited signals, it is possible to find a valid solution arbitrarily far from the original signal. As a regularization we propose to constrain the sample positions by another function and we show that it is equivalent to sampling at known locations of a composite of functions.

We would like to emphasize that we propose sampling a composite of functions due to its previously unseen practical relevance for SfM. However, we do not analyse the full SfM setups. The algorithms we propose are not in anyway intended to be practical algorithms that compete with the state of the art in these fields. We study, instead, two simple incarnations of a constrained sampling at unknown locations:

1. periodic bandlimited signals with sample positions given by an unknown linear warping;
2. polynomial signals with sample positions constrained by a rational function.

In both cases we provide recovery guarantees in the noiseless case. We believe that these two incarnations provide a first step towards a deeper theoretical understanding of the more complex SLAM and SfM problems.

We also provide an a toy example of surface retrieval. We consider a textured plane that is sampled by a camera. We assume that the texture is parametric: polynomial or bandlimited. The sample locations are unknown, but they are related to pixel locations by the camera projection.

This chapter is divided in two main parts. In the first, we focus on theory (Section 2.1) and, in the second, we provide an application (Section 2.2). In the first part, we state the problem formally, show why it is difficult (Section 2.1.1), and then state and prove our uniqueness results (Section 2.1.2). In Section 2.2.1, we show how to use our results in practice and provide recovery algorithms. In Section 2.2.2, we show the results of our numerical experiments. Finally, in Section 2.3, we discuss the potential directions of future work.

2.1. Theory

The purpose of sampling at unknown locations is to recover a function from its sample values, when we do not know the positions where, or times when, the samples were taken. More generally, we could consider sampling when we have only partial knowledge of operators. Intuitively, this problem is ill-posed and has many solutions. In this work, we introduce a number of restrictions in order make the solution unique. The first and the weakest restriction we consider is fixing the order of samples. This leads to the following formulation:

Problem 2.1: Sampling at Unknown Locations

Consider a linear, K-dimensional space $\mathcal{F}$ of real or complex valued functions defined on a closet interval $\mathcal{I} \subseteq \mathbb{R}$[a]:

$$\mathcal{F} \subset \mathbb{F}^{\mathcal{I}},\ \mathbb{F} \in \{\mathbb{R}, \mathbb{C}\},\ \dim(\mathcal{F}) = K$$

We sample an unknown function $f \in \mathcal{F}$ at N unknown, but distinct points t_n such that $t_1 < \cdots < t_N \in \mathcal{I}$ and obtain measurements

$$y_n = f(t_n) \text{ for } n = 1, \ldots, N. \tag{2.1}$$

We call a *solution* to the problem of sampling at unknown location any function $\widetilde{f} \in \mathcal{F}$ for which there exist points $\widetilde{t}_1 < \cdots < \widetilde{t}_N \in \mathcal{I}$, such that

$$y_n = \widetilde{f}(\widetilde{t}_n) \text{ for } n = 1, \ldots, N. \tag{2.2}$$

---

[a] For the definitions of the function spaces see Section 1.5.

We consider, in this general formulation, a general linear space of real- or complex-valued functions of one variable. Some results in this chapter are restricted to some specific classes of functions: in particular, polynomials, bandlimited functions, and real-valued functions.

2.1.1. Intuitions and Challenges

As mentioned in the Introduction, the problem of sampling at unknown locations does not have a unique solution. Here, we present a number of examples illustrating different aspects of this problem. Although, in general, we study polynomials and band-limited functions, we begin with a very broad class of real-valued functions that we refer to as "reasonable" functions:

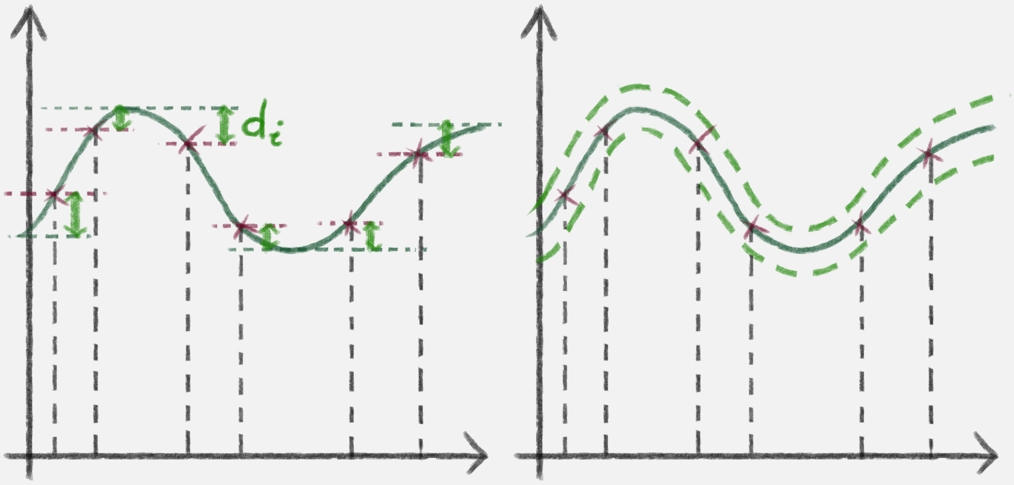
Definition 2.1: Reasonable Function

We call *a reasonable function* a continuous real-valued function that is defined on a finite interval $\mathcal{I}$ and that is constant or has a finite number of extrema.

These functions are reasonable in the sense that, in the real world, it is extremely rare to encounter a function with an infinite number of extrema. Furthermore, by discrete measurements alone, we cannot tell if a function is continuous or not. Of course, both polynomials and bandlimited periodic functions have this property. For more information about functions that are not reasonable, see Box 2.4 at the end of this section.

In Example 2.1 we show that if the function is reasonable, then there is a high chance that we can perturb it so that it will still be a solution.

Example 2.1: A ball of solutions, inspired by Browning [92]



Assume that $\mathcal{F}$ is a family of reasonable functions and that none of samples of f is taken at a local extremum. Since f is continuous, every t_n lies between a local minimum and local maximum — including the extrema at the ends of the interval. Let $t_{n,min}$ and $t_{n,max}$ be the positions of these extrema. Let r_n be the vertical distance to the closest extremum $r_n := \min(f(t_n) - f(t_{n,min}), f(t_{n,max}) - f(t_n))$ and let $r = \min_i r_n$. This means that r is the closest vertical distance between a sample value and a neighbouring extremum, see the plots in this box.

Then, for any function $g \in \mathcal{F}$ such that $|g(t)| < r$ for $t \in \mathcal{I}$, $\widetilde{f} = f + g$ is a solution. Indeed, for every n, $f(t_n) \leq f(t_{n,max}) - r < f(t_{n,max}) + g(t_{n,max}) = f_s(t_{n,max})$ and similarly $f(t_n) > f_s(t_{n,min})$. From the Darboux theorem, we know that there is a point $\widetilde{t}_i$ between $t_{n,min}$ and $t_{n,max}$ such that $\widetilde{f}(\widetilde{t}_n) = f(t_n)$.

When there are two or more points between consecutive extrema, we have to choose new samples in order for the sample order to be preserved.

Finally, observe that the set $\mathcal{G}$ of functions g such that $|g(t)| < r$ is a ball in $\mathcal{L}_\infty$ norm.[a]

---

[a] Unless we restrict $\mathcal{F}$ to be a finite dimensional space, we will not be able to fit any $\mathcal{L}_2$ balls in $\mathcal{G}$ even though it is an open set.

Although obtaining samples exactly at local extrema is quite improbable, taking samples close to extrema can be achieved by taking many samples. Unfortunately, we can find multiple solutions no matter the number of samples, in both the polynomial and bandlimited cases, see Examples 2.2 and 2.3.

Example 2.2: Shifting in Time

Let $\mathcal{F}$ be the space of polynomials of a finite degree or a space of periodic bandlimited functions on an interval $(0, T)$. Then, for every $\tau \in [-t_1, T - t_N]$, the function $\widetilde{f}_\tau(t) = f(t + \tau)$ is a solution.

Example 2.3: Scaling in Time

Let $\mathcal{F}$ be the space of polynomials of a finite degree on an interval $(0, T)$. For every $\alpha \in [0, T/t_N]$, the function $\widetilde{f}_\alpha(t) = f(\alpha t)$ is a solution. This is not the case for bandlimited functions, as scaling in time changes the support in the Fourier domain.

Note that, in the previous two examples, we do not require the functions to be real-valued.

To avoid the problems shown in Examples 2.2 and 2.3, we require any solution to be equal to f at the ends of $\mathcal{I}$. We call such a solution a solution *with fixed ends*. Henceforth, we will use $\mathcal{I} = [0, T]$ to explicitly label the ends of $\mathcal{I}$.

Unfortunately, even with these assumptions, the solution can be arbitrarily far from the original function. The lemma below characterises how such a solution might look. Here, we present it, together with its proof, because the proof is short and gives some useful intuitions.

Lemma 2.1: Arbitrarily Distant Solutions

Let $\mathcal{F}$ be a finite dimensional linear space of reasonable functions defined on the interval $[0, T]$, let $f \in \mathcal{F}$ be a non-constant function, and let $t_{max,1}, t_{max,2} \dots$ and $t_{min,1}, t_{min,2}, \dots$ be the arguments for which this function attains its local maxima and minima, respectively.

If there exists a non-constant $g \in F$ such that $g(0) = 0$, $g(T) = 0$, $g(t_{min,i}) \leq 0$ and $g(t_{max,i}) \geq 0$ for every i, then for every $M > 0$ there exists α such that $f_M = f + \alpha g$ is a solution with fixed ends and $||f_M - f|| > M$.

In Lemma 2.1, we assume that f is not constant, because a large number of constant samples of a reasonable function f could uniquely identify f as constant; for example, when $\mathcal{F}$ is the family of polynomials. However, in this case we are not able to recover sample positions.

Proof of Lemma 2.1

The idea behind the proof is that, when we add the function αg to a function f, we can move outwards the points that are near extrema, see Figure 2.2.

Consider the interval $[a, b]$ between two consecutive extrema. Without loss of generality, we can assume that a is the maximum and b is the minimum. According to assumptions of Lemma 2.1 $g(a) \geq 0$. As a result, $f(a) + \alpha g(a) \geq f(a)$. Similarly, $f(b) + \alpha g(b) \leq f(b)$. Hence, as those functions are continuous, for any sample position t_n in the interval $[a, b]$, we can find a new point $\widetilde{t}_n \in [a, b]$ such that $f(\widetilde{t}_n) + \alpha g(\widetilde{t}_n) = f(t_n)$. The continuity

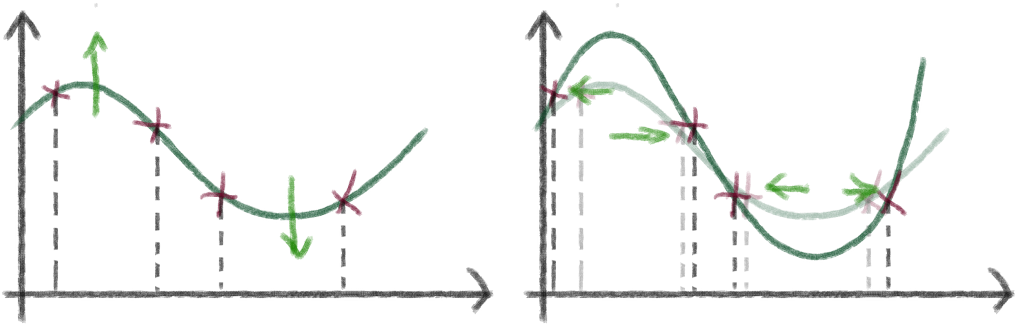


Figure 2.2.: Sample movement when extrema are amplified. When the value at maximum, the samples move outwards, to the left. There is always space to move, because the value at next minimum does not decrease.

of those functions also makes it possible to preserve the order of the samples on this interval. A similar argument can be used on intervals between minima and maxima and between an extremum and the boundary. Therefore, we can find new samples in the same order. ■

Of course, to show that functions in $\mathcal{F}$ cannot be uniquely recovered, we have yet to find such a function $g \in \mathcal{F}$. In Lemma 2.2, we show how to do this for real-valued polynomials (Equation (1.2)) and for real-valued bandlimited functions on the interval $[0, T]$ (Equation (1.3)).

Lemma 2.2: Lemma 2.1 for Polynomials and Bandlimited Functions

Let $\mathcal{I} = [0, T]$. If

1. $\mathcal{F}$ is the class of real valued polynomials of degree at most $K - 1$, or
2. $\mathcal{F}$ is the class of real-valued, K-bandlimited functions,

then for any $M > 0$ there exist a solution f_M with fixed ends such that $\|f - f_M\| \geq M$.

To prove Lemma 2.2, we only need to find a function g from Lemma 2.1. To find such functions, we use the fact that for polynomials of degree at most $K - 1$ or real-valued K-bandlimited functions the number of extrema *inside* interval $\mathcal{I}$ does not exceed the maximum number of zeros of a (non-zero) function in $\mathcal{F}$. We construct a function g that has a single zero between any pair of extrema of f inside $\mathcal{I}$, hence it changes sign between extrema of f. We then need to make sure that the sign is correct, at each extrema, including the ends of interval.

Proof of Lemma 2.2

A polynomial of degree at most $K - 1$ has at most $K - 2$ extrema inside the interval $\mathcal{I}$, but it can have $K - 1$ zeros. For bandlimited functions, both the number of zeros and and extrema are $K-1$, because the derivative of a bandlimited function is also a bandlimited function with *the same* bandwidth.

For every pair of consecutive extrema of f inside of $\mathcal{I}$, we choose a point τ_l separating them. We have in total $L_p \leq K-3$ points τ_l for a polynomial f and $L_b \leq K-2$ for a bandlimited function. For polynomials, we can now define

$$h_p(t) := \prod_{l=1}^{L_p}(t-\tau_l).$$

A polynomial h_p at most has $K-3$ zeros. Consider a polynomial $g(t) = \pm h_p(t)t(T-t)$. It has at most $K-1$ zeros, hence $g \in \mathcal{F}$. Moreover, as it changes the sign between consecutive extrema, we can choose the sign ($\pm$) so that g satisfies Lemma 2.1.

The bandlimited case is slightly more tricky, because there is no simple replacement for $(t-\tau_l)$. We could use $e^{it} - e^{i\tau_l}$, but it is not real valued, hence we do not control the sign. Using $\cos(t) - \cos(\tau_l)$ could solve it, but it "uses up" zeros twice too fast. Finally, we could use factors of the form $1 + a\cos(t) + b\sin(t)$ and solve equations for a and b to fit two τ_l at a time. But we can also find g by solving for a function that satisfies the following $K-1$ constraints:

$$\begin{aligned} g(\tau_k) &= 0 \text{ for } k \in [K-1] \\ g(0) &= 0; \end{aligned}$$

these are effectively linear equations on coefficients a_k and b_k that define g up to scaling and a sign that we choose in order for g to satisfy Lemma 2.1. ■

Although this result seems discouraging, it actually gives us some hope that the solution might be unique in specific cases. First, as the sign of g has to be specified at n points, it imposes some restrictions on the possible directions of change. For example, $-g$ is not a good direction of change, nor is any $\widetilde{g}$ that does not match the sign at at least one extremum, hence we could expect that there are more g that do not satisfy Lemma 2.1. Furthermore, the lemma gives us a path of solutions, $f = \alpha g$ for $\alpha \in \mathbb{R}_+$. This, in turn, defines a trajectory for the sample positions, see Figure 2.3. If the samples can move freely, they will adjust to any function on the path. However, if we restrict the way samples can move, our intuition is that there is a high probability that at least one of the samples will not lie on the trajectory defined by the path. Therefore, adding any constraint will remove at least *some* of the large-scale ambiguity.[1]. This observation is the reason we regularize the inverse problem by adding a constraint on the allowed sample trajectories; we describe this in the next section.

---

[1] Note, that Lemma 2.1 is likely not the only way to find a solution arbitrarily far from the true f. As a result, we might not be able to remove all ambiguity.

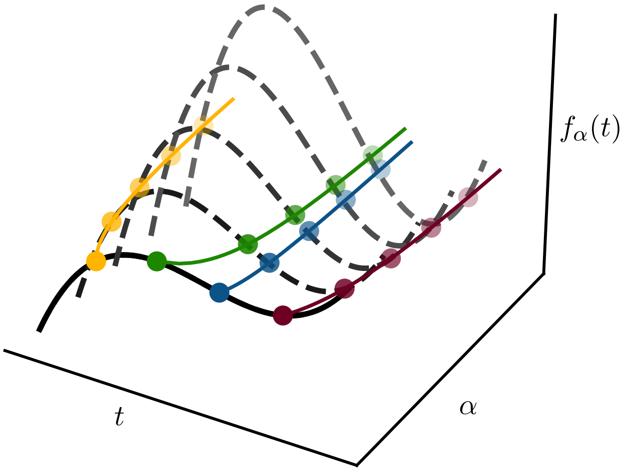
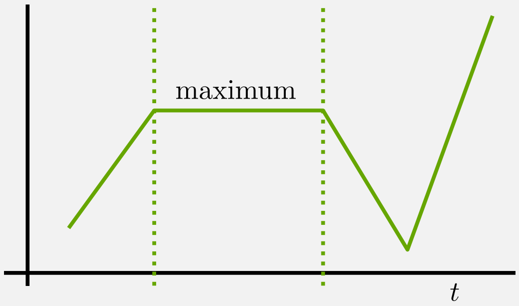
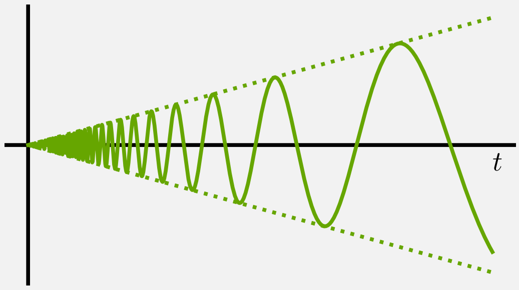


Figure 2.3.: Illustration of the path of solutions generated by moving along $g \in \mathcal{F}$ from the initial f. The path of solutions, $f_\alpha = f + \alpha g$ (dashed lines), defines a trajectories along which samples can move (solid lines).

Want to know more? 2.4: Unreasonable Functions

The simplest example of an “unreasonable” function is a piece-wise linear function that has constant maximum value on an interval (see the left plot below). This function has an uncountable number of extrema. But is it possible for continuous function to have *countable* number of extrema on a finite interval? We show such function on the right.





The function of the right is given by the following equation:

$$f(t) = \begin{cases} t\sin\left(\frac{1}{t}\right) & \text{for } t \in (0, T] \\ 0 & \text{for } t = 0 \end{cases}$$

This function has an countably infinite number of extrema at $t = \frac{2}{\pi + k\pi}$ for $k \in \mathbb{N}$, $k > \frac{1}{T\pi}$. The only point where the continuity of f can be questioned is $t = 0$. Consider the right limit of f at this point. Since $|f(t)| \leq t$ we have $\lim_{t \to 0^+} f(t) = 0 = f(0)$ and f is continuous at $t = 0$.

2.1.2. Theorems and Proofs

In this section, we present our results based on restricting sample positions. We explain how restricting sampling positions can be interpreted as the sampling of composite functions. Here, we focus on polynomials and bandlimited functions, and we do not require them to be real-valued. For bandlimited functions, we consider a linear constraint. For polynomials, such a constraint does not work, hence we consider a more complex one.

Inspired by Figure 2.3, we propose to restrict possible sample positions $\widetilde{t}_n$, $n \in [N]$ to only those that satisfy

$$\widetilde{t}_n = \psi_{\boldsymbol{\theta}}(t_n) \text{ for } n \in [N],$$

where ψ_θ belongs to a known family of functions Ψ that are parameterized by vectors of parameters $\boldsymbol{\theta}$ see Figure 2.4 (a).

An alternative way to constrain how samples move is to consider the sampling of a composite of functions at known locations, see Figure 2.4 (b). We consider a warping function φ and assume that the sample positions are $t_n = \varphi_{\boldsymbol{\beta}}(s_n)$ where s_n are known and the vector of parameters $\boldsymbol{\beta}$ is not. This way, we obtain samples

$$y_n = (f \circ \varphi_{\boldsymbol{\beta}})(s_n) \text{ for } n \in [N].$$

To maintain the order of samples, we restrict $\varphi_{\boldsymbol{\beta}}$ to be a monotonically increasing function. It follows that $\varphi_{\boldsymbol{\beta}}$ is invertible and, for a given warping, we can define constraint function $\psi_{\boldsymbol{\beta},\widetilde{\boldsymbol{\beta}}} := \varphi_{\widetilde{\boldsymbol{\beta}}} \circ \varphi_{\boldsymbol{\beta}}^{-1}$.

Therefore, by constraining the sampling positions, we changed the problem from sampling at arbitrary unknown locations to sampling a composite of functions at known locations. We often think about $\varphi_{\boldsymbol{\beta}}$ as a warping of f. Hence, we sample a warped version of f and seek to recover both f and the parameters $\boldsymbol{\beta}$.

Formally, let $f \in \mathcal{F}$ be the signal of interest, let s_n be the sample positions and let $\varphi_{\boldsymbol{\theta}}$ be a warping function. The problem to solve is

$$\begin{array}{ll} \text{find} & \{f \in \mathcal{F}, \boldsymbol{\beta} \in \mathcal{B}\} \\ \text{s.t.} & y_n = (f \circ \varphi_{\boldsymbol{\beta}})(s_n), n \in [N] \end{array} \tag{2.3}$$

Polynomials

We first consider the family of polynomials, of a degree at most $K-1$, on a real line (Equation (1.2)). We can consider real or complex coefficients. We show that, if the positions are constrained by a rational function, then the solution is unique with $O(K)$ samples, with constants dependent on the degree of the rational function. More precisely,

Lemma 2.3: Uniqueness of a Polynomial Signal; Lemma 3 in [217]

Let $\mathcal{F}$ be the space of polynomials of degree at most $K-1$. Let $\widetilde{t}_n$ be possible sampling positions satisfying the constraint $\psi_{\boldsymbol{\theta}}$, i.e. $\widetilde{t}_n = \psi_{\boldsymbol{\theta}}(t_n)$. Furthermore, let

$$\psi_{\boldsymbol{\theta}}(t_n) = \frac{p_{\boldsymbol{\theta}}(t_n)}{q_{\boldsymbol{\theta}}(t_n)} \quad \text{for all } n \in [0 \ldots N-1], \tag{2.4}$$

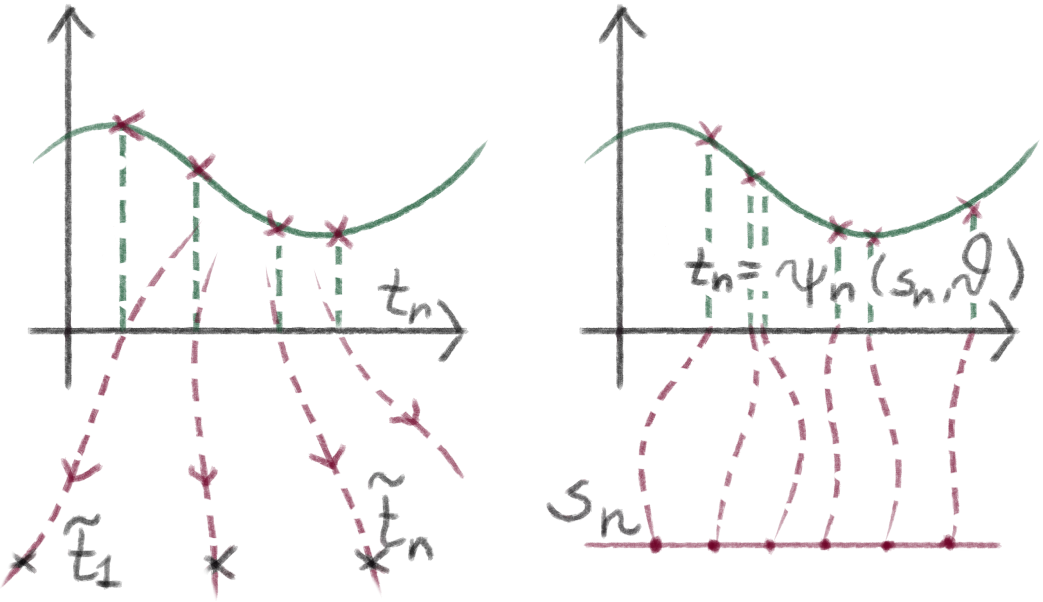


Figure 2.4.: Different views on restricted sample movement. On the left, samples can move along some trajectories parameterized by θ (in this case θ is a scalar), and if these trajectories are different from those in Figure 2.3, then we can hope for recovery conditions. On the right, sample positions depend on some known variable s_n via an unknown warping.

be a proper rational function, which is where $p_{\boldsymbol{\theta}}$ and $q_{\boldsymbol{\theta}}$ are irreducible polynomials with degrees satisfying $\deg(p_{\boldsymbol{\theta}}) \leq \deg(q_{\boldsymbol{\theta}})$. If the number of samples $N > (K-1)(deg(q_{\boldsymbol{\theta}})+1)$, then function f and samples t_n are a unique pair satisfying both Equations (2.2) and (2.4).

If this lemma confuses the reader, they should not worry, in Section 2.2, we will see how to use it in practice, with $\deg(q_{\boldsymbol{\theta}}) = \deg(p_{\boldsymbol{\theta}}) = 1$. In fact, the case described by Corollary 2.5 in Section 2.2 is how we found this lemma. When we proved this specific case, we noticed that proof can be easily extended to $\deg(q_{\boldsymbol{\theta}}), \deg(p_{\boldsymbol{\theta}}) \geq 1$ as long as $\deg(p_{\boldsymbol{\theta}}) \leq \deg(q_{\boldsymbol{\theta}})$.

Proof of Lemma 2.3

Let $g \in \mathcal{F}$ be a polynomial such that

$$g(\widetilde{t_n}) = g\left(\frac{p_{\boldsymbol{\theta}}(t_n)}{q_{\boldsymbol{\theta}}(t_n)}\right) = f(t_n)$$

for every $i = 1, \dots, m$. All t_n satisfy the following equation:

$$\sum_{k=0}^{K-1} a_k t^k = \sum_{k=0}^{K-1} b_k \left(\frac{p_{\boldsymbol{\theta}}(t)}{q_{\boldsymbol{\theta}}(t)}\right)^k, \tag{2.5}$$

where a_k and b_k, $k = 1, \dots, K-1$ are the coefficients of the polynomials f and g, respectively. We can rewrite this as

$$(q_{\boldsymbol{\theta}}(t))^{K-1} \sum_{k=0}^{K-1} a_k t^k = \sum_{k=0}^{K-1} b_k (p_{\boldsymbol{\theta}}(t))^k (q_{\boldsymbol{\theta}}(t))^{K-1-k}. \tag{2.6}$$

This equation defines a polynomial with degree at most $\nu = \max(\deg(q_{\boldsymbol{\theta}})+1, \deg(p_{\boldsymbol{\theta}}))(K-1)$. But, since $\deg(q_{\boldsymbol{\theta}}) \geq \deg(p_{\boldsymbol{\theta}})$, then $\nu = (\deg(q_{\boldsymbol{\theta}}) + 1)(K-1)$.

If the degree of f is greater than 1, the left-hand side of Equation (2.6) cannot be equal to the right-hand side everywhere. Therefore, Equation (2.5) has at most ν solutions, hence the polynomial f is unique provided that $m > (\deg(q_{\boldsymbol{\theta}}) + 1)(K-1)$.

If f is constant it is possible that both sides of Equation (2.6) are equal everywhere but this can only occur if $f \equiv g$. ■

It is important that $\psi_{\boldsymbol{\theta}}$ is proper because, for example, for $\deg(p_{\boldsymbol{\theta}}) = 1$ and $\deg(q_{\boldsymbol{\theta}}) = 0$, the constraint $\psi_{\boldsymbol{\theta}}$ becomes simply a linear scaling. Yet, as we know from Example 2.3 in this case, the unique recovery is not possible.

Bandlimited Functions

The second class of functions we consider are T-periodic bandlimited functions. As we do not require them to be real valued, we use a formulation from Equation (1.5), and coefficient a_k are considered complex. Note that we define $\mathcal{F}$ to be a space of T-periodic functions, but we could also think about it as a function on an interval $[0, T]$. The periodic iterations is simpler, because we will scale time t and this could mean taking samples outside $[0, T]$.

With bandlimited functions we consider linear warping,

$$\varphi_{\boldsymbol{\beta}}(s) = \beta s.$$

That also leads to linear constraint $\varphi(t) = \widetilde{\beta}/\beta s$. Hence, we can write the signal as a composite of (unknown) linear scaling with a bandlimited function:

$$f(\varphi_{\boldsymbol{\beta}}(s)) = f(\beta s) = \sum_{k=-(K-1)/2}^{(K-1)/2} a_k e^{i\beta \frac{2\pi}{T} sk}.$$

In the Fourier domain, f consists of a finite number of frequencies:

$$\widehat{f}(\omega) = \sum_{k=-(K-1)/2}^{(K-1)/2} a_k \delta(\omega - \beta \frac{2\pi}{T} k),$$

where FT denotes a Fourier transform and δ is a Dirac delta. If the reader is curious about this creature, we refer them to Box 2.5. To estimate the spectrum, we use Prony's method, see Box 2.7. For now however, it suffices to know that spectral estimation is much simpler when samples t_n are taken uniformly. For some musing on the non-uniform spectral estimation methods, we refer the reader to Section 2.3.

In summary, we assume that composite $f(\varphi_{\boldsymbol{\beta}})$ is sampled uniformly. N uniform samples on the interval $(0, T]$ will be at positions $t_n = nT/N$, hence we obtain

$$y_n = \sum_{k=-(K-1)/2}^{(K-1)/2} a_k e^{i\beta 2\pi nk/N}.$$

As we have chosen the uniform samples, we can calculate the Discrete Time Fourier Transform (DTFT) of y_n:

$$\widehat{y}(\omega) = \sum_{k=-(K-1)/2}^{(K-1)/2} a_k \delta(e^{i\omega} - e^{i\beta 2\pi k/N}).$$

If two different composites — for different βs — have the same set of frequencies on the interval $\omega \in 0, 2\pi$, they have the same frequencies overall and are indistinguishable. Therefore, we consider for which $\alpha = \beta/N$ such an overlap is possible. We begin with the following definition:

Definition 2.2: Equivalence

Let S_α be the set of frequencies of a warped signal $S_\alpha = \{e^{j2\pi k\alpha} \,|\, k \in \{-(K-1)/2, \ldots, K-1)/2\}\}$. We say that α and $\widetilde{\alpha}$ are *equivalent*, if $\alpha \neq \widetilde{\alpha}$ but $S_\alpha = S_{\widetilde{\alpha}}$,

Clearly, α and $\widetilde{\alpha} = \pm\alpha + k$ are equivalent for $k \in \mathbb{N}$. As a result, we are interested only in $\alpha \in [0, 0.5]$. Surprisingly, even on this interval, there are equivalent αs. However, there are very few of them and they can be characterised by the following lemma:

Lemma 2.4: Uniqueness of a Bandlimited Signal

Let $\alpha \in [0, 0.5]$. Then, there exist $\widetilde{\alpha} \in [0, 5]$ equivalent to α if and only if α is rational $\alpha = p/Q$ for $p, Q \in \mathbb{N}$, such that $Q \leq K + 1$ with p and Q co-prime. Moreover, all equivalent $\widetilde{\alpha}$ are also of the form $\widetilde{p}/Q$, where $\widetilde{p}$ and Q are co-prime.

The proof of this lemma has some elegant bits from group theory. Before we prove the lemma, we illustrate how the equivalent classes look in Figure 2.5.

Proof of Lemma 2.4

Assume that $\alpha, \widetilde{\alpha}$ are equivalent, $\alpha, \widetilde{\alpha} \in [0, 0.5]$ and thus $S_\alpha = S_{\widetilde{\alpha}}$. It follows that $e^{j2\pi\alpha} \in S_{\widetilde{\alpha}}$ and there exists $m \in \{-(K-1)/2, \ldots, (K-1)/2\}$ such that $\alpha = m\widetilde{\alpha} \mod 1$.

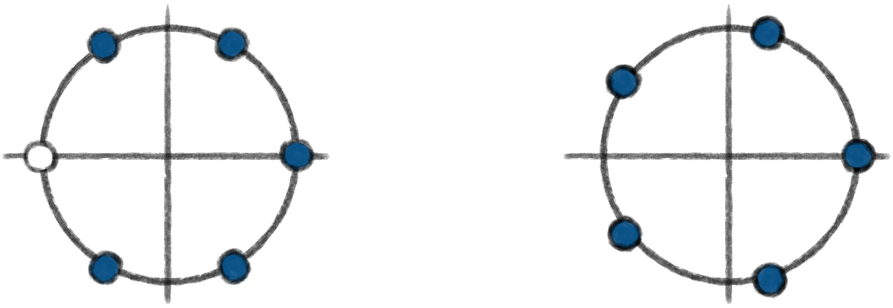


Figure 2.5.: Two different types of equivalence classes. Left: $Q = K + 1$, there are $K + 1$ slots, and K of them are covered by the frequencies present in the signal. Curiously, the slot at $z = -1$ is always empty, see the proof of Lemma 2.4. Right: $Q \leq K$, there are Q slots, and they are all covered by the frequencies of the signal.

Similarly, we can write $\widetilde{\alpha} = \widetilde{m}\alpha \mod 1$. This leads to $\alpha = m\widetilde{m}\alpha \mod 1$. Therefore,

$$\alpha(m\widetilde{m} - 1) = l, \tag{2.7}$$

for some $l \in \mathbb{Z}$. Note that if $m\widetilde{m} - 1 = 0$ then $m = \widetilde{m} = 1$, since m and $\widetilde{m}$ are integers. In this case, $\widetilde{\alpha} = \alpha$, which is a contradiction. Therefore, we divide by $m\widetilde{m} - 1 = 0$ and obtain $\alpha = l/(m\widetilde{m} - 1) = p/Q$, $p, Q \in \mathbb{Z}$, that is to say, α is a rational. Then, $\widetilde{\alpha}$ is also a rational, which is clear because $\widetilde{\alpha} = \pm\widetilde{m}\alpha + n$, $n \in \mathbb{Z}$.

We can now use some properties of arithmetic modulo Q that can also be described as a ring of integers modulo Q, $\mathbb{Z}/Q$. If the reader is curious about this, we refer them to Box 2.6. To simplify notation, instead of using set $\mathcal{S}_\alpha$ of points on a circle, we consider a subset of this ring, $\mathcal{W} \subseteq \mathbb{Z}/Q$, $\mathcal{W} = \{-(K-1)/2, \dots, (K-1)/2\}$, see Figure 2.6. We denote

$$\begin{aligned} \mathcal{W}a &= \{xa \mod Q : x \in \mathcal{W}\} \\ \mathcal{W} + a &= \{x + a \mod Q : x \in \mathcal{W}\} \end{aligned}$$

For $\alpha = p/Q$, we can relate set $\mathcal{S}_\alpha$ and $\mathcal{W}$ as

$$\mathcal{S}_{p/Q} = \{e^{2\pi x} : x \in \mathcal{W}p\}$$

Since p is co-prime with Q, there exist p^{-1} in $\mathbb{Z}/Q$ and it is co-prime with Q (Preposition 2.1.15 [219]). Let $q = p^{-1}\widetilde{p}$. Then, $\mathcal{W}p = \mathcal{W}\widetilde{p}$ if and only if $\mathcal{W} = \mathcal{W}q$.

Consider set $\mathcal{W}q + q$ shifted around the circle by q, see Figure 2.6. We have

$$\mathcal{W}q+q = \{-(K-3)q/2, \dots, (K+1)q/2\} = \underbrace{\{(K-3)q/2, \dots, (K-1)q/2\}}_{\subsetneq \mathcal{W}} \cup \{(K+1)q/2\}.$$

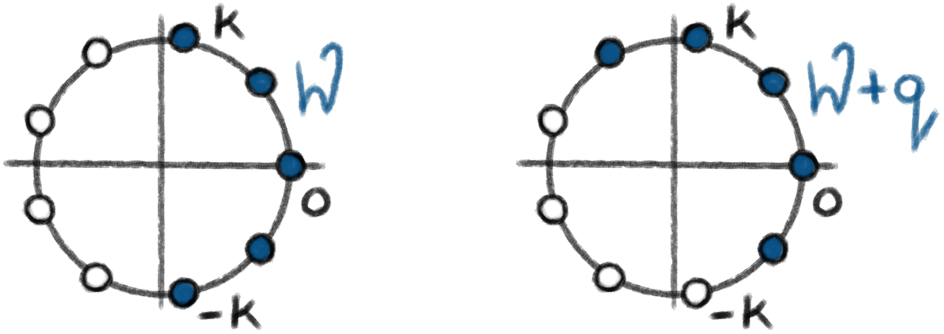


Figure 2.6.: Set $\mathcal{W}$ illustrated on a circle. The small circles illustrate all points in $\mathbb{Z}/Q$, thus the points on the complex plane can be described as $e^{i2\pi p/Q}$, $p \in \{0, Q-1\}$. Left: set $\mathcal{W}$ is concentrated around 0. Right: set $\mathcal{W}+q$ is the set $\mathcal{W}$ moved q elements counter clockwise; and if there are sufficient elements in the complement of $\mathcal{W}$, q of the previously empty circles will become coloured.

This means that shifting by q of the whole set $\mathcal{W}q$ — which is equal to $\mathcal{W}$ — we move out at most one element. If $Q > K+1$, this is possible only when $q = \pm 1$, this is a contradiction with the assumption that $\alpha \neq \pm\widetilde{\alpha}$. Therefore, we obtain that $Q \leq K+1$.

To prove the last part of the lemma, we know already that $\widetilde{\alpha} = \widetilde{p}/Q$. If $\widetilde{p}$ and Q were not co-prime, then there would be some r and $\widetilde{Q} < Q$ (and $< Q-1$ because Q and $Q-1$ are co-prime), such that $\widetilde{\alpha} = r/\widetilde{Q}$. But then the number of elements in $\mathcal{S}_{\widetilde{\alpha}}$ would be smaller than in $\mathcal{S}_{\alpha}$, which is a contradiction. ■

Want to know more? 2.5: Dirac Deltas and Distributions

Distributions are generalised functions, hence they are defined to behave as functions. There are entire books that discuss distributions, but for a formal and short introduction we recommend Paul DuChateau's notes [220]. If the reader is interested in Dirac deltas and is not interested in distributions, we refer them to Foundations of Signal Processing [44]. Intuitively, a Dirac delta is a function that has a spike at zero and is zero everywhere else. Alternatively, it is a function that if we multiply any other function f by it and integrate it, we obtain f at zero. In the following paragraph, we formalise the second intuitions to a definition of a Dirac delta as functional on the space of functions.

For simplicity, we will consider the space of smooth functions with compact support, $C_c^\infty(\mathbb{R})$. DuChateau [220] calls them test functions. We are interested in the functionals on this space; they are scalar-valued linear operators. For every function $f \in \mathcal{L}_2$, we can define a functional $v_f : C_c^\infty(\mathbb{R}) \to \mathbb{R}$ by using the inner product:

$$v_f(g) = \langle f, g \rangle \text{ for every } g \in C_c^\infty(\mathbb{R}),$$

where the product is well-defined due to the restrictions we imposed on g. However, there are functionals for which there is no function that satisfies the above equation. In particular, there is no function $f \in \mathcal{L}_2$ such that

$$v_0(g) = g(0) = \langle f, g \rangle \text{ for every } g \in C_c^\infty(\mathbb{R}).$$

Such a function would have to be zero everywhere except 0, but its integral would have to be non-zero. Indeed, this is the definition of Dirac delta δ — a symbol used to write functional v_0 in the integral form

$$\langle \delta, g \rangle = \int_{-\infty}^{\infty} g(t)\delta(t)dt = g(0) \text{ for every } g \in C_c^\infty(\mathbb{R}).$$

Any properties of the Dirac delta can then be derived from the properties of the inner product. For example, it is not obvious at first sight what the derivative of the Dirac delta should be. Using the inner product, we can observe that for any function in $f \in C_c^\infty(\mathbb{R})$, we have

$$\langle f', g \rangle = -\langle f, g' \rangle \text{ for every } g \in C_c^\infty(\mathbb{R})$$

We can then define δ' to satisfy $\langle \delta', g \rangle = -g'(0)$ for every $g \in C_c^\infty(\mathbb{R})$.

Want to know more? 2.6: Ring $\mathbb{Z}/Q$

The reader might be more familiar with frequencies on a circle, with integer modulo arithmetic or with rings. These three things have much in common and, here, we illustrate this. If the reader wants to know more about $\mathbb{Z}/Q$ and its applications, we refer them to the book by Stein [219]. For more a high-level abstract-algebraic point of view, we recommend a free book by Judson; it explains "everything you wanted to know about abstract algebra, but were afraid to buy" [221].

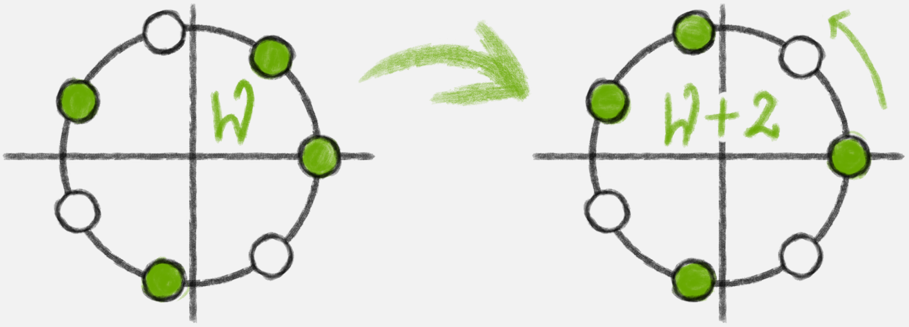
Let us begin with a group $(\{0, \dots, Q-1\}, +)$. A Commutative (or Abelian) group is a set $\mathcal{G}$ — in this case $\mathcal{G} = \{0, \dots, Q-1\}$ — with an operation — in this case $+$, such that for any $a, b, c \in \mathcal{G}$

$$\begin{aligned} (a+b)+c &= a+(b+c) \\ 0+a &= a \\ a+b &= b+a, \end{aligned}$$

where the latter is an additional property that makes the group commutative. Often, the operation in the group is denoted by multiplication $\cdot$ and not addition, and the neutral element is denoted as 1. Here, we use $+$ because we will later add a second operation. This also highlights that the group is commutative. The notion of *group action* on a set is a key concept in group theory. If we have a subset $\mathcal{W}$ of $\mathcal{G}$, we can add an element $a \in \mathcal{G}$ to all its elements obtaining a new set:

$$\mathcal{W} + a = \{x + a \mod Q : x \in \mathcal{W}\}.$$

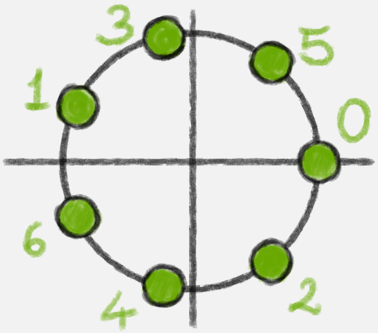
To visualise this on a circle, adding $+a$ to a set is a counter-clockwise rotation by a. We can see that the order of rotations does not matter, and we can always rotate our set back, thus we have all the group properties.

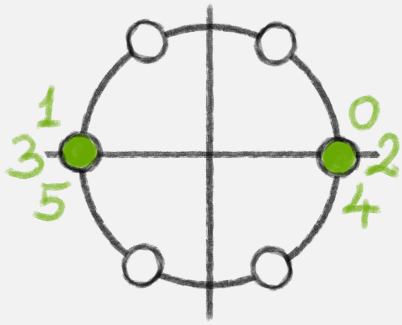


However, this does not describe the operation we are most interested in: multiplication of the elements of $\mathcal{W}$. If we add $\cdot$ to our group, we obtain a ring $\mathbb{Z}/Q = (\{0, \dots, Q-1\}, +, \cdot)$[a] that for any $a, b, c \in \mathcal{G}$ has the following additional properties:

$$\begin{aligned} 1a &= 1a = a \\ (ab)c &= a(bc) \\ a(b+c) &= ab + ac. \end{aligned}$$

If Q is prime, $\mathbb{Z}/Q$ has a convenient property that every element $a \in \mathcal{G}, a \neq 0$ has an inverse that is b such that $ab = ba = 1$ and is often $\mathbb{Z}/Q$ denoted as $\mathbb{Z}_Q$. A consequence is illustrated below, with $Q = 7$ and $Q = 6$.





On the left, for any element $a \in \mathcal{G}, a \neq 0$ multiplication of $\mathcal{G}$ by a is still $\mathcal{G}$. On the right, we have $\mathcal{G} \cdot 3 \subsetneq \mathcal{G}$. For those who are not familiar with group theory, it might be surprising that $\mathcal{G} \cdot a = \mathcal{G}$ only when a is co-prime with Q. For more about that see [219].

In Lemma 2.4, we ask for which Q there are integers p and $\max(2K+1, Q)$ element sets $\mathcal{W}$ such that $\mathcal{W}q = \mathcal{W}$. As previously discussed, if $Q \leq 2K+1$ set $\mathcal{W} = \mathcal{G}$ and every p co-prime with Q has this property. Therefore, the interesting question is, Can this happen when $Q > 2K+1$?

---

[a] To clarify the notation we have chosen $\mathbb{Z}/Q$ because it is concise. In fact, we should have written $\mathbb{Z}/Q\mathbb{Z}$ — the equivalence class of integers when a and $a + Qn$ are equivalent for every $n \in \mathbb{Z}$.

2.2. Application: Surface Retrieval

In this section, we show, with an example of surface retrieval from perspective projection, how to use our results in practice. We consider a camera in a flatland, i.e., a 2D world, the view of a linear surface $z(x)$ painted with an unknown texture f as illustrated in Figure 2.7. We recover, from a set of observations, both the texture and the surface. We first formulate the warping that a surface introduces. We explain the meaning of Lemmas 2.3 and 2.4 in this case and provide algorithms for recovering surface tilt and distance from camera. We also illustrate our results with numerical simulations.

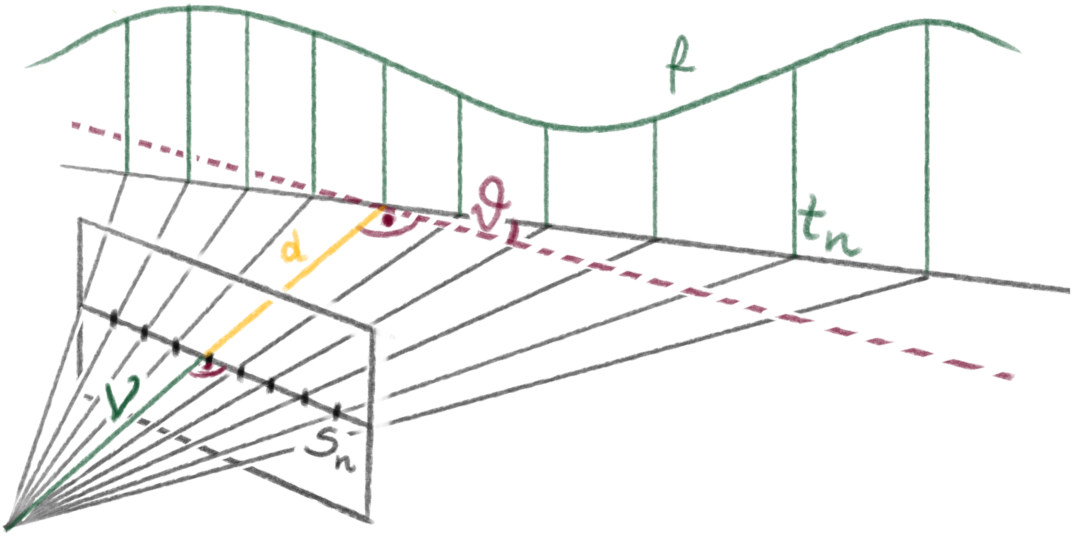


Figure 2.7.: Sampling of a warped signal, where the warping is defined by the perspective projection of the camera.

We assume that the camera model is a perfect perspective projection depicted in Figure 2.7. The samples s_n are taken uniformly on the camera plane by the CCD, where T is the distance between pixels. By d, we denote the distance between the camera plane and the surface at $x = 0$; the surface is tilted by θ; the surface equation is $y(x) = b + x\tan(\theta)$; and the distance along the surface (arc length) is $t = x/\cos(\theta)$. We can relate the distance along trajectory t with the image plain sample s via

$$\frac{t\cos\theta}{t\sin\theta + d} = \frac{s}{v} \quad \Rightarrow \quad t = \frac{ds}{v\cos\theta - s\sin\theta} = \varphi_{d,\theta}(s). \tag{2.8}$$

We can see that $\varphi_{b,\theta}$ is a rational function of s. Hence, if the painted function is a polynomial, we can use Lemma 2.3 to obtain unique recovery conditions. We will see that by using a polynomial, we can recover only the angle θ. To recover the distance b from the camera we can use bandlimited functions. In the following subsection, Methods, we describe how to do this, and in Section 2.2.2 we show results of this approach.

2.2.1. Methods

As we have seen in Equation (2.8), the warping function $\varphi_{b,\theta}$ introduced by perspective projection is a rational function. Unfortunately, the lemma is formulated in terms of constraints ψ and of not warping φ. As a result, we have to do some maths to obtain

Corollary 2.5: Corollary of Lemma 2.3

Let $\mathcal{F}$ be the space of polynomials of degree at most $K-1$. Let the warping be

$$\varphi_{d,\theta}(s) = \frac{ds}{v\cos(\theta) + s\sin(\theta)},$$

where v is a known parameter. If the number of samples is $N > 2(K-1)$, parameter θ is unique and function f is unique up to scaling.

Proof of Corollary 2.5

Let d and θ be the parameters of the true surface and $\widetilde{d}$ and $\widetilde{\theta}$ be the parameters of any other surface. Then, since

$$\frac{t\cos\theta}{t\sin\theta + d} = \frac{\widetilde{t}\cos\widetilde{\theta}}{\widetilde{t}\sin\widetilde{\theta} + \widetilde{d}},$$

we can find the constraint function from

$$\psi_{d,\theta,\widetilde{d},\widetilde{\theta}}(t) = \frac{\widetilde{d}t\cos\theta}{d\cos\widetilde{\theta} + t\sin\left(\theta - \widetilde{\theta}\right)}.$$

If $\theta \neq \widetilde{\theta}$, then the degrees of both numerator and nominator are equal to one; and from Lemma 2.3, we know that the only sample positions satisfying Equation (2.2) are the true positions. However, when $\widetilde{\theta} = \theta$, then

$$\psi_{d,\theta,\widetilde{d},\theta}(t) = \frac{\widetilde{d}}{d}t,$$

and we cannot apply Lemma 2.3. Moreover, this is scaling, hence we know that recovery of $\widetilde{d}$ is not possible. ■

Corollary 2.5 gives us precise results about what can be recovered and when. But how can we recover the f in practice? As we only have access only to sample values, we consider the cost function

$$c(\widetilde{\theta},\widetilde{b}) = \sum_{i=0}^{N}\left(f_n - f(\phi_{\widetilde{\theta},\widetilde{b}}(s_n))\right)^2.$$

If $\mathcal{F}$ is a linear space of polynomials, we can write this equation in the matrix form; this matrix can be written as

$$c(\widetilde{\theta},\widetilde{b}) = ||\boldsymbol{f} - \boldsymbol{V}_{\widetilde{\theta},\widetilde{b}}\boldsymbol{a})||^2,$$

where $\boldsymbol{a}$ is the vector of coefficients of f and matrix $\boldsymbol{V}_{\widetilde{\theta},\widetilde{b}}$ is the basis matrix:

$$\boldsymbol{V}_{\widetilde{\theta},\widetilde{b}} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \psi_{\widetilde{\theta},\widetilde{b}}(s_1) & \psi_{\widetilde{\theta},\widetilde{b}}(s_2) & \cdots & \psi_{\widetilde{\theta},\widetilde{b}}(s_N) \\ \vdots & \vdots & \ddots & \vdots \\ (\psi_{\widetilde{\theta},\widetilde{b}}(s_1))^{K-1} & (\psi_{\widetilde{\theta},\widetilde{b}}(s_2))^{K-1} & \cdots & (\psi_{\widetilde{\theta},\widetilde{b}}(s_N))^{K-1} \end{bmatrix} \tag{2.9}$$

Unfortunately, this cost function is not convex. We propose Algorithm 2.1 that is based on an algorithm proposed by Browning [92].

Algorithm 2.1: Alternating Least-Squares Algorithm (ALS)

```
Input: Sampled vector $\mathbf{f}$, initial sample positions $\widetilde{\mathbf{t}}$
Output: Sample positions $\widetilde{\mathbf{t}}$ and polynomial coefficients $\widetilde{\mathbf{p}}$.
Initialize sample transformation parameters $\alpha$
while not converged do
    For current matrix $\mathbf{V} := \mathbf{V}$ calculate:
        $\widetilde{\mathbf{p}} := (\mathbf{V}^\top \mathbf{V})^{-1} \mathbf{V}^\top \mathbf{f}$
    Calculate $\boldsymbol{\Phi}'$
    Update $\alpha$ :
        $\alpha := \alpha - \beta \boldsymbol{\Phi}'\widetilde{\mathbf{t}}\big((\mathbf{V}\mathbf{p} - \mathbf{f}) \circ (\mathbf{V}'\mathbf{p})\big)$
    Calculate $\widetilde{\mathbf{t}} = \varphi(\alpha)$
end while
```

We know now how to recover the angle θ. But how about the distance from the camera? The change of the distance from the camera results in only scaling. Could painting a bandlimited pattern provide uniqueness?

From Lemma 2.4, we know that the distance d is unique up to a measure-zero set, provided that the value of d is restricted to an interval. We could adapt ALS algorithm to the bandlimited case. In Section 2.1.2, we specifically consider uniform sampling of bandlimited signals so that spectral methods can be applied.

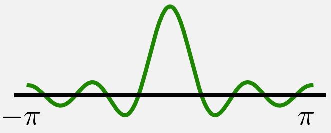
We use Prony's method to recover the frequencies of the warped signal (see Algorithm 2.2) because, in the noiseless case, Prony's method recovers frequencies exactly, hence we obtain recovery guarantees in the form of Corollary 2.6. If the reader is not familiar with Prony's method, or if they want to known more about it, we refer them to Box 2.7.

Once we have estimated all K frequencies ω_k by using Prony's method, we need to find the angular distance α (to estimated the warping β). It might be tempting to set α to the non-zero frequency $\alpha = \omega_1$, but due to the aliasing, the order of the frequencies can change, and ω_k might not correspond to αk, see Figure 2.8. Therefore, we propose to use the Dirichlet kernel as an invariant:

Definition 2.3: Dirichlet kernel

We define the Dirichlet kernel as a function $h : \mathbb{R} \to \mathbb{R}$:

$$h(\alpha) = \sum_{k=-(K-1)/2}^{(K-1)/2} e^{-i2\pi k\alpha} = \frac{\sin(\pi\alpha K)}{\sin(\pi\alpha)}.$$



We can estimate $h(\alpha)$ by taking

$$\widetilde{h} = \sum_{k=-(K-1)/2}^{(K-1)/2} e^{-i2\pi\omega_k},$$

as it does not depend on the order of the frequencies. It does, however, require all K frequencies to be estimated. For simplicity, let us assume that this is the case; in the Algorithm 2.2 and Corollary 2.6 we account for this case.

Once we have $\widetilde{h}$, we find all α such that $h(\alpha) = \widetilde{h}$. From Definition 2.3, we can see that the Dirichlet kernel is not invertible. Hence for α above a certain threshold, this search will return multiple αs, which can occur even if α is unique, see Figure 2.8. For this reason, we check if the recovered alphas α fit the measurement. For each α, we find β and the true sample positions. We can then solve the least squares in order to calculate coefficients a_k.

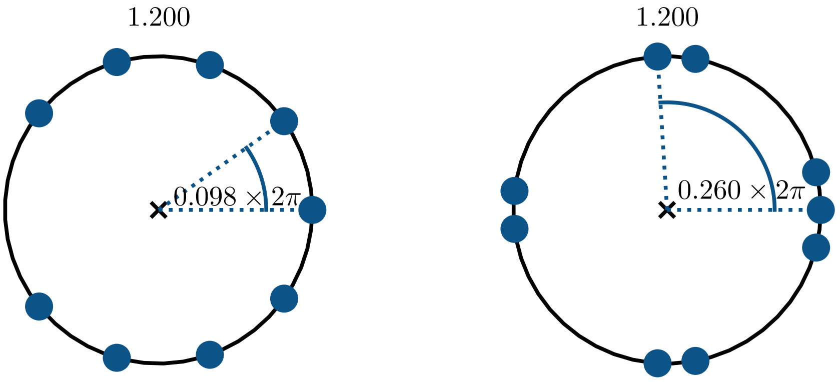


Figure 2.8.: Two sets of frequencies for different α that give the same value of h. In this case $K = 9$, on the left $\alpha \sim 0.098$ and on the right $\alpha \sim 0.26$, but in both cases $h(\alpha) = 1.2$. Note that the frequency closest to zero is not always the one that corresponds to $k = \pm 1$

Algorithm 2.2: DIRAChlet algorithm

```
Input: $y_n$, $n \in [N]$, $N \geq 2K$
Output: Set of possible warpings $\mathcal{B}$ and $a_k$ or low-pass $a_k$
$\mathcal{T} := \{\theta_k : k = -(K-1)/2, \dots, (K-1)/2\}$ estimated using Prony's method.
if $|\mathcal{T}| < K$ then
    $K := |\mathcal{T}| - 1$ if $|\mathcal{T}| \mod 2 = 0$ else $|\mathcal{T}|$
    $\mathcal{T} := \{\theta_k : k = -(K-1)/2, \dots, (K-1)/2\}$
end if
$\widetilde{h} := \sum_{k=-(K-1)/2}^{(K-1)/2} e^{-j\theta_k}$.
$\mathcal{A} := \{\widetilde{\alpha} : \widetilde{\alpha} \in (0, \pi), h(\widetilde{\alpha}) = \widetilde{h}\}$
$\mathcal{A} := \{\widetilde{\alpha} : \widetilde{\alpha} \in \mathcal{A}, \mathcal{S}_{\widetilde{\alpha}} = \mathcal{T}\}$
$\mathcal{B} := \{\widetilde{\alpha}/\tau : \widetilde{\alpha} \in \mathcal{A}\}$
for $\widetilde{\beta} \in \mathcal{B}$ do
    $a_k := \arg\min \sum_{n=1}^{N} \left(y_n - \sum_{k=-(K-1)/2}^{(K-1)/2} a_k^{i\widetilde{\beta}\tau n k}\right)^2$
end for
```

Corollary 2.6: Corollary of Lemma 2.4

If $a_k \neq 0$ for $k = -(K-1)/2, \dots, (K-1)/2$ and $0 \leq \beta \leq N/2$. Then the DIRAChlet algorithm returns only all the warpings β that fit the data (i.e., no other warpings). Unless $h(\beta/N) \in \{0, 1\}$, the returned β is unique. For each β, such that the frequencies do not overlap, the DIRAChlet algorithm returns the corresponding $\widetilde{f}$ and, if the frequencies overlap, then it returns the low-pass version of $\widetilde{f}$.

Before we prove Corollary 2.6, let us elaborate on the set $\{0, 1\}$. The specific values in this set are not important, whereas it is important that this set is small (measure zero, see Section 1.5 for definition). We include precise numbers here because they are the same as the numbers in a simulation-based conjecture that we have sought for a long time. They could also provide additional checks for the algorithm implementation.

Proof of Corollary 2.6

Prony's method Algorithm 2.2 finds all frequencies present in the DTFT of y_n. If all a_k are not zero, then Prony's method can return less than $2K + 1$ frequencies, but only if they overlap. We can therefore recover only a low-pass version of the signal. We then reduce K to the bandwidth of the low-pass signal.

If we have $\mathcal{S}_\alpha = \mathcal{S}_{\widetilde{\alpha}}$, then also $h(\alpha) = h(\widetilde{\alpha})$, thus if one of the equivalent αs is in $\mathcal{A}$, then all are. At the same time, we do check if $\mathcal{T} = \mathcal{S}_{\widetilde{\alpha}}$, thus removing all non-equivalent αs.

Now we need to show that if $h(\alpha) \notin \{0, 1\}$ then α is unique. From Lemma 2.4, we know α is unique, unless $\alpha = p/Q$, $Q < K$. If $Q \leq K$, then $\mathcal{T} = \mathcal{S}_\alpha$ has Q elements. But we have reduced K and $\mathcal{T}$ such that $K = |\mathcal{T}|$, hence we have two options for the reduced

K. First, $Q = K$ and $\alpha = p/K$ and $\sin(\pi K \alpha) = 0$ so $h(\alpha) = 0$. The other option is that $Q = K + 1$ and $\alpha = p/(K+1)$ But then

$$\sin(\pi K\alpha) = \sin\left(\pi p \frac{K}{K+1}\right) = \sin\left(\pi p - \pi \underbrace{\frac{p}{K+1}}_{\alpha}\right) = \sin(\pi\alpha),$$

because $p \in \mathbb{N}$. Therefore $h(\alpha) = 1$. ■

Want to know more? 2.7: Prony's Method [77, 222]

This method is attributed to Prony [77], but is also known as the Annihilating Filter Method [78]. Originally it was developed for estimating a sparse spectrum with a few frequencies that are not necessary on a grid. Since then, it has been generalised for sampling various functions that have a finite rate of innovation (FRI) [69]. Here, we adapt the description from [69], but if the reader interested in a more robust implementation of this concept, they can see the works by Pan summarized in his PhD thesis [80].

Prony's method consists of the following steps (illustrated at the end of the box). We begin with uniform samples x_n of a signal in the time domain. We know that the DTFT of the signal is

$$\widehat{x}(e^{i\omega}) = \sum_{k=-(K-1)/2}^{(K-1)/2} a_k \delta(e^{i\omega} - e^{i\theta_k}).$$

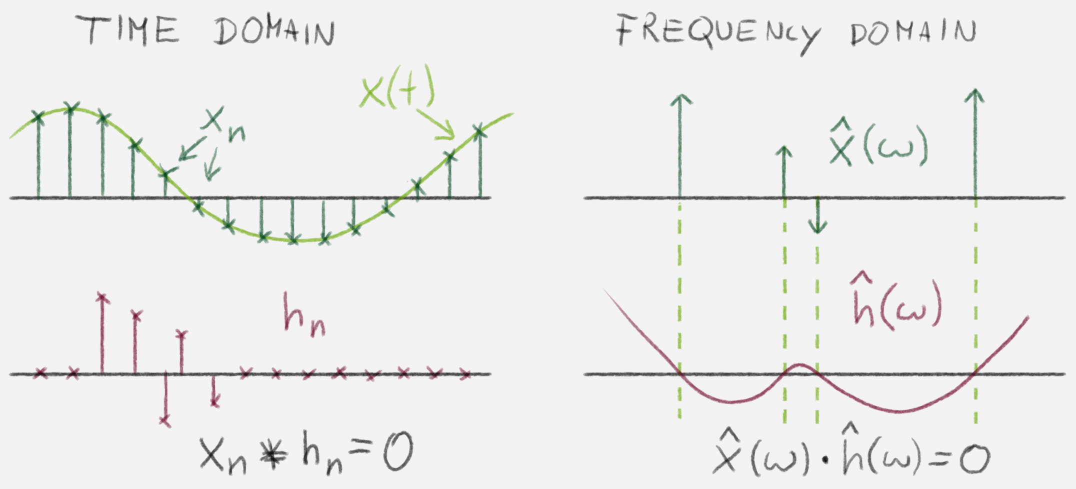
We look for a length-M *annihilating filter* h_n such that $h_n * x_n = 0$. This might not seem intuitive at fist, but let us observe that the DTFT of h_n is a polynomial in $e^{i\omega}$:

$$\widehat{h}(e^{i\omega}) = \sum_{m=1}^{M} h_m e^{i\omega m} = \sum_{m=1}^{M} h_m (e^{i\omega})^m.$$

Therefore, if we want to have $h_n * x_n = 0$ or equivalently $\widehat{x}(e^{i\omega})\widehat{h}(e^{i\omega}) = 0$, we need zeros of $\widehat{h}(e^{i\omega})$ to be at θ_k. Hence, we need $M = K + 1$. To find h_n, we can write convolution in the matrix form:

$$\begin{bmatrix} x_{K+1} & x_K & \cdots & x_1 \\ x_{K+2} & x_{K+1} & \cdots & x_2 \\ \vdots & \vdots & \ddots & \vdots \\ x_{2K+1} & x_{2k} \cdots & x_{K+1} & \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_{K+1} \end{bmatrix} = 0$$

and solve this system of equations with additional condition $h_1 = 1$ or $||h_n|| = 1$. To do this, we need at least $2K + 1$ samples. Then, we find roots of a complex polynomial with coefficients h_n and hope they are on the unit circle.



The main advantage of this method is that in the noiseless case it returns the exact values of θ_k, without any grid search that is computationally expensive and less precise. It is, however, unstable in the presence of noise and hard to extend to non-uniform sampling because it relies on discrete convolution.

2.2.2. Experiments

In this chapter, we studied two main scenarios for sampling at unknown locations with constrained sampling positions: a periodic bandlimited signal with a linear warping and a polynomial with sampling positions constrained by a rational function.

In this section, we present the simulation results for these two problems, respectively. The simulation code will be available online.

Periodic Bandlimited Signals

We begin by evaluating the behaviour of Algorithm 2.2 for unwarping periodic bandlimited signals in the presence of noise. We set $T = K$ and fixed $\boldsymbol{a} = [a_{-(K-1)/2}, \ldots, a_{(K-1)/2}]^\top$ and b to the following values:

$$\begin{aligned}\boldsymbol{a} =& [0.43, -0.15, -0.44, 0.67, -0.32, -0.76, -0.32,\\ & 0.67, -0.44, -0.15, 0.43]^\top,\\ \beta =& 4.\end{aligned}$$

Here, the values of $\boldsymbol{a}$ are arbitrary random values and $K = 11$, as there are K values in $\boldsymbol{a}$. As set out above, we sample $f(bx)$ with N samples per period. We study four different values of N, each corresponding to one of the regimes studied above:

Case 1: $N > 2\pi\beta/\alpha_c$: According to Corollary 2.6, this value of N guarantees that, in the noiseless case, the DIRAChlet algorithm will return one candidate $\alpha \in (0, \pi)$

and it will be the correct solution. Note that this value of N corresponds to $\alpha < \alpha_c$.

Case 2: $(K-1)\beta \leq N < 2\pi\beta/\alpha_c$: We choose this value of N so that the line s has several intersections with the Dirichlet kernel $h(e^{j\omega})$. In terms of α, this corresponds to $\alpha_c < \alpha \leq 2\pi/(K-1)$.

Case 3: $N < (K-1)\beta$: This value of N corresponds to $\alpha > 2\pi/(K-1)$.

Case 4: $N = \beta T$: This case results in $s = 0$ in the noiseless case when $\tau = K$ (our assumption in the simulations).

In each case, we fix the value of N in the interval of interest and run the simulation. Each value of N results in different samples y_n. We contaminate these samples by random Gaussian noise with zero mean and varying variance: $\widetilde{y}_n = y_n + \epsilon_n$. We use the noise such that the value of SNR ranges from -10dB to 40dB, where we define the SNR as

$$\mathrm{SNR} = 10\log_{10}\left(\sigma_{y_n}^2/\sigma_{\epsilon_n}^2\right) . \tag{2.10}$$

Here, σ_{y_n} and σ_{ϵ_n} are the empirical standard deviations of the signal y_n and noise ϵ_n, respectively. This results in the noisy observations $\widetilde{y}_n$. Then, we apply Algorithm 2.2 on these noisy samples to estimate the values of β and a_k (thus y_n); we call these estimated values $\widetilde{\beta}$, $\widetilde{a}_k$ and $\widetilde{y}_n$, respectively. For each value of SNR, we run 10,000 simulations.

When running the algorithm, one of three cases can occur:

1. The algorithm can return no solution, for example when the s-line does not intersect the kernel.
2. The algorithm can return multiple valid solutions (even after pruning), for example when $s = 0$.
3. The algorithm can return a single solution (after pruning).

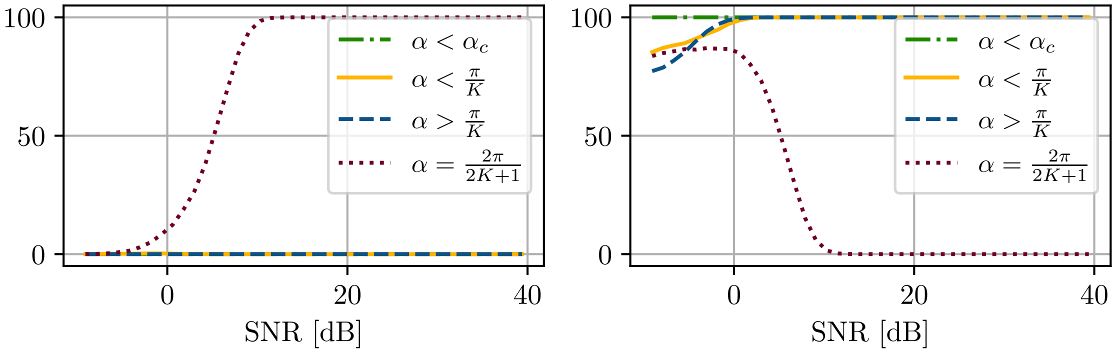


Figure 2.9.: Left: percentage of cases with more than one valid solution for β. Right: percentage of cases where Algorithm 2.2 results in a single solution.

Figure 2.9 depicts the percentage of these cases. The left plot shows the percentage of cases that have multiple solutions and the right plot shows the percentage of cases that have only a single solution. We see that, except for Case 4, there is always a unique

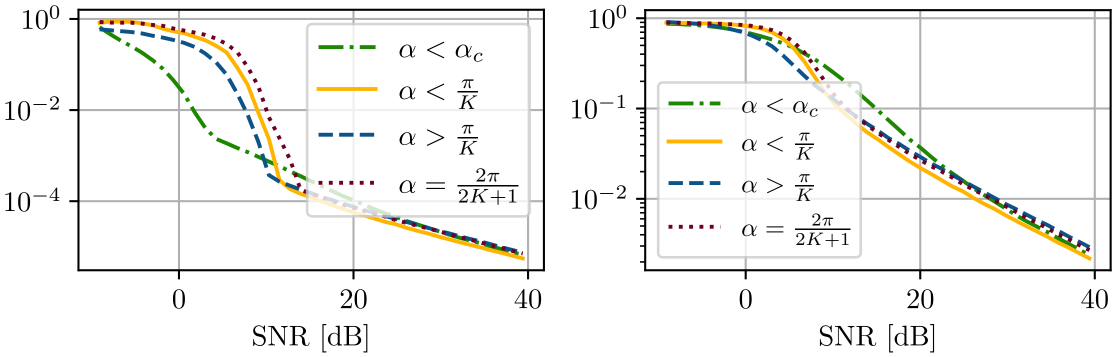


Figure 2.10.: Left: error in reconstructing β from noisy observations $\widetilde{y}_n$ versus SNR. Here, the original β was set to 4. Right: error in reconstruction of y_n from noisy observations $\widetilde{y}_n$ versus SNR.

solution, for the full SNR range considered. This is expected, as the non-unique cases form a finite set for $\alpha \in (0, \pi)$. In Case 4 (s = 0), we expect to have multiple solutions for high SNR, as we have fixed α to one of the cases from this finite set. In the right plot, we see a similar behaviour; except that, at low SNR, we fail to always return a single solution; this occurs when the algorithm returns no solution.

Next, in Figure 2.10, we show the error in reconstructing β and y_n, respectively. Again, we use the four cases detailed above, with the original value of β set to 4. We compute the average of the errors defined by

$$\text{error}(\widetilde{\beta}) = \frac{|\widetilde{\beta} - \beta|}{\beta},$$

$$\text{error}(\widetilde{y}_n) = \frac{\|\widetilde{y}_n - y_n\|_2}{\|y_n\|_2}.$$

In the case where the algorithm generates multiple solutions, we use the solution closest to the true value, when computing the above error. In Figure 2.10 (left), we observe that all cases, except Case 1, have a similar behaviour, with a break point in the curve at around 10dB. Case 1 has lower reconstruction errors for low SNR values, which is not surprising as this case has more oversampling. Also, the Dirichlet function has a small slope in this part of its curve, which results in smaller variation in $\widetilde{\alpha}$ with changes in s. We also observe that Case 1 does slightly worse for mid-SNR's. This can be explained by ill-conditioning, which occurs as $\alpha \to 0$. As shown in Figure 2.10 (right), although Case 1 has a lower error for estimating β in low SNR regimes, it has almost the same performance in estimating y_n. However, the ill-conditioning still makes the error slightly worse at mid-SNR's.

Polynomials

To evaluate the performance of the ALS algorithm in the polynomial case (see Algorithm 2.1), we simulate the surface retrieval problem we have introduced in Section 2.2. We describe how to alter Algorithm 2.1 in order to retrieve the angle.

We assume a polynomial texture and linear surface, with unknown angle and offset. Recall that, assuming the pinhole camera model, the sample positions are defined by

$$\varphi(nT) = \frac{nTd}{v\cos\theta - nT\sin\theta}.$$

Note that, from Corollary 2.5, we know that $2(K-1)$ samples are sufficient to distinguish between different angles of the surface. Once the angle is found, the constraint ψ becomes a linear function, and Corollary 2.5 does not tell us anything about the recovery of the offset d. Note that changing the distance of the surface from the camera is equivalent to scaling the polynomial. But a scaled polynomial is also a polynomial, hence it is impossible to recover this offset. This also suggests that it is difficult to relax the assumptions of the lemma.

From the above reasoning, we know that we will be unable to recover the offset of the surface. Therefore, we ran a number simulations with different polynomial degrees, surface orientations and noise levels. We set the irretrievable distance d and the focal length v to 1.

For each polynomial degree, surface orientation and noise level, we ran 100 experiments with arbitrary random polynomials. The polynomials were generated in the standard polynomial basis. The coefficient of the highest power was fixed to 0.5, and the remaining coefficients were generated randomly from a standard normal distribution $\mathcal{N}(0, 1)$. We needed to fix the first coefficient to ensure that it is not zero. This is because, if the polynomial is similar to a polynomial of smaller degree, the model becomes too powerful with respect to the data.

If not stated otherwise, each of the 100 tests was done for 13 different angles uniformly spaced between $-20°$ and $20°$. The alternating algorithm is always initialized with $b = 0$. The initialized sample values range from -1 to 1, hence angles close to 45° cannot be recovered, because the line from the origin to the last sample would also be 45° and would thus never cross the surface. We restrict the angles even further due to stability problems, see Figure 2.13.

We add, to the sample values, noise generated from the normal distribution $\mathcal{N}(0, \sigma)$, for different values of σ. The signal to noise ratio (SNR) – defined in (2.10) – varies between -10dB and Thank you! 200dB.

We report the error in position estimates defined as

$$\text{error}(\widetilde{\theta}) = |\widetilde{\theta} - \theta|. \tag{2.11}$$

Recall also that, as the algorithm knows only the sample values, the cost function it minimizes is

$$c(\boldsymbol{x}, \boldsymbol{a}) = \|\boldsymbol{V}(\boldsymbol{x})\boldsymbol{a} - \boldsymbol{y}\|^2. \tag{2.12}$$

This cost function is in general not convex, see Figure 2.11. This means that, without any additional modifications, the algorithm will sometimes miss the global minimum. This can be corrected by choosing a number of different starting positions, or other standard methods. In the noiseless case, we know that the cost function is equal to 0 if and only if we found the global minimum. However, in the noisy case, distinguishing between local minima might be difficult[2].

Figure 2.11.: The cost function is not convex. The plot shows the cost function for different estimated errors, for a fixed polynomial of degree 9, with no noise added. The true angle is 10 degrees. As the algorithm is initialized at 0, it will stop at the local minimum.

In the simulations, we have seen the problems with local minima for all degrees of polynomials. In our experience, when the polynomial degree is small, local minima do not appear often. As the degree of the polynomial increases, they became more common and lead to increased error even in the noiseless case, see Figure 2.12 and Figure 2.12.

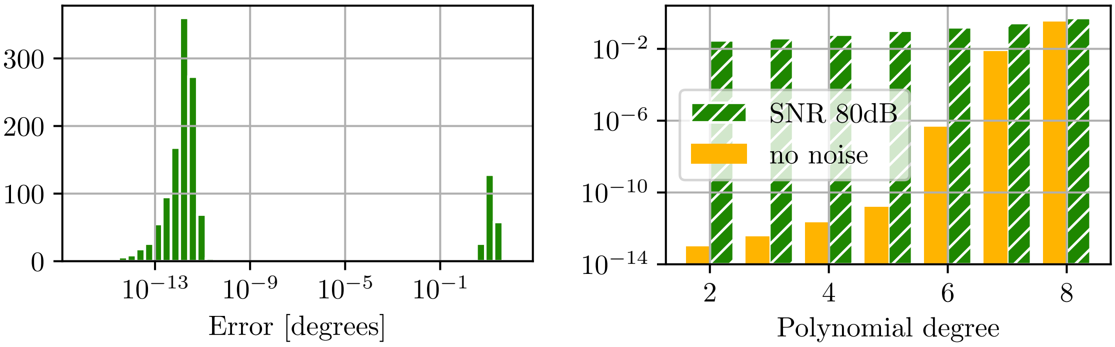


Figure 2.12.: Left: histogram of the errors for a polynomial of degree 4, with no additional noise added. The distribution of the error is clearly bi-modal. In more than 80% of cases the error is lower than 10^{-10}, yet the mean is around 2.5. Right: results of the ALS algorithm for different polynomial degrees. The median error with no noise is shown in orange (solid) and the median error with a small amount of additive noise (SNR 80dB) is shown in blue (hatched). As we can see, in the noiseless case, the algorithm breaks down at around degree 6. In the noisy case, the errors are much larger, even when the algorithm finds the global minimum. This makes the error less dependent on the number of local minima, as a result less dependent on the degree of the polynomial.

The reconstruction is not robust to noise, see Figures 2.12 and 2.13. This is not a problem with the alternating algorithm, but with the chosen cost function. With noise, the minima of the cost function flatten out, because perfect fitting of the polynomial to the samples is no longer possible. This is also the reason why oversampling does not give

---

[2] Note that, for this 1D problem, it is easy to come up with a more robust scheme, such as a simple grid-search. However, we have chosen the ALS algorithm, because it generalizes easily to higher dimensions therefore is more illustrative.

a significant improvement — although the oversampling reduces the relative power of the noise, it does not prevent minima from flattening out.

For angles inside the interval $[-20°, 20°]$, the error does not vary significantly, see Figure 2.13. Outside this interval, the algorithm becomes unstable. This is due to the geometry of the problem and the fact that a small change in θ leads to a big change of the estimated sample positions hence a substantial change in the estimated coefficients of the polynomial. We can imagine that a change of variables or introducing a varying step size, depending on the current angle, could widen the stable region.

Finally, we adjust the step size and the stopping criteria of Algorithm 2.1. We use step size β to be inversely proportional to the oversampling factor, in order to prevent large gradients. If the gradients are too large, they can cause the algorithm to move to angles θ outside the permitted $(-45°, 45°)$ interval. Therefore, we multiplicatively decrease β every time θ could become extreme. We use different stopping criteria: when the cost function is sufficiently low, when the cost function stops changing, and after a certain number of iterations. We noticed that increasing the number of iterations does not improve the results, and limiting the number of iterations might be seen as a version of early stopping.

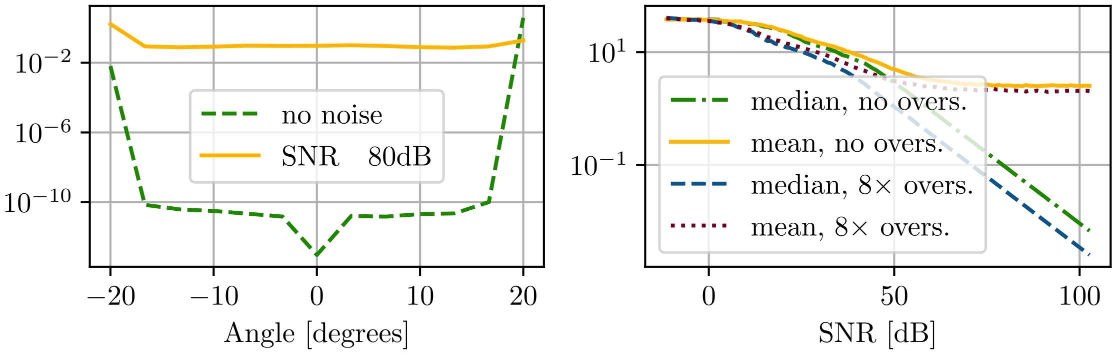


Figure 2.13.: Left: median error for different angles, aggregated for 7 different polynomial degrees (2,. . . ,8). The blue dashed line (left scale) shows the noiseless case, and orange (solid) line shows an SNR of around 80dB. The decrease of error around 0 for the noiseless case is due to algorithm initialization at 0. Right: Error for different signal to noise ratios, for polynomials of degree 4. Around 40dB SNR, the mean error and median error begin to differ. Unsurprisingly, the mean error flattens when the few large errors dominate the mean. Oversampling improves the results but not significantly. Oversampling 8 times gives error equivalent to no oversampling, with an SNR 10dB larger, but this results in the small difference in error.

2.3. Future Work

In this section, we highlight some questions left unanswered by this chapter, and we describe the difficulties that arose when we tried to answer them. We also suggest some general directions in which this work might be further developed.

The most noticeable gap in this chapter is the difference in sampling schemes between polynomials and bandlimited functions. Do bandlimited functions really need uniform sampling? Would the solution be unique if we warped them with a rational function? One fact is clear: linear warping with polynomials does not work.

To extend our results to non-uniform sampling of bandlimited signals, we would need a provable spectral estimation method. Although there are many non-uniform spectral estimation methods that work well in practice [80, 81, 223, 224], we are not aware of any that provides exact reconstruction, even in the noiseless case.

The problem with non-uniform samples in Prony's method is that the annihilating polynomial in the frequency domain (or "spike" domain) corresponds naturally to *uniform* Diracs in the time domain. Therefore, we need either some interpolation, a new annihilating function (not a polynomial), or a new transform (not the Fourier transform). The same problems appear in recovery of non-uniformly sampled FRI signals.

There exist various generalisations of Prony's method, but most of them focus on different classes of functions, not different sampling schemes [225–230]. The work by Plonka [231, 232] presents a general approach for the reconstruction of sparse expansions of eigenfunctions of suitable linear operators; it allows for a family of sampling operators instead of a single one. This gives us a more flexible sampling scheme, but ultimately still requires uniform sampling, in the sense that each sampling operator requires K samples at uniform (integer) powers of the aforementioned operator.

Whereas, work by Senay [233] uses both non-uniform samples and Prony's method, the setup is entirely different from ours – in particular, a version of Prony's method is used to find the non-uniform sample positions, and the frequency content is known. The work by Coluccio [234] provides guarantees in non-uniform case when the integrals can be measured (it a special case of [232]), but to use this method in practice would require approximating the integrals.

If the reader interested crafting a provable spectral estimation method from non-uniform samples, our problem of recovery of linearly warped bandlimited signals would be good start. Our problem has one unknown parameter — the warping β — that defines the support of the spectrum, instead of K parameters defining each spectral component separately. At the same time, it poses the same difficulties described above. A possible approach to this problem would be to apply methods from polynomial case, see Example 2.8, or to use general methods from algebraic geometry, similar to methods used in structured matrix decomposition [235].

Prony's method is also not good for rational warping because, after such warping, the spectrum is no longer a sum of Dirac deltas. The signal is still a FRI signal, but now it has an unknown shape. The work of Baechler et al., [229] which considers FRI signals with partially known shape, might give some inspiration to tackle this problem. We see no reason why rational warping would not give uniqueness of recovery of the composite signal. Again, methods from polynomial case might give some recovery guarantees, see Example 2.8.

It is worth noting that problems with non-uniform sampling arise only if we want to have recovery guarantees. If not, Pan and Simeoni propose robust algorithms for recovery of FRI signals [80, 81]. It would be interesting to see how generalised FRI methods improve over the DIRAChlet algorithm.

Example 2.8: Different Approach to Linear Warping

We consider a linear warping, described with a parameter β. As in the polynomial case, we want to study the solutions to the following system of equations:

$$y_n = f(t_n) = \widetilde{f}(\beta \widetilde{t}_n) \text{ for } n \in [N].$$

If we know that for any $f, \widetilde{f} \in \mathcal{F}$ and any β the difference $f(t) - \widetilde{f}(\beta t)$ can have at most $N_{\max}$ zeros (or be constantly zero), we could say that any number of samples above $N_{\max}$ guarantees uniqueness. It can be shown that this difference has a finite number of zeros on a finite interval. For example, by showing that otherwise there would be a point on the interval, where the difference would have to have all derivatives equal to zero.

More precisely, if the difference had an infinite number of zeros on an interval, then according to Bolzano–Weierstrass theorem the set of zeros would have a limit on this interval. By using the limit-based definition of the derivative, we obtain that all derivatives have to be zero. For bandlimited f, this means that the difference has to be zero.

However, it is unclear exactly how many zeros it can have. Possibly, tricks similar to counting polynomial zeros, which we mention further in this section, could be employed.

As we know that there is only a measure zero set of times that satisfy the above equation, perhaps we could achieve uniqueness of recovery of f and β up to measure zero set of samples positions t_n? The problem with this approach is that this set is of measure zero for a fixed β. It is still possible that for any fixed t_ns, we can find new $\boldsymbol{\beta}$ and new $\widetilde{t}_n$s that match the samples.

Another difference between the polynomial and bandlimited cases is that, for the DIRAChlet algorithm, we have recovery guarantees in the noiseless case, whereas for ALS there are no guarantees, and we have seen that it gets stuck in local minima. Therefore, there is certainly some room for improvement here. As the cost function is not convex, some relaxation methods might be needed. Also, note that the DIRAChlet algorithm requires all amplitudes $a_k \neq 0$, because it needs to calculate the Dirichlet kernel. ALS-like optimisation could work in practice when $a_k = 0$ for some k but it would not give theoretical guarantees either. Alternatively, once the positions of the Dirac deltas are estimated using Prony's method, we could apply some modification of Euclidean algorithm for noisy measurements [236].

The next issue with polynomials constrained by rational functions is that the oversampling factor depends on the degree of the rational function, rather than on its number of parameters. By considering a polynomial with few monomials, so-called *fewnomial* [237], we could lower the oversampling factor in specific cases. For a start, Descartes' rule of sign could be used if we know that all times t_n are positive and the polynomial in question has positive coefficients; or Strum's theorem [238] could be used if we knew that the polynomials in question do not have multiple roots.

The last direction we suggest is multi-channel sampling at unknown locations. It might seem to be a strange idea for now, but we will revise it in the later chapters, where it will be relevant. By multi-channel sampling at unknown locations, we mean sampling a length-M vector signal $\boldsymbol{s}(t)$, $s_m \in \mathcal{F}, m \in [M]$ in one of the following scenarios:

- $\boldsymbol{s}(t)$ is sampled at times $\varphi(t_n)$ (t_n might be uniform or not)
- each $s_m(t)$ is sampled at its own times $\varphi(t_{n,m})$, but the warping φ is common
- each $s_m(t)$ is sampled synchronously, but with different warpings $\varphi_m(t_n)$

Depending on what we want to use, polynomials or band-limited functions, the difficulties will vary between the scenarios above. Due to all problems with Prony's method that we have described before, we believe the non-uniform sampling with bandlimited signals is the most difficult.

Chapter 3.

Rank One Sensing

We discovered the rank-one sensing problem discussed in this chapter while studying the continuous localisation, which we develop in Chapter 4, and the event-based sampling, which we describe in Section 3.2. Both these problems require full-rank matrix recovery from rank-one measurements.

As described in Section 1.3, various prominent problems can be described as a matrix recovery from linear measurements. This includes matrix completion [181–183] and low-rank matrix recovery [180, 187].

Although all these problems could be vectorised and described as linear inverse problems, it is often better to use the matrix structure explicitly. When few measurements are available, the assumption on the matrix structure is crucial for uniqueness. Therefore, low-rank [180–183, 187], sparsity [240], and other constraints are used, despite their non-linearity. In other cases, exploiting the matrix structure can lead to more efficient algorithms than vectorisation does [241].

Another difference between vector and matrix recovery is that, in the case of missing measurements, it matters which measurements are missing. In low-rank matrix completion, the matrix cannot be recovered if there are no samples in at least one of its columns (or rows) [181]. This lead many researches to study matrix completion with deterministic masks, see Section 1.3. When the measurements are rank-one, even without low-rank assumptions, it is difficult to know whether the solution is unique, without directly checking the rank of the system. In this work, we give sufficient and, in some cases, necessary conditions for recovery from incomplete measurements of an unknown full-rank matrix $\boldsymbol{C}$ sandwiched between two known matrices $\boldsymbol{G}$ and $\boldsymbol{F}$ ($\boldsymbol{G}^\top \boldsymbol{C}\boldsymbol{F}$).

Although in this work we focus on the theory, our results can be applied to various problems, from localisation [242] to time encoding machines [243]. More generally, we believe our results can be used in any case of non-synchronised sampling with array sensors. With the rise of event-based sensors [50, 244] (Section 1.1), we expect to discover new applications of this work.

Contributions: The work in this chapter is joint work of the author (MP) with Karen Adam (KA), Martin Vetterli (MV) and Adam Scholefield (AS) [239]. AS and MP formulated the initial problem; MP generalised the problem, formulated and proved the all results; KA and MP found the connection to TEMs and KA formalised this connection. MV and AS advised the research. MP wrote this chapter, taking liberty to rearrange the results and supplement them with a history of the research.

Previous results on matrix recovery from rank-one measurements assume that the matrix $\boldsymbol{C}$ to be recovered is low-rank. As low-rank matrices are sparse in the spectral domain, the low-rank assumption enables the use of compressed sensing methods that have been extremely successful in low-rank matrix recovery [180–183, 187]. Unfortunately, the main tool of compressed sensing, the Restricted Isometry Property (RIP) is not satisfied by the operators made of rank-one matrices [245, 246]. This problem, however, can be circumvented with custom criteria that replace the RIP [246, 247]. Compressed sensing that relies on incoherence properties [186, 190, 194], instead of the RIP, is also a viable solution for rank-one sensing, as shown by Jain et al. [245]. There is also a body of work that focuses on the rank-one formulation, but it relies less on standard matrix recovery methods [246, 248–250]. These methods assume that $\boldsymbol{C}$ is low rank and most of them [246, 248, 249] additionally require both $\boldsymbol{C}$ and the sensing matrices to be symmetric. The setup by Zhong et al. [250] is the most similar to our formulation.

In addition to the low-rank assumption, it is generally assumed that the measurement vectors have i.i.d. sub-Gaussian entries, with the notable exception of the work by Jain et al. [245]. Although this assumption allows for the use of powerful statistical techniques, it is often not realistic, in particular when the sampling model is defined by the environment.

In this chapter, we place no low-rank assumption on $\boldsymbol{C}$ and do not require the measurement vectors to be i.i.d. We formalise the problem in Section 3.1 and describe our assumptions in more detail in Section 3.1.1. As usual, we begin with intuitions in Section 3.1.2; they can be skipped if the reader is interested only in essentials. In Section 3.1.3, we present a condition for unique recovery with the smallest possible number of measurements and a more general uniqueness condition at a price of two-times oversampling. We postpone the proofs to Section 3.1.4.

Finally, in Section 3.2, to illustrate how to apply our theory in practice, we present a number of applications. We consider distributed sampling with a noisy communication channel, sampling with time encoding machines, and sampling signals with hidden structures.

3.1. Theory

The problem considered in this work can be studied from a few different angles, three of which we describe below.

Matrix completion Consider the recovery of matrix $\boldsymbol{C} \in \mathbb{R}^{J\times K}$ from measurements obtained by matrix multiplication on "both sides":

$$\widetilde{\boldsymbol{Y}} = \boldsymbol{W} \odot \boldsymbol{G}^\top \boldsymbol{C}\boldsymbol{F}, \tag{3.1}$$

where $\odot$ is the point-wise (Hadamard) product, $\boldsymbol{G} \in \mathbb{R}^{J\times M}$, $\boldsymbol{F} \in \mathbb{R}^{K\times N}$ are known and $\boldsymbol{W} \in \mathbb{R}^{M\times N}$ is a known binary mask ($w_{m,n}$ is one if the measurement is available). As we do not make any assumptions on the structure of $\boldsymbol{C}$, we assume that $M \geq J$ and $N \geq K$. This formulation can be interpreted as a matrix completion problem: to recover matrix

$\boldsymbol{Y}$ from incomplete $\widetilde{\boldsymbol{Y}}$ while using the partial information about the nullspace and the range of $\boldsymbol{Y}$.

Rank-one sensing Alternatively, the problem we study can be formulated as matrix recovery from rank-one measurements. A single entry of $\boldsymbol{Y}$ can be written as

$$y_{m,n} = \boldsymbol{g}_m^\top \boldsymbol{C} \boldsymbol{f}_n, \tag{3.2}$$

where $\boldsymbol{g}_m$ is m-th column of matrix $\boldsymbol{G}$ and $\boldsymbol{f}_n$ is n-th column of matrix $\boldsymbol{F}$. As not all pairs of measurements are available, we introduce the set of available measurement pairs $\mathcal{M} \subseteq [M] \times [N]$. A pair (m, n) belongs to $\mathcal{M}$ if and only if $w_{m,n} = 1$.

In this notation, Equation (3.1) can be rearranged, using trace properties[1], into inner products with **rank-one** matrices $\boldsymbol{g}_m \boldsymbol{f}_n^\top$:

$$y_{m,n} = \langle \boldsymbol{g}_m \boldsymbol{f}_n^\top, \boldsymbol{C} \rangle = \langle \boldsymbol{g}_m \otimes \boldsymbol{f}_n, \boldsymbol{C} \rangle \quad (m, n) \in \mathcal{M}. \tag{3.3}$$

We usually use only matrix product $\boldsymbol{g}_m \boldsymbol{f}_n^\top$, but the tensor product formulation enables a simple extension of this problem to higher dimensions. We can measure an order-D tensor $\boldsymbol{T}$ with rank one tensors, also called simple tensors:

$$y_{\boldsymbol{n}} = \langle \boldsymbol{g}_{n_1}^{(1)} \otimes \boldsymbol{g}_{n_2}^{(2)} \otimes \cdots \otimes \boldsymbol{g}_{n_D}^{(D)}, \boldsymbol{T} \rangle, \quad \boldsymbol{n} \in \mathcal{M}_T \tag{3.4}$$

where $\boldsymbol{n} = (n_1, n_2, \ldots, n_D)$, $\boldsymbol{g}_n^{(d)} \in \mathbb{R}^{N_d}$, and $\mathcal{M}_T \subseteq [N_1] \times [N_2] \times \cdots \times [N_D]$. Here we use superscript to denote different frames.

Equation (3.3) can, of course, be written in the matrix form by vectorizing matrix $\boldsymbol{C}$:

$$\boldsymbol{y} = \boldsymbol{\Gamma} \operatorname{vec}(\boldsymbol{C})$$

where $\boldsymbol{y}$ is the vector of *available* measurements and $\boldsymbol{\Gamma}$ is the matrix of vectorised measurements:

$$\boldsymbol{\Gamma} = \begin{bmatrix} \operatorname{vec}(\boldsymbol{g}_{m_1} \boldsymbol{f}_{n_1}^\top) \\ \operatorname{vec}(\boldsymbol{g}_{m_2} \boldsymbol{f}_{n_2}^\top) \\ \vdots \\ \operatorname{vec}(\boldsymbol{g}_{m_{|\mathcal{M}|}} \boldsymbol{f}_{n_{|\mathcal{M}|}}^\top) \end{bmatrix}, \tag{3.5}$$

where pairs (m_i, n_i) cover all set $\mathcal{M}$.

Tensor product of frames Finally, the problem can be formulated in terms of frames. Since $\boldsymbol{G}$ and $\boldsymbol{F}$ are fat matrices, if they are full rank, they define two frames, respectively, in $\mathbb{R}^J$ and $\mathbb{R}^K$. At the same time, Equation (3.3) is a system of linear equations, thus its solution is unique if matrices $\boldsymbol{g}_m \boldsymbol{f}_n^\top$, $(m, n) \in \mathcal{M}$ span the whole $\mathbb{R}^{J \times K}$.

Therefore, asking if there exists a unique $\boldsymbol{C}$ satisfying Equation (3.3) is equivalent to asking *which* ***subsets*** *of the tensor product of frames are themselves frames.*

Although various aspects of frames in the tensor product of Hilbert spaces have been studied [251–253], we have not found any direct study of subsets of the tensor products of frames.

---

[1] For details see Section 1.5.

3.1.1. Assumptions

In this section, we introduce and discuss two definitions; we will later use them to quickly refer to different assumptions we make. We show in Section 3.1.2 why these assumptions are needed. We assume all our functions and matrices have values in $\mathbb{R}$ but, in principle, we could change it to $\mathbb{C}$, without the need to alter the proofs.

Definition 3.1: Full Spark

We say a matrix $\boldsymbol{M} \in \mathbb{R}^{R\times N}$, $N \geq R$, is *full spark* if it has a spark of $R+1$, i.e., every R of its columns are linearly independent, and at least one set of $R+1$ is not. We can also say that the Kruskal's rank [254] of $\boldsymbol{M}$ is R. By extension, we call a N-element frame $\boldsymbol{g}_n$ in $\mathbb{R}^R$ a full-spark frame if the corresponding matrix is full spark.

Definition 3.2: Polynomial Model

We say that $\boldsymbol{F}$ follows the polynomial model if the entries of vectors $\boldsymbol{f}_n \in \mathbb{R}^K$ are polynomials of hidden variables $\boldsymbol{z}_n \in \mathcal{I} \subset \mathbb{R}^D$, $1 \leq D \leq K$:

$$\boldsymbol{f}_n = \boldsymbol{p}(\boldsymbol{z}_n) \text{ for } n \in [N],$$

where the measure of $\mathcal{I}$ is non-zero and $\boldsymbol{p}$ is such that for sequences $(\boldsymbol{z}_n)_{n=1}^N \in \mathcal{I}^N$, matrix $\boldsymbol{F}$ is full spark.

Definition 3.2 is more complex than Definition 3.1, hence we avoid using it if possible. As it might seem a bit abstract, we present some examples of $\boldsymbol{F}$; they follow the polynomial model where $\boldsymbol{F}$ is full spark for almost all $(\boldsymbol{z}_n)_{n=1}^N$.

Example 3.1: Independent Entries

Let $r = K$, $\mathcal{I} = \mathbb{R}^K$ and $\boldsymbol{p}(\boldsymbol{z}) = \boldsymbol{z}$. Almost all matrices in $\mathbb{R}^{K\times N}$ are full spark.

Example 3.2: Trigonometric Polynomials

Let $r = 1$, $\mathcal{I} = [0, 1]$ and

$$\boldsymbol{p}(z) = \begin{bmatrix} 1 & z & z^2 & \dots & z^{K-1} \end{bmatrix}^\top .$$

In this case $\boldsymbol{F}$ is a Vandermonde matrix and, unless $z_j = z_i$ for $i \neq j$, it is full spark. Therefore, it is full spark for almost all $(\boldsymbol{z}_n)_{n=1}^N$.

Now, if $z_n \in [0, 1]$, we can write it as $z_n = \cos(\theta_n)$, and obtain measurements in the cosine-transform domain. Similarly, we can use this model to express measurements in the Fourier domain if we permit complex values.

For completeness, let us note that in our pre-print [239] we have used a slightly different set of assumptions, i.e., we have assumed that the k-th entry of $\boldsymbol{f}_n$ has the form

$$[\boldsymbol{f}_n]_k = f_k(t_n), \tag{3.6}$$

where $f_k : \mathcal{I} \to \mathbb{R}$ $k = 0, \dots, K-1$ are *linearly independent* functions from a linear space of functions $\mathcal{F}$, $\mathcal{I} \in \mathbb{R}$ is an interval or the whole real line and $t_n \in \mathcal{I}$, $n = 0 \dots N-1$ are sampling times. Moreover, in the preprint, we assumed that the sampling times $(t_0, \dots, t_{N-1})$ follow a continuous probability distribution on $\mathcal{I}^N$ and that for every non-zero element $f \in \mathcal{F}$, the set of zeros of f has Lebesgue measure (λ) equal to zero: $\lambda(\{t | f(t) = 0\}) = 0$.

The assumptions we make here are more general in the sense that we can have multiple hidden variables $\boldsymbol{z}$ and not only the sampling time t. However, they restricts functions f_k to be polynomials (p_k in Definition 3.2) and require matrix $\boldsymbol{F}$ to be full spark (Definition 3.1). These two sets of assumptions concern the linear independence of f_ks and the measure zero[2] of zeros of the non-zero function in $\mathcal{F}$[3]. We believe the new restriction of functions to polynomials is worth the clarity gained by this assumption. However, our results still hold for real-valued bandlimited functions and other families that satisfy Equation (3.6).

Finally, note that a definition similar to Definition 3.2 can be formulated for measurements generated from a random process. We simply need for $\boldsymbol{z}_n$ to be independent continuous random variables and replace *almost all* with *almost surely*.

Want to know more? 3.3: K-Regular Mappings

Definition 3.2 and the definition from [239] could be also replaced by a requirement that is somewhat in between. We could assume that the measurements have the following form:

$$\boldsymbol{f}_n = \boldsymbol{f}(\boldsymbol{z}_n) \text{ for } n \in [N],$$

where $\boldsymbol{f}$ is *K-regular* — for any K distinct variables $\boldsymbol{z}_n$, vectors $\boldsymbol{p}(\boldsymbol{z}_n)$ are linearly independent. Additionally, we have to assume either that $\boldsymbol{p}$ are polynomials or that, for every non-zero element $f \in \operatorname{span}(f_1, \dots, f_K)$, the set of zeros of f has Lebesgue measure (λ) equal to zero: $\lambda(\{t | f(t) = 0\}) = 0$.

Assuming that $\boldsymbol{f}$ is K-regular implies that, for any *distinct* variables $\boldsymbol{z}_n$, matrix $\boldsymbol{F}$ is full spark. This definition is stronger than Definition 3.2. An advantage of this definition is that it is general and connects to the body of work on existence of K-regular maps [255, 256] and to Chebyshev interpolation [257].

3.1.2. Intuitions and Challenges

In this section, we present the story of how we obtained results that we describe in Section 3.1.3. We believe that, for many readers, it would be interesting to see the

---

[2] For the definition of measure zero see Section 1.5.

[3] It can be shown that if $\mathcal{F}$ contains a constant function, then the above assumptions on $\mathcal{F}$ and t_n guarantee that the vectors $\boldsymbol{f}_n$ are pairwise different with probability one.

discovery process, even if it is somewhat redacted. Readers interested only in our achievements might want to skip this section altogether.

With a quick look at Equation (3.3), we can observe that if $\boldsymbol{f}_n$ and $\boldsymbol{g}_m$ form bases in their respective spaces, nothing interesting will occur: we need all JK measurements to recover matrix $\boldsymbol{C}$. Therefore, we are interested in frames. The tensor product of two frames is a frame in the product space. Hence, if we take all MN measurements, the recovery of $\boldsymbol{C}$ is unique. But what happens if we take JK measurements from this MN-element frame?

To investigate this, we generated random Gaussian frames, polynomial frames, and bandlimited frames (at random times). Then, we estimated how likely it is that JK measurements are sufficient to recover matrix $\boldsymbol{C}$. We chose JK measurements at random multiple times and obtained the following plots shown in Figure 3.1.

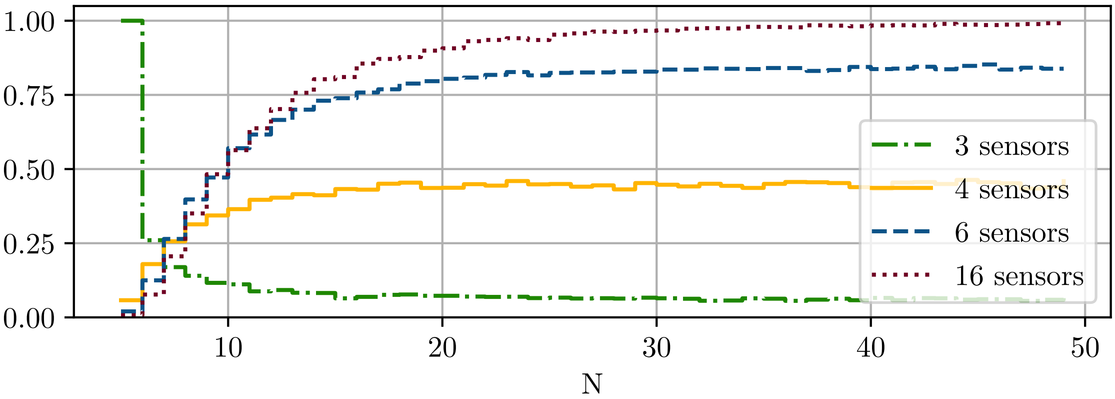


Figure 3.1.: Dependence of the rank probability for JK measurements on the number of points along the trajectory. Here $J = 3$ and $K = 5$. Note that there is, so far, no difference between vectors $\boldsymbol{g}$ and $\boldsymbol{f}$ and that if we swap J, M with K, N, the plots will look the same.

We observed two interesting behaviours. First, the plots did not seem to depend on the type of frames we used. Second, for large M the probability was monotonic with N, but this was not the case for small M.

In order to avoid repeating the same reasoning for each type of frame, we conjectured that the only assumption about the frames is that they have to be full-spark frames. This gave us one of our main assumptions. Full-spark property of the frames is needed, because all vectors $\boldsymbol{g}_m$ and $\boldsymbol{f}_n$ have to be interchangeable, i.e., their order cannot matter. Without this assumption, problems similar to Example 3.4 below can arise.

Example 3.4: A Singled-Out Coordinate

Consider a set of vectors $(\boldsymbol{g}_m)_1^M$ such that $\boldsymbol{g}_1 = [1, 0, \ldots, 0]^\top$ and $\boldsymbol{g}_1 \notin \operatorname{span}(\boldsymbol{g}_2, \ldots, \boldsymbol{g}_M)$. In this case, we need precisely K measurements $\boldsymbol{g}_1 \boldsymbol{f}_n^\top$ to recover the first row of $\boldsymbol{C}$. These measurements cannot be replaced by any other $\boldsymbol{g}_m$ and do not contribute to the recovery of any other rows of $\boldsymbol{C}$.

The non-monotonic behaviour of the plots is more difficult to explain. We observed that the probability seems to have a limit for large N. For a fixed number of measurements, N going to infinity means that the probability of having multiple measurements in one column goes to zero. This inspired us to investigate the case when one set of measurements does not repeat.

Roughly at the same time we observed that too many measurements in a row are redundant, see Example 3.5 below.

Example 3.5: More than K in a Row

Consider a set of measurements as in Equation (3.3), $|\mathcal{M}| = JK$. Without loss of generality, assume that the first $K+1$ elements of the first row of the mask $\boldsymbol{W}$ are non-zero. The space spanned by these $K+1$ measurements is only K dimensional:

$$\begin{aligned}\operatorname{span}\left(\boldsymbol{g}_1\boldsymbol{f}_1^\top,\dots,\boldsymbol{g}_1\boldsymbol{f}_{K+1}^\top\right) &= \operatorname{span}\left(\boldsymbol{g}_1\right)\otimes\operatorname{span}\left(\boldsymbol{f}_1,\dots,\boldsymbol{f}_{K+1}\right)\\ &= \operatorname{span}\left(\boldsymbol{g}_1\right)\otimes\mathbb{R}^K,\end{aligned}$$

because vectors $\boldsymbol{f}_1,\dots,\boldsymbol{f}_{K+1}$ are a frame in $\mathbb{R}^K$. Therefore, the space spanned by the all JK measurements is at most $JK-1$ dimensional, and the unique recovery of matrix $\boldsymbol{C}$ is not possible.

Combining this observation with the idea of taking $N\to\infty$, we conjectured that unique recovery is possible for any mask that has at most one non-zero element (n.z. element) in each column and that satisfies the *row condition*:

$$\sum_{m=1}^{M}\max(k_m,K) = JK, \tag{3.7}$$

where k_m is the number of n.z. elements in the m-th row of the mask $\boldsymbol{W}$. We plotted this condition and observed that its probability indeed matches the probability recovery for large N and that it is an upper bound for smaller N, see Figure 3.2. Note that here we estimate the probability by generating random sets of measurements. For curious readers, Box 3.10 explains how to calculate the probability directly.

This conjecture became Theorem 3.1, see Section 3.1.3. At first, we used a determinant-based proof, thus obtaining a measure-zero result. Naturally, we wondered if the measure zero is necessary. The answer is yes; there are examples when the bound above does not hold:

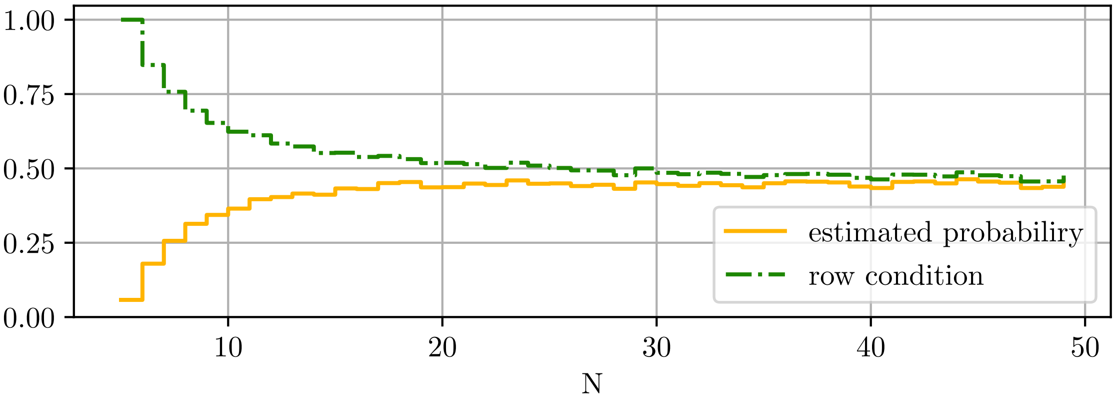


Figure 3.2.: The row condition (green, dotted-dashed) is an upper bound for the probability of matrix $\mathbf{\Gamma}$ being full rank. As the number of measurement times N increases, the bound becomes tighter.

Example 3.6: Almost all, but not all

Let $J = K$ and $\boldsymbol{g}_n = \boldsymbol{f}_n = [1, t_n, \ldots, t_n^{J-1}]$. Then the entries of a measurement matrix are also polynomials: $t_n^j t_n^k = t_n^{j+k}$ for $j = 0, \ldots J-1$ and $k = 0, \ldots J-1$.

There are only $2J - 1$ monomials of degree at most $2(J-1)$, hence the subspace spanned by the measurements matrix is at most $2J-1$ dimensional. However, to recover *symmetric* $\boldsymbol{C}$, we need $J(J-1)/2$ independent measurements.

Having solved a corner case, we returned to the general case when there is more than one n.z. element per column. In such a situation, Equation (3.11) is still a necessary but no longer a sufficient condition. Simply by symmetry — by swapping J and K — we found a dual necessary condition, called the *column condition*:

$$\sum_{n=1}^{N} \min(j_n, J) \geq JK, \tag{3.8}$$

where j_n is the number of n.z. elements in the n-th column of mask $\boldsymbol{W}$. In Figure 3.3, we can see that the probabilities of these two conditions upper bound the probability of unique recovery of $\boldsymbol{C}$. They also explain the qualitative difference between small and large M. For small M, the row condition dominates, whereas for large M, the column condition dominates.

We hoped that these two conditions together are sufficient, but this is not the case, as shown in Figure 3.3 and Example 3.7.

As our result for the corner case depends only on the mask $\boldsymbol{W}$ and not on specific vectors $\boldsymbol{f}_n$ and $\boldsymbol{g}_m$, we conjectured that this would also be the case in general. Our simulations did not disprove this, see Figure 3.4. It was difficult, however, to pinpoint which masks are "good" and which are "bad", and it seems that the problem is combinatorial in nature. We tried writing a reduction algorithm that would classify the masks, but we failed. This

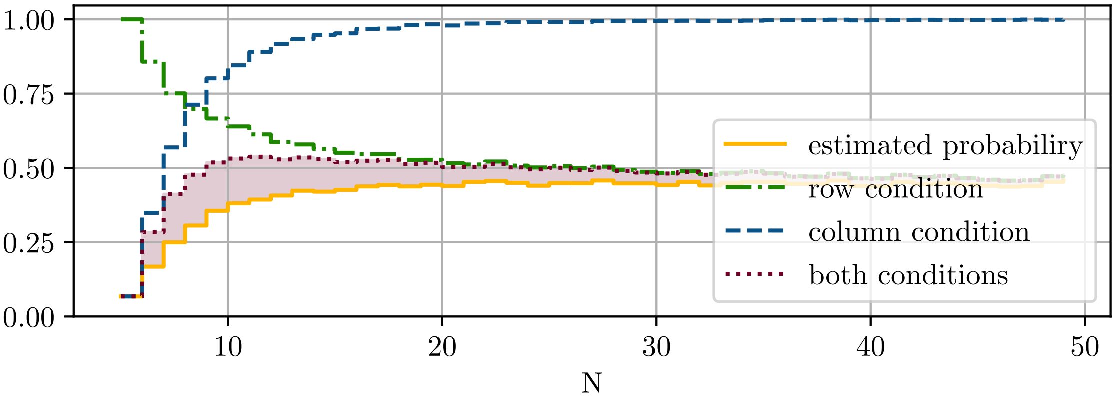


Figure 3.3.: Estimated probability of the following conditions: the row condition Equation (3.11) (orange, dashed), the column condition Equation (3.8) (green, dotted), both conditions (red, dotted-dashed), and the independence of measurement matrices (solid blue). All are plotted against number N of distinct measurement vectors $\boldsymbol{f}_n$. The number of available measurements is always exactly JK. We can see there is a gap (shaded), where both conditions are satisfied more often than $\boldsymbol{C}$ is unique.

did, however, prompted us to think about how to reduce the mask into another mask for which we already know if it is good. In time, this led us to the reduction method that we describe in Sections 3.1.3 and 3.1.4.

Due to the combinatorial nature of the problem, we decided to search for sufficient, but not necessary, conditions that would give an upper bound on the number of measurements needed. We knew that oversampling two times is sufficient. Example 3.7 shows that we cannot do much better than this. However, the determinant-based proof of Theorem 3.1 is difficult to generalise, because it requires exactly JK measurements. Here, the mask reduction ideas became convenient. We showed that $4JK$ is sufficient and, after a few iterations, we proved Theorem 3.3 that requires only $2JK$ oversampling.

Example 3.7: Hidden Dependence of Measurements

Assume that $\boldsymbol{W}$ is empty, except for two blocks of size $(J-1) \times (K-1)$ that contain only ones, and one block of size 1×1 that contain a one. Assume that blocks do not share rows nor columns. Let 1 to $J-1$ and 1 to $K-1$ be the coordinates of the first block, J to $2J-2$ and K to $2K-2$ be the coordinates of the second block, and $J-1$ and $K-1$ be the coordinates of the last (single entry) block. Such a mask has $2(J-1)(K-1)+1$ n.z. elements in total. Below, the matrices (b), (c) and (d) depict such masks for $K = J$ and $K = 3, 4$ and 5, respectively.

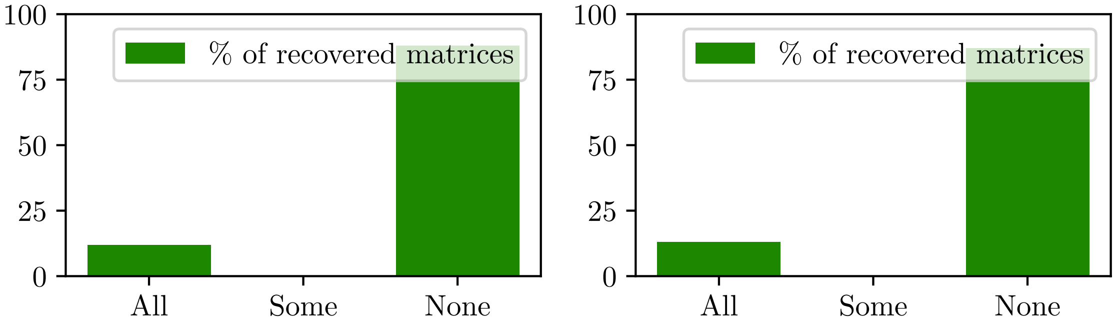
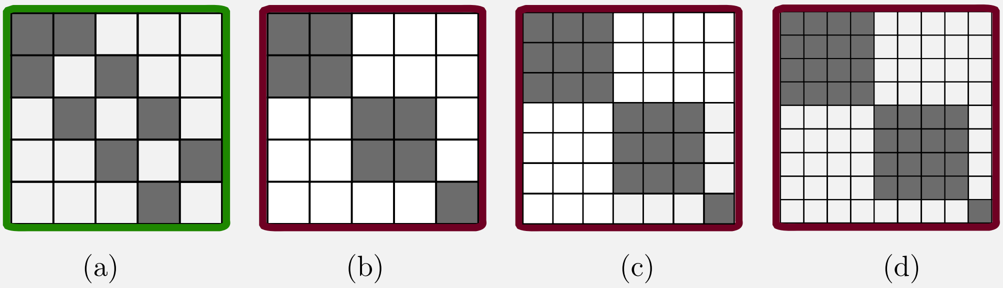


Figure 3.4.: For a given mask, either all recoveries are possible or none are. Both experiments were made for $K = 3$, $N = 6$, $J = 3$, $M = 4$. 100 different masks were generated and, for each mask, 100 example pairs of frames were generated from i.i.d. normal distribution. Then, matrix rank was calculated and, if it was $JK = 9$, then the recovery was counted as possible. On the left, the masks always had only 9 non-zero entries, whereas on the right, each matrix entry had the same probability $p = 0.4$ of being equal to one. We can see that, for some masks, all setups were recoverable, for some none were recoverable, but never anything in between.


(a) (b) (c) (d)

The sub-spaces $\mathcal{S}_1$ and $\mathcal{S}_1$ spanned by the first two blocks are

$$\begin{aligned}\mathcal{S}_1 &= \operatorname{span}\left(\boldsymbol{g}_1, \dots, \boldsymbol{g}_{J-1}\right) \otimes \operatorname{span}\left(\boldsymbol{f}_1, \dots, \boldsymbol{f}_{K-1}\right)\\ \mathcal{S}_2 &= \operatorname{span}\left(\boldsymbol{g}_J, \dots, \boldsymbol{g}_{2J-2}\right) \otimes \operatorname{span}\left(\boldsymbol{f}_K, \dots, \boldsymbol{f}_{2K-2}\right),\end{aligned}$$

respectively. If $\boldsymbol{G}$ (and $\boldsymbol{F}$) are full spark, then

$$\dim\left(\operatorname{span}\left(\boldsymbol{g}_1, \dots, \boldsymbol{g}_{J-1}\right) \cap \operatorname{span}\left(\boldsymbol{g}_J, \dots, \boldsymbol{g}_{2J-2}\right)\right) = J - 2,$$

and similarly for vectors $\boldsymbol{f}_n$. Then sub-spaces $\mathcal{S}_1$ and $\mathcal{S}_1$ share a $(J-2)(K-2)$ dimensional sub-space ($\dim(\mathcal{S}_1 \cap \mathcal{S}_2) = (J-2)(K-2)$). Therefore the total dimension of the span of the measurements of this mask $\boldsymbol{W}$ is at most $2(J-1)(K-1)-(J-2)(K-2)+1 = JK-1$ and is not sufficient for unique recovery of matrix $\boldsymbol{C}$.

Finally, inspired by applications such as video, we wondered whether our results can be extended to the recovery of an unknown tensor $\boldsymbol{T}$ of order-D. At the beginning, it was not clear how such an extension should even be formulated.

For an illustration, we consider a third-order tensor $\boldsymbol{T} \in \mathbb{R}^{J\times K\times L}$ and three frames $\boldsymbol{f}_n, \boldsymbol{g}_m, \boldsymbol{h}_p$ for $n \in [N], m \in [M]$, and $p \in [P]$ for some fixed N, M, P. Recall the row and column conditions we have considered so far (Equations (3.7) and (3.8)). The first intuition would be that we need one additional condition, similar to

$$\sum_{p=1}^{P} \min(l_p, \, ?\,) \leq \, ??,$$

where l_p is the number of times that each vector from the set $\boldsymbol{h}_p$ is used. But what do we replace the question marks ? and ?? with? We would like the conditions to be satisfied for setups, such as in Example 3.8, where it is clear that recovery of $\boldsymbol{C}$ is possible.

Example 3.8: Cube $J \times K \times J$

Consider the case when $\boldsymbol{f}_n, \boldsymbol{g}_m$ and $\boldsymbol{h}_p$ are all bases in their respective spaces. In this case, we need exactly JL vectors $\boldsymbol{f}_n$ — one for each $\boldsymbol{g}_m \otimes \boldsymbol{h}_p$ — exactly KL vectors $\boldsymbol{g}_m$ and exactly JK vectors $\boldsymbol{h}_{\boldsymbol{p}}$.

Therefore, the condition has to be

$$\sum_{p=1}^{P} \min(l_p, JK) \leq JKL,$$

and also conditions Equations (3.7) and (3.8) have to be updated to take into account the third dimension:

$$\sum_{n=1}^{N} \min(k_p, KL) \leq JKL \quad \text{and} \quad \sum_{m=1}^{M} \min(j_m, JL) \leq JKL.$$

Similarly as in the matrix case, we thought it might be easier to prove the case where vectors in one frame do not repeat. It turns out that, in this case, we can apply Corollary 3.6 (or condition Equation (3.7)) two times to obtain a sufficient result. Unfortunately, this result is not symmetric with respect to constants J, K, L, as illustrated in Example 3.9. In Section 3.1.3 we formulate a version of this example for an order-D tensor as Corollary 3.6.

Example 3.9: A Third-Order Tensor

Consider the case when each $\boldsymbol{f}_n$ is repeated at most once. Then, from Equation (3.7), every JK vectors $\boldsymbol{g}_{m_n} \otimes \boldsymbol{f}_n$, such that every vector $\boldsymbol{g}_m$ is used at most K times, form a basis of $\mathbb{R}^{J\times K}$. The matrix consisting of $\operatorname{vec}(\boldsymbol{g}_{m_n}\boldsymbol{f}_n^\top)$ is full spark. Since we assumed that each $\boldsymbol{f}_n$ is used at most once, then each $\operatorname{vec}(\boldsymbol{g}_{m_n}\boldsymbol{f}_n^\top)$ is used at most once. Applying

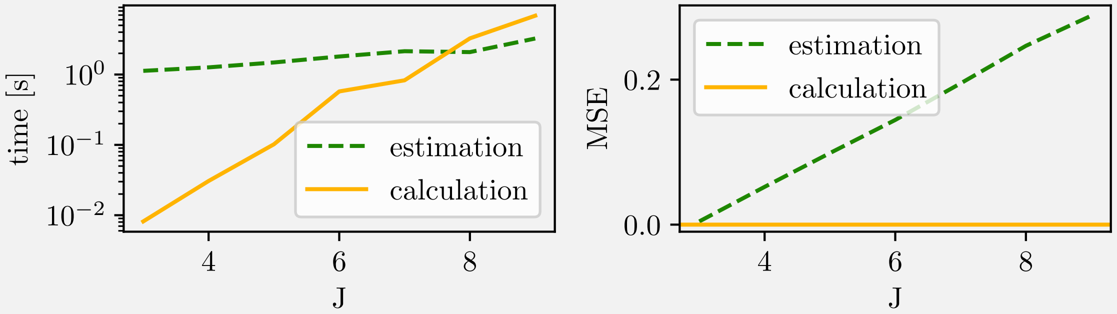
Equation (3.7) to frames $\mathrm{vec}(\boldsymbol{g}_{m_n}\boldsymbol{f}_n^\top)$ and $\boldsymbol{h}_p$, we obtain that if each vector $\boldsymbol{h}_p$ is repeated at most JK times, then JKL measurements are sufficient to recover our tensor $\boldsymbol{C}$.

Observe that the conditions in the example are not symmetric because we limit the number of repetitions of vectors $\boldsymbol{g}_m$ to K and number of vectors $\boldsymbol{h}_p$ to JK. This means that a different version of the result, with repetitions of vectors $\boldsymbol{g}_m$ limited to KL and repetitions of $\boldsymbol{h}_p$ limited to K, also has to be true. This, in turn, means that none of these results are necessary conditions, and that there is room for improvement.

We believe an extension of Algorithm 3.1 to D dimensions would be needed to obtain both sufficient and necessary condition. Although not impossible, this might not be worth the time until some applications pop out. For this reason, we do not propose any extension to Theorem 3.3, where extension of Algorithm 3.2 to D dimensions would require much care.

Want to know more? 3.10: Estimate or Calculate the Probability

We plotted estimated probabilities that Equation (3.7) is satisfied; for all purposes these plots were sufficient. We wondered if there is a formula for these probabilities. Could this formula help us calculate a lower bound on the probability that $\boldsymbol{C}$ is recoverable? Here, we give a formula for the probability. Unfortunately, it is not in closed form and takes time to calculate, which creates a trade-off between time and accuracy, see the figure below:



We assume that we take L ($L = 5J$ on the plot) measurements at random among MN measurements available in total ($10J^2$ measurements in the plot). The total number of possible measurement sets is $\binom{MN}{L}$. To calculate the probability that the measurement set satisfies Equation (3.7), we need to calculate the number of measurement sets that satisfy Equation (3.7). This condition depends on how many elements there are in each row of mask $\boldsymbol{W}$, but it does not depend on where exactly the elements are. Therefore, we introduce

Definition 3.3: Partition M-tuple

An M element tuple $\boldsymbol{k} = (k_1, \dots, k_M)$ such that

$$\sum_{m=1}^{M} k_m = L$$

is a partition M-tuple of L. If additionally, $K \geq k_m \geq 0$ for all $m \in [M]$, then we call this tuple K-limited.

For a fixed partition tuple, either all measurements that have corresponding numbers of elements satisfy Equation (3.7), or all do not satisfy this condition. Hence, we simply need to count the number of measurement corresponding to a given tuple. For row m we have N possible columns and we choose k_m out of them to be non-zero. Thus, we have $\prod_{m=0}^{M-1} \binom{N}{k_m}$ sets of measurements. Let $\mathcal{K}_{ok}$ be a set of all K-limited partition tuples.

Then the total number of valid measurements satisfying Equation (3.7) is

$$\sum_{\boldsymbol{k} \in \mathcal{K}_{ok}} \prod_{m=0}^{M-1} \binom{N}{k_m}. \tag{3.9}$$

Unfortunately, calculating this sum is quite slow because there are many partitions. We can accelerate the computations a little by grouping partitions that have the same entries (but in a different order). To do this, we would need to iterate over ordered partitions $\mathcal{K}_{oo}$:

$$\sum_{\boldsymbol{k} \in \mathcal{K}_{oo}} F(\boldsymbol{k}) \prod_{m=0}^{M-1} \binom{N}{k_m},$$

where $F(\boldsymbol{k})$ is a frequency of partition $\boldsymbol{k}$; that is to say, the number of partitions with the same elements. If $\boldsymbol{k}$ has p unique elements, its frequency is

$$F(\boldsymbol{k}) = \frac{M!}{u_1(\boldsymbol{k})! \cdot \dots \cdot u_p(\boldsymbol{k})!}$$

where $u_1(\boldsymbol{k}), \dots, u_p(\boldsymbol{k})$ are the number of repetitions of the unique elements. We can generate all the ordered partitions recursively in the inverse lexicographical order. Consider all ordered partition M-tuples of L that have a first element k_0. We can generate them by generating all k_0-limited partition $(M-1)$-tuples of $(L - k_0)$. This is the basis of the PARTITION function that is recursively called in the pseudo-code below in order to generate all the partitions.

```
function PARTITIONS($L, M, k$)
    $\boldsymbol{p}$ := empty list of $M$-tuples
    if $L = 0$ then
```

```
        append $\underbrace{(0,\dots,0)}_{M}$ to $\boldsymbol{p}$
    else if $Mk \geq L$ then
        for $i \in \{\max(0, L-k), \dots, L\}$ do
            for $\boldsymbol{q} \in \textsc{partitions}(i, M-1, L-i)$ do
                append $(L-i, q_1, \dots, q_{M-1})$ to $\boldsymbol{p}$
            end for
        end for
    end if
    return $\boldsymbol{p}$
end function
```

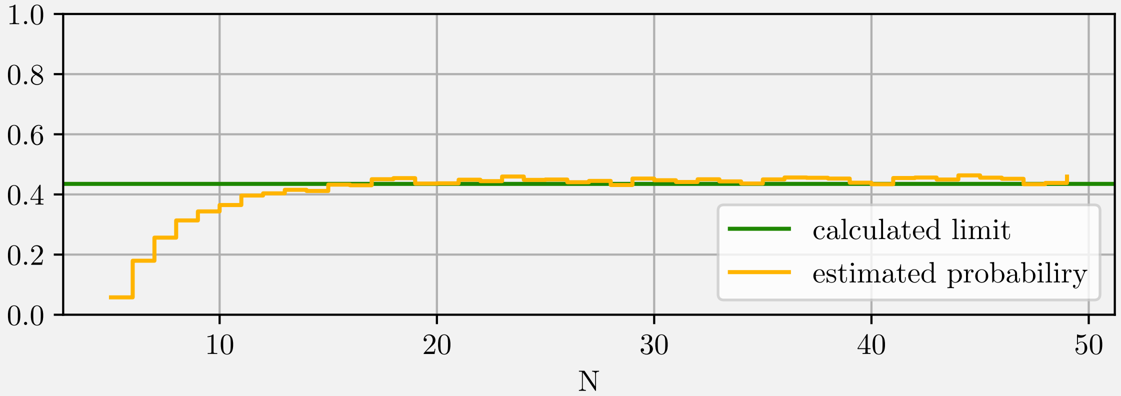
We were unable to find a closed-form expression for Equation (3.9), and therefore we studied its limits and approximations. To approximate binomial coefficients, we could use Stirling's approximation, but the results are ugly and not worth the space. A better approach is to consider the $N \to 0$ as we did before. In this case, the probability that Equation (3.9) is satisfied can be approximated by the probability of Equation (3.9), given one measurements per column.

Note that we cannot directly take a limit $N \to \infty$ because the limit of uniform (discrete) random variable does not exist for $N \to \infty$. Yet we can still calculate the limit of the probability as a number for $N \to \infty$. An interesting exercise would be to consider some discrete distribution that has a limit: for example, a Bernoulli distribution that becomes a Poisson distribution in the limit, see Poisson Limit Theorem [258].

When $N \to \infty$, the total number of measurement sets is M^L because rows are assigned independently for each measurement. The number of measurement sets that lead to partition $\boldsymbol{k}$ is the number of ways of splitting L into M groups $k_0, \dots, k_{M-1}$ when the order of elements in the group does not matter, thus we obtain the limit:

$$\sum_{\boldsymbol{k} \in \mathcal{K}_{ok}} \frac{L!}{M^L \prod_{m=1}^{M} k_m!}$$

Observe that this expression does not depend on N, hence it passes a sanity check as a limit of Equation (3.9) and, in the figure below, we can see that it indeed aligns with the probability on the right side of the plot.



Finally, let us note that instead of picking L measurements at random, we could pick every measurement with probability p. This would correspond to the erasure channel model described in Section 3.2.1. We considered this route, but the plots seem much less interesting, because most of the behaviour is explained simply by the expected value of the number of measurements (if it is larger than JK).

3.1.3. Theorems

In this section, we present our two main results, Theorem 3.1 and Theorem 3.3, together with Corollary 3.6 and some of the lemmas needed to prove them. We state these lemmas here because we believe they provide further insight into the problem. For this reason, we also present Lemma 3.2. It would be redundant if the sole purpose were to prove Theorem 3.1, but can be a starting point for new considerations and results. We postpone proofs to Section 3.1.4, presenting here only sketches of proofs of Theorem 3.1.

Theorem 3.1: One Set of Unique Vectors

Assume $\boldsymbol{G}$ is full spark and $\boldsymbol{F}$ follows the polynomial model. If in each row of $\boldsymbol{W}$ there are at most K n.z. elements and in each column of $\boldsymbol{W}$ there is at most one n.z. element, then for almost all $(z_n)_{n=1}^{N}$,

$$JK \text{ measurements} \tag{3.10}$$

are necessary and sufficient for the unique recovery of matrix $\boldsymbol{C}$.

The above theorem can also be formulated as the *row condition* on the number of measurements in each row of the mask $\boldsymbol{W}$ (Section 3.1.2). More precisely, assume that $\boldsymbol{G}$ is full spark, $\boldsymbol{F}$ follows the polynomial model, and that each column of $\boldsymbol{W}$ contains at most one n.z. element. If

$$\sum_{m=1}^{M} \min(k_m, K) \geq JK, \tag{3.11}$$

where k_m is the number of n.z. elements in the m-th row of $\boldsymbol{W}$, then matrix $\boldsymbol{C}$ can be uniquely recovered using measurements $\mathcal{M}$. Moreover, if Equation (3.11) is not satisfied, then the solution to Equation (3.1) is not unique.

Indeed, if the assumptions of Theorem 3.1 are satisfied, then for every row m the number of n.z. elements $k_m \leq K$, thus Equation (3.11) becomes the sum of all elements, $\sum_{m=1}^{M} k_m = JK$. Similarly, if we assume that Equation (3.11) is satisfied, we can remove from every row any extra elements above the first K and obtain the assumptions of Theorem 3.1.

We show two methods to prove Theorem 3.1. Both ways use the fact that if we have $N = JK$ measurements, we can calculate the determinant of matrix $\boldsymbol{\Gamma}$ defined in Equation (3.5). The first method relies on the determinant only, hence we call it *the determinant method.* The second one uses the determinant indirectly and, in our opinion, leads us to better geometric intuitions. This second method is based on the reduction of a given set of measurements to a set that we already know allows for a unique recovery of matrix $\boldsymbol{C}$, hence we call it *the reduction method.*

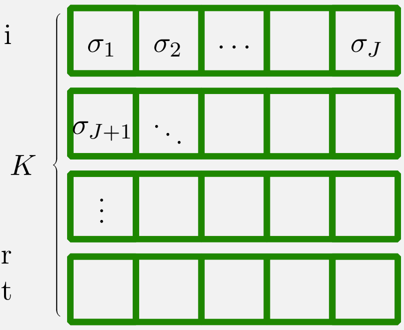
We first explain the determinant method. If the reader is not interested in this, they can resume reading at Theorem 3.3. To formulate the basis of the determinant method — Lemma 3.2 — we first define a partition of set $[JK]$.

Definition 3.4: Partition

A *partition* $\boldsymbol{\xi}$ is family of K J-elements sets ($\xi_k \in \boldsymbol{\xi}$), i $|\xi_k| = J$ such that:

$$\bigcup_{k}^{K} \xi_k = [JK].$$

The important part about this definition is that the order of the sets ξ_k and the order within these sets does not matter.



We also define, based on the sign of a permutation, a sign of a partition. We discuss permutations in more detail in Box 3.11. Here we define the sign of $\boldsymbol{\xi}$ to be the sign of the first in lexicographical order permutation $\boldsymbol{\sigma}$ that “matches” $\boldsymbol{\xi}$:

$$\operatorname{sgn}(\boldsymbol{\xi}) = \operatorname{sgn}(\boldsymbol{\sigma}) \text{ for } \boldsymbol{\xi} = \left\{ \{\sigma_{J(k-1)+j} : j \in [J]\} : k \in [K] \right\}.$$

Lemma 3.2: Permutations

Let $\boldsymbol{\Gamma}$ be the matrix of vectorised measurements with $N = JK$, and let Ξ be the set of partitions of $[JK]$ into K bins of J elements. Then, the determinant of $\boldsymbol{\Gamma}$ can be expressed as

$$\det(\boldsymbol{\Gamma}) = \sum_{\boldsymbol{\xi} \in \Xi} \operatorname{sgn}(\boldsymbol{\xi}) \alpha(\boldsymbol{F}^{\boldsymbol{\xi}}) \left(\prod_{k=0}^{K-1} \det(\boldsymbol{G}_{\boldsymbol{\xi},k}) \right), \tag{3.12}$$

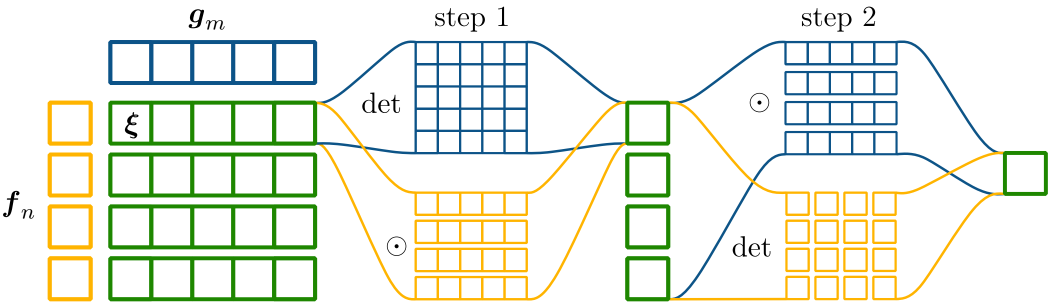


Figure 3.5.: How to create matrices $\boldsymbol{G}_{\boldsymbol{\xi}}$ (top row, blue) and $\boldsymbol{F}^{\boldsymbol{\xi}}$ (bottom row, yellow). On the left rows depict a partition $\boldsymbol{\xi}$ of $N = JK$ into K groups of J elements (green). Vectors $\boldsymbol{g}_m$ and $\boldsymbol{f}_n$ are depicted as a reminder that the size constants J and K correspond to their sizes. In the first step, we consider measurement vectors indexed by one row of the partition $\boldsymbol{\xi}$, and in the second step we consider all the rows of $\boldsymbol{\xi}$. In the case of vectors $\boldsymbol{g}$, we first take the determinant of a block of J vectors, then multiply the results. In the case of vectors $\boldsymbol{f}$, we first apply point-wise multiplication $\odot$ then calculate the determinant.

where $\boldsymbol{G}_{\boldsymbol{\xi},k}$ is a $J \times J$ matrix created by stacking vectors $\boldsymbol{g}_m$ such that $m \in \xi_k$, $\boldsymbol{F}^{\boldsymbol{\xi}}$ is a $K \times K$ matrix created by stacking vectors $\bigodot_{j \in \xi_k} \boldsymbol{f}_j$ and α is a *determinant* if J is odd and a *permanent* if J is even.

Equation (3.12) in the lemma is convoluted, but all the reader needs to know is that we multiply the determinants of groups of vectors $\boldsymbol{g}_m$. The factors $\alpha(\boldsymbol{F}^{\boldsymbol{\xi}})$ are simply some polynomials in entries of vectors $\boldsymbol{f}_n$. Note that, for different $\boldsymbol{\xi}$, the factors $\alpha(\boldsymbol{F}^{\boldsymbol{\xi}})$ are different. For more intuition on matrices $\boldsymbol{G}_{\boldsymbol{\xi},k}$ and $\boldsymbol{F}^{\boldsymbol{\xi}}$, see Figure 3.5.

But how does Lemma 3.2 give us Theorem 3.1? Below, we provide the intuition how to prove it.

Sketch of a Proof of Theorem 3.1

In Equation (3.12), $\det\{\Gamma\}$ is a polynomial in entries of $\boldsymbol{f}_n$ thus a polynomial in $\boldsymbol{z}_n$. As vectors $\boldsymbol{f}_n$ do not repeat, $\det\left(\boldsymbol{F}^{\boldsymbol{\xi}}\right)$ are different monomials of order JK. Therefore, $\det(\boldsymbol{\Gamma})$, as a function of $\boldsymbol{z}_n$, either has a measure zero set of zeros and we are done, or it is constantly zero. But this would mean that all products

$$\boldsymbol{G}_{\boldsymbol{\xi}} = \prod_{k=0}^{K-1} \det(\boldsymbol{G}_{\boldsymbol{\xi},k}) \tag{3.13}$$

are zero. This would mean that for every $\boldsymbol{\xi}$ there is k such that $\det(\boldsymbol{G}_{\boldsymbol{\xi},k}) = 0$, hence vectors in k-th group of the partition $\boldsymbol{\xi}$ are not linearly independent. Since $\boldsymbol{G}$ has full spark, this means that at least one of the columns in $\boldsymbol{G}_{\boldsymbol{\xi},k}$ repeats *in every partition* $\boldsymbol{\xi}$.

From the pigeonhole principle, this means that at least one of the vectors $\boldsymbol{g}_n$ is used at least K times, which proves Theorem 3.1. ■

Note that although Lemma 3.2 is general, the above proof requires vectors $\boldsymbol{f}_n$ to be pairwise different. This means it cannot be easily generalised. When $\boldsymbol{f}_n$ repeat, some of the $\boldsymbol{F}^{\boldsymbol{\xi}}$ will be the same, and we will have to consider sums of $\det(\boldsymbol{G}_{\boldsymbol{\xi},k})$ for different $\boldsymbol{\xi}$. This reflects combinatorial problems described in Section 3.1.2. To avoid these problems, we oversample the system to obtain a simple and sufficient, but not necessary, condition:

Theorem 3.3: Oversampling

Assume that $\boldsymbol{G}$ is full spark and $\boldsymbol{F}$ follows the polynomial model. If in each row of $\boldsymbol{W}$ there are at most K n.z. elements and, in each column of $\boldsymbol{W}$, there are at most J n.z. elements, then, for almost all $(z_n)_{n=1}^N$,

$$2JK \text{ measurements} \tag{3.14}$$

are sufficient for the unique recovery of matrix $\boldsymbol{C}$.

If the reader is interested in the discussion about why the constant 2 in Equation (3.14) is optimal, check Example 3.7 in Section 3.1.2.

As Lemma 3.2 requires exactly JK measurements, we cannot use it to prove Theorem 3.3. Therefore, we propose a reduction method that can prove both theorems. We first state the lemmas needed to prove Theorem 3.1. The first one describes a simple setup when the recovery is possible and is of course a sufficient, but not necessary, condition. In the proof of Theorem 3.3, we reduce any other masks to this setup.

Lemma 3.4: Minimal Setup

Let $\boldsymbol{G}$ and $\boldsymbol{F}$ be full spark. If the mask $\boldsymbol{W}$ contains K columns with J non-zero elements, then matrix $\boldsymbol{C}$ can be recovered from measurements Equation (3.1). Analogous statement is true with rows and columns swapped.

The main insight that the reduction method is based on is that if we decrease the number of random vectors, e.g., by replacing every occurrence of $\boldsymbol{f}_N$ by $\boldsymbol{f}_1$, we can only decrease the dimension of the span of the measurement matrices and never increase it.

Lemma 3.5: Replacing Vectors from Polynomial Model

Consider a set of measurements $\mathcal{M}$ as in Equation (3.3), $|\mathcal{M}| = JK$ and let $\boldsymbol{F}$ follow the polynomial model, $\boldsymbol{f}_n = \boldsymbol{p}(\boldsymbol{z}_n)$. Create a reduced set of indices , $\mathcal{M}_r$, by replacing all index pairs $(\cdot, N)$ with $(\cdot, 1)$, in order for the second indices n in $\mathcal{M}_r$ to be in $[N-1]$.

If measurements indexed by $\mathcal{M}_r$ span the whole space $\mathbb{R}^J \otimes \mathbb{R}^K$, then the measurements indexed by $\mathcal{M}$ span $\mathbb{R}^J \otimes \mathbb{R}^K$ for almost all $(\boldsymbol{z}_n)_{n=1}^N$.

Let us now sketch another proof of Theorem 3.1. For the somewhat longer full proof, see Section 3.1.4.

Another Sketch of a Proof of Theorem 3.1

When each index n is used only once or, equivalently, when each column of $\boldsymbol{W}$ contains at most one n.z. element, replacing vector $\boldsymbol{f}_n$ changes only one measurement. If there are at most K elements in each row, we can replace measurements one by one until we fill the first column with J n.z. elements. Then, we repeat this to create K columns with J n.z. elements each, to satisfy the assumptions of Lemma 3.4. ■

In the above proof, we cannot afford to "lose" any measurements; that is to say, we begin with an independent JK measurement and need an independent JK measurement to satisfy Lemma 3.4. In the proof of Theorem 3.3, during the reduction process, we can afford to make some measurements dependent. At the same time, we can no longer reduce one measurement at the time — we have to reduce the whole column, which can contain up to J measurements.

The proof of Theorem 3.3 is therefore similar in flavour, but requires two additional technical lemmas described in Section 3.1.4; they give a method of joining/replacing columns without wasting too many measurements.

Theorem 3.1 can be also generalised to higher dimensions. Sometimes, instead of recovering a matrix $\boldsymbol{C}$ we would like to recover an order-D tensor $\boldsymbol{T}$ from measurements made with rank-one tensors. Tensor $\boldsymbol{T}$ is useful for describing videos or other data collected on a surface or in space over time; for instance, the output from a robo-skin or atmospheric data. For examples of how to use it in practice, see Section 3.2.

For an unknown order-D tensor $\boldsymbol{C}$, if one set of measurement vectors is unique, we can apply Theorem 3.1 $D-1$ times and obtain

Corollary 3.6: Generalisation of Theorem 3.1

Assume that we measure a rank-D tensor $\boldsymbol{C}$ using D full spark frames:

$$\left(\boldsymbol{g}_n^{(1)}\right)_{n=1}^{N_1}, \dots, \left(\boldsymbol{g}_n^{(D)}\right)_{n=1}^{N_D}$$

spanning, respectively, $\mathbb{R}^{K_1}$ to $\mathbb{R}^{K_D}$. Furthermore, assume that the first frame $\left(\boldsymbol{g}_n^{(1)}\right)_{n=1}^{N_1}$ follows the polynomial model. Then, the set of $\prod_{d=1}^{D} K_d$ measurements such that there are at most $\prod_{j=1}^{d-1} K_j$ repeated vectors of the frame number d is sufficient to uniquely recover tensor $\boldsymbol{C}$.

Note that, in the corollary above, we have $\prod_{j=1}^{0} K_j = 1$ and that vectors in the first frame are unique, as in Theorem 3.1. However, the result is not symmetric with respect to remaining dimensions $2, \dots, D$. We find it unintuitive and further discuss it at the end of Section 3.1.2.

In summary, under the full-spark assumption and the polynomial model (Definitions 3.1 and 3.2), mask $\boldsymbol{W}$ can be used to assess whether matrix $\boldsymbol{C}$ can be recovered uniquely. More precisely, Theorem 3.1 gives a condition that is both sufficient and necessary, but it requires at most one n.z. element per column of $\boldsymbol{W}$. Theorem 3.3 gives only a sufficient condition, hence only an upper bound on the number of measurements needed, but it allows up to J n.z. elements per column of $\boldsymbol{W}$. Finally, Theorem 3.1 generalises to higher dimensions, but we are not aware of any generalisation of Theorem 3.3.

3.1.4. Proofs

In this section, we provide proofs of the results from Section 3.1.3. We begin with a somewhat ugly proof of Lemma 3.2. We do formalise the hand-waved proof of Theorem 3.1 based on Lemma 3.2 because we deem it sufficiently good. Therefore, we proceed to prove Theorem 3.1, based on Lemmas 3.4 and 3.5 and Algorithm 3.1. Only then can we prove Lemmas 3.4 and 3.5.

Subsequently, we prove Theorem 3.3 with the necessary lemmas. The structure is very similar to the second proof of Theorem 3.1, i.e., we use the reduction method. Finally, we present a short proof of Corollary 3.6. For convenience, we restate the corresponding results, prior to giving the proofs.

Lemma 3.3: Permutations

Let $\mathbf{\Gamma}$ be the matrix of vectorised measurements with $N = JK$, and let Ξ be the set of partitions of $[JK]$ into K bins of J elements. Then, the determinant of $\mathbf{\Gamma}$ can be expressed as

$$\det(\mathbf{\Gamma}) = \sum_{\boldsymbol{\xi}\in\Xi} \operatorname{sgn}(\boldsymbol{\xi})\alpha(\boldsymbol{F}^{\boldsymbol{\xi}}) \left(\prod_{k=0}^{K-1} \det(\boldsymbol{G}_{\boldsymbol{\xi},k})\right), \tag{3.15}$$

where $\boldsymbol{G}_{\boldsymbol{\xi},k}$ is a $J \times J$ matrix created by stacking vectors $\boldsymbol{g}_m$ such that $m \in \xi_k$, $\boldsymbol{F}^{\boldsymbol{\xi}}$ is a $K \times K$ matrix created by stacking vectors $\bigodot_{j\in\xi_k} \boldsymbol{f}_j$ and α is a *determinant* if J is odd and a *permanent* if J is even.

Proof of Lemma 3.2

We use the definition of the determinant, based on permutations (see Box 3.11):

$$\det(\mathbf{\Gamma}) = \sum_{\boldsymbol{\sigma}\in\mathcal{P}_N} \operatorname{sgn}(\boldsymbol{\sigma}) \prod_{n=1}^{N} \gamma_{n,\sigma_n}.$$

We consider only one element of this sum, for a fixed permutation $\boldsymbol{\sigma}$. Recall from Equation (3.5) that $\boldsymbol{\gamma}_n = \operatorname{vec}(\boldsymbol{g}_n\boldsymbol{f}_n^\top)$, thus $\gamma_{(J(k-1)+j),n} = f_{k,n}g_{j,n}$, as illustrated in Figure 3.6. If we change variables $n = J(k-1)+j$ and separate the factors dependent

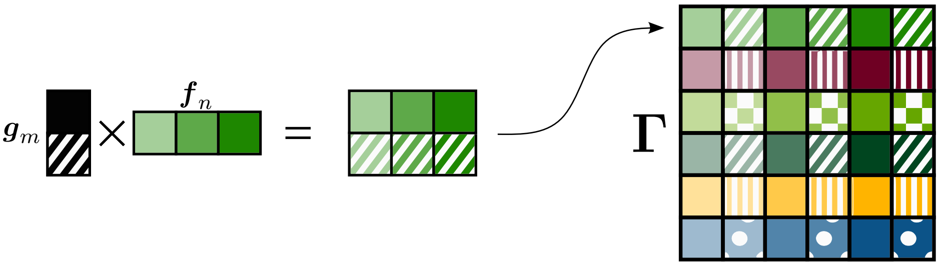


Figure 3.6.: Structure of the matrix $\boldsymbol{\Gamma}$: as it is created from vectorised products of vectors $\boldsymbol{g}_m$ and $\boldsymbol{f}_n$, every J-th (in this case every second) element of every row is a product of the first element of some vector $\boldsymbol{g}_m$, $g_{1,m}$ and an element of $\boldsymbol{f}_n$. At the same time, first J elements of every row are products of the first element of some vector $\boldsymbol{f}_n$, $f_{1,n}$ with an element of $\boldsymbol{g}_m$.

on the different frames, we obtain:

$$\prod_{n=1}^{N} \gamma_{n,\sigma_n} = \underbrace{\left(\prod_{k=1}^{K}\prod_{j=1}^{J} f_{k,\sigma_{J(k-1)+j}}\right)}_{a_{\boldsymbol{\sigma}}} \underbrace{\left(\prod_{k=1}^{K}\prod_{j=1}^{J} g_{j,\sigma_{J(k-1)+j}}\right)}_{b_{\boldsymbol{\sigma}}}. \tag{3.16}$$

We now analyse factors $a_{\boldsymbol{\sigma}}$ and $b_{\boldsymbol{\sigma}}$ separately. Before we do this, consider the relation between partitions and permutations. There is a natural way to map permutations into partitions, as illustrated in Figure 3.6:

$$\boldsymbol{\sigma} \mapsto \boldsymbol{\xi} = \{\{\sigma_{kJ+j} : j \in [J]\} : k \in [K]\} \tag{3.17}$$

All permutations that map to the same partition $\boldsymbol{\xi}$ can be described by applying a smaller permutation to $\boldsymbol{\xi}$. More precisely, we need to define the order (a permutation $\boldsymbol{\pi} \in \mathcal{P}_K$) on the sets in $\boldsymbol{\xi}$, and k permutations within each set $\boldsymbol{\xi}$, $\boldsymbol{\tau}^k \in \mathcal{P}_J$, $k \in [K]$. Index-wise, we can write

$$\sigma_{J(k-1)+j} = (\xi_{\pi_k})_{\tau^k_j}. \tag{3.18}$$

Moreover, we can write the sign of the permutation $\boldsymbol{\sigma}$ as

$$\operatorname{sgn}(\boldsymbol{\sigma}) = \operatorname{sgn}(\boldsymbol{\xi}) \operatorname{sgn}(\boldsymbol{\pi})^J \operatorname{sgn}(\boldsymbol{\tau}^1) \cdots \operatorname{sgn}(\boldsymbol{\tau}^K).$$

We therefore split the summation of $\boldsymbol{\sigma}$ to summation over $\boldsymbol{\xi}$ and the smaller permutations.

Consider the first term in Equation (3.16):

$$a_{\boldsymbol{\sigma}} = \prod_{k=1}^{K}\prod_{j=1}^{J} f_{k,(\xi_{\pi_k})_{\tau^k_j}} = \prod_{k=1}^{K}\prod_{j=1}^{J} f_{k,(\xi_{\pi_k})_j}$$

where the first equality is based on Equation (3.18) and the second results from the fact that $a_{\boldsymbol{\sigma}}$ depends on the index j only through permutation $\boldsymbol{\tau}^k$. Since $\boldsymbol{\tau}^k$ changes only the order of multiplication, then $a_{\boldsymbol{\sigma}}$ does not depend on $\boldsymbol{\tau}^k$ but only on $\boldsymbol{\xi}$ and $\boldsymbol{\pi}$. Summing both sides over all possible $\boldsymbol{\pi}$, we obtain

$$\sum_{\boldsymbol{\pi}\in\mathcal{P}_K} \operatorname{sgn}(\boldsymbol{\pi})^J a_{\boldsymbol{\sigma}} = \sum_{\boldsymbol{\pi}\in\mathcal{P}_K} \operatorname{sgn}(\boldsymbol{\pi}) \prod_{k=1}^{K}\prod_{j=1}^{J} f_{k,(\xi_{\pi_k})_j} = \det\Big(\boldsymbol{F}^{\boldsymbol{\xi}}\Big)$$

for an odd J and for an even J we obtain the permanent:

$$\sum_{\boldsymbol{\pi}\in\mathcal{P}_K} \operatorname{sgn}(\boldsymbol{\pi})^J a_{\boldsymbol{\sigma}} = \sum_{\boldsymbol{\pi}\in\mathcal{P}_K} \prod_{k=1}^{K}\prod_{j=1}^{J} f_{k,(\xi_{\pi_k})_j} = \operatorname{perm}\Big(\boldsymbol{F}^{\boldsymbol{\xi}}\Big)$$

Similarly, the second term does not depend on $\boldsymbol{\pi}$, but here we do not have problems with the sign

$$b_{\boldsymbol{\sigma}} = \prod_{k=1}^{K}\prod_{j=1}^{J} g_{j,(\xi_{\pi_k})_{\tau^k_j}} = \prod_{k=1}^{K}\prod_{j=1}^{J} g_{j,(\xi_k)_{\widetilde{\tau}^k_j}},$$

where we have introduced a change of variables $\widetilde{\boldsymbol{\tau}}^{\pi_k} = \boldsymbol{\tau}^k$ to remove the dependence on the permutation $\boldsymbol{\pi}$. We now drop the hat over $\boldsymbol{\tau}^k$ to simplify the notation. We can write

$$b_{\boldsymbol{\sigma}} = \left(\prod_{j=1}^{J} g_{j,(\xi_1)_{\tau^1_j}}\right)\left(\prod_{k=2}^{K}\prod_{j=1}^{J} g_{j,(\xi_k)_{\tau^k_j}}\right).$$

But the first term, summed over all possible $\boldsymbol{\tau}^1$ is

$$\sum_{\boldsymbol{\tau}^1\in\mathcal{P}_J} \operatorname{sgn}(\boldsymbol{\tau}^1)\prod_{j=1}^{J} g_{j,(\xi_1)_{\tau^1_j}} = \det(\boldsymbol{G}_{\boldsymbol{\xi},1}).$$

Applying this trick for the remaining $K-1$ permutations τ^k, we obtain that the sum of the terms b over all permutations $\boldsymbol{\tau}$ is exactly

$$\sum_{\boldsymbol{\tau}^1\in\mathcal{P}_J} \operatorname{sgn}(\boldsymbol{\tau}^1)\cdots\sum_{\boldsymbol{\tau}^K\in\mathcal{P}_J} \operatorname{sgn}(\boldsymbol{\tau}^K) b_{\boldsymbol{\sigma}} = \prod_{k=1}^{K}\det(\boldsymbol{G}_{\boldsymbol{\xi},k}).$$

To obtain the sum over all permutations $\boldsymbol{\sigma}$, we need to first sum over all $\boldsymbol{\tau}_k$ for each $k\in[K]$, then over all $\boldsymbol{\pi}$, and finally over all partitions $\boldsymbol{\xi}$. Moreover, we have $\operatorname{sgn}(\boldsymbol{\sigma}) = \operatorname{sgn}(\boldsymbol{\xi})\operatorname{sgn}(\boldsymbol{\pi})^J\operatorname{sgn}(\boldsymbol{\tau}^1)\cdots\operatorname{sgn}(\boldsymbol{\tau}^K)$. Combining all these observations we obtain

$$\begin{aligned}\det(\boldsymbol{\Gamma}) &= \sum_{\boldsymbol{\sigma}\in\mathcal{P}_{JK}} \operatorname{sgn}(\boldsymbol{\sigma}) a_{\boldsymbol{\sigma}} b_{\boldsymbol{\sigma}} \\ &= \sum_{\boldsymbol{\xi}\in\Xi} \operatorname{sgn}(\boldsymbol{\xi})\alpha(\boldsymbol{F}^{\boldsymbol{\xi}})\prod_{k=1}^{K}\det(\boldsymbol{G}_{\boldsymbol{\xi},k}).\end{aligned}$$

■

This concludes the determinant method of proving Theorem 3.1. Now, we present the reduction method. We begin with the formal proof of Theorem 3.1, leaving the proofs of the lemmas for later.

Theorem 3.3: One Set of Unique Vectors

Assume $\boldsymbol{G}$ is full spark and $\boldsymbol{F}$ follows the polynomial model. If in each row of $\boldsymbol{W}$ there are at most K n.z. elements and in each column of $\boldsymbol{W}$ there is at most one n.z. element, then for almost all $(z_n)_{n=1}^{N}$,

$$JK \text{ measurements} \tag{3.19}$$

are necessary and sufficient for the unique recovery of matrix $\boldsymbol{C}$.

The proof we present here is based on the mask-reduction algorithm (Algorithm 3.1) and Lemma 3.4. More precisely, we show that if $\boldsymbol{W}$ satisfies the assumptions of Theorem 3.1, then a sequence of replacements given by Algorithm 3.1 results in a mask satisfying the assumptions of Lemma 3.4. Then, as the algorithm makes replacements that are consistent with Lemma 3.5, matrix $\boldsymbol{C}$ can be reconstructed from the original measurement set $\mathcal{M}$ for almost all latent variables $(\boldsymbol{z}_n)_{n=1}^{N}$.

Algorithm 3.1: Matrix reduction algorithm for Theorem 3.1

Input: Mask $\boldsymbol{W} \in \mathbb{R}^{M \times N}$, satisfying the assumptions of Theorem 3.1, constants J and K

Output: Modified $\boldsymbol{W}$, satisfying the assumptions of Lemma 3.4

```
for $k \in [K]$ do
    for $j \in [J]$ do
        $m := \arg\max_{i: w_{i,k}=0} \sum_{j=k}^{N} w_{i,j}$
        set $n > k$ such that $w_{m,n} = 1$
        if $j = 1$ then
            $(\boldsymbol{w}_k, \boldsymbol{w}_n) := (\boldsymbol{w}_n, \boldsymbol{w}_k)$
        else
            $\boldsymbol{w}_k := \boldsymbol{w}_k \vee \boldsymbol{w}_n$
            $\boldsymbol{w}_n := \boldsymbol{0}$
        end if
    end for
end for
```

Proof of Theorem 3.1

Without any additional assumptions, JK measurements are necessary for recovering $\boldsymbol{C} \in \mathbb{R}^{J \times K}$, because we need JK linear equations to uniquely recover JK unknowns. We show that, under the assumptions of Theorem 3.1, JK measurements are sufficient.

Observe that Lemma 3.5 can be written in terms of mask $\boldsymbol{W}$. Indeed, replacing all occurrences of a vector $\boldsymbol{f}_n$ with a vector $\boldsymbol{f}_k$ corresponds to replacing column k of $\boldsymbol{W}$

with the pointwise *logical or* of columns number k and n ($\boldsymbol{w}_k := \boldsymbol{w}_k \vee \boldsymbol{w}_n$) and replacing column n with zeros ($\boldsymbol{w}_n = \boldsymbol{0}$).

To prove that the algorithm is correct, we consider two invariants:

1) at the beginning of step k, each of the first $k-1$ columns of $\boldsymbol{W}$ contains J n.z. elements;
2) at the beginning of step k, there are at most $K-k+1$ n.z. elements in each row of $[\boldsymbol{W}]_{k:}$.

Here, by $[\boldsymbol{W}]_{k:}$ we mean the matrix created from columns $k, k+1, \dots N$ of $\boldsymbol{W}$. These two invariants are satisfied at the first step ($k=1$).

Using these invariants, we first show that as long as column $\boldsymbol{w}_k$ has less than J n.z. elements, it is possible to choose m and n as in Lines 3 and 4 of Algorithm 3.1. After this, we show that if the invariants are true at step k, they are true at step $k+1$; hence, after K steps, we obtain a mask $\boldsymbol{W}$ satisfying Lemma 3.4.

Consider a sub-step of step k, when $\boldsymbol{w}_k$ has $j < J$ n.z. elements. Then, by Invariant 2 there are at most $(K-k+1)$ n.z. elements in each row of $[\boldsymbol{W}]_{k:}$. Therefore, in rows of $[\boldsymbol{W}]_{k+1:}$ in which column $\boldsymbol{w}_k$ is non-zero there are at most $(K-k)$ n.z. elements. This gives a total of at most $j(K-k)$ n.z. elements in all j such rows of $[\boldsymbol{W}]_{k+1:}$.

By Invariant 1, there are $J(k-1)+j$ n.z. elements in the first $k+1$ columns of $\boldsymbol{W}$, thus there are $J(K-k+1)-j$ n.z. elements in $[\boldsymbol{W}]_{k+1:}$.

Since $j(K-k) < J(K-k+1)-j$ for $j < J$ and $k \geq K$, there is at least one index m such that $w_{m,k} = 0$ and the m-th row has at least one n.z. element in $[\boldsymbol{W}]_{k+1:}$ ($w_{m,n} = 1$ for some $n > k$). We then apply Lines 8 and 9 of Algorithm 3.1, the number of n.z. elements in $\boldsymbol{w}_k$ increases to $j+1$ and the invariants remain true.

To prove the invariants for $k+1$, observe that when we add J-th element to the column k, we create a kth column with J n.z. elements and Invariant 1 is satisfied for $k+1$. Since at step k there were at most $J(K-k+1)$ n.z. elements in $\boldsymbol{W}_{k+1,:}$, there could be at most J rows with exactly $K-k$ n.z. elements in $\boldsymbol{W}_{k+1:}$. In the inner loop we always choose the row with the highest number of n.z. elements and move one of its n.z. elements from $\boldsymbol{W}_{k+1:}$ to the column $\boldsymbol{w}_k$. We do it exactly J times, thus Invariant 2 is satisfied for $k+1$. Therefore both invariants are satisfied at the beginning of step $k+1$.

After step $k = K$, due to Invariant 1, we have K columns with J n.z. elements. This means that the resulting mask satisfies the assumptions of Lemma 3.4. Observe that each step of the algorithm is consistent with Lemma 3.5 and counting backwards, at each step the measurements continue to span $\mathbb{R}^{J \times K}$ everywhere except a measure zero set. Since a finite sum of measure zero sets is measure zero, we obtain that the original measurements span $\mathbb{R}^{J \times K}$ for almost all $(\boldsymbol{z}_n)_{n=1}^{N}$. ■

Now we prove Lemmas 3.4 and 3.5, to conclude proof of Theorem 3.1.

Lemma 3.3: Minimal Setup

Let $\boldsymbol{G}$ and $\boldsymbol{F}$ be full spark. If the mask $\boldsymbol{W}$ contains K columns with J non-zero elements, then matrix $\boldsymbol{C}$ can be recovered from measurements Equation (3.1). Analogous statement is true with rows and columns swapped.

The proof of Lemma 3.4 is based on the observation that if the n-th column of $\boldsymbol{W}$ contains J n.z. elements, then the corresponding measurements span $\mathbb{R}^J \otimes \boldsymbol{f}_n$, no matter where in the column (or row) these n.z. elements are located. This observation is also useful later in the proof of Lemma 3.8.

Proof of Lemma 3.4

Let $\mathcal{S}$ be the subspace spanned by the measurement matrices:

$$\mathcal{S} = \operatorname{span}\left(\{\boldsymbol{g}_m \boldsymbol{f}_n^\top : (m,n) \in \mathcal{M}\}\right) \subseteq \mathbb{R}^J \otimes \mathbb{R}^K,$$

and let n be a column of $\boldsymbol{W}$ with J n.z. elements in rows m_j, i.e., $w_{m_j,n} = 1$ for every $j = 1, \ldots, J$. Then,

$$\operatorname{span}\left(\boldsymbol{g}_{m_1}\boldsymbol{f}_n^\top, \ldots, \boldsymbol{g}_{m_J}\boldsymbol{f}_n^\top\right) = \operatorname{span}\left(\boldsymbol{g}_{m_1}, \ldots \boldsymbol{g}_{m_{J-1}}\right) \otimes \operatorname{span}\left(\boldsymbol{f}_n\right) \subseteq \mathcal{S}.$$

Because $\boldsymbol{G}$ is full-spark, every J vectors $\boldsymbol{g}_n$ span $\mathbb{R}^J$ and

$$\mathbb{R}^J \otimes \operatorname{span}\left(\boldsymbol{f}_n\right) \subseteq \mathcal{S}. \tag{3.20}$$

Since there are K different columns $(n_1, \ldots, n_K)$ with J n.z. elements each, we can write K equations equivalent to Equation (3.20) for $n_1, \ldots, n_K$, and we obtain

$$\mathbb{R}^J \otimes \operatorname{span}\left(\boldsymbol{f}_{n_1}, \ldots, \boldsymbol{f}_{n_K}\right) \subseteq \mathcal{S}.$$

Since $\boldsymbol{F}$ is also full-spark, we have $\mathbb{R}^J \otimes \mathbb{R}^K \subseteq \mathcal{S}$, thus $\mathbb{R}^J \otimes \mathbb{R}^K = \mathcal{S}$ and matrix $\boldsymbol{C}$ can be recovered from measurements Equation (3.1). ■

Lemma 3.3: Replacing Vectors from Polynomial Model

Consider a set of measurements $\mathcal{M}$ as in Equation (3.3), $|\mathcal{M}| = JK$ and let $\boldsymbol{F}$ follow the polynomial model, $\boldsymbol{f}_n = \boldsymbol{p}(\boldsymbol{z}_n)$. Create a reduced set of indices , $\mathcal{M}_r$, by replacing all index pairs $(\cdot, N)$ with $(\cdot, 1)$, in order for the second indices n in $\mathcal{M}_r$ to be in $[N-1]$.

If measurements indexed by $\mathcal{M}_r$ span the whole space $\mathbb{R}^J \otimes \mathbb{R}^K$, then the measurements indexed by $\mathcal{M}$ span $\mathbb{R}^J \otimes \mathbb{R}^K$ for almost all $(\boldsymbol{z}_n)_{n=1}^N$.

Proof of Lemma 3.5

Let $\mathbf{\Gamma}$ be the measurement matrix, as defined in Equation (3.5), and let $\mathbf{\Gamma}_r$ be the measurement matrix of the reduced set $\mathcal{M}_r$. Matrix $\mathbf{\Gamma}$ is a square matrix, and if $\mathcal{M}_r$ spans $\mathbb{R}^J \otimes \mathbb{R}^K$ then $|\mathcal{M}_r| = JK$ and $\mathbf{\Gamma}_r$ is square too; therefore, their determinants are well defined.

We consider the determinants of $\mathbf{\Gamma}$ and $\mathbf{\Gamma}_r$ as functions of the variable $\boldsymbol{f}_N$ we replace. Without loss of generality, we can sort indices in $\mathcal{M}$ according to the second index in a pair. Let i be the first index at which $n_i = N$. Recall that according to the polynomial model $\boldsymbol{f}_n = \boldsymbol{p}(\boldsymbol{z}_n)$, $\boldsymbol{z}_n \in \mathbb{R}^D$ (Definition 3.2). We consider a matrix valued function $\widetilde{\mathbf{\Gamma}} : \mathbb{R}^D \to \mathbb{R}^{JK \times JK}$, where entries with index N in $\mathbf{\Gamma}$ are replaced with $p(\boldsymbol{x})$:

$$\widetilde{\mathbf{\Gamma}}(\boldsymbol{x}) = \left[\begin{array}{c} \operatorname{vec}(\boldsymbol{g}_{m_1}\boldsymbol{f}_{n_1}^\top)^\top \\ \vdots \\ \operatorname{vec}(\boldsymbol{g}_{m_{i-1}}\boldsymbol{f}_{n_{i-1}}^\top)^\top \\ \hline \operatorname{vec}(\boldsymbol{g}_{m_i}\boldsymbol{p}(\boldsymbol{x})^\top)^\top \\ \vdots \\ \operatorname{vec}(\boldsymbol{g}_{m_{JK-1}}\boldsymbol{p}(\boldsymbol{x})^\top)^\top \end{array}\right]. \tag{3.21}$$

The determinant $\det\left(\widetilde{\mathbf{\Gamma}}(\boldsymbol{x})\right)$ is a polynomial in $\boldsymbol{p}(\boldsymbol{x})$, hence it is a polynomial in $\boldsymbol{x}$. Therefore, it is either zero on a measure zero set or zero everywhere. Observe that $\widetilde{\mathbf{\Gamma}}(\boldsymbol{z}_N) = \mathbf{\Gamma}$ and $\widetilde{\mathbf{\Gamma}}(\boldsymbol{z}_1) = \mathbf{\Gamma}_r$. Since the measurement set $\mathcal{M}_r$ spans the whole $\mathbb{R}^{J\times K}$, matrix $\mathbf{\Gamma}_r = \widetilde{\mathbf{\Gamma}}(\boldsymbol{z}_1)$ is full rank, therefore $\det\left(\widetilde{\mathbf{\Gamma}}(\boldsymbol{z}_1)\right) \neq 0$, thus the $\det\left(\widetilde{\mathbf{\Gamma}}\right) \neq 0$ for almost all $\boldsymbol{z}_N$. ■

Finally, we introduce the proof of Theorem 3.3 that is based on oversampling. To prove Theorem 3.3, we make use of Lemmas 3.7 and 3.8 stated below. We present the proofs of these lemmas at the end of this section. The proof Theorem 3.3 is similar to the proof of Theorem 3.1 in that Algorithm 3.2 provides a method for reducing the set of measurements to a set satisfying Lemma 3.4. Then, by Lemma 3.5, matrix $\boldsymbol{C}$ can be uniquely recovered up to a measure zero set. The main difference is that, in Algorithm 3.2 the columns to join are given by Lemma 3.7 instead of being generated one by one as in Algorithm 3.1.

Lemma 3.7: Choosing Columns

Assume that the set of measurements $\mathcal{M}_l$, $|\mathcal{M}_l| = 2Jl$, is such that the columns of corresponding mask $\boldsymbol{W}^{(l)}$ have at most J n.z. elements each, and are sorted in a non-increasing order by their number of elements. Furthermore, in each row of $\boldsymbol{W}^{(l)}$, there are at most l n.z. elements. Then, there exists a set $\mathcal{C}$ of columns such that

1. the logical disjunction (the logical or) of columns in $\mathcal{C}$ has J n.z. elements;
2. the total number of elements in the columns of $\mathcal{C}$ does not exceed $2J$;
3. the first column of $\boldsymbol{W}^{(l)}$ is a column in $\mathcal{C}$.

Lemma 3.8: Disjoint Sub-masks

Assume that matrices $\boldsymbol{G}$ and $\boldsymbol{F}$ are full spark, and consider a mask $\boldsymbol{W}$ that consists of two (disjoint) sub-masks, one of them contains at least l, $l \in \{0, \dots, K\}$ columns with at least J n.z. elements each. Then, the change of position of the n.z. elements within any row of the other sub-mask, which contains at least $K - l$ n.z. elements, does not affect the span of the measurement set corresponding to $\boldsymbol{W}$.

With these two lemmas, we are ready to prove Theorem 3.3.

Theorem 3.3: Oversampling

Assume that $\boldsymbol{G}$ is full spark and $\boldsymbol{F}$ follows the polynomial model. If in each row of $\boldsymbol{W}$ there are at most K n.z. elements and, in each column of $\boldsymbol{W}$, there are at most J n.z. elements, then, for almost all $(z_n)_{n=1}^{N}$,

$$2JK \text{ measurements} \tag{3.22}$$

are sufficient for the unique recovery of matrix $\boldsymbol{C}$.

Algorithm 3.2: Matrix-Reduction Algorithm for Theorem 3.3

Input: Mask $\boldsymbol{W} \in \mathbb{R}^{M \times N}$, satisfying the assumptions of Theorem 3.3
Output: Modified $\boldsymbol{W}$ satisfying the assumptions of Lemma 3.4

```
$k := 1$
while $k \leq K$ and there are $< J$ rows with $K - k + 1$ n.z. elements each in $[\boldsymbol{W}]_{k:}$ do
    sort columns $\boldsymbol{W}$ in non-increasing order by the number of n.z. elements
    while there is a row $m$ with exactly $K - k + 1$ n.z. elements and $w_{m,k} = 0$ do
        let $l$ be such that $w_{m,l} = 1$
        $w_{m,k} := 1$
        $w_{m,l} := 0$
    end while
    if $\boldsymbol{w}_k$ has $< J$ n.z. elements then
        $(k, n_1, \dots, n_I) :=$ sequence of column indices given by Lemma 3.7
        $\boldsymbol{w}_k := \boldsymbol{w}_k \vee \boldsymbol{w}_{n_1} \vee \cdots \vee \boldsymbol{w}_{n_I}$
        for $i = 1, \dots, I$ do
            $\boldsymbol{w}_{n_i} := \boldsymbol{0}$
        end for
    end if
    $k := k + 1$
end while
```

Proof of Theorem 3.3

First, observe that reordering the columns of $\boldsymbol{W}$ corresponds to a change of variables, and this does not change the dimension of the span of the corresponding measurement set.

We consider the following invariants:

1) at step k, each of the first $k-1$ columns of $\boldsymbol{W}$ contain *at least* J n.z. elements;
2) at step k, there are at most $K-k+1$ n.z. elements in each row of $[\boldsymbol{W}]_{k:}$;
3) at step k, there are at least $2(K-k+1)J$ n.z. elements in $[\boldsymbol{W}]_{k:}$.

Again, these three invariants are clearly satisfied for $k=1$. Since we use Lemma 3.7 for choosing which columns to join, we only need to prove the induction step.

After Line 8, there are at most $K-k$ n.z. elements in each row of the sub-mask $\boldsymbol{W}_{k+1:}$, therefore Invariant 2 is satisfied. If $\boldsymbol{w}_k$ has at least J n.z. elements, then Invariant 1 is also satisfied. Since we moved at most $J-1$ n.z. elements, $\boldsymbol{w}_k$ has at most $2J-1$ elements, and the last invariant is also satisfied.

When $\boldsymbol{w}_k$ has less than J elements, we use Lemma 3.7 with $l=K-k+1$ and mask $\boldsymbol{W}^{(l)}=[\boldsymbol{W}]_{k:}$. The invariants for k, together with sorting of $\boldsymbol{W}$, guarantee that the assumptions of the lemma are satisfied. Therefore, Line 10 returns a set of columns that, after joining in the Line 11, produces a column $\boldsymbol{w}_k$ with at least J n.z. elements. Thus, Invariant 1 is satisfied for $k+1$. Moreover, as the columns returned in the lemma had in total at most $2J$ n.z. elements, there are at least $2(K-k+1)J-2J=2(K-k)J$ n.z. elements in $[\boldsymbol{W}]_{k:}$, and Invariant 3 is satisfied.

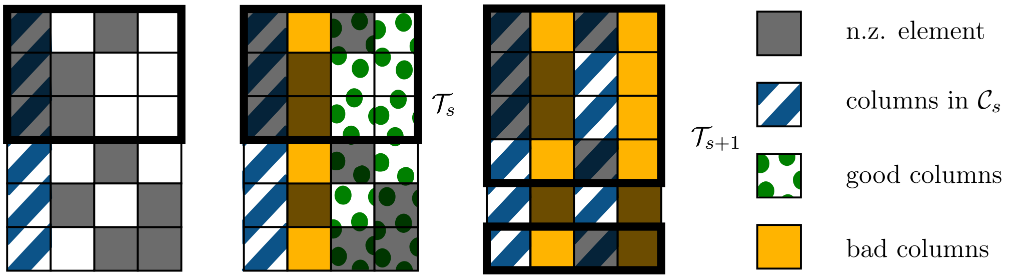
The algorithm terminates at k_t, when the resulting mask contains a sub-mask with k_t-1 columns with J n.z. elements each and a disjoint sub-mask with J rows with $K-k_t+1$ n.z. elements each. From Lemma 3.8, we know that we can move elements in the second sub-mask, freely and without affecting the span of the measurements. Hence, we move them to the first $K-k_t+1$ columns of the sub-mask, thus obtaining in total K columns with J measurements and a mask that satisfies Lemma 3.4. As in the proof of Theorem 3.1, the steps of Algorithm 3.2 do not change the dimension of the span of the measurements. The lines from Line 10 until the end are consistent with Lemma 3.5, and the inner loop (Line 4) is consistent with Lemma 3.8. Therefore, the original measurements span $\mathbb{R}^{J\times K}$ for almost all $(\boldsymbol{z}_n)_{n=1}^{N}$. ■

To conclude the proof of Theorem 3.3, we need only to prove Lemmas 3.7 and 3.8.

Lemma 3.3: Choosing Columns

Assume that the set of measurements $\mathcal{M}_l$, $|\mathcal{M}_l|=2Jl$, is such that the columns of corresponding mask $\boldsymbol{W}^{(l)}$ have at most J n.z. elements each, and are sorted in a non-increasing order by their number of elements. Furthermore, in each row of $\boldsymbol{W}^{(l)}$, there are at most l n.z. elements. Then, there exists a set $\mathcal{C}$ of columns such that

1. the logical disjunction (the logical or) of columns in $\mathcal{C}$ has J n.z. elements;
2. the total number of elements in the columns of $\mathcal{C}$ does not exceed $2J$;
3. the first column of $\boldsymbol{W}^{(l)}$ is a column in $\mathcal{C}$.



(a) (b) (c)

Figure 3.7.: Process described in Lemma 3.7. We consider a mask $\boldsymbol{W}$ with at most $J = 4$ n.z. elements in each column and at most $k = 2$ n.z. elements in each row. We start with only the first column in the set $\mathcal{C}_1$ marked with blue stripes (a). Because this column has n.z. elements in the first three rows, the first three columns are already covered (black outline) and $\mathcal{T}_1 = \{1, 2, 3\}$. Next, we look to add columns to $\mathcal{C}_1$ (b). The second column is *bad* (solid yellow), because it has too many n.z. elements in the covered rows. We choose one of the remaining *good* (green dots) columns and add it to our set $\mathcal{C}_1$ (c). All the remaining rows are then bad, but now $\mathcal{T}_2 = \{1, 2, 3, 4, 6\}$ has more than four n.z. elements, and we have created a correct group.

Proof of Lemma 3.7

We create the desired set of column indices $\mathcal{C}$ by starting with a singleton $\mathcal{C}_1$ of a column with the maximal number of n.z. elements. We then append "good" columns in a sequence, until we obtain J n.z. elements: see Figure 3.7.

More formally, for the n-th column, we define the set $\mathcal{R}_n \subseteq [M]$ of rows where the column is non-zero:

$$\mathcal{R}_n = \{r : w_{r,n} = 1\}.$$

Let $\mathcal{C}_s$ be the set of columns created after step s, and let $\mathcal{T}_s$ be the set of rows where at least one of the columns in $\mathcal{C}_s$ is non-zero:

$$\mathcal{T}_s = \bigcup_{i=\mathcal{C}_s} \mathcal{R}_i.$$

We then call the n-th column of $\boldsymbol{W}$ *good* — green in Figure 3.7 — if it has, in the new rows, at least as many n.z. elements as it has in the already covered rows:

$$|\mathcal{T}_s \cap \mathcal{R}_n| \leq |\mathcal{R}_n \setminus \mathcal{T}_s| \,. \tag{3.23}$$

If we are to add only good columns to the set $\mathcal{C}_s$, we need to check that if $|\mathcal{T}_s| < J$, then there exists at least one good column. Let $\mathcal{B}_s$ be the set of indices of *bad* columns that do not satisfy Equation (3.23). Since in each row there are at most l n.z. elements, there can be at most $l|\mathcal{T}_s|$ n.z. elements in rows $\mathcal{T}_s$, hence there can be at most $(l-1)|\mathcal{T}_s|$ n.z. elements in rows $\mathcal{T}_s$ of the "bad" columns:

$$\bigcup_{b \in \mathcal{B}_s} |\mathcal{R}_b \cap \mathcal{T}_s| \leq (l-1)|\mathcal{T}_s|.$$

Since for each $b \in \mathcal{B}_s$ we have $|\mathcal{R}_b \setminus \mathcal{T}_s| < |\mathcal{T}_s \cap \mathcal{R}_b|$, then also $|\mathcal{R}_b| < 2\,|\mathcal{T}_s \cap \mathcal{R}_b|$, and therefore

$$\bigcup_{b \in \mathcal{B}_s} |\mathcal{R}_b| \leq 2(k-1)|\mathcal{T}_s|. \tag{3.24}$$

The number of available measurements is the difference between the total number of measurements "covered" so far and the upper bound on measurements lost so far:

$$\underbrace{2Jl}_{\text{total}} - \underbrace{|\mathcal{T}_s|}_{\text{covered}} - \underbrace{(|\mathcal{T}_s| - |\mathcal{R}_1|)}_{\geq \text{lost}}. \tag{3.25}$$

Here, the number of measurements lost is equal to $|\mathcal{T}_s| - |\mathcal{R}_1|$ because at each step, except the first one, we lose (i.e., duplicate) at most as many measurements as we add to the set $\mathcal{T}_s$.

If $|\mathcal{T}_s| < J$, then the number of available measurements Equation (3.25) is larger than the number of bad measurements Equation (3.24), and there is at at least one measurement in a good row, and we can add it to $\mathcal{C}_s$ and obtain $\mathcal{C}_{s+1}$. There are two options: Either $|\mathcal{T}_{s+1}| < J$ and we can continue adding columns, or $|\mathcal{T}_{s+1}| \geq J$. Then, the total number of measurements used is at most the number of measurements used for s plus the number of measurements in the last added column:

$$\underbrace{|\mathcal{T}_s| + (|\mathcal{T}_s| - |\mathcal{R}_1|)}_{\text{old}} + \underbrace{|\mathcal{R}_{s+1}|}_{\text{new}} = 2|\mathcal{T}_s| - |\mathcal{R}_1| + |\mathcal{R}_{s+1}| <\leq 2J,$$

because $|\mathcal{T}_s| \leq J$ and because we started with the column with maximal number of elements, hence $|\mathcal{R}_1| \geq |\mathcal{R}_{s+1}|$. Therefore, we create the set of columns $\mathcal{C} = \mathcal{C}_{s+1}$ which satisfies both conditions of the lemma. ■

Lemma 3.3: Disjoint Sub-masks

Assume that matrices $\boldsymbol{G}$ and $\boldsymbol{F}$ are full spark, and consider a mask $\boldsymbol{W}$ that consists of two (disjoint) sub-masks, one of them contains at least l, $l \in \{0, \dots, K\}$ columns with at least J n.z. elements each. Then, the change of position of the n.z. elements within any row of the other sub-mask, which contains at least $K - l$ n.z. elements, does not affect the span of the measurement set corresponding to $\boldsymbol{W}$.

Proof of Lemma 3.8

Once again, let $\mathcal{S}$ be the subspace spanned by the measurement matrices:

$$\mathcal{S} = \operatorname{span}\left(\{\boldsymbol{g}_m \boldsymbol{f}_n^\top : (m,n) \in \mathcal{M}\}\right) \subseteq \mathbb{R}^J \otimes \mathbb{R}^K.$$

Like Equation (3.20) in the proof of Lemma 3.4, $\boldsymbol{G}$ being full spark means that

$$\mathbb{R}^J \otimes \operatorname{span}(\boldsymbol{f}_n) \subseteq \mathcal{S},$$

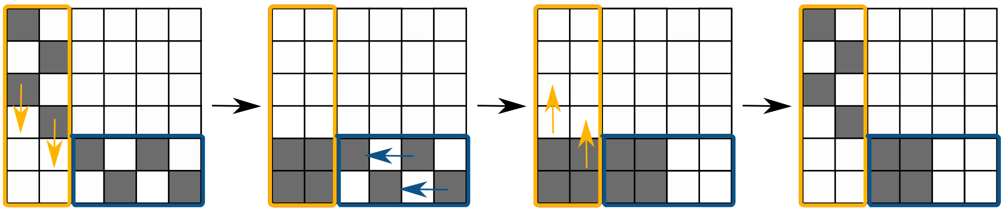


Figure 3.8.: Sometimes n.z. elements in a row can be moved around, even if they do not span the whole $\mathrm{span}(\boldsymbol{g}_m) \otimes \mathbb{R}^K$, where m is the row index. Here $K = 4$ and $J = 2$, and the first sub-mask of $\boldsymbol{W}$ (yellow outline) contains the first two columns, which have J n.z. elements, and the second sub-mask contains the remaining part of the last two rows (blue outline). Therefore, the first two columns span $\mathbb{R}^J \otimes \mathrm{span}(\boldsymbol{f}_1, \boldsymbol{f}_2)$, and their n.z. elements can be moved down (yellow arrows). This creates two rows that have $K = 4$ n.z. elements, hence n.z. elements in these rows can be moved to the left (blue arrows). Their n.z. elements can be moved back (yellow arrows) because, in this step, we have not moved n.z. elements in the first two columns. The resulting matrix differs from the original only in the second (blue) sub-mask.

for every column n of $\boldsymbol{W}$ that contains J elements. Without loss of generality, let the first row of $\boldsymbol{W}$ be the one containing l elements in the second sub-mask. We then also have

$$\mathrm{span}(\boldsymbol{g}_1) \otimes \mathrm{span}(\boldsymbol{f}_n) \subseteq \mathcal{S}.$$

If $n_1, \ldots, n_l$ are the columns with J elements, let $n_{l+1}, \ldots n_K$ be the columns in which the first row in the second sub-mask is n.z. element, see Figure 3.8. Since the masks are disjoint, $n_i \neq n_j$ for $j \neq i$, and

$$\mathrm{span}(\boldsymbol{g}_1) \otimes \mathrm{span}\left(\boldsymbol{f}_{n_1}, \ldots \boldsymbol{f}_{n_l}, \boldsymbol{f}_{n_{l+1}}, \ldots \boldsymbol{f}_{n_K}\right) \subseteq \mathcal{S}.$$

Therefore, since $\boldsymbol{F}$ is full spark, $\mathrm{span}(\boldsymbol{g}_1) \otimes \mathbb{R}^K \subseteq \mathcal{S}$, and it does not matter which values have the indices $n_{l+1}, \ldots, n_K$, as long as they are different than $n_1, \ldots, n_l$. ■

Finally, we prove the generalisation of Theorem 3.1 for order-D tensors. The proof we present below inductively relies on Theorem 3.1.

Corollary 3.3: Generalisation of Theorem 3.1

Assume that we measure a rank-D tensor $\boldsymbol{C}$ using D full spark frames:

$$\left(\boldsymbol{g}_n^{(1)}\right)_{n=1}^{N_1}, \ldots, \left(\boldsymbol{g}_n^{(D)}\right)_{n=1}^{N_D}$$

spanning, respectively, $\mathbb{R}^{K_1}$ to $\mathbb{R}^{K_D}$. Furthermore, assume that the first frame $\left(\boldsymbol{g}_n^{(1)}\right)_{n=1}^{N_1}$ follows the polynomial model. Then, the set of $\prod_{d=1}^{D} K_d$ measurements such that there are at most $\prod_{j=1}^{d-1} K_j$ repeated vectors of the frame number d is sufficient to uniquely recover tensor $\boldsymbol{C}$.

Proof

Recall that tensor $\boldsymbol{C}$ can be recovered exactly if and only if vectors defined by $\mathcal{M}_D$ span $\mathbb{R}^{K_1\times\cdots\times K_D}$. Assume that Corollary 3.6 is true for $D-1$. Then, any $\prod_{d=1}^{D-1} K_d$ vectors

$$\boldsymbol{f}_{\boldsymbol{n}} = \operatorname{vec}\left(\boldsymbol{g}_{n_1}^{(1)} \otimes \boldsymbol{g}_{n_2}^{(2)} \otimes \cdots \otimes \boldsymbol{g}_{n_{D-1}}^{(D-1)}\right).$$

are linearly independent, because frames from 1 to $D-1$ satisfy assumptions of Corollary 3.6.

Since each index n_1 of the first frame is used at most once, we can replace multi-index $\boldsymbol{n}$ of $\boldsymbol{f}_{\boldsymbol{n}}$ by n_1. By assumption, the vectors $\boldsymbol{g}_n^{(1)}$ in the first frame follow the polynomial model, hence the re-indexed vectors $\boldsymbol{f}_{n_1}$ do too.

Therefore, $\boldsymbol{f}_{n_1}$ and $\boldsymbol{g}_{n_D}$ satisfy the assumptions of Theorem 3.1, and any $\prod_{d=1}^{D} K_d$ of them are linearly independent, as long as vectors $\boldsymbol{g}_{n_D}^{D}$ are repeated at most $\prod_{d=1}^{D-1} K_d$ times.

To conclude the proof, we observe that Theorem 3.1 is exactly Corollary 3.6 for $D = 2$. ■

Want to know more? 3.11: Permutations, partitions, equivalence classes

Definition 3.5: Permutation

A permutation $\boldsymbol{\sigma}$ of numbers $1, \dots, N$ is any sequence of length N in which each number from 1 to N appears exactly once. We will call $\mathcal{P}_N$ the set of all permutations of $[N]$. As with vectors, we denote the n-th element of the permutations $\boldsymbol{\sigma}$ by σ_n. The n-th element of $\boldsymbol{\sigma}$ can be also denoted by $\boldsymbol{\sigma}(n)$. We denote the space of permutations by $\mathcal{P}_N$.

An *inversion* in a permutation is a pair $k < n$ such that $\sigma_k > \sigma_n$. The number of inversions of a permutation is denoted by $I(\boldsymbol{\sigma})$ and is used to define the *sign* $\operatorname{sgn}(\boldsymbol{\sigma})$ of the permutation. A sign — or a parity — of a permutation $\boldsymbol{\sigma}$ is defined as a parity of the number of inversions of $\boldsymbol{\sigma}$:

$$\operatorname{sgn}(\boldsymbol{\sigma}) = (-1)^{I(\boldsymbol{\sigma})}.$$

Alternatively, the sign can be also defined using transpositions, which we do not introduce here.

Permutations can be composed as functions. We will denote composition as multiplication. In our notation, the following are equivalent:

$$(\boldsymbol{\sigma}\boldsymbol{\pi})_k = \boldsymbol{\sigma}(\boldsymbol{\pi}(k)) = \boldsymbol{\sigma}_{\boldsymbol{\pi}_k}.$$

The sign of the permutation has a convenient property that the sign of the composition of two permutations is the product of the signs:

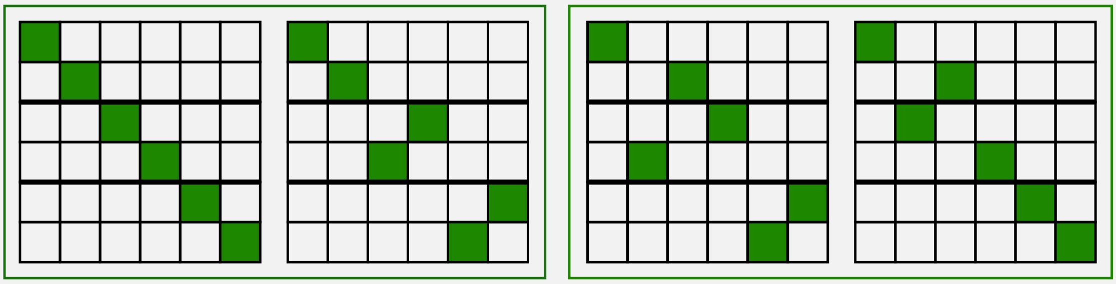
$$\operatorname{sgn}(\boldsymbol{\sigma}\boldsymbol{\pi}) = \operatorname{sgn}(\boldsymbol{\sigma}) \operatorname{sgn}(\boldsymbol{\pi}).$$

This is useful in the proof of Lemma 3.2, because we want to decompose the sign of permutation $\boldsymbol{\sigma} \in \mathcal{P}_{JK}$ into signs of partition $\boldsymbol{\xi}$ and signs of smaller permutations $\boldsymbol{\pi} \in \mathcal{P}_K$ and $\boldsymbol{\tau}^k \in \mathcal{P}_J$ for $k \in [K]$. To do this, we need to redefine the partition in terms of permutations. In the proof of Lemma 3.2, we do this explicitly in Equation (3.17):

$$\boldsymbol{\sigma} \mapsto \boldsymbol{\xi} = \left\{ \{ \sigma_{kJ+j} : j \in [J] \} : k \in [K] \right\}.$$

The mapping above also defines an equivalence relation: We say that two permutations $\boldsymbol{\sigma}$ and $\boldsymbol{\pi}$ are equivalent ($\boldsymbol{\sigma} \sim \boldsymbol{\pi}$) if it maps them to the same $\boldsymbol{\xi}$.

Below, the permutation $\boldsymbol{\sigma}$ is illustrated by shading the square in the row n and column k if $\boldsymbol{\sigma}_n = k$. Two permutations on the left are equivalent to each other; and two permutations on the right are equivalent to each other. However, the left permutations are are not equivalent to the right permutations.



The fact that partitions define equivalence relation on permutations means a partition itself can be thought of as an *equivalence class* of permutations under relation $\sim$. This enables us to use the language of equivalence classes. An equivalence class is often defined by its representative. In our case, the representative is the permutation $\boldsymbol{\sigma}$, which is the smallest in the lexicographical order that maps to $\boldsymbol{\xi}$, or in other words the first element of equivalence class $\boldsymbol{\xi}$.

The concept of representative enables us to extend the definitions of inversion and sign of a permutation to partitions. If $\boldsymbol{\sigma}$ is a representative of $\boldsymbol{\xi}$, then we define:

$$I(\boldsymbol{\xi}) = I(\boldsymbol{\sigma}) \text{ and } \operatorname{sgn}(\boldsymbol{\xi}) = \operatorname{sgn}(\boldsymbol{\sigma}).$$

Together with the property of composition of signs, this gives us a method to calculate the sign of the composition of a partition with a permutation *of the same length* JK. Let $\boldsymbol{\pi}$ and $\boldsymbol{\tau}^k$ for $k \in [K]$ be the permutations that satisfy equation Equation (3.18):

$$\sigma_{J(k-1)+j} = (\xi_{\pi_k})_{\tau_j^k}$$

Applying permutation $\boldsymbol{\tau}^k \in \mathcal{P}_J$ to the k-th set of $\boldsymbol{\xi}$ is equivalent to a composition with $\widetilde{\boldsymbol{\tau}}^k \in \mathcal{P}_{JK}$ created by extending $\boldsymbol{\tau}^k$ with an identity. More formally,

$$\widetilde{\tau}_n = \begin{cases} J(k-1) + \tau_j^k & \text{for } n = J(k-1) + j \\ n & \text{else.} \end{cases}$$

It is easy to see that $\operatorname{sgn}(\widetilde{\boldsymbol{\tau}}) = \operatorname{sgn}(\boldsymbol{\tau})$, because the all the inversions in $\widetilde{\boldsymbol{\tau}}$ happen on the range $J(k-1)+1$ to Jk and there are as many of them as inversions in $\boldsymbol{\tau}$.

We would like to apply the same trick to the permutation $\boldsymbol{\pi} \in \mathcal{P}_k$, but $\boldsymbol{\pi}$ is applied to all sets of $\boldsymbol{\xi}$. We can think about it as applying it to the smallest elements in their respective sets, second in order, and so on:

$$\widetilde{\pi}_n = J\pi_k + (j-1) \text{ for } n = Jk + (j-1)$$

Therefore, $\operatorname{sgn}(\widetilde{\boldsymbol{\pi}}) = \operatorname{sgn}(\boldsymbol{\pi})^J$ and, putting this all together, we obtain

$$\begin{aligned} \operatorname{sgn}(\boldsymbol{\sigma}) &= \operatorname{sgn}(\boldsymbol{\xi}\widetilde{\boldsymbol{\pi}}\widetilde{\boldsymbol{\tau}}^1 \cdots \widetilde{\boldsymbol{\tau}}^K) \\ &= \operatorname{sgn}(\boldsymbol{\xi}) \operatorname{sgn}(\widetilde{\boldsymbol{\pi}}) \operatorname{sgn}(\widetilde{\boldsymbol{\tau}}^1) \cdots \operatorname{sgn}(\widetilde{\boldsymbol{\tau}}^K) \\ &= \operatorname{sgn}(\boldsymbol{\xi}) \operatorname{sgn}(\boldsymbol{\pi})^J \operatorname{sgn}(\boldsymbol{\tau}^1) \cdots \operatorname{sgn}(\boldsymbol{\tau}^K) \end{aligned}$$

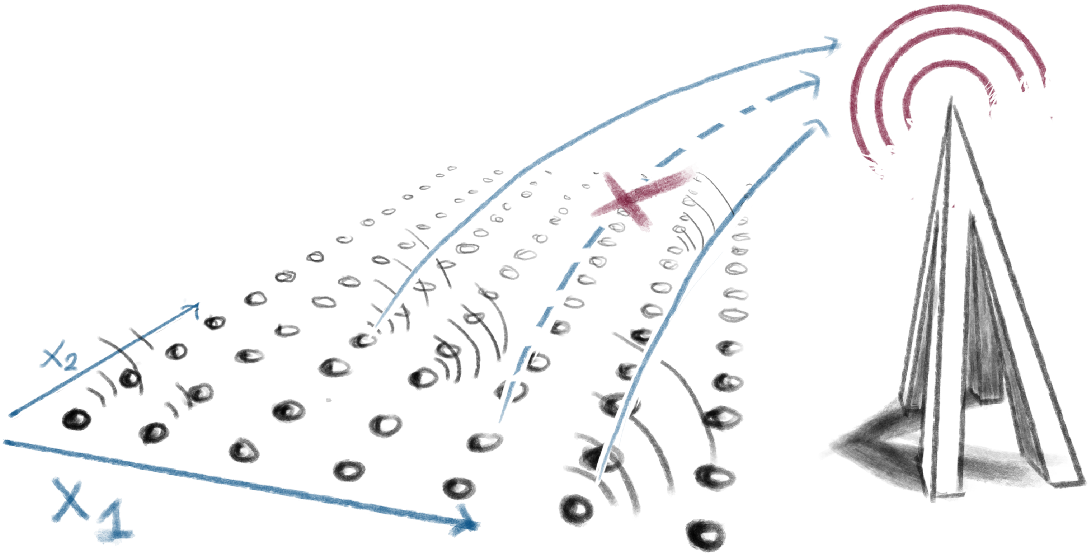


Figure 3.9.: Measurements with noisy communication channel. We assume that measurements are on a two dimensional grid. This could be one dimension in time and one in space, or two dimensions in space, as illustrated here. Measurements are synchronised, but some are lost due to noisy communication channel. The resulting measurements are no longer uniform.

3.2. Applications

In this section, we present a few toy-applications for rank one sensing: sampling with missing samples in Section 3.2.1, neuromorphic sensing in Section 3.2.2, and signals with hidden variables in Section 3.2.3. We illustrate how to apply our results in practice and point out general application areas, rather than establish state-of-the-art results. In Chapter 4, we present our main application, continuous localisation, and we describe the performance of our method on real data.

3.2.1. Erasure Channel

Consider a number of sensors that measure some signal that changes over time, for example temperature. The sensor nodes then send measurements to one server. Assume that the communication between the sensors and the server is unstable and some measurements are lost, see Figure 3.9. We assume that we know which measurements are lost and the situation can be modelled as an erasure channel. To counter the problem with the communication, we oversample the signal.

Such a scenario can occur when sensory nodes are inexpensive and have low computational power or memory, hence it is more feasible to deploy more of them than to implement more robust communication protocols. Another use-case is when the sensors move and there is a possibility that they leave the range of the network for a time. The cheap sensor model is interesting because matrix multiplication by a rank-one matrix, given in the form of two vectors, is less computationally expensive than matrix multiplication in the general case.

We assume here that the devices are synchronised. For example, temperature measurements taken every day can be assumed to be synchronised, even if they are not synchronised to the exact minute. The situation when the measurements are not synchronised is better described in Section 3.2.2.

We model our signal $y(x,t)$ to be either polynomial or bandlimited in space and time. Signal y can be the written as a sum of monomials

$$y(x,t) = \sum_{j=0}^{J-1} \sum_{k=0}^{J-1} c_{j,k} x^j t^k,$$

in the polynomial case and similarly in the bandlimited case

$$y(x,t) = \sum_{j=-\widehat{J}}^{\widehat{J}} \sum_{k=-\widehat{K}}^{\widehat{K}} c_{j,k} e^{ijx} e^{ikt},$$

where we use $\widehat{J} = (J-1)/2$ and $\widehat{K} = (K-1)/2$. Thus, J and K describe the number of degrees of freedom. Both these families can be written in the form of Equation (3.2):

$$y(x,t) = \boldsymbol{g}^\top(x) \boldsymbol{C} \boldsymbol{g}(t),$$

where $\boldsymbol{g}$ is either polynomial or bandlimited basis. Assuming uniform sampling both in time and space, we obtain

$$y_{m,n} = \boldsymbol{g}_m^\top \boldsymbol{C} \boldsymbol{f}_n.$$

for $\boldsymbol{g}_m = \boldsymbol{g}(\delta m)$ and $\boldsymbol{f}_n = \boldsymbol{g}(\tau n)$ where δ is the distance between samples and τ is the time between samples.

In order to see if we can apply Theorem 3.1 or 3.3 in this scenario, we check if their assumptions are satisfied. In case of polynomials or bandlimited functions, frames $\boldsymbol{g}_m$ and $\boldsymbol{f}_n$, respectively, are full spark and follow the polynomial model. There are multiple measurements in each column and in each row of the grid, hence we can use only Theorem 3.3. We know that, if we have JK^2 measurements such that no more than J of them are taken at the same time and no machine generates more than K measurements, we can perfectly recover $\widehat{\boldsymbol{C}}$.

Theorem 3.3 does not tell us if our sampling setup is efficient. The imperfect communication between the server and the nodes can be written as a noisy communication channel, in particular an erasure channel. Using this formulation, we can compare how efficient our setup is compared to the channel capacity or to optimal codes. Erasure codes are of particular interest. The idea is, if we have a message of length K, we can encode it with a code of length N_e, such that recovering every K' letters of the code (it can be that $K' = K$) enables us to recover the original message. This would be the case, for example, if we vectorised $\boldsymbol{C}$, obtaining a message length KJ, and then encoded it with a full-spark frame, i.e., multiplied it by a full-spark $N_e \times JK$ matrix.

3.2.2. Event-Based Sampling

As described in Section 1.1: Implicit Sampling, event-based sampling is a sampling paradigm where the sampling times depend on the changes in the signal: the events. The information is encoded in the sampling times, rather than in the sample value.

Here we are interested in simple models of neurons that can be described as event-based sampling. Broadly speaking, there are two neuron models: integrate and fire (IAF) neurons[4] and threshold and fire (TAF) neurons. The IAF model matches the biological neurons more closely than the TAF model. The TAF model can be described as send-on-delta sampling. Here we adopt TAF neurons due to their simplicity. As we show in our joint work with Karen Adam [239, 243], our results apply also to the time-encoding machines that model IAF neurons.

To more precisely describe the TAF model, consider sampling of a function $y : \mathbb{R} \to [-A, A]$ for some known $A \in \mathbb{R}_+$. The TAF neuron will output spikes at times t_n, $n \in [N]$ such that

$$y_n = y(t_n) - y(t_{n-1}) = \pm\delta.$$

It is important that this can be viewed as a non-uniform sampling scheme with $\widehat{y}_n = y(t_n) = \sum_{k \leq n} y_n$. There are variations on this model, but they all can be phrased as non-uniform sampling. For example there are TAF models that include more than the last value and $y_n = y(t_n) - \sum_k y_{n-k} h_k(t_n - t_{n-k})$ or do not include previous sample at all, but they include more than two thresholds. Event-based cameras usually sample the logarithm of the intensity of the image, $\log(y(t))$.

Assume now that we have M sensors, each sampling a signal y_m, and producing samples

$$\widehat{y}_{n,m} = y_m(t_n) + \xi_m \text{ for } (m, n) \in \mathcal{M},$$

where t_n is the sampling time of the n-th sample produced by any machine, ξ_m is the known initialisation of the sensor — value of the signal at zero[5], and $\mathcal{M}$ is the set of available measurements, that is to say $(m, n) \in \mathcal{M}$ if and only if m-th machine produced a sample at the time t_n.

Consider a camera made of event-based sensors, depicted in Figure 3.10. Let us begin with a "1D" camera that consists of a number of sensors on the line, that samples a signal that changes over time. We assume that the signal is a real-valued periodic bandlimited function (see Section 1.5). Again, we can write the measured signal in the form of Equation (3.2):

$$\widehat{y}(x, t) = \boldsymbol{g}^\top(x) \boldsymbol{C} \boldsymbol{g}(t),$$

where $\boldsymbol{g}$ is either a polynomial or (real-valued) bandlimited basis. This time, we can assume only uniform sampling in space and we obtain

$$y_{m,n} = \boldsymbol{g}_m^\top \boldsymbol{C} \boldsymbol{f}_n,$$

for $\boldsymbol{g}_m = \boldsymbol{g}(\delta m)$ and $\boldsymbol{f}_n = \boldsymbol{g}(t_n)$ where δ is the distance between samples.

---

[4] Time encoding machines are a special case of the IAF model.

[5] In practice initialisation of the sensor is not always known, but for illustrative purposes we assume it is.

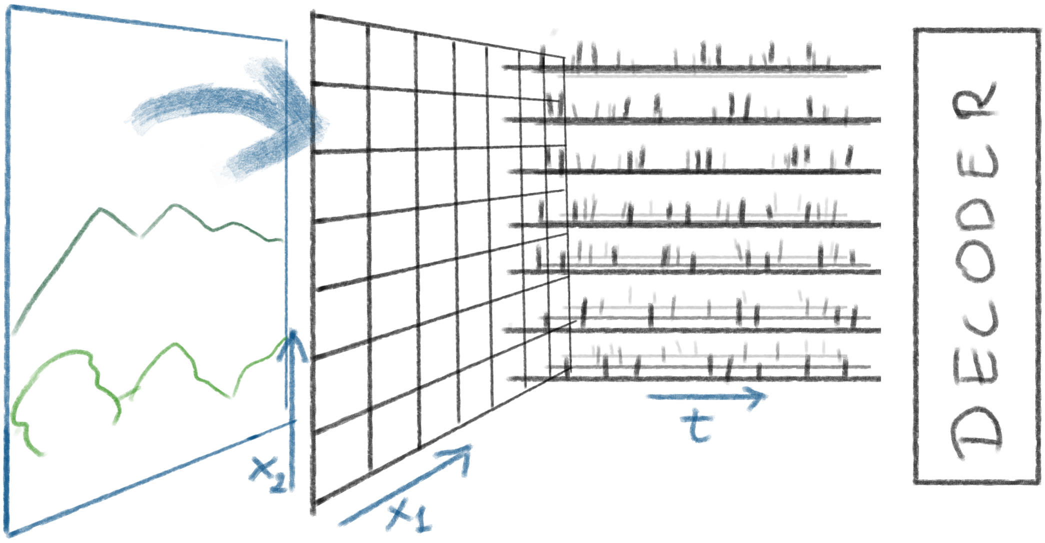


Figure 3.10.: Event-based camera. The image is mapped on a grid of even-based sensors. Each sensors creates a stream of events independently of its neighbours, resulting in non-uniform multi-channel samples, that need to be converted to video format.

Observe that as $\boldsymbol{F}$ is full spark, signals y_m are pair-wise different, therefore we can assume that sampling times t_n are different for different machines, hence they are all different. This means that if

$$\sum_{m=1}^{M} \min(N_m, K) \geq JK,$$

the recovery of matrix $\boldsymbol{C}$ is unique thus is the recovery of $\boldsymbol{x}$ and $\boldsymbol{y}$. Note that if we were to recover every signal y_m separately, we would need at least K samples for each machine; this would amount to $MK \geq JK$ samples.

We can now move to the 2D case, with sensors at positions $(x_m^{(2)}, x_p^{(2)})$. In this case, we can use an extension to Theorem 3.1 in order to show that if we take at most J samples in each column of the sensor grid and at most J^2 samples in each row of the sensor grid, then J^2K measurements in total would be sufficient for unique recovery of $\boldsymbol{C}$ hence of $y(t, x^{(1)}, x^{(2)})$.

3.2.3. Signals With Latent Variables

Thus far, we have assumed that the signals we want to recover are polynomials or bandlimited functions, hence the matrix (or tensor) structure arises from separation of the basis vectors. Here, we consider signals with some latent structure.

We measure signals $\boldsymbol{y} : \mathbb{R} \to \mathbb{R}^M$, but this time they linearly depend on latent signals $\boldsymbol{x} : \mathbb{R} \to \mathbb{R}^J$:

$$\boldsymbol{y}(t) = \boldsymbol{G}\boldsymbol{x}(t).$$

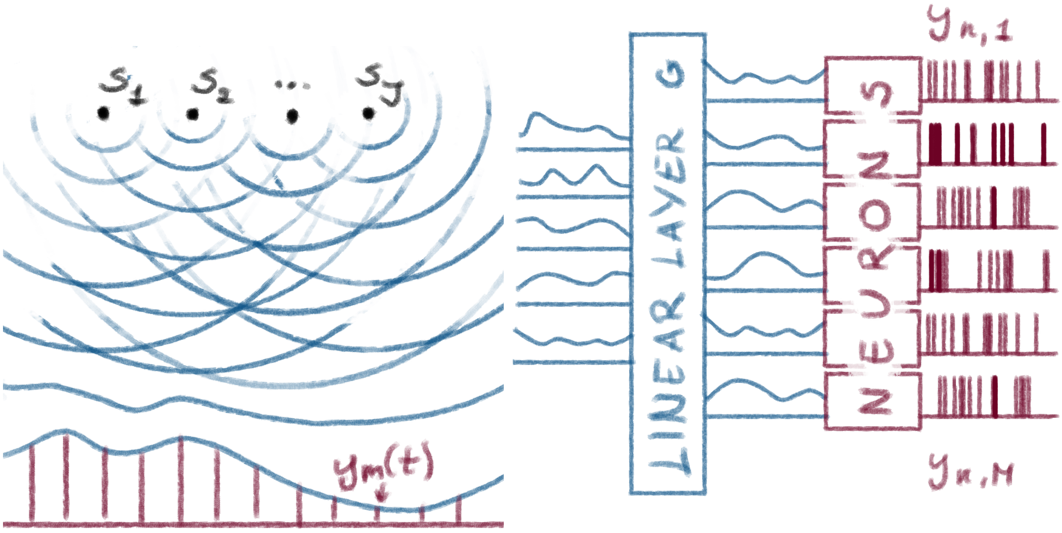


Figure 3.11.: Signals with hidden variables. Left: source separation. We are given source positions and propagation model and we want to recover the source signals over time. Right: one-layer spiking neural network. The neurons, for example TEMs, introduce non-linearity (red) after a fully connected linear layers (blue).

Such a relation appears, for example, in the sound source separation, where we have J far-away sources emitting bandlimited signals that are measured by M microphones; see Figure 3.11 (left). If we know the directions of the sounds and the positions within the microphone array, then we know the matrix $\boldsymbol{G}$ is known. Indeed, if $\boldsymbol{e}_j$ is the normalised vectors in the direction of the source j and $\boldsymbol{u}_m$, $m \in [M]$, are positions of the microphones, then

$$g_{j,m} = e^{-iv\boldsymbol{e}_j^\top \boldsymbol{u}_m}, \tag{3.26}$$

where v is the speed of sound, and we ignore the global phase.

Another situation where hidden variables come into play is in modelling spiking neural networks. On a high level, every network consists of linear layers interlaced with non-linear operations. In spiking neural networks, this non-linearity is an artificial neuron. If $\boldsymbol{x}$ is the input to the linear layer and $\boldsymbol{G}$ are the weights of the layer, then elements of $\boldsymbol{y}$ are the inputs to neurons, and samples $y_m(t_n)$ represent the output of the layer; see Figure 3.11(right).

3.3. Future Work

We have studied the problem of matrix recovery from rank-one measurements. Without any assumptions on the unknown matrix structure and for very general assumptions on the measurement matrices, we provided a condition for unique recovery with the least possible measurements in the case where there is at most one measurement in each column of the mask $\boldsymbol{W}$. We also provide a more general uniqueness condition for multiple measurements, in each column of the mask $\boldsymbol{W}$ and at a price of two-times oversampling.

In Section 3.1.2, we show that Theorem 3.3 cannot be made tight without considering the detailed structure of the mask $\boldsymbol{W}$. However, there could be stronger results available in special cases. In particular, the dual of Theorem 3.1 is interesting, see Table 3.1. We could expect a similarly simple result when the number of n.z. elements in the *rows* of $\boldsymbol{W}$ is limited to one or, in other words, when the vectors following the polynomial model are allowed to repeat while the vectors $\boldsymbol{g}_n$ have to be unique. However, this case would require a more general version of Lemma 3.5.

While not of immediate practicality, studying detailed structure of $\boldsymbol{W}$ might be interesting from theoretical point of view. In particular, as the problem of rank one sensing can be interpreted as matrix completion with additional constraints, results from matrix completion with deterministic masks [191–196]. can be of useful here.

Furthermore, stronger results might be obtained via additional assumptions on matrix $\boldsymbol{C}$, such as the low-rank property. Although our method uses a different methodology than compressed sensing techniques, we believe our results can inform further research into low-rank matrix recovery from rank-one measurements. In particular, Gross [190] studies recovery of matrix given in any basis, and our results give conditions on when a set of measurements of the form Equation (3.1) is a basis.

In the proofs in Section 3.1.3, we use algorithms to describe a reduction of the mask $\boldsymbol{W}$, but we do not describe any algorithms for the recovery of matrix $\boldsymbol{C}$. For simulations (i.e., Figure 3.3), we simply use the (pseudo)inverse, but this method could be compared with various task-specific algorithms [245, 247, 250]. Other algorithms, such as orthogonal rank-one matrix pursuit [259], could be adapted to our case. A comparison of a state-of-the-art completion of matrix $\boldsymbol{Y}$ with two step recovery of matrix $\boldsymbol{C}$ and then $\boldsymbol{Y}$ would be interesting and is a topic of our future work.

Another interesting question is, What is the structure of the measure-zero set where the recovery is not unique? We think that some information could be inferred from Lemma 3.2.

In the simplest case, when $\boldsymbol{f}_n = \boldsymbol{z}_n$ (or $p(\boldsymbol{x}) = \boldsymbol{x}$) and $\boldsymbol{f}$ do not repeat, Equation (3.12) is an affine transformation in any single $z_{d,n}$, and we can always find its zero crossing. However, it might be that this zero crossing gives a vector $\widetilde{\boldsymbol{z}}_n$ that is linearly dependent on some number (smaller than K) of vectors $\boldsymbol{f}_n$, thus contradicting our assumptions.

Finally, in some applications only one set of vectors $\boldsymbol{g}$ or $\boldsymbol{f}$ is known. This can be when sampling a signal with latent variable Section 3.2.3 if the model of the latent signal is known, but the linear mapping of the latent signal to the measurements is not known, or when sampling video with unknown low-dimensional structure [260]. In this case our results do not apply. However, as some of the structure is known, it should be possible to design methods that perform better that out of the box matrix completion. Such methods could be also useful in recovery of structured matrices, such as point-to-plane distance matrices used in shape recovery form echos [261].

Table 3.1.: Overview of the possible results for the problem of matrix sensing from rank one measurements. In this chapter, we present results for almost all latent vector sequences $(\boldsymbol{f}_n)_1^N$, and the number of n.z. elements per row/column is either $K/1$ (Theorem 3.1) or K/J (Theorem 3.3), respectively. The case where there is at most one n.z. element in each row *and* column is simply a special case of Theorem 3.1. We also show examples where our results cannot be generalised to sequences of vectors $(\boldsymbol{f}_n)_1^N$. As we consider only results up to a subset of measure zero of $(\boldsymbol{f}_n)_1^N$, and not $(\boldsymbol{g}_n)_1^N$, there is an asymmetry, and a case of 1 n.z. element per row and up to J n.z. elements per column is not covered by Theorem 3.1. Although Theorem 3.3 still applies in this case, we could expect that there exists a *sufficient and necessary* condition for this case.

How many n.z. elements per row?	How many n.z. elements per column?	
	at most 1	at most J
at most 1	sufficient and necessary conditions Theorem 3.1	can we do better than Theorem 3.3?
at most K	sufficient and necessary conditions Theorem 3.1	sufficient condition Theorem 3.3

Chapter 4.

Continuous Localisation

As we have seen in Section 1.2, there are many impressive results in geometry reconstruction, and in the positioning of moving objects in particular. Yet, many of these methods do not provide any guarantees for the recovery of the device position.

For range measurements, position-recovery guarantees exist for traditional lateration in static setups, and the closed-form solutions have been known for decades [2] (see Section 1.2 for more details). Lateration works well for static points, but increasingly often we would like to localise a moving device. In this case, localisation might be too restrictive, because it needs to be performed successfully at each time step. We can obtain better results by enforcing temporal consistency. This way, measurements at a single time instant can inform multiple consecutive positions.

Practical systems predominately recover trajectories by coupling partial lateration with filtering techniques [114]. Although these approaches lead to good performance, they offer us little hope for providing fundamental guarantees for the recovery of the objects continuous trajectory. To the best of our knowledge, there are neither closed-form solutions nor guarantees for the localisation of a moving device.

Although there are many theoretical results in the area of distance geometry, or dynamical distance geometry [174, 175], these results answer questions about the *existence* of device trajectories that match the measurements. This is a dual problem to ours (see Section 1.2), and we do not study it here.

To provide recovery guarantees, we propose to enforce temporal consistency by assuming that the device travels on a smooth trajectory. We analyse the localisation of one or multiple smoothly moving devices by using only range measurements. We provide sufficient and, in some cases, necessary conditions for unique recovery. It is not our goal to provide state-of-the-art algorithms. Rather, we believe that our conditions can be used to compare the performance of existing algorithms, with respect to the theoretical limit.

Contributions: The work in this chapter is based on the joint work of the author of this thesis (MP) with Frederike Dümbgen (FD), Martin Vetterli (MV) and Adam Scholefield (AS) [239, 242]. AS formulated the single trajectory problem with known anchors Section 4.2.1; MP generalised it and found connections to rank one sensing and Kinetic EDMs. MP, FD and AS designed the algorithm for the single trajectory case; FD and MP implemented this algorithm and run simulations; FD worked on real data; MP proved algorithm properties, analysed simulations and wrote this chapter.

The code used to produce the results of this chapter is available at github.com/lcav/continuous-localization.

In Section 4.1, we formulate the continuous localisation problem. As the problem is not convex, we introduce a number of relaxation methods, with a focus on linearisation. We discuss some intrinsic ambiguities in the localisation of devices using only relative distance measurements (Section 4.1.1), and we introduce some tools needed further in this chapter (Section 4.1.2).

These tools, as well as tools from Chapter 3, enable us to formulate and to prove *sufficient* conditions for recovery in specific cases, which we discuss in Section 4.2. In Section 4.2.1, we consider the localisation of a single trajectory using range measurements to known static anchors [242]. We can think of this as having all trajectories constant, except for one. In Section 4.2.2, we stay in the realm of one trajectory and consider "self-localisation". More precisely, we assume that we measure some distances between points on the trajectory. Such a scenario describes a toy problem of molecule reconstruction, given the distances between the atoms, for example, distances obtained from nuclear magnetic resonance (NMR). It also describes a problem finding a path along which multiple devices move, given the distances between the devices. In Section 4.2.3, we provide theoretical guarantees for the recovery of multiple trajectories, when no device positions are known, and we explain the potential practical approaches to the problem. Finally, in Section 4.3, we present potential directions for future work. One of the directions we propose is to integrate different types of measurements in our model.

4.1. Theory

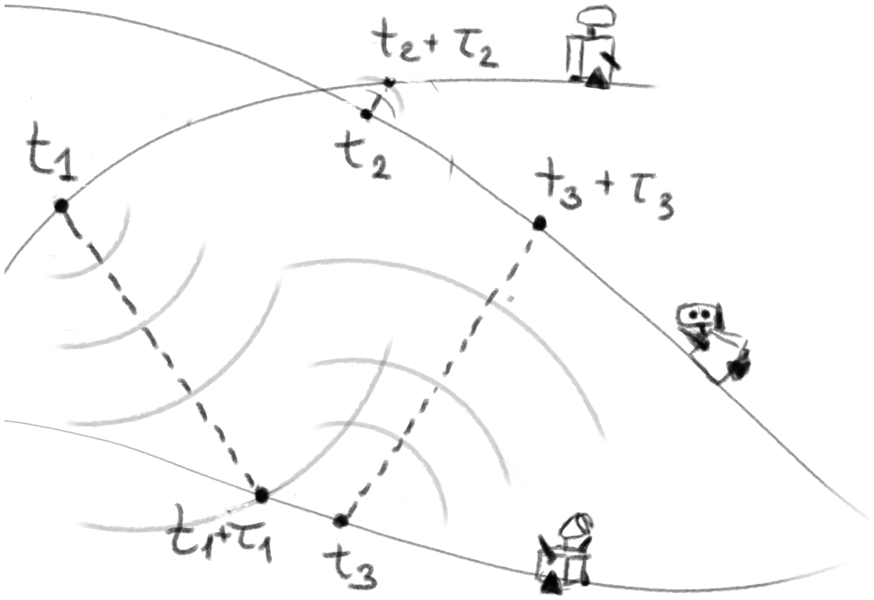


Figure 4.1.: Robots moving along smooth trajectories and measuring distances with time of flight (τ_n) of a signal. Signals can be emitted at non-uniform times (t_b).

In this section, we formally state the problem of continuous localisation in the general case. We explain how we approach the probem by using linearisation and propose methods to recover trajectories. In Section 4.1.1, we discuss some of the ambiguities that arise in the localisation and, in Section 4.1.2, we prove a few results that will be useful later in Section 4.2.

We consider M objects (e.g., robots) that travel along continuous trajectories $\boldsymbol{r}_m(t)$ for $m \in [M]$ in D — usually 2 or 3 — dimensions, see Figure 4.1. The robots measure squared[1] distances between each other at different times:

$$d_{m_1,m_2}(t,\tau) = \|\boldsymbol{r}_{m_2}(t+\tau) - \boldsymbol{r}_{m_1}(t)\|^2. \tag{4.1}$$

Note here that the distance is measured between the position of the first robot at time t and the position of the second robot at time $t+\tau$. This is because, if we use time of flight of a signal or received signal strength to estimate the distance, the signal needs time to propagate through space. As a result, $\tau = d_{m_1,m_2}(t,\tau)/v$, where v is the speed of the signal in the medium. If the robots are on Earth and the propagation time is measured with electromagnetic waves, the propagation time is negligible and we can set $\tau = 0$.

Finally, as usual in this thesis, we assume the trajectories are polynomials of degree at most $K-1$ or real-valued K-bandlimited functions, as described in Section 1.5. For a single trajectory we can write

$$\boldsymbol{r}_m(t) = \boldsymbol{C}_m \boldsymbol{f}(t) \text{ for } m \in [M], \tag{4.2}$$

where $\boldsymbol{C} \in \mathbb{R}^{D\times K}$ is an unknown matrix of coefficients and $\boldsymbol{f}$ is the known vector of basis functions in $\mathcal{F}$ — the family of polynomials or bandlimited functions. We can also write the trajectories in the matrix form:

$$\boldsymbol{R}(t) = \boldsymbol{T}\boldsymbol{f}(t),$$

where $\boldsymbol{R} : \mathbb{R} \to \mathbb{R}^{D\times M}$ describes the trajectories and $\boldsymbol{T} \in \mathbb{R}^{D\times M\times K}$ is an unknown tensor of coefficients of all the trajectories, and where matrices $\boldsymbol{C}_m$ are its slices, $[\boldsymbol{T}]_{:,:,m} = \boldsymbol{C}_m$.

Combining the trajectory model (Equation (4.2)) and the measurement model (Equation (4.1)), we obtain the following distance as function of time t and delay τ

$$d_{m_1,m_2}(t,\tau) = \|\boldsymbol{C}_{m_2}\boldsymbol{f}(t+\tau) - \boldsymbol{C}_{m_1}\boldsymbol{f}(t)\|^2. \tag{4.3}$$

We can also write the distances between the trajectories as a matrix $\boldsymbol{D}(t,\tau)$, such that, as usual, $[\boldsymbol{D}(t,\tau)]_{m_1,m_2} = d_{m_1,m_2}(t,\tau)$. To highlight the dependence of $\boldsymbol{D}$ on $\boldsymbol{R}$, we sometimes write $\boldsymbol{D}^{(\boldsymbol{R})}(t,\tau)$, especially when considering trajectories producing the same measurements in Section 4.1.1.

As we do not measure distances to any object of which we know the global position, we can recover the trajectory only up to to global rotations, reflections and translations. This means we cannot distinguish *equivalent* coefficient matrices (Definition 4.1).

---

[1] Here we use squared distances to for simplicity. What is actually measured effects the noise distribution, and will be our interest in Section 4.2.

Definition 4.1: Equivalent Coefficients

We say that coefficient matrices $\boldsymbol{C}_m \in \mathbb{R}^{D\times K}$ and $\boldsymbol{B}_m \in \mathbb{R}^{D\times K}$ are equivalent, if $\boldsymbol{C}_m = \boldsymbol{U}\boldsymbol{B}_m + \begin{bmatrix}\boldsymbol{v} & \boldsymbol{0} & \cdots & \boldsymbol{0}\end{bmatrix}$ for every $m \in [M]$, where $\boldsymbol{U}$ is an unitary matrix and $\boldsymbol{v}$ is a D dimensional vector. For a trajectory $\boldsymbol{r}_m(t) = \boldsymbol{B}_m\boldsymbol{f}(t)$ we have

$$\begin{aligned}\boldsymbol{C}_m\boldsymbol{f}(t) &= \left(\boldsymbol{U}\boldsymbol{B}_m + \begin{bmatrix}\boldsymbol{v} & \boldsymbol{0} & \cdots & \boldsymbol{0}\end{bmatrix}\right)\boldsymbol{f}(t)\\ &= \boldsymbol{U}\boldsymbol{r}_m(t) + \boldsymbol{v}.\end{aligned}$$

Therefore, the equivalent matrices describe trajectories that differ by *constant* rotation and reflection and *constant* translation.

To sum up, we measure squared distances d_{m_1,m_2} at times t_n (possibly different for different pairs of trajectories), and we ask to recover trajectory coefficients $\boldsymbol{C}_m, m \in [M]$ up to rotations, translations, and reflections.

Problem 4.1: Continuous Localisation

Consider M unknown continuous trajectories $\boldsymbol{r}_m(t) = \boldsymbol{C}_m\boldsymbol{f}(t)$, where $\boldsymbol{f}$ is a vector of basis functions of family $\mathcal{F}$ of either K-bandlimited functions or polynomials of degree at most $K-1$. We are given *squared* distances between these trajectories:

$$d_{m_1,m_2,n} = \|\boldsymbol{C}_{m_2}\boldsymbol{f}(t_n + \tau_{n,m_1,m_2}) - \boldsymbol{C}_{m_1}\boldsymbol{f}(t_n)\|^2 \text{ for } (m_1,m_2,n) \in \mathcal{M}, \tag{4.4}$$

where $\mathcal{M} \subsetneq [M]\times[M]\times\mathbb{N}$ is the finite set of measurement indexes and times t_n are known. The problem of continuous localisation asks us to recover trajectory coefficient $\boldsymbol{C}_m, m \in [M]$ up to an equivalence from Definition 4.1.

In general, M trajectories from K dimensional families in D dimensions have MKD degrees of freedom; and MKD distance measurements should be enough to recover the trajectories. However, not all measurements are equivalent. For example, if we measure MKD distances, but none of them is from (or to) the first trajectory $\boldsymbol{r}_1$, then we cannot recover $\boldsymbol{r}_1$.

Problem 4.1 is formulated in the ideal noiseless case. In the noisy case, we can formulate an equivalent optimisation problem.

Problem 4.2: Continuous Localisation with Noise

Consider M unknown, continuous trajectories $\boldsymbol{r}_m(t) = \boldsymbol{C}_m\boldsymbol{f}(t)$, where $\boldsymbol{f}$ is a vector of basis functions of $\mathcal{F}$ and $\mathcal{F}$ is either the family of K-bandlimited functions or polynomials of degree of at most $K-1$. Find

$$\operatorname*{arg\,min}_{\boldsymbol{C}_1,\ldots,\boldsymbol{C}_M} \sum_{(m_1,m_2,n)\in\mathcal{M}} \left(\widetilde{d}^2_{m_1,m_2,n} - \|\boldsymbol{C}_{m_2}\boldsymbol{f}(t_n + \widetilde{\tau}_{m_1,m_2,n}) - \boldsymbol{C}_{m_1}\boldsymbol{f}(t_n)\|^2\right) \tag{4.5}$$

where $\mathcal{M} \subsetneq [M] \times [M] \times \mathbb{N}$ is the finite set of measurement indexes, times t_n are known and $\widetilde{d}^2_{m_1,m_2,n}$ and $\widetilde{\tau}_{m_1,m_2,n}$ are estimated quantities.

Note that, in Problem 4.2, we optimise the least squares cost between *squared* distances and their *squared* estimates. This makes the problem easier to analyse and easier to solve in the lateration case, but the cost function (4.5) is non-convex. Furthermore, if we assume i.i.d additive noise on the measured distances, then squaring the distances transforms the noise into multiplicative noise, which might cause problems. We discuss noise in more detail in Section 4.2.

If Problem 4.1 has a unique solution, then Problem 4.2 has a unique global minimiser, up to equivalences from Definition 4.1. Therefore, for the remainder of this chapter, we study variations of the following two questions:

1. When does Problem 4.1 have a unique solution and Problem 4.2 a unique minimiser?
2. How can we efficiently find the minimiser of Problem 4.2?

In some cases, such as in the single trajectory case (Section 4.2.1), we present a single method that answers both these questions. In general, however, the method for proving the uniqueness of and the algorithm for finding the minimiser of Problem 4.2 might be different; see Section 4.2.3.

All the methods we consider here can be thought of as simplifications: linearisation, semidefinite relaxation, and (based on [173]) the Euclidean distance matrix (EDM) approach.

Let us begin with linearisation, the main method that we use throughout this chapter. First, we expand the quadratic terms in Equation (4.3):

$$\begin{aligned} d_{m_1,m_2}(t,\tau) = \boldsymbol{f}^\top(t+\tau)\boldsymbol{C}_{m_2}^\top \boldsymbol{C}_{m_2}\boldsymbol{f}(t+\tau) \\ -2\boldsymbol{f}^\top(t+\tau)\boldsymbol{C}_{m_2}^\top \boldsymbol{C}_{m_1}\boldsymbol{f}(t) \\ +\boldsymbol{f}^\top(t)\boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_1}\boldsymbol{f}(t). \end{aligned} \tag{4.6}$$

Then, we introduce a new matrix variable for each product of two matrices $\boldsymbol{L}_{m_1,m_2} = \boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_2}$, $\boldsymbol{L}_{m_1,m_2} \in \mathbb{R}^{K\times K}$ (including products $\boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_1}$). Equation (4.6) is a linear equation in $\boldsymbol{L}_{m_1,m_2}$, and the cost function (4.5) is convex in $\boldsymbol{L}_{m_1,m_2}$. We can find matrices $\boldsymbol{L}_{m_1,m_2}$ by solving the least squares problem. Once we have estimated the matrices $\boldsymbol{L}_{m_1,m_2}$, we will need to factorise them in order to obtain matrices $\boldsymbol{C}_m$.

This operation changes the number of unknowns from MKD to M^2K^2 (M^2 different matrices $\boldsymbol{L}_{m_1,m_2}$, each of size $K \times K$). In Section 4.2.1, we will see that the effective increase in the size of the problem can be linear and not quadratic. The main benefit of linearisation is that the terms in Equation (4.6) are rank-one measurements of matrices $\boldsymbol{L}_{m_1,m_2}$. This enables us to use theorems developed in Chapter 3. As the results from Chapter 3 apply only to the terms Equation (4.6) and not to the sum, we need some additional tools that we develop in Section 4.1.2.

An alternative approach to consider is semi-definite relaxation [204]; it gives more constraints on matrices $\boldsymbol{L}_{m_1,m_1}$ while maintaining the problem convex. Instead of dropping entirely the condition $\boldsymbol{L}_{m_1,m_1} = \boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_2}$, we can replace it with $\boldsymbol{L}_{m_1,m_1} \succcurlyeq \boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_2}$. This enables us to formulate the problem as *semi-definite programming* (Section 1.5).

In fact, we can combine all $\boldsymbol{L}_{m_1,m_1}$ into one large $KM \times KM$ matrix $\boldsymbol{L}$ that is low rank K and positive semi-definite. Then we can drop the rank requirement and use semi-definite programming and obtain possibly higher rank estimate $\widetilde{\boldsymbol{L}}$. There are various methods for recovering a low-rank matrix $\boldsymbol{L}$ from an estimate $\widetilde{\boldsymbol{L}}$. Ultimately, we do not adopt this approach, because it is difficult to obtain recovery guarantees for semi-definite relaxation. We could expect, however, that recovery guarantees for linearisation should apply if we have stronger constraints.

The third option to consider is to ignore the delays τ, as this would enable an elegant formulation of the problem in terms of Gramian $\boldsymbol{G}(t)$ of the trajectories, $\boldsymbol{G}(t) = \boldsymbol{R}^\top(t)\boldsymbol{R}(t)$. Later in this chapter, we will see that, by not including τ_n, we can lose some reconstruction accuracy (Section 4.2.3) and uniqueness results (Section 4.1.1).

When distances depend only on the time (and not on the delay), we will write $\boldsymbol{D}(t) := \boldsymbol{D}(t, 0)$. At each time t matrix $\boldsymbol{D}(t)$ is an Euclidean distance matrix (EDM), thus $\boldsymbol{D}$ is a Kinetic EDM, as defined by Tabaghi [173] and as discussed in Section 1.2. The key insight we gain from the EDM theory is that a KEDM can be written in terms of the Gramian $\boldsymbol{G}(t)$ of trajectories $\boldsymbol{R}(t)$:

$$\boldsymbol{D}(t) = \operatorname{diag}(\boldsymbol{G}(t))\mathbf{1}^\top - 2\boldsymbol{G}(t) + \mathbf{1}\operatorname{diag}(\boldsymbol{G}(t))^\top.$$

Similarly, the Gramian $\boldsymbol{G}$ can be written as a function of $\boldsymbol{D}$. This means there is a one-to-one mapping between the Gramian and EDMs, and we can formulate the problem in terms of the Gramian $\boldsymbol{G}(t)$. This formulation can then be relaxed to a semi-definite program in the coefficients of the Gramian, from which the coefficients of the trajectories can be recovered. For details, see the original work on KEDMs [173].

4.1.1. Intuitions and Challenges

In this section, we consider inherent ambiguities in the reconstruction of the trajectories from distance measurements only. We show that the time delay τ plays a role in the uniqueness of recovery.

In lateration, to uniquely recover positions, we need to know positions of $D + 1$ points — anchors. If no positions are known, but we still measure enough distances between $M \geq D + 1$ points, we can reconstruct their geometry up to rotations, symmetry, and translations.

The most naive way to reconstruct trajectories from distance measurements is to perform lateration at each measurement time. This way, at each measurement time we can have a different translation ($\boldsymbol{c}(t)$), rotation, and symmetry ($\boldsymbol{U}(t)$). This shows that the ambiguities in trajectory recovery can be time dependent; see Figure 4.2. As shown by Tabaghi et al. [173, 262, 263], introducing a polynomial or bandlimited trajectory model can greatly reduce this ambiguity. For example, unless the trajectories are somehow degenerate, the reflection (orientation) will not change through time.

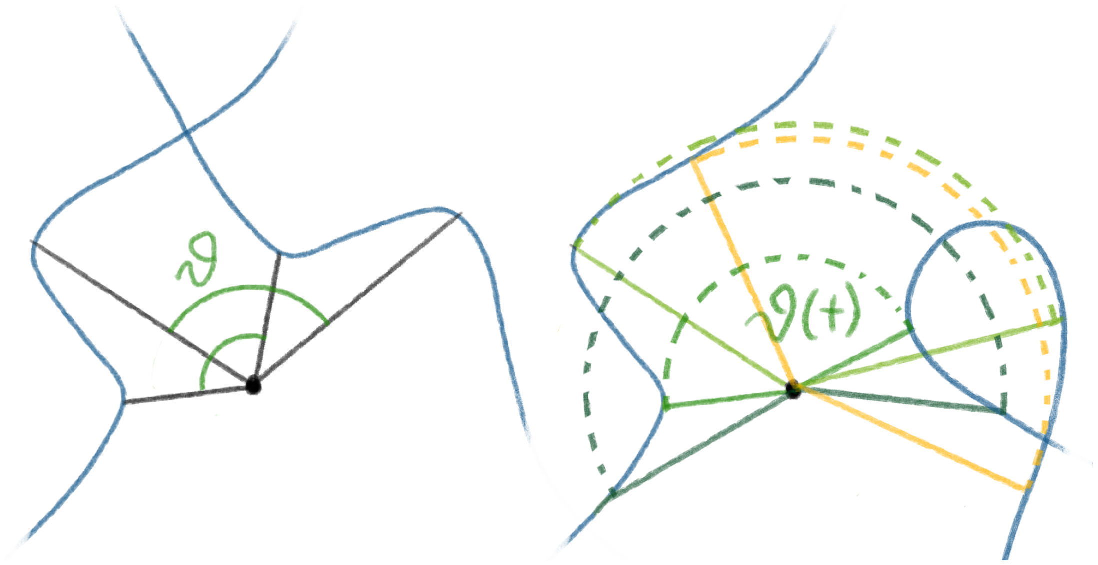


Figure 4.2.: Left: two trajectories (blue) on a plane rotated by a constant angle θ (green). Right: two trajectories (blue) that are rotated by the varying angle $\theta(t)$ (varying darkness). The dashed lines show constant distance from the centre of rotation.

To explain these ambiguities, let us review the existing results that assume the time delay is zero [173]. The intuition that the ambiguity might be time dependent can be formalised as follows:

Lemma 4.1: Tabaghi et al., Preposition 3 [173]

Let $\boldsymbol{R}$ and $\boldsymbol{S}$ be two sets of trajectories, $\boldsymbol{R}, \boldsymbol{S} \in \mathbb{R}^{D\times M}[t]^{a}$. Then, the following statements are equivalent:

1. $\boldsymbol{D}^{(\boldsymbol{R})}(t) = \boldsymbol{D}^{(\boldsymbol{S})}(t)$
2. $\boldsymbol{S}(t) = \boldsymbol{U}(t)\boldsymbol{R}(t) + \boldsymbol{c}(t)\mathbf{1}^\top$ where $\boldsymbol{U}^\top(t)\boldsymbol{U}(t) = \boldsymbol{I}$ and $\boldsymbol{c}(t)$ is a d-dimensional time-varying vector.

---

[a] We took the liberty to add $\times M$ to the exponent. Without it, the statement is trivially true because in this case $\boldsymbol{D}^{(\boldsymbol{R})}(t) = \boldsymbol{D}^{(\boldsymbol{S})}(t) = 0$ (a scalar zero), and because any two trajectories differ by $\boldsymbol{c}(t)\mathbf{1}_1^\top = \boldsymbol{c}(t) = \boldsymbol{S}(t) - \boldsymbol{R}(t)$. Therefore, both statements are always true.

It is believed that the known results on spectral factorisation [211, 264] (see Section 1.5) restrict possible $\boldsymbol{U}(t)$ to constant transformations, both in the polynomial and bandlimited cases [173, 263]. Although this is generally the case, there are some conditions that need to be satisfied. Before we discuss these conditions, let us introduce the following definition:

Definition 4.2: Degenerate Trajectory

By a degenerate set of trajectories, we mean a set of trajectories such that the *affine* subspace spanned by all the points along all trajectories is of a dimension smaller than D:

$$\dim(\operatorname{aff}(\{\boldsymbol{r}_m(t) \text{ for } m \in [M], t \in [0, T]\})) < D$$

If a set of trajectories $\boldsymbol{R}$ is degenerate, then all coefficient matrices $\boldsymbol{C}_m$ have a rank smaller than D.

The rotation $\boldsymbol{U}(t)$ does not have to be constant when there are not enough trajectories. In Example 4.1 and Lemma 4.2, we show that if we have measurements from $M < D$ non-degenerate trajectories, we can find M *degenerate* trajectories that produce the same distance measurements.

Example 4.1: Changing the Matrix of Rotation

Consider two trajectories in three dimensions, $\boldsymbol{R}(t) \in \mathbb{R}^{3\times 2}$, such that $\boldsymbol{r}_1(t) = \begin{bmatrix}1 & t & t^2\end{bmatrix}^\top$ and $\boldsymbol{r}_2(t) = \boldsymbol{0}$. The Gramian of this set of trajectories is simply

$$\boldsymbol{G}^{(\boldsymbol{R})}(t) = \begin{bmatrix}\|\boldsymbol{r}_1(t)\|^2 & 0\\ 0 & 0\end{bmatrix} = \begin{bmatrix}t^4 + t^2 + 1 & 0\\ 0 & 0\end{bmatrix}.$$

Factorising $\|\boldsymbol{r}_1(t)\|^2$, we obtain $\boldsymbol{s}_1(t) = \begin{bmatrix}t^2 - 1 & \sqrt{3}t & 0\end{bmatrix}^\top$ such that $\|\boldsymbol{s}_1(t)\|^2 = \|\boldsymbol{r}(t)\|^2$. Then from this, we can create a new set of trajectories, $\boldsymbol{S} = \begin{bmatrix}\boldsymbol{s}(t) & \boldsymbol{0}\end{bmatrix}$, which results in the same Gramian: $\boldsymbol{G}^{(\boldsymbol{R})} = \boldsymbol{G}^{(\boldsymbol{S})}$.

According to Lemma 4.1, we have $\boldsymbol{S}(t) = \boldsymbol{U}(t)\boldsymbol{R}(t) + \boldsymbol{c}(t)\boldsymbol{1}^\top$ for some unitary matrix $\boldsymbol{U}$ and time varying vector $\boldsymbol{c}(t)$. For two trajectories we have two equations:

$$\begin{aligned}\boldsymbol{s}_1(t) &= \boldsymbol{U}(t)\boldsymbol{r}_1(t) + \boldsymbol{c}(t)\\ \boldsymbol{s}_2(t) &= \boldsymbol{U}(t)\boldsymbol{r}_2(t) + \boldsymbol{c}(t)\end{aligned}$$

As $\boldsymbol{s}_2 = \boldsymbol{r}_2 = \boldsymbol{0}$, we obtain that $\boldsymbol{c}(t) = 0$ for all t. We want to show that $\boldsymbol{U}(t)$ cannot be constant. To do this, we find the only constant matrix $\boldsymbol{U}^{(c)}$ that maps $\boldsymbol{r}$ to $\boldsymbol{s}$, and we show that it is not unitary. Indeed, the only $\boldsymbol{U}^{(c)}$ that satisfies the above equations

$$\boldsymbol{U}^{(c)} = \begin{bmatrix}-1 & 0 & 1\\ 0 & \sqrt{3} & 0\\ 0 & 0 & 0\end{bmatrix}, \tag{4.7}$$

is not unitary. Therefore, the unitary $\boldsymbol{U}(t)$ from Lemma 4.1 cannot be constant, and there is no constant rotation of $\boldsymbol{R}$ to $\boldsymbol{S}$,

In Example 4.1, we consider only polynomial trajectories. By identifying $s = \arccos(t)$ and applying the same reasoning for a polynomial in $\cos(s)$, we can obtain an example for bandlimited functions. Lemma 4.2 also applies to both polynomial and bandlimited functions.

Lemma 4.2: Too Few Trajectories

Let $\boldsymbol{R} : \mathbb{R} \to \mathbb{R}^{D\times M}$ be a non-degenerate trajectory. If $M < 2D$ or if $M < D$ and $\boldsymbol{G}^{(\boldsymbol{R})}$ has real spectral factors, then there exists a trajectory $\boldsymbol{S}$ that has the same Gramian $\boldsymbol{G}^{(\boldsymbol{R})} = \boldsymbol{G}^{(\boldsymbol{S})}$ but $\boldsymbol{S}$ cannot be obtained by a constant rotation of $\boldsymbol{R}$ followed by a time dependent translation.

The lemma above gives a necessary condition on M for the recovery to be unique up to constant rotations and translations. This should not be confused with the case of simple lateration, where the sufficient condition is $M \geq D+1$ (different) distance measurements. In the lateration, the extra measurement is to resolve symmetry/orientation. Here, we permit global symmetry/orientation to be unknown and want to resolve only local transformations.

Proof

First, if $\boldsymbol{R}(t) \in \mathbb{R}^{D\times M}$, then the Gramian $\boldsymbol{G}^{(\boldsymbol{R})}(t) = \boldsymbol{R}^\top(t)\boldsymbol{R}(t) \in \mathbb{R}^{N\times N}$. By the spectral factorisation theorem (Theorem 1.1), this Gramian can be factorised into polynomial matrices $\boldsymbol{G}^{(\boldsymbol{R})}(t) = \boldsymbol{S}_M^*(t)\boldsymbol{S}_M(t)$, where $\boldsymbol{S}_M(t) \in \mathbb{C}^{M\times M}[t]$. If $\boldsymbol{S}_M$ has real coefficients and $M < D$, we could embed $\boldsymbol{S}_M$ into D dimensions by simply appending zeros in the missing dimensions. This way we could obtain a set of trajectories $\boldsymbol{S} \in \mathbb{R}^{D\times N}$ that has the same Gramian as $\boldsymbol{R}$.

If $\boldsymbol{S}_M$ is not real-valued, then, by the corollary to Theorem 1.1 (Corollary 1.2), the Gramian can be factorised into matrices $\boldsymbol{G}^{(\boldsymbol{R})} = \boldsymbol{S}_{2M}^\top \boldsymbol{S}_{2M}$, where $\boldsymbol{S}_M(t) \in \mathbb{R}^{2M\times M}$. In this case, we need $2M < D$ to embed $\boldsymbol{S}_M$ in D dimensions and to obtain $\boldsymbol{S} \in \mathbb{R}^{D\times N}$ that has the same Gramian as $\boldsymbol{R}$.

Consider then $\boldsymbol{U}(t)$, such that $\boldsymbol{S}(t) = \boldsymbol{U}(t)\boldsymbol{R}(t)$. Assume that the unitary matrix $\boldsymbol{U}(t)$ is constant, $\boldsymbol{U}(t) = \boldsymbol{U}^{(c)}$. Then $\boldsymbol{U}$ maps the non-degenerate trajectories $\boldsymbol{R}$ into trajectories $\boldsymbol{S}$ that belong to $M < D$ or to $M < 2D$ dimensional subspaces. This contradicts the assumption that $\boldsymbol{U}^{(c)}$ is unitary, thus $\boldsymbol{U}$ cannot be constant. ■

In Example 4.2 and Lemma 4.3, we show that, even if $M \geq D$, the rotation is not unique when the highest frequencies are not present.

Example 4.2: Swings and Roundabouts

Consider a single trajectory $\boldsymbol{r}(t) \in \mathbb{R}^2$ of a point rotating on a circle around centre $\boldsymbol{c}$.

$$\boldsymbol{r}(t) = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \cos(\Omega t) + \begin{bmatrix} v_2 \\ -v_1 \end{bmatrix} \sin(\Omega t),$$

where $\Omega = 2\pi/T$. We split matrix $\boldsymbol{C} \in \mathbb{R}^{2\times 3}$ into three vectors for simplicity:

$$\boldsymbol{C} = \begin{bmatrix} c_1 & v_1 & v_2 \\ c_2 & v_2 & -v_1 \end{bmatrix}.$$

If we rotate trajectory $\boldsymbol{r}$ in the opposite direction,

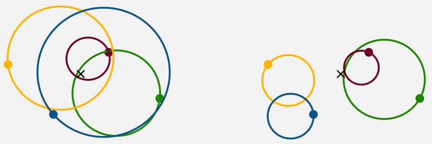
$$\boldsymbol{s}(t) = \begin{bmatrix} \cos(\Omega t) & -\sin(\Omega t) \\ \sin(\Omega t) & \cos(\Omega t) \end{bmatrix} \boldsymbol{r}(t),$$

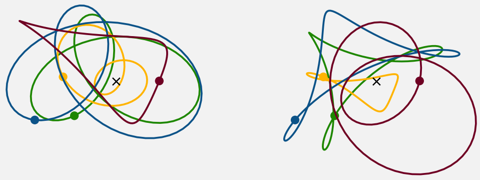
we obtain a trajectory rotating in the opposite direction around $\boldsymbol{v}$:

$$\boldsymbol{s}(t) = \begin{bmatrix} c_1 \\ -c_2 \end{bmatrix} \cos(\Omega t) + \begin{bmatrix} c_2 \\ c_1 \end{bmatrix} \sin(\Omega t) + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}.$$

This trajectory has the same bandwidth as $\boldsymbol{r}$.

Now, consider multiple points that rotate in the same direction as $\boldsymbol{r}$ with the same angular speed — possibly around different origins and with different radii. If we apply the same rotation to all trajectories, the distances between them do not change. This way, we obtain a different set of trajectories with the same bandwidth producing the same distances, as illustrated below on the left.





The same behaviour can be observed in a more complicated example $K = 5$, when the rotation of the highest frequency is a circle, as formalised in Lemma 4.3.

Lemma 4.3: Ambiguous Circles

If the highest frequency component of all trajectories of a bandlimited $\boldsymbol{R}$ is a rotation around the origin with a constant radius and in a plane, then there exists a non-constant unitary $\boldsymbol{U}(t)$, such that $\boldsymbol{U}(t)\boldsymbol{R}(t)$ is bandlimited with the same bandwidth.

Proof

We consider a trajectory $\boldsymbol{R}(t)$ in two dimensions. The same reasoning could be applied to a two-dimensional subspace in higher dimensions. At time t, rotate $\boldsymbol{R}$ by $\alpha\Omega t = \alpha\frac{2\pi}{T}t$,

$\alpha \in \mathbb{Z}$:

$$\boldsymbol{U}(t) = \begin{bmatrix} \cos(\alpha\Omega t) & \sin(\alpha\Omega t) \\ -\sin(\alpha\Omega t) & \cos(\alpha\Omega t) \end{bmatrix}.$$

Similarly as in Example 4.2, we split the coefficients tensor $\boldsymbol{T}$ of trajectories $\boldsymbol{R}$ into K matrices: $\boldsymbol{A}_0, \dots, \boldsymbol{A}_{(K-1)/2}$ and $\boldsymbol{B}_1, \dots, \boldsymbol{B}_{(K-1)/2}$. In this notation, the component of $\boldsymbol{R}$ that rotates at frequency k is

$$\boldsymbol{A}_k \cos(k\Omega t) + \boldsymbol{B}_k \sin(k\Omega t).$$

After rotation, this component becomes

$$\begin{aligned} &\frac{1}{2}\left((\boldsymbol{A}_k + \boldsymbol{E}\boldsymbol{B}_k)\cos((k-\alpha)\Omega t) + (-\boldsymbol{E}\boldsymbol{A}_k + \boldsymbol{B}_k)\sin((k-\alpha)\Omega t)\right) \\ +&\frac{1}{2}\left((\boldsymbol{A}_k - \boldsymbol{E}\boldsymbol{B}_k)\cos((k+\alpha)\Omega t) + (\boldsymbol{E}\boldsymbol{A}_k + \boldsymbol{B}_k)\sin((k+\alpha)\Omega t)\right), \end{aligned}$$

where $\boldsymbol{E} = \left[\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right]$ is the matrix of rotation by $\pi/2$. The rotation created two frequencies from frequency k: $k + \alpha$ and $k - \alpha$. If one of these frequencies is larger than K, its component has to be zero. Otherwise, the trajectory would not be K-bandlimited. Without loss of generality, assume that $k + \alpha > K$. Then we have

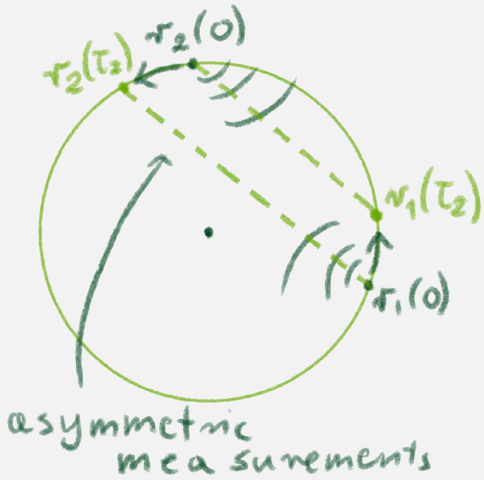
$$\boldsymbol{A}_k = \boldsymbol{E}\boldsymbol{B}_k \quad \text{and} \quad \boldsymbol{E}\boldsymbol{A}_k = \boldsymbol{B}_k.$$

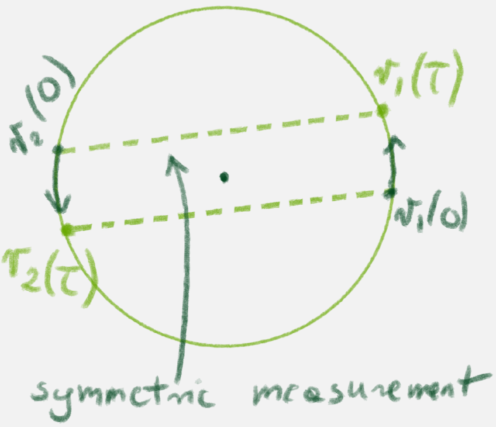
These two equations are equivalent, because $\boldsymbol{E}^2 = -\boldsymbol{I}$. They enforce that the cosine component is a rotated sine component, hence that at frequency k the trajectory is a circle. The smallest α for which $\boldsymbol{U}(t)$ is not constant is 1. Therefore, we need at least the highest frequency to satisfy the above equation. ■

So far, we have considered ambiguities that exist when we ignore the time delay τ. But, does re-introducing τ solve this problem? In the case of rotation, the time delay helps by breaking symmetry; see Example 4.3. In the case when $M < D$, described in Lemma 4.2 and Example 4.1, the ambiguity persists even if we re-introduce the time delay τ, unless we allow measurements of distances within the same trajectory; see Example 4.4.

Example 4.3: Breaking Symmetry

Consider two trajectories that move around the circle at the same speed, as shown below on the left. If we ignore the time delay, they are always at the same distance $\sqrt{2}r$ from each other, hence we cannot tell if they are moving nor in which direction. However, as $\boldsymbol{r}_2$ moves in the direction away from $\boldsymbol{r}_1$ (and $\boldsymbol{r}_1$ moves towards $\boldsymbol{r}_2$), the signal from $\boldsymbol{r}_1$ will take longer to reach $\boldsymbol{r}_2$ than the signal moving the other way around. And we can discern that they move in the direction of $\boldsymbol{r}_1$, as shown below. The remaining ambiguity (reflection) is not time dependent.





However, if the points are on opposite sides of the circle, as depicted on the right, both measured distances will be shorter than the diameter of the circle. And we cannot determine if, or in which direction the points move.

Example 4.4: Distance from the Centre

Consider the same trajectories $\boldsymbol{R}$ and $\boldsymbol{S}$ as in Example 4.1. The distances measured with the time delay for $\boldsymbol{R}$ would be exactly the same for $\boldsymbol{S}$ and equal

$$\begin{aligned} d_{1,2}(t) &= \|\boldsymbol{r}_2(t + d_{1,2}/v) - \boldsymbol{r}_1(t)\|^2 = t^4 + t^2 + 1 \\ d_{2,1}(t) &= \|\boldsymbol{r}_1(t + d_{2,1}/v) - \boldsymbol{r}_2(t)\|^2 = (t + d_{2,1}/v)^4 + (t + d_{2,1}/v)^2 + 1. \end{aligned}$$

It is impossible to distinguish between these two trajectories in this case. However, if we allow the measurements along the trajectory, we will have

$$\begin{aligned} d_{1,1}(t) &= \|\boldsymbol{r}_1(t + \tau_{\boldsymbol{r}}) - \boldsymbol{r}_1(t)\|^2 = (\tau_{\boldsymbol{r}}^2 - 2\tau_{\boldsymbol{r}} t)^2 + \tau_{\boldsymbol{r}}^2 \text{ for } \boldsymbol{R} \\ d_{1,1}(t) &= \|\boldsymbol{s}_1(t + \tau_{\boldsymbol{s}}) - \boldsymbol{s}_1(t)\|^2 = (\tau_{\boldsymbol{r}}^2 - 2\tau_{\boldsymbol{r}} t)^2 + 3\tau_{\boldsymbol{s}}^2 \text{ for } \boldsymbol{S}. \end{aligned}$$

Knowing a sufficient number of measurements of this form enables us to distinguish between $\boldsymbol{S}$ and $\boldsymbol{R}$, but obtaining such measurements is not possible without the time delay τ.

Before we conclude this example, we have to ask, Are measurements $d_{1,1}$ within the trajectory realistic? In order to measure such distances, the device has to move faster than the speed of the signal. However, there are planes that fly faster than the speed of sound. And, such distances can be acquired from other sorts of measurements, such as forces in Section 4.2.2, or IMU measurements in Box 4.8. Hence, there is nothing in principle that prevents us from including them.

As noted at the beginning, KEDMs are also not unique up to a drift, i.e., time-dependent, trajectory-independent translation. For the same reason as in Example 4.3, considering the time delay can break symmetry and lead to uniqueness.

In our last example, we explain the intuition about when it is realistic to include the time delay τ.

Example 4.5: GPS satellites

GPS satellites make a full circle around Earth in 12 h. As a result, their trajectories are the same every day. They orbit at an altitude of about 20.180 km, which puts them at about 26.551 km from Earth's centre. They communicate via electromagnetic waves, i.e., at the speed of light. Can we, in this case, neglect the time delay τ?

We can estimate the time it takes for the signal from the satellite to reach Earth's surface to be between $\tau = 60\,\mathrm{ms}$ to $\tau = 0.1\,\mathrm{s}$. This delay might seem negligible, but during this time the satellite will travel

$$s = 2\pi \cdot 26\,551\,\mathrm{km}\frac{\tau}{12\,\mathrm{h}} > 230\,\mathrm{m}.$$

Although this is very small compared to the satellites' altitude, it is not an error we want the GPS to make. The GPS satellites' positions can be predicted up to a few metres, two days ahead [1].[a]

---

[a] A reader familiar with work by Tabaghi et al. [173] will recall that they consider satellite localisation without the time delay. However, they use arbitrary distances, so it is hard to tell if it is realistic.

4.1.2. Theorems and Proofs

In this section, we analyse linearisation and introduce two lemmas that will be useful in Section 4.2. We keep these lemmas as general as possible. This makes them quite abstract and mathematical. For the readers interested in practical implications of our work, the applications presented in the next section will provide context and, hopefully, some intuition about the theorems.

Consider Equation (4.6). Using properties of trace (see Section 1.5), we write Equation (4.6) as an inner product, introducing $\boldsymbol{L}_{m_1,m_2} = \boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_2}$ for all pairs $m_1, m_2 \in [M] \times [M]$:

$$\begin{aligned} d_{m_1,m_2}(t,\tau) = \langle \boldsymbol{f}(t+\tau)\boldsymbol{f}^\top(t+)\tau, \boldsymbol{L}_{m_2,m_2}\rangle \\ -2\langle \boldsymbol{f}(t)\boldsymbol{f}^\top(t+\tau), \boldsymbol{L}_{m_1,m_2}\rangle \\ +\langle \boldsymbol{f}(t)\boldsymbol{f}^\top(t), \boldsymbol{L}_{m_1,m_1}\rangle. \end{aligned}$$

Consider now all distances measured between points of two trajectories. We can put all the equations for these distances in canonical form:

$$\boldsymbol{d}_{m_1,m_2} = \begin{bmatrix} \boldsymbol{\Gamma}_{m_1,m_1} & -2\boldsymbol{\Gamma}_{m_2,m_1} & \boldsymbol{\Gamma}_{m_1,m_1} \end{bmatrix} \begin{bmatrix} \mathrm{vec}(\boldsymbol{L}_{m_1,m_1}) \\ \mathrm{vec}(\boldsymbol{L}_{m_1,m_2}) \\ \mathrm{vec}(\boldsymbol{L}_{m_2,m_2}) \end{bmatrix}, \tag{4.8}$$

where by $\boldsymbol{d}_{m_1,m_2}$ we denote the vector of all measured distances between trajectory $\boldsymbol{r}_{m_1}$ and trajectory $\boldsymbol{r}_{m_2}$, and

$$\boldsymbol{\Gamma}_{m_1,m_2} = \begin{bmatrix} \operatorname{vec}(\boldsymbol{f}(t_1)\boldsymbol{f}^\top(t_1+\tau_1)) \\ \operatorname{vec}(\boldsymbol{f}(t_2)\boldsymbol{f}^\top(t_2+\tau_2)) \\ \vdots \\ \operatorname{vec}(\boldsymbol{f}(t_N)\boldsymbol{f}^\top(t_N+\tau_N)) \end{bmatrix}.$$

A reader familiar with Chapter 3 will recognize that matrix $\boldsymbol{\Gamma}_{m_1,m_2}$ for $m_1 \neq m_2$ has the same form as matrix $\boldsymbol{\Gamma}$ in Equation (3.5). We can directly apply Theorem 3.1 or Theorem 3.3 to it, provided that we have enough measurements of the correct kind. This would give us the invertibility of the middle term from Equation (4.8).

The side terms $\boldsymbol{\Gamma}_{m_1,m_1}$ and $\boldsymbol{\Gamma}_{m_2,m_2}$ also appeared in Chapter 3, in the form of Example 3.6, hence we can expect them to not be full (column) rank. We formalise this in Lemma 4.4 and Box 4.6. If the side terms are not full (column) rank, then the whole system will not be full column rank. Therefore, its solution is not unique. However in Sections 4.2.1 and 4.2.2, we will show that this might not necessarily be a problem, because in order to recover trajectories we do not need to recover all matrices $\boldsymbol{L}_{m_1,m_2}$.

The existence of a unique solution to Equation (4.8) does not depend only on the ranks of $\boldsymbol{\Gamma}_{m_1,m_2}$ but also on how the concatenation affects the rank. Indeed, for polynomials and bandlimited systems, we can “glue” matrices $\boldsymbol{\Gamma}_{m_1,m_2}$ rather easily, as we show in Lemma 4.5.

We begin with a lemma concerning the symmetric $\boldsymbol{\Gamma}_{m,m}$.

Lemma 4.4

For both polynomials of degree at most $K-1$ and (real or complex) bandlimited functions of bandwidth K, there exists a (constant) linear transformation $\boldsymbol{\Phi}$ such that

$$\operatorname{vec}(\boldsymbol{f}(t)\boldsymbol{f}^\top(t)) = \boldsymbol{\Phi}\boldsymbol{f}^{2K-1}(t),$$

where $\boldsymbol{f}^{2K-1}(t)$ is the basis vector of a larger space $\mathcal{F}^{2K-1}$ with $2K-1$ degrees of freedom: space of polynomials of degree at most $2K-2$ and $2K-1$-bandlimited functions.

Furthermore, if $\boldsymbol{f} \in \mathcal{F}$ are extended to multiple variables, then these results still hold, with $\mathcal{F}^{2K-1}$ and $\boldsymbol{f}^{2K-1}$ also being functions of multiple variables.

This means that the columns of $\boldsymbol{\Gamma}_{m,m}$ are samples taken at times t_n of some functions in a $2K-1$-dimensional space. Therefore, they can only span a $2K-1$ dimensional subspace of $\mathbb{R}^{K^2}$ and $\boldsymbol{\Gamma}_{m,m}$ cannot be full rank.

Proof

In the polynomial case, it is enough to note that since $f_k(t) = t^{k-1}$, $k \in [K]$ are monomials, then all entries of $\operatorname{vec}(\boldsymbol{f}_m(t)\boldsymbol{f}_m^\top(t))$ are also monomials of degree at most $2(K-1)$. Therefore, they can be written in a basis of the space polynomials of degree at most K, and we can define $\boldsymbol{\Phi}$ to be the synthesis operator. The same applies to complex bandlimited functions. For real bandlimited functions, observe that multiplication becomes a convolution in the frequency domain. If we convolve two length-K sequences, we obtain a length $2K-1$ sequence, hence the resulting functions are $2K-1$-bandlimited. Again, $\boldsymbol{\Phi}$ is the synthesis operator, from the basis of $2K-1$-bandlimited functions to $\boldsymbol{f}(t)$.

In the multivariate case, the entries $\boldsymbol{f}$ are polynomials in entries $\boldsymbol{z} \in \mathbb{R}^P$ of degree at most $K-1$ or multivariate bandlimited functions of bandwidth K in each coordinate. We can apply the above-mentioned reasoning to each coordinate and see that the resulting functions are polynomials of degree at most $2K-2$ in each coordinate or multivariate bandlimited functions of bandwidth $2K-1$ in each coordinate. ■

Lemma 4.4 and the proof above can be generalised to other families of functions. In this chapter, as we assume that trajectories are polynomial or bandlimited, we leave the discussion about other families to Box 4.6.

Now, that we know how the entries of $\boldsymbol{\Gamma}_{m,m}$ look, we introduce our second lemma that explains how to glue polynomial columns to a matrix $\boldsymbol{A}$ that already has a full column rank.

Lemma 4.5: Expanding a Matrix with Polynomials

Let $\mathcal{F}$ be a family of polynomials or bandlimited functions, let $\boldsymbol{A}_r \in \mathbb{R}^{N\times r}$ be a full rank matrix, and let $\boldsymbol{A}_{r+1}$ be a matrix constructed by first appending any column $\boldsymbol{c}_{r+1}$ to $\boldsymbol{A}_r$ and then appending a row of the form

$$\begin{bmatrix} p_1(t) & \dots & p_{r-1}(t) & p_{r+1}(t) \end{bmatrix},$$

where $p_j \in \mathcal{F}$. In the polynomial case, we also require the degree of p_{r+1} to be greater than the degree of any other p_j. In the bandlimited case, we require either the highest frequency of p_{r+1} to be outside the bandwidth of p_j or, if it is equal, then p_{r+1} must be a different phase than p_j in the highest frequency. Then, for almost all t, matrix $\boldsymbol{A}_{r+1}$ is full rank.

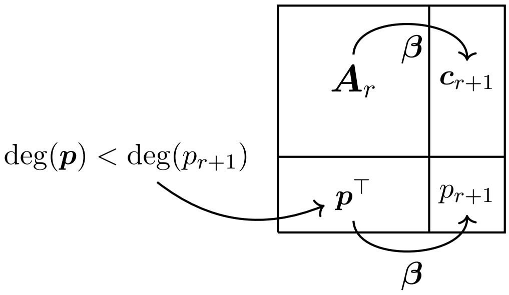


Figure 4.3.: Building matrix $\boldsymbol{A}_{r+1}$ from matrix $\boldsymbol{A}_r$. For $\boldsymbol{A}_{t+1}$ to not be full rank, $p_{r+1}(t)$ has to be related to the rest of the row via the same linear combination $\boldsymbol{\beta}$ that $\boldsymbol{c}_{r+1}$ relates to $\boldsymbol{A}_r$

Proof of Lemma 4.5

Let $\boldsymbol{c}_r$ be the appended column. Since $\boldsymbol{A}_r$ is full rank, there is a unique linear combination $\boldsymbol{\beta}$, such that $\boldsymbol{A}_r\boldsymbol{\beta} = \boldsymbol{c}_{r+1}$ see Figure 4.3. For $\boldsymbol{A}_{r+1}$ to be not full rank, it would mean that the same linear combination $\boldsymbol{\beta}$ of the added row would have to be equal to the last diagonal element of $\boldsymbol{A}_{r+1}$, $p_{r+1}(t)$. We could write it as

$$p_{r+1}(t) - \begin{bmatrix} p_1(t) & \dots & p_r(t) \end{bmatrix} \boldsymbol{\beta} = 0. \tag{4.9}$$

The left-hand side of this equation is not constantly zero. In the polynomial case, it cannot be constantly zero, because the coefficient of the monomial with the highest power of t is zero in p_j but is not zero for p_{r+1}. In the bandlimited case, the highest frequency of p_{r+1} cannot be subtracted by p_j, because it is either not present there or in a different phase. Therefore, the measure of the set of zeros of Equation (4.9) is zero. As a result, $\boldsymbol{A}_{i+1}$ is full rank for almost all t. ■

Want to know more? 4.6: Musings about Polynomials and Bandlimited Functions

A perceptive reader might have noticed that we often obtain the same results for polynomials and bandlimited functions. Here discuss the we similarities between those classes of functions, but we do not give a definitive answer.

Some of the similarities come from the fact that real and complex-valued bandlimited functions, as well as real-valued polynomials, are slices of complex-valued polynomials. However, some functions, e.g. the logarithm, behave differently on real numbers than on complex numbers with absolute value 1.

The results in this section can be formulated more generally using polynomial rings. In abstract algebra *polynomials* and *polynomial functions* are two different creatures, but here we consider function spaces that have structure of the ring (see Box 2.6).

A canonical example of polynomial rings is, of course, the family of polynomials $f_k(t) = t^k$. We can set $\mathcal{F}$ to be the space of polynomials of degree smaller than K and $\mathcal{R}$ to be ring of polynomials $\mathcal{R}[t]$. Trigonometric polynomials, in other words, real symmetric bandlimited functions on $(-\pi, \pi)$, can be extended to $\mathcal{R} = \mathcal{R}[\cos(t)]$. And, similarly, real bandlimited functions on $(-\pi, \pi)$ can be extended to $\mathcal{R} = \mathcal{R}[X, Y]/_{[X^2+Y^2-1]}$, where we identify $X = \cos(t)$ and $Y = \sin(t)$. We can also take $\mathcal{F}$ to be the space of complex bandlimited functions on $(-\pi, \pi)$ and $\mathcal{R}$ to be $\mathcal{R}[e^{it}]$

The proof of Lemma 4.4 in terms of polynomial rings could be formulated as follows: Assume that $\deg(f_k) \leq \boldsymbol{\alpha}$, for some $\boldsymbol{\alpha} \in \mathbb{N}^m$. Then the entries of $\boldsymbol{f}(t)\boldsymbol{f}(f)^\top$ are also polynomials over $\mathbb{F}$, of degrees $\leq 2\boldsymbol{\alpha}$, because $\deg(ab) = \deg(a) + \deg(b)$ for $a, b \in \mathcal{R}$ or $\deg(ab) = \deg(a) + \deg(b)$ up to the appropriate modulo relation for a quotient ring. This means that, among M different vectors $\operatorname{vec}(\boldsymbol{f}(t)\boldsymbol{f}^\top(t))$ of degrees up to $\boldsymbol{\alpha}$, there are **at most** $(a_0+1)(a_1+1)(a_2+1)\dots$ linearly independent vectors. If, additionally, for every degree $\boldsymbol{\beta} < \boldsymbol{\alpha}$ there is an index k such that $\deg(f_k) = \boldsymbol{\beta}$, then each degree $\leq 2\boldsymbol{\alpha}$ will have a corresponding entry in $\boldsymbol{f}(t)\boldsymbol{f}^\top(t)$.

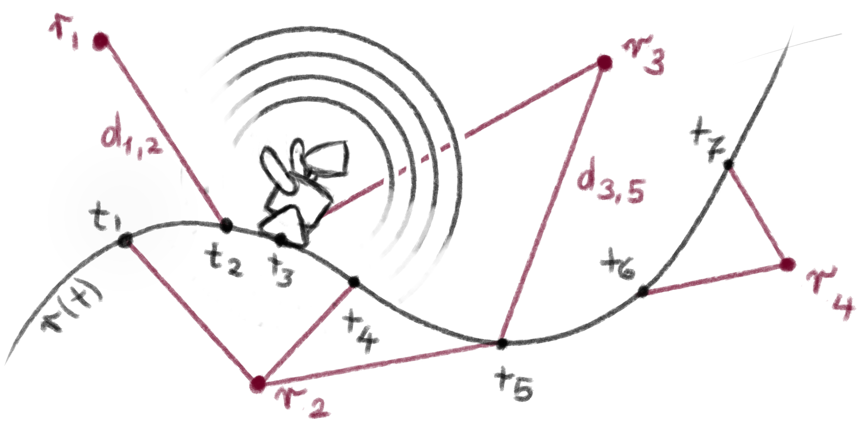


Figure 4.4.: Localisation of a single device using known anchors. The device moves on the parametric trajectory $\boldsymbol{r}(t)$ (black). At given times, the device measures distances to one or more anchors $\boldsymbol{r}_m$ (red).

Another way to look at polynomials and bandlimited functions is through the convolution. Multiplication of bandlimited functions is a convolution in the spectral domain, while multiplication of polynomials is a convolution of sequences of their coefficients.

Finally, in the same way as algebraic geometry studies the sets of zeros of polynomials, analytic geometry studies the sets of zeros of analytic functions, that include bandlimited functions.

4.2. Applications

In this section, we show how to use results from Section 4.1.2 and Chapter 3 in practice. Our main application is trajectory localisation from known anchors (Section 4.2.1), based on our previous publication [242]. We propose a trajectory recovery algorithm and show its performance on simulated and real data. In Section 4.2.2, we present new results concerning trajectory recovery using distance measurements between the points of the trajectory. We illustrate these results with simulations.

4.2.1. Localisation with Anchors

In this section, we present a number of conditions under which the unique recovery of a single trajectory $\boldsymbol{r} = \boldsymbol{C}\boldsymbol{f}$ is possible. We also compare the performance, on simulated and real data, of different localization algorithms.

In our model, anchors are known constant trajectories — known points as in our publication [242]. However, in the work by Tabaghi [173], anchors are assumed to be moving, and their whole trajectories are known. Here, anchors can be independent but can also be positions along a known trajectory, at known times.

In our case, there is only one trajectory to localise, hence we drop the m index of the matrix $\boldsymbol{C}$. By $\boldsymbol{r}_m(t) = \boldsymbol{r}_m, m \in [M]$, we denote the known trajectories (anchors); and the unknown trajectory is the $(M+1)$-st trajectory. We define t_n as the time of the moving device, thus we do not have to use the variable τ. This gives us the formulation described in Problem 4.3.

Problem 4.3: Localisation with Anchors

Consider an unknown continuous trajectory $\boldsymbol{r}(t) = \boldsymbol{C}\boldsymbol{f}(t)$, where $\boldsymbol{f}$ is a vector of basis functions of family $\mathcal{F}$ of either K-bandlimited functions or polynomials of degree of at most $K-1$. We are given *squared* distances between this trajectories and known anchor points $\boldsymbol{r}_m$, $m \in [M]$:

$$d_{m,n} = \|\boldsymbol{r}_m - \boldsymbol{C}\boldsymbol{f}_n\|^2 \text{ for } (m,n) \in \mathcal{M},$$

where $\mathcal{M} \subseteq [M] \times [N]$ contains the pair (m,n) if at time t_n the distance from the trajectory to the m-th anchor was measured and the times t_n, $n \in [N]$ are known. The problem of continuous localisation with anchors asks us to recover trajectory coefficient $\boldsymbol{C}$.

Introducing $\boldsymbol{L} = \boldsymbol{C}^\top\boldsymbol{C}$ and vectorising all the matrices, (see Section 1.5 for all the tricks), we write the linearised equations on the squared distances:

$$d_m - \|\boldsymbol{r}_m\|^2 = \begin{bmatrix} -2\operatorname{vec}(\boldsymbol{r}_m\boldsymbol{f}_n^\top)^\top & \operatorname{vec}(\boldsymbol{f}_n\boldsymbol{f}_n^\top)^\top \end{bmatrix} \begin{bmatrix} \operatorname{vec}(\boldsymbol{C}) \\ \operatorname{vec}(\boldsymbol{L}) \end{bmatrix}. \tag{4.10}$$

We will again write this equation in matrix form, but before we do, let us first consider $\operatorname{vec}(\boldsymbol{f}_n\boldsymbol{f}_n^\top)$. By Lemma 4.4 we have

$$\operatorname{vec} \boldsymbol{f}_n\boldsymbol{f}_n^\top = \boldsymbol{\Phi}\boldsymbol{f}_n^{(2K-1)} = \boldsymbol{\Phi}\begin{bmatrix} \boldsymbol{f}_n^\top & f_K(t_n) & \dots & f_{2K-1}(t_n) \end{bmatrix}^\top,$$

where we use the fact that first K elements of $\boldsymbol{f}^{(2K-1)}(t)$ belong to $\mathcal{F}$. Therefore, Equation (4.10) can be written as

$$d_{m,n} - \|\boldsymbol{r}_m\|^2 = \begin{bmatrix} \operatorname{vec}(\boldsymbol{r}_m\boldsymbol{f}_n^\top)^\top & \boldsymbol{f}_n^\top & f_K(t_n) & \dots & f_{2K-1}(t_n) \end{bmatrix} \begin{bmatrix} \operatorname{vec}(\boldsymbol{C}) \\ -\frac{1}{2}\boldsymbol{\Psi}\operatorname{vec}(\boldsymbol{L}) \end{bmatrix}.$$

Now, we can identify the first two elements of the measurement vector as rows of the matrix $\boldsymbol{\Gamma}_R \in \mathbb{R}^{N \times (D+1)K}$:

$$\begin{bmatrix} \operatorname{vec}(\boldsymbol{r}_m\boldsymbol{f}_n^\top)^\top & \boldsymbol{f}_n^\top \end{bmatrix} = \operatorname{vec}\left(\begin{bmatrix} \boldsymbol{r}_m \\ 1 \end{bmatrix} \boldsymbol{f}_n^\top\right)^\top, \tag{4.11}$$

where the equality is due to the fact that $\boldsymbol{f}_n$ can be expressed as $1 \cdot \boldsymbol{f}_n$. Using the remaining elements, we create the second matrix $\boldsymbol{\Gamma}_F \in \mathbb{R}^{N \times K-1}$ with rows $\begin{bmatrix} f_K(t_n) & \dots & f_{2K-2}(t_n) \end{bmatrix}$. Putting this all together we have

$$d_{m,n} - \|\boldsymbol{r}_m\|^2 = \begin{bmatrix} \boldsymbol{\Gamma}_R & \boldsymbol{\Gamma}_F \end{bmatrix} \begin{bmatrix} \operatorname{vec}(\boldsymbol{C}) \\ -\frac{1}{2}\boldsymbol{\Psi}\operatorname{vec}(\boldsymbol{L}) \end{bmatrix}. \tag{4.12}$$

If we knew that matrix $[\mathbf{\Gamma}_R \, \mathbf{\Gamma}_F]$ were invertible, we could simply solve this system of linear equations. This is the basis for Algorithm 4.1, where instead of calculating $\mathbf{\Psi}$, we use singular value decomposition (SVD, see Section 1.5) to obtain the same dimensionality reduction, but hopefully with better numerical properties. We later provide a number of conditions under which Algorithm 4.1 recovers a unique solution.[2]

Algorithm 4.1: Relax & Recover

```
Input: Anchor coordinates $\boldsymbol{r}_m$, distance measurements $d_{m,n}$ and times $t_n$.
Output: Trajectory coefficients $\widetilde{\boldsymbol{C}}$.
  $\boldsymbol{f}_n \leftarrow \begin{bmatrix} f_0(t_n) & \dots & f_{K-1}(t_n) \end{bmatrix}$
  $b_{n,m} \leftarrow \left( \|r_m\|^2 - d_n^2 \right)/2$ for $(m,n) \in \mathcal{M}$
  $\boldsymbol{b} \leftarrow \text{concatenate}(b_{n,m})$
  $\mathbf{\Gamma}_1 \leftarrow \text{concatenate}\left(\text{vec}(\boldsymbol{r}_m \boldsymbol{f}_n^\top)^\top\right)$
  $\mathbf{\Gamma}_2 \leftarrow \text{concatenate}\left(\text{vec}(\boldsymbol{f}_n \boldsymbol{f}_n^\top)^\top\right)$
  $\boldsymbol{U}, \mathbf{\Sigma}, \boldsymbol{V} \leftarrow \text{SVD}\left(\mathbf{\Gamma}_2\right)$
  $\boldsymbol{A} \leftarrow \text{concatenate}\left(\mathbf{\Gamma}_1, \boldsymbol{U}\mathbf{\Sigma}\right)$
  $\widetilde{\boldsymbol{x}} \leftarrow \text{linsolve}\left(\boldsymbol{A}, \boldsymbol{b}\right)$
  $\widetilde{\boldsymbol{C}} \leftarrow \text{reshape}\left(\widetilde{\boldsymbol{x}}[1:DK], (D, K)\right)$
```

We make some general observations regarding Equation (4.12). Then, we state the sufficient conditions of recovery of $\boldsymbol{C}$ (Corollaries 4.6 to 4.8), and only then do we prove all the conditions.

First, the matrix $\mathbf{\Gamma}_R$ has the same structure as matrix $\mathbf{\Gamma}$ from Section 3.1 (Equation (3.5)); hence, we can apply Theorem 3.1 or Theorem 3.3 to it. To do this, we need the vectors $[\boldsymbol{r}_m^\top \, 1]^\top$ to be a full-spark frame. This is the case when anchors $\boldsymbol{r}_m$ are non-degenerate, as we illustrate in Figure 4.5

Definition 4.3: Degenerate Anchors

Anchors $\boldsymbol{r}_m \in \mathbb{R}^D, m \in [M]$ are said to be degenerate if there are D anchors among them such that the *affine* subspace spanned by them is of dimension smaller than D.

The assumption that anchors are non-degenerate is only slightly stronger than the common requirement that not all anchors lie on the same affine subspace. Randomly placed anchors will satisfy this condition almost surely. Moreover, if the anchors are generated by some trajectory, this assumption will be satisfied for almost all sequences of times $(t_n)_1^N$ if the trajectory is itself non-degenerate (Definition 4.2).

Second, the matrix $\mathbf{\Gamma}_F$ was constructed to be full rank, given enough measurements. However, $\mathbf{\Psi}$ is not invertible, and we cannot directly recover $\boldsymbol{L}$. But, we do not need to do so if we recover $\boldsymbol{C}$ correctly. An advantage of $\mathbf{\Psi}$ is that it reduces the dimension of our system. Indeed, $[\mathbf{\Gamma}_R \, \mathbf{\Gamma}_F]$ has $(D+1)K + (K-1)$ columns.

---

[2] Note that in our publication [242] the algorithm explicitly checked the uniqueness conditions. Here we omit it, because we have multiple different sufficient conditions that all apply to the same algorithm.

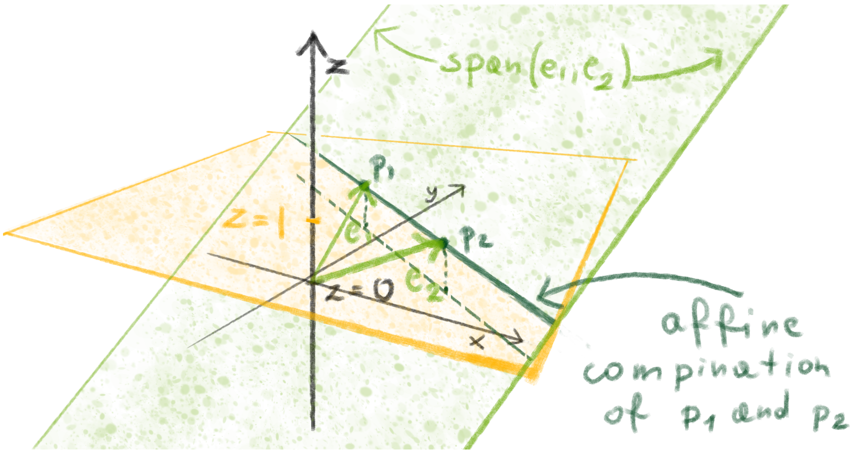


Figure 4.5.: We can identify every point in $\mathbb{R}^D$ with a point in an affine subspace of $\mathbb{R}^{D+1}$ such that the last coordinate is equal to one. Here for illustration $D = 2$. Then, every *affine* combination of points in $\mathbb{R}^D$ corresponds to a slice of a *linear* combination of their embeddings $\mathbb{R}^{D+1}$.

Furthermore, both matrices $\mathbf{\Gamma}_R$ and $\mathbf{\Gamma}_F$ are full rank but, to show that the concatenated matrix $[\mathbf{\Gamma}_R\,\mathbf{\Gamma}_F]$ is also full rank, we need to use Lemma 4.5.

Finally, even without the relaxation, we would expect to need $K(D+1)$ measurements for recovery. Indeed, to recover a trajectory of complexity K, we can independently localize K points along it; and to localize a single point, we need $D+1$ distance measurements. Our below-mentioned conditions all require $O(KD)$ measurements, even though they apply to the relaxed system of equations.

To obtain our first condition, we assume that the measurement times are unique, as in our publication [242]. This setup allows for non-synchronised measurements and, if the device can send only one signal at a time, is realistic.

Corollary 4.6: Theorem 1 in [242]

Assume that $|\mathcal{M}| \geq K(D+2)-1$, the anchors are non-degenerate and the measurement times are unique. Let k_m be the number of measurements in which the m-th anchor is used. If the following condition is satisfied

$$\sum_{m=1}^{M} \min(k_m, K) \geq K(D+1), \tag{4.13}$$

then, for almost all sequences $(t_n)_1^N$, there is a unique matrix $\boldsymbol{C}$ satisfying Problem 4.3.

The second condition (Corollary 4.7) relaxes the requirement that the anchors are different. This might be useful when the measurement device is already designed to make multiple measurements simultaneously for the purpose of lateration. Corollary 4.7 requires oversampling four times. Tighter bounds can be found at the expense of complexity. Before we prove the corollary, we discuss the exact reasons for this oversampling.

Corollary 4.7: Corollary of Theorem 3.3

Assume that $|\mathcal{M}| \geq 2(D+1)(2K-1)$ and that the anchors are non-degenerate. Furthermore, assume that the following conditions are satisfied

$$\sum_{m=1}^{M} \min(k_m, 2K-1) \geq (2K-1)(D+1), \tag{4.14}$$

where k_m is the number of measurements in which the m-th anchor is used in the set $\mathcal{M}_R$, and

$$\sum_{n=1}^{N} \min(j_n, D+1) \geq (2K-1)(D+1), \tag{4.15}$$

where j_n is the number of measurements at time t_n in the set $\mathcal{M}_R$. Then, for almost all sequences $(t_n)_1^N$, there is a unique matrix $\boldsymbol{C}$ satisfying Problem 4.3.

There are two sources of oversampling in Corollary 4.7. One is caused by using Theorem 3.3 that requires twice the number of degrees of freedom. This enables us to avoid combinatorial conditions. The second factor stems from treating matrices $\boldsymbol{\Gamma}_{\boldsymbol{R}}$ and $\boldsymbol{\Gamma}_{\boldsymbol{F}}$ jointly, as we will see in the proof. We could decrease the number of measurements by assigning measurements separately to $\boldsymbol{\Gamma}_R$ and to $\boldsymbol{\Gamma}_F$. But this again would lead to an impractical combinatorial condition ("if you can split the set $\mathcal{M}$ such that..").

Among all the situations where we measure multiple distances at one time, the case when anchors are unique is worth further investigation. If the anchors are fixed devices, then having many different ones might be too expensive. However, if the anchors are simply points along the trajectory of a moving device, they are likely to be all different.

Corollary 4.8: Moving Anchors

Assume that $|\mathcal{M}| \geq K(D+2)-1$, and that the anchors are unique and non-degenerate, i.e., none are used in more than one measurement. Let j_n be the number of measurements at time t_n. Given that

$$\sum_{n=1}^{N} \min(j_n, D+1) \geq K(D+1) \tag{4.16}$$

is satisfied, for almost all sequences of *anchors* $(\boldsymbol{r}_m)_1^M$ and almost all times $(t_n)_1^N$, there is a unique matrix $\boldsymbol{C}$ satisfying Problem 4.3.

Before we prove Corollaries 4.6 to 4.8, let us discuss some of their implications. Equations (4.13) and (4.14) describe how measurements cannot be arbitrarily distributed between anchors. In particular, if an anchor provides more than K measurements, only the first K have an effect on uniqueness; see the examples in Figure 4.6. Moreover, unique recovery is not possible with measurements from less than $D+1$ anchors. Note that assuming unique anchors in Corollary 4.6 gives Equation (4.13) for free.

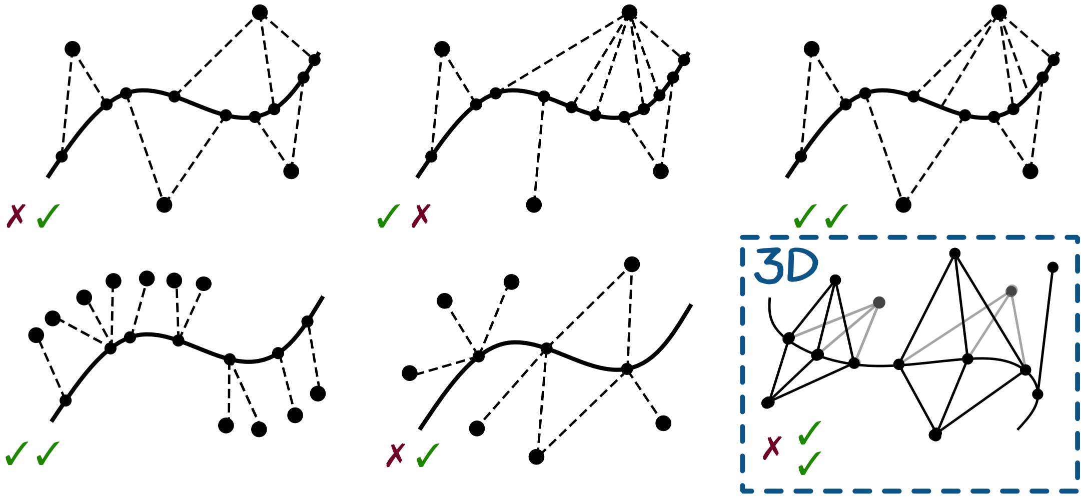


Figure 4.6.: Examples of sufficient and insufficient measurements, with model degree $K=3$ and embedding dimension $D=2$; i.e., $K(D+2)-1=11$ and $K(D+1)=9$. Top left: Equation (4.13) is satisfied (right tick), but there are not enough measurements in total (left cross). Top middle: there are enough measurements in total, but Equation (4.13) is not satisfied, because too many measurements involve the same anchor. Top right: both conditions are satisfied and recovery is guaranteed. Bottom left: according to Corollary 4.8 the recovery is possible with unique anchors. Bottom middle: recovery is possible via lateration, but not via linearisation. Bottom right: a 3D case. The recovery is not possible with $K(D+2)-1$ measurements, even though Equations (4.14) and (4.15) are satisfied.

As all the conditions are based on Theorems 3.1 and 3.3, all the analyses from Chapter 3 on how likely the conditions are to be satisfied apply in this case. In Figure 4.7, we can see that the probability of success heavily depends on the rank of matrix $\mathbf{\Gamma}_R$, whereas the extra $K-1$ measurements simply change the minimum number of measurements needed for success.

We can also observe in Figure 4.7 that, once we have $2(D+1)K$ measurements that satisfy the conditions from Theorem 3.3, we need only a handful of extra measurements to recover $\boldsymbol{C}$ from the linearised system. Experimentally, we also observed that, given Equations (4.14) and (4.15) and $2(D+1)K$ measurements, matrix $\mathbf{\Gamma}$ is full rank for the vast majority of time.

Now, we can prove the conditions. The following three proofs are very similar, but we keep them separate for clarity.

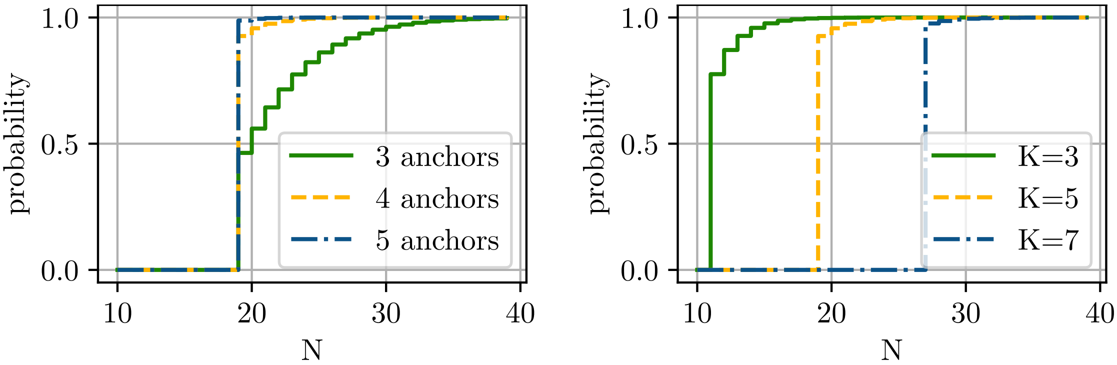


Figure 4.7.: The probability of recovery is zero if the number of measurements (N) is smaller than $(D+2)K-1$. After this, the probability of recovery is the probability that Equation (4.13) is satisfied. Left: parameters $D=2$, $K=5$ are fixed and the number of anchors changes. Right: number of anchors is fixed to 4, $D=2$ and the number of degrees of freedom K changes.

Proof of Corollary 4.6

We will show that under the conditions of Corollary 4.6 matrix $[\mathbf{\Gamma}_R\, \mathbf{\Gamma}_F]$ is full rank.

We can apply Theorem 3.1 to any $K(D+1)$ rows of $\mathbf{\Gamma}_R$, with $\boldsymbol{g}_n^\top = [\boldsymbol{r}_{m_n}^\top\, 1]$ and $J = D+1$, because the assumptions that anchors are non-degenerate translates to the assumptions that $\boldsymbol{g}_n$ are independent. Equation (4.13) is exactly the row condition Equation (3.11), hence $\mathbf{\Gamma}_R$ is full column rank.

We can assume that, without loss of generality, the first $K(D+1)$ rows of $\mathbf{\Gamma}_R$ are linearly independent. For any $r \in [K(D+1) \dots K(D+1)+K-1]$, let $\boldsymbol{A}_r$ be the top left $r \times r$ sub-matrix of $\begin{bmatrix}\mathbf{\Gamma}_R & \mathbf{\Gamma}_F\end{bmatrix}$. By Lemma 4.5, if $\boldsymbol{A}_r$ is full rank, then $\boldsymbol{A}_{r+1}$ is full rank for almost all t_r. Therefore, as $\boldsymbol{A}_{(D+1)K}$ is full rank for almost all $(t_n)_1^{(D+1)K}$, by mathematical induction, the matrix $\boldsymbol{A}_{K(D+2)-1}$ is full rank for almost all $(t_n)_1^{(D+2)K-1}$, thus the whole matrix $\begin{bmatrix}\mathbf{\Gamma}_R & \mathbf{\Gamma}_F\end{bmatrix}$ is full *column* rank for almost all $(t_n)_1^N$. ■

Proof of Corollary 4.7

In this proof, instead of expanding the matrix $\mathbf{\Gamma}_R$, we start from a larger matrix and prune it. Consider a matrix $\mathbf{\Gamma}_R^{(2K-1)} \in \mathbb{R}^{|\mathcal{M}\times(D+1)(2K-1)}$ that is the same as defined in Equation (4.11), except with K replaced by $2K-1$ and $\boldsymbol{f}_n$ replaced by $\boldsymbol{f}_n^{(2K-1)}$. From Theorem 3.3, if we have at least $2(D+1)(2K-1)$ measurements that satisfy Equations (4.14) and (4.15), then $\mathbf{\Gamma}_R^{(2K-1)}$ is full column rank and its columns are independent. However, $[\mathbf{\Gamma}_R\, \mathbf{\Gamma}_F]$ is a sub-matrix of $\mathbf{\Gamma}_R^{(2K-1)}$, hence its columns are also independent. ■

Proof of Corollary 4.8

The anchors in Corollary 4.8 are unique, hence we can use Theorem 3.1, this time identifying vectors $\boldsymbol{f}_n$ here as vectors $\boldsymbol{g}_n$ in Theorem 3.1, and vectors $[\boldsymbol{r}_m^\top \, 1]$ as vectors $\boldsymbol{f}_m$ in Theorem 3.1. If the anchors are non-degenerate, vectors $[\boldsymbol{r}_m^\top \, 1]$ are full spark. Vectors $\boldsymbol{f}_n$ are full spark for almost all times $(t_n)_1^N$ because they follow the polynomial model (Definition 3.2).

Therefore, matrix $\boldsymbol{\Gamma}_R$ is full rank for almost all sequences of *anchors*. Without loss of generality, assume that the first $(D+1)K$ rows $\boldsymbol{\Gamma}_R$ are independent. Then, the first $(D+1)K$ rows of $\boldsymbol{\Gamma}_R$ constitute our matrix $\boldsymbol{A}_{(D+1)K,(D+1)K}$ from Lemma 4.5, to which we will append columns and rows.

Consider adding row $r+1$. If the row we need to add contains a new vector $\boldsymbol{f}_n$ that has not been used in the first r rows, we apply Lemma 4.5 and the resulting matrix is full rank for almost all times t_n.

If the row we need to add contains a vector $\boldsymbol{f}_n$ that has appeared in the first r rows, then we need to use the fact that anchors are different. Assume that we are adding a new measurement that contains $\boldsymbol{f}_1$ that has been used before in the first row of $\boldsymbol{A}_{r,r}$. As in the proof of Lemma 4.5, as $\boldsymbol{A}_{r,r}$ is full-rank, there is exactly one vector $\boldsymbol{\beta}$ such that $\boldsymbol{A}_{r,r}\boldsymbol{\beta} = \boldsymbol{c}_{r+1}$, where $\boldsymbol{c}_{r+1}$ is the added column. Now, $\boldsymbol{c}_{r+1}$ depends only on times, not on anchors, and contains an entry $f_{r,1} = [\boldsymbol{f}_1]_r$ that is the same as the last entry in the row $\boldsymbol{p}_{r+1}$ we want to add. Therefore, we have that

$$[\boldsymbol{A}_{r,r}\boldsymbol{\beta}]_1 = \boldsymbol{\beta}^\top \boldsymbol{p}_{r+1}.$$

The row $\boldsymbol{p}_{r+1}$ depends linearly on $\boldsymbol{r}_m$, hence the above is a linear equation in $\boldsymbol{r}_m$ that is satisfied by only a measure-zero set of vectors $\boldsymbol{r}_m$. ■

Simulations

In this section, we introduce noise and perform simulations to evaluate Algorithm 4.1.

In the presence of noise, Equation (4.10) is no longer exact. We measure noisy estimates $\widetilde{d}_{m,n}$ of the distances $d_{m,n}$ for m, n in $\mathcal{M}$. In the single trajectory case, without acceleration measurements, the optimisation problem defined in Section 4.1 (Problem 4.2) becomes

$$\underset{C}{\arg\min} \sum_{(m,n)\in\mathcal{M}} \left(\widetilde{d}_{m,n} - \|\boldsymbol{r}_m - \boldsymbol{C}\boldsymbol{f}(t_n)\|^2 \right). \tag{4.17}$$

Algorithm 4.1 minimises a relaxed version of this problem:

$$\underset{C}{\arg\min} \sum_{(m,n)\in\mathcal{M}} \left(\widetilde{d}_{m,n} - \|\boldsymbol{r}_m\|^2 - 2\boldsymbol{r}_m^\top \boldsymbol{C}\boldsymbol{f}(t_n) + \boldsymbol{f}(t_n)^\top \boldsymbol{L}\boldsymbol{f}(t_n) \right). \tag{4.18}$$

The estimated $\widetilde{\boldsymbol{C}}$ is no longer equal to $\boldsymbol{C}$. However, the conditions for the invertibility of the linear system are still valid because, for known anchors, the matrix that is inverted does not depend on measured distances. This means that Corollaries 4.6 to 4.8 give us conditions for when there exists a unique minimiser to Equation (4.18).

We also note here that Algorithm 4.1 might suffer, not only from the presence of noise but also from ill-conditioning, when anchors are too close or the measurement times are too close. This is the reason uniform samples might be better than random samples and anchors should be spread out.

As mentioned before, we assume additive Gaussian noise on distance measurements $\sqrt{\widetilde{d}_n} = \sqrt{d_n} + \epsilon_n$, where ϵ_n are i.i.d. random variables, $\epsilon_n \sim \mathcal{N}(0, \sigma)$. An immediate problem comes from the fact that, in Problem 4.2, we fit the parameters to the *squared* distances. The distribution of *squared* distances is both additive and multiplicative:

$$\widetilde{d}_n = d_n + 2\sqrt{d_n}\epsilon_n + \epsilon_n^2,$$

whereas our recovery method implicitly assumes only centred additive noise. Indeed, in the proposed algorithm, we solve a linear system of equations, and we obtain a solution that would be a MLE if the Gaussian noise were added to the *squared* distances.

To alleviate this problem, we propose a weighted least-squares (WLS) approach. If the noise is small, the ϵ_n^2 term is negligible. The remaining noise has distribution $2d_n\epsilon_n \sim \mathcal{N}(0, 2d_n\sigma)$. If we know d_n, we could use WLS, with weights $1/d_n$, to bring the system back to an i.i.d. noise model. But as we do not know the distances, we use the *measured* distances $\widetilde{d}_n$ and obtain a noisy version of a row of Equation (4.12):

$$\sqrt{\frac{d_n}{\widetilde{d}_n}}\epsilon_n + \frac{1}{\sqrt{\widetilde{d}_n}}\begin{bmatrix}\boldsymbol{\Gamma}_R & \boldsymbol{\Gamma}_F\end{bmatrix}_n \begin{bmatrix}\operatorname{vec}(\boldsymbol{C}) \\ -\frac{1}{2}\boldsymbol{\Phi}\operatorname{vec}(\boldsymbol{L})\end{bmatrix} = \frac{\widetilde{d}_{m,n} - \|\boldsymbol{r}_m\|^2}{\sqrt{\widetilde{d}_n}}.$$

Again, assuming that the noise is small, $d_n/\widetilde{d}_n$ is close to 1, and the noise is approximately i.i.d. Gaussian. In practice, to avoid dividing by extremely small numbers, we can add some small regularisation γ to the distance and divide by $\widetilde{d}_n + \gamma$.

We report the root squared error E between the estimate $\widetilde{\boldsymbol{C}}$ and the ground truth $\boldsymbol{C}$:

$$E(\boldsymbol{C}, \widetilde{\boldsymbol{C}}) = \|\boldsymbol{C} - \widetilde{\boldsymbol{C}}\|_F, \tag{4.19}$$

where $\|\cdot\|_F$ is the Frobenius norm. This norm is roughly equivalent to the power of the signal: using power as opposed to energy makes errors comparable between trajectories with different periods. For more intuition, see Figure 4.8.

In the simulations, we take samples t_n uniformly in the interval $[0, T]$ and, at each time, we choose an anchor uniformly at random. To simulate different values for N, we discard some measurements uniformly at random. We fix $T = 2$. Figure 4.9 shows the reconstruction error of coefficients obtained using the weighted reconstruction. We can see that the error decreases with oversampling, and the fitted slope is roughly -0.6. This means that, for 10× oversampling, we get more than a 5× reconstruction improvement. The regular (non-weighted) solver performed similarly, with a smaller improvement from oversampling.

Real-World Experiments

We test our trajectory estimation algorithm on two real-world datasets provided by Djugash *et al.* [265]. The datasets consist of an autonomous lawnmower moving on a

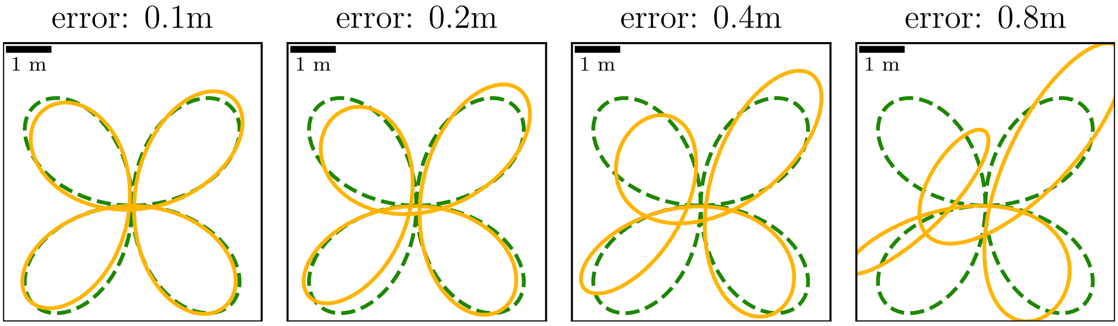


Figure 4.8.: Visualisation of the distance between trajectories. Original bandlimited trajectory of order $K = 7$ (dashed blue) and a randomly perturbed trajectory (solid orange). The Frobenius distance between the trajectory coefficients is displayed above each subplot.

field of grass, using ultra-wideband (UWB) signals to four stationary anchors for range measurements, and densely sampled kinematic GPS for ground truth. The distance measurements have an average standard deviation of ca. 0.5 m, with a tendency to overestimate [265].

We first evaluate the *Plaza2* dataset. The trajectory completed by the robot does not perfectly fit our models, but we will see that it can be approximated by the bandlimited model. We can estimate its period T by visual inspection. Using all 499 range measurements, we use Algorithm 4.1 to estimate the coefficients for different complexities K, and report the obtained trajectories in Figure 4.10 (a). We see that the trajectory is well approximated with degrees of $K = 5$ or higher.

Next, we fix K to the smallest sensible value for the given trajectory ($K = 5$) and test the performance of our algorithm when dropping measurements uniformly at random, down to the minimum number required (19 measurements). Figure 4.10 (b) shows that the obtained reconstruction quality remains satisfactory, down to as few as 30 measurements, and it is not too sensitive to the specific distance measurements selected. As seen already in simulation, the variance of the reconstructions is higher for fewer measurements.

For comparison, we also plot localization results with the point-wise RLS method [145], using the latest distance measurements from $D + 1$ anchors. The method uses brute force on a uniform grid initialized in the bounding box of the anchors with a grid size of 0.5 m. The trajectory starts to be recognizable only from $N = 200$ measurements upwards, and the individual estimates are more noisy than ours.

Finally, we compare the provided algorithms, quantitatively with standard lateration methods and solvers. Tables 4.1 and 4.2 shows the trajectory recovery accuracy for the two UWB-based datasets from [265]. As we have seen before, the trajectory in *Plaza2* is approximately bandlimited. In contrast, trajectory in *Plaza1* covers the lawn in a zigzag manner. In practice, for such a trajectory, an iterative application of our algorithm would be appropriate. Here, we simply report the average reconstruction accuracy over 20 different linear parts of the trajectory. We compute the error by sampling the parametric

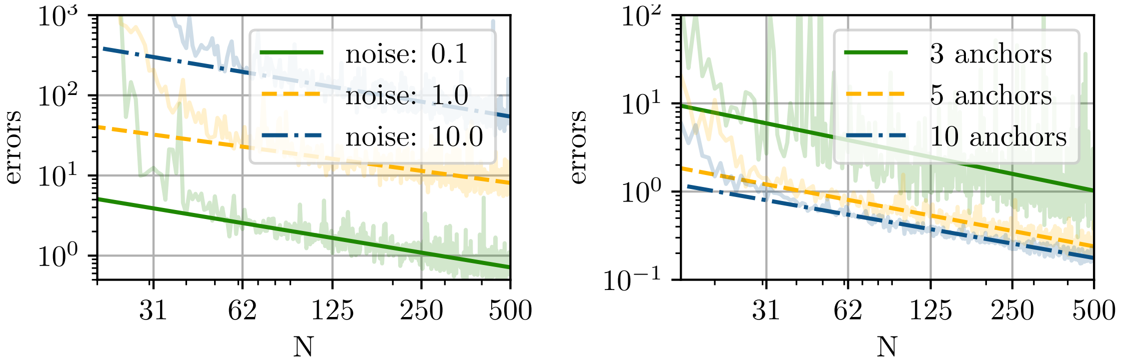


Figure 4.9.: Reconstruction from noisy measurements using the weighted system of equations. For clarity, slopes (dark lines) were fitted to the averages over 1000 simulations (light lines). The simulated distances were between 0.1 m and 10 m. Left: the trajectory degree was set to $K = 5$; $M = 4$ anchors were used; and the magnitude of the noise changes. In this setup, the minimum number of measurements required is 19. We can see that, starting from about 3× oversampling, the algorithm is robust to noise. Right: the trajectory degree was set to $K = 3$ and the noise magnitude was set to $\sigma = 1$ m; the number of anchors changes. The reconstruction error does not depend heavily on the number of anchors, but it has a much higher variance for $D + 1 = 3$ anchors.

model at the times at which we have ground-truth measurements; we compute the mean squared error (MSE) between the predicted and ground truth points; and we average these errors over 20 different realizations.

We evaluate the proposed solution with and without the weighting introduced at the beginning of this section. We compare our method with lateration, which finds each point independently from the other points, with a parametric trajectory fitted to these points (*trajectory from lat.*), and with direct optimisation of the original, non-squared problem by using the `scipy.optimize` [266] implementation of Levenberg-Marquardt (LM) algorithm [147]. We compare two different initializations: a simple ellipse / line in the correct order of magnitude (*LM ellipse/line*) and the result of our weighted algorithm as initialization (*LM ours weighted*). The row *model mismatch* shows the error of the model fitted directly to the ground-truth position measurements and gives an upper bound on how well we can hope to recover the positions.

We can draw a number of interesting conclusions from the quantitative evaluation. First of all, we note that weighting distance measurements for real data improves the reconstruction accuracy significantly, especially for the bandlimited trajectory (Table 4.1). We believe that this is due to a positive bias in the distance measurements; indeed, if the bias is removed by an oracle, the weighted and non-weighted algorithms perform almost identically.

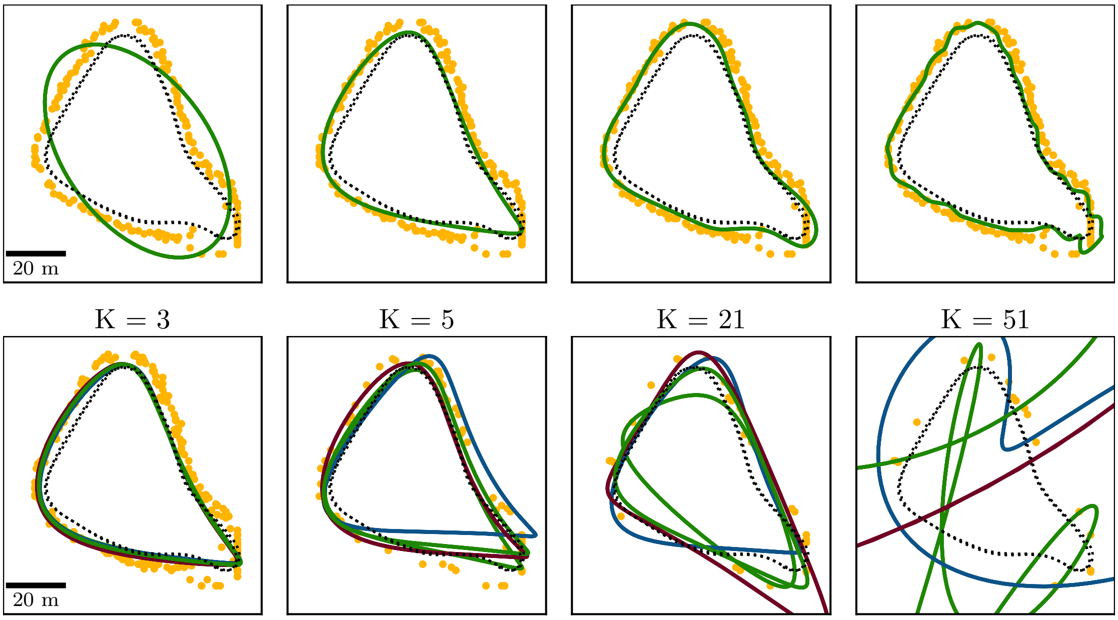
We further observe that the reconstruction behaves poorly when the number of measurements is close to the limit given in Corollary 4.6. For the bandlimited trajectory of degree $K = 19$ this limit is 75 measurements, and $N = 100$ noisy measurements are too

Table 4.1.: MSE of recovered bandlimited trajectories for a real-world UWB-based localization dataset (*Plaza2*) [265].

# measurements N	**100**			**300**			**499**		
model complexity K	**5**	**11**	**19**	**5**	**11**	**19**	**5**	**11**	**19**
model mismatch	3.7	1.8	1.5	3.8	2.0	1.7	3.8	2.1	1.8
lateration	113.6	134.3	106.1	16.5	17.2	16.2	9.7	9.7	9.7
trajectory from lat.	62.1	91.5	5971.6	13.2	12.8	12.9	10.5	9.2	9.4
LM ellipse	12.7	147.6	339.8	11.3	25.9	13.2	11.4	11.5	12.0
LM ours weighted	12.7	13.0	20.5	11.3	11.9	12.2	11.4	11.5	12.0
ours	13.9	14.1	69.5	13.2	11.5	11.6	13.2	11.4	11.3
ours weighted	10.3	10.6	68.2	8.8	7.3	7.3	8.7	7.2	6.9

Table 4.2.: MSE of recovered polynomial trajectories a real-world UWB-based localization dataset (*Plaza1*) [265].

# measurements N	**10**	**20**	**30**	**50**
model complexity K	**2**	**2**	**2**	**2**
model mismatch	0.3	0.3	0.4	0.7
lateration	32.5	14.1	12.7	6.3
trajectory from lat.	48.5	198.8	12.9	6.7
LM line	253.4	24.6	16.3	6.6
LM ours weighted	18.7	7.2	7.5	6.6
ours	51.7	4.9	4.8	4.1
ours weighted	5639.1	4.9	4.8	4.3



K = 3 K = 5 K = 21 K = 51

N = 200 N = 60 N = 30 N = 19

Figure 4.10.: Reconstruction accuracy of lawnmower trajectory using period $T = 54\,\mathrm{s}$. In the black dots is the ground truth trajectory (from GPS), in solid lines are our reconstructions, and the orange dots show the RLS estimates. The top row shows different complexities K that use all available distance measurements. In the bottom row, we fix K to 5 and drop measurements, uniformly at random. The different colours correspond to different sets of measurements.

few to accurately estimate the trajectory. For the polynomial trajectories, we observe the same behaviour for $N = 10$ and $K = 2$, where $N = 10$ is close to the minimum of 7.

Finally, we note that the accuracy of the LM methods depends highly on the initialization, which is a well-known limitation [267]. There is a significant difference between the errors for an initialization with an ellipse or a line and an initialization with result of the weighted version of our algorithm. This difference is particularly pronounced for high-model orders and few measurements. Quite surprisingly, even though LM optimizes the non-squared cost function hence aims to recover the maximum likelihood estimate of $\boldsymbol{C}$ for zero-mean Gaussian noise on measurements, it does not compare favourably with our proposed algorithm. We believe that both the fact that the solution is only sub-optimal and the bias in the distance measurements plays against this method. However, we suggest that if LM is the preferred solution in a complete system for different reasons, our proposed method is an attractive candidate for an accurate initialization.

4.2.2. Trajectory Self-Localisation

In this section, we still consider a single trajectory, but we do not have anchors. We assume we measure distances between a series of positions of the robot. This could be the case when the robot moves faster than the speed of sound but, more realistically, it could be the case when we want to estimate a path along which multiple devices are moving, or if the trajectory is a physical object with forces acting on it. For example, it could be a wire with known distances between points or a molecule with distances estimated from nuclear magnetic resonance (NMR).

The squared distance between a point on a trajectory at time t and a point at time $t + \tau$ is

$$d(t, \tau) = \|\boldsymbol{C}\boldsymbol{f}(t+\tau) - \boldsymbol{C}\boldsymbol{f}(t)\|^2 = \|\boldsymbol{C}\left(\boldsymbol{f}(t+\tau) - \boldsymbol{f}(t)\right)\|^2. \tag{4.20}$$

We measure distances at known times t_n and we know the delays τ_n. Here the delays might mean the delay between the devices moving on the same trajectory, or the distance between atoms in the molecule but, for consistency, we continue to call them delays. This leads to the following problem:

Problem 4.4: Trajectory Self-Localisation

Consider an unknown, continuous trajectory $\boldsymbol{r}(t) = \boldsymbol{C}\boldsymbol{f}(t)$, where $\boldsymbol{f}$ is a vector of basis functions of family $\mathcal{F}$ of either K-bandlimited functions or polynomials of degree of at most $K-1$. We measure distances between robot positions at known times t_n and $t_n + \tau_n$, $n \in [N]$:

$$d_n = \|\boldsymbol{C}\boldsymbol{f}(t_n+\tau_n) - \boldsymbol{C}\boldsymbol{f}(t_n)\|^2 = \|\boldsymbol{C}\left(\boldsymbol{f}(t_n+\tau_n) - \boldsymbol{f}(t_n)\right)\|^2. \tag{4.21}$$

The problem of a trajectory self-localisation is to recover trajectory coefficients $\boldsymbol{C}$ up to an equivalence described in Definition 4.1.

To linearise the problem, we replace $\boldsymbol{L} = \boldsymbol{C}^\top\boldsymbol{C}$. We then write Equation (4.20) in the standard form of Equation (4.12):

$$d(t, \tau) = \operatorname{vec}\left((\boldsymbol{f}(t+\tau) - \boldsymbol{f}(t))(\boldsymbol{f}(t+\tau) - \boldsymbol{f}(t))^\top\right)^\top \operatorname{vec}(\boldsymbol{L}).$$

This describes a rank-one measurement. Unfortunately, we cannot apply the methods from Chapter 3 directly, because both vectors in the product depend on the same parameters t and τ. And we have seen in Section 3.1.2 that this might lead to an underdetermined system of equations. To avoid such situations, we need to assume that the delays vary between measurements. If not, then $\boldsymbol{f}(t_n + \tau) - \boldsymbol{f}(t_n)$ is simply a polynomial (or bandlimited) function in one variable, and from Lemma 4.5 we know that the rank of $\boldsymbol{\Gamma}$ can be at most $2K - 1$. If both variables t and τ can vary, we have the following result:

Corollary 4.9

Assume that we measure $N \geq (K-1)K/2$ distances, as in Problem 4.4. Then, for almost all times $(t_n)_1^N$ and almost all time differences $(\tau_n)_1^N$ the matrix $\boldsymbol{C}$ is unique up to rotations, reflections and translations.

Proof of Corollary 4.9

First observe that we cannot recover the constant component of the trajectory $\boldsymbol{r}$. Furthermore, if we remove the constant component of $\boldsymbol{C}$ the measurements are the same. Therefore, we recover instead the matrix $\boldsymbol{C}^{(r)} \in \mathbb{R}^{D\times(K-1)}$ created by removing the first column from $\boldsymbol{C}$. From Equation (4.21), we recover the symmetric matrix $\boldsymbol{L}_r = \boldsymbol{C}^{(r)\top}\boldsymbol{C}^{(r)} \in \mathbb{R}^{(K-1)\times(K-1)}$ that has $(K-1)K/2$ degrees of freedom.

The measurement matrices are symmetric:

$$(\boldsymbol{f}(s) - \boldsymbol{f}(t))(\boldsymbol{f}(s) - \boldsymbol{f}(t))^\top,$$

thus they can have at most $(K-1)K/2$ degrees of freedom. In the above equations, we changed variables $s := t + \tau$ for simplicity. Hence, we need to show that functions

$$h_{k,l}(s,t) := (f_k(s) - f_k(t))(f_l(s) - f_l(t)) \text{ for } 2 \leq l \leq k \leq K, \tag{4.22}$$

are linearly independent.

In the polynomial case, Equation (4.22) describes a homogeneous polynomial of degree $k+l$. A homogeneous polynomial $h_{k,l}$ is linearly independent from all polynomials of smaller degree, thus from all combinations of $h_{k',l'}$ with sum $k'+l'$ smaller than $k+l$. As a result, we need to show only the independence of $h_{k,l}$ from all other $h_{k',l'}$, such that $n := k+l = k'+l'$. But if $k_1 \neq k_2$ and $k_1 \neq n - k_2$, the terms

$$\begin{aligned} h_{k,n-k} &= t^n + s^n - (t^k s^{n-k} + s^k t^{n-k}) \\ h_{k',n-k'} &= t^n + s^n - (t^{k'} s^{n-k'} + s^{k'} t^{n-k'}) \end{aligned}$$

are linearly independent, because their right-hand sides (in the brackets) are different elements of the basis of the space of homogeneous polynomials in two variables.

In the bandlimited case, we can apply the same reasoning. Except now, instead of considering the sum of degrees, we need to consider sum of frequencies. For all components of $\boldsymbol{h}_{k,l}$ this sum is constant. But, because we have sines and cosines it is no longer equal to $k+l$, rather about half of $k+l$. As in the polynomial case, h with sum of frequencies n is independent of all functions with a smaller sum of frequencies.

For a fixed sum of frequencies, functions with different phases in the highest frequency are also independent; that is to say, $\cos(nt)$ is independent from $\sin(nt)$ and $\cos(kt)\sin((n-k)t)$ for all $k < n$.

The last step is the same as in the polynomial case. ■

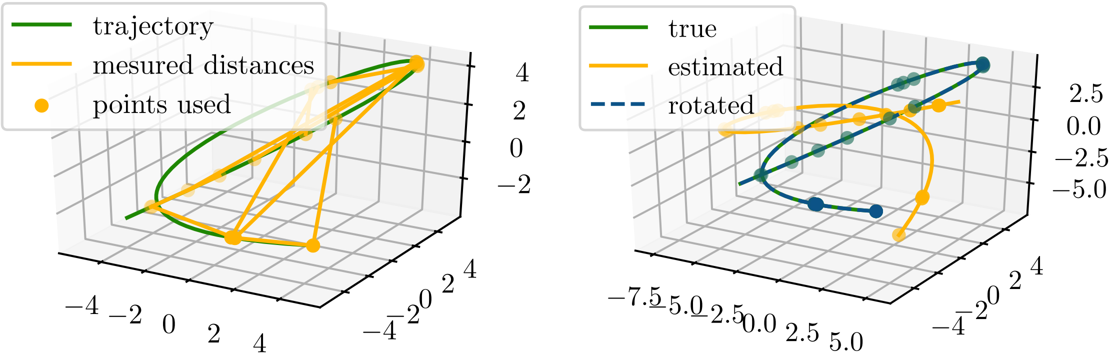


Figure 4.11.: An example of working self-localisation. No noise was added to the simulation. Left: the original trajectory with marked the measured distances. There might be many distances measured from one point. Right: the trajectory can be reconstructed up to unitary operations; after the rotation to the right frame the trajectory matches the reference perfectly.

In Figure 4.11, we show an example of recovery and in Figure 4.12 the aggregated results of simulations.

4.2.3. General Case

In this section, we revisit the general problem introduced in Section 4.1 (Problem 4.1). We provide a sufficient condition for recovery in the ideal case. However, as the method we use does not generalise well to the noisy case, we propose to use semidefinite relaxation or gradient descent and to show that they indeed work when our condition is satisfied.

We state our result first and then describe steps needed to arrive at it.

Theorem 4.10: Sufficient condition for Problem 4.1

Assume that we measure distances between M trajectories, as given in Problem 4.1. Consider an undirected graph $\mathcal{G}$ with M nodes, each representing a trajectory, with edges between the nodes if there are $(K-1)(K+3)+1$ measurements between the corresponding trajectories. If the graph is connected and there is a 3-node clique[a] in $\boldsymbol{G}$, then all trajectories can be uniquely recovered up to ambiguities described in Definition 4.1.

---

[a] A clique in a graph is a subset of nodes that are all connected to each other.

From Section 4.1.1, we know that we cannot recover the global translation of the trajectories. This means that we cannot recover a constant component of any of the trajectories. We can, however, recover the constant component of the difference between trajectories. To highlight this, we separate the constant from $\boldsymbol{C}$. More formally, we write the distance equation as

$$d_{m_1,m_2}(t,s) = \|\boldsymbol{v}_{m_1,m_2} + \boldsymbol{C}^{(r)}_{m_2}\boldsymbol{f}^{(r)}(s) - \boldsymbol{C}^{(r)}_{m_1}\boldsymbol{f}^{(r)}(t)\|^2,$$

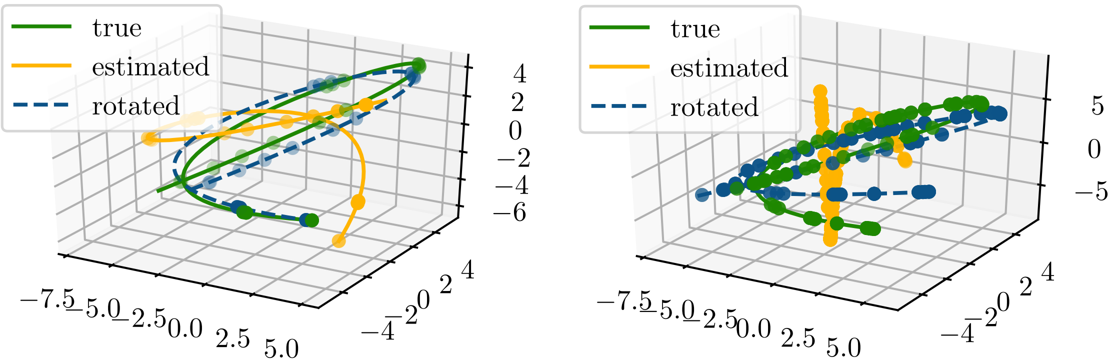


Figure 4.12.: The recovery is sensitive to the structure of the distance measurements. Left: the same $N = 10$ measurements as in Figure 4.11, but with noise added with $\sigma = 1$, the recovery is not perfect due to noise, but it is close to the ground truth. Right: Large number $N = 155$ short distances used (shorter than 2 units). With the same noise variance $\sigma = 1$, the recovery is less table.

where and $s = t + \tau$, $\boldsymbol{C}_m^{(r)} \in \mathbb{R}^{D\times(K-1)}$ are matrices $\boldsymbol{C}$ without the first column, $\boldsymbol{f}_n^{(r)}$ are vectors $\boldsymbol{f}_n$ without the first (constant) element, and $\boldsymbol{v}_{m_1,m_2}$ is the difference between constant components of trajectories m_1 and m_2.

We analyse this equation by using the same methods as in the previous section, and we obtain the following result for the reduced matrices $\boldsymbol{C}_m^{(r)}$:

Lemma 4.11: Two trajectories

Assume we have two trajectories $\boldsymbol{r}_1 = \boldsymbol{C}_1\boldsymbol{f}$ and $\boldsymbol{r}_2 = \boldsymbol{C}_2\boldsymbol{f}$ and measure $N \geq (K-1)(K+3)+1$ distances as described in Problem 4.1. Then we can recover products $\boldsymbol{C}_1^{(r)\top}\boldsymbol{C}_2^{(r)}$, uniquely for almost all times t_n and time delays τ_n.

We prove Lemma 4.11 at the end of this section, but first we discuss its implications. This result itself is not sufficient to recover the coefficients $\boldsymbol{C}_m^{(r)}$, because the factorisation of $\boldsymbol{C}_1^{(r)\top}\boldsymbol{C}_2^{(r)}$ is not unique (not even up to unitary transformations), see Example 4.7. Fortunately, if we have three trajectories, we can jointly factorise products $\boldsymbol{C}_1^{(r)\top}\boldsymbol{C}_2^{(r)}$, $\boldsymbol{C}_2^{(r)\top}\boldsymbol{C}_3^{(r)}$ and $\boldsymbol{C}_3^{(r)\top}\boldsymbol{C}_1^{(r)}$ as summarised in Lemma 4.12.

Example 4.7: Non-Unique Factorisation

Consider any invertible $K \times K$ matrix $\boldsymbol{A}$. Let $\boldsymbol{B}^\top = \boldsymbol{A}^{-1}$. Then let $\boldsymbol{B}^\top\boldsymbol{A} = \boldsymbol{Id}$. But as matrix $\boldsymbol{A}$ is arbitrary, it does not have to be unitary or diagonal, only invertible. Therefore, the factorisation is not unique.

Lemma 4.12: Collective Matrix Decomposition

Let $\boldsymbol{A}, \boldsymbol{B}$ and $\boldsymbol{C}$ be unknown matrices $D \times K$ of rank D. Assume that we know the products:

$$\boldsymbol{L} = \boldsymbol{A}^\top \boldsymbol{B},\ \boldsymbol{M} = \boldsymbol{B}^\top \boldsymbol{C} \text{ and } \boldsymbol{N} = \boldsymbol{C}^\top \boldsymbol{A}.$$

Then matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are unique up to a common unitary transformation $\boldsymbol{U}$.

These two lemmas enable us to prove Theorem 4.10.

Proof

If there is a 3-vertex in the graph $\mathcal{G}$, then per Lemma 4.11, there are three pairs of trajectories for which we can recover products $\boldsymbol{C}_{m_2}^{(r)\top} \boldsymbol{C}_{m_1}^{(r)}$. Then, from Lemma 4.12 we can factorise these products and obtain matrices $\boldsymbol{C}_m^r$, up to unitary transformations. We can then fix the orientation in space, i.e., the unitary transformation, and recover $\boldsymbol{v}_{m_1,m_2}$ from $D+1$ the distance equations between trajectories m_1 and m_2. Therefore, we can recover all three matrices $\boldsymbol{C}_m$ up to constant translations and constant unitary transformations.

To recover any other trajectories, we need K^2 measurements between this trajectory and one of the already known trajectories. ■

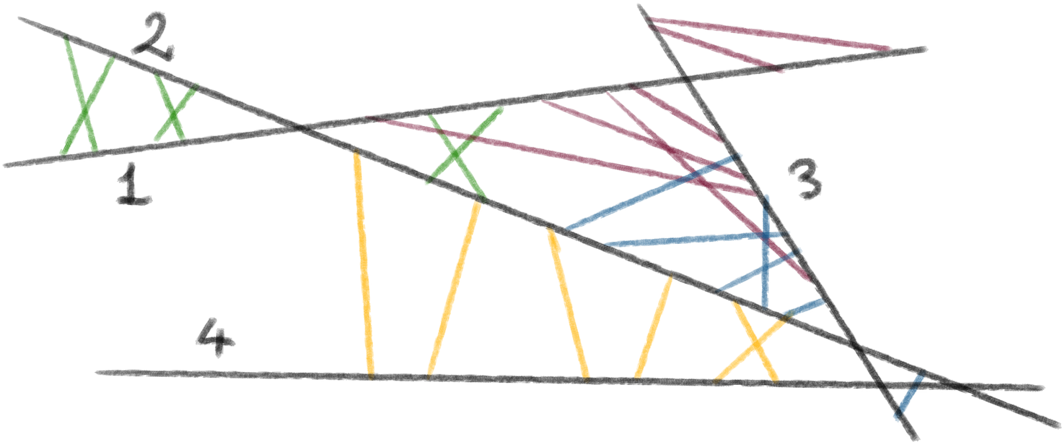


Figure 4.13.: Example of trajectories recoverable with our method. The upper three create a clique that can be recovered using our factorisation method, and the lower trajectory can be localised by its measurements to one of the three.

This method is, of course, not very practical. First, we need $O(K^2M)$ measurements, where ideally we would need $O(KDM)$. Then, for joint matrix factorisation to work as above, the measurements need to be noiseless. In the noisy case, we could use joint singular value decomposition [268] that is a generalisation of joint matrix denationalisation

[269] to non-symmetric or rectangular matrices. However, to do this we need to solve a nonlinear optimisation problem and the gain from the linearisation is lost. Nevertheless, this method gives us a threshold above which we would expect any respectable method to work, see Figure 4.14.

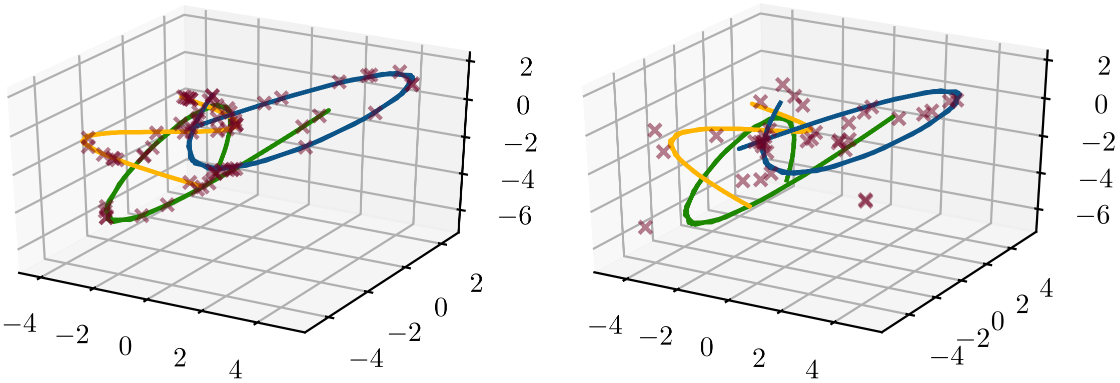


Figure 4.14.: Results of gradient descent optimisation with small amount of noise ($\sigma = 10^{-3}$). Continuous lines denote ground truth and the crosses recovered trajectories; the points are the end points of the measured distances. Left: given $N = (K-1)(K+3)+1$ measurements per trajectory (as given by Theorem 4.10), the recovery error is small. Right: with $N = DK$ measurements per trajectory (the number of degrees of freedom), the recovery error is noticeable.

To conclude this section, we provide the proofs of Lemmas 4.11 and 4.12.

Proof of Lemma 4.11

In this proof, we drop the superscript for simplicity. Using the same algebraic tricks as before, we can expand all the terms. Although it might look discouraging, the equation below enables us to analyse different terms separately:

$$
\begin{align}
d_{m_1,m_2,n} =& \langle 1, \|\boldsymbol{v}_{m_1,m_2}\|^2 \rangle && \text{constant in } s,t \notag \\
&+2\langle \boldsymbol{f}(t)\boldsymbol{f}^\top(s), \boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_2} \rangle && \text{degree 1 to } K-1 \text{ in } s,t \tag{4.23} \\
&-\langle \boldsymbol{f}^\top(t), \boldsymbol{v}_{m_1,m_2}^\top \boldsymbol{C}_{m_2} \rangle && \text{degree 1 to } K-1 \text{ in } t \tag{4.24} \\
&+\langle \boldsymbol{f}(t)\boldsymbol{f}^\top(t), \boldsymbol{C}_{m_1}^\top \boldsymbol{C}_{m_1} \rangle && \text{degree 2 to } 2K-2 \text{ in } t \tag{4.25} \\
&+\langle \boldsymbol{f}^\top(s), \boldsymbol{v}_{m_1,m_2}^\top \boldsymbol{C}_{m_1} \rangle && \text{degree 1 to } K-1 \text{ in } s \tag{4.26} \\
&+\langle \boldsymbol{f}(s)\boldsymbol{f}^\top(s), \boldsymbol{C}_{m_2}^\top \boldsymbol{C}_{m_2} \rangle && \text{degree 2 to } 2K-2 \text{ in } s, \tag{4.27}
\end{align}
$$

We can observe that Line (4.23), as a function of s and t, is linearly independent of all other lines. Hence, if we have enough measurements for the system to be maximal possible rank, we can recover product $\boldsymbol{C}_{m_1}^{(r)\top} \boldsymbol{C}_{m_2}^{(r)}$ and some combination of the remaining terms. Lines (4.24) and (4.25) together span the space of polynomial of degrees from 1 to $2K-2$ and, similarly, Lines (4.26) and (4.27) together span the space of polynomial of degrees from 1 to $2K-2$, and the maximal rank of the system is $1+(K-1)^2+(2K-2)+(2K-2)=(K-1)(K+3)+1$ measurements. ■

Proof of Lemma 4.12

First, we consider full-rank matrices and then show how to extend them to the general case. We have the following equations:

$$\boldsymbol{L}=\boldsymbol{A}^\top\boldsymbol{B},\ \boldsymbol{M}=\boldsymbol{B}^\top\boldsymbol{C} \text{ and } \boldsymbol{N}=\boldsymbol{C}^\top\boldsymbol{A}. \tag{4.28}$$

By assumption, $\boldsymbol{A}$ is invertible thus $\boldsymbol{B}=\boldsymbol{A}^{-1\top}\boldsymbol{L}$. Plugging it in to the second equation, $\boldsymbol{M}=\boldsymbol{L}^\top\boldsymbol{A}^{-1}\boldsymbol{C}$. At the same time, $\boldsymbol{C}=\boldsymbol{A}^{-1\top}\boldsymbol{N}$, hence we obtain $\boldsymbol{M}=\boldsymbol{L}^\top\boldsymbol{A}^{-1}\boldsymbol{A}^{-1\top}\boldsymbol{N}$ and consequently

$$\boldsymbol{A}^\top\boldsymbol{A}=\boldsymbol{N}\boldsymbol{M}^{-1\top}\boldsymbol{L}.$$

The factorisation of $\boldsymbol{A}^\top\boldsymbol{A}$ is unique up to unitary transformation. From $\boldsymbol{A}$ and Equation (4.28), we can calculate $\boldsymbol{B}$ and $\boldsymbol{C}$.

If we want to solve the general case, we need to first apply SVD (see Section 1.5) to Equation (4.28). Up to a permutation of singular values and scaling, the right singular vectors of $\boldsymbol{L}$ will be the same as the right singular values of $\boldsymbol{B}$ and as the left singular vectors of $\boldsymbol{M}$; and similarly for $\boldsymbol{A}$ and $\boldsymbol{C}$. Therefore, we can calculate the permutations and scaling and can reduce the low rank matrices to the full rank case. ■

4.3. Future Work

In this chapter, we have developed a framework that can be used to provide guarantees for localisation from range measurements, and we hope that more results of this flavour can be proved in the future. In Box 4.8 we show an example how in theory we could supplement range measurements with an inertial measurement unit.

So far, we have given little thought to the practicality of the proposed methods. Although Algorithm 4.1 is preforming well in practice, methods from Section 4.2.3 were developed purely for the purposes of proofs. There are several directions we could take to improve this aspect of our work.

The first direction, which we have hinted at, is to find more suitable optimisation algorithms for the general case. This could include collective (or joint) matrix decomposition, semi-definite programming or gradient descent. For real-time optimisation iterative algorithms can be designed to integrate new measurements on the go and unroll only as many iterations as possible at a given time.

The second direction is to create more realistic models of the trajectory: for example, piece-wise smooth trajectories (spline trajectories) or trajectories with an unknown number of parameters K.

A quick check is needed to see that spline trajectories cannot be plugged into our setup, without having the number of samples grow linearly or even quadratically with the number of spline nodes. A better approach would be to estimate trajectories consecutively over intervals in order to use use our method without modification. An interesting question is, How can we estimate the trajectory if spline nodes are not known? We think that the Kalman-filter or particle-filter approaches could be used, but instead of estimating the position directly, we could estimate $\boldsymbol{C}$. Such approaches could detect a spline node when the likelihood of the new distance measurement, given the current estimate of $\boldsymbol{C}$, is too small.

Finally, on the more theoretical side, we provide only sufficient, but not necessary conditions for trajectory recovery. It would be interesting to see if the results from rigidity theory can be extended to smoothly changing trajectories.

Want to know more? 4.8: Integration with an Inertial Measurement Unit

In this section, we discuss how our results could be extended to incorporate different kinds of measurements. We show an example of supplying range measurement with measurements from an inertial measurement unit (IMU). We provide sketches of corollaries that could be obtained. This section might be a good starting point for learning how to extend our results to new cases.

A IMU usually consists of accelerometers, gyroscopes, and a magnetometer. If mounted on a device, it can provide many inexpensive measurements. The errors from IMU accumulate, hence it is a common practice to use both range measurements and IMU measurements. There are many ways in which IMU measurements can be combined but, to avoid integration errors, we focus here on integrating measurements as close to raw as possible.

The main challenge in combining IMU measurements with our method is that all the IMU measurements are in the internal frame of reference that is unknown. Gyroscopes measure orientation and rotation up to an unknown global rotation. Furthermore, the rotation does not fit our model well, because it is not linear in the trajectory parameters $\boldsymbol{C}$. Magnetometers provide the north direction, which could help align the internal frame of reference with the global frame of reference, at least when the robot has a fixed up-down direction.

The simplest measurement type for us to integrate is acceleration. As they measure force in the internal frame, not the acceleration, they also measure gravity:

$$\boldsymbol{a}(t) = \boldsymbol{U}(t)(\boldsymbol{C}\ddot{\boldsymbol{f}}(t) + \boldsymbol{g}),$$

where $\boldsymbol{g}$ is the gravity acceleration, and $\boldsymbol{U}(t)$ is an unitary matrix that rotates from global coordinate system to the internal coordinates of the device at time t; and it is unknown. One inefficient way to deal with this ambiguity is to include in the equations only the magnitude of the acceleration:

$$\begin{aligned}\|\boldsymbol{a}(t)\|^2 &= \|\boldsymbol{g} + \ddot{\boldsymbol{r}}_m(t_n)\|^2 \\ &= \|\boldsymbol{g}\|^2 + \boldsymbol{g}^\top \boldsymbol{C}\ddot{\boldsymbol{f}}(t) + \ddot{\boldsymbol{f}}^\top(t)\boldsymbol{C}^\top \boldsymbol{C}\ddot{\boldsymbol{f}}(t).\end{aligned}$$

Using the same operations as in the previous sections, we could transform them to the linear form:

$$a_n^2 - g^2 = \begin{bmatrix} \mathrm{vec}(\boldsymbol{g}\ddot{\boldsymbol{f}}_n^\top) & \mathrm{vec}(\ddot{\boldsymbol{f}}_n\ddot{\boldsymbol{f}}_n^\top) \end{bmatrix} \begin{bmatrix} \mathrm{vec}(\boldsymbol{C}) \\ \mathrm{vec}(\boldsymbol{L}) \end{bmatrix}.$$

We can immediately see that acceleration can produce only up to $K-1$ (or even only up to $K-2$ in the polynomial case) independent measurements of matrix $\boldsymbol{C}$. They alone are insufficient for recovering $\boldsymbol{C}$, but they can replace or supplement distance measurements, hence making the recovery algorithm work with close to $(D+1)K$ distance measurements.

Corollary 4.13: 3D

We can replace $2K-8$ measurements in Corollaries 4.6 and 4.7 with acceleration magnitude measurements, all as one *anchor*.

If the robot moves on a flat terrain, then the gravity can be assumed to be always down in the robot frame of reference and can be removed. The acceleration amplitude then depends only on $\boldsymbol{L}$:

$$\|\boldsymbol{a}(t)\|^2 = \|\boldsymbol{C}\ddot{\boldsymbol{f}}(t)\|^2 = \ddot{\boldsymbol{f}}^\top(t)\boldsymbol{L}\ddot{\boldsymbol{f}}(t).$$

Therefore, we can use acceleration measurements to provide extra measurements that are required due to the linearisation we use.

Corollary 4.14: 2D

We can replace $K-6$ measurements in Corollaries 4.6 and 4.7, but they do not count against Equation (4.13).

We now sketch the proofs of the corollaries.

Sketch of the Proof of Corollary 4.14

To prove Corollary 4.14, we have to show that using Lemma 4.5 we can append rows given by the acceleration measurements. Any acceleration measurement row consists of zeros, then more zeros, then polynomials of degree up to $2K-6$. We need to check that $\boldsymbol{\Phi}$ maps them correctly. Then, as each appended row will have zeros or smaller degrees on the left, we can use Lemma 4.5 as usual. This only issue is that, in the cases where new vector is zero, we have to use some other vector.

Sketch of the Proof of Corollary 4.13

The new measurements are independent of each other, as long as there are at most $K-2$ of them. They are also almost always independent from the original vectors $\boldsymbol{f}_n$.

They all correspond to one "anchor": the acceleration. If other anchors are placed randomly, it is unlikely that one will be placed exactly at $[0,0,-1]^\top$. Hence, we obtain an extra anchor that we can use at most $K-2$ times.

We check again the assumptions of Lemma 4.5, and they are satisfied for no more than $K-6$ rows.

Corollary 4.14 and Corollary 4.13 are extensions of Corollary 4.6. Similar extensions could be made in the oversampling case of Corollary 4.7, but the gain would be small compared with the rate of oversampling we need for Corollary 4.7.

Using only squared acceleration is a simple way to integrate some of the IMU measurements. We present this method here as we can prove some guarantees in this case. This method is however wasteful because it ignores the direction of the acceleration, as well as the gyroscopes and the magnetometer. A more efficient method would be to first integrate all measurements from IMU then to obtain the vector between two points along the trajectory:

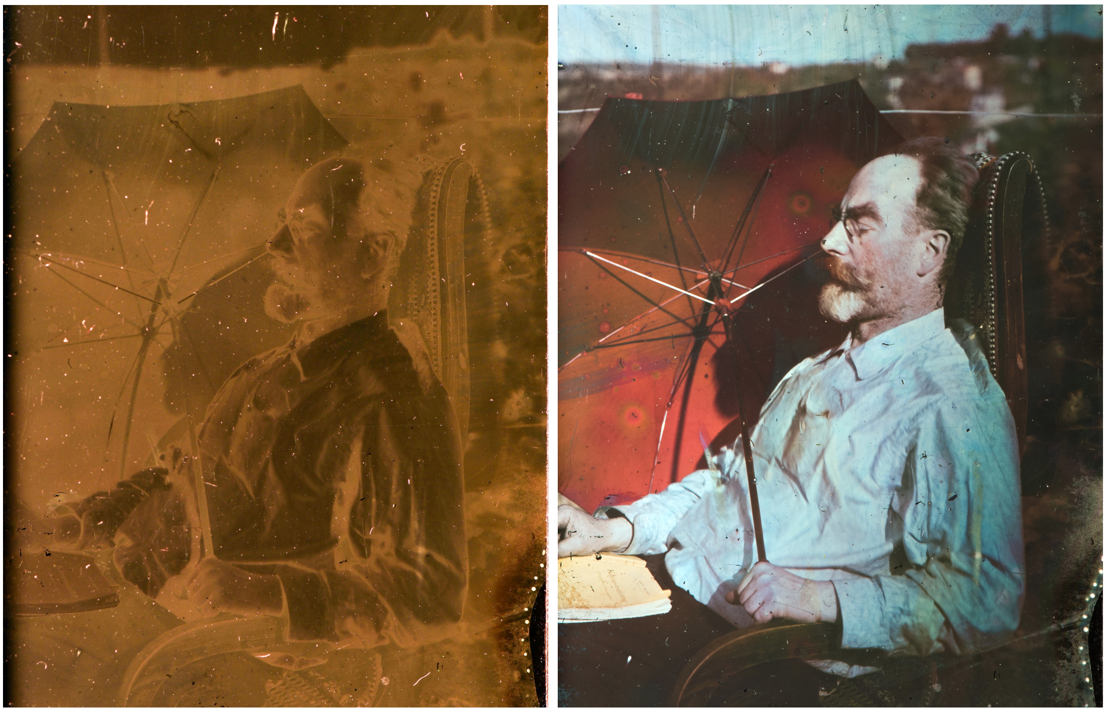
$$\boldsymbol{v}(t_n, t_{n+1}) = \boldsymbol{C}\boldsymbol{f}(t_{n+1}) - \boldsymbol{C}\boldsymbol{f}(t_n), \tag{4.29}$$

and then add this integrated measurement to our system of equations. This method provides more information but also requires some pre-processing, and it is harder to provide guarantees for it. The integrated measurement is linear in $\boldsymbol{C}$ and can be simply added to the linear system of equations we already have.

Part II.

Lippmann Photography

Introduction



Self-portrait of Gabriel Lippmann viewed under different illuminations. Left: diffuse illumination. Right: directed light, incoming at the angle equal to the viewing angle.

Gabriel Lippmann's photographic method, for which he was awarded the Nobel Prize in 1908, is almost as old as three-colour photography. It is very interesting from a theoretical perspective, as it stores the whole spectrum of light in an interference pattern. Lippmann's photographs are therefore the oldest multi-spectral images. Under the right illumination, they produce vivid colours, which we attempt to show in the figure above. Of course, static images do not do these works justice, and we strongly recommend that the reader views a real plate, if they ever have the opportunity.

Despite how intriguing Lippmann photography is, it is almost forgotten today and, until recently, it was not fully understood. We believe that studying this unique method can not only explain the inner workings of the handful of Lippmann's photographs stored in museums but also inspire new technologies.

Thesis Structure

In Part II of this thesis we tell the story of Lippmann's method (Chapter 1) and explain the current understanding of the physics behind colours in Lippmann photography (Chapter 2), and we imagine the future for Lippmann photography in new technologies (Chapter 3).

In Chapter 1, we explore how Lippmann photography was invented, why it earned a Nobel Prize, and why it was subsequently forgotten.

In Chapter 2, we show that the multi-spectral image reflected from a Lippmann plate contains distortions that are not explained by current models. We describe these distortions by directly modelling the process for general spectra, and we propose an algorithm for recovering the original spectra. We demonstrate the accuracy of our recovery algorithm on self-made Lippmann plates, for which the acquisition setup is fully understood. Finally, we show that there are too many unknowns to reliably recover 19^{th}-century spectra of historical plates.

Finally, in Chapter 3, we propose a glass-storage method inspired by Lippmann photography. If built, such storage could be both high capacity and very durable. Moreover, the methods developed to create such storage could be used to duplicate historical plates and provide a new tool for artists.

Chapter 1.

Past

The Nobel Prize is awarded to “those who, during the preceding year, have conferred the greatest benefit to humankind” [271]. In 1908, the Nobel Prize in physics was awarded to Gabriel Lippmann for his invention of interference-based colour photography [272]. However, at the basis of almost all analog and digital contemporary colour photography techniques is three-colour photography, not interference photography. Some even say that Lippmann did not have any influence on the modern colour photography [273]. How did this happen?

It could be said that Lippmann owed his award to politics. French science needed a Nobel laureate after a few lean years [273]. Lippmann’s biggest competitor in 1908 was Max Planck whose discoveries were too theoretical for the liking of the Swedish Academy of Science [274]. Still, this does not provide the full picture.

To fully appreciate Lippmann’s invention, we need to consider how his contemporaries understood colour (Section 1.1), how innovative his method was (Section 1.2), and how the understanding of Lippmann’s method evolved over time (Section 1.3).

1.1. Colour Photography

In this section, we present the context in which Lippmann photography was created: the understanding of colour and of photography at the end of the XIX century. First, we go back almost two centuries to tell the story of how the current *scientific* understanding of colour emerged. We introduce two approaches to colour; they led, respectively, to three-colour photography and to Lippmann photography. Afterwards, we briefly discuss black-and-white and colour photographic methods.

The approach to the theory of colour that led to Lippmann photography can be traced to Isaac Newton and his book *Opticks* published in 1704 [275]. Newton studied the refraction of light through a prism, thus creating a rainbow from white light. He postulated that objects themselves do not have colour. They simply reflect colours with varying strength, and our perception of a colour of an object is the colour this object reflects most strongly. Newton also studied the reflection of light on thin films, such as soap bubbles that reflect different colours, depending on their thickness and viewing angle. Today, we know that

---

Contributions: The description of the Lippmann process in this chapter is based on the joint investigation of the author of this thesis (MP) with Gilles Baechler (GB), Arnaud Latty (AL) Martin Vetterli (MV) and Adam Scholefield (AS) [270]. MP wrote this chapter.

this phenomenon is due to the wave nature of light, whereas Newton considered light to be corpuscular. Nevertheless, due to his observations, he associated physical length with colour: He claimed that the colour change of leaves in autumn occurs because leaves shrink when they dry and this shifts their colour towards red [273].

Newton was not entirely correct in his observations. Years later, Thomas Young pointed out that, in contrast to thin films, the colour of leaves do not change with the viewing angle, hence the mechanism of colour reflection cannot be the same [273]. However, Newton's idea that colour is a measurable property of light and independent of human perception will accompany us throughout this chapter.

It is not a coincidence that we single out Thomas Young as a critic of Newton's work. At the beginning of the twentieth century, Young studied the wave nature of light and demonstrated the interference of light waves [276]. He created a predecessor of the famous double-slit experiment. More importantly for us, he studied the perception of colour. He postulated that as there is no reason for the light waves to come in only three colours, then our three-colour perception must be caused by the construction of the eye [277].[1] This focus on human perception is the foundation of the second of the approaches to understanding colour.

Let us now shift our attention to the origins of photography. In 1839, J.M. Daguerre and Nicéphore Niépce published the first black-and-white photographic process called daguerreotype [273], while William Talbot worked on the process called talbotype or calotype that he eventually patented in 1847. In 1842, John Herschel discovered the cyanotype (a.k.a. blueprint). To discuss all these processes is outside the scope of this work. Here, we describe only the general steps that most photographic methods follow; it will be useful later in our explanation of Lippmann's method.

Photographic Process

In order to take a photograph, we need a photosensitive surface, created using silver salts of halogens: silver halides. In the case of daguerreotype, the surface is a silver-coated copper plate, subsequently sensitised to light; and in the case of calotype, the suface is paper covered with silver iodide. Next, the photosensitive surface needs to be exposed to the image in the camera. From a chemical point of view, exposing the photographic emulsion to light breaks some of the bonds between silver particles and halide ions by reducing them. Once free, these metallic silver atoms form tiny specks that combine into a *latent image*. As silver salts are sensitive to only blue and violet light, dyes are added to the emulsion to extend the sensitivity across the full visible spectrum. This process used to take hours for the first photographic methods, but was subsequenly shortened to fractions of a second. The latent image is often faint or entirely invisible and has to be strengthened through a development process. The development is a chemical reaction that completes the reduction of silver halide into metallic silver in the exposed grains. The latent image serves as a catalyst for this reaction and, as a result, the processed plate contains an enhanced version of the latent image. Interestingly, daguerreotype was

---

[1] His work was further developed in 1850 by Hermann von Helmholtz who was generally interested in human perception.

not developed chemically but through the additional exposure to red light (to which the unexposed plate was not sensitive). Usually the exposed part of the image becomes dark, creating a *negative*. In the case of daguerreotype, the exposed part was less specular than the rest of the silver, thus it was darker or brighter, depending on the viewing conditions. Finally, the image has to be stabilised, or *fixed*, to prevent the plate from further reacting to light. This is done by washing the photograph with an appropriate solution. For example, Herschel discovered that sodium thiosulfate is a solvent of silver halides and could be used for this purpose.

Colour Photography

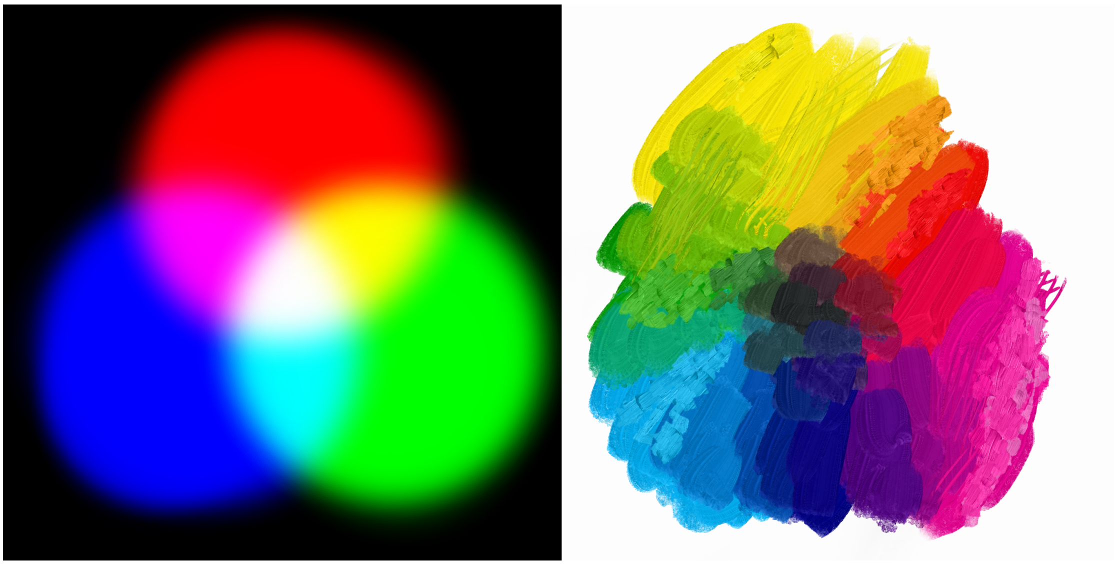


Figure 1.1.: Additive and subtractive colour-mixing. Left: additive colour-mixing, e.g., by overlapping coloured beams of light. Right: subtractive colour-mixing, e.g., of paints of printer inks.

Soon after the invention of the first black-and-white photographic processes, the public demanded colour photographs. To people used to seeing world in colour, it was natural that the photographs should look the same.[2] In 1850, it was common to colour photographs by hand. The first known colour photograph — of light refracted on a prism — was taken as early as 1848 by Edmond Becquerel [278, 279][3]. Becquerel plates were mysterious and a bit controversial because they were impossible to fix. Any attempt to make them resistant to light also removed the colours. The reason for this was eventually learned, as we explain in Section 1.3.

---

[2] Paradoxically, with all the modern technology, we are more used to gray-scale inventions preceding their colour equivalents: from television, through home printers to most recently, e-ink displays.

[3] He was a son of Henri Becquerel who discovered radioactivity by mistake.

In the 1860s, the public demands were answered by three-colour photography. In fact, two different methods were invented almost simultaneously. The first approach — additive colour-photography — was proposed by physicist James Clerk Maxwell and photographer Thomas Sutton. Yes, this is the same Maxwell who formulated the Maxwell equations of electromagnetism and who proposed that light is an electromagnetic wave. We could expect that Maxwell would have shared Newton's perspective that colour is only a wavelength of light. However, in his study of colour, Maxwell adapted Young's approach. He performed a systematic study of mixing coloured light. For example, he showed that combining certain blue and yellow *light* resulted in not green, but pale pink light! In 1861, he proposed a method of producing a full colour *projected* image [280] by using only red, green, and blue lights, see Figure 1.1 (left). With help of Sutton, he used this method to photograph a piece of patterned coloured ribbon. One of the problems with his additive method was that the images had to be either projected or illuminated from the back.

At the same time, Charles Cros and Louis Ducous du Hauron proposed subtractive colour-photography [274] that was more akin to how paints and pigments work. As it worked by combining cyan, yellow, and magenta (see Figure 1.1, right) on the white background, it could be viewed with reflected light. However, for a long time, this method was too complicated for amateurs.

In 1891, when Gabriel Lippmann proposed his method, none of the three-colour methods were widely adopted. These methods were called *indirect* colour methods and, in the press opinion, were not scientific [274]. The Becquerel plates were not understood, although the wave nature of light had been known for some time. Physicists, such as Whilhelm Zenker [281] or Otto Wiener [282] discussed the potential applications of interference and of standing waves of light. There was a need for a working "natural" or *direct* colour photography, i.e., photography that would have strong foundations in physics. The interference-based photography was the perfect answer to this need.

1.2. Lippmann's Method

Lippmann subscribed to Newton's view that colours were properties of the light waves and not sensations in the human eye [274]. Therefore, he wanted the process to produce not only nice images but also *physically correct* images. To this end, he first developed the theory of interference photography based on the Fourier transform that was not widely used in optics at that time [283]. Only after developing theoretical foundations for his process did he spend years perfecting a suitable photographic emulsion [273].

His idea was to put a mirror in contact with thick photosensitive material. The light reflected from the mirror would create an interference pattern that would be recorded inside the photosensitive emulsion. Lippmann calculated that for a monochromatic light wave — a light wave with a single frequency — the light reflected from the interference pattern would have the same frequency as the recorded wave.

From his theory, Lippmann inferred that an interference-based photographic method had to satisfy three conditions [273]. First, the emulsion had to be in contact with the mirror for a strong reflection. He proposed to use liquid mercury to create a reflective surface. Second, the emulsion had to be transparent and have extremely fine silver grains—around 10 nm in diameter—to enable the recording of the interference patterns of light. Such small grains meant long exposure times. Finally, the emulsion had to be isochromatic, i.e., had to react with the same strength to all wavelengths. Initially, Lippmann did not have isochromatic plates, hence he selectively exposed the photographs to different colours by using filters to account for the intensity differences [274]. Then, following a suggestion from Labatut, he dyed the plate to make it sensitive to different colours [274]. To make the plates truly isochromatic, he collaborated with the Lumière brothers [284, 285] who eventually produced plates better than Lippmann himself had created [274]. With isochromatic plates, Lippmann was able to separate the precise physical theory from the fuzzy chemical process that he did not seem to like [274].

After Lippmann published his work [286, 287], he had to convince others that the plates indeed worked as described. A strong argument for his theory was that the plate colours changed hue when viewed from an angle, hence the colour has to be interference based. Interestingly, the plates also changed colours when they swelled due to moisture [274]. Newton would have been happy to see this effect, even if not in the autumn leaves. The most difficult part to believe was that the polychromatic — not single wavelength — colours can be reproduced by interference [273]. To demonstrate this, Lippmann took numerous photos of natural objects, including the (stuffed) parrot depicted on the left in Figure 1.2. To prove his point, Lippmann eventually extended his calculations to general spectra. He was also supported by Wilhelm Zenker [281] who wrote about the Lippmann-Zenker Theory, despite the fact that Lippmann did not cite his work.

We give an overview of Lippmann's method (depicted on the right of Figure 1.2), with an emphasis on the differences between black-and-white photography, as described in Section 1.1: Photographic Process.

Recording

Lippmann's process works by capturing an interference pattern in a photosensitive emulsion. More precisely, the image is focused onto a photographic plate by standard camera optics. As shown in Figure 1.2 (right), the photographic plate consists of a light-sensitive emulsion on a sheet of glass. A mirror is created at the surface of the emulsion, traditionally by putting the emulsion in contact with liquid mercury. For the image acquisition, the plate is oriented such that the light from the scene passes through the glass then through the emulsion, before reflecting back from the mirror. The reflected light interferes with the incoming light and and the resulting interference pattern exposes the emulsion differently at different depths. For example, in the simple case of monochromatic light, sinusoidal standing waves appear.

Development and Fixing

Once the plate is exposed, it is removed from the liquid mercury and processed via standard photographic development techniques. As Lippmann plates need to store an

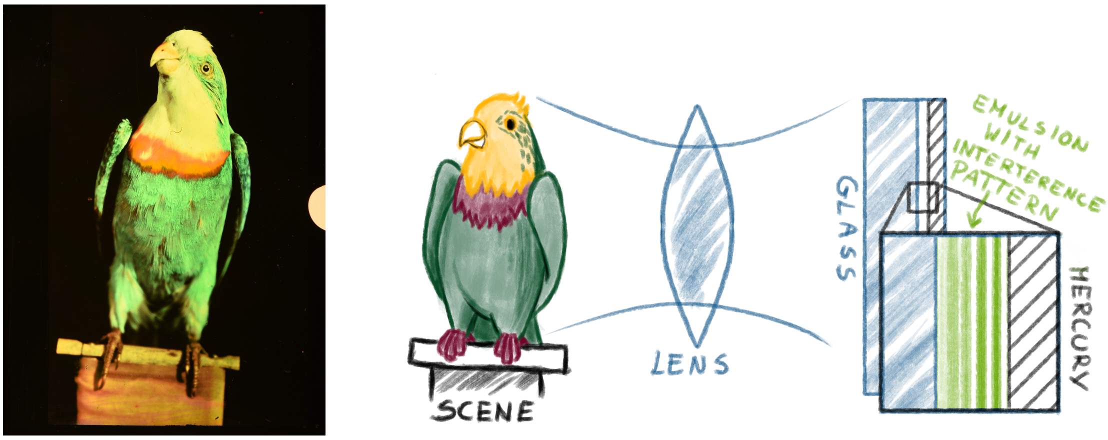


Figure 1.2.: Left: photograph of a parrot made by Gabriel Lippmann in 1891; currently in the collection of Musée de l'Elysée in Lausanne. Right: principle of Lippmann photography: The scene on the left is focused with a photographic lens on the plane of the emulsion. On the back of the emulsion is a mercury mirror from which the light reflects and the incoming and reflected waves interfere. Its spatially varying intensity creates patterns that are captured inside the depth of the emulsion (see the close-up).

interference pattern in their depth, they are thicker than standard photographs. This makes the development process more difficult; as the developer solution diffuses from the top surface of the emulsion, the enhancement weakens towards the emulsion-glass interface.

After the development bath, a fixing process can be applied to wash out the dyes, halide crystals and silver halides [288], and to make the plate insensitive to light.

Viewing

The recorded image becomes visible when the plate is illuminated by a white light source. The incoming light is scattered (see Figure 1.4) by the tiny silver particles that are distributed throughout the depth of the emulsion. The light waves reflected at different depths interfere with each other, and depending on the recorded interference patterns, some reflected colours interfere constructively and others do so destructively.

To make the most of the colours, the plate is viewed from the back — where the mirror previously was — and where the interference pattern is the strongest. The other side of the plate is painted black to prevent any light from going through. Furthermore, a prism with a refractive index close to that of gelatin is attached to the top of the plate in order to separate the strong front-surface reflection from the reflection from inside the plate.

1.3. After Lippmann

The Lumière brothers were not the only contemporaries of Lippmann interested in his work. For example, Otto Wiener [290] extended Lippmann's theory. He studied the phase

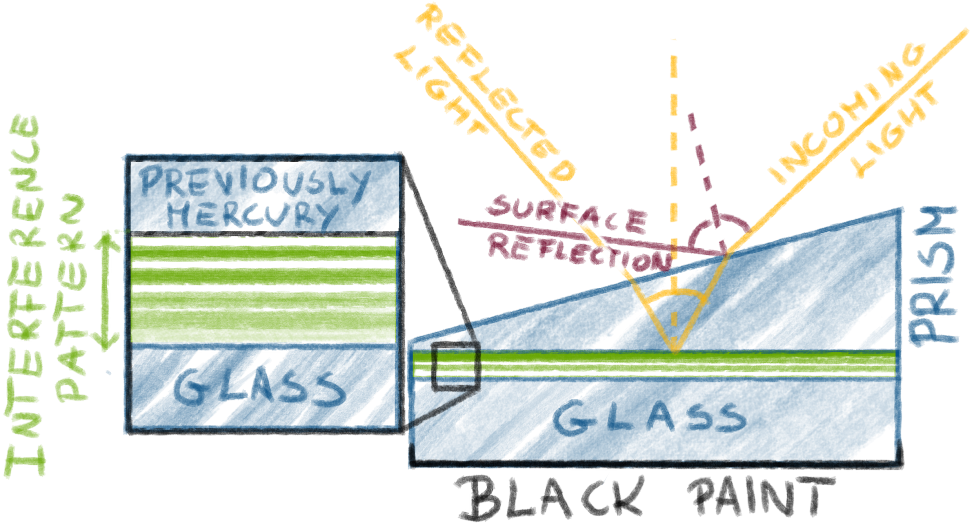


Figure 1.3.: A Lippmann plate prepared for viewing after development. The glass plate is painted in black to minimize reflections, and a prism is mounted on the side of the plate that was in contact with the mercury layer. The light rays reflecting at the surface of the prism do not coincide with the rays reflected from the plate.

of the light reflected off the surface of the plate when a prism is not added and the effect this phase has on the combined reflection. Rothé [291] showed that the interface between the gelatin and air generates reflections that are sufficiently strong to create interferences. The elimination of mercury was important, due to its toxicity. Ives [292] achieved a closer match between the Lippmann process and Lippmann's theory, mainly by refining the recipes and the chemicals used to make and develop the Lippmann plates.

The creation of an interference pattern in the photographic emulsion was experimentally confirmed multiple times. Neuhauss [293], Lehmann [294], and Cajal [295, 296] observed the interference patterns of Lippmann plates under optical microscopes by swelling the gelatin. Wiener demonstrated the existence of the standing waves by using a very thin

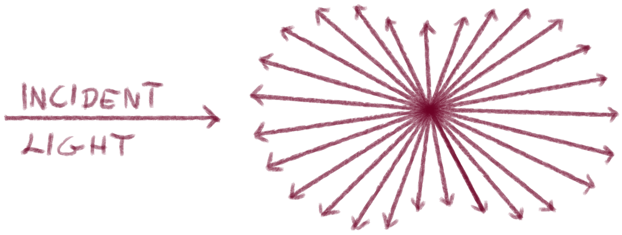


Figure 1.4.: Directions in Rayleigh scattering. Silver grains are around 10 nm and the wavelength of visible light is of the order of ∼ 500 nm, therefore reflection of visible light from silver grains can be described by Rayleigh scattering [289].

film, but on a small angle with respect to the mirror, effectively imaging a slice through the interference pattern [282].

The mystery of the Becquerel plates — colour photography that could not be fixed — was also solved [273]. The initial hypothesis was that they worked like Lippmann plates, but this did not explain the fixing problem. The alternative hypothesis was based on Carey Lea's study of coloured, light-sensitive silver halides. Sliver halides of a certain colour are less likely to be reduced by this colour, because they reflect it strongly. Therefore, if it is used in a photographic plate, in the region illuminated by, for example, blue light, only the blue sliver halides would remain. But then fixing, which removes silver halides, would also remove the colour. This latter hypothesis was confirmed by Weiner. He observed that the transmission through a Becquerel plate does not have the opposite colour of the reflection, hence some chromatic absorption must occur.

Despite all the interest in Lippmann photography, it never became practical. The Lumière brothers never managed to make the Lippmann process reproducible or to make the plates isochromatic enough to be commercialized. [274]. Due to small the size of the silver grains, the exposure times were long. Moreover, the Lippmann plates were impossible to copy.

The Lumière brothers turned their interest to three-colour photography and developed Autochrome, an additive colour photography that is relatively simple to use. Its most significant advantage was that it combined red, green, and blue filters inside the plate. The photosensitive material was covered with a thin layer of grains of starch in one of the three colours; these grains acted as tiny filters.

In 1935, L. D. Mannes and L. Godowsky, working with Kodak Research Laboratories [297] introduced Kodachrome: a subtractive "tripack" process, where the three light-sensitive layers where stacked tightly together. After the development of the film, the Kodachrome film could be projected without filters.[4] Finally, people stopped seeing the indirect three-colour photography as inferior to direct interference-based photography, and the terms direct and indirect were abandoned [274].

In 1940s, an invention similar to Lippmann photography came from the field of X-ray microscopy. Dennis Gabor proposed to use holography to create lensless microscopy to avoid aberrations created by lenses [299, 300]. He was awarded the Nobel Prize in Physics in 1971 "for his invention and development of the holographic method" [301]. Like Lippmann photography, holography is also based on the photographic recording of the interference of light. Contrary to Lippmann photography, holography uses a reference wave — not the reflected wave — to create interference. This means that holography can record the phase of light, but also that it has to be recorded with coherent light (e.g., laser). Not surprisingly, three-colour holography was proposed.

The holographic process is very similar to the Lippmann process. The photosensitive emulsions can be used interchangeably. This means that industrially produced holography plates can be re-purposed for Lippmann photography by Lippmann photography enthusiasts [283]. The most significant difference in these two processes is that, at the very end, after fixing, holographic plates are bleached. Bleaching converts the metallic

---

[4] In 2010 the last roll of Kodachrome was developed [298].

silver grains back into silver halides. This makes the plate transparent. However, silver has a higher index of refraction than gelatin, hence the exposed areas are more refractive, and the reflected hologram is still visible. We will discuss refraction based plates in more detail in Chapter 3.

More recently, various aspects of the Lippmann process were revisited. Phillips et al. [302, 303] precisely described the scattering of the silver grains, as well as their size limitations. They established constraints regarding the aperture (f-number) of the lens for a successful Lippmann photograph. Fournier et al. [304, 305] studied the creation of extremely fine-grain emulsions and analyzed the structure and spectrum reproduction of historical and contemporary photographic plates.

Nareid and Pederson [306, 307] introduced a mathematical model based on local changes in the refractive index of the plate. Their model is useful for crafting alternative methods for recording Lippmann or holographic plates, as we will see in Chapter 3.

Last but not least, Bjelkhagen studied many practical aspects of the technique, such as recording with holographic plates and the use of the method for security purposes [308–311], and Alschuler described observations from creating his own Lippamnn photographs [312].

Despite all this research, some misconceptions about Lippmann plates persist. The most common one is that the Lippmann method enables a perfect reconstruction of the original spectrum, at least from a theoretical point of view. This is not true. In Chapter 2, we show a number of inconsistencies between the recorded and reflected spectra, many of which have never been documented even in modern studies [305–307, 311].

Chapter 2.

Present

In Chapter 1, we learned that Lippmann's idea attracted the attention of his contemporaries and more researchers. Since the invention of Lippmann photography, the most controversial question is whether the Lippmann process exactly reproduces the polychromatic spectra. Some modern studies [305, 311] accept the hypothesis of perfect reconstruction, assuming a plate of infinite thickness, perfectly uniform development, and re-illumination with a flat spectrum. Other authors, such as Nareid and Pedersen [306, 307], show models that predict that recorded and reflected spectra are not the same.

In this chapter, we revisit the question about spectrum reconstruction and show that the reflected spectrum is not the same as the original one, even assuming no additional development effects. We explain the artefacts of the reflected spectrum that have never been documented in the literature. To do this, we apply signal-processing methods that are not very distant from Lippmann's original theory that is based on the Fourier transform. In Section 2.1, we formulate a mathematical model of the Lippmann plate. Then, in Section 2.2, we estimate how much information of the original spectrum can be recovered from the recorded spectrum. We verify our model experimentally in Section 2.3 and conclude this chapter in Section 2.4.

We adopt the notation described in List of Symbols that is provided at the beginning of this thesis. This way our notation is consistent with the rest of this thesis, but not with the optics literature. We sincerely apologise for this to any physicist that might be reading this work.

2.1. Mathematical Model

In this section, we propose a mathematical model for the Lippmann process. On a high level, we model the Lippmann process as a signal-processing pipeline. We interpret the recoding of a plate as applying an analysis operator, the effects of development as filtering,

---

Contributions: The work in this chapter is joint work of the author of this thesis (MP) with Gilles Baechler (GB), Arnaud Latty (AL), Martin Vetterli (MV), and Adam Scholefield (AS) [270]. This work has also been included in GB's thesis [313]. GB, MV, and AS designed research; GB, AL, AS, and MP performed research; GB and AL performed experiments; GB and MP analysed data and ran simulations; MP wrote this chapter.

The code used for simulations is available at `https://github.com/LCAV/lippmann-photography`.

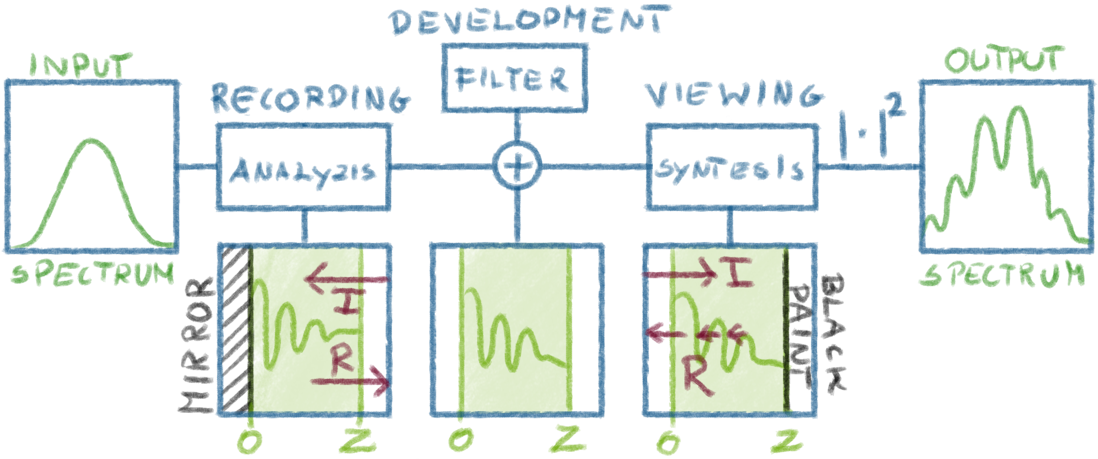


Figure 2.1.: Lippmann photography from signal processing point of view. We can interpret each stage as applying consecutively analysis, filtering and synthesis operators, followed by a non linearity in the form of squared norm.

and the viewing as a applying synthesis operator and taking the squared magnitude, see Figure 2.1.

To calculate these operators, we need to model the light wave. The most general approach would be to use Maxwell's equations. However, to model the Lippmann process we do not need to take into account the polarisation of light, the electric currents, or magnetisation in the material. Therefore, we can use the wave equation to describe the propagation of light. The wave equation is easier to analyse in the Fourier domain. In short, the wave equation can be solved for each frequency separately, then the solution can be combined to describe the propagation of any wave. In Fourier optics, a wave is usually described by the spatial frequency k, called the *wave number* [314]. In our case, as we consider only planar waves travelling perpendicularly to the direction of the plate, the wave number is proportional to the frequency ω.

We assume that all waves travel in the direction perpendicular to the surface of the plate, because such waves give the strongest interference patterns and the strongest reflections. Our assumption is not realistic because a planar wave has the same colour and intensity everywhere. Nevertheless, such waves are a reasonable approximation if the scale of variation in colour or phase is much larger than the wavelength of light, which is true for artistic photography.

Another assumption we make is that all the media (air, photographic emulsion, etc.) are homogeneous, i.e., have a constant index of refraction. In this case, a solution $u(z,t)$ to the wave equation for a single frequency, travelling along the z axis is given by

$$u(z,t) = a_k e^{i(\omega t - kz)}$$

where a is the complex amplitude (amplitude and phase) and $k = \frac{\omega n}{C}$ is the wave number, where C is the speed of light in vacuum and n is the index of refraction of the medium. Note that we adopt the standard convention from optics that the wave propagates along the z direction, instead of, e.g., the x direction; z does not refer to the z-transform.

For positive k, the wave travels in the positive z direction and for negative k it travels in the negative z direction. If we want to consider jointly the wave numbers k and $-k$, we can include the dependence on z and the sign into the amplitude $a(z)$:

$$u(z,t) = a(z)e^{i\omega t}.$$

Therefore, any polychromatic wave $u(z,t)$ travelling through a homogeneous medium can be written as an integral of all its monochromatic components:

$$u(z,t) = \int_{-\infty}^{\infty} a(k)e^{i(\omega t+kz)}dk. \tag{2.1}$$

If we prefer to parameterise the wave by only the positive wave numbers, we can write

$$u(z,t) = \int_{0}^{\infty} a(k,z)e^{i\omega t}dk. \tag{2.2}$$

Note that in Equation (2.1), we do not explicitly write the dependence of the wave number k on the frequency ω. We use this notation because, in our work, the wave number is more relevant than the frequency. The oscillations of the visible light waves are too fast to be registered [314], and the phase of a wave cannot be measured without some reference phase. Therefore, we often drop the $e^{i\omega t}$ factor and, with a slight abuse of notation, write $a(k)$ meaning $a(k(\omega))$.

Instead of measuring the amplitude, we measure the power spectrum of a wave:

$$p(\omega,z) = |a(\omega,z)|^2 \text{ and } p(k,z) = |a(k,z)|^2. \tag{2.3}$$

If a wave propagates through a homogeneous medium, its spectrum does not depend on z, and we write $p(z)$. We also define the intensity of a light wave at a given point z:

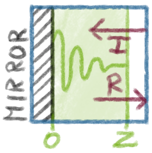
$$l(z) = \int_{-\infty}^{\infty} p(k)dk. \tag{2.4}$$

Note that the intensity depends only on the total power of the signal and not on its phase.

This concludes the core optical concepts needed in this chapter. For a good general reference on optics, we can recommend the freely available book by J. Peatross and M. Ware [314].

We now describe in more detail how we model each step of the Lippmann process. In each step, we have to make additional decisions regarding which physical phenomena have a significant influence on the reflected spectrum and which are negligible. We introduce the required concepts as we go.

Analysis: Recording a Lippmann Plate



To describe the recording of a Lippmann plate, we first model the wave, its reflection and interference, then the chemical process of exposing the silver halides.

We first consider a planar wave travelling in the negative z direction. We set $z = 0$ to be the position of the mirror. Upon hitting it, the wave is reflected back. We define the reflection coefficient of the mirror as $R = \rho e^{i\theta}$, where ρ is the attenuation factor and θ is the phase shift. Later, in Section 2.3, we will see that the value of R has a strong influence on the resulting spectrum. Before the recording, the Lippmann plate is homogeneous. Therefore, for a single frequency ω, the combined incoming and reflected wave is

$$a(k,z) = \underbrace{a_0(k)e^{i(\omega t - kz)}}_{\text{incoming}} + \underbrace{Ra_0(k)e^{i(\omega t + kz)}}_{\text{reflected}} = a_0(k)e^{i\omega t}\left(e^{-ikz} + Re^{ikz}\right),$$

where a_0 is the amplitude of the incoming wave, thus $a_0(k) = 0$ for negative k. According to Equation (2.3), the combined wave has a power spectrum that varies depending on z:

$$p(k,z) = |a_0(k)|^2 \left|e^{-ikz} + Re^{ikz}\right|^2 = |a_0(k)|^2 \left(1 + \rho^2 + 2\rho\cos\left(2kz - \theta\right)\right).$$

From Equation (2.4), we obtain that the intensity of the polychromatic wave, i.e., the interference pattern is

$$l(z) = \int_0^\infty p_0(k)\left(1 + \rho^2 + 2\rho\cos\left(2kz - \theta\right)\right) dk, \tag{2.5}$$

where p_0 is the power spectrum of the recorded wave, $p_0(k) = |a_0(k)|^2$. Note that Equation (2.5) is the generalized Fourier cosine transform of the recorded power spectrum p_0.

Interestingly, the intensity varies spatially in z and takes the form of *standing waves*[1], or more generally *partial* standing waves when $\rho < 1$. In the case of monochromatic light, the interference pattern takes the form of cosines, whose frequency of oscillation depends on the wavelength (see the left side of Figure 2.2). For a Gaussian spectrum, the interference pattern is more complex, see the right side of Figure 2.2.

The amount of silver halides that are reduced to silver and that create silver grains under exposure to light depends on the energy of the interference pattern. Here, we model the density of silver grains as a continuous function and assume that the process is deterministic. This is not entirely accurate, but it is a standard assumption that makes the subsequent steps of the modelling easier.

We assume that the silver density after exposure is proportional to the intensity of the interference pattern. In practice, the response of a photographic plate to the *amplitude* of light is described by the so-called *Hurter-Driffield curves* [315], see Figure 2.3. Here, we assume that the intensity of the interference pattern is in the regime where the Hurter-Driffield curve is linear. Furthermore, we assume that the slope of the linear part

---

[1] Technically speaking, standing waves only form in the monochromatic case. In the polychromatic case, we can think of the shape of the resulting wave as an infinite sum of standing waves.

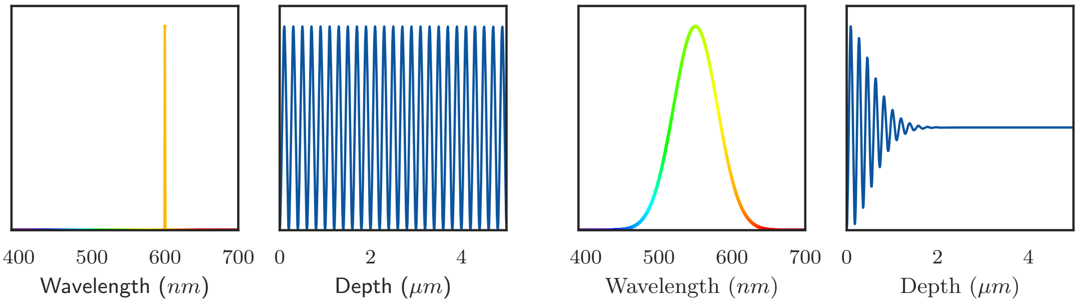


Figure 2.2.: Interference patterns created by different waves, using perfect mirror ($r = -1$). First plot in the pair depict the incoming eave, the second plot depicts the interference pattern. Left: a monochromatic wave creates sinusoidal standing waves in the photographic emulsion. Right: a wave with a Gaussian spectrum creates an interference pattern localised close to the mirror (depth 0).

of the curve, called γ in photography, is equal to 2 for a Lippmann plate. As the plot is logarithmic, a slope equal to 2 corresponds to a quadratic response to the amplitude of light or a linear response to the intensity of light.

Although γ can vary widely for holographic plates (from 1 to 6 [316]), the assumption $\gamma = 2$ seems to be popular in the literature [307].

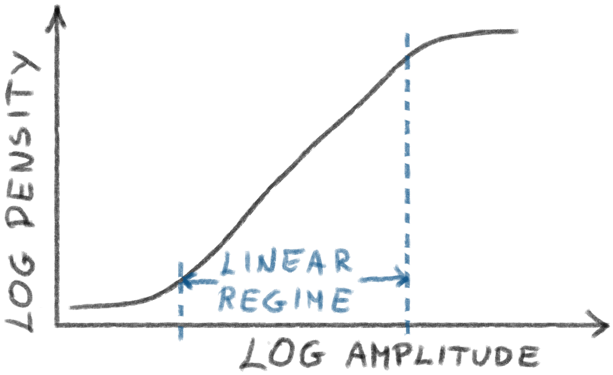


Figure 2.3.: H-D curve. Note that the axis are logarithmic. The slope of the flat part of the curve is called γ. In our model, we assume that slope is equal to $\gamma = 2$ everywhere; our assumption is correct if we are in the middle of the curve.

Filtering: Development

Here, we model the development of a Lippmann plate. The development reduces the remaining silver proportionally to the latent image, hence we model it as a multiplicative factor $c(z)$. The factor c depends on the depth of the plate because the developer diffuses from the top of the plate: the deeper layers are generally less developed than the top ones.

The concentration of the developer is given by the diffusion equation. For simplicity, we approximate the solution to the diffusion equation with a decaying exponential:

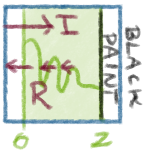
$$c(z) \approx e^{-\tau z}, \tag{2.6}$$

where τ is the decay rate.

In summary, the density of reflective silver grains after the development is

$$r(z) = e^{-\tau z} \int_0^\infty p(k) \left(1 + \rho^2 + 2\rho \cos\left(2kz - \theta\right)\right) dk. \tag{2.7}$$

Synthesis: Viewing a Lippmann Plate



In order to model a reflection from a silver plate with varying density, we first assume that the plate is illuminated with monochromatic light. As the light transport is linear[2], any other reflection can be obtained as a superposition of the monochromatic waves.

We assume that the plate is illuminated by a light wave u_i travelling in the direction of positive z. This light wave reflects from silver grains inside the plate. As we model the distribution of silver grains as a continuous function, we consider a reflection from a thin layer of the plate at depth z, of thickness Δz, see Figure 2.4. After the wave u_i goes through this layer, $r(z)u_i(z,t)\Delta z$ of it is reflected; therefore $(1 - r(z)\Delta z)u_i(z,t)$ is transmitted. Similarly, a wave u_r, which has been reflected deeper in the plate, travels in the opposite direction. At depth z it will to grow by the reflected part of u_i, $r(z)u_i(z,t)\Delta z$ and will shrink by its own reflection from the plate: $r(z)u_r(z)\Delta z$. This second order reflection — the reflection of the reflected wave — adds to the forward travelling u_i.

This dependence of change in u_r on u_i and change in u_i on u_r leads to a system of differential equations that is difficult to solve analytically for an arbitrary r. As it is a system of linear differential equations, it can be reliably solved numerically. We describe this system Box 2.1, but here we study an approximate analytical solution of the light transport in the plate, the same way Lippmann did. This enables us to study the combined recording and reflection in an analytical form.

To obtain the analytical solution, we neglect all the reflections of an order greater than one, and we assume that the incoming wave is not affected by the reflections: $u_i(z) = a_i e^{ikz}$ for all z. This pair of assumptions is often called the first-order (or Born) approximation, as it is equivalent to taking the smallest-order expression in the powers of r, see Box 2.1. Therefore, it is realistic only for the small r, see Figure 2.5.

Under these assumptions, the reflected wave (at $z = 0$) is an integral over z of waves reflected at depth z:

$$\widetilde{u}(k, z = 0) = a_i(k) \int_0^Z r(z') e^{-i2kz'} dz'. \tag{2.8}$$

Here, Z is the depth of the plate, and we use $\sim$ to denote the observed waves to distinguish them from recorded waves or waves used to illuminate the plate. The phase under the

---

[2] The light transport is only linear for energies that are of interest here; in Chapter 3, we will see a situation when light transport is not linear.

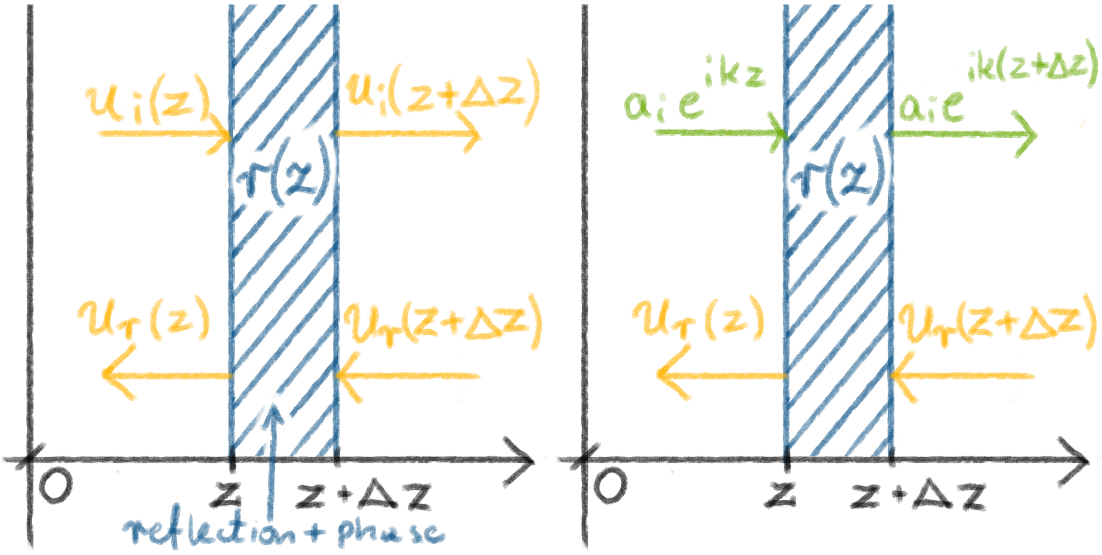


Figure 2.4.: 1D reflection model from a single layer. Left: all reflections. Right: first order approximation we make.

integral is e^{-i2kz} because partial waves reflected at depth z have to travel to depth z and back. We ignore the phase change caused by the reflection of the silver grains, as it is the same for all partial waves.

Here, $\widetilde{u}$ is defined only for $k > 0$, that is to say, we consider only the waves reflected from the left side of the plate; the right side of the plate is painted black and the reflection from it is 0.

Note that Equation (2.8) does not preserve the energy; the intensity of the incoming wave is the same throughout the depth of the plate, even though some of it is previously reflected. This is problematic when the plate is of large (infinite) thickness. Then, we also need the integral of r over the depth of the plate to be small or at least finite. The development factor $c(z) = e^{-\tau z}$ from Equation (2.7) is helpful, because if p is bounded and if $\tau > 0$, then r is indeed integrable.

Equation (2.8) gives us the wave at the edge of the plate. As always, we measure its power spectrum:

$$\widetilde{p}(k) = |\widetilde{u}(k)|^2 = |a_i(k)|^2 \left| \int_0^Z r(z) e^{-i2kz} dz \right|^2 . \tag{2.9}$$

Unless stated otherwise, we assume that the plate is illuminated with a flat spectrum, thus $a_i(k)$ is constant; for simplicity we set $|a_i(k)|^2 = 1$.

Combined Model, a.k.a. The Lippmann Transform

Here, we combine all three steps of Lippmann's method to model how the recorded power spectrum maps to the reflected power spectrum. We begin with modelling the linear part of the mapping, i.e., up to Equation (2.8). Later in this section, we show that it can be

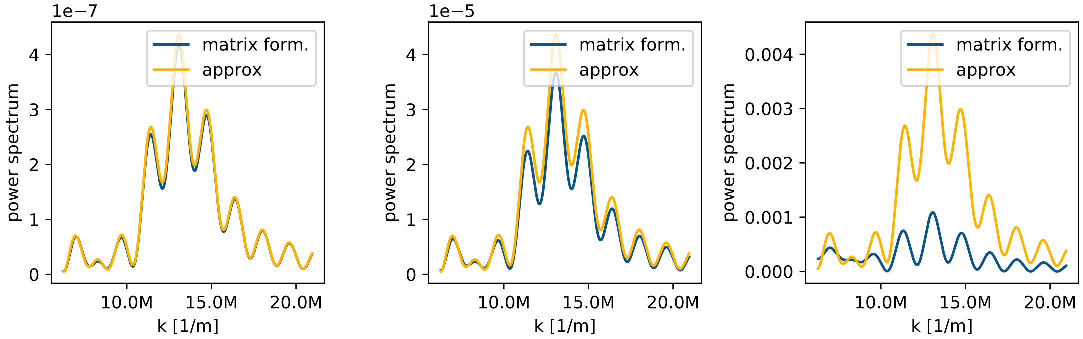


Figure 2.5.: Accuracy of the first order approximation. Power spectra of light reflected from the same interference patterns, but with the reflectivity increasing by factor of 10 between consecutive figures. The reflected power of the first order approximation increases by the order of 100, between consecutive figures, but the reflected power calculated numerically increases more slowly than that.

written as an integral operator. More formally, the amplitude at frequency k' of a wave reflected from a plate of depth Z is

$$\widetilde{a}(k') = \int_{-\infty}^{\infty} p_0(k)\widehat{h}(k,k')dk, \tag{2.10}$$

where we assume that $p_0(k) = 0$ for $k < 0$ and the *kernel* $\widehat{h}$ is defined as follows:

$$\begin{aligned}\widehat{h}(k,k') &= \rho e^{-i\theta}\widehat{s}(k'-k) + \left(1+\rho^2\right)\widehat{s}(k') + \rho e^{i\theta}\widehat{s}(k'+k), \text{ where}\\ \widehat{s}(k) &= \frac{1}{(2ik+\tau)}\left(1 - e^{-(\tau+i2k)Z}\right).\end{aligned}$$

Here, we explicitly write the hat over $\widehat{s}$ and $\widehat{h}$ to highlight that they are expressed in the spatial frequency domain k. In the spatial domain, the inverse Fourier transform of $\widehat{s}$ is

$$s(x) = \frac{1}{2}\mathbb{1}_{[0,2Z]}e^{-\frac{\tau x}{2}}, \tag{2.11}$$

which we calculate later, together with the derivation of Equation (2.10).

The formulation of the Lippmann transform that uses $\widehat{h}$ resembles the formulation of integral transforms in kernel-reproducing Hilbert spaces (KRHS, Part I: Section 1.1). The difference is that $\widehat{h}$ is not symmetric, hence we cannot use the core results from KRHS that are based on spectral decomposition of the integral operator. An example $\widehat{h}$ is shown in Figure 2.6.

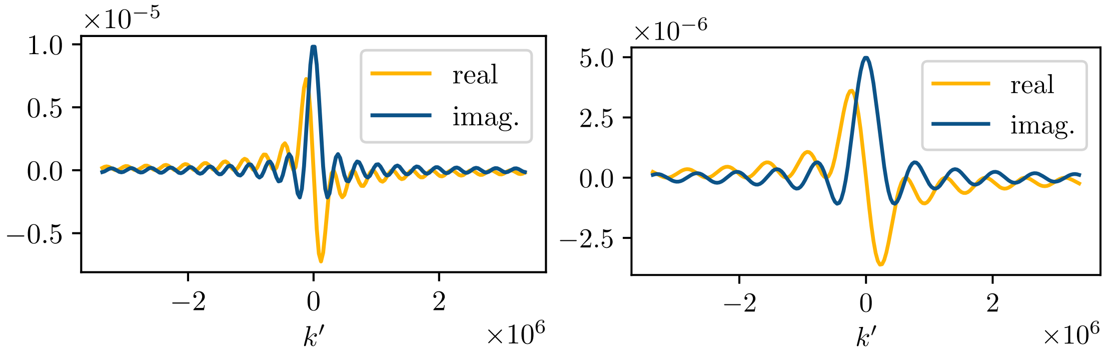


Figure 2.6.: Filters $\widehat{s}(k')$ for different values of Z. Left: $Z = 10\ \mu m$. Right: $Z = 5\ \mu m$. Compared to the left, the oscillations are less pronounced.

Proof of Equation (2.10) and Equation (2.11)

First, let us calculate that indeed the Fourier transform of $s(y)$ is $\widehat{s}(k')$ (Equation (2.11)). We have

$$\begin{aligned}\widehat{s}(k') &= \int_{-\infty}^{\infty} \frac{1}{2}\mathbb{1}_{[0,2Z]} e^{-\frac{\tau x}{2}} e^{-ixk'} dx \\ &= \frac{1}{2}\left[\frac{-1}{ik'+\tau/2} e^{-\frac{\tau z}{2}} e^{-ik'x}\right]_{x=0}^{x=2Z} \\ &= \frac{1}{(2ik'+\tau)}\left(1-e^{-(\tau Z+i2k'Z)}\right).\end{aligned}$$

Using this result, we prove Equation (2.10). Combining Equations (2.5) and (2.8) gives a double integral equation:

$$\widetilde{a}(k') = \int_0^Z c(z) \int_0^\infty p_0(k) \left(1+\rho^2+2\rho\cos\left(2kz-\theta\right)\right) e^{-i2k'z} dk dz$$

By changing the order of integration, we can integrate out the variable z:

$$\widetilde{a}(k') = \int_0^\infty p_0(k) \underbrace{\int_0^Z c(z) \left(1+\rho^2+\rho\cos\left(2kz-\theta\right)\right) e^{-i2k'z} dz\, dk,}_{\widehat{h}(k,k')}$$

That gives us the definition of $\widehat{h}$:

$$\widehat{h}(k,k') = \int_0^Z \left(\underbrace{1+\rho^2}_{\text{(i)}} + \underbrace{\rho e^{-i\theta} e^{i2kz}}_{\text{(ii)}} + \underbrace{\rho e^{i\theta} e^{-i2kz}}_{\text{(iii)}}\right) e^{-\tau z} e^{-i2k'z} dz.$$

Each term in the brackets can be expressed in terms of $\widehat{s}(x)$ if we apply a change of variables $x := 2z$. Indeed, we obtain

$$\widehat{h}(k,k') = \int_{-\infty}^{\infty} \left(\left(1+\rho^2\right) \widehat{s}(x) e^{-ik'x} \rho e^{-i\theta} \widehat{s}(x) e^{-i(k'-k)x} + \rho e^{i\theta} \widehat{s}(x) e^{-i(k'+k)x} \right) dy,$$

where the integral can be from minus infinity to infinity because $\widehat{s}(x)$ already contains the indicator function. Therefore,

$$\widehat{h}(k,k') = \rho e^{-i\theta} \widehat{s}(k'-k) + \left(1+\rho^2\right) \widehat{s}(k') + \rho e^{i\theta} \widehat{s}(k'+k).$$

■

It is convenient for us to consider separately the three parts of the amplitude $\tilde{a}$ (Equation (2.10)), as in the equation above. We write

$$\begin{aligned} \widetilde{a}(k') &= \int_0^{\infty} p_0(k) \widehat{h}(k,k') dk \\ &= \overline{r} b_-(k') + (1+\rho^2) b_0(k') + r b_+(k') \end{aligned} \tag{2.12}$$

where $b_-(k') = \overline{b_+(-k')}$, and

$$\begin{aligned} b_0(k') &= \widehat{s}(k') \int_0^{\infty} p_0(k) dk, \\ b_+(k') &= \int_0^{\infty} p_0(k) \widehat{s}(k'-k) dk \end{aligned}$$

As discussed before, in this model we do not consider the negative frequencies. The negative k' corresponds to the reflection from the back of the plate , and $p_0(k) = 0$ for $k < 0$ because the plate is illuminated only from one direction. For this region of interest, the term $b_-(k')$ is negligible compared to b_0 and b_+. As our equations are not exact, but only a first order approximation, we often omit b_-.

The reader familiar with the work of Nareid and Pedersen [307] might notice that the operator they derive is very similar but does not describe the decay due to the development, and it does not describe the phase change at the mirror and does not contain the term b_0. A model without b_0 describes a holographic plate more closely than a Lippmann plate, where reflection is due to varying index of refraction rather than reflective silver grains[3]. Nevertheless, the reflection-based model was considered by Lippmann [287] and Wiener [290]. We will see in Section 2.3 that both the b_0 term and the phase change have a significant effect on the plates we analyse.

---

[3] A fair comparison between the models is not possible without explaining details of the model by Nareid and Pedersen. For optically curious, let us note here that, to keep their model general, Nareid and Pedersen consider also complex-valued index of refraction; this index describes both reflection and refraction. However, the term b_0 can be ignored only when considering real-valued index of refraction; thus their model does not give the correct results in the reflection case.

Finally, we do not observe $\widetilde{u}(k', t)$ directly, rather its power spectrum given by

$$\widetilde{p}(k') = \left| \int_0^\infty p_0(k) \widehat{h}(k, k') dk \right|^2 . \tag{2.13}$$

2.2. Information Contained in the Lippmann Plate

In this section, we estimate how much information about the original spectrum can be inferred from the measured spectrum. We first study the channel capacity of a Lippmann plate, which we estimate using the number of degrees of freedom of the reflected spectrum. Then, we propose two methods for recovering the original spectrum. One of the methods is a custom alternating algorithm that we describe in Section 2.3 to estimate recorded spectra of the historical plates.

Channel Capacity

To estimate the channel capacity — or more precisely the upper bound on the channel capacity as we do not consider noise — we will make a few simplifications to Equation (2.13). First, we observe that by considering the power spectrum $\widetilde{p}$, instead of the amplitude $\widetilde{a}$, we lose the phase information. Therefore, the number of complex degrees of freedom of $\widetilde{a}$ corresponds to the number of real-valued degrees of freedom of $\widetilde{p}$.

We consider the amplitude given by Equation (2.12):

$$\widetilde{a}(k') = \overline{r} b_-(k') + (1 + \rho^2) b_0(k') + r b_+(k').$$

As discussed before, the term b_- can be omitted. For fixed parameters of the plate, the term b_0 has only one degree of freedom: the value of $\int_0^\infty p_0(k) dk$. Therefore, to estimate the number of degrees of freedom of $\widetilde{a}$, we study only

$$b_+(k') = \int_{-\infty}^\infty p_0(k) \widehat{s}(k' - k) dk,$$

where we can integrate from minus infinity because $p_0(k)$ is equal to zero for the negative k. Function b_+ is a convolution of p_0 with $\widehat{s}$. This means it is a multiplication in the spatial domain.

The inverse transform of $\widehat{s}$ has compact support (Equation (2.11)) and is not zero anywhere on $[0, 2Z]$. Therefore, $b_0(\widetilde{k})$ is band-pass and, from the canonical sampling theorem (Part I: Section 1.1), can be uniquely described with samples that are $\frac{2\pi}{2Z}$ apart. Assuming that we measure part of the reflected spectrum between wave numbers $k_{\max}$ and $k_{\min}$, e.g., the visible spectrum, the measured $\widetilde{a}$ contains around

$$\frac{Z}{\pi} \left(k_{\max} - k_{\min} \right)$$

complex degrees of freedom. We measure $\widetilde{p}$ instead of $\widetilde{a}$, thus we can distinguish up to$Z \left(k_{\max} - k_{\min} \right) / \pi$ *real-valued* degrees of freedom.

We could obtain the same estimate by assuming that p_0 has support in $[k_{\min}, k_{\max}]$. This is a reasonable assumption, because the Lippmann plates are sensitive to only certain wavelengths of light. Then, we can use the cardinal sampling theorem to observe that we need *spatial* samples of $\widetilde{p}$ to be $(k_{\max} - k_{\min})/(2\pi)$ apart. Therefore, in the interval $[0, 2Z]$, we can measure $\frac{Z}{\pi}(k_{\max} - k_{\min})$ *real-valued* samples. Here, we have real-valued samples because p_0 is real-valued.

To estimate the number of degrees of freedom of a real Lippmann plate, we assume that the visible spectrum spans wavelengths λ from 400–700 nm, the refractive index n of the plate is 1.5 and that the emulsion thickness Z lies between 4–10 µm. Using the relation that $k = n/\lambda$, we obtain the 13–32 sample range.[4]

Invertibility

In the previous subsection, we have seen that the reflected spectrum can be described with a finite number of samples. This means that for the Lippmann transform to be invertible, we need to restrict the original power spectrum p_0 to some finite dimensional family of functions. In this section, we propose such a restriction and propose to use *phase-lift* method to recover p_0 from the measured power spectrum $\widetilde{p}$ by Equation (2.13).

Using the observation that the Lippmann transform limits the degrees of freedom of the power spectrum both in the frequency and in the spatial dimension, we propose to restrict p_0 to be limited in space to $[0, 2Z]$. Similarly to bandlimited functions, p_0 can be written as

$$p_0(k) = \sum_{n \in \mathcal{N}} p_0(\kappa n)\widehat{\phi}(k - \kappa n), \tag{2.14}$$

where κ is the distance between the frequency samples, $\kappa = \frac{\pi}{Z}$, $\mathcal{N} \subset \mathbb{N}$ is the range of n such that $\kappa n \in [k_{\min}, k_{\max}]$, and finally ϕ is almost a sinc function[5]:

$$\widehat{\phi}(k) = e^{ikZ}\,\mathrm{sinc}(kZ); \text{ in spatial domain: } \phi(y) = \frac{1}{2}\mathbb{1}_{[0,2Z]}. \tag{2.15}$$

The shifts of $\widehat{\phi}$ by κn form a basis of waves limited in space to $[0, Z]$. The result of the Lippmann transform on $\widehat{\phi}(k - \kappa n)$ is equal to the kernel $\widehat{h}$ at $k = \kappa n$:

$$\int_0^\infty \widehat{\phi}(k - \kappa n)\widehat{h}(k, k')dk = \widehat{h}(\kappa n, k'). \tag{2.16}$$

We provide a detailed derivation at the end of this subsection.

For p_0 as defined in Equation (2.14), we can express the reflected wave using Equation (2.16) as follows:

$$\widetilde{a}(\widetilde{\kappa}) = \sum_{n \in \mathcal{N}} p_0(\kappa n)\widehat{h}(\kappa n, k')$$

---

[4] A reader familiar with our work might notice that our result here is half of the result in [270]. This is because in [270] we did not assume that only real-valued samples can be measured.

[5] For the definition of sinc see Part I Section 1.5.

We can write the equation in vector form. Let $n_{\min}$ and $n_{\max}$ be, respectively, the smallest and the largest element in $\mathcal{N}$. Then

$$\begin{aligned}\boldsymbol{p}_0 &= \begin{bmatrix} p_0(\kappa n_{\min}) & p_0(\kappa n_{\min+1}) & \cdots & p_0(\kappa n_{\max}) \end{bmatrix}^\top \\ \overline{\boldsymbol{h}} &= \begin{bmatrix} \widehat{h}(\kappa n_{\min}, k') & \widehat{h}(\kappa n_{\min+1}, k') & \cdots & \widehat{h}(\kappa n_{\max}, k') \end{bmatrix}.\end{aligned}$$

In this notation, we obtain

$$\widetilde{a}(k') = \boldsymbol{h}^\dagger(k')\boldsymbol{p}_0.$$

Assume now that we measure the reflected power spectrum $\widetilde{p}$ at points ${k'}_m$, $m \in [M]$ and obtain measurements $\widetilde{p}_m = \widetilde{p}({k'}_m)$. By setting $\boldsymbol{h}_m = \boldsymbol{h}({k'}_m)$ and by using the trace tricks described in Part I: Section 1.5 we obtain

$$\begin{aligned}\widetilde{p}_m &= |a({k'}_m)|^2 = |\boldsymbol{h}_m^\dagger \boldsymbol{p}|^2 \\ &= \operatorname{vec}(\boldsymbol{h}_m \boldsymbol{h}_m^\dagger)^\dagger \operatorname{vec}(\boldsymbol{p}\boldsymbol{p}^\dagger)\end{aligned}$$

This way, we formulated a linear system of equations in the new variable $\boldsymbol{P} = \boldsymbol{p}\boldsymbol{p}^\dagger$ that is a rank-one matrix. A reader familiar with Part I:Chapters 3 and 4 might notice similarity with measurements $\boldsymbol{f}\boldsymbol{f}^\top$. If we had enough independent vectors $\operatorname{vec}(\boldsymbol{h}_m \boldsymbol{h}_m^\dagger)$, we could recover $\boldsymbol{P}$ from this system of equations and factorise it to obtain $\boldsymbol{p}$. However, there are two difficulties.

First, as seen in the previous subsection, the reflected spectrum can have $D = 32$ degrees of freedom. To recover $\boldsymbol{P}$ as a full rank matrix, we would need $D(D+1)/2 = 528$ samples. This is feasible with spectrometer measurements that can produce 1000 spectral samples but not possible with the multi-spectral camera we have access to, as it produces 200 spectral samples.

The second difficulty is that the span of $\operatorname{vec}(\boldsymbol{h}_m \boldsymbol{h}_m^\dagger)$ is less than $D(D+1)/2$ dimensional, which we checked numerically. This means that no matter the number of samples, $\boldsymbol{P}$ cannot be calculated without additional assumptions.

Therefore, we propose two recovery methods. The first method is inspired by algorithms from irregular sampling [317]. We propose an iterative algorithm that exploits the prior information that the true original spectrum must be a valid power spectrum. The algorithm enforces this by alternatively modifying the current estimate so that it satisfies each of the following two properties: (i) when passed through the Lippmann transform it matches the measurements, and (ii) it is real, positive, and does not contain frequencies corresponding to depths outside the estimated thickness of the emulsion. To enforce the second property, a non-negative least-squares algorithm (NNLS) [318] is applied. We use this algorithm to generate all the figures in this chapter. Simulations have demonstrated that, in the noiseless case with an infinitely thick emulsion and known and invertible dye and development profiles, the proposed algorithm converges exactly to the original spectrum. Moreover, the algorithm is robust to noise, see [313].

The second method we propose is to formulate the recovery of the original spectrum as a semidefinite program:

$$
\begin{aligned}
\underset{\boldsymbol{P}}{\arg\min} & \sum_{m=1}^{M} \Big(\widetilde{p}_m - \operatorname{vec}(\boldsymbol{h}_m \boldsymbol{h}_m^{\dagger})^{\dagger} \operatorname{vec}(\boldsymbol{P})\Big) \\
& \boldsymbol{P} \succcurlyeq 0 \\
& p_{jl} \geq 0 \text{ for } j, l \in [D]
\end{aligned}
$$

Thus far, we have assumed that all the parameters of the plate, in particular the depth Z and decay τ, are known. In practice, however, we have only some estimates of these quantities. In the case of the historical Lippmann plates that we study in Section 2.3, we cannot cut the plates open to measure them, hence we have to incorporate their estimates into the algorithm. For the iterative algorithm, this can be done by adding one extra step to estimate τ and Z. In the case of semidefinite relaxation, we would have to run a semidefinite program for different τ and Z.

Proof of Equation (2.16)

Observe that Equation (2.16) is a convolution between $\widehat{\phi}$ and $\widehat{h}_{k'} := \widehat{h}(\cdot, k')$. Therefore, in the spatial domain it is multiplication:

$$
e^{-i\kappa n y} \phi(y) \cdot h_{k'}(y),
$$

where the factor $e^{-i\kappa n y}$ appears because of the shift of $\widehat{\phi}$ by κn. But from the definition of $\widehat{h}$, $h_{k'}(y)$ also has support only in $[0, 2Z]$, and the term $\phi(y)$ does not have an effect:

$$
e^{-i\kappa n y} \phi(y) \cdot h_{k'}(y) = e^{-i\kappa n y} h_{k'}(y)
$$

Moving back to the Fourier domain, we obtain $\widehat{h_{k'}}$ shifted by $-\kappa n$, and $\widehat{h}_{k'}(k - \kappa n) = \widehat{h}(k - \kappa n, k')$; this proves Equation (2.16). ■

2.3. Model Verification

We now use our mathematical model to study the effects and artefacts generated by Lippmann's procedure. If not stated otherwise, we assume that $\tau = 0$, i.e., we do not take the development effects into account.

To gain some intuition, we refer to Figure 2.15a, which shows monochromatic and Gaussian spectra, and Figure 2.15b-d, which illustrate their reproduced spectra as predicted by Equation (2.13) for plates of differing thicknesses. We notice that, although the overall shape of the synthesised spectra is relatively accurate, it contains several artefacts. We identify four effects that are worth further investigation.

1. First, recall that we do not observe $\widetilde{a}(k')$ directly, rather its power spectrum $\widetilde{p} = |\widetilde{a}(k')|$. As a result, the filtered power spectrum p_0 is squared; this increases

the contrast in the synthesised colours and explains why Lippmann plates tend to exhibit a wider tonal range, as well as strong and bold colours.

2. Second, $\widehat{s}$ acts as a low-pass filter on the spectrum. This effect confirms early experiments conducted by Ives [292] who reported that the thickness of the plate controls the spectral resolution. Additionally, the imaginary part of $\widehat{s}$ distorts the overall shape of the spectrum: As we observe the squared magnitude of the analytic signal $\widetilde{a}$, its real and imaginary parts are blended together; this results in broader spectra.
3. Third, the term $p_c(k')$ contains some strong oscillations that leak into the visible spectrum, as illustrated in Figure 2.15b-c. As seen in Figure 2.2, for Gaussian waves the interference patterns tend to stabilize around a value proportional to the energy of $p_0(k)$. The relative strength of the oscillations is influenced by this value: a signal with its energy spread over the entire visible spectrum will have more pronounced relative oscillations.
4. Finally, the heavy tails of the term $p_c(k')$ and in particular its imaginary component also influence the visible spectrum. Perhaps surprisingly, this effect depends on the reflection coefficient r.

To the best of our knowledge, points 2 to 4 have never been documented. We will now investigate these concepts in more detail.

Plate Thickness

As we have seen in Section 2.1, the recording of the interference pattern in the plate is a generalized Fourier cosine transform of the original spectrum. As the lower frequencies (of the spectrum) are located at the entrance of the plate and higher frequencies deeper in the plate, the thickness of the plate has a low-pass effect on the synthesised spectrum. Not only are the high frequencies clipped in the synthesised spectrum, but also the sharp cutoff at the end of the plate results in ringing artefacts due to Gibbs phenomenon, which can be observed in Figure 2.15.

Another direct consequence of the finite thickness of the plate is the frequency of oscillations of the middle component $p_c(k)$.

As shown in Figure 2.7, we performed a hyperspectral acquisition of a Lippmann plate. The plate was created by photographer Filipe Alves with a home-made albumen emulsion. It is clear that the spectra exhibit the oscillations predicted by our reflection-based theoretical model. Furthermore, we can approximate the thickness of the plate from these oscillations. The spectra from Figure 2.7 present 8 oscillations in the visible frequency range Δk, hence they have a period of $T = \frac{\Delta k}{8}$. Comparing this value with the expression for $\widehat{s}(k)$, we deduce that $\frac{2\pi}{T} = \frac{2Z}{c}$, or $Z \approx 3.7\ \mu$m. This value is in line with the reported plate's thickness.

Development

To underline the effect of the development, we propose to look directly at the density of developed silver inside a plate. Today, we do not have to expand the gelatin by swelling

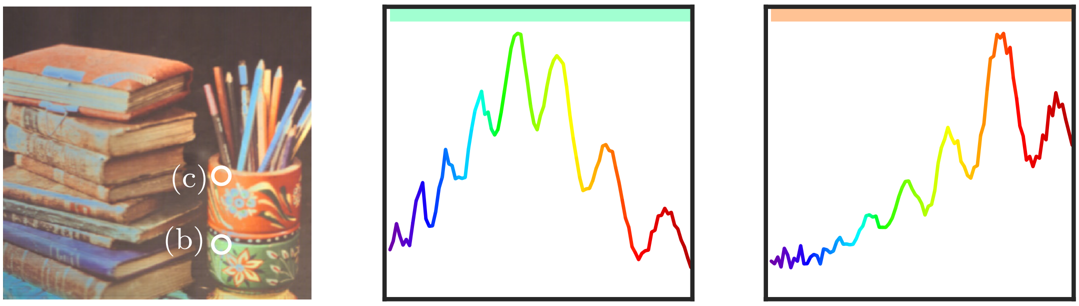


Figure 2.7.: Hyperspectral acquisition of a Lippmann plate. Left: close-up of the plate in RGB colours. Middle: a green spectrum. Right: a red-orange spectrum measured on the pot. The spatial location of the spectra displayed is shown on the left, and the uniform band on top of each spectrum plots represents the corresponding RGB colour.

it (see Section 1.3) to observe the interference pattern recorded in the plate. We can use electron microscopy to obtain sub-wavelength resolution imaging. Electron microscopes cannot distinguish between undeveloped silver halide and metallic silver particles. To observe interference patterns, it is necessary to fix the plate in order to wash out the remaining silver halide.

Figure 2.8 shows an example of the density of the metallic silver particles that results from the exposure of a plate to a 531 nm laser. The interference pattern is clearly noticeable in the micrograph, as well as when averaging it across pixels located at the same depth. We can also observe the decay due to development in the averaged pattern.

Reflection Coefficient

Another parameter that alters the colour rendition is the reflection coefficient $r = \rho e^{i\theta}$. We voluntarily described our model as a function of r and did not restrict it to $r = 1$ or $r = -1$, as is commonly done in the literature. We now investigate the effects of changing it.

In Lippmann photography, there are essentially two mediums that are used to create interferences: mercury and air. For an interface between glass and mercury, the parameters of the reflection are $\rho = 0.71$ and $\theta = 148°$. And for an interface between glass and air, the reflection coefficient is $r = 0.2$, or equivalently, $\rho = 0.2$ and $\theta = 0$. It is evident that different values of r will generate different interference patterns inside the plate. For example, with $r = 1$ a peak is formed at the entrance ($z = 0$) of the plate, whereas with $r = -1$, we have a node. We are interested how these different values influence the synthesised spectra. The side components p_s and $\overline{p_s}$ are modulated with r and r^*, hence they undergo a rotation in the complex plane that is determined by the value of θ. Whereas, the centre component p_c is modulated by $1 + \rho^2$ and is never rotated. This means that these two kinds of components interfere differently with each other for different values of r.

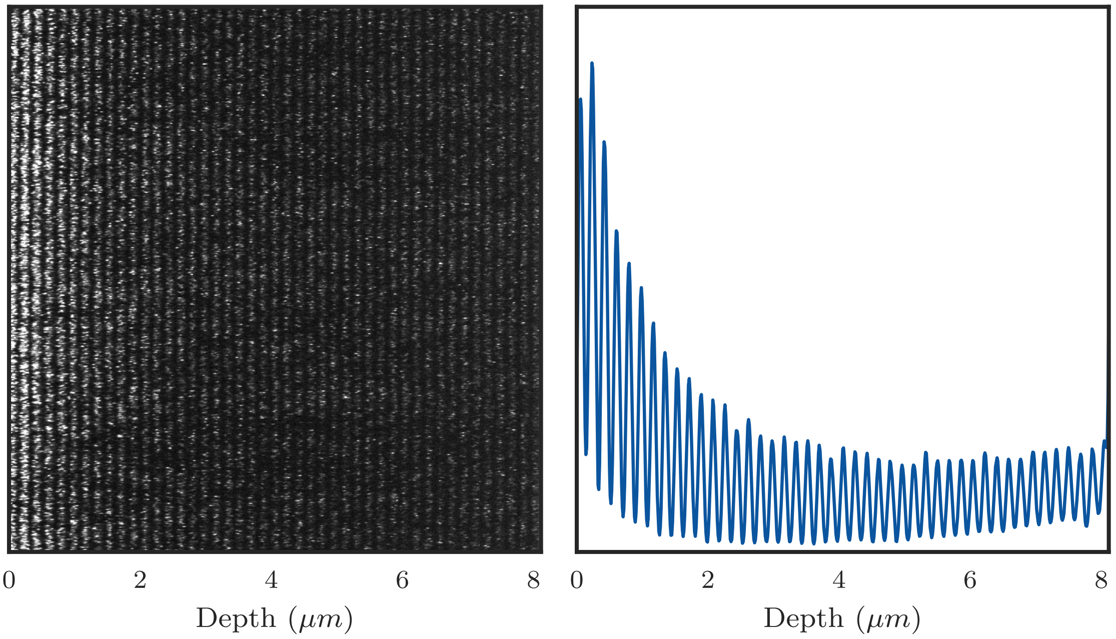


Figure 2.8.: Capture of the interference patterns. Left: electron microscope image of a slice of a Lippmann plate created with a 531 nm laser. After fixing, we can observe interference patterns created by the varying density of the metallic silver grains. Right: these patterns are clearly visible after averaging the columns.

Figure 2.10 shows a *skewing* effect on the spectrum. On one hand, for $r = -1$, the imaginary parts of $\widehat{s}(k')$ and $-\widehat{s}(k' - k)$ add up constructively for red colours and destructively for blue colours, as seen in Figure 2.10a. On the other hand, for $r = 1$, Figure 2.10b shows the opposite phenomenon, namely that the spectrum is skewed such that it favours blue colours over red colours. In practice, when the reflection is due to the interface with air, we have $r = 0.2$, hence we should expect stronger blues, to the detriment of the reds as $\theta < \pi/2$. With mercury, we have $\rho = 0.71$ and $\theta = 148°$, hence red colours should predominate as $\theta > \pi/2$. Given a Gaussian spectrum centred around a specific wavelength, Figure 2.11 illustrates the expected reproduced colours for different values of ρ and θ. It suggests that bright reds are more challenging to reproduce with air and that blues are harder to represent with mercury. We also notice a shift of the reflected spectra: for instance, greens should appear blueish with air and yellowish with mercury. Moreover, a smaller value of $|\rho|$ leads to darker colours; this explains why, in general, plates made with mercury appear brighter than those made with air.

For a final remark, note that the strength of the skewing effect is correlated with the spread of the energy of the original spectrum. A spectrum that is concentrated around a single frequency will have a relatively weaker centre component and, as a consequence, the skewing effect is less noticeable. In the extreme case where the original spectrum is a Dirac, the tails of the centre component are not strong enough to influence the visible spectrum, as illustrated in Figure 2.15c-d.

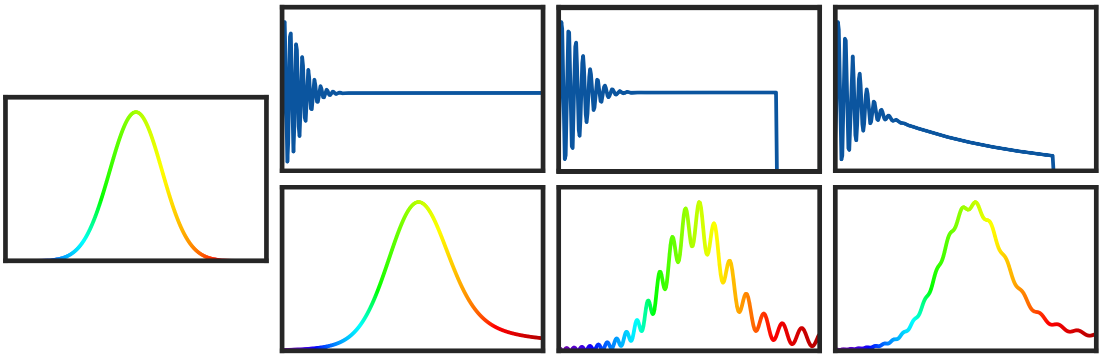


Figure 2.9.: The effect of the plate's thickness and development on spectrum reproduction. Simulation of different silver density profiles (top) and the corresponding reflected spectra (bottom). From left to right: first, the original spectrum; second, infinite plate with homogeneous development: the only effect on the spectrum is the skewing described; third, a finite plate with homogeneous development: pronounced Gibbs ripples are introduced. Finally, a finite plate with depth-decaying development: Gibbs ripples are damped.

Monochromatic Spectrum To analyse the skewing effect, we decomposed white light through a prism. All visible wavelengths are represented and the light of the spectrum obtained from this decomposition is locally monochromatic. Under such light, we exposed two Lippmann plates: one using mercury as a reflector and one using the interface with air. We then developed these two plates by using standard techniques and photographed them. For reference, we also photographed the light spectrum coming from the prism: the original spectrum, along with the spectra of the two plates, is shown in Figure 2.11 (right).

To quantify the accuracy of our model, we also ran a simulation of this experiment using our analytical model, illustrated in Figure 2.11 (left). For it to be as close as possible to the experiment, we set the thickness of the plate to $Z = 7$ μm and combined it with a development window $c(z) = 1 - \operatorname{erf}(3z/Z)$.

We observe that the measured colours closely resemble those from our simulations. In particular, the purple colours are weaker in plates made with mercury, and the plates made with air struggle at reproducing bright reds. Also, the reproduced spectra are subject to some shifts: for instance, the green colours tend to have a stronger red tint when mercury is used and a more dominant blue component with air.

Gaussian-Shaped Spectrum We run a second experiment by placing Gaussian-shaped bandpass filters having a bandwidth of approximately 10 nm in front of a halogen light source. As before, we create two plates: one with a mercury reflector and one with air reflector. The photographs of the resulting plates, along with the colour of the filters, are shown in Figure 2.16a. For this experiment, we also acquired hyperspectral measurements of the original spectra, as well as the spectra reflected by these plates (see Figure 2.16b-f). We see that the reproduced colours are in agreement with our predictions. Indeed, the

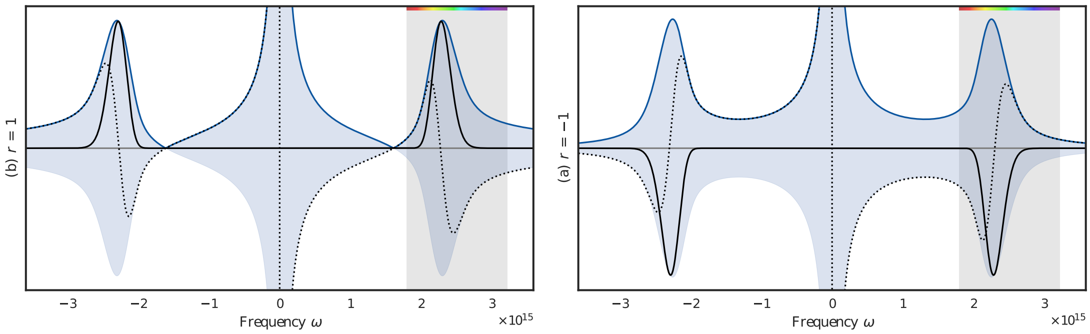


Figure 2.10.: Skewing effect on the spectrum reflected by an infinite plate that recorded the Gaussian-shaped spectrum from Figure 2.15 for $r = -1$ (left) and $r = 1$ (right). The black lines show the complex wave (the continuous curve is the real part and the dotted curve the imaginary part) and the blue lines show its envelope (which correspond to the square root of the replayed intensity). We observe that the reflected spectrum for $r = -1$ exhibits a negative skew and that the reflected spectrum for $r = 1$ a positive skew.

spectra of the plates realized with a mercury reflector exhibit a negative skew, whereas those made with an air reflector have a positive skew. Moreover, the synthesised spectra appear broader than the original one; the main reason is that the imaginary part of the reflected wave combines with the real part in the squared magnitude operation, resulting in larger spectra.

Strength of Oscillations Regarding the oscillations, we notice two effects: First, they are not significant, which suggests that they are damped by a development curve strongly decaying with the depth. Second, their amplitude is slightly larger in the plates made with air. This latter effect is studied in greater detail in the following paragraph.

Besides skewing the reflected spectrum, the reflection coefficient also has some influence on the strength of the oscillations. When $\rho < 1$, the interference patterns take the form of partial standing waves. The baseline term in Equation (2.5) is multiplied by the value $1 + \rho^2$ and the oscillating part by the value 2ρ. Subsequently, the baseline becomes relatively stronger than the relevant part of the signal as ρ becomes smaller. This baseline is then directly mapped to the centre component $p_c(k)$ in the reflected spectrum. As a result, the spectrum contains stronger oscillations for smaller values of ρ. This effect is illustrated in Figure 2.12. It also explains why the oscillations are more intense in plates made with air reflectors than in those made with mercury reflectors.

Reflection vs. Refraction

In Section 2.1, we have mentioned that an alternative model that does not contain the term b_0 was proposed [287, 290, 307]. Here, we illustrate the differences between the models with simulation and an experiment. Simulated reflected spectra, according to the reflection-based and refraction-based models, are shown in Figure 2.13. The refraction-

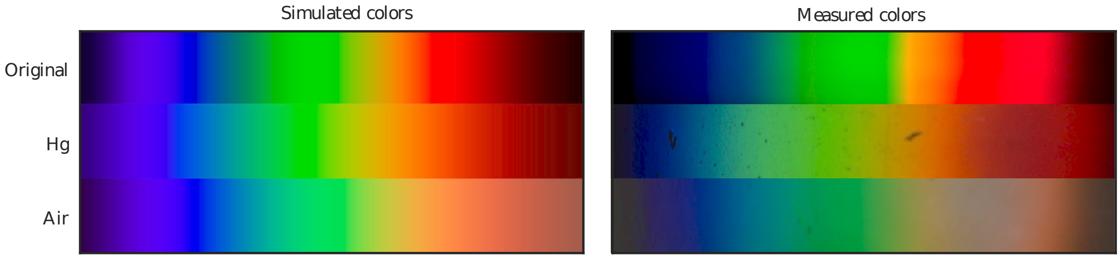


Figure 2.11.: Colour reproduction experiment: simulated colours on the left, and measured colours on the right. We can observe the predicted skewing effect on the spectrum. To measure the colours, we first decomposed white light into monochromatic colours with a prism and recorded Lippmann photographs by using mercury (Hg) and air reflectors. The resulting reflected spectra and the original spectrum were then recorded using a digital camera. The black speckles in the plate with mercury are due to tiny particles of dust floating on the mercury surface. A version of this figure appeared in [270].

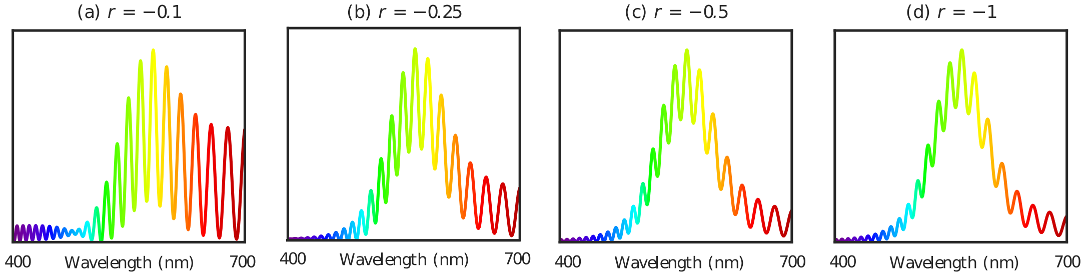


Figure 2.12.: Reflected spectra based on the Gaussian spectrum from Figure 2.2, for $r = -0.1, -0.25, -0.5, -1$. We observe that the oscillations are proportionally more important in spectra with lower intensities. The spectra have been re-scaled such that they appear similar, but in practice the proportion of reflected light is higher for higher absolute values of r.

based reflection does not contain the effect of the term b_0: the oscillations and the skewing effect.

The spectra reflected from physical plates are shown in Figure 2.14. One plate was developed using Lippmann's method (left) and the other one was developed and bleached with holographic recipes (right). We notice that the spectrum on the left clearly exhibits the skewing effect and moderate oscillations; on the contrary, the spectrum on the right is relatively symmetric and does not present any of these effects.

Reconstructing Recorded Spectra

The main practical challenge comes from the dye profile (also called spectral sensitivity) of the plate. In particular, our modelling up to this point has assumed that the plate is isochromatic, i.e.,the dyes are such that the plate is equally sensitive to all wavelengths.

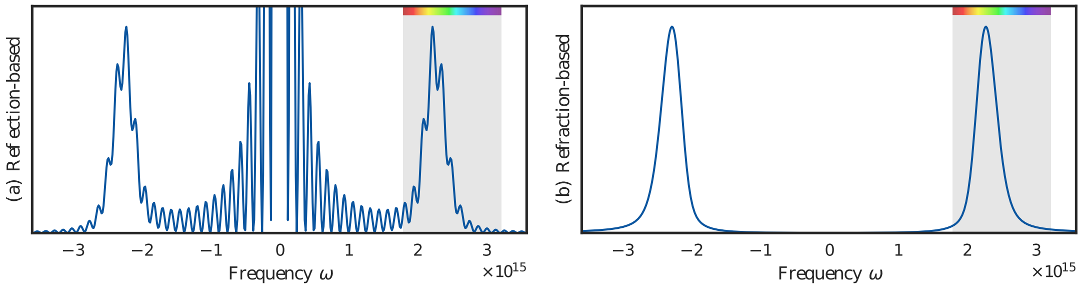


Figure 2.13.: Simulation of the Gaussian spectrum from Figure 2.2 reflected by a plate of width $Z = 5\ \mu$m, $r = -1$ and with a constant illumination according to the reflection-based model (left) and the refraction-based model (right).

When looking at colour reproduction on real plates in Figure 2.11, we saw that this is not the case with dark patches appearing for wavelengths greater than 600 nm. Fortunately, the analysis model can be extended to incorporate a general dye profile by simply replacing $P(\omega)$ by the product of the dye profile and the input power spectrum; however, to recover the original spectrum, the dye profile should be known and invertible.

For example, Figure 2.17a shows a Lippmann photograph of a colour checker. As we made this photograph, we were able to measure the original spectrum of the scene, the reflected spectrum from the Lippmann plate and the dye response of the photographic emulsion. Measurements of the original and measured spectra, as well as the estimated recovery returned by our inversion algorithm, are shown in the figure. We see that the reproduction is accurate, except for regions where the dye response is weak.

Finally, in Figure 2.17b, we present the measured and inverted spectra of four points of two historical plates. The first was made by Gabriel Lippmann between 1891 and 1899 in Saas Fee, Switzerland. Hence, it contains century-old measurements of the spectrum of a natural scene. The second was made by Richard Neuhauss in 1899 and is of a parrot.

Looking at the measured spectra of these historical plates, the lack of oscillations immediately stands out. From the analysis presented in Figure 2.9, we know that this can be explained by a smoother decay to zero in the development window function $W(z)$. This could be due to a slightly different development or an aging effect that reduces the density of silver at both boundaries of the emulsion.

Unfortunately, it is impossible to measure the dye response of these plates because the dyes are washed out of the plates as part of the chemical processing. Furthermore, the dye recipes were, in all probability, tuned to reproduced accurate and faithful colours, rather than to achieve perfectly isochromatic solutions. Another aspect that we do not know about is whether the photographers used colour filters to improve the colour rendition. For instance in [311], which is one of the few contemporary publications on the topic, a photo of the apparatus used to capture Lippmann photographs (with an air-based reflection) clearly shows a warming filter in front of the lens. These unknown factors are unfortunately impossible to compensate for in our algorithm. Finally, the plates most probably were captured with mercury for a reflective medium, as they were created before

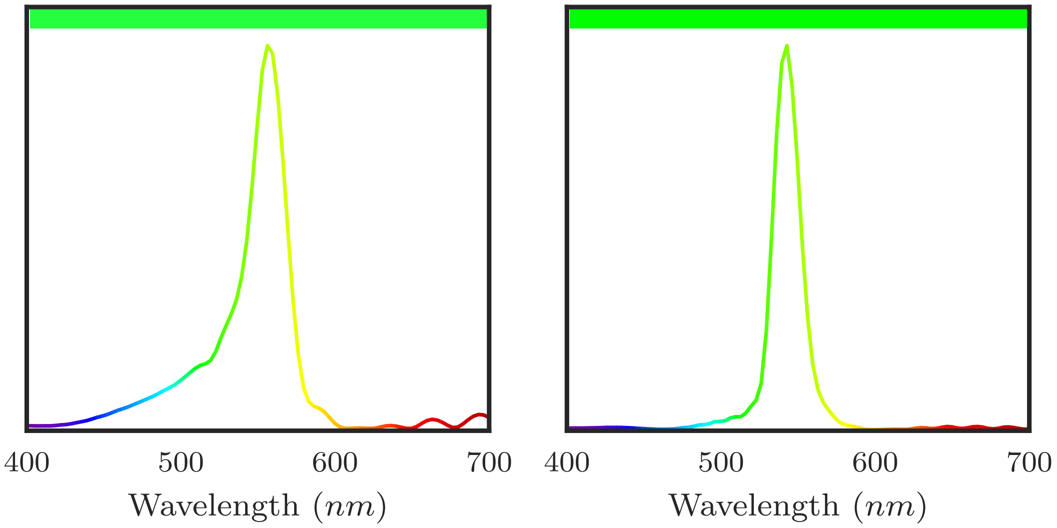


Figure 2.14.: Two different development procedures and their effects. Left: the Lumière developer results in spectra that can be explained with the reflection-based model. Right: holographic recipes produce spectra that can be inferred from the refraction-based model.

E. Rothé proposed the use of air as a viable alternative [291]. However, we cannot be certain about this.

Despite these limitations, we present spectrum recovery results in Figure 2.17b, assuming uniform spectral sensitivity and a mercury reflective interface. Qualitatively, the recovered colours appear too blue, which hints that either the emulsion was not isochromatic, or that the cooling filters were used during the capture. Another, less-likely, explanation is that the plates were made with air reflectors, even though they predate Rothé's publication.

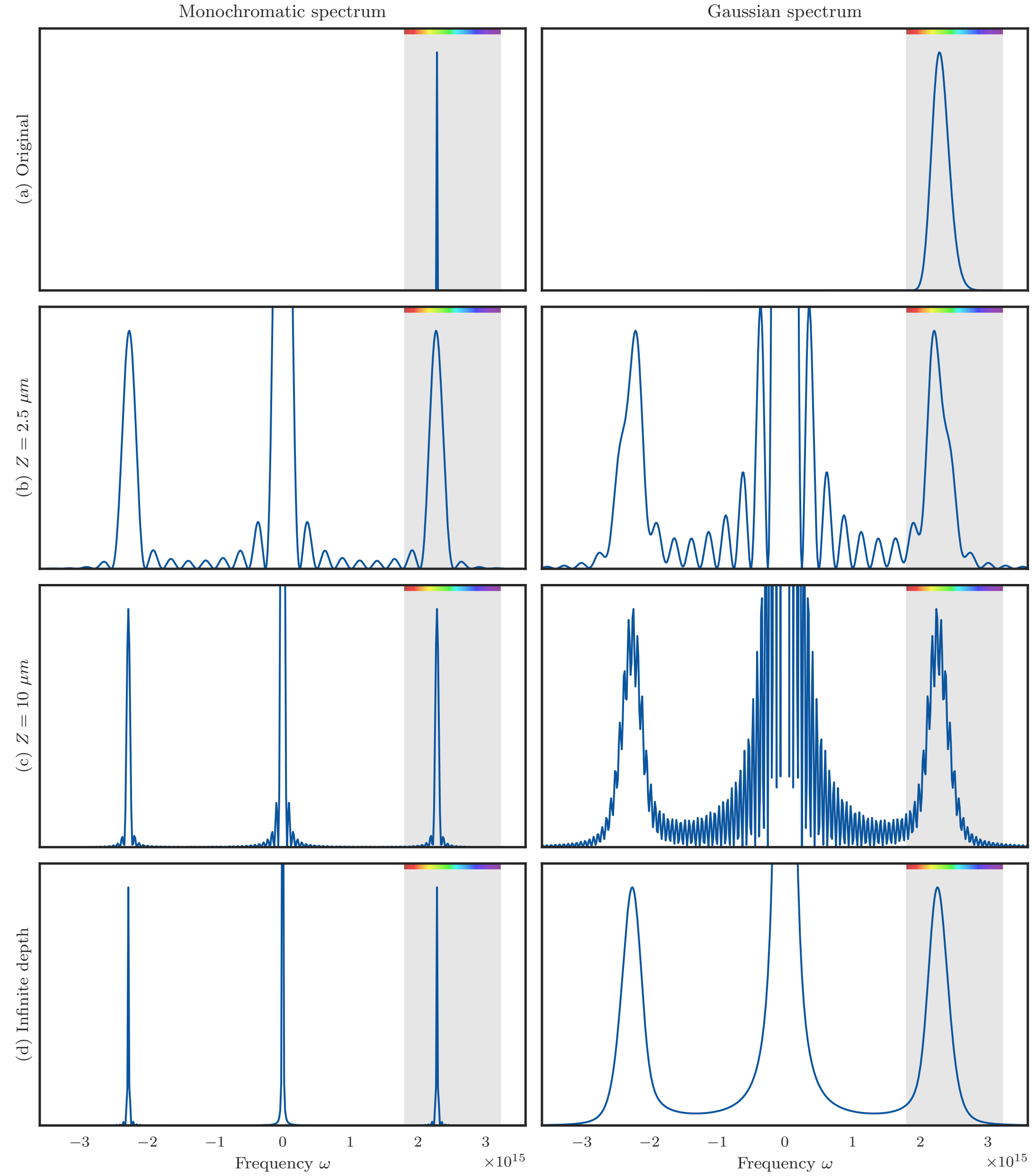


Figure 2.15.: Influence of the plate's thickness on the reflected spectrum. Given (a) a spectrum recorded in a plate of thickness Z, we show the reflected spectrum for (b) $Z = 2.5\ \mu$m, (c) $Z = 10\ \mu$m, and (d) Z approaching infinity. The thickness controls the sharpness of the synthesised spectrum (this is especially noticeable with monochromatic light that has high frequency components) and the frequency $Z/(\pi c)$ of the ringing artefacts. Shaded areas represent visible light.

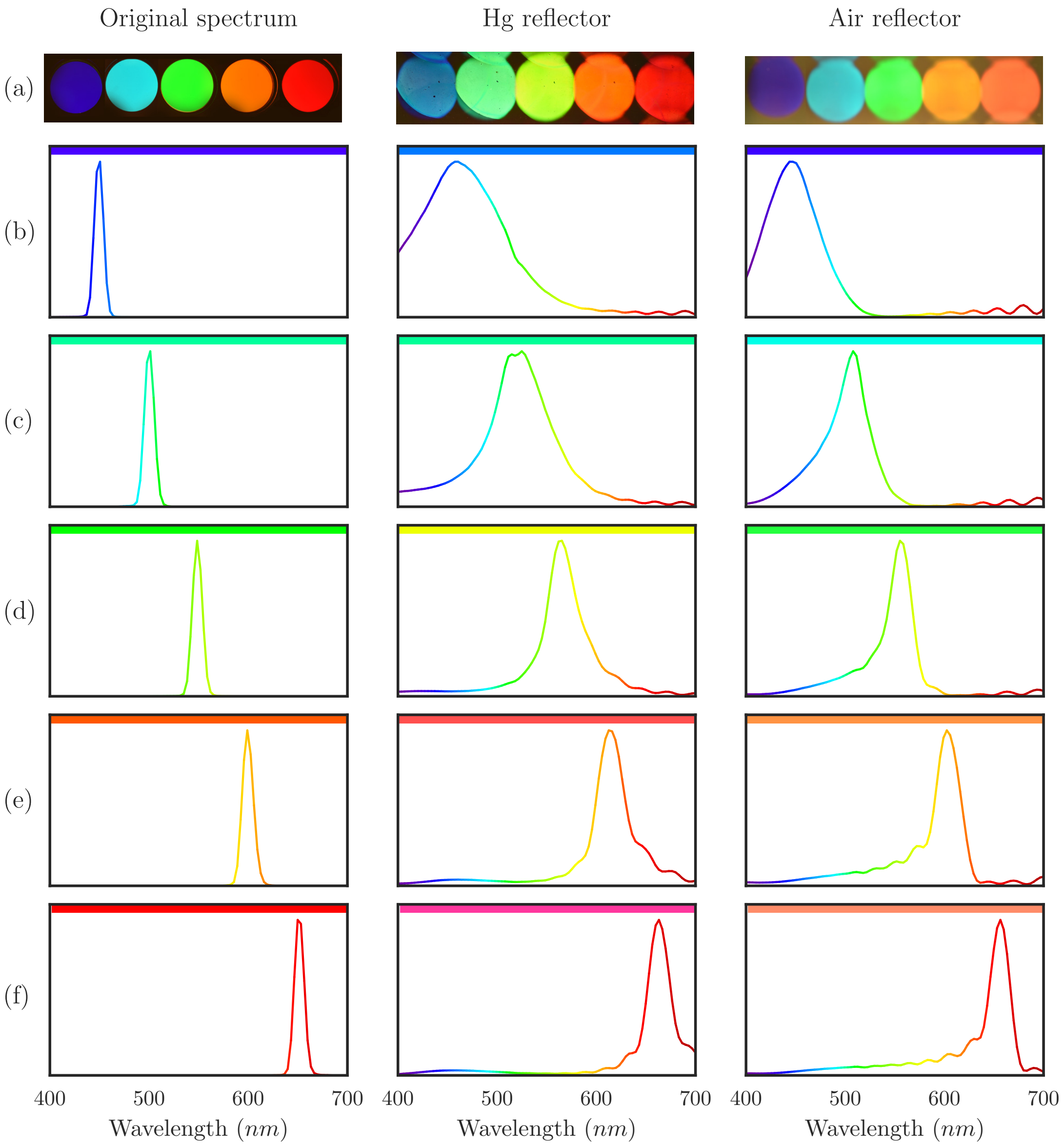
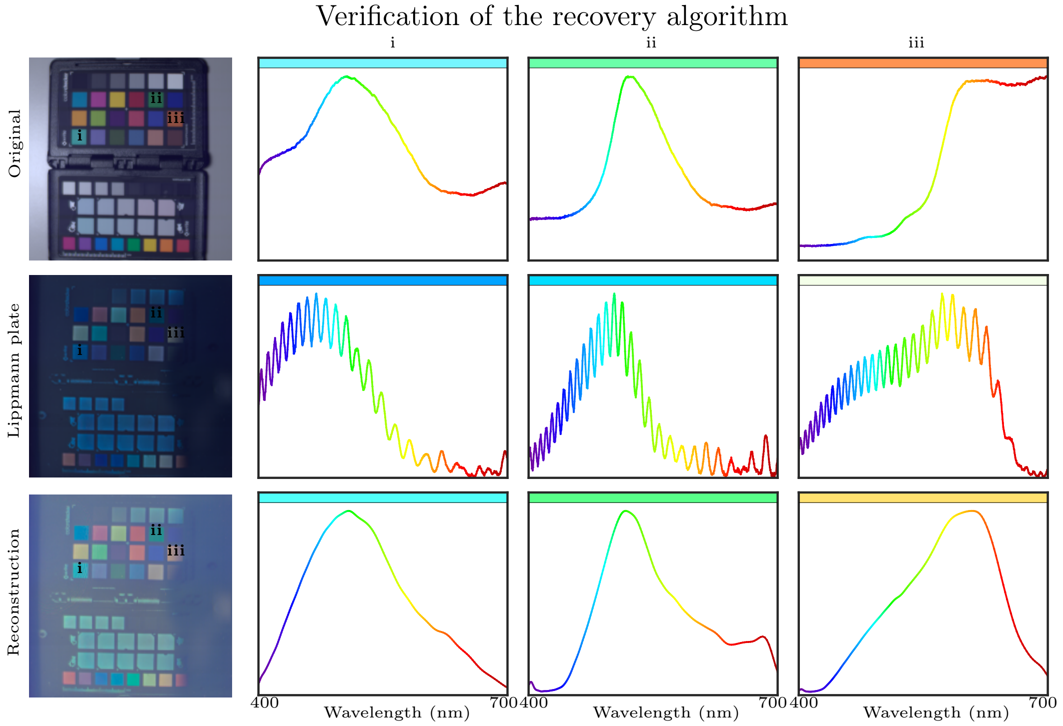
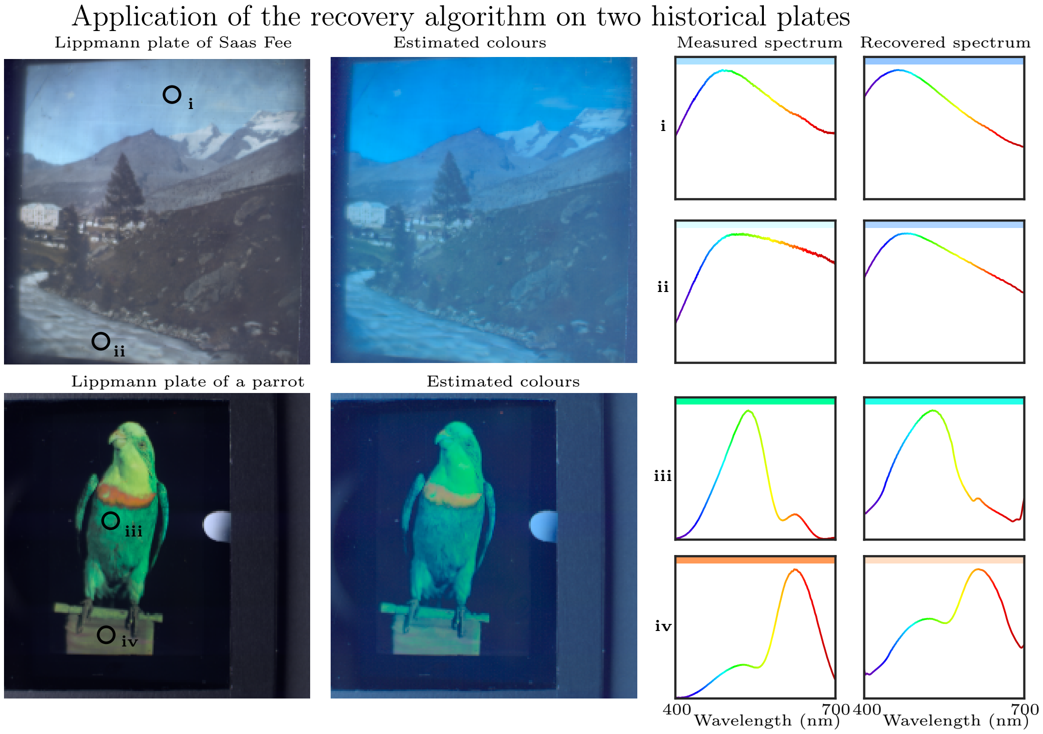


Figure 2.16.: Skewing effect on Gaussian-shaped spectra: (a) the left column shows the five filters used to expose the plates, whereas the middle and right columns show the resulting plates after development. Additionally, we show the spectral data corresponding to each of these five bands: (b) 450 nm, (c) 500 nm, (d) 550 nm, (e) 600 nm, and (f) 650 nm.



Verification of the recovery algorithm



Application of the recovery algorithm on two historical plates

Figure 2.17.: Spectrum recovery; figure from [270]. Top: the recovery algorithm is verified using self-made plates. Left: photographs of the colour checker and of the Lippmann plate of the colour checker. Right: the original, measured and recovered spectra. Bottom: the recovery algorithm is applied to two historical plates. Left: photographs of the plates. Right: the measured and recovered spectra.

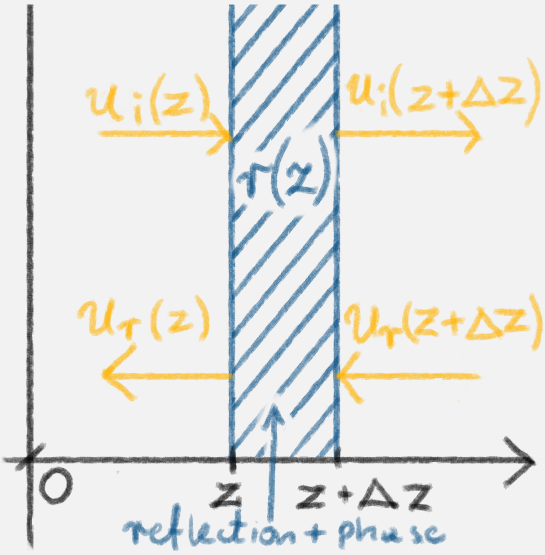
2.4. Conclusions

A thorough inspection of Lippmann's Nobel-Prize winning paper [287] enabled us to elucidate a misconception that went unnoticed for more than a century regarding the perfect reconstruction of the synthesised spectrum with infinite plates. Furthermore, our formulation provides a clear explanation of the colour reproduction of Lippmann plates. This enabled us to characterize and describe various effects, such as the role of a low-pass filter effect due to the finite thickness of the plate, the skewing effect induced by the choice of reflector, and the presence of oscillations in the synthesised spectra.

Want to know more? 2.1: Exact Reflection

In this box, we formulate the differential equations exactly and show how to formally obtain the same approximation we made in Section 2.1.

As in Section 2.1, we assume that we have an incoming wave $u_i(k, z)$ travelling towards positive z and a reflected wave $u_r(k, z)$ travelling towards negative z. We consider the reflection of a thin layer of thickness Δz, as illustrated below:



The reflected and transmitted waves have to satisfy the following equations:

$$\begin{aligned} e^{ik\Delta z}u_i(z+\Delta z) &= (1-r(z)\Delta z)u_i(z) + r(z)\Delta z u_r(z+\Delta z) \\ u_r(z) &= r(z)u_i(z)\Delta z + e^{-ik\Delta z}(1-r(z)\Delta z)u_r(z+\Delta z) \end{aligned}$$

The factor $e^{ik\Delta z}$ comes from propagation of the wave through the depth Δz. By multiplying the first equation by e^{ikz} and the second equation by e^{-ikz} and moving the terms that do not contain r as a factor to the left, we obtain

$$\begin{aligned} e^{ik(z+\Delta z)}u_i(z+\Delta z) - e^{ikz}u_i(z) &= -e^{ikz}r(z)u_i(z)\Delta z + e^{ik(z+\Delta z)}r(z)u_r(z+\Delta z)\Delta z \\ e^{-ikz}u_r(z) - e^{-ik(z+\Delta z)}u_r(z+\Delta z) &= e^{-ikz}r(z)u_i(z)\Delta z - e^{-ik(z+\Delta z)}r(z)u_r(z+\Delta z)\Delta z \end{aligned}$$

By dividing by Δz and taking the limit $\Delta z \to 0$, we obtain a system of differential equations:

$$\begin{aligned} e^{-ikz}\frac{\mathrm{d}}{\mathrm{d}z}\Big(e^{ikz}u_i(z)\Big) &= -r(z)u_i(z) + r(z)u_r(z) \\ e^{ikz}\frac{\mathrm{d}}{\mathrm{d}z}\Big(e^{-ikz}u_r(z)\Big) &= -r(z)u_i(z) + r(z)u_r(z). \end{aligned}$$

This system of equations is difficult to solve analytically for arbitrary $r(z)$. Here, we assume that $r(z) \ll 1$, and we apply the first-order approximation, i.e., from each equation, we preserve only terms with the lowest power of $r(z)$. In the first equation, as u_i is the incoming wave, we assume it is of order of 1. As the right side of this equation behaves similarly to r, it is negligible compared to u_i. We obtain that $e^{ikz}u_i$ is constant. In the second equation, u_r behaves similarly to r, hence we can discard only $r(z)u_r(z)$:

$$e^{ikz}\frac{\mathrm{d}}{\mathrm{d}z}\Big(e^{-ikz}u_r(z)\Big) = -r(z)ae^{-ikz}$$

which we can integrate and obtain the same result as in Section 2.1:

$$\Big[e^{-ikz}u_r(z)\Big]_0^Z = 0 - u_r(0) = -\int_0^Z r(z)ae^{-i2kz}dz,$$

where $u_t(Z) = 0$ because we assume that there is no light going trough back of the plate; it is painted black.

Using this method, we could consider keeping not only the first term in the powers of r, but also the second. Then, the upper equation would have to be solved first, but the energy of the incoming wave would no longer be constant.

Finally, the linear differential equations we discuss in this box can be solved numerically using any standard solver.

Chapter 3.

Future

Introduction

As we discussed in Chapter 1, Lippmann photography never became a mainstream photographic technique. The reasons for this include the long exposure time needed to record the plates, the variability of the photographic process, the difficulty in producing plates of sufficient quality, and the impossibility to duplicate plates. Despite this, we believe that Lippmann photography has the potential to inspire innovation in digital technologies such as digital displays, multi-spectral cameras, colour reproduction and, last but not least, data storage.

For example, consider electronic paper displays, i.e., displays that reflect light rather than emit it. They use much less energy than light-emitting displays and are less tiring for the eyes; they are very similar to paper in terms of fatigue [320]. At the time of this writing, all colour electronic paper available on the market uses the three-colour method. As electronic paper pixels are large, the reduced colour resolution (due to the need for 3 pixels to display a single colour) is much more visible than on light-emitting displays. There are prototypes of interference-based displays that can display any monochromatic colour with a single pixel [321]. However, there are colours, e.g., magenta, that cannot be displayed with a monochromatic light. Lippmann photography is capable of polychromatic reflection, hence it could inspire new types of electronic paper displays.

The technology for multi-spectral cameras is more mature than for colour electronic paper. Nevertheless, we think that a Lippmann inspired multi-spectral camera, i.e., a Michelson interferometer with an attached imaging sensor, can adapt more easily to the lighting conditions than existing technologies; it can trade exposure time for the "depth of the plate", which affects the spectral resolution. A prototype of such camera, built in LCAV, is depicted on the left in Figure 3.1.

Contributions: the research proposal described in this chapter arose during discussions between the author of this thesis (MP), Adam Scholefield, Martin Vetterli, Yves Bellouard (YB), Gilles Baechler, and Arnaud Latty [319]. The preliminary results in this chapter would not have been possible without the femto-laser expertise of YB, as well as Julien Gateau, Ruben Ricca and Yacine Hedef and the help of other members of the GALATEA laboratory. MP wrote and ran simulations and wrote this chapter.

The code used for simulations is available at `https://github.com/LCAV/femto-lippmann`.

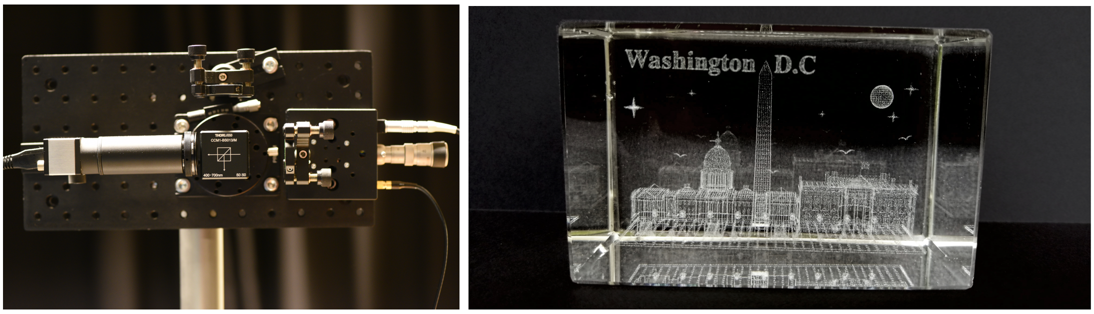


Figure 3.1.: Left: a prototype of a Lippmann-inspired multi-spectral camera. The beam splitter is visible in the centre, mirrors are to the right and at the top. Right: a bubblegram, also known as laser crystal, 3D crystal engraving or vitrography: a 3D image created using subsurface laser engraving. During the engraving, focused laser beams make small cracks in the glass. The cracks scatter light, so they seem white. Currently there are no mulit-colour bubblegrams.

In the remainder of this chapter, we discuss two applications: colour reproduction and data storage. We propose to use femto-second lasers to create durable colours inside glass; this results in something similar to a bubblegram (Figure 3.1, right) but in colour and potentially with smaller features. Such a recording could have multiple applications, including in security and archival data storage. As we have seen in Section 2.2, the Lippmann plates can store up to 30 spectral samples in the depth of the plate, hence the information could be encoded in colour. The data storage application might be the most interesting because the growth of data production exceeds our current storage capability and new storage technologies are needed.

In terms of security applications, the colour images we propose to embed in glass require specialised equipment to write. Verifying their authenticity, e.g., by checking that a pattern invisible to human eye is present, requires less specialised equipment, such as a multi-spectral camera or a spectrometer. As a bonus, this method would also enable us to copy a Lippmann plate, for example, to make reproductions of the original Lippmann plates; this is not possible with current technologies.

In this chapter, we imagine how such a colour printing technology could work. We begin with an overview in Section 3.1 of current long-term storage technologies and their capacities. We focus on glass-based storage as the most relevant for our work. In Section 3.2, we describe our idea for Femto-Lippmann storage [319] that could double as a colour bubblegram. In Section 3.3, we describe how to adjust the mathematical model from Chapter 2 for media with varying *refractivity* rather than *reflectivity*. Finally, in Section 3.4, we show our preliminary results that can guide development of Femto-Lippmann printing.

3.1. Data Storage

What does the future of data storage look like? In our homes, solid state drives (SSD) are replacing hard disk dries (HDD). At the same time, the demand for HDD from data centres is growing [322]. This is because the way we store data is changing. We do not own movies anymore, we stream them. We backup our data in the cloud. Companies produce increasingly more data, partly due to regulations that require them to keep the documentation for longer and partly to run deep learning algorithms on the gathered data.

SSDs are not suitable for data centres because they leak the charge and need to be regularly replaced. Furthermore, flash memory, on which they are based, seems to be close to its maximum capacity per inch [323, 324]. The capacity of HDDs is growing but, despite new promising technologies such as microwave-assisted magnetic recording, it might also be at its limit. The size of magnetic grains in which the information is encoded is reaching the so-called superparamagnetic limit; i.e., the size at which magnetisation can randomly flip due to temperature.

Surprisingly, the old magnetic tape, which is based on the same technology as HDD, doubles its capacity every two to three years. This increase is possible because the tape's reduced popularity means that tape technology has not been pushed to the superparamagnetic limit [323]. In 2017, IBM achieved a tape of capacity 201 GB/in^2 [325]. Another advantage of the tape is that it is less expensive than HDD and SSD. The rapid growth of tape capacity also has its drawbacks, as it causes backward-compatibility problems. Tape is also much slower to read than HDD, but it is much more stable hence more suitable for long-term storage.

For long-term storage, the capacity per inch is less important than sustainability; As data centres grow, they can provide storage with different price/performance trade-offs. For archival data, media lifetime is critical. Currently, archival data is more often moved to new a medium than it is read for any other reason. Although the tape is currently the best medium for archival storage, it is prone to humidity, temperature, and dust, and its lifetime is limited to around 30 years [326]. New more durable technology is needed; it should be inexpensive to produce and store, whilst maintaining relatively fast read times. The writing speed can be slower because, in principle, archival data is written only once but can be read many times.

If such a technology appears, it could be easily adopted by the market, as huge data centres can adopt technology much faster than home users [327]. Some promising storage mediums are DNA storage, holographic storage, and glass storage. The main advantage of DNA storage is that it is extremely dense; in theory it could achieve 10^9 GB/mm^3. Moreover it has a half-life of 500 years [326]. The DNA storage has been demonstrated to work on small amounts of data. However, the synthesis and sequencing are expensive and far from perfect; and DNA storage is not yet ready to enter the market.

Holographic storage and glass storage are more closely related to our work; we describe them in the following subsection.

Light-Based Storage

By light-based storage, we mean all storage based on the modification of material using light, as opposed to magnetic storage. This includes the so-called optical storage: CDs, DVDs and blue-ray discs.[1] We give here only a short overview of promising optical storage methods: glass storage and holographic storage.

Holographic storage is a more sophisticated cousin of optical discs. Optical discs store binary data in a few layers per disc. Holography, similarly to Lippmann photography, stores information in the bulk of the photosensitive material. During writing, two beams interfere inside the material: One is the reference beam and one describes the data, and the interference pattern is stored in the medium. To read the data, the disc is illuminated with the reference beam [327]. Thus, the reference beam acts as a memory address. The advantage of this method is that many holograms can be recorded in the same volume, each with its own reference beam. Capacities of 250 GB per disc are feasible [328]. As reading the whole area amounts to recording the reflection of the reference beam, holographic storage has extremely short access times, over 10 Gb/s.

Unfortunately, holographic storage suffers from similar problems as Lippmann photography did in its time. Although elegant in theory, it requires very responsive photosensitive material that is also transparent and not too scattering. To be commercially useful, it has to be comparable with tape in terms of durability and has to be easy to manufacture and be inexpensive [328]. Two approaches have been investigated. One uses photosensitive organic media, similar to photographic plates we discussed in previous chapters. The recording in such media is permanent, and the resulting storage is of write once read many (WORM) type. Another problem is that the storage can physically shrink (similarly to Lippmann plates when they dry); this causes mismatch between the recording and the reference beams. A second approach uses inorganic photorefractive media. In principle, such storage can be modified. However, the reading erases the data. Hence, in order for the photorefractive memory to be permanent, it has to be chemically fixed [328].

The difficulty in finding the right material makes holographic storage highly expensive [327]. As of today, although progress has been made, it has not outpaced the advancement in magnetic storage [328].

This brings us to glass storage. Glass is an inexpensive and durable material that can withstand temperatures of 1000C. It could be read optically in the same way as holographic storage, hence it should have comparable reading speeds. But how can we record light inside glass?

In 1996 Glezer et al. [329] demonstrated that it is possible to create permanent glass storage by using high energy lasers. More precisely, they used femto-second lasers: lasers that fire femto-second-long pulses. Short pulses can achieve very high powers, as power is energy over time. In such conditions, the reaction of glass to light is no longer linear and nano-structures are created. They have a different refractive index than the un-modified glass. Furthermore, they are birefringent, i.e., their refractive index depends on the polarisation of light. Their decay time has been estimated to be 10^{20} years at room

---

[1] The word "disk" usually refers to magnetic disks, and "disc" to optical discs. Here we use "disc" for everything that is not a magnetic disk.

temperature [330], thus making glass storage a WORM medium. In fact, the glass is not the only transparent medium in which this effect has been observed; nano-structures can be written also in sapphire and acrylic [329].

Recently, femto-lasers were used to produce "5D storage" in fused silica, i.e., glass consisting of almost pure silicon dioxide (SiO_2) [327, 330]. In this project, called Project Silica, a grid of voxels is created in the glass. A single voxel size is $1\,\mu m \times 1\,\mu m \times 10\,\mu m$ and there is some additional spacing between voxels. The first three dimensions are simply the three dimension of the silica disc. The other two dimensions are the intensity and polarisation (birefringerence) of the voxel. Using all this degrees of freedom results in capacities 360 Tb per disc [330].

Writing is done voxel-by-voxel and requires powerful femto-second lasers. However, to read the data, it suffices to have an optical microscope that can measure the polarisation of light. Reading is done layer-by-layer by focusing the microscope optics at a given depth. This means that multiple voxels can be imaged concurrently and the larger the voxel density is the faster the read is. Unfortunately, this also means that voxels introduce noise in the readings of their neighbours. The closer the voxels are, the more noise is introduced. To efficiently read the noisy data, machine-learning methods have been proposed [327].

3.2. Femto-Lippmann

Conceptually, Femto-Lippmann storage is somewhere between holographic and glass storage. The core idea is use femto-second lasers to write artificial interference patterns inside the glass. This way we can create durable colours or a durable hologram in a material less expensive than photosensitive emulsions.

Our method has a potential to increase the capacity of the glass storage. In Project Silica, the layers of voxels are 15 µm apart (10 µm for voxel plus space between voxels). In Section 2.2, we have seen that a Lippmann plate of depth 12 µm contains 32 colour samples per pixel. Therefore, using colours, we can add an extra "dimension" to the 5D storage. Our design also enables the development of new methods of reading the data, e.g., a scan of full column of voxels at a time. Moreover, durable colours embedded in glass could be used by artists or to replicate historical Lippmann plates.

At this stage of the Femto-Lippmann project, our goal is to create patterns in glass; these patterns should lead to different reflected spectra. We also want to develop methods for measuring these spectra. In other words, we want to create a proof of concept that the additional capacity of glass can be used in practice.

The creation of new storage media requires interdisciplinary teams [324]. Our team consists of computer scientists from the AudioVisual Communications Laboratory (LCAV) and physicists from the Galatea Laboratory; both laboratories are at EPFL. The researchers in Galatea are responsible for the heavy lifting, that is to say, for developing the optical setup for the laser and for using it to modify the glass. The LCAV part of the team is responsible for a *digital twin* of the plate that is used to test the approaches faster than in the physical setup[2]. In the following section (Section 3.3), we describe the mathematical model behind the digital twin and give a quick overview of the implementation. In Section 3.4, we show the insights we gained from analysing it.

3.3. Mathematical Model

In this section, we describe the mathematical model we developed to simulate the modification of glass and propagation of light through it. In Box 3.2 we give a quick overview of our implementation of this model.

Recording: Glass Modification

On a high level, we model the Femto-Lippmann recording by using the same steps as in Section 2.1. We calculate the intensity of the wave though the glass and the modification of the glass properties. However, there are some changes. We do not model the interference pattern separately. Instead, we treat it as a special case of propagation through non-uniform media, which we describe later in this section.

Another difference stems from the fact that our model needs to handle multiple consecutive recordings (we will discuss this in more detail in Section 3.4). As the possible change of the index of refraction is limited, the changes induced by the consecutive recordings cannot be simply added. To model the material better, we introduce the *deposited energy* that is added between recordings. The non-linearity is in the dependence of the change in the index of refraction of the deposited energy, see Figure 3.2 (right). As we know that the light pulse can modify glass only if it has high power, we can no longer assume that we are in the flat region of the Hurter-Driffield curve, see Figure 3.2 (left). The specific shape of these curves is not known; Figure 3.2 shows our educated guess.

Viewing: Propagation through Non-uniform Media

As the femto-laser pulse modifies the index of refraction and refractivity, we cannot use the model described in Chapter 2. There are two approaches we could take: One is to use the wave equation directly, and the second is to model the material as thin layers. Here, we adopt the second approach, but for curious readers there is an overview of the first method in Box 3.1.

The thin-layers approach can be derived similarly to the model in Section 2.1 (Figure 2.4) and is well-known as a matrix method or a matrix formulation of wave propagation [331, 332]. We consider only planar waves travelling in the z direction. We assume that the index of refraction $n(z)$ varies in the z direction but that it is constant in the x and y directions. We approximate the material as piece-wise uniform; that is, we assume it consists of layers of thickness Δz of constant index of refraction, see Figure 3.3.

Contrary to the reflection-based model, in the refraction-based model, the reflection occurs at the boundary between two layers, not inside the layer. Without loss of generality,

---

[2] The division between the computer scientists and physicists is not clear cut. I had an opportunity to use the femto lasers in GALATEA, while Ruben Ricca implemented his own simulation of layered materials.

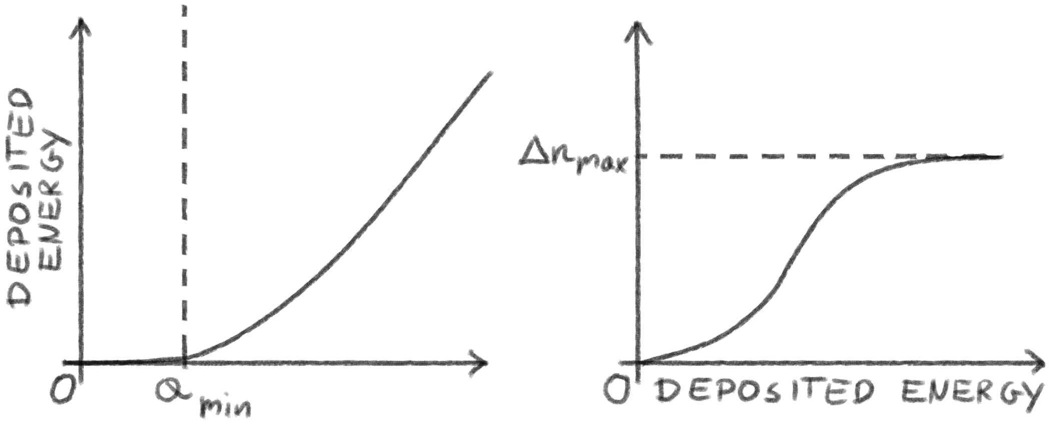


Figure 3.2.: Sketch of the material dependence on the energy of the pulse. Left: if the amplitude of the pulse is too small, the light transport is linear and the energy is not deposited. Right: the change of the index of refraction is limited and for some deposited energy it saturates.

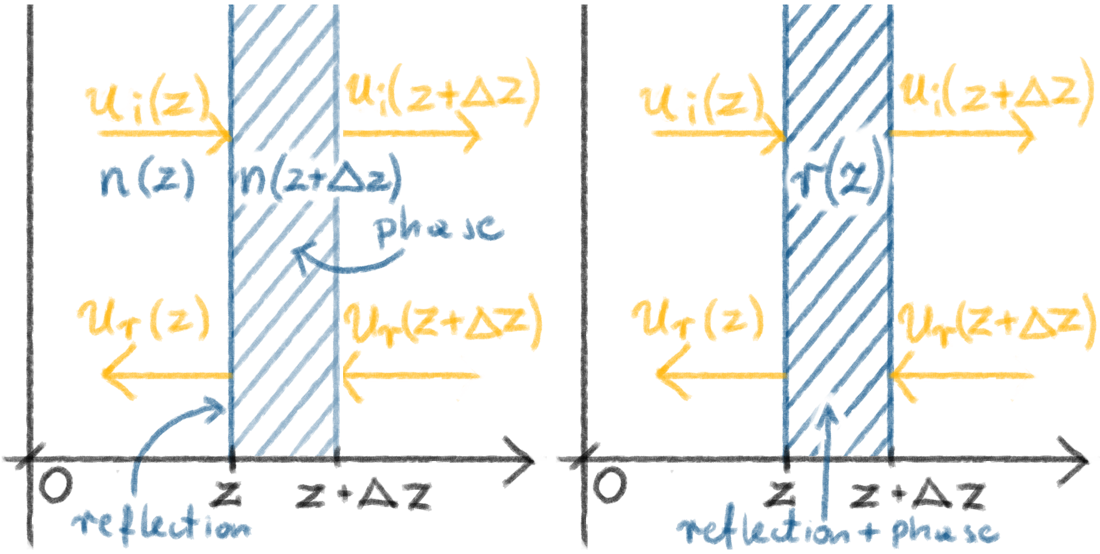


Figure 3.3.: Comparison between refraction model. Left: in the refraction-based model, the reflection happens on the boundary, but the phase is accumulated through the layer. Right: reflection-based model for reference.

we can assume that the reflection occurs on the left boundary of the layer. According to Fresnel's equations, if the incoming wave is $u_i(z)$, then the reflected wave is

$$u_r(z) = \frac{n(z+\Delta z) - n(z)}{n(z+\Delta z) + n(z)} u_i(z), \tag{3.1}$$

and the rest is transmitted. As the transmitted wave propagates through the layer, it gains the phase $e^{ik(z)\Delta z} = e^{in(z)k_0\Delta z}$, where $k_0 = k(z)/n(z)$ is the wave number in the free space. Similarly, a wave $u_r(z+\delta z)$ travels in the opposite direction (reflected deeper in the plate). The reflection of $u_r(z+\delta z)$ from the layer is a second order reflection that contributes to $u_i(z+\delta z)$. As in Section 2.1, we have a system of equations that couple u_r and u_i.

At this stage, we could apply an approximation similar to the one from Section 2.1, in order to obtain an analytical solution. The solution would be similar to the model proposed by Nareid and Pedersen [333], see Box 3.1 for further discussion.

We solve for u_i and u_r numerically for a few reasons. First, we do not know a priori if we are in the regime of small changes of n, where the first order approximation is applicable. More importantly, the patterns we simulate in the glass are not necessarily given in an analytical form, but they can be simulated. Finally, the numerical implementation is fairly simple, as we describe below. One drawback of a numerical approach is that it provides fewer intuitions than an analytical solution. However, as we have seen in Section 2.3: Reflection vs. Refraction, there are some parallels between reflection and refraction-based models, hence some of the intuitions from Section 2.1 can be applied here.

To calculate the propagation numerically, we relate the waves reflected from and transmitted through one layer by using linear equations,

$$\begin{bmatrix} u_i(z+\Delta z) \\ u_r(z+\Delta z) \end{bmatrix} = \boldsymbol{L}(z) \begin{bmatrix} u_i(z) \\ u_r(z) \end{bmatrix},$$

where $\boldsymbol{L}(z)$ depends on the index of refraction $n(z)$ of the two neighbouring layers and can be calculated from Fresnel's equations (Equation (3.1)). To calculate the outgoing waves, we solve this system of equations for $u_i(z+\Delta z)$ and $u_r(z)$. To calculate how the wave propagates through m layers, we can first multiply the matrices over all layers:

$$\begin{bmatrix} u_i(z+m\Delta z) \\ u_r(z+m\Delta z) \end{bmatrix} = \underbrace{\boldsymbol{L}(z+m\Delta z) \cdots \boldsymbol{L}(z+\Delta z)\boldsymbol{L}(z)}_{\boldsymbol{L}_{\text{tot}}} \begin{bmatrix} u_i(z) \\ u_r(z) \end{bmatrix},$$

and, then we solve the resulting equations for the outgoing waves $u_i(z+m\Delta z)$ and $u_r(z)$.

This matrix theory can be seen as a special case of the forward Euler method. It can be also used in the reflection-based model, with different matrices; in fact, the numerical spectrum in Figure 2.5 was calculated using matrix theory.

Want to know more? 3.1: Using the Wave Equation

Under the assumptions made at the beginning of Section 3.3, the wave equation can be written as follows:

$$\frac{\partial^2}{\partial z^2}u(z,t) - \frac{n(z)^2}{C^2}\frac{\partial^2 u}{\partial t^2}(z,t) = 0.$$

In the Fourier domain, for a single wave number $k_0 = \omega/C$, it can be simplified to

$$\frac{\partial^2}{\partial z^2}u(z) - n^2(z)k_0^2 u(z) = 0.$$

This equation cannot be solved analytically for general $n(z)$. However, similarly to the equations derived in Box 2.1, it can to be approximated. Nareid and Pedersen use the Born approximation in their work [306, 333]. More precisely, they assume that the variation in the index of refraction is small, i.e., it can be written as $n(z)^2 = n_0^2 + o(z)$, where $o(z)$ is small. After a few steps, including calculating the Green's system for the simplified differential equation, they arrive at a model similar to the one described in Section 2.1, but without the central term b_0.

In situations where the change of the index of refraction n is not small but regular in some way, for example, if the recorded spectrum is almost monochromatic, the coupled-mode theory [334, 335] can be used to obtain an approximate solution.

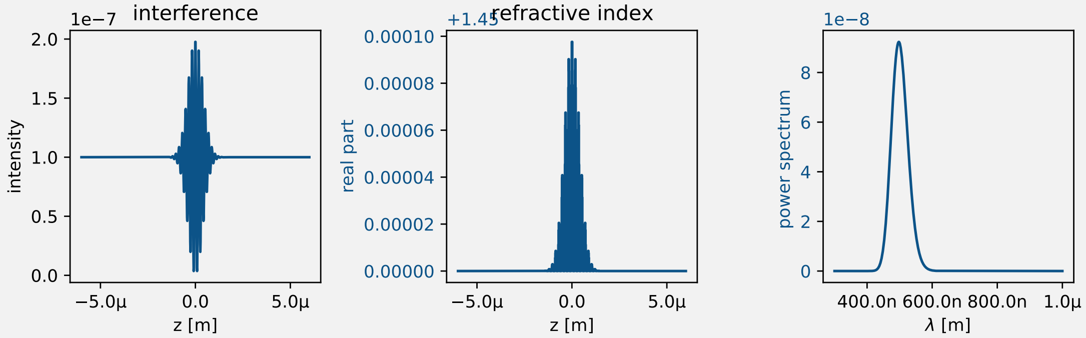
The wave equation could also be solved numerically; in fact the matrix formulation is one way to solve the wave equation numerically. Although, both the Born approximation and the matrix formulation give only an approximate solution, the quality of the result of the matrix formulation depends on how fine the discretisation is, therefore on the compute time. The quality of the Born approximation depends on how well $n(z)$ fits the assumptions and cannot be improved with more computation time.

Want to know more? 3.2: Implementation

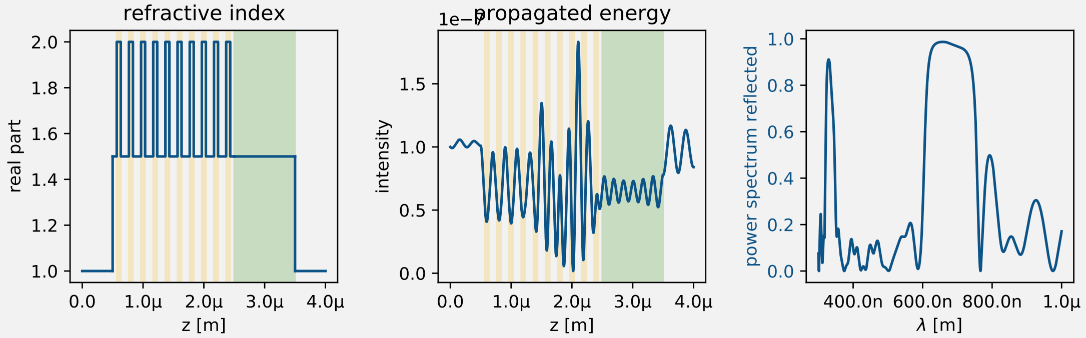
In this box, we give a quick overview of the propagation implementation we made for the digital twin. As always, we treat the materials as one dimensional. The main material model we implement is the dielectric, hence the model behaves as described in Section 3.3. The index of refraction in the dielectric can be modified by the energy of the propagating pulse. Therefore, by using matrix theory, all intermediate values of $u(z)$ are calculated, not only at the ends of the material.

Note that all the calculations in Section 3.3 are derived for a fixed wave number k_0. For performance purposes, we stack the propagation matrices into an order-3 tensor. This improves the computation over 100 times over iterative calculation (for 10000 different frequencies).

Below, we illustrate the simulated recording process. On the left, the interference pattern created in the material is shown. In the middle, we can see the index of refraction after recording. On the right, there is the reflected pulse; the glass was illuminated with the flat spectrum.



We can also simulate more complex materials. On the left, we have a composite material that consists of air ($n(z) = 1$), a layered material ($n(z)$ between 1.5 and 2), a block of a material with a constant index of refraction ($n(z) = 1.5$) and another layer of air. In the middle is the estimated propagation of the material, and on the right is the reflected pulse.



The flat region of high reflection is the reflection layered part of the material.

3.4. Preliminary Results

In this section, we show simulations of the recording and how they guided the Femto-Lippmann project. First, we discuss a simple method (Pulse by Pulse) that we have ruled out using simulations. Then, we propose to use two laser pulses, describe the challenges that arise in this setup (Interference), and consider methods of obtaining different patterns with pulse shaping (Increasing the Capacity).

3.4.1. Pulse by Pulse

Our first method was to write a repeating pattern in glass, known in optics as a *volume grating* or a *fiber grating*, see Figure 3.4. We could write such a pattern by using the same method used in Project Silica [330].

A grating structure will reflect the light of the wavelength equal to twice the grating period Z. More precisely, the wavelength reflected most strongly will be $\lambda = 2n_{\mathrm{eff}}Z \sim 2n_{\mathrm{glass}}Z$, where n_{eff} is the index of refraction calculated from the effective speed of light in the grating. Here, we can assume that it is simply the index of refraction of the glass, $n_{\mathrm{glass}} \sim 1.5$.

This means that to reflect the visible light, e.g., $\lambda_{\mathrm{red}} = 700\,\mathrm{nm}$, the period of the grating would have to be $Z = \frac{1}{2n_{\mathrm{glass}}}\lambda_{\mathrm{red}} \sim 230\,\mathrm{nm} = 0.23\,\mu\mathrm{m}$. The size of the nano-structure used in Project Silica, $1\,\mu\mathrm{m} \times 1\,\mu\mathrm{m} \times 10\,\mu\mathrm{m}$, is too large for that. With smaller power and more focused laser objectives, nano-structures can be shrunk to $0.2\,\mu\mathrm{m} \times 0.2\,\mu\mathrm{m} \times 1\,\mu\mathrm{m}$, and the interference pattern can be produced along the x or y direction, instead of z, see Figure 3.4 (left). Even with these modifications, the grating period is still too large to produce colours in the visible spectrum. Finally, we can force the period of about 230 nm by overlapping the nano-structures, but this leads to low contrast in the index of refraction and weak reflection, see Figure 3.5. (right).

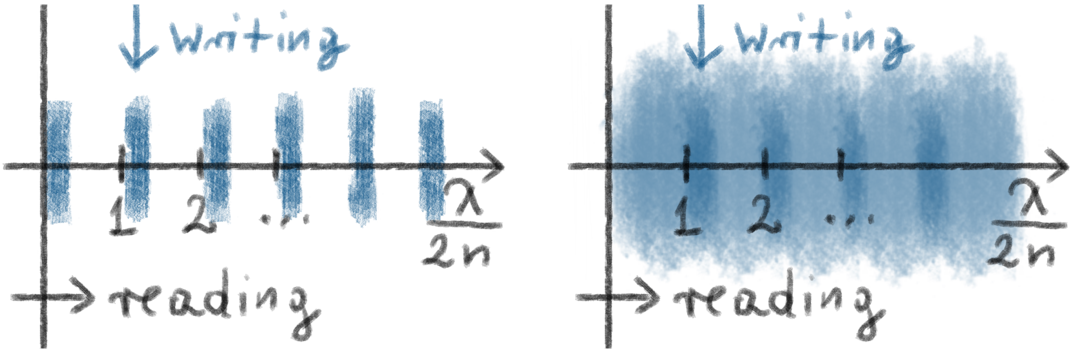


Figure 3.4.: The simplest writing method. Left: by writing in the direction perpendicular to reading, we can obtain the period of the pattern smaller than if the reading and writing directions were parallel. Right: we can write the pattern, even if the desired period of the pattern is too small compared to the pulse size. But as the pulses overlap, the contrast is reduced, and so is the reflection, see Figure 3.5.

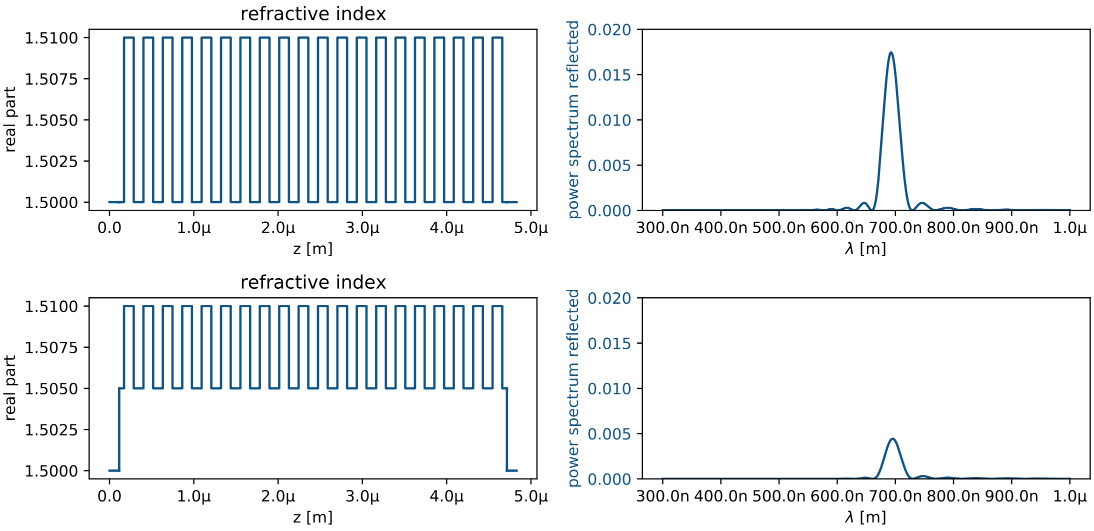


Figure 3.5.: Reflection from the grating. Left: the grating we would like to create, with high contrast. On the left is the index of refraction of the grating, on the right is the reflection form the grating. Top row: an ideal grating that we would like to create. Bottom row: the grating creating by overlapping the too-wide pulses. We can see that the reflected power spectrum is about four times smaller in the second row than in the first.

3.4.2. Interference

To create smaller structures, we take inspiration from Lippmann recording and propose to use the interference of the laser beam with its reflection. The simplest way would be to coat the surface of the glass with metal, mimicking Lippmann's process.

A drawback of this approach is that the interference pattern can be created only close to the mirror. To achieve high capacity, we would like to be able to use the whole volume of the glass. Therefore, we propose to use two beams travelling in opposite directions, as depicted in Figure 3.6.

The two-beam approach has its own challenges. In both approaches, the modification of the index of refraction from a single interference pattern is too small to obtain significant reflection. Therefore, the energy of multiple pulses has to be deposited to create one nano-structure. The laser produces a train of pulses with frequency 100 kHz; hence, to record 100 pulses, we can simply wait one millisecond. When recording by using only one beam (see Pulse by Pulse), the shape of the beam depends only on the optics, therefore the modifications overlap perfectly and enhance the nano-structure.

However, in the multiple pulses case, the difference in phase between the two interfering pulses causes a shift in the interference pattern. Indeed, we have seen the difference in phase in Section 2.1 in the form of the phase shift at the reflection. If the phase $\theta = 0$, then the peak of the interference pattern is at the boundary, and if the $\theta = \pi$, there is a valley of the interference pattern at the boundary. This is a problem, because the phase

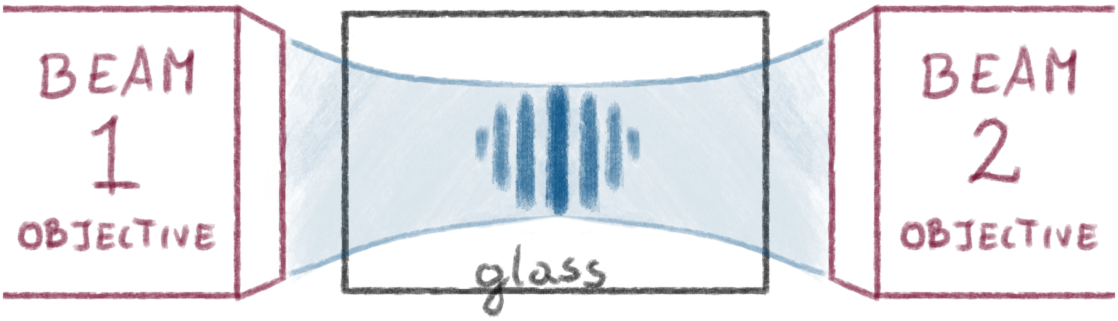


Figure 3.6.: Two-beam writing method. Two beams create an interference pattern inside the glass, and the period of the pattern is half the wavelength of the laser; much finer than with a single laser beam. Note that the pattern is perpendicular to the direction of the propagation of the laser.

of light from the laser can change from pulse to pulse, and adding multiple pulses with different shifts can lead to a blurred pattern, see Figure 3.7

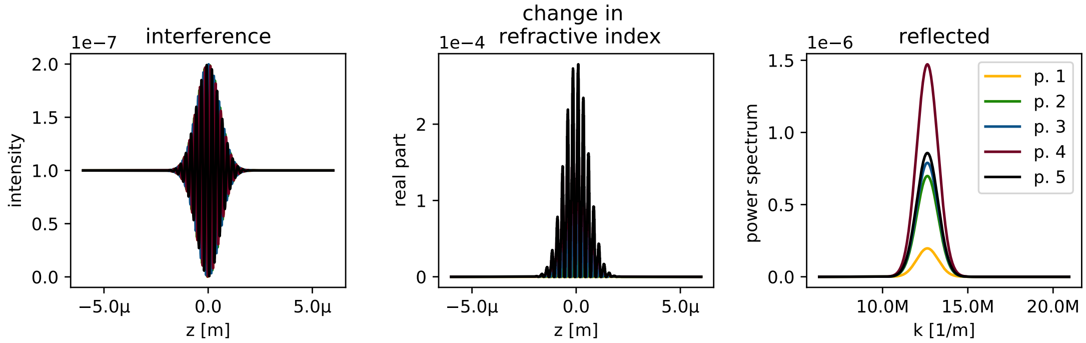


Figure 3.7.: Naive random-phase simulation. One pulse is kept the same, and the phase of the second one is chosen uniformly at random from the interval $(0, \pi)$. When calculating an interference pattern, the previously recorded patterns are not taken into account. Left: since they are independent, the interference patterns create a blurred shape. Middle: the refractive index grows over time, but it loses much of variation. Right: the reflection is not monotonic over time, e.g., the reflection after four pulses is stronger than after five.

One way to avoid this problem is to use a beam-splitter to interfere the beam with itself. This way, the difference in phases is only due to the difference of the paths of the two beams, and it is the same for consecutive pulses, see Figure 3.8. More formally, if $a(\omega)$ is the amplitude of the pulse, the wave from one path will be

$$u_1(k, z) = a(\omega)e^{i\theta}e^{ikz_1}e^{i(kz-\omega t)},$$

where z_1 is the length of the first path, including all material changes and reflections and θ is the random phase. The wave taking the second path will be travelling in the opposite

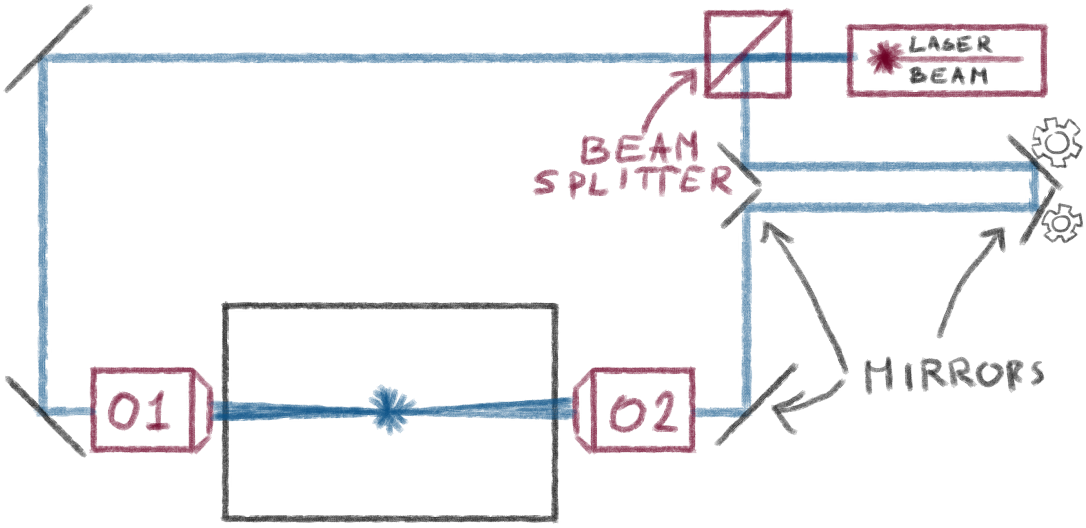


Figure 3.8.: Two objective setup. Left: schematic illustration. The laser beam is split with the beam splitter, and each of the resulting beams (blue) is focused with its own objective (O1 and O2). Both positions of the two right most mirrors have to be adjusted so that the path lengths of the two beams are exactly the same.

direction and have the same random phase θ:

$$u_2(k,z) = e^{i\theta} e^{ikz_2} e^{-i(kz+\omega t)}.$$

Summing up these waves, we obtain the following interference pattern:

$$p(k,z) = |u_1(k,z) + u_2(k,z)|^2 = \left|A(\omega)e^{i\theta}\right|^2 (2 + 2\cos(kz_1 - kz_2)).$$

As $\left|e^{i\theta}\right|^2 = 1$, the interference pattern no longer depends on the phase. Note that if we had two independent phases, θ_1 and θ_2, the expression would depend on their difference.

The setup with the beam splitter is challenging to build in practice. All the elements have to be precisely aligned. In particular, both objectives have to be focused in the point where path lengths of the two beams are the same.

Although the team at GALATEA has managed this feat, see Figure 3.9, we have also explored alternatives. We simulated whether it is possible to record a pattern with high contrast by using an only somewhat aligned phase, as usually the phase changes slowly between the pulses. Our intuition is that, even if the pattern in the glass is weak, the light from the next pulse will be partially reflected from it and the phase difference will be partially corrected.

We simulated two scenarios, depicted in Figure 3.10. In both cases the phase of one pulse is fixed. In the first scenario, the phase of the second pulse changes slowly over 5 pulses. In the second scenario, the phase of the second pulse is random, thus following the uniform distribution on the interval $[0, \pi]$. In both cases, we can see that the pattern is not entirely blurred. From the pattern created by 5 consecutive pulses, we gain a reflection much stronger than from the pattern created from a single pulse.

Finally, our simulations also have the potential to explain some recently observed phenomena caused by the high-power laser modification of glass. It has been observed that patterns form spontaneously in the glass; the patterns are parallel to the laser beam and depend on the polarisation of the laser. Inspired by this observation, we simulated self-organisation of nano-structures in the material, also depicted in Figure 3.10. We show that even a weak pattern can be amplified by laser pulses that come from one direction. The pulses reflect from the pattern and create partial standing wave, like in the Lippmann plate (but, of course, weaker). In this manner, the impurities in the glass could form a seed that, under a high-power laser, is amplified to patterns, without the need for interfering beams.

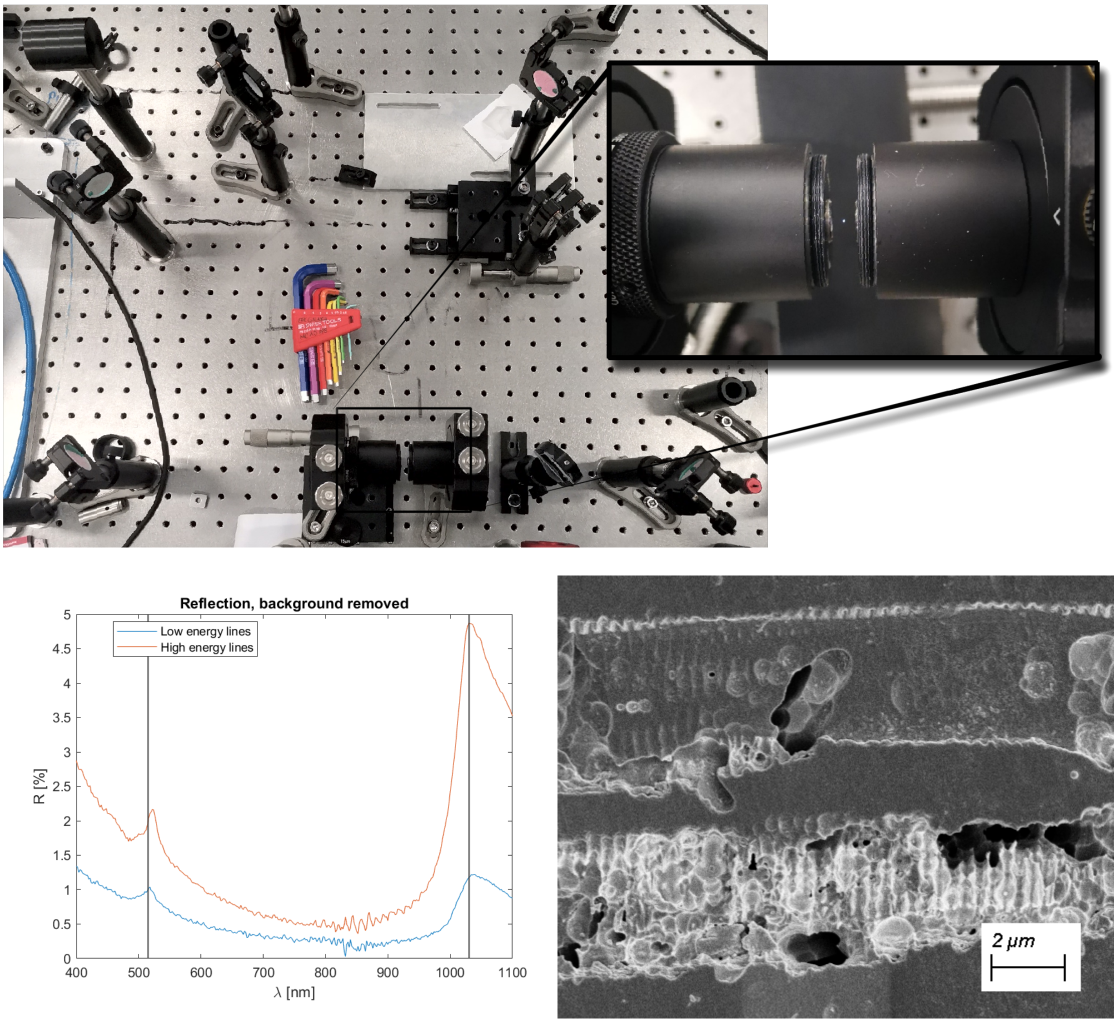


Figure 3.9.: Results from GALATEA laboratory. Top: photo of a setup in GALATEA, implementing the schematic in Figure 3.8. The inset shows two objectives focusing laser beam in the point in the middle. Bottom left: spectrum reflected from the recorded interference pattern. The peaks in the spectrum match the expected maxima (black lines). Bottom right: a section through the interference pattern recorded in the glass, as seen under electron microscope. Images courtesy of Rubben Ricca and Julien Gateau.

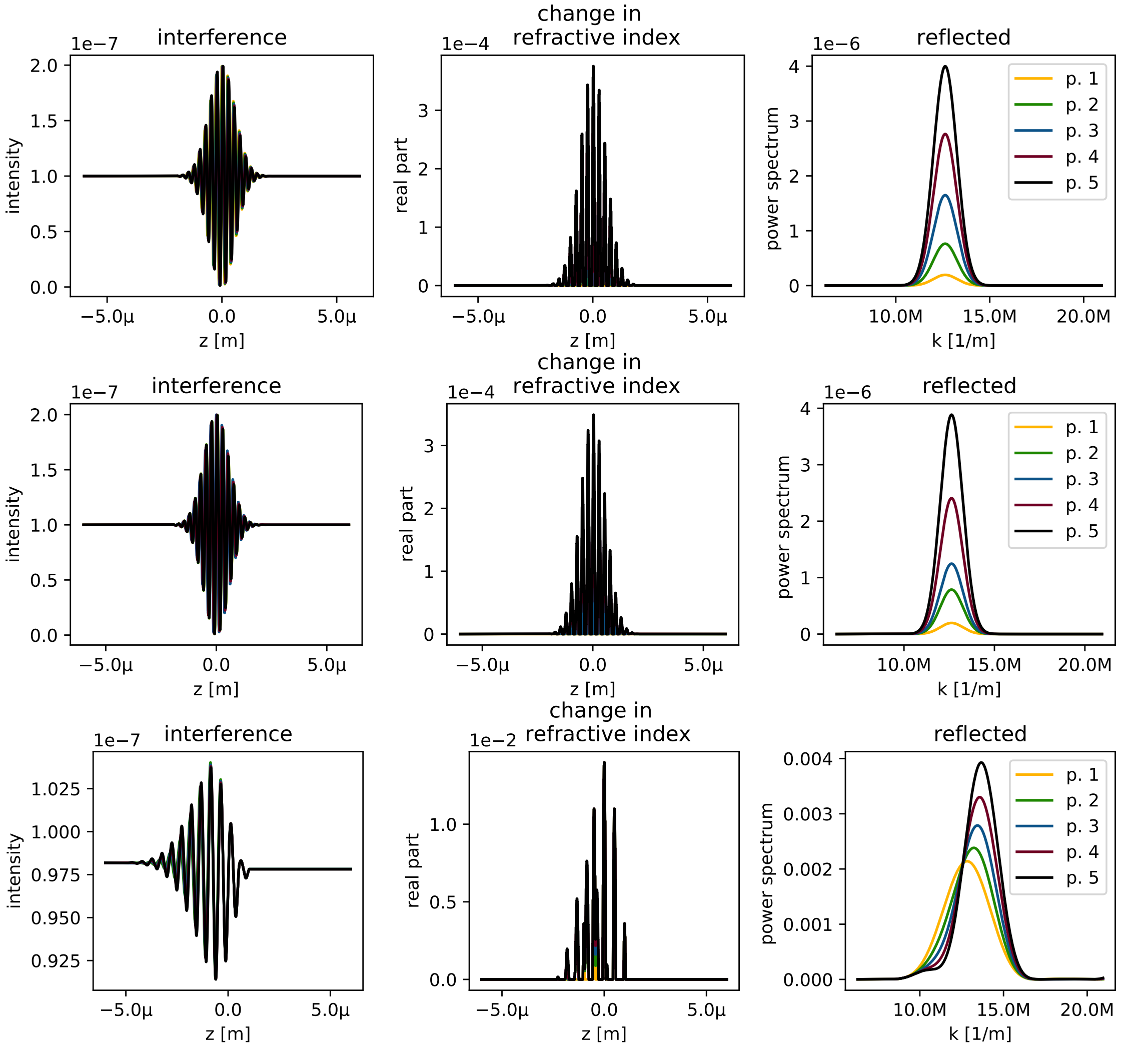


Figure 3.10.: Simulations of recording with multiple pulses. Left: overlapped interference patterns Middle: the change in the refractive index. Right: the reflection after one to five pulses. First row: one pulse is fixed, and the phase of the second one changes by $\pi/10$ with each pulse. Second row: one pulse is fixed, and the phase of the second one is chosen uniformly at random from the interval $(0, \pi)$. In both cases, the amount of reflected light rows with each pulse. Third row: self organisation. A glass with a weak interference pattern recorded is illuminated by a single pulse (travelling from left to right). Witch each pulse, more energy is reflected and the interference pattern is stronger; the refractive index pattern is asymmetric, because it is created by pulses travelling from left to right. The reflection grows over time, especially for later pulses. It is not comparable with the first two rows because, in order to observe self-organisation, we simulated higher energies.

3.4.3. Increasing the Capacity

So far, we have discussed interference patterns created by a laser beam that has a predefined spectrum. As we have seen in Section 2.1, the reflected spectrum depends only on the spectrum of the recording. With a single laser spectrum, we cannot encode any information — we need to have multiple possible patterns that lead to different reflections. We propose to modify one or two of the pulses used to create the interference.

From a technical point of view, the simplest way to modify the pulse is to introduce a frequency dependent phase shift. Such pulses are called chirps, after their acoustic equivalents. In this section, we analyse whether a chirped pulse can create different interference pattern than the original pulse. From the previous subsection, we know that the interference pattern of a pulse with itself depends only on the power spectrum of the pulse, thus chirping both pulses in the same way does not change the interference pattern. From simulations, we observe that chirping just one pulse also does not create variation in the reflected spectrum, see Figure 3.11.

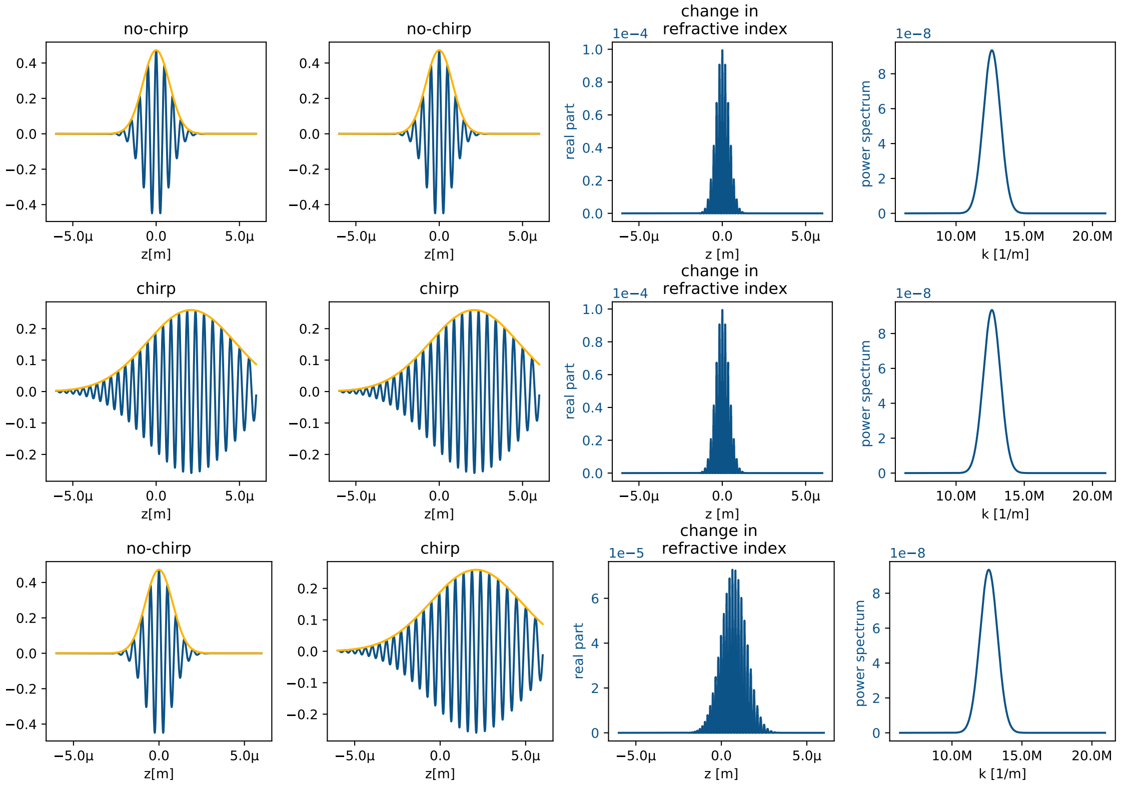


Figure 3.11.: Different chirps interfering. Top row: interference with two non-chirped pulses. Middle row: interference with two chirped pulses. Bottom row: interference with one chirped and one non-chirped pulse. Left: chirping Gaussian spectrum leads to spreading of the pulse amplitude in time (hence in space). Middle and right: the recorded patterns and reflected spectra look the same across the experiments.

Fortunately, it is also possible to filter the laser spectrum differently at different frequencies. As the duration of laser pulses is very short, the spectrum of the laser is rather wide; in the order of tens to hundreds of nano-meters. We do not present here patterns created with different spectra, as this case is very similar to the case analysed in Chapter 2.

3.5. Conclusions

In this chapter, we have proposed glass storage inspired by Lippmann photography. The Femto-Lippmann project has a long way to go before being marketed, but we believe we have shown that it has potential to increase the capacity of glass storage.

We have shown the simulations that guided some of the decisions in the project and that gave us intuitions on which approaches might work. Our simulations are in line with the experiments performed in the GALATEA Laboratory.

As our implementation was designed to be extendable, e.g., to describe the propagation in 3D space. In order to model planar waves travelling in different directions (not only along z), it is sufficient to replace the transfer matrices with 6×6 matrices (two waves, each described by wave vector $\boldsymbol{k} \in \mathbb{R}$), see for example [336]. Under this model, any wave can be decomposed into planar waves via the Fourier transform, at the computational cost of modelling all the planar waves. Our implementation could also be extended to specific beams, such as a Gaussian beam that can be described with a fixed number of parameters. This will be less flexible, but more computationally efficient, as small transfer matrices can be calculated [337, 338] for the Gaussian beam.

Bibliography

[1] P. Misra and P. Enge, “Global positioning system: signals, measurements and performance second edition,” *Global Positioning System: Signals, Measurements And Performance Second Editions,*, 2006.

[2] J. Smith and J. Abel, “Closed-form least-squares source location estimation from range-difference measurements,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 12, pp. 1661–1669, 1987.

[3] C. Cadena *et al.*, “Past, present, and future of simultaneous localization and mapping: toward the robust-perception age,” *IEEE Trans Robot*, vol. 32, no. 6, pp. 1309–1332, 2016.

[4] IEEE Signal Processing Society. “Fifty years of signal processing.” (1998), [Online]. Available: `https://signalprocessingsociety.org/uploads/history/history.pdf`. (accessed: 22.06.2021).

[5] E. Meijering, “A chronology of interpolation: from ancient astronomy to modern signal and image processing,” *Proceedings of the IEEE*, vol. 90, no. 3, pp. 319–342, 2002.

[6] J. R. Higgins, “Five short stories about the cardinal series,” *Bulletin of the American Mathematical Society*, vol. 12, no. 1, pp. 45–89, 1985.

[7] C. E. Shannon, “Communication in the presence of noise,” *Proc. of the IRE*, vol. 37, no. 1, pp. 10–21, Jan. 1949, ISSN: 0096-8390. DOI: `10.1109/JRPROC.1949.232969`.

[8] E. T. Whittaker, “On the functions which are represented by the expansions of the interpolation-theory,” *Proc. of the Royal Society of Edinburgh*, vol. 35, pp. 181–194, 1915. DOI: `10.1017/S0370164600017806`.

[9] H. Nyquist, “Certain topics in telegraph transmission theory,” *Trans. of the American Institute of Electrical Engineers*, vol. 47, no. 2, pp. 617–644, Apr. 1928, ISSN: 0096-3860. DOI: `10.1109/T-AIEE.1928.5055024`.

[10] В. А. Котельников, «О пропускной способности “эфира” и проволоки в электросвязи», *Всесоюзному съезду по вопросам технической реконструкции дела связи и развития слаботочной промышленности*, 1933.

[11] V. A. Kotel’nikov, “On the transmission capacity of the “ether” and wire in electrocommunications,” in *Modern Sampling Theory*, Springer, 2001, pp. 27–45.

[12] K. Ogura, “On a certain transcendental integral function in the theory of interpolation,” *Tohoku Mathematical Journal, First Series*, vol. 17, pp. 64–72, 1920.

[13] D. Slepian, “On bandwidth,” *Proceedings of the IEEE*, vol. 64, no. 3, pp. 292–300, 1976.

[14] M. Unser, "Sampling-50 years after Shannon," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 569–587, 2000.

[15] A. Aldroubi and K. Gröchenig, "Nonuniform sampling and reconstruction in shift-invariant spaces," *SIAM Rev.*, vol. 43, no. 4, pp. 585–620, Apr. 2001, ISSN: 0036-1445. DOI: `10.1137/S0036144501386986`. [Online]. Available: `http://dx.doi.org/10.1137/S0036144501386986`.

[16] A. J. Jerri, "The Shannon sampling theorem — its various extensions and applications: a tutorial review," *Proceedings of the IEEE*, vol. 65, no. 11, pp. 1565–1596, 1977.

[17] J. Yen, "On nonuniform sampling of bandwidth-limited signals," *IRE Transactions on circuit theory*, vol. 3, no. 4, pp. 251–257, 1956.

[18] C. J. De la Vallée Poussin, « Sur la convergence des formules d'interpolation entre ordonnées équidistantes », *Bull. Acad. Sei. Belg*, p. 314-410, 1908.

[19] M. Theis, „Über eine Interplationsformel von de la Vallée Poussin", *Mathematische Zeitschrift*, Jg. 3, Nr. 1, S. 93–113, 1919.

[20] K. Yao, "Applications of reproducing kernel Hilbert spaces–bandlimited signal models," *Information and Control*, vol. 11, no. 4, pp. 429–444, 1967.

[21] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE signal processing magazine*, vol. 8, no. 4, pp. 14–38, 1991.

[22] L. Shao, R. Yan, X. Li, and Y. Liu, "From heuristic optimization to dictionary learning: a review and comprehensive comparison of image denoising algorithms," *IEEE transactions on cybernetics*, vol. 44, no. 7, pp. 1001–1013, 2013.

[23] S. Klimenko and G. Mitselmakher, "A wavelet method for detection of gravitational wave bursts," *Classical and Quantum Gravity*, vol. 21, no. 20, S1819, 2004.

[24] R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, R. Adhikari, V. Adya, C. Affeldt, M. Agathos, *et al.*, "Gw190521: a binary black hole merger with a total mass of $150 M_\odot$," *Physical review letters*, vol. 125, no. 10, p. 101 102, 2020.

[25] S. Mallat, *A wavelet tour of signal processing.* Elsevier, 1999.

[26] R. Polikar, "The story of wavelets," *Physics and modern topics in mechanical and electrical engineering*, pp. 192–197, 1999.

[27] I. Daubechies, *Ten lectures on wavelets.* SIAM, 1992.

[28] A. Bultheel *et al.*, "Learning to swim in a sea of wavelets," *Bulletin of the Belgian Mathematical society-simon stevin*, vol. 2, no. 1, pp. 1–45, 1995.

[29] I. Daubechies, "Where do wavelets come from? a personal point of view," *Proceedings of the IEEE*, vol. 84, no. 4, pp. 510–513, 1996.

[30] M. Unser and J. Zerubia, "Generalized sampling: stability and performance analysis," *IEEE Transactions on Signal Processing*, vol. 45, no. 12, pp. 2941–2950, 1997.

[31] R. Venkataramani and Y. Bresler, “Sampling theorems for uniform and periodic nonuniform mimo sampling of multiband signals,” *IEEE transactions on signal processing*, vol. 51, no. 12, pp. 3152–3163, 2003.

[32] ——, “Multiple-input multiple-output sampling: necessary density conditions,” *IEEE transactions on information theory*, vol. 50, no. 8, pp. 1754–1768, 2004.

[33] R. Vio, T. Strohmer, and W. Wamsteker, “On the reconstruction of irregularly sampled time series,” English, *Publications of the Astronomical Society of the Pacific*, vol. 112, no. 767, pp. 74–90, 2000, ISSN: 00046280. [Online]. Available: `http://www.jstor.org/stable/10.1086/316495`.

[34] K. Gröchenig, “Reconstruction algorithms in irregular sampling,” *Mathematics of computation*, vol. 59, no. 199, pp. 181–194, 1992.

[35] H. Landau, “Necessary density conditions for sampling and interpolation of certain entire functions,” *Acta Mathematica*, vol. 117, no. 1, pp. 37–52, 1967.

[36] K. Gröchenig and H. Razafinjatovo, “On Landau’s necessary density conditions for sampling and interpolation of band-limited functions,” *Journal of the London Mathematical Society*, vol. 54, no. 3, pp. 557–565, 1996.

[37] Y. Liu, “Irregular sampling for spline wavelet subspaces,” *IEEE Trans. on Information Theory*, vol. 42, no. 2, pp. 623–627, Mar. 1996, ISSN: 0018-9448. DOI: `10.1109/18.485731`.

[38] W. Chen, S. Itoh, and J. Shiki, “Irregular sampling theorems for wavelet subspaces,” *IEEE Trans. on Information Theory*, vol. 44, no. 3, pp. 1131–1142, May 1998, ISSN: 0018-9448. DOI: `10.1109/18.669187`.

[39] A. Aldroubi and K. Gröchenig, “Beurling-Landau-type theorems for non-uniform sampling in shift invariant spline spaces,” *Journal of Fourier Analysis and Applications*, vol. 6, no. 1, pp. 93–103, 2000.

[40] Y. Lyubarskii and W. Madych, “The recovery of irregularly sampled band limited functions via tempered splines,” *Journal of Functional Analysis*, vol. 125, no. 1, pp. 201–222, 1994.

[41] M. Unser and A. Aldroubi, “A general sampling theory for nonideal acquisition devices,” *IEEE Transactions on Signal Processing*, vol. 42, no. 11, pp. 2915–2925, 1994.

[42] T. Blu and M. Unser, “Quantitative fourier analysis of approximation techniques. I. Interpolators and projectors,” *IEEE Transactions on signal processing*, vol. 47, no. 10, pp. 2783–2795, 1999.

[43] ——, “Quantitative fourier analysis of approximation techniques. II. Wavelets,” *IEEE Transactions on Signal Processing*, vol. 47, no. 10, pp. 2796–2806, 1999.

[44] M. Vetterli, J. Kovacevic, and V. K. Goyal, *Foundations of Signal Processing.* Cambridge University Press, 2014.

[45] Y. C. Eldar and T. Werther, "General framework for consistent sampling in Hilbert spaces," *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 3, no. 04, pp. 497–509, 2005.

[46] B. Adcock and A. C. Hansen, "A generalized sampling theorem for stable reconstructions in arbitrary bases," *Journal of Fourier Analysis and Applications*, vol. 18, no. 4, pp. 685–716, 2012.

[47] B. Adcock, A. C. Hansen, and C. Poon, "Beyond consistent reconstructions: optimality and sharp bounds for generalized sampling, and application to the uniform resampling problem," *SIAM Journal on Mathematical Analysis*, vol. 45, no. 5, pp. 3132–3167, 2013.

[48] B. A. Moser and T. Natschlaeger, "On stability of distance measures for event sequences induced by level-crossing sampling," *IEEE Transactions on Signal Processing*, vol. 62, no. 8, pp. 1987–1999, 2014.

[49] S.-C. Liu and T. Delbruck, "Neuromorphic sensory systems," *Current opinion in neurobiology*, vol. 20, no. 3, pp. 288–295, 2010.

[50] G. Gallego, T. Delbruck, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. Davison, J. Conradt, K. Daniilidis, *et al.*, "Event-based vision: a survey," *arXiv preprint arXiv:1904.08405*, 2019.

[51] A. A. Lazar and E. A. Pnevmatikakis, "Video time encoding machines," *IEEE Transactions on Neural Networks*, vol. 22, no. 3, pp. 461–473, 2011.

[52] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "High speed and high dynamic range video with an event camera," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[53] F. Bond and C. Cahn, "On the sampling the zeros of bandwidth limited signals," *IRE transactions on information theory*, vol. 4, no. 3, pp. 110–113, 1958.

[54] B. F. Logan Jr, "Information in the zero crossings of bandpass signals," *Bell System Technical Journal*, vol. 56, no. 4, pp. 487–510, 1977.

[55] I. Bar-David, "An implicit sampling theorem for bounded bandlimited functions," *Information and Control*, vol. 24, no. 1, pp. 36–44, 1974.

[56] J. Mark and T. Todd, "A nonuniform sampling approach to data compression," *IEEE Transactions on communications*, vol. 29, no. 1, pp. 24–32, 1981.

[57] A. A. Lazar and L. T. Tóth, "Perfect recovery and sensitivity analysis of time encoded bandlimited signals," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 10, pp. 2060–2073, 2004.

[58] D. Gontier and M. Vetterli, "Sampling based on timing: time encoding machines on shift-invariant subspaces," *Applied and Computational Harmonic Analysis*, vol. 36, no. 1, pp. 63–78, 2014.

[59] H. Inose, T. Aoki, and K. Watanabe, "Asynchronous delta-modulation system," *Electronics Letters*, vol. 2, no. 3, pp. 95–96, 1966.

[60] B. E. Boser and B. A. Wooley, "The design of sigma-delta modulation analog-to-digital converters," *IEEE Journal of solid-state circuits*, vol. 23, no. 6, pp. 1298–1308, 1988.

[61] M. Miskowicz, "Send-on-delta concept: an event-based data reporting strategy," *sensors*, vol. 6, no. 1, pp. 49–63, 2006.

[62] Y. M. Lu and M. N. Do, "A theory for sampling signals from a union of subspaces," *IEEE transactions on signal processing*, vol. 56, no. 6, pp. 2334–2345, 2008.

[63] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 5302–5316, 2009.

[64] M. F. Duarte and Y. C. Eldar, "Structured compressed sensing: from theory to applications," *IEEE Transactions on signal processing*, vol. 59, no. 9, pp. 4053–4085, 2011.

[65] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE transactions on information theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[66] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on information theory*, vol. 52, no. 2, pp. 489–509, 2006.

[67] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[68] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: universal encoding strategies?" *IEEE transactions on information theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[69] M. Vetterli, P. Marziliano, and T. Blu, "Sampling signals with finite rate of innovation," *IEEE Trans. on Signal Processing*, vol. 50, pp. 1417–1428, 6 Aug. 2002.

[70] P. L. Dragotti, M. Vetterli, and T. Blu, "Sampling moments and reconstructing signals of finite rate of innovation: Shannon meets Strang–Fix," *IEEE Transactions on signal processing*, vol. 55, no. 5, pp. 1741–1757, 2007.

[71] P. Feng and Y. Bresler, "Spectrum-blind minimum-rate sampling and reconstruction of multiband signals," in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, IEEE, vol. 3, 1996, pp. 1688–1691.

[72] E. J. Candes *et al.*, "The restricted isometry property and its implications for compressed sensing," *Comptes rendus mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

[73] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on information theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[74] E. J. Candès and Y. Plan, "Near-ideal model selection by $\ell 1$ minimization," *The Annals of Statistics*, vol. 37, no. 5A, pp. 2145–2177, 2009.

[75] E. J. Candes, Y. C. Eldar, D. Needell, and P. Randall, “Compressed sensing with coherent and redundant dictionaries,” *Applied and Computational Harmonic Analysis*, vol. 31, no. 1, pp. 59–73, 2011.

[76] B. Adcock, A. C. Hansen, C. Poon, and B. Roman, “Breaking the coherence barrier: a new theory for compressed sensing,” in *Forum of Mathematics, Sigma*, Cambridge University Press, vol. 5, 2017.

[77] G. Prony, « Essai experimental et analytic », *J. d’Ecole Polytech. (Paris)*, t. 1, p. 24-76, 1795.

[78] P. Stoica, R. L. Moses, *et al.*, “Spectral analysis of signals,” 2005.

[79] J. M. Vieira and P. J. Ferreira, “Interpolation, spectrum analysis, error-control coding, and fault-tolerant computing,” in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, IEEE, vol. 3, 1997, pp. 1831–1834.

[80] H. Pan, “Looking beyond pixels theory, algorithms and applications of continuous sparse recovery,” Ph.D. dissertation, EPFL, Lausanne, 2018. DOI: `10.5075/epfl-thesis-8340`. [Online]. Available: `http://infoscience.epfl.ch/record/255725`.

[81] M. Simeoni, A. Besson, P. Hurley, and M. Vetterli, “Cadzow plug-and-play gradient descent for generalised FRI,” 2020.

[82] B. Adcock and A. C. Hansen, “Generalized sampling and infinite-dimensional compressed sensing,” *Foundations of Computational Mathematics*, vol. 16, no. 5, pp. 1263–1323, 2016.

[83] M. Cheraghchi and J. Ribeiro, “An overview of capacity results for synchronization channels,” *IEEE Transactions on Information Theory*, vol. 67, no. 6, pp. 3207–3232, 2020.

[84] P. Marziliano and M. Vetterli, “Reconstruction of irregularly sampled discrete-time bandlimited signals with unknown sampling locations,” *IEEE Trans. on Signal Processing*, vol. 48, no. 12, pp. 3462–3471, Dec. 2000, ISSN: 1053-587X. DOI: `10.1109/78.887038`.

[85] V. Emiya, A. Bonnefoy, L. Daudet, and R. Gribonval, “Compressed sensing with unknown sensor permutation,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2014, pp. 1040–1044.

[86] G. Elhami, A. Scholefield, B. B. Haro, and M. Vetterli, “Unlabeled sensing: reconstruction algorithm and theoretical guarantees,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Ieee, 2017, pp. 4566–4570.

[87] A. Pananjady, M. J. Wainwright, and T. A. Courtade, “Linear regression with shuffled data: statistical and computational limits of permutation recovery,” *IEEE Trans. on Information Theory*, pp. 1–15, 2017.

[88] D. J. Hsu, K. Shi, and X. Sun, “Linear regression without correspondence,” in *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., 2017, pp. 1531–1540.

[89] J. Unnikrishnan, S. Haghighatshoar, and M. Vetterli, “Unlabeled sensing with random linear measurements,” *IEEE Trans. on Information Theory*, vol. 64, no. 5, pp. 3237–3253, May 2018.

[90] S. Haghighatshoar and G. Caire, “Signal recovery from unlabeled samples,” *IEEE Trans. on Signal Processing*, pp. 1242–1257, 05 Mar. 2018.

[91] I. Dokmanić, “Permutations unlabeled beyond sampling unknown,” *IEEE Signal Processing Letters*, vol. 26, no. 6, pp. 823–827, 2019.

[92] J. Browning, “Approximating signals from nonuniform continuous time samples at unknown locations,” *IEEE Trans. on Signal Processing*, vol. 55, no. 4, pp. 1549–1554, Apr. 2007, ISSN: 1053-587X. DOI: `10.1109/TSP.2006.889979`.

[93] A. Kumar, “On bandlimited signal reconstruction from the distribution of unknown sampling locations,” *IEEE Trans. on Signal Processing*, vol. 63, no. 5, pp. 1259–1267, Mar. 2015, ISSN: 1053-587X. DOI: `10.1109/TSP.2015.2394248`.

[94] A. Kumar, “On bandlimited field estimation from samples recorded by a location-unaware mobile sensor,” *IEEE Trans. on Information Theory*, vol. 63, no. 4, pp. 2188–2200, Apr. 2017.

[95] S. Salgia and A. Kumar, “Bandlimited spatiotemporal field sampling with location and time unaware mobile sensors,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2018, pp. 4574–4578.

[96] M. Pai and A. Kumar, “Distribution learning of a random spatial field with a location-unaware mobile sensor,” in *Advances in Neural Information Processing Systems*, 2019, pp. 12 458–12 466.

[97] S. Azizi, D. Cochran, and J. McDonald, “On the preservation of bandlimitedness under non-affine time warping,” in *Proc. Int. Workshop on Sampling Theory and Applications*, 1999.

[98] J. Clark, M. Palmer, and P. Lawrence, “A transformation method for the reconstruction of functions from nonuniformly spaced samples,” *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 33, no. 5, pp. 1151–1165, 1985.

[99] K. Horiuchi, “Sampling principle for continuous signals with time-varying bands,” *Information and Control*, vol. 13, no. 1, pp. 53–61, 1968.

[100] D. Cochran and J. J. Clark, “On the sampling and reconstruction of time warped band-limited signals,” in *Proc. of the Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, Apr. 1990.

[101] X.-G. Xia and Z. Zhang, “On a conjecture on time-warped band-limited signals,” *IEEE Trans. on Signal Processing*, vol. 40, no. 1, pp. 252–254, 1992.

[102] A. Scholefield, B. Béjar Haro, and M. Vetterli, “Shape from bandwidth: the 2-D orthogonal projection case,” in *Proc. of the 42nd IEEE International Conf. on Acoustics, Speech and Signal Processing*, 2017.

[103] J. L. Horowitz, *Semiparametric and Nonparametric Methods in Econometrics*. Springer US, 2009, ISBN: 9780387928708.

[104] A. Wolodtschenko and T. Forner, “Prehistoric and early historic maps in Europe: conception of cd-atlas,” *e-Perimetron*, vol. 2, no. 2, pp. 114–116, 2007.

[105] N. G. Society, “Here be dragons,” *National Geographic*, 2020, accessed: 2021-07-26. [Online]. Available: `https://www.nationalgeographic.org/article/here-be-dragons/`.

[106] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, “Pixel2Mesh: generating 3D mesh models from single RGB images,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 52–67.

[107] L. Hejtmánek, I. Oravcová, J. Motỳl, J. Horáček, and I. Fajnerová, “Spatial knowledge impairment after gps guided navigation: eye-tracking study in a virtual town,” *International Journal of Human-Computer Studies*, vol. 116, pp. 15–24, 2018.

[108] L. Doherty, L. El Ghaoui, *et al.*, “Convex position estimation in wireless sensor networks,” in *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No. 01CH37213)*, IEEE, vol. 3, 2001, pp. 1655–1663.

[109] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, “Locating the nodes: cooperative localization in wireless sensor networks,” *IEEE Signal processing magazine*, vol. 22, no. 4, pp. 54–69, 2005.

[110] J. W. Lichtman, H. Pfister, and N. Shavit, “The big data challenges of connectomics,” *Nature neuroscience*, vol. 17, no. 11, pp. 1448–1454, 2014.

[111] T. F. Havel and K. Wüthrich, “An evaluation of the combined use of nuclear magnetic resonance and distance geometry for the determination of protein conformations in solution,” in *Nmr In Structural Biology: A Collection of Papers by Kurt Wüthrich*, World Scientific, 1995, pp. 305–318.

[112] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, K. Tunyasuvunakool, O. Ronneberger, R. Bates, A. Žídek, A. Bridgland, C. Meyer, S. A. A. Kohl, A. Potapenko, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, M. Steinegger, M. Pacholska, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, “High accuracy protein structure prediction using deep learning,” in *Fourteenth Critical Assessment of Techniques for Protein Structure Prediction*, 2020.

[113] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, "Highly accurate protein structure prediction with AlphaFold," *Nature*, pp. 1–11, 2021.

[114] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT press, 2005.

[115] M. Montemerlo and S. Thrun, *FastSLAM: A Scalable Method for the Simultaneous Localization and Mapping Problem in Robotics*, 1st. Springer Publishing Company, Incorporated, 2010.

[116] G. Younes, D. Asmar, E. Shammas, and J. Zelek, "Keyframe-based monocular slam: design, survey, and future directions," *Robotics and Autonomous Systems*, vol. 98, pp. 67–88, 2017.

[117] C. Evers and P. A. Naylor, "Acoustic SLAM," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 1484–1498, 2018.

[118] M. Kreković, I. Dokmanić, and M. Vetterli, "EchoSLAM: Simultaneous localization and mapping with acoustic echoes," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 11–15.

[119] M. Krekovic, I. Dokmanic, and M. Vetterli, "Look, no beacons! Optimal all-in-one EchoSLAM," 2016, Invited Paper. [Online]. Available: `http://infoscience.epfl.ch/record/221396`.

[120] A. J. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *null*, IEEE, 2003, p. 1403.

[121] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[122] S. Huang and G. Dissanayake, "A critique of current developments in simultaneous localization and mapping," *International Journal of Advanced Robotic Systems*, vol. 13, no. 5, p. 1 729 881 416 669 482, 2016.

[123] D. Zou, P. Tan, and W. Yu, "Collaborative visual SLAM for multiple agents: a brief survey," *Virtual Reality & Intelligent Hardware*, vol. 1, no. 5, pp. 461–482, 2019.

[124] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.

[125] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Second. Cambridge University Press, 2004, ISBN: 0521540518.

[126] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003, ISBN: 0387008934.

[127] C. Wu, “Towards linear-time incremental structure from motion,” in *2013 International Conference on 3D Vision-3DV 2013*, IEEE, 2013, pp. 127–134.

[128] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, “Unsupervised learning of depth and ego-motion from video,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1851–1858.

[129] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, “Differentiable monte carlo ray tracing through edge sampling,” *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, pp. 1–11, 2018.

[130] G. Loubet, N. Holzschuch, and W. Jakob, “Reparameterizing discontinuous integrands for differentiable rendering,” *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–14, 2019.

[131] J. Aloimonos, “Shape from texture,” *Biological cybernetics*, vol. 58, no. 5, pp. 345–360, 1988.

[132] M. Clerc and S. Mallat, “The texture gradient equation for recovering shape from texture,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 536–549, 2002. DOI: `10.1109/34.993560`.

[133] J. J. Gibson, “The perception of the visual world.,” 1950.

[134] J. Malik and R. Rosenholtz, “Computing local surface orientation and shape from texture for curved surfaces,” *International journal of computer vision*, vol. 23, no. 2, pp. 149–168, 1997.

[135] G. Elhami, A. J. Scholefield, and M. Vetterli, “Shape from bandwidth: central projection case,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 1808–1812.

[136] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, “Shape-from-shading: a survey,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 21, no. 8, pp. 690–706, 1999.

[137] B. K. Horn, “Shape from shading: a method for obtaining the shape of a smooth opaque object from one view,” 1970.

[138] H. Fan, H. Su, and L. J. Guibas, “A point set generation network for 3D object reconstruction from a single image,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 605–613.

[139] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry, “A papier-mâché approach to learning 3D surface generation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 216–224.

[140] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, “Occupancy networks: learning 3D reconstruction in function space,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4460–4470.

[141] A. Bahr, J. J. Leonard, and M. F. Fallon, “Cooperative localization for autonomous underwater vehicles,” *Int Jour Rob Res*, vol. 28, no. 6, pp. 714–728, 2009, ISSN: 02783649. DOI: `10.1177/0278364908100561`.

[142] G. Vallicrosa, P. Ridao, D. Ribas, and A. Palomer, “Active range-only beacon localization for AUV homing,” in *Int Conf Int Rob Sys (IROS)*, 2014, pp. 2286–2291, ISBN: 9781479969340. DOI: `10.1109/IROS.2014.6942871`.

[143] F. R. Fabresse, F. Caballero, L. Merino, and A. Ollero, “Active perception for 3D range-only simultaneous localization and mapping with UAVs,” in *Int Conf Unm Air Sys*, 2016, pp. 394–398, ISBN: 9781467393331. DOI: `10.1109/ICUAS.2016.7502639`.

[144] F. Dümbgen *et al.*, “Multi-modal probabilistic indoor localization on a smartphone,” in *Int Conf Ind Pos and Ind Navi (IPIN)*, 2019, ISBN: 9781728117881.

[145] A. Beck, P. Stoica, and J. Li, “Exact and approximate solutions of source localization problems,” *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1770–1778, 2008, ISSN: 1053-587X. DOI: `10.1109/TSP.2007.909342`.

[146] E. G. Larsson and D. Danev, “Accuracy comparison of ls and squared-range ls for source localization,” *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 916–923, 2009.

[147] J. J. Moré, “The Levenberg-Marquardt algorithm: implementation and theory,” in *Numerical analysis*, Springer, 1978, pp. 105–116.

[148] J. Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, B. D. Anderson, and P. N. Belhumeur, “A theory of network localization,” *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663–1678, 2006.

[149] I. Dokmanić, L. Daudet, and M. Vetterli, “How to localize ten microphones in one finger snap,” in *2014 22nd European Signal Processing Conference (EUSIPCO)*, IEEE, 2014, pp. 2275–2279.

[150] I. Dokmanić, R. Parhizkar, A. Walther, Y. M. Lu, and M. Vetterli, “Acoustic echoes reveal room shape,” *Proceedings of the National Academy of Sciences*, vol. 110, no. 30, pp. 12 186–12 191, 2013.

[151] M. Krekovic, I. Dokmanic, and M. Vetterli, “Omnidirectional bats, point-to-plane distances, and the price of uniqueness,” *2017 Ieee International Conference On Acoustics, Speech And Signal Processing (Icassp)*, International Conference on Acoustics Speech and Signal Processing ICASSP, pp. 5. 3261–3265, 2017. [Online]. Available: `http://infoscience.epfl.ch/record/221397`.

[152] L. Holm and C. Sander, “Protein structure comparison by alignment of distance matrices,” *Journal of molecular biology*, vol. 233, no. 1, pp. 123–138, 1993.

[153] A. Mead, “Review of the development of multidimensional scaling methods,” *Journal of the Royal Statistical Society: Series D (The Statistician)*, vol. 41, no. 1, pp. 27–39, 1992.

[154] A. Singer, “A remark on global positioning from local distances,” *Proceedings of the National Academy of Sciences*, vol. 105, no. 28, pp. 9507–9511, 2008.

[155] B. Berger, J. Kleinberg, and T. Leighton, “Reconstructing a three-dimensional model with arbitrary errors,” *Journal of the ACM (JACM)*, vol. 46, no. 2, pp. 212–235, 1999.

[156] L. Liberti, C. Lavor, N. Maculan, and A. Mucherino, “Euclidean distance geometry and applications,” *SIAM review*, vol. 56, no. 1, pp. 3–69, 2014.

[157] L. Liberti, G. Swirszcz, and C. Lavor, “Distance geometry on the sphere,” in *Japanese Conference on Discrete and Computational Geometry and Graphs*, Springer, 2015, pp. 204–215.

[158] L. Liberti *et al.*, “Distance geometry and data science,” *TOP: An Official Journal of the Spanish Society of Statistics and Operations Research*, pp. 1–69, 2019.

[159] W. S. Torgerson, “Multidimensional scaling: I. Theory and method,” *Psychometrika*, vol. 17, no. 4, pp. 401–419, 1952.

[160] M. A. Cox and T. F. Cox, “Multidimensional scaling,” in *Handbook of data visualization*, Springer, 2008, pp. 315–347.

[161] P. H. Schönemann, “On metric multidimensional unfolding,” *Psychometrika*, vol. 35, no. 3, pp. 349–366, 1970.

[162] A. Y. Alfakih, A. Khandani, and H. Wolkowicz, “Solving Euclidean distance matrix completion problems via semidefinite programming,” *Computational optimization and applications*, vol. 12, no. 1, pp. 13–30, 1999.

[163] R. Parhizkar, “Euclidean distance matrices: properties, algorithms and applications,” *School of Computer and Communication Sciences, Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland*, 2013.

[164] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, “Euclidean distance matrices: essential theory, algorithms, and applications,” *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 12–30, 2015.

[165] S. Gujarathi, C. Farrow, C. Glosser, L. Granlund, and P. Duxbury, “Ab-initio reconstruction of complex Euclidean networks in two dimensions,” *Physical Review E*, vol. 89, no. 5, p. 053 311, 2014.

[166] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, “Semidefinite programming based algorithms for sensor network localization,” *ACM Transactions on Sensor Networks (TOSN)*, vol. 2, no. 2, pp. 188–220, 2006.

[167] P. Furgale, C. H. Tong, T. D. Barfoot, and G. Sibley, “Continuous-time batch trajectory estimation using temporal basis functions,” *Int Jour Rob Res*, vol. 34, no. 14, pp. 1688–1710, 2015, ISSN: 17413176. DOI: `10.1177/0278364915585860`.

[168] T. Li, H. Chen, S. Sun, and J. M. Corchado, “Joint smoothing and tracking based on continuous-time target trajectory function fitting,” *IEEE Trans Aut Sci Eng*, vol. 16, no. 3, pp. 1476–1483, 2019, ISSN: 15455955. DOI: `10.1109/TASE.2018.2882641`.

[169] E. Mueggler, G. Gallego, and D. Scaramuzza, “Continuous-time trajectory estimation for event-based vision sensors,” *Rob: Sci Sys (RSS)*, 2015, ISSN: 2330765X. DOI: `10.15607/RSS.2015.XI.036`.

[170] D. Droeschel and S. Behnke, "Efficient continuous-time SLAM for 3D lidar-based online mapping," in *IEEE Int Conf Rob and Aut (ICRA)*, 2018, pp. 5000–5007, ISBN: 9781538630815. DOI: `10.1109/ICRA.2018.8461000`.

[171] R. Alexandru, T. Blu, and P. L. Dragotti, "Localising diffusion sources from samples taken along unknown parametric trajectories,"

[172] J. Zhou, T. Li, X. Wang, and L. Zheng, "Target tracking with equality/inequality constraints based on trajectory function of time," *IEEE Signal Processing Letters*, 2021.

[173] P. Tabaghi, I. Dokmanić, and M. Vetterli, "Kinetic Euclidean distance matrices," *IEEE Transactions on Signal Processing*, vol. 68, pp. 452–465, 2019.

[174] A. Mucherino and D. S. Gonçalves, "An approach to dynamical distance geometry," in *International Conference on Geometric Science of Information*, Springer, 2017, pp. 821–829.

[175] A. Mucherino, J. Omer, L. Hoyet, P. R. Giordano, and F. Multon, "An application-based characterization of dynamical distance geometry problems," *Optimization Letters*, vol. 14, no. 2, pp. 493–507, 2020.

[176] J. D. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 713–719.

[177] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *International journal of computer vision*, vol. 70, no. 1, pp. 77–90, 2006.

[178] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: a factorization method," *International journal of computer vision*, vol. 9, no. 2, pp. 137–154, 1992.

[179] P. Chen and D. Suter, "Recovering the missing components in a large noisy low-rank matrix: application to SFM," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 8, pp. 1051–1063, 2004.

[180] M. A. Davenport and J. Romberg, "An overview of low-rank matrix recovery from incomplete observations," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 608–622, 2016.

[181] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, p. 717, 2009.

[182] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, 2013, pp. 665–674.

[183] E. J. Candes and T. Tao, "The power of convex relaxation: near-optimal matrix completion," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2053–2080, May 2010, ISSN: 1557-9654. DOI: `10.1109/TIT.2010.2044061`.

[184] Y. Chen, "Incoherence-optimal matrix completion," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2909–2923, 2015.

[185] Y. Chen, S. Bhojanapalli, S. Sanghavi, and R. Ward, "Coherent matrix completion," in *International Conference on Machine Learning*, PMLR, 2014, pp. 674–682.

[186] B. Recht, "A simpler approach to matrix completion.," *Journal of Machine Learning Research*, vol. 12, no. 12, 2011.

[187] E. J. Candes and Y. Plan, "Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements," *IEEE Transactions on Information Theory*, vol. 57, no. 4, pp. 2342–2359, 2011.

[188] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.

[189] D. Gross, Y.-K. Liu, S. T. Flammia, S. Becker, and J. Eisert, "Quantum state tomography via compressed sensing," *Physical review letters*, vol. 105, no. 15, p. 150 401, 2010.

[190] D. Gross, "Recovering low-rank matrices from few coefficients in any basis," *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1548–1566, 2011.

[191] D. L. Pimentel-Alarcón, N. Boston, and R. D. Nowak, "A characterization of deterministic sampling patterns for low-rank matrix completion," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 623–636, 2016.

[192] S. Bhojanapalli and P. Jain, "Universal matrix completion," in *International Conference on Machine Learning*, PMLR, 2014, pp. 1881–1889.

[193] G. Liu, Q. Liu, X.-T. Yuan, and M. Wang, "Matrix completion with deterministic sampling: theories and methods," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[194] A. Singer and M. Cucuringu, "Uniqueness of low-rank matrix completion by rigidity theory," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 4, pp. 1621–1641, 2010.

[195] A. Shapiro, Y. Xie, and R. Zhang, "Matrix completion with deterministic pattern: a geometric perspective," *IEEE Transactions on Signal Processing*, vol. 67, no. 4, pp. 1088–1103, 2018.

[196] F. Király and R. Tomioka, "A combinatorial algebraic approach for the identifiability of low-rank matrix completion," *arXiv preprint arXiv:1206.6470*, 2012.

[197] D. I. Bernstein, G. Blekherman, and R. Sinn, "Typical and generic ranks in matrix completion," *Linear Algebra and its Applications*, vol. 585, pp. 71–104, 2020.

[198] R. Calderbank, S. Howard, and S. Jafarpour, "Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property," *IEEE journal of selected topics in signal processing*, vol. 4, no. 2, pp. 358–374, 2010.

[199] G. B. Dantzig, “Linear programming,” *Operations research*, vol. 50, no. 1, pp. 42–47, 2002.

[200] L. Vandenberghe and S. Boyd, “Semidefinite programming,” *SIAM review*, vol. 38, no. 1, pp. 49–95, 1996.

[201] A. M.-C. So and Y. Ye, “Theory of semidefinite programming for sensor network localization,” *Mathematical Programming*, vol. 109, no. 2-3, pp. 367–384, 2007.

[202] L. Lovász, “On the Shannon capacity of a graph,” *IEEE Transactions on Information theory*, vol. 25, no. 1, pp. 1–7, 1979.

[203] M. X. Goemans and D. P. Williamson, “Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming,” *Journal of the ACM (JACM)*, vol. 42, no. 6, pp. 1115–1145, 1995.

[204] Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, “Semidefinite relaxation of quadratic optimization problems,” *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 20–34, 2010.

[205] K. Adam, A. Scholefield, and M. Vetterli, “Multi-channel time encoding for improved reconstruction of bandlimited signals,” *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (Icassp)*, pp. 7963–7967, 2019. DOI: `10.1109/ICASSP.2019.8682361`. [Online]. Available: `http://infoscience.epfl.ch/record/265342`.

[206] J. E. Gentle, “Matrix algebra,” *Springer texts in statistics, Springer, New York, NY, doi*, vol. 10, pp. 978–, 2007.

[207] M. Marcus and H. Minc, “Permanents,” *The American Mathematical Monthly*, vol. 72, no. 6, pp. 577–591, 1965.

[208] M. Elad, *Sparse and redundant representations: from theory to applications in signal and image processing.* Springer Science & Business Media, 2010.

[209] Y. Chi, Y. M. Lu, and Y. Chen, “Nonconvex optimization meets low-rank matrix factorization: an overview,” *IEEE Transactions on Signal Processing*, vol. 67, no. 20, pp. 5239–5269, 2019.

[210] J. van der Woude, “A straightforward proof of the polynomial factorization of a positive semi-definite polynomialmatrix,” *Linear Algebra and its Applications*, vol. 456, pp. 214–220, 2014.

[211] L. Ephremidze, “An elementary proof of the polynomial matrix spectral factorization theorem,” *Proceedings of the Royal Society of Edinburgh Section A: Mathematics*, vol. 144, no. 4, pp. 747–751, 2014.

[212] D. Ž. Djoković, “Hermitian matrices over polynomial rings,” *Journal of Algebra*, vol. 43, no. 2, pp. 359–374, 1976.

[213] A. C. Ran and L. Rodman, “Factorization of matrix polynomials with symmetries,” *SIAM Journal on Matrix Analysis and Applications*, vol. 15, no. 3, pp. 845–864, 1994.

[214] M. Itskov, *Tensor algebra and tensor analysis for engineers*. Springer, 2007, Available: `https://pdf-drive.com/pdf/Mikhail20Itskov20-20Tensor20algebra20and20tensor20analysis20for20engineers-Springer2028200729.pdf`.

[215] A. J. Weir, *Lebesgue integration and measure*. Cambridge University Press, 1973, vol. 1.

[216] M. Pacholska, "Sampling at unknown locations: existence, uniqueness, solution methods and algorithms," M.S. thesis, University of Warsaw, 2016.

[217] M. Pacholska, B. Béjar Haro, A. Scholefield, and M. Vetterli, "Sampling at unknown locations, with an application in surface retrieval," in *Proc. of the 12th Int. Conf. on Sampling Theory and its Applications (SampTA)*, 2017, pp. 364–368.

[218] G. Elhami, M. Pacholska, B. B. Haro, M. Vetterli, and A. Scholefield, "Sampling at unknown locations: uniqueness and reconstruction under constraints," *IEEE Transactions on Signal Processing*, vol. 66, no. 22, pp. 5862–5874, 2018.

[219] W. Stein, *Elementary number theory: primes, congruences, and secrets: a computational approach*. Springer Science & Business Media, 2008, Accessed: 2021-05-30. [Online]. Available: `https://wstein.org/ent/ent.pdf`.

[220] P. DuChateau, *Introduction to the theory of distributions*, Accessed: 2021-03-30, 2004. [Online]. Available: `https://www.math.colostate.edu/~pauld/M545/6Distrib.pdf`.

[221] T. Judson, *Abstract algebra: theory and applications*. Stephen F. Austin State University, 2020, Accessed: 2021-05-30. [Online]. Available: `http://abstract.ups.edu/`.

[222] J. F. Hauer, C. J. Demeure, and L. L. Scharf, "Initial results in Prony analysis of power system response signals," *IEEE Trans. on Power Systems*, vol. 5, no. 1, pp. 80–89, 1990.

[223] J. T. VanderPlas, "Understanding the Lomb–Scargle periodogram," *The Astrophysical Journal Supplement Series*, vol. 236, no. 1, p. 16, 2018.

[224] M. Rubsamen and A. B. Gershman, "Direction-of-arrival estimation for nonuniform sensor arrays: from manifold separation to fourier domain music methods," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 588–599, 2008.

[225] M. Ben-Or and P. Tiwari, "A deterministic algorithm for sparse multivariate polynomial interpolation," in *Proceedings of the twentieth annual ACM symposium on Theory of computing*, 1988, pp. 301–309.

[226] J. Berent, P. L. Dragotti, and T. Blu, "Sampling piecewise sinusoidal signals with finite rate of innovation methods," *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 613–625, 2009.

[227] T. Peter, G. Plonka, and D. Roşca, "Representation of sparse Legendre expansions," *Journal of Symbolic Computation*, vol. 50, pp. 159–169, 2013.

[228] D. Potts and M. Tasche, "Sparse polynomial interpolation in Chebyshev bases," *Linear Algebra and its Applications*, vol. 441, pp. 61–87, 2014.

[229] G. Baechler, A. Scholefield, L. Baboulaz, and M. Vetterli, “Sampling and exact reconstruction of pulses with variable width,” *IEEE Transactions on Signal Processing*, vol. 65, no. 10, pp. 2629–2644, 2017.

[230] B. Mourrain, “Polynomial–exponential decomposition from moments,” *Foundations of Computational Mathematics*, vol. 18, no. 6, pp. 1435–1492, 2018.

[231] T. Peter and G. Plonka, “A generalized Prony method for reconstruction of sparse sums of eigenfunctions of linear operators,” *Inverse Problems*, vol. 29, no. 2, p. 025 001, 2013.

[232] K. Stampfer and G. Plonka, “The generalized operator based Prony method,” *Constructive Approximation*, pp. 1–36, 2020.

[233] S. Senay, L. F. Chaparro, and L. Durak, “Reconstruction of nonuniformly sampled time-limited signals using prolate spheroidal wave functions,” *Signal Processing*, vol. 89, no. 12, pp. 2585–2595, 2009.

[234] L. Coluccio, A. Eisinberg, and G. Fedele, “A Prony-like method for non-uniform sampling,” *Signal Processing*, vol. 87, no. 10, pp. 2484–2490, 2007.

[235] I. Domanov and L. De Lathauwer, “Generic uniqueness of a structured matrix factorization and applications in blind source separation,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 701–711, 2016.

[236] S. D. Casey and B. M. Sadler, “Modifications of the Euclidean algorithm for isolating periodicities from a sparse set of noisy measurements,” *IEEE Transactions on Signal Processing*, vol. 44, no. 9, pp. 2260–2272, 1996.

[237] A. G. Khovanskij, *Fewnomials. Translated from the Russian by Smilka Zdravkovska*, English. Providence, RI: American Mathematical Society, 1991, vol. 88, pp. viii + 139, ISBN: 0-8218-4547-0.

[238] M.-F. R. Saugata Basu Richard Pollack, *Algorithms in Real Algebraic Geometry*, 2nd ed, ser. Algorithms and Computation in Mathematics 10. Springer Berlin Heidelberg, 2003, ISBN: 978-3-540-33099-8.

[239] M. Pacholska, K. Adam, A. Scholefield, and M. Vetterli, *Matrix recovery from bilinear and quadratic measurements*, 2020. arXiv: `2001.04933 [eess.SP]`.

[240] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky, “Rank-sparsity incoherence for matrix decomposition,” *SIAM Journal on Optimization*, vol. 21, no. 2, pp. 572–596, 2011.

[241] C. F. Van Loan, “The ubiquitous kronecker product,” *Journal of computational and applied mathematics*, vol. 123, no. 1-2, pp. 85–100, 2000.

[242] M. Pacholska, F. Duembgen, and A. Scholefield, “Relax and recover: guaranteed range-only continuous localization,” *IEEE Robotics and Automation Letters*, Feb. 2020, ISSN: 2377-3774. DOI: `10.1109/LRA.2020.2970952`.

[243] K. Adam, A. Scholefield, and M. Vetterli, "Encoding and decoding mixed bandlimited signals using spiking integrate-and-fire neurons," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 9264–9268.

[244] T. Delbrück, B. Linares-Barranco, E. Culurciello, and C. Posch, "Activity-driven, event-based vision sensors," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, IEEE, 2010, pp. 2426–2429.

[245] P. Jain and I. S. Dhillon, "Provable inductive matrix completion," *arXiv preprint arXiv:1306.0626*, 2013.

[246] T. T. Cai, A. Zhang, *et al.*, "ROP: matrix recovery via rank-one projections," *The Annals of Statistics*, vol. 43, no. 1, pp. 102–138, 2015.

[247] M. Lin and J. Ye, "A non-convex one-pass framework for generalized factorization machine and rank-one matrix sensing," in *Advances in Neural Information Processing Systems*, 2016, pp. 1633–1641.

[248] R. Kueng, H. Rauhut, and U. Terstiege, "Low rank matrix recovery from rank one measurements," *Applied and Computational Harmonic Analysis*, vol. 42, no. 1, pp. 88–116, 2017.

[249] Y. Chen, Y. Chi, and A. J. Goldsmith, "Exact and stable covariance estimation from quadratic sampling via convex programming," *IEEE Transactions on Information Theory*, vol. 61, no. 7, pp. 4034–4059, 2015.

[250] K. Zhong, P. Jain, and I. S. Dhillon, "Efficient matrix sensing using rank-1 Gaussian measurements," in *International conference on algorithmic learning theory*, Springer, 2015, pp. 3–18.

[251] M. Asgari and A. Khosravi, "Frames and bases in tensor product of Hilbert spaces," *Intern. Math. J.*, vol. 4, no. 6, pp. 527–538, 2003.

[252] A. Khosravi and B. Khosravi, "Frames and bases in tensor products of Hilbert spaces and Hilbert C*-modules," *Proceedings Mathematical Sciences*, vol. 117, no. 1, pp. 1–12, 2007.

[253] A. Bourouihiya, "The tensor product of frames," *Sampling theory in signal and Image processing*, vol. 7, no. 1, pp. 65–77, 2008.

[254] J. B. Kruskal, "Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear algebra and its applications*, vol. 18, no. 2, pp. 95–138, 1977.

[255] J. Buczyński, T. Januszkiewicz, J. Jelisiejew, and M. Michałek, "Constructions of k-regular maps using finite local schemes," *Journal of the European Mathematical Society*, vol. 21, no. 6, pp. 1775–1808, 2019.

[256] P. Blagojević, W. Lück, and G. Ziegler, "On highly regular embeddings," *Transactions of the American Mathematical Society*, vol. 368, no. 4, pp. 2891–2912, 2016.

[257] I. Singer, "Best approximation in normed linear spaces by elements of linear subspaces," *New York*, 1970.

[258] A. Papoulis and H. Saunders, "Probability, random variables and stochastic processes," 1989.

[259] Z. Wang, M.-J. Lai, Z. Lu, W. Fan, H. Davulcu, and J. Ye, "Orthogonal rank-one matrix pursuit for low rank matrix completion," *SIAM Journal on Scientific Computing*, vol. 37, no. 1, A488–A514, 2015.

[260] K. Adam, A. Scholefield, and M. Vetterli, "Asynchrony increases efficiency: time encoding of videos and low-rank signals," *arXiv preprint arXiv:2104.14511*, 2021.

[261] M. Kreković, I. Dokmanić, and M. Vetterli, "Shapes from echoes: uniqueness from point-to-plane distance matrices," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2480–2498, 2020.

[262] P. Tabaghi and I. Dokmanić, "Real polynomial gram matrices without real spectral factors," *arXiv preprint arXiv:1903.04085*, 2019.

[263] P. Tabaghi, I. Dokmanić, and M. Vetterli, "On the move: localization with kinetic Euclidean distance matrices," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 4893–4897.

[264] L. Ephremidze, I. Spitkovsky, and E. Lagvilava, "Rank-deficient spectral factorization and wavelets completion problem," *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 13, no. 03, p. 1 550 013, 2015.

[265] J. Djugash, B. Hamner, and S. Roth, "Navigating with ranging radios: five data sets with ground truth," *Journal of Field Robotics*, vol. 26, no. 9, pp. 689–695, 2009, ISSN: 14746670. DOI: `10.1002/rob`.

[266] P. Virtanen *et al.*, "SciPy 1.0–fundamental algorithms for scientific computing in Python," *arXiv preprint arXiv:1907.10121*, arXiv:1907.10121, 2019. arXiv: `1907.10121 [cs.MS]`.

[267] D. M. Rosen, C. DuHadway, and J. J. Leonard, "A convex relaxation for approximate global optimization in simultaneous localization and mapping," in *IEEE Int Conf Rob and Aut (ICRA)*, IEEE, 2015, pp. 5822–5829.

[268] M. Congedo, R. Phlypo, and D.-T. Pham, "Approximate joint singular value decomposition of an asymmetric rectangular matrix set," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 415–424, 2010.

[269] A. Yeredor, "Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation," *IEEE Transactions on signal processing*, vol. 50, no. 7, pp. 1545–1553, 2002.

[270] G. Baechler, A. Latty, M. Pacholska, M. Vetterli, and A. Scholefield, "Shedding light on 19th century spectra by analyzing Lippmann photography," *Proceedings of the National Academy of Sciences*, vol. 118, no. 17, 2021.

[271] A. Nobel, *Full text of Alfred Nobel's will*, 1895. [Online]. Available: `http://nobelprize.org/alfred%5C_nobel/will/will-full.html`.

[272] N. Foundation, *The Nobel Prize in Physics*, access: 2021-07-16, 2021. [Online]. Available: `http://nobelprize.org/nobel_prizes/physics/laureates/1908/index.html`.

[273] P. Connes, "Silver salts and standing waves: the history of interference colour photography," *Journal of Optics*, vol. 18, no. 4, p. 147, 1987.

[274] D. J. Mitchell, "Reflecting nature: chemistry and comprehensibility in Gabriel Lippmann's "physical" method of photographing colours," *Notes and Records of the Royal Society*, vol. 64, no. 4, pp. 319–337, 2010.

[275] I. Newton, *Opticks*. Prabhat Prakashan, 1730.

[276] T. Young, "The Bakerian lecture. Experiments and calculations relative to physical optics," in *Abstracts of the Papers Printed in the Philosophical Transactions of the Royal Society of London*, The Royal Society London, 1832, pp. 131–132.

[277] R. C. Dougal, C. A. Greated, and A. E. Marson, "Then and now: James Clerk Maxwell and colour," *Optics & Laser Technology*, vol. 38, no. 4-6, pp. 210–218, 2006.

[278] E. Becquerel, « De l'image photographique colorée du spectre solaire », *Comptes Rendus Hebdomadaires à l'Académie Des Sciences*, t. 26, p. 181-184, 1848.

[279] ——, « De l'image photochromatique du spectre solaire », *Annales de chimie et de physique*, p. 447-474, 1849.

[280] J. C. Maxwell, "XVII—I. Experiments on colour, as perceived by the eye, with remarks on colour-blindness," *Earth and Environmental Science Transactions of the Royal Society of Edinburgh*, vol. 21, no. 2, pp. 275–298, 1857.

[281] W. Zenker, *Lehrbuch der Photochromie*. W. Zenker, 1868.

[282] O. Wiener, „Stehende Lichtwellen und die Schwingungsrichtung polarisirten Lichtes", *Annalen der Physik und Chemie*, Jg. 40, S. 203–243, 1890.

[283] H. I. Bjelkhagen, "Lippmann photography and color holography: 2-d and 3-d color imaging techniques," in *Color and Imaging Conference*, Society for Imaging Science and Technology, vol. 2001, 2001, pp. 120–125.

[284] A. Lumière et L. Lumière, « Note sur la photographie des couleurs », *Revue suisse de photographie*, t. 5, p. 150-153, 1893.

[285] ——, « Sur les procédés pour la photographie des couleurs d'après la méthode de M. Lippmann », *Bulletin de la Société française de Photographie*, t. 9, p. 249-251, 1893.

[286] G. Lippmann, "La photographie des couleurs," *Comptes Rendus Hebdomadaires à l'Académie Des Sciences*, vol. 112, pp. 274–275, 1891.

[287] ——, "Sur la théorie de la photographie des couleurs simples et composées par la méthode interférentielle," *Journal de Physique Théorique Appliquée*, vol. 3, no. 1, pp. 97–107, 1894. doi: `10.1051/jphystap:01894003009700`.

[288] G. Saxby and S. Zacharovas, *Practical Holography, Fourth Edition.* CRC Press, 2015, ISBN: 9781482251586.

[289] L. Rayleigh, “X. On the electromagnetic theory of light,” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 12, no. 73, pp. 81–101, 1881.

[290] O. Wiener, „Ursache und Beseitigung eines Fehlers bei der Lippmann’schen Farbenphotographie, zugleich ein Beitrag zu ihrer Theorie“, *Annalen der Physik*, Jg. 305, Nr. 10, S. 488–530, 1899.

[291] E. ROTHÉ, « Photographies en couleurs obtenues par la méthode interférentielle sans miroir de mercure », *Comptes Rendus Hebdomadaires à l’Académie Des Sciences*, t. 139, p. 565-567, 1904.

[292] H. Ives, *An Experimental Study of the Lippmann Color Photograph.* John Hopkins University, 1908.

[293] R. Neuhauss, *Die Farbenphotographie nach Lippmann’s Verfahren: neue Untersuchungen und Ergebnisse.* Knapp, 1898.

[294] H. Lehmann, *Beiträge zur Theorie und Praxis der direkten Farbenphotographie mittels stehender Lichtwellen nach Lippmanns Methode.* Trömer, 1906.

[295] S. y Cajal, “Estructura de las imágenes fotocrómicas de G. Lippmann,” *Revista de la real Academia de Ciencias Exactas, FÃsicas y Naturales de Madrid*, vol. 4, pp. 386–428, 1906.

[296] S. Ramón y Cajal, “Chromomicrophotographie par la méthode interférentielle,” *La Photographie des Couleurs*, vol. 2, no. 7, pp. 97–101, 1907.

[297] L. Mannes and L. Godowsky, “The Kodachrome process for amateur cinematography in natural colors,” *Journal of the Society of Motion Picture Engineers*, vol. 25, no. 1, pp. 65–68, 1935.

[298] *Last Kodachrome roll processed in Parsons*, accessed: 2020-07-16, 2010. [Online]. Available: `https://web.archive.org/web/20140408044418/http://www.kansas.com/2010/07/14/1403115/last-kodachrome-roll-processed.html`.

[299] D. Gabor, “A new microscopic principle,” *Nature*, vol. 161, no. 4098, pp. 777–778, May 1948, ISSN: 1476-4687. DOI: `10.1038/161777a0`. [Online]. Available: `https://doi.org/10.1038/161777a0`.

[300] ——, “Microscopy by reconstructed wave-fronts,” *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, vol. 197, no. 1051, pp. 454–487, 1949.

[301] N. Foundation, *The Nobel Prize in Physics*, access: 2021-07-16, 2021. [Online]. Available: `http://nobelprize.org/nobel_prizes/physics/laureates/1971/index.html`.

[302] N. J. Phillips, H. Heyworth, and T. Hare, “On Lippmann’s photography,” *Journal of Photographic Science*, vol. 32, no. 5, pp. 158–169, 1984.

[303] N. Phillips and R. V. der Werf, “The creation of effective Lippmann layers in ultrafine grain silver halide materials using non-laser sources,” *Journal of Photographic Science*, vol. 33, no. 5, pp. 22–28, 1985.

[304] J.-M. R. Fournier, “Investigation on Lippmann photographs: materials, processes, and color rendition,” in *SPIE 2176, Practical Holography VIII*, 1994, pp. 144–152. DOI: `10.1117/12.172627`.

[305] J.-M. R. Fournier, R. Benjamin, R. Alexander, L. Paul, L. Burnett, E. Sarah, and E. Stamper, “Recent developments in Lippmann photography,” in *SPIE 3358, Sixth International Symposium on Display Holography*, 1998, pp. 95–102.

[306] H. Nareid, “A review of the Lippmann colour process,” *The Journal of Photographic Science*, vol. 36, no. 4, pp. 140–147, 1988. DOI: `10.1080/00223638.1988.11736992`.

[307] H. Nareid and H. M. Pedersen, “Modeling of the Lippmann color process,” *JOSA A*, vol. 8, no. 2, pp. 257–265, 1991.

[308] H. I. Bjelkhagen, “Lippman photographs recorded in DuPont color photopolymer material,” in *Practical Holography XI and Holographic Materials III*, International Society for Optics and Photonics, vol. 3011, 1997, pp. 358–367.

[309] ——, “Lippmann photography: its history and recent development,” *The PhotoHistorian*, pp. 11–19, 2003.

[310] ——, “Super-realistic-looking images based on colour holography and Lippmann photography,” *The International Society for Optical Engineering*, 2002. DOI: `10.1117/12.474952`.

[311] H. I. Bjelkhagen and D. Brotherton-Ratcliffe, *Ultra-Realistic Imaging: Advanced Techniques in Analogue and Digital Colour Holography*. CRC Press, 2013, ISBN: 9781439828007.

[312] W. R. Alschuler, “Lippmann photography and the glory of frozen light: eternal photographic color real and false,” in *Proceedings of the 31st Applied Image Pattern Recognition Workshop on From Color to Hyperspectral: Advancements in Spectral Imagery Exploitation*, Washington, DC, USA: IEEE Computer Society, 2002, pp. 3–12.

[313] G. Baechler, “Sampling the multiple facets of light,” p. 183, 2018. DOI: `10.5075/epfl-thesis-8795`. [Online]. Available: `http://infoscience.epfl.ch/record/257308`.

[314] J. Peatross and M. Ware, *Physics of Light and Optics*. Optical Society of America, 2015, Accessed: 2020-07-14. [Online]. Available: `https://optics.byu.edu/textbook`.

[315] F. Hurter and V. C. Driffield, “Photochemical investigations and a new method of determination of the sensitiveness of photographic plates,” *Journal of the Society of the Chemical Industry*, vol. 9, p. 76, 1890.

[316] G. Phipps and C. Robertson, "Characteristics of holograms recorded on Agfa 10E75 plates and processed in D6X developer," Sandia Labs., Albuquerque, NM (USA), Tech. Rep., 1977.

[317] H. Feichtinger and K. Gröchenig, "Theory and practice of irregular sampling," in *Wavelets: Mathematics and Applications*, Jan. 1994.

[318] C. Lawson and R. Hanson, *Solving Least Squares Problems.* Society for Industrial and Applied Mathematics, 1995. DOI: `10.1137/1.9781611971217`.

[319] A. Scholefield, M. Vetterli, Y. Bellouard, M. Pacholska, G. Baechler, and A. Latty, "Multi-spectral image priniting method," Tech. Rep., 2019.

[320] S. Benedetto, V. Drai-Zerbib, M. Pedrotti, G. Tissier, and T. Baccino, "E-readers and visual fatigue," *PloS one*, vol. 8, no. 12, e83676, 2013.

[321] M. Gugole, O. Olsson, S. Rossi, M. P. Jonsson, and A. Dahlin, "Electrochromic inorganic nanostructures with high chromaticity and superior brightness," *Nano Letters*, 2021.

[322] W. V. Winkle. "The death of disk? HDDs still have an important role to play." (2019), [Online]. Available: `https://venturebeat.com/2019/09/02/the-death-of-disk-hdds-still-have-an-important-role-to-play/` (visited on 06/30/2020).

[323] M. Lantz, "Why the future of data storage is (still) magnetic tape," *IEEE Spectrum: Technology, Engineering, and Science News, IEEE Spectrum*, 2018.

[324] A. Chatzieleftheriou, I. Stefanovici, D. Narayanan, B. Thomsen, and A. Rowstron, "Could cloud storage be disrupted in the next decade?" In *12th {USENIX} Workshop on Hot Topics in Storage and File Systems (HotStorage 20)*, 2020.

[325] S. Furrer, M. A. Lantz, P. Reininger, A. Pantazi, H. E. Rothuizen, R. D. Cideciyan, G. Cherubini, W. Haeberle, E. Eleftheriou, J. Tachibana, *et al.*, "201 Gb/in^2 recording areal density on sputtered magnetic tape," *IEEE Transactions on Magnetics*, vol. 54, no. 2, pp. 1–8, 2017.

[326] J. Bornholt, R. Lopez, D. M. Carmean, L. Ceze, G. Seelig, and K. Strauss, "A DNA-based archival storage system," in *Proceedings of the Twenty-First International Conference on Architectural Support for Programming Languages and Operating Systems*, 2016, pp. 637–649.

[327] P. Anderson, R. Black, A. Cerkauskaite, A. Chatzieleftheriou, J. Clegg, C. Dainty, R. Diaconu, R. Drevinskas, A. Donnelly, A. L. Gaunt, *et al.*, "Glass: a new media for a new era?" In *10th {USENIX} Workshop on Hot Topics in Storage and File Systems (HotStorage 18)*, 2018.

[328] L. Hesselink, S. S. Orlov, and M. C. Bashaw, "Holographic data storage systems," *Proceedings of the IEEE*, vol. 92, no. 8, pp. 1231–1280, 2004.

[329] E. Glezer, M. Milosavljevic, L. Huang, R. Finlay, T.-H. Her, J. P. Callan, and E. Mazur, "Three-dimensional optical storage inside transparent materials," *Optics letters*, vol. 21, no. 24, pp. 2023–2025, 1996.

[330] J. Zhang, A. Čerkauskaitė, R. Drevinskas, A. Patel, M. Beresna, and P. G. Kazansky, "Eternal 5D data storage by ultrafast laser writing in glass," in *Laser-based Micro- and Nanoprocessing X*, International Society for Optics and Photonics, vol. 9736, 2016, 97360U.

[331] M. Born and E. Wolf, *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light.* Elsevier, 2013.

[332] L. N. Acquaroli, "Matrix method for thin film optics," *arXiv preprint arXiv:1809.07708*, 2018.

[333] H. Nareid and H. M. Pedersen, "Modeling of the Lippmann color process," *JOSA A*, vol. 8, no. 2, pp. 257–265, 1991.

[334] H. Kogelnik, "Coupled wave theory for thick hologram gratings," *The Bell System Technical Journal*, vol. 48, no. 9, pp. 2909–2947, Nov. 1969, ISSN: 0005-8580. DOI: `10.1002/j.1538-7305.1969.tb01198.x`.

[335] J. E. Sipe, L. Poladian, and C. M. de Sterke, "Propagation through nonuniform grating structures," *J. Opt. Soc. Am. A*, vol. 11, no. 4, pp. 1307–1320, Apr. 1994. DOI: `10.1364/JOSAA.11.001307`. [Online]. Available: `http://josaa.osa.org/abstract.cfm?URI=josaa-11-4-1307`.

[336] G. D. Landry and T. A. Maldonado, "Gaussian beam transmission and reflection from a general anisotropic multilayer structure," *Applied optics*, vol. 35, no. 30, pp. 5870–5879, 1996.

[337] Y. Antar and W. Boerner, "Reflection and refraction of a Gaussian beam at a planar dielectric interface," in *1974 Antennas and Propagation Society International Symposium*, IEEE, vol. 12, 1974, pp. 105–107.

[338] V. M. Serdyuk and J. A. Titovitsky, "A simple analytic approximation for the refracted field at Gaussian beam incidence upon a boundary of absorbing medium," *Journal of Electromagnetic Analysis and Applications*, vol. 2, no. 11, p. 640, 2010.

Curriculum Vitae

Michalina Pacholska

Education

October 2016	PhD in Computer Science at **EPFL** (École Polytechnique Fédérale de Lausanne) Thesis: Sampling Geometry and Colour
July 2016	Masters in Mathematics at **University of Warsaw** Grade: 5.0/5.0 Thesis: Sampling at unknown locations: existence, uniqueness, solution methods and algorithms
2014/2015	Exchange year at **EPFL** I've done a project on super-resolution for light microscopy.
July 2014	Bachelor Degree in Mathematics at College of Inter-Faculty Individual Studies in Mathematics and Natural Sciences **University of Warsaw** Grade: 4.5/5.0

Work Experience

Autumn 2018	Research Intern at **DeepMind** I have worked on protein structure prediction with neural networks in the early stages of AlphaFold2 [1]. I have used supervised learning approach in TensorFlow.
Summer 2016	Software Engineering Intern at **Google** I have build a C++ pipeline for preprocessing large amount of internal data for later analysis. The data to process was reduced from 2TB per day to a few GB per day.
2013–2014	Mathematics and programming teacher at **ICMCM** (Interdisciplinary Center for Mathematical and Computational Modeling) I was teaching high school students especially interested in mathematics.

Other Activities

2011–2015	Teacher & Organizer of Summer Science School Volunteering on a mathematics, physics and computer science focused camp for high school students.
2015–current	Volunteer & Designer for Helvetic Coding Contest — Switzerland's biggest programming contest.

Publications

[1] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, "Highly accurate protein structure prediction with AlphaFold," *Nature*, pp. 1–11, 2021.

[2] G. Baechler, A. Latty, M. Pacholska, M. Vetterli, and A. Scholefield, "Shedding light on 19th century spectra by analyzing Lippmann photography," *Proceedings of the National Academy of Sciences*, vol. 118, no. 17, 2021.

[3] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, K. Tunyasuvunakool, O. Ronneberger, R. Bates, A. Žídek, A. Bridgland, C. Meyer, S. A. A. Kohl, A. Potapenko, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, M. Steinegger, M. Pacholska, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, "High accuracy protein structure prediction using deep learning," in *Fourteenth Critical Assessment of Techniques for Protein Structure Prediction*, 2020.

[4] M. Pacholska, K. Adam, A. Scholefield, and M. Vetterli, *Matrix recovery from bilinear and quadratic measurements*, 2020. arXiv: `2001.04933 [eess.SP]`.

[5] M. Pacholska, F. Duembgen, and A. Scholefield, "Relax and recover: guaranteed range-only continuous localization," *IEEE Robotics and Automation Letters*, Feb. 2020, ISSN: 2377-3774. DOI: `10.1109/LRA.2020.2970952`.

[6] G. Elhami, M. Pacholska, B. B. Haro, M. Vetterli, and A. Scholefield, "Sampling at unknown locations: uniqueness and reconstruction under constraints," *IEEE Transactions on Signal Processing*, vol. 66, no. 22, pp. 5862–5874, 2018.

[7] M. Pacholska, B. Béjar Haro, A. Scholefield, and M. Vetterli, "Sampling at unknown locations, with an application in surface retrieval," in *Proc. of the 12th Int. Conf. on Sampling Theory and its Applications (SampTA)*, Jul. 2017, pp. 364–368.

Patents

[1] A. Scholefield, M. Vetterli, Y. Bellouard, M. Pacholska, G. Baechler, and A. Latty, *Multi-spectral image priniting method*, US Patent App. 16/757,232, Sep. 2020.